Journal of Machine Learning Research 24 (2023) 1-40 Submitted 12/22; Revised 5/23; Published 6/23

Unbiased Multilevel Monte Carlo Methods for Intractable Distributions: MLMC Meets MCMC

Tianze Wang TIANZE.WANG@RUTGERS.EDU
Department of Statistics
Rutgers University, Piscataway, NJ 08854

Guanyang Wang GUANYANG.WANG@RUTGERS.EDU
Department of Statistics
Rutgers University, Piscataway, NJ 08854

Editor: Pierre Alquier

Abstract

Constructing unbiased estimators from Markov chain Monte Carlo (MCMC) outputs is a difficult problem that has recently received a lot of attention in the statistics and machine learning communities. However, the current unbiased MCMC framework only works when the quantity of interest is an expectation, which excludes many practical applications. In this paper, we propose a general method for constructing unbiased estimators for functions of expectations and extend it to construct unbiased estimators for nested expectations. Our approach combines and generalizes the unbiased MCMC and Multilevel Monte Carlo (MLMC) methods. In contrast to traditional sequential methods, our estimator can be implemented on parallel processors. We show that our estimator has a finite variance and computational complexity and can achieve ε-accuracy within the optimal $O(1/\varepsilon^2)$ computational cost under mild conditions. Numerical experiments confirm our theoretical findings and demonstrate the benefits of unbiased estimators in the massively parallel regime.

Keywords: unbiased estimator, function of expectation, parallel computation, nested expectation, coupling

1. Introduction

Monte Carlo methods generate unbiased estimators for the expectation of a distribution. In practice, however, it may be impractical to sample from the underlying distribution and the quantity of interest may not be an expectation. In the context of statistical inference, it is very common that estimation problems can be represented as estimating a quantity of the form $\mathcal{T}(\pi)$, where π is one or a group of distributions and $\mathcal{T}$ is a functional of π. We begin by considering several motivating examples to gain a deeper understanding of the different forms that $\mathcal{T}(\pi)$ might take.

Example 1 (Integration) *Let π be a probability distribution and f a π-integrable function. The problem of estimating $\mathbb{E}_\pi[f]$ can be viewed as estimating $\mathcal{T}(\pi)$ where $\mathcal{T}$ is the integral operator: $\mathcal{T}(\pi) := \int f(x)\pi(dx)$.*

Example 2 (Nested Monte Carlo) *Let π be a probability distribution, and suppose the quantity of our interest has the form $\mathcal{T}(\pi) := \mathbb{E}_\pi[\lambda]$, where λ is itself intractable. The in-*

tractable function λ may take the form $\lambda(x) := f(x, \gamma(x))$, where $\gamma(x) = \mathbb{E}_{y\sim p(y|x)}[\phi(x, y)]$ is a conditional expectation. One concrete example is the two-stage optimal stopping problem, where $\gamma(x) = \max\{x, \mathbb{E}[y|x]\}$. Estimating the nested expectation is known as a challenging problem in Monte Carlo methods due to its involved structure (Rainforth et al., 2018).

Example 3 (Ratios of normalizing constants) *Suppose $\pi_1(x) = f_1(x)/Z_1$ and $\pi_2(x) = f_2(x)/Z_2$ are two probability densities with common support. We assume f_1 and f_2 can be easily evaluated, but the normalizing constants Z_1 and Z_2 are computationally intractable. Consider the task of estimating the ratio of normalizing constants, i.e., Z_1/Z_2, standard calculation shows $Z_1/Z_2 = \mathbb{E}_{\pi_2}[f_1]/\mathbb{E}_{\pi_1}[f_2]$. The problem can be viewed as estimating $\mathcal{T}(\pi)$ by choosing π as the product measure $\pi_1 \times \pi_2$, and $\mathcal{T}(\pi) := \mathbb{E}_{\pi_2}[f_1]/\mathbb{E}_{\pi_1}[f_2]$. The problem finds statistical and physics applications, including hypothesis testing, Bayesian inference, and estimating free energy differences. We refer the readers to Meng and Wong (1996) for other applications.*

Example 4 (Quantile estimation) *Let π be a probability distribution with cumulative distribution function F_π and q a constant in $(0, 1)$. Estimating the q-th quantile of π can be formulated as estimating $\mathcal{T}(\pi)$ where $\mathcal{T}(\pi) := \inf_v\{F_\pi(v) \geq q\}$. Quantile estimation problem has applications in statistics, economics, and other fields. We refer the readers to Koenker and Hallock (2001); Takeuchi et al. (2006); Romano et al. (2019) for more discussions, and Doss et al. (2014) for an MCMC-based method.*

In all the examples above, the distribution π can be intractable. In some cases, such as Example 1 and 2, the quantity of interest is an expectation under π, although the function inside the expectation may or may not be intractable. In other cases, including Example 3 and 4, $\mathcal{T}$ is a functional of π, but not an expectation.

Throughout this paper, we focus on designing unbiased estimators of $\mathcal{T}(\pi)$ assuming one can only access outputs from some MCMC algorithm that leaves π as stationary distribution. Unbiased estimators are of particular interest because they can help users save computation time in a parallel implementation environment by separating the issue of bias from the issue of variance. To elaborate, classical MCMC estimators, which are based on the empirical distribution after running the MCMC algorithm for a fixed number of iterations, are generally biased unless the algorithm is initialized at the target distribution π. This bias can be problematic in a parallel computing environment, where the number of processors is huge but the computational budget per processor is limited. In contrast, unbiased estimators can be computed on different devices in parallel without communication, allowing users to control the mean-squared error (which is only determined by the variance) to an arbitrarily low level by simply increasing the number of processors. Evidences support the advantage of unbiased estimators in parallel Monte Carlo algorithms are provided in Rosenthal (2000); Nguyen et al. (2022).

To see the advantage of unbiased estimators clearly, one can consider the following comparison: Suppose we are in an environment with sufficiently many processors. Given a quantity $\mathcal{T}(\pi)$ that users wish to estimate, and an ϵ^2-tolerance level for the mean-squared error (MSE). It is known (page 21 of Geyer (2011)) that the bias of standard MCMC estimators is of order $1/n$, where n is the number of iterations. No matter how many processors there are, users have to run each MCMC algorithm on each processor for $n \geq$

$O(1/\epsilon)$ steps to ensure the bias is less than ϵ (otherwise, the squared bias would be greater than ϵ^2, which leads to the MSE being larger than ϵ^2). Therefore the completion time for a standard MCMC estimator is at least of order $1/\epsilon$, which grows to infinity as ϵ goes to 0. In contrast, a typically unbiased Monte Carlo estimator (such as Glynn and Rhee (2014); Jacob et al. (2020); Zhou et al. (2022) and the estimator presented in this paper) has zero bias, finite variance, and a random computational cost with finite expectation. Therefore the users can independently implement the estimators on V/ϵ^2 processors to achieve ϵ^2-MSE, where V is the variance of one such estimator. The average completion time is $O(1)$, which compares much more favorably than the standard MCMC's $O(1/\epsilon)$ completion time. Our empirical experiments also justify the practical usefulness of our unbiased estimators in the parallel environment.

On top of parallel computing, the confidence intervals can be easily constructed using unbiased estimators from Monte Carlo outputs to improve uncertainty quantification in cases where the variance is hard to estimate. Moreover, these unbiased estimators are often more adaptable and can be used as subroutines in more complicated Monte Carlo problems like pseudo-marginal MCMC algorithms (Andrieu and Roberts, 2009) and nested Monte Carlo problems (Rainforth et al., 2018; Zhou et al., 2022).

Without further assumption on $\mathcal{T}$ and π, it is well known that constructing unbiased estimators of $\mathcal{T}(\pi)$ is difficult. Computational challenges appear in both components of the pair $(\mathcal{T}, \pi)$. The bias of standard Monte Carlo estimators arises from the nonlinearity of $\mathcal{T}$ and the sampling error of the MCMC algorithm. Fortunately, recent works provide promising solutions when one component of the above $(\mathcal{T}, \pi)$ pair is easy while the other is relatively difficult. We briefly review the following two cases separately:

- (Case 1: Easy $\mathcal{T}$, difficult π): When $\mathcal{T}$ is an integral operator with respect to some tractable function f, but π is infeasible to sample from, i.e., $\mathcal{T}(\pi) := \mathbb{E}_\pi[f]$ for some intractable π. The problem is considered by Jacob, O'Leary, and Atchadé (JOA henceforth) (Jacob et al., 2020). The JOA estimator, which follows the idea of Glynn and Rhee (2014), solves this problem via couplings of Markov chains. The unbiased MCMC framework has recently raised much attention. It has been applied in convergence diagnostics (Biswas et al., 2019; Biswas and Mackey, 2021; Biswas et al., 2022), gradient estimation (Ruiz et al., 2020), asymptotic variance estimation Douc et al. (2022), and so on.

- (Case 2: Easy π, difficult $\mathcal{T}$): When π can be sampled perfectly, but $\mathcal{T}(\pi) := g(\mathbb{E}_\pi[f])$ is a function of the expectation, or $\mathcal{T}$ is an expectation with respect to a function which further depends on an expectation (e.g, the nested expectation), the state of the art debiasing technique is the unbiased MLMC method developed by McLeish, Glynn, Rhee, and Blanchet (Blanchet et al., 2015; Rhee and Glynn, 2015; McLeish, 2011) which is a randomized version of the celebrated (non-randomized) MLMC methods pioneered by Heinrich and Giles (Heinrich, 2001; Giles, 2008, 2015). Unbiased MLMC methods have also found many applications, including gradient estimation (Shi and Cornish, 2021), optimal stopping (Zhou et al., 2022), robust optimization (Levy et al., 2020).

In summary, the unbiased MCMC method assumes easy $\mathcal{T}$ (an integral operator) but difficult π, and the unbiased MLMC method assumes easy π (perfectly simulable) but

difficult $\mathcal{T}$. Both assumptions can be violated in many practical applications, such as Example 2—4. Although immense progress has been made, there is no systematic way of constructing unbiased estimators for general $\mathcal{T}(\pi)$ beyond special cases.

In this article, we present a step toward designing unbiased estimators of $\mathcal{T}(\pi)$ for the general $(\mathcal{T}, \pi)$ pair by combining and extending the ideas of the unbiased MCMC and MLMC methods. We propose generic unbiased estimators for functions of expectations, i.e., $\mathcal{T}(\pi) = g(m(\pi)) := g(\mathbb{E}_\pi[f(X)])$ where π is a d-dimensional probability measure that can only be approximately sampled by MCMC methods, $f : \mathbb{R}^d \to \mathbb{R}^m$ is a deterministic map, and $g : \mathbb{R}^m \to \mathbb{R}$ is a deterministic function [1]. Other technical assumptions will be made clear in the subsequent sections. The unbiased estimator is easily parallelizable. It has both finite variance and computational cost for a general class of problems, which implies a 'square root convergence rate' that matches the optimal rate of Monte Carlo methods (Novak, 2006) given by the Central Limit Theorem. Moreover, some technical assumptions on g relax the standard 'linear growth' assumption in Blanchet and Glynn (2015) and Blanchet et al. (2019), which may be of independent interest.

Our method can be naturally generalized to the unbiased estimation of the nested expectation introduced in Example 2 under intractable distributions. The nested expectation is commonly regarded as a challenging task for Monte Carlo simulation. Even if one can sample perfectly from the underlying distribution, the standard 'plug-in' Monte Carlo estimator is not only biased but also has a suboptimal computational cost ($\mathcal{O}(\epsilon^{-3})$ or $\mathcal{O}(\epsilon^{-4})$) under varying assumptions to achieve a mean square error (MSE) of ϵ^2. The proposed estimator has three advantages over the standard 'plug-in' estimator. It is unbiased, has $\mathcal{O}(\epsilon^{-2})$ expected computational cost to achieve ϵ^2-MSE, and works when the conditional distribution can only be approximated by MCMC methods.

Our method naturally connects the unbiased MCMC with the MLMC method. Unbiased MCMC is an emerging area in statistics and machine learning for its potential for parallelization. The methodology in Jacob et al. (2020) has been extended to different MCMC algorithms, including the Hamiltonian Monte Carlo (Heng and Jacob, 2019) and the pseudo-marginal MCMC (Middleton et al., 2020). Meanwhile, the MLMC method (both the non-randomized and randomized version) is shown to be successful in applied math, operation research, and computational finance for estimating the expectation of SDE solutions (Giles, 2008; Rhee and Glynn, 2015), option pricing (Belomestny et al., 2015; Zhou et al., 2022), and inverse problems (Hoang et al., 2013; Dodwell et al., 2015; Beskos et al., 2017; Jasra et al., 2018). When the quantity of interest is $\mathbb{E}_\pi[f]$ for challenging underlying distribution π (in contrast to $g(\mathbb{E}_\pi[f])$ that we considered here), there already exists similar ideas on combining the unbiased MLMC and MCMC framework on specific problems. In Heng et al. (2023), Heng et al. (2021), the authors propose a four-way coupling mechanism to unbiasedly estimate $\mathbb{E}_\pi[f]$ when π arises from some stochastic differential equations. Nevertheless, overall, the connections between unbiased MCMC and MLMC methods still seem largely unexplored. We hope this work will serve as a bridge for these communities and invite researchers from broader areas to develop these methods together.

The rest of this paper is organized as follows. Section 1.1 introduces the notations. In Section 2, we describe the high-level idea behind our method without diving into details.

1. For simplicity, we only consider scalar-valued g in this paper, though our method can be naturally generalized to vector-valued functions.

This section will also clarify the connections between unbiased MCMC and MLMC methods. We formally propose our unbiased estimator in Section 3.1. In Section 3.2, we generalize our estimator for estimating nested expectations. In Section 3.4, we state the assumptions and prove the theoretical properties. In Section 4, we implement our method on several examples to study its empirical performance. We conclude this paper in Section 5. Technical details such as proofs and additional experiments are deferred to the Appendix.

1.1 Notations

Throughout this article, we preserve the notation g to denote a function from its domain $\mathcal{D} \subset \mathbb{R}^m$ to $\mathbb{R}$. We write π as a d-dimensional probability measure, and $\pi_1, \cdots, \pi_d$ for its marginal distributions. We denote by $m_f(\pi) := \mathbb{E}_\pi[f(X)]$ the expected value/vector of f under π, and write it as $m(\pi)$ when it is unlikely to cause confusion. The L^p norm of $v \in \mathbb{R}^d$ is written as $\|v\|_p := \left(\sum_{i=1}^d |v_i|^p\right)^{1/p}$. For the L^2 norm, we simply write $\|v\| := \|v\|_2$. The geometric distribution with success probability r is denoted by $\mathsf{Geo}(r)$, and write its probability mass function as $p_n = p_n(r) = (1-r)^{n-1}r$. The uniform distribution on $[0,1]$ is denoted by $\mathsf{U}[0,1]$. The multivariate normal with mean μ and covariance matrix Σ is denoted by $\mathsf{N}(\mu, \Sigma)$. The binomial distribution with N trials and parameter p is denoted by $\mathsf{Binom}(N,p)$. The Poisson distribution with parameter λ is denoted by $\mathsf{Poi}(\lambda)$. Given a set $A \subset \mathbb{R}^d$, we denote by A° all the interior points of A. For a differentiable function $h : \mathbb{R}^d \to \mathbb{R}$, we denote by $Dh := (\frac{\partial h}{\partial x_1}, \frac{\partial h}{\partial x_2}, \cdots, \frac{\partial h}{\partial x_d})$ the gradient of h. Given two probability measures μ and ν, we write their total variation (TV) distance as $\|\mu - \nu\|_{\mathsf{TV}} := \sup_A |\mu(A) - \nu(A)|$. We adopt the convention that $\sum_{i=m}^n a_i = 0$ if $m > n$.

2. A Simple Identity: Unbiased MCMC Meets MLMC

Consider the task of designing unbiased estimators of $g(m(\pi)) = g(\mathbb{E}_\pi[f(X)])$. The problem is extensively studied in the literature when one can draw independent and identically distributed (*i.i.d.*) samples from π. Unbiased estimators are known to exist or not exist under different contexts (Keane and O'Brien, 1994; Jacob and Thiery, 2015). Different debiasing techniques (Nacu and Peres, 2005; Blanchet et al., 2015; Blanchet and Glynn, 2015; Vihola, 2018) have been proposed and analyzed. Among existing methods, the unbiased MLMC framework works with the greatest generality.

When π is infeasible to sample from, our first observation is based on the following simple identity. For every random variable H with $\mathbb{E}[H] = m(\pi)$, we have:

$$g(m(\pi)) = g(\mathbb{E}[H]). \tag{1}$$

Formula (1) is mathematically straightforward, but the right-hand side of (1) is computationally more tractable than the left-hand side. To be more precise, one main difficulty in estimating $g(m(\pi))$ arises from the difficulty in sampling π. However, our observation is the quantity $g(m(\pi))$ essentially depends only $m(\pi)$—an expectation under π, but not π itself. Therefore, the quantity $m(\pi)$ can be replaced by the expectation of any unbiased estimator of $m(\pi)$. In other words, we can relax the previous assumption 'i.i.d. samples from π' by 'i.i.d. unbiased estimators of $m(\pi)$'. Suppose $H_1, H_2, \ldots$ are i.i.d. unbiased estimators of $m(\pi)$ that we can sample from. Then it suffices to estimate $g(\mathbb{E}[H_1])$ unbiasedly. The

difficulty is now reduced to estimating a function of expectation, and the existing unbiased MLMC methods can be applied.

After observing (1), it suffices to construct unbiased estimators of $m(\pi)$ provided that π cannot be directly simulated. The unbiased MCMC framework provides us with natural solutions. Suppose a Markov chain with transition kernel P that targets π as stationary distribution. It is often possible to construct a pair of coupled Markov chains $(Y, Z) = (Y_t, Z_t)_{t=1}^{\infty}$ that both evolve according to P. By design, if the pair (Y_t, Z_{t-1}) meets at some random time τ and stays together after meeting, then the Jacob-O'Leary-Atchadé (JOA) estimator, which will be formally introduced in shortly later, is unbiased for $m(\pi)$. Putting the unbiased MLMC and JOA estimator together, we can unbiasedly estimate $g(m(\pi))$ using the following two-step strategy described in Figure 1 below. The unbiased MCMC algorithm is used here as a generator for random variables with expectation $m(\pi)$. We will use the outputs of the unbiased MCMC algorithm as inputs to feed into the unbiased MLMC approach and eventually construct an unbiased estimator of $g(m(\pi))$.

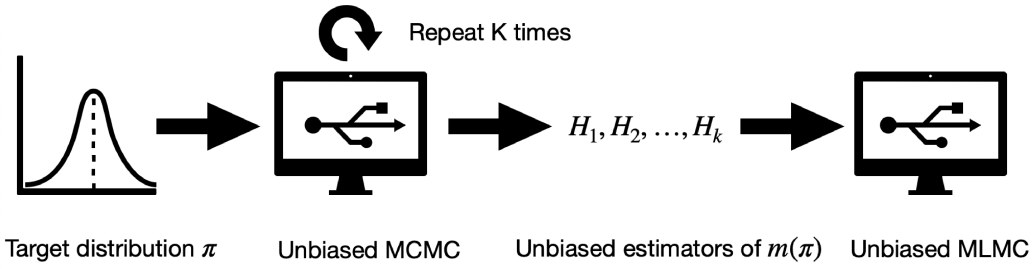


Figure 1: The workflow for constructing an unbiased estimator of $g(m(\pi))$.

3. Unbiased Estimators for Functions of Expectation

In this section, we discuss our estimator for $g(m(\pi))$ from MCMC outputs in detail. We start with a brief review of the JOA estimator of $m(\pi)$ in Section 3.1.1. Our general framework is described in Section 3.1.2. A family of simplified estimators is given in Section 3.1.3 when g admits additional structures. In Section 3.2, we discuss the unbiased estimation of nested expectations using a generalized version of our approach. In Section 3.3, we discuss the problem regarding the domain of g and provide a transformation to avoid the domain problem. In Section 3.4, we give theoretical justifications for our method.

3.1 Constructing an Unbiased Estimator

3.1.1 The Jacob-O'Leary-Atchadé (JOA) estimator of $m(\pi)$

Let Ω be a Polish space equipped with the standard Borel σ-algebra $\mathcal{F}$. Let $P : \Omega \times \mathcal{F} \to [0, 1]$ be the Markov transition kernel that leaves π as stationary distribution. The Jacob-O'Leary-Atchadé (JOA) estimator uses a coupled pair of Markov chains that both have transition kernel P. Formally, the coupled pair $(Y, Z) = (Y_t, Z_t)_{t=1}^{\infty}$ is a Markov chain on the product space $\Omega \times \Omega$. The transition kernel $\bar{P}$, which is also called the coupling of (Y, Z), satisfies $\bar{P}((x, y), A \times \Omega) = P(x, A), \bar{P}((x, y), \Omega \times B) = P(y, B)$ for every $x, y \in \Omega$ and

$A, B \in \mathcal{F}$. The coupled chain starts with $Y_0 \sim \pi_0, Y_1 \sim P(Y_0, \cdot)$ and $Z_0 \sim \pi_0$ independently. Then at each step $t \geq 2$, one samples $(Y_t, Z_{t-1}) \sim \bar{P}((Y_{t-1}, Z_{t-2}), \cdot)$. Suppose the coupling $\bar{P}$ is 'faithful' (Rosenthal, 1997), meaning that there is a random but finite time τ such that $Y_\tau = Z_{\tau-1}$, and $Y_t = Z_{t-1}$ for every $t \geq \tau$. Then for every k, the estimator $H_k(Y, Z) := f(Y_k) + \sum_{i=k+1}^{\tau-1}(f(Y_i) - f(Z_{i-1}))$ is unbiased for $\mathbb{E}_\pi[f]$. The following informal calculation shows the unbiasedness in Jacob et al. (2020):

$$\begin{aligned} m(\pi) = \lim_{n\to\infty} \mathbb{E}[f(Y_n)] &= \mathbb{E}[f(Y_k)] + \sum_{n=k+1}^{\infty} (\mathbb{E}[f(Y_n)] - \mathbb{E}[f(Y_{n-1})]) \\ &= \mathbb{E}[f(Y_k)] + \sum_{n=k+1}^{\infty} \mathbb{E}[f(Z_n) - f(Y_{n-1})] \\ &= \mathbb{E}[f(Y_k)] + \sum_{n=k+1}^{\tau-1} \mathbb{E}[f(Z_n) - f(Y_{n-1})] = \mathbb{E}[H_k(Y, Z)]. \end{aligned}$$

The rigorous proof requires assumptions on the target π and the distribution of τ, see Jacob et al. (2020); Middleton et al. (2020) and our appendix for details. In principle, the above construction works for arbitrary initialization π_0, though the efficiency depends crucially on the initialization. In practice, users typically choose π_0 in the same way as they initialize their standard MCMC algorithm. Furthermore, for any fixed integer $m \geq k$, the 'time-averaged' estimator $H_{k:m}(Y, Z) := (m-k+1)^{-1} \sum_{l=k}^{m} H_l(Y, Z)$ clearly retains unbiasedness and reduces the variance. In practice, users typically choose k to be a large quantile of the coupling time and m to be several multiples of k. Theoretical and empirical investigations of these methods are provided in O'Leary and Wang (2021); Wang et al. (2021). More sophisticated estimators using L-lag coupled chains are discussed in Biswas et al. (2019), but the main idea remains the same.

3.1.2 Unbiased Estimator of $g(m(\pi))$

Suppose we can access a routine $\mathcal{S}$ such as the JOA estimator in Section 3.1.1, which outputs unbiased estimators of $m(\pi)$. The estimator of $g(m(\pi))$ can then be constructed by the randomized MLMC method. Let $H_1, H_2, \cdots, H_{2m}$ be a sequence of *i.i.d.* random variables. We let $S_H(2m) := \sum_{k=1}^{2m} H_i$ be the summation of all the $2m$ terms, and let $S_H^{\mathsf{O}}(m) := \sum_{k=1}^{m} H_{2k-1}, S_H^{\mathsf{E}}(m) := \sum_{k=1}^{m} H_{2k}$ be the summation of all the odd and even terms, respectively. Our estimator is described by Algorithm 1.

Now we discuss the construction of our estimator W for $g(m(\pi))$. Our approach is closely related to the Blanchet—Glynn estimator (Blanchet et al., 2015). The critical difference is that our method relaxes the assumption '*i.i.d.* samples from π' by 'unbiased estimator of $m(\pi)$' and incorporates the JOA estimator as a subroutine. Since exact sampling from π is generally challenging, this relaxation is crucial for practical applications.

After rewriting $g(m(\pi)) = g(\mathbb{E}[H])$, the core idea is to write $g(\mathbb{E}[H])$ as the limit of a sequence of expectations. Here we use the Law of Large Numbers and write

$$g(\mathbb{E}[H]) = \mathbb{E}[\lim_{n\to\infty} g(S_H(2^n)/2^n)] = \lim_{n\to\infty} \mathbb{E}[g(S_H(2^n)/2^n)].$$

After introducing our technical assumptions, we will justify the validity of exchanging the order between the expectation and limit. Then one can write the limit of expectations as

Input:

- A subroutine $\mathcal{S}$ for generating unbiased estimators of $m(\pi)$
- A function $g : \mathcal{D} \to \mathbb{R}$
- The parameter p for geometric distribution

Output: Unbiased estimator of $g(m(\pi))$

1. Sample N from the geometric distribution $\mathsf{Geo}(p)$
2. Call $\mathcal{S}$ for 2^N times and label the outputs by $H_1, ..., H_{2^N}$
3. Calculate the quantities $S_H(2^N)$, $S_H^{\mathsf{O}}(2^{N-1})$ and $S_H^{\mathsf{E}}(2^{N-1})$ defined above
4. Calculate $\Delta_N = g\left(S_H(2^N)/2^N\right) - \frac{1}{2}\left(g\left(S_H^{\mathsf{O}}(2^{N-1})/2^{N-1}\right) + g\left(S_H^{\mathsf{E}}(2^{N-1})/2^{N-1}\right)\right)$

Return: $W = \Delta_N/p_N + g(H_1)$.

Algorithm 1: Unbiased Multilevel Monte-Carlo estimator

an infinite summation of consecutive sums, i.e.,

$$g(E[H]) = \lim_{n\to\infty} \mathbb{E}[g(S_H(2^n)/2^n)] = \mathbb{E}[g(H_1)] + \sum_{n=1}^{\infty} \mathbb{E}[g(S_H(2^n)/2^n)] - \mathbb{E}[g(S_H(2^{n-1})/2^{n-1})]$$
$$= \mathbb{E}[g(H_1)] + \sum_{n=1}^{\infty} \mathbb{E}[\Delta_n],$$

where Δ_n is defined in Step 4 in Algorithm 1. For each fixed n, the random variable Δ_n can be simulated with cost 2^n. To tackle the infinite summation of the expectations, one can choose a random level N with probability p_N and construct the importance sampling-type estimator Δ_N/p_N. The following informal calculation justifies the unbiasedness of W (output of Algorithm 1).

$$\mathbb{E}[W] = \mathbb{E}[g(H_1)] + \mathbb{E}[\Delta_N/p_N] = \mathbb{E}[g(H_1)] + \mathbb{E}[\mathbb{E}[\Delta_N/p_N \mid N]]$$
$$= \mathbb{E}[g(H_1)] + \sum_{n=1}^{\infty} \mathbb{E}[\Delta_n] = \mathbb{E}[g(H_1)] + \sum_{n=1}^{\infty} (\mathbb{E}[g(S_H(2^n)/2^n)] - \mathbb{E}[g(S_H(2^{n-1})/2^{n-1})])$$
$$= \lim_{n\to\infty} \mathbb{E}[g(S_H(2^n)/2^n)] = g(\mathbb{E}[H]) = g(m(\pi)).$$

Moreover, constructing Δ_n is a crucial step in Algorithm 1. The construction in Step 4 of Algorithm 1 is often referred to as the 'antithetic difference estimator,' which is also used in Giles and Szpruch (2014); Blanchet et al. (2015). A natural question is whether one can replace the antithetic difference design with the following seemingly more straightforward estimator: $\tilde{\Delta}_n = g\left(S_H(2^n)/2^n\right) - g\left(S_H(2^{n-1})/2^{n-1}\right)$. It turns out we cannot. The rationale behind the antithetic difference design is that we want to control both the variance and computational cost simultaneously. As we will see from Section 3.4, the antithetic difference design allows one to cancel both the constant and linear terms in the Taylor expansion. In contrast, $\tilde{\Delta}_n$ only cancels the constant term. This difference eventually implies our unbiased estimator (output of Algorithm 1) will have both finite variance and finite computational cost only if we use the antithetic difference design.

It may seem daunting that Algorithm 1 generates 2^N samples for each implementation. However, the actual computational cost is reasonable as the random variable N follows a geometric distribution and therefore has an exponentially light tail that compensates for the exponentially increasing term 2^N. To be more precise, suppose it takes unit cost to call $\mathcal{S}$ once, in several practical cases including Blanchet and Glynn (2015), the authors choose $p = 1 - 2^{-1.5} \approx 0.646$, the expected computational cost for implementing Algorithm 1 once is then around $\sum_{n=0}^{\infty} 2^n(1-p)^{n-1}p = \frac{p}{1-p}\sqrt{2} \approx 2.580$. Therefore, the expected cost of Algorithm 1 is shorter than calling the subroutine $\mathcal{S}$ three times. Detailed discussion on the computational cost and the choice of p can be found in Section 3.4.

We use the JOA estimator in Algorithm 1 as our algorithm needs a subroutine to sample unbiased estimators of $m(\pi)$. In principle, any unbiased estimator of $m(\pi)$ (see, e.g., Agapiou et al. (2018); Ruzayqat et al. (2022)) can also be fed into Algorithm 1 as a subroutine. On the other hand, the JOA estimator appears to be the most general framework for constructing unbiased estimators of $m(\pi)$ given intractable π. For concreteness, we will assume the subroutine $\mathcal{S}$ is the JOA estimator subsequently.

3.1.3 Unbiased estimator of polynomials and other special functions

Section 3.1.2 provides us with a relatively general framework for unbiased estimators of $g(m(\pi))$. In some situations where the target function g has certain nice properties, the unbiased estimators can be easily obtained without resorting to the unbiased MLMC framework. For example, if $g(x) = x^k$ is a univariate monomial function, one can call the unbiased MCMC algorithm k times and obtain unbiased estimators $H_1, \cdots, H_k$ of $\mathbb{E}_\pi[X]$. The estimator $\prod_{l=1}^{k} H_l$ will then be unbiased for $m(\pi)^k$. The argument above can be naturally extended to the case where $m(\pi) \in \mathbb{R}^m$ and $g : \mathbb{R}^m \to \mathbb{R}$ is a multivariate polynomial function. We use the multi-index $k = (k_1, \cdots, k_m)$ with $\sum_{i=1}^{m} k_i \leq n$ where $k_1, \ldots, k_m$ are non-negative integers, and $x^k = x_1^{k_1} x_2^{k_2} \cdots x_m^{k_m}$. Let $g(x) = \sum_{k \leq n} \alpha_k x^k$ denote a multivariate polynomial with degree at most n. The unbiased estimator of $g(m(\pi))$ can be constructed as follows. First, we call the unbiased MCMC subroutine $\mathcal{S}$ for n times and label the outputs by $H_1, \cdots, H_n$, each is an independent vector-valued unbiased estimator of $m(\pi)$. Then for each $k = (k_1, \cdots, k_m)$ we calculate the quantity $\hat{H}(k) = \prod_{l_1=1}^{k_1} H_{l_1,1} \prod_{l_2=k_1+1}^{k_1+k_2} H_{l_2,2} \cdots \prod_{l_m=k_1+\cdots+k_{m-1}+1}^{k_1+\cdots+k_m} H_{l_m,m}$, where $H_{a,b}$ stands for the b-th coordinate of $H_a \in \mathbb{R}^d$. It is clear from the independence of $H_1, \cdots, H_n$ that $\mathbb{E}[\hat{H}(k)] = m(\pi)^k$. Finally, we output $\sum_k \alpha_k \hat{H}(k)$, which is unbiased for $g(m(\pi))$ by the linearity of expectation. It is different from Algorithm 1 as it requires a fixed number of calls for $\mathcal{S}$.

When $g : \mathbb{R} \to \mathbb{R}$ is a real analytic function on $\mathcal{D}$, i.e., $g(x) = \sum_{n=0}^{\infty} a_i(x-a)^n$ for some real number a, where $a_i = g^{(i)}(a)/i!$. Suppose $\tilde{N}$ is a non-negative integer random variable with $\mathbb{P}(\tilde{N} = k) = q_k$. The unbiased estimator for $g(m(\pi))$ can be constructed by first generating $\tilde{N}$, and then calling the subroutine $\mathcal{S}$ for $\tilde{N}$ times to generate unbiased estimators of $\mathbb{E}_\pi[X]$. Denote the outputs by $H_1, \cdots H_{\tilde{N}}$, the final estimator can be expressed by $(a_{\tilde{N}}/q_{\tilde{N}}) \cdot (\prod_{j=1}^{\tilde{N}}(H_j - a)/\tilde{N}!)$ This idea exists in previous literature, such as Blanchet et al. (2015), when π can be perfectly simulated. We generalize this idea to the case where π is intractable. In particular, when $g(x) = e^x$ and $\tilde{N}$ follow from the Poisson distribution, the

estimator is known as the 'Poisson estimator', which is used in both physics and statistics, see Wagner (1987); Papaspiliopoulos (2009); Fearnhead et al. (2010).

As discussed above, these power-series-type estimators require the function g to have all order derivatives at a. Meanwhile, constructing the estimator typically necessitates users having access to all the higher-order derivatives of g. However, this becomes impractical when g is complicated. Therefore, throughout this paper, we will primarily focus on using the unbiased MLMC framework for estimating $g(m(\pi))$ given its generality. This subsection intends to remind our readers that more straightforward choices may exist when g behaves 'nice' enough.

3.2 Nested Expectations

Now we extend our method to estimate the nested expectations. Recall that a nested expectation can be written as $\mathbb{E}_\pi[\lambda]$, where $\lambda(x) := f(x, \gamma(x))$, where $\gamma(x) = \mathbb{E}_{y\sim\pi(y|x)}[\phi(x, y)]$ is another expectation under the conditional distribution. We first decompose the joint distribution $\pi(x, y)$ as the marginal distribution $\pi(x)$ times the conditional distribution of $\pi(y|x)$. When fixing $x = x_0$, then $\lambda(x_0) = f(x_0, \mathbb{E}_{y\sim\pi(y|x_0)}[\phi(x_0, y)])$ is a function of $\mathbb{E}_{\pi(y|x_0)}[\phi(x_0, y)]$ and our previous framework can be applied. Our estimator is as follows.

1. Sample x from $\pi(x)$
2. Given x fixed, generate an unbiased estimator $\hat{\lambda}(x)$ of $\lambda(x)$ using Algorithm 1

Return: $\hat{\lambda}(x)$.

Algorithm 2: Unbiased Multilevel Monte-Carlo estimator for nested expectation

Algorithm 2 can be viewed as the 'conditional' version of Algorithm 1. We first sample x and apply Algorithm 1 to generate an unbiased estimator under $\pi(\cdot|x)$. After taking the randomness of x into account, we show the output Algorithm 2 is unbiased for $\mathbb{E}_\pi[\lambda]$.

Proposition 1 *We have* $\mathbb{E}[\hat{\lambda}] = \mathbb{E}_\pi[\lambda]$.

Proposition 1 will be established as part of the proof for Theorem 4, with detailed explanations provided in Appendix A.4.

Algorithm 2 is useful when $\pi(x)$ can be directly sampled from, and $\pi(y|x)$ can be approximated sampled from some MCMC algorithms. To see the potential applications of Algorithm 2, we present a typical example of the nested expectation, namely estimating the expected utility under partial information (Giles, 2018; Giles and Goda, 2019). Other examples, including the Bayesian experimental design and variational autoencoders, are given in Rainforth et al. (2018); Hironaka and Goda (2023); Goda et al. (2022).

Example 5 (The utility under partial information) *Suppose we have a two-stage process* (X, Y) *with joint distribution* $\pi(x, y)$. *Suppose we have* D *possible strategies (for example, treatments), each with corresponding utility* $f_d(x, y)$ *for* $d \in \{1, \cdots, D\}$. *If we have to choose a strategy without seeing the values of* (X, Y), *the optimal expected utility would be* $\max_d \mathbb{E}[f_d(X, Y)]$. *Similarly, after seeing the whole information, the optimal utility would be* $\mathbb{E}[\max_d f_d(X, Y)]$. *In the intermediate case, if one has observed only* X, *the optimal strategy*

would maximize the conditional utility, i.e., $d^*(X) = \arg\max_d \mathbb{E}[f_d(X,Y)|X]$. *The optimal utility with partial information is* $\mathbb{E}[\max_d \mathbb{E}[f_d(X,Y)|X]]$, *which is a nested expectation.*

The expected utility under partial information finds applications in computational finance, especially in option pricing Belomestny et al. (2015); Zhou et al. (2022). Meanwhile, the difference between full and partial utility $\mathbb{E}[\max_d f_d(X,Y)] - \mathbb{E}[\max_d \mathbb{E}[f_d(X,Y)|X]]$ *quantifies the 'value' of the information in* Y, *which also has applications in the evaluation of Value-at-Risk (VaR) (Giles, 2018) and medical areas (Ades et al., 2004). Existing literature typically assumes one can sample directly from* $\pi(x,y)$, *and regard the intractable* $\pi(x,y)$ *as an open question, see Section 5 of Giles and Goda (2019) for discussions.*

3.3 The Domain Problem and the δ-transformation

There is an extra subtlety in implementing Algorithm 1. Besides requiring H to be an unbiased estimator of $m(\pi)$, Algorithm 1 implicitly requires the range of $S_H(m)/m$ to be a subset of the domain of g. This constraint is naturally satisfied when $g : \mathcal{D} \to \mathbb{R}$ has domain $\mathcal{D} = \mathbb{R}^m$, such as $g(x) = e^x$, or $g(x_1, x_2) = \max\{x_1, x_2, 1\}$. However, many natural functions are not defined on the whole space, such as $g(x) = 1/x$, or $g(x_1, x_2) = x_1/x_2$. These functions arise in statistical applications such as doubly-intractable problems (Lyne et al., 2015), estimating the ratio of normalizing constants (Meng and Wong, 1996). Unfortunately, Algorithm 1 cannot be implemented if $S_H(m)/m$ falls outside $\mathcal{D}$.

Consider a concrete problem of estimating $g(m(\pi))) = 1/m(\pi)$ where π is a probability measure on Ω. The problem can be naturally avoided if $S_H(m)/m \neq 0$ almost surely, which is often the case for continuous state-space Ω. However, the algorithm may fail for discrete state spaces. Even if Ω only contains positive numbers, the resulting JOA estimator may still take 0 with positive probability. The same problem gets worse if the domain of g is of the form $\{x \mid \|x\| \geq c\}$, where both continuous and discrete Markov chains may fail.

We add an extra δ-transformation to address this issue when needed. Suppose $\mathcal{D} \supset \mathbb{R}^d \setminus B_\delta$, where $B_\delta := \{x \mid \|x\| \leq \delta\}$. In other words, $\mathcal{D}$ contains everything in $\mathbb{R}^d$ except for a compact set. Let H be the output of the unbiased MCMC subroutine $\mathcal{S}$. If $\|H\| \leq \delta$, we flip a fair coin and move H to $H + (2\delta/\sqrt{d})\vec{1}$ given head and $H - (2\delta/\sqrt{d})\vec{1}$ given tail, where $\vec{1}$ is the all-one vector in $\mathbb{R}^d$. Formally the transformation can be defined as $H \to \tilde{H} := H 1_{\|H\| \geq \delta} + (H + (2\delta/\sqrt{d})\vec{1}B) 1_{\|H\| < \delta}$, where B follows a uniform two-point distribution on $\{-1, 1\}$. After the transformation, $\tilde{H}$ has support in $\mathcal{D}$, and the next proposition shows $\tilde{H}$ has the same expectation as H (and therefore still unbiased), with covariance matrix no larger than the covariance of H plus a scalar times identity matrix.

Proposition 2 *Let* $\tilde{H}$ *be* δ*-transformation of* H, *then* $\|\tilde{H}\| \geq \delta$ *and* $\mathbb{E}[\tilde{H}] = \mathbb{E}[H] = m(\pi)$, *and* $\mathsf{Cov}[\tilde{H}] = \mathsf{Cov}[H] + \frac{4\delta^2 \mathbb{P}[\|H\| \leq \delta]}{d} I_d \preccurlyeq \mathsf{Cov}[H] + \frac{4\delta^2}{d} I_d$. *Here* $A \preccurlyeq B$ *means* $B - A$ *is a symmetric positive semi-definite matrix.*

The δ-transformation can be used after Step 2 of Algorithm 1 for the outputs of the unbiased MCMC algorithm. After getting $H_1, \ldots, H_{2^N}$ of $m(\pi)$, we could apply the δ-transformation on each of them to ensure every $\tilde{H}_i$ is still unbiased but has support inside $\mathcal{D}$. Since the above proposition shows the δ-transformation only increases the variance by no more than $4\delta^2$, theoretical results in Section 3.4 below also hold for estimators after the transformation, albeit a slightly worse dependency on the constants. To mitigate the

increase in variance resulting from the δ-transformation, Proposition 2 also advises users to select the smallest feasible value of δ that satisfies our assumption $\mathcal{D} \supset \mathbb{R}^d \setminus B_\delta$.

3.4 Theoretical Results

With all the notations above, we are ready to state our technical assumptions and prove the theoretical results. Our theoretical analysis will focus on the unbiased estimator described in Algorithm 1. All the results still go through if the δ-transformation is needed. Recall that g is a function from $\mathcal{D}$ to $\mathbb{R}$, and $H_1, H_2, \cdots$ are *i.i.d.* unbiased estimators of $m(\pi)$. Now we denote by $V_n \subset \mathbb{R}^d$ the range of $(H_1 + \cdots + H_n)/n$ for every n and $V := \cup_{n=1}^{\infty} V_n$. Our assumptions are posed on both g and H_i:

Assumption 3.1 (Domain) *The function $g : \mathcal{D} \to \mathbb{R}$ satisfies $V \subset \mathcal{D}$. Moreover, $m(\pi)$ is in the interior of $\mathcal{D}$, i.e., $m(\pi) \in \mathcal{D}^\circ$.*

Assumption 3.2 (Consistency) $\mathbb{E}[g(S_H(n)/n)] \to g(m(\pi))$ *as* $n \to \infty$.

Assumption 3.3 (Smoothness) *The function g is continuously differentiable in a neighborhood of $m(\pi)$, and $Dg(\cdot)$ is locally Hölder continuous with exponent $\alpha > 0$. In other words, there exists $\varepsilon > 0$, $\alpha > 0$ and $c = c(\epsilon) > 0$ such that s for every $x, y \in (m(\pi) - \epsilon, m(\pi) + \epsilon)$, $\|Dg(x) - Dg(y)\| \le c\|x - y\|^\alpha$.*

Assumption 3.4 (Moment) *There exists some $l > 2 + \alpha$ such that H has finite l-th moments, i.e., $\mathbb{E}[\|H_1\|_l^l] = \sum_{i=1}^m \mathbb{E}[|H_{1,i}|^l] < \infty$.*

Assumption 3.5 (Smoothness—Moment Tradeoff) *Under the assumption that 3.4 is already satisfied, there exist constants $s > 1$, $\alpha_s \in \mathbb{R}$, and $\mathcal{C}_s > 0$ such that $2\alpha_s + (s-1)l > 2s$ and $\mathbb{E}(|\Delta_n|^{2s}) \le \mathcal{C}_s 2^{-\alpha_s n}$ for every $n \ge 0$, where*

$$\Delta_n = \begin{cases} g\left(S_H(2^n)/2^n\right) - \frac{1}{2}\left(g\left(S_H^{\mathsf{O}}(2^{n-1})/2^{n-1}\right) + g\left(S_H^{\mathsf{E}}(2^{n-1})/2^{n-1}\right)\right) & n \ge 1 \\ g(H_1) & n = 0. \end{cases}$$

We briefly comment on the Assumptions 3.1—3.5. The descriptions below are mostly pedagogical, and the detailed proofs are deferred to the Appendix (Section A).

The Domain Assumption 3.1 guarantees Algorithm 1 can be implemented. When g does not directly satisfy this assumption, but $\mathcal{D} \supset \mathbb{R}^d \setminus B_\delta$, then we apply the δ-transformation to enforce the first half of Assumption 3.1 holds. All the theoretical results still hold.

The consistency Assumption 3.2 is expected and somewhat necessary. It appears in related works, including Vihola (2018); Blanchet and Glynn (2015) explicitly or implicitly. Due to the Law of Large Numbers, we already know $S_H(n)/n \to m(\pi)$ almost surely, therefore $g(S_H(n)/n)$ converges to $g(m(\pi))$ due to the continuity of g. Assumption 3.2 requires the expectation of $g(S_H(n)/n)$ to converge to the expectation of its limit. When g is itself bounded, or $g(S_H(n)/n)$ can be uniformly bounded by a random variable with finite mean, then Assumption 3.2 holds automatically due to the dominant convergence theorem. In addition, since the sequence of random variables $\{S_H(n)/n\}$ is uniformly integrable, $S_H(n)/n$ converges to $\mathbb{E}[H]$ both almost surely and in L^1. If we know $|g(x)| \le c(1 + \|x\|)$

for some universal constant c, then Assumption 3.2 still holds by the generalized dominant convergence theorem (Folland, 1999). In other cases, it is necessary to manually verify Assumption 3.2, such as demonstrating the uniform integrability of $g(S_H(n)/n)$.

The Smoothness Assumption 3.3 guarantees both g is smooth enough at a neighborhood of $m(\pi)$, and the derivative of g is Hölder continuous. When g is infinitely differentiable, and there is no singularity on a neighborhood of $m(\pi)$, then we expect Assumption 3.3 to hold with $\alpha \geq 1$. We emphasize that we only require Dg to be locally Hölder continuous near $m(\pi)$, which is much weaker than requiring Dg to be globally Hölder continuous.

The Moment Assumption 3.4 requires more than l-th moment of the unbiased estimator H_i, where l is strictly larger than $2+\alpha$. When the JOA estimator is used for generating H_i, Assumption 3.4 generally holds when f has strictly more than l-th moment under π, and the coupling time τ has a very light tail. The tail behavior of τ is closely related to the mixing time of the underlying MCMC algorithm. We recall that a π-stationary Markov chain with transition kernel P is said to be geometrically ergodic if there is a $\gamma \in (0,1)$ and a function $C : \Omega \to (0,\infty)$ such that $\|P^n(x,\cdot) - \pi\|_{\mathsf{TV}} \leq C(x)\gamma^n$, for π—a.s. x. Geometric ergodicity is a central notion in MCMC theory. There is a large body of literature, including but not limited to, Mengersen and Tweedie (1996); Roberts and Tweedie (1996a,b); Wang (2022); Livingstone et al. (2019), that shows a wide family of MCMC algorithms is geometrically ergodic.

Our result for guaranteeing Assumption 3.4 is the following.

Proposition 3 (Verifying Assumption 3.4, informal) *Suppose the Markov chain P is π-stationary and geometrically ergodic, and f is a measurable function with finite p-th moment under π for any $p > l$. Suppose also there exists a set $\mathcal{S} \subset \Omega$, a constant $\tilde{\epsilon} \in (0,1)$ such that $\inf_{(x,y)\in\mathcal{S}\times\mathcal{S}} \bar{P}((x,y),\mathcal{D}) \geq \tilde{\epsilon}$, where $\mathcal{D} := \{(x,x) : x \in \Omega\}$ is the diagonal of $\Omega\times\Omega$. Then the JOA estimator $H_k(Y,Z) := f(Y_k) + \sum_{i=k+1}^{\tau-1}(f(Y_i) - f(Z_{i-1}))$ has a finite l-th moment, and therefore satisfies Assumption 3.4.*

The formal description of the above proposition and the detailed proofs will be deferred to Appendix A.3. It can be viewed as a slightly stronger version of Proposition 3.1 in Jacob et al. (2020), where the authors established the finite second-order moment.

The Tradeoff Assumption 3.5 bounds $\mathbb{E}[|\Delta_n|^{2s}]$. The condition $2\alpha_s+(s-1)l > 2s$ reflects the tradeoff between the smoothness of g and the moment assumption on H_i. Consider the following scenarios: *1:* Suppose g is at least twice continuously differentiable, and the derivative Dg is Lipschitz continuous. Then we have $\Delta_n = \mathcal{O}((S_H(2^n)/2^n)^2)$ by Taylor expansion. Meanwhile, the Central Limit Theorem (CLT) shows $\Delta_n = \mathcal{O}_p(2^{-n})$. Therefore we choose $\alpha_s = s$, and Assumption 3.5 is true for positive l. In this case, Assumption 3.5 is weaker than Assumption 3.4. *2:* Suppose g is at most of linear growth, i.e., $|g(x)| \leq c(1+\|x\|)$. In this case we can only bound Δ_n by $\mathcal{O}_p(2^{-n/2})$ again by the CLT. We choose $\alpha_s = s/2$ and it thus requires $l > s/(s-1)$. This is also the assumption in Blanchet and Glynn (2015); Blanchet et al. (2019). *3:* Assuming that $\mathbb{E}[|\Delta_n|^{2s}]$ is upper bounded by a constant $C(s)$ that solely relies on s, this occurs when either the function g is bounded, or H is supported in a compact region. Then we can choose $\alpha_s = 0$, and therefore $l > 2s/(s-1)$. In summary, stronger smoothness requirements on g result in weaker assumptions on the moment of H_i, and vice versa.

Our main theoretical result is as follows.

Theorem 1 *Under Assumptions 3.1—3.5, define* $\gamma := \min\{\alpha, \frac{\alpha_s}{s} + \frac{(s-1)l}{2s} - 1\}$ *(which is strictly positive according to Assumption 3.5), if* $N \in \{1, 2, \ldots\}$ *is geometrically distributed with success parameter* $p \in \left(\frac{1}{2}, 1 - \frac{1}{2^{(1+\gamma)}}\right)$*, then the estimator* $W := \frac{\Delta_N}{p_N} + g(H_1)$ *described in Algorithm 1 satisfies:*

1. $\mathbb{E}[W] = g(m(\pi))$,
2. *There exists a constant* C *such that* $\mathsf{Var}(W) \leq \mathbb{E}[W^2] \leq Cp^{-1}\frac{2^{-(1+\gamma)}}{1-\left((1-p)2^{1+\gamma}\right)^{-1}} \leq Cp^{-1}\frac{2^{-(1+\gamma)}}{1-\left((1-p)2^{1+\gamma}\right)^{-1}} < \infty$.
3. *The expected computational cost of Algorithm 1 is finite.*

The proof of Theorem 1 relies on the following key lemma to bound Δ_n:

Lemma 2 *Under Assumptions 3.1—3.5,* $\mathbb{E}[|\Delta_n|^2] = C2^{-(1+\gamma)n}$*, where* $\gamma = \{\alpha, \frac{\alpha_s}{s} + \frac{(s-1)l}{2s} - 1\} > 0$*, and* $C = C(m, l, \epsilon, s, \alpha)$ *is a constant.*

The proof is deferred to Appendix A.2, but the main idea is to use the antithetic design to cancel the linear term in the Taylor expansion. This cancellation in turn gives us $\mathbb{E}[|\Delta_n|^2] = \mathcal{O}(2^{-(1+\Omega(1))n})$, which has an $\mathcal{O}(2^{-(\Omega(1))n})$ gain over the canonical rate from the CLT. With Lemma 2 in hand, we are ready to show Theorem 1.

Proof [Proof of Theorem 1] We will first show Statement 1 assuming Statement 2 holds. Then we show both Statement 2 and 3 holds.

Proof of Statement 1: Suppose W has a finite second moment, then the conditional expectation $\mathbb{E}[W|N]$ is well defined (see Section 4.1 of Durrett (2019)). By the law of iterated expectation: $\mathbb{E}[W] = \mathbb{E}\big[\mathbb{E}[W \mid N]\big] = \mathbb{E}[g(H_1)] + \mathbb{E}\big[\frac{\mathbb{E}[\Delta_n|N]}{p_N}\big] = \mathbb{E}[g(H_1)] + \mathbb{E}\big[d_N/p_N\big]$, where $d_n = \mathbb{E}[g(S_H(2^n)/2^n)] - \mathbb{E}[g(S_H(2^{n-1})/2^{n-1})]$. We can further calculate $\mathbb{E}\big[d_N/p_N\big]$: $\mathbb{E}\big[d_N/p_N\big] = \sum_{i=1}^{\infty}(d_i/p_i)p_i = \sum_{i=1}^{\infty} d_i$. Therefore $\mathbb{E}[W] = \lim_{n\to\infty}\mathbb{E}[g(S_H(2^n)/2^n)] = g(m(\pi))$, as desired. The last equality uses Assumption 3.2.

Proof of Statement 2: Since $\mathbb{E}[W^2] \leq 2\big(\mathbb{E}[g(H_1)^2] + \mathbb{E}\big[\Delta_N^2/p_N^2\big]\big)$, it suffices to show $\mathbb{E}\big[\Delta_N^2/p_N^2\big] < \infty$. We have $\mathbb{E}\big[\Delta_N^2/p_N^2\big] = \sum_{n=1}^{\infty}\mathbb{E}[\Delta_n^2](1-p)^{-n+1}p^{-1}$. By Lemma 2,

$$\mathbb{E}\big[\frac{\Delta_N^2}{p_N^2}\big] \leq Cp^{-1}(1-p)\sum_{n=1}^{\infty}2^{-(1+\gamma)n}(1-p)^{-n} = Cp^{-1}(1-p)\sum_{n=1}^{\infty}\left((1-p)2^{1+\gamma}\right)^{-n}$$
$$= Cp^{-1}\frac{2^{-(1+\gamma)}}{1-\left((1-p)2^{1+\gamma}\right)^{-1}} < \infty,$$

where the last inequality follows from $(1-p) > 2^{-(\gamma+1)}$.

Proof of Statement 3: Let C_H be the computation cost for implementing the unbiased MCMC subroutine $\mathcal{S}$ once. It is shown in Jacob et al. (2020) that $C_H < \infty$. The computation cost for implementing Algorithm 1 essentially comes from 2^N calls of the subroutine $\mathcal{S}$, where $N \sim \mathsf{Geo}(p)$. Therefore it suffices to show 2^N has a finite expectation. We calculate

$$\mathbb{E}[2^N] = \sum_{n=1}^{\infty}2^n p(n) = \sum_{n=1}^{\infty}2^n(1-p)^{n-1}p = \frac{2p}{2p-1} < \infty,$$

where the last inequality follows from $p > 1/2$. ■

Theorem 1 immediately implies the following corollary on the computational cost, with proof given in Appendix A.5.

Corollary 3 *Under Assumption 3.1—3.5, for any $\varepsilon > 0$, we can construct an estimator $\tilde{W} := \sum_{i=1}^{n} W_i/n$ by applying Algorithm 1 repeatedly for n iterations and calculating their average. This construction This construction allows us to achieve an expected computational cost of $\mathcal{O}(1/\epsilon^2)$, ensuring that the mean square error between $\tilde{W}$ and the ground truth $g(m(\pi))$ remains bounded by ϵ^2, i.e. $\mathbb{E}[(\tilde{W} - g(m(\pi)))^2] \leq \epsilon^2$.*

The computation cost $O(1/\epsilon^2)$ is shown to be rate-optimal for Monte Carlo estimators (Heinrich, 1992; Dagum et al., 2000). Moreover, when many processors are available, users often care more about the completion time of each processor rather than the total computational cost of all the processors. The following proposition compares the expected completion time between our unbiased estimator with the standard Monte Carlo estimator. To fix ideas, we assume the standard single-chain Monte Carlo estimator is of the form $S_{\mathrm{MC}}(k, n) := g(\sum_{i=k}^{n} f(X_i)/(n - k + 1)$, where k, n are chosen by users. Here $\{X_i\}_{i=1}^{n}$ is a Markov chain with the same transition kernel P used in the subroutine $\mathcal{S}$, which satisfies the assumptions in Proposition 3. The constant k is known as the 'burn-in' period, and n is the total number of iterations of the Monte Carlo algorithm. We also define the standard multiple-chain Monte Carlo estimator as $\sum_{j=1}^{m} S_{\mathrm{MC},j}(k, n)/m$, which is the average over m independent single-chain estimators. Proposition 4 shows our unbiased estimators have a much faster completion time than the classical estimators in the massively parallel regime. The proof of Proposition 4 is given in Appendix A.5.

Proposition 4 *Under Assumption 3.1—3.5, fix any error tolerance level $\epsilon > 0$, any $p \in (1/2, 1/2^{1+\gamma})$, let W denote the output random variable after implementing Algorithm 1 with parameter p. Suppose the users have more than $\mathsf{Var}_p[W]/\epsilon^2$ available processors, then the users can construct an estimator with MSE at most ϵ^2 within expected computing time $O(1)$ per processor. In contrast, the standard multiple-chain Monte Carlo estimators with any fixed burn-in period k (which is independent of ϵ) have computing time $O(1/\epsilon)$ per processor.*

It is important to note that the analysis of completion time for both methods is conducted relative to the value of ϵ. This means that we are holding the dimensionality and other parameters constant throughout the analysis.

Now we discuss the choice of the parameter p when implementing Algorithm 1 in practice. Theorem 1 suggests every $p \in (1/2, 1 - 1/2^{1+\gamma})$ guarantees unbiasedness, finite variance, and finite computational cost. On the other hand, a larger value of p yields a faster completion time but a larger variance for obtaining one estimator using Algorithm 1. The actual choice of p depends on the user's objective and the number of available processors. Here we discuss two practical scenarios:

- Suppose the user has sufficiently many processors and wants to minimize the completion time. The users should choose the parameter p as larger as possible (but

no larger than the theoretical limit $1 - 1/2^{1+\gamma}$) to fully utilize their parallel computation capacity. To be precise, for fixed $p \in (1 - 1/2^{1+\gamma})$ and error tolerance level $\epsilon > 0$, 'sufficiently many' means more than $\mathsf{Var}_p(W)/\epsilon^2$ processors, where $\mathsf{Var}_p(W)$ is the variance of the output of Algorithm 1 with input parameter p. In practice, the quantity $\mathsf{Var}_p(W)$ is usually unknown to the users as a-priori. Nevertheless, users can either use the upper bound in Theorem 1 as a conservative estimate or run some pre-experiments to estimate $\mathsf{Var}_p(W)$.

- Suppose the user wants to minimize the total computational cost over all the processors (which is different from the completion time when multiple processors are available). Then the objective is to minimize the work-normalized variance $\tilde{\sigma}_p^2(W)$ defined in Glynn and Whitt (1992), which is the product of the computation cost and the variance of an individual estimator. Then it follows from the above calculation that the $\tilde{\sigma}^2(W)$ is upper bounded by a constant multiple of $\sum_{n=1}^{\infty} \left((1-p)2^{1+\gamma}\right)^{-n} \times \sum_{n=1}^{\infty} \left(2(1-p)\right)^n$. Applying the Cauchy-Schwarz inequality to this product of sums, one obtains an upper bound which can be minimized in closed form by taking $p = 1 - 2^{-1-\frac{\gamma}{2}}$. When $\gamma = 1$, p can be chosen as $1 - 2^{-\frac{3}{2}} \approx 0.646$, recovering the result in Blanchet and Glynn (2015).

We also present two CLTs of our estimator. These results directly follow the standard arguments from Glynn and Heidelberger (1991); Blanchet and Glynn (2015). These results show our estimator has the 'square-root' convergence rate. The CLTs can also help establish confidence intervals.

- When the number of estimators $W_1, W_2, \ldots, W_n, \ldots$ in Algorithm 1 goes to infinity, we have $\left(\frac{\sum_{i=1}^n W_i}{\sqrt{n}} - g(m(\pi))\right) \to \mathsf{N}(0, \mathsf{Var}(W_1))$ as $n \to \infty$.
- Given a fixed budget b, let $N(b)$ be the number of *i.i.d.* estimators $W_1, W_2, \ldots, W_{N(b)}$ that can be generated by time b. Then we have $\sqrt{b} \cdot \left(\frac{\sum_{i=1}^{N(b)} W_i}{N(b)} - g(m(\pi))\right) \to \mathsf{N}(0, \tilde{\sigma}^2(W))$ as $b \to \infty$, where $\tilde{\sigma}^2(W)$ is the work-normalized variance defined above.

3.5 Theoretical Results for Unbiased Estimators of Nested Expectation

To conclude this section, we present a theoretical justification of our extension for nested expectations (Section 3.2). Let us recall that our objective is to obtain an unbiased estimation of $\mathbb{E}_x[f(x, \mathbb{E}_y[\phi(x, y)|x])]$. We make the assumption that x can be simulated perfectly, whereas $\pi(y \mid x)$ can only be sampled approximately using an MCMC algorithm. We also define a family of functions $\{g_x\}$ where $g_x(y) := f(x, y)$.

To begin our Algorithm 2, we initially generate a sample x from $\pi(x)$. Then, with x fixed, we sample $N \sim \mathsf{Geo}(p)$ and apply the unbiased MCMC subroutine for 2^N times to generate $H_1(x), \cdots, H_{2^N}(x)$ which are all i.i.d. unbiased estimators of $\mathbb{E}_y[\phi(x, y)|x]$. Then we create the Δ_N as described in Algorithm 1 and output $\hat{\lambda}(x) = \Delta_N/p_N + g_x(H_1)$ which is unbiased for $g_x(\mathbb{E}[\phi(x, y)|x])$.

To state our assumptions, we define that g_x uniformly satisfies the domain assumption 3.1 if each g_x fulfills Assumption 3.1 with $m(\pi(y|x)) := \mathbb{E}_y[\phi(x, y)|x]$. Likewise, we

state that g_x uniformly satisfies assumption 3.2 if $\mathbb{E}[g_x(S_H(n)/n)|x] \to g(m(\pi(y|x)))$. Additionally, we state that g_x uniformly satisfies assumption 3.3 if every g_x satisfies 3.3, and the corresponding Hölder constants are independent of x. The following theorem presents theoretical results for unbiased estimators of nested expectations.

Theorem 4 *Under the assumptions that g_x uniformly satisfies Assumptions 3.1 through 3.3, and that Assumptions 3.4 and 3.5 are valid, let $\gamma := \min\{\alpha, \frac{\alpha_s}{s} + \frac{(s-1)l}{2s} - 1\}$ (which is strictly positive according to Assumption 3.5). If $N \in \{1, 2, \ldots\}$ is geometrically distributed with success parameter $p \in \left(\frac{1}{2}, 1 - \frac{1}{2^{(1+\gamma)}}\right)$, then the estimator $\hat{\lambda}(x)$ described in Algorithm 2 satisfies:*

1. $\mathbb{E}[\hat{\lambda}(x)] = \mathbb{E}_x[f(x, \mathbb{E}_y[\phi(x, y)|x])]$,
2. *There exists a constant C such that*

$$\mathsf{Var}(\hat{\lambda}(x)) \leq Cp^{-1}\frac{2^{-(1+\gamma)}}{1 - \left((1-p)2^{1+\gamma}\right)^{-1}} \leq Cp^{-1}\frac{2^{-(1+\gamma)}}{1 - \left((1-p)2^{1+\gamma}\right)^{-1}} < \infty.$$

3. *The expected computational cost of Algorithm 2 is finite.*

The proof is presented in Appendix A.4.

4. Numerical Examples

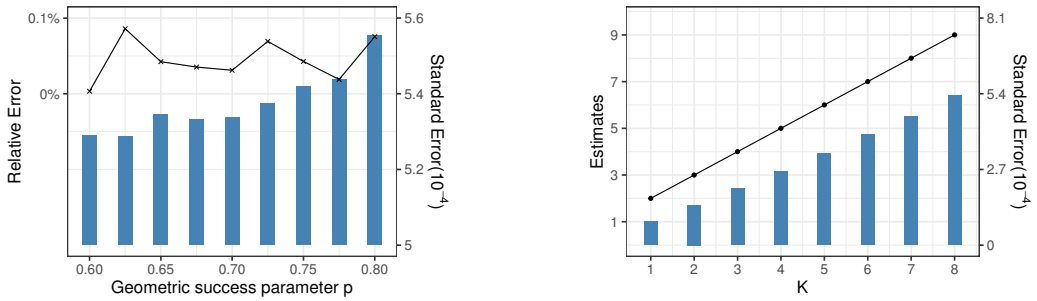
Now we investigate the empirical performance of the proposed method with several examples. We first implement the algorithm on a multivariate Beta distribution and then on a 2-D Ising model with periodic boundaries. In both examples, we compare the performance of our estimator with the standard Monte Carlo estimator when multiple processors are available. Finally, we estimate the nested expectations using a small real-data example modeled by the cut-distribution. Throughout this section, the standard Monte-Carlo (or MCMC/Metropolis—Hastings/Gibbs sampler) estimator for $g(\mathbb{E}_\pi[f])$ should be understood as the the 'plug-in' estimator $g(\sum_{i=l}^{n} f(X_i)/(n-l+1))$, where $\{X_i\}$ follows some MCMC algorithm targeting at π with a burn-in period l. Fix any quantity μ that users wish to estimate, we define the relative error of an estimator X as $\sqrt{\mathbb{E}[(X-\mu)^2]}/|\mu|$.

4.1 Product of Inverse Expectations

We begin with a toy model with known ground truth. Let $X = (X_1, \cdots, X_K)$ be a random vector with independent components $X_i \sim \mathsf{Beta}(i, 1)$. We are interested in the product of the inverse expectation: $g_K(\mathbb{E}[X]) = \prod_{i=1}^{K} 1/\mathbb{E}[X_i]$. Standard calculation shows $g_K(\mathbb{E}[X]) = K + 1$. Meanwhile, g_K cannot be expressed as an expectation, so existing methods fail to provide unbiased estimators.

We apply our method to this problem. We first test the sensitivity of Algorithm 1 to the parameter p, the success probability of the geometric distribution. Setting $K = 8$, and using the R package 'unbiasedmcmc' in Jacob et al. (2020) for estimating $\mathbb{E}[X_i]$ [2], we generate

2. Here the Beta distribution can be perfectly sampled, and there is no need to use the JOA estimator in practice. However, for illustrating our general framework, we still implement the JOA estimators for estimating $\mathbb{E}[X_i]$ via couplings of MCMC algorithms.



(a) p changes, $K = 8$ (b) K changes, $p = 0.7$

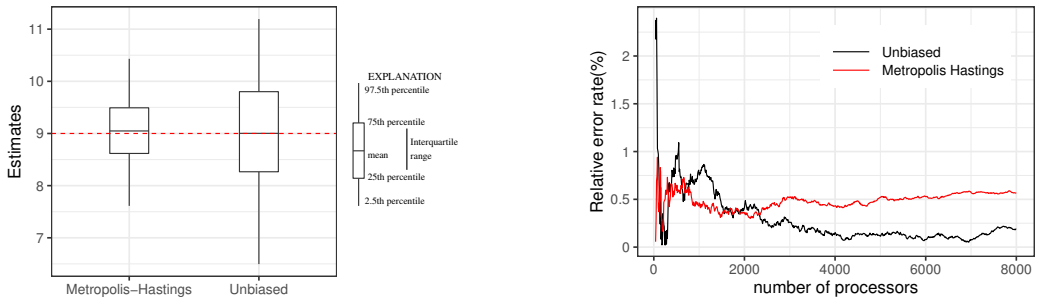
Figure 2: The relative error (line plot) and standard error (histogram) plots for g_K based on 5×10^4 unbiased estimators. Left: Fix dimension $K = 8$, parameter p varies from $0.6 - 0.8$. Right: Fix parameter $p = 0.7$, dimension K varies from 1 to 8.

5×10^4 unbiased estimates of $g_K(\cdot)$ using Algorithm 1 with parameter p ranging from 0.6 to 0.8, $k = 4 \times 10^4$ and $m = 4k$. Figure 2(a) reports the relative and standard errors for each p. The plot shows that the estimates are pretty accurate and vary little for different p. We set $p = 0.7$ in the following experiments to ensure high accuracy and efficient computation. Then we let K change from 1 to 8 and test the accuracy of our method when the dimension varies. For each K, we implement Algorithm 1 for 5×10^4 times independently to generate unbiased estimators of g_K. Our point estimates and the corresponding standard errors are reported in Figure 2(b). It is clear that the point estimates are highly accurate and fit the ground truth almost perfectly. The standard error gets larger when K increases, indicating a higher uncertainty under higher dimensionality.

Now we compare our estimator with a Metropolis-Hastings estimator to show the performance of our method in the parallel regime. To make a fair computation, we use the same random-walk transition kernel in both the unbiased MCMC subroutine $\mathcal{S}$ of Algorithm 1 and the Metropolis-Hastings algorithm. Since Algorithm 1 takes a random computation time per run, we follow Nguyen et al. (2022) to ensure equal computation time across processors as follows: On each processor, we always first run Algorithm 1 and record its running time. Then we run the standard MCMC algorithm for the same time and discard the first 10% samples as burn-in. This way, the two algorithms have the same computational cost for each processor. Finally, we run both methods independently on multiple processors and compare their accuracy after averaging their results respectively over all the processors.

Figure 3(a) depicts the different bias/variance behaviors between a single standard MCMC estimator and our unbiased estimator. A standard MCMC estimator is typically slightly biased but with a smaller variance. Here, the MCMC estimator slightly overestimates the ground truth. In contrast, our unbiased estimator completely eliminates the bias but has a larger variance. For a single estimator, the standard MCMC estimator has a smaller MSE.

Nevertheless, the benefit of unbiasedness becomes significant in the parallel regime, as averaging over multiple processors significantly decreases the variance but keeps the bias the same. As shown in Figure 3(b), when we increase the number of processors, the relative



(a) Box plot of estimators for g_K, $K = 8$

(b) Empirical relative error for g_K, $K = 8$

Figure 3: Left: Box plot of 5×10^4 estimators generated by Metropolis-Hastings and Algorithm 1. The red dashed line represents the true value. Right: Relative error of the standard MCMC estimator (red) and unbiased estimator (black) as a function of the number of processors.

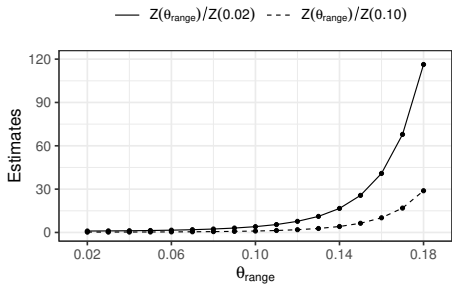
error of our unbiased estimators eventually vanishes. In contrast, the error of the MCMC estimator will never converge to 0 due to its systematic bias. Here the relative error from the systematic bias of MCMC is around 0.5%. In this example, our estimator becomes more accurate than the standard MCMC estimator when there are more than 2500 processors.

4.2 Ising Model

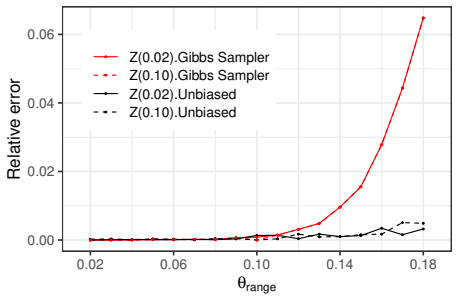
We examine our method on the 2-D square-lattice Ising model. Let Λ be a set of $n \times n$ lattice sites with periodic boundary conditions. A spin configuration $\sigma \in \{-1, 1\}^{n \times n}$ is an assignment of spins to all the lattice vertices. A 2-D Ising model is a probability distribution over all the spin configurations, defined as $p_\theta(\sigma) = \exp(-\theta H(\sigma))/Z(\theta)$. Here $H(\sigma) = -\sum_{\langle I,J \rangle} \sigma_i \sigma_j$ is the 'the Hamiltonian function', where the sum is over all pairs of neighboring sites. The normalizing constant $Z(\theta) = \sum_\sigma \exp(-\theta H(\sigma))$ is the partition function. The parameter $\theta \geq 0$ is interpreted as the inverse temperature in physics.

Now we consider the problem of estimating the ratio of normalizing constant $Z(\theta_1)/Z(\theta_2)$. The problem, also known as estimating the free energy differences, is of great interest in computational physics and statistics (Bennett, 1976; Meng and Wong, 1996). Previous literature has faced challenges in obtaining unbiased estimators of $Z(\theta_1)/Z(\theta_2)$ due to the computational intensity involved in sampling the Ising model perfectly (see Propp and Wilson (1996)). Therefore, our approach can be considered as an unbiased alternative to existing methods such as importance or bridge samplers (Meng and Wong, 1996; Gelman and Meng, 1998).

We will use our method to construct unbiased estimators of $Z(\theta_1)/Z(\theta_2)$. First, we notice that the ratio can be written as $Z(\theta_1)/Z(\theta_2) = \mathbb{E}_{\theta_2}[e^{\theta_2 H(\sigma)}]/\mathbb{E}_{\theta_1}[e^{\theta_1 H(\sigma)}]$. For fixed θ_1, θ_2, we call the JOA estimators for unbiased estimation of $Z(\theta_1)$ and $Z(\theta_2)$ independently and feed them into Algorithm 1 for unbiased estimators of the ratio. The JOA estimators can be obtained via coupling two Gibbs samplers using the package 'unbiasedmcmc' in Jacob et al. (2020). We implement our method using $n = 12, p = 0.7, k = 4 \times 10^3, m = 2k$,



(a) Estimates for $Z_{\theta_1}/Z_{\theta_2}$ as a function of θ_1



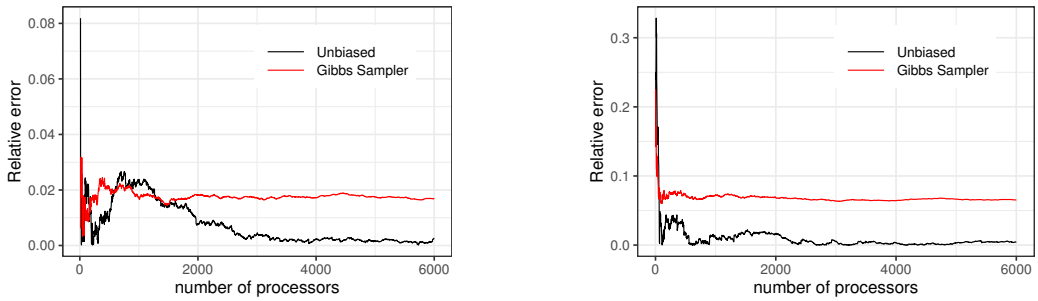
(b) Relative error of different methods as a function of θ_1

Figure 4: Left: The unbiased estimates of $Z(\theta_1)/Z(\theta_2)$ for $n = 12$. Solid lines represent $\theta_2 = 0.02$ and dash lines represent $\theta_2 = 0.10$. Right: Relative error for different algorithms. Black lines are unbiased estimators, and red lines are standard Gibbs sampler estimators.

$\theta_1 \in \{0.02, 0.03, \ldots, 0.18\}$ and $\theta_2 \in \{0.02, 0.10\}$ on a CPU-based computer cluster. For each combination of (θ_1, θ_2), we use our unbiased method to generate 2×10^4 unbiased estimators each. We present results in Figure 4(a). The solid line represents our estimates for $Z(\theta_1)/Z(0.02)$ and dash line represents our estimates for $Z(\theta_1)/Z(0.10)$. For comparison, we also run 2×10^4 independent repetitions of the standard Gibbs sampler estimators for each combination of (θ_1, θ_2). Using the same method described in the previous example (Section 4.1), each run of the Gibbs sampler takes the same amount of time as the unbiased estimator.

To check the accuracy and compare with the standard Gibbs sampler estimator, we need to know the ground truth for every $Z(\theta_1)/Z(\theta_2)$, which is not analytically tractable. Here for each pair (θ_1, θ_2), we run a very long Gibbs sampler for 2×10^5 steps with half burn-in and run 10^4 independent repetitions to estimate both $\mathbb{E}_{\theta_2}[e^{\theta_2 H(\sigma)}]$ and $\mathbb{E}_{\theta_1}[e^{\theta_1 H(\sigma)}]$. Then we use their ratio as a proxy for our ground truth for $Z(\theta_1)/Z(\theta_2)$. Figure 4(b) compares these two methods in terms of their estimation error as a function of θ_1. As shown in the plot, for every (θ_1, θ_2) pair, our unbiased estimator has a relative error very close to 0. This suggests our estimator is highly accurate. In contrast, the Gibbs sampler has a non-negligible bias, which grows as θ_1 grows. In particular, the error (which comes from bias) of the standard Gibbs sampler estimator is more than 6% when θ_1 gets closer to 0.18, while our unbiased estimator has an error much less than 1%.

To further examine the error of both methods as a function of the number of processors, we fix $\theta_2 = 0.1$ and choose $\theta_1 = 0.15$ and 0.18 to plot the relative error versus the number of processors in Figure 5. The behavior is very similar to Figure 3 for the Beta example. Again, as the number of processors increases, the error of the unbiased Monte Carlo estimator vanishes when the number of processors increases. In contrast, the systematic bias causes the error of the Gibbs sampler is always no less than 1.5% and 6% for $\theta_1 = 0.15$ and 0.18, respectively, no matter how many processors are used. Together with the experiments in Section 4.1, it is clear that our estimator is significantly preferable to the standard



(a) Error comparison for $Z_{0.15}/Z_{0.10}$

(b) Error comparison for $Z_{0.18}/Z_{0.10}$

Figure 5: Relative error of the standard MCMC estimator (red) and unbiased estimator (black) as a function of the number of processors.

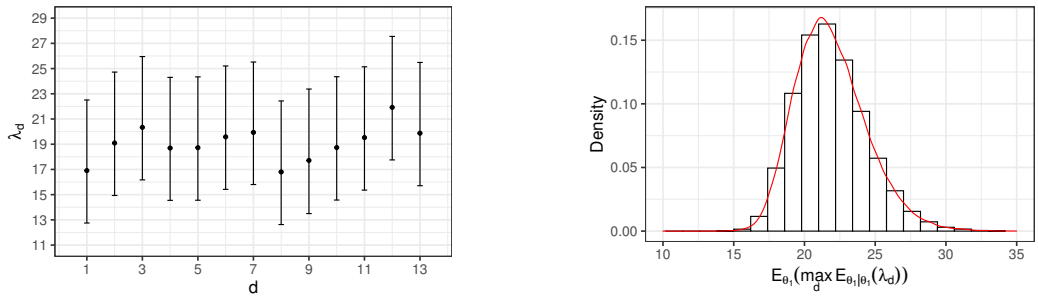
Monte Carlo method when the users have many parallel processors but a limited budget per processor.

4.3 Nested Expectation

Finally, we estimate the following nested expectation: $U := \mathbb{E}_{\theta_1}[\max_d \mathbb{E}_{\theta_2|\theta_1}[f_d(\theta_1, \theta_2)|\theta_1]]$. The quantity $\max_d \mathbb{E}_{\theta_2|\theta_1}[f_d(\theta_1, \theta_2)|\theta_1]$ is often interpreted as the utility or the optimal outcome over D possible choices given the information of θ_1. Since U contains a nested expectation, with an outer expectation over θ_1 and an inner expectation over $\theta_2|\theta_1$, the vanilla Monte Carlo approach (sample N_1 realizations of θ_1, and sample N_2 realizations of θ_2 given each $\theta_1^{(i)}$) typically has suboptimal computational complexity $\mathcal{O}(\epsilon^{-3})$ or even $\mathcal{O}(\epsilon^{-4})$ for ϵ root mean square error (rMSE) under varying assumptions. Therefore, MLMC methods have been proposed when both θ_1 and $\theta_2|\theta_1$ can be perfectly sampled. The case where $\theta_2|\theta_1$ can only be approximately sampled is considered open in (Giles and Goda, 2019).

We construct unbiased estimators of U using the method described in Section 3.2. In this example, suppose we have two models. The first model comprises parameter θ_1 with prior $\pi_1(\theta_1)$, data Y_1 with likelihood $p_1(y|\theta_1)$, the second model comprises parameter θ_2 with prior $\pi_2(\theta_2)$, data Y_2 with likelihood $p_2(y|\theta_1, \theta_2)$. The cut distribution is defined as $\pi^\star(\theta_1, \theta_2) := \pi(\theta_1|Y_1)\pi(\theta_2|Y_2, \theta_1)$. This is different from the usual posterior distribution $\pi(\theta_1, \theta_2|Y_1, Y_2) = \pi(\theta_1|Y_1, Y_2)\pi(\theta_2|Y_2, \theta_1)$. In the cut model, the distribution of θ_1 depends on the observations from the first model (Y_1) but not the second model (Y_2). Since the cut model prevents the information in the second model from influencing the inference on the first, it is often used as an alternative to Bayes full posterior in the presence of model misspecification. Conducting inference on the cut model is challenging. The conditional distribution $\pi(\theta_2|Y_2, \theta_1)$ is usually only known up a normalizing constant $Z(\theta_1)$. Standard MCMC methods on the joint space (θ_1, θ_2) cannot be directly implemented due to the intractability of $Z(\theta_1)$, see (Plummer, 2015) for detailed discussions.

In our case, we consider the real-data example used in (Plummer, 2015; Jacob et al., 2020) from epidemiology, which is motivated by a study of the international correlation



(a) Estimates and 95% confidence intervals for λ_d

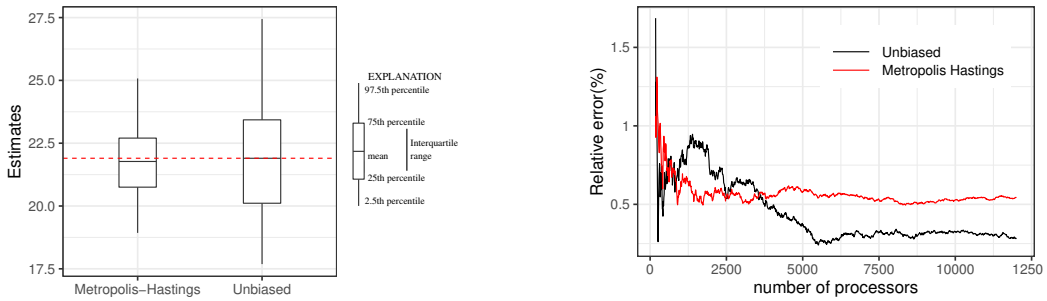
(b) Histogram of estimators

Figure 6: Left: Estimates and 95% CIs for λ_d computed from 10^5 JOA estimators. Right: Histogram of estimators for $\mathbb{E}_{\theta_1}[\max_d \mathbb{E}_{\theta_2|\theta_1}[\lambda_d]]$ computed from 10^5 calls of Algorithm 2.

between human papilloma virus (HPV) prevalence and cervical cancer incidence (Maucort-Boulch et al., 2008). The first module consists of high-risk HPV prevalence data from 13 countries. The data $Y_1 = \{(Z_i, N_i)\}_{i=1}^{13}$ consists of 13 pair of integers, where Z_i is the number of women infected with HPV, from country i with population N_i. We assume a prior $\mathsf{Beta}(1,1)$ on each component of θ_1 independently, and an independent binomial likelihood $Z_i \sim \mathsf{Binom}(N_i, \theta_i)$ for each i. This yields a product beta posterior for θ_1. The second module consists of the cancer data from the same 13 countries. The data $Y_2 = \{(X_{1,i}, X_{2,i})\}_{i=1}^{13}$ consists 13 pair of integers, where $X_{1,i}$ is numbers of cancer cases arising from $X_{2,i}$ woman-years of follow-up. We assume a bivariate normal prior with mean $\mathbf{0}$ and a diagonal covariance matrix with variance 10^3 per component on the parameter $\theta_2 \in \mathbb{R}^2$, and a Poisson regression model $X_{1,i} \sim \mathsf{Poi}(\exp(\lambda_i))$, where $\lambda_i = \theta_{2,1} + \theta_{1,i}\theta_{2,2} + X_{2,i}$.

Under the cut model, the first parameter $\pi(\theta_1|Y_1)$ can be sampled from the product beta, and the second parameter can be approximately sampled from $\pi(\theta_2|Y_2, \theta_1)$ using MCMC. Suppose we are interested in $U := \mathbb{E}_{\theta_1}[\max_{d\in\{1,2,\ldots,13\}} \mathbb{E}_{\theta_2|\theta_1}[\lambda_d]]$, which corresponds to the expectation of the largest parameter in the Poisson regression after observing θ_1. We implement Algorithm 2 with parameter $p = 0.7$ to get unbiased estimators of U. In each run, we first sample one θ_1 from the product beta posterior, then use the JOA estimator with $k = 2 \times 10^3, m = 3 \times 10^3$ by the R package 'unbiasedMCMC' to generate unbiased estimators of $\mathbb{E}_{\theta_2|\theta_1}[\lambda_d]$. Finally, we use the unbiased MLMC method to eliminate the bias. Our estimates are presented in Figure 6 below. Figure 6(a) gives the estimates and their CIs of λ_d for each d. Figure 6(b) gives the histogram and the fitted curve from 10^5 unbiased estimators of U. Figure 6(a) suggests the 12-th country has the largest λ_d, which is around 21, which is consistent with the result from our unbiased estimator on Figure 6(b).

In order to assess the accuracy of our estimator and compare it with the standard Metropolis-Hastings estimator, we initially execute 6×10^3 steps of the Metropolis-Hastings algorithm. The outcome of this execution is then utilized as a substitute for the true value of $\mathbb{E}_{\theta_1}[\max_{d\in 1,2,\ldots,13} \mathbb{E}_{\theta_2|\theta_1}[\lambda_d]]$. In Figure 7(a), we can observe the distinct bias/variance characteristics displayed by a single standard MCMC estimator in comparison to our unbiased estimator. Similar to our previous example illustrated in Figure 3(a), a single standard



(a) Box plot of estimators for $\mathbb{E}_{\theta_1}[\max_d \mathbb{E}_{\theta_2|\theta_1}[\lambda_d]]$ (b) Relative error for $\mathbb{E}_{\theta_1}[\max_d \mathbb{E}_{\theta_2|\theta_1}[\lambda_d]]$

Figure 7: Left: Box plot of 10^5 estimators generated by Metropolis-Hastings and Algorithm 2. The red dashed line represents the true value. Right: Relative error of the standard MCMC estimator (red) and unbiased estimator (black) as a function of the number of processors.

Metropolis—Hastings estimator exhibits a smaller variance but also slightly underestimate the true value, whereas our estimator is unbiased but has a larger variance. Moving on to Figure 7(b), as the number of processors increases, the error of the unbiased Monte Carlo estimator progressively diminishes. In contrast, the systematic bias leads to the error of the Metropolis—Hastings estimator always remaining at no less than 0.5%.

5. Future Works

Based on the combination and generalization of the unbiased MCMC and MLMC method, we propose general unbiased estimators of $g(\mathbb{E}_\pi[f])$ when π can only be approximately sampled. We further extend this framework to estimate nested expectations under intractable distributions. Although promising, the existing framework (Algorithm 1 and its variants) still has the potential to be generalized. We highlight the potential paths forward.

First, $\mathcal{T}$ is assumed to be a function of the expectation. This assumption excludes many important applications, including the quantile and maximum a posteriori (MAP) estimations, where $\mathcal{T}$ depends directly on the probability measure instead of the expectation of some probability measure. We plan to develop a general method to include some/all of the applications above. Taking a step back, many computational challenges remain even assuming $\mathcal{T}(\pi) := g(\mathbb{E}_\pi[f])$. Algorithm 1 implicitly requires the range of $S_H(m)/m$ is a subset of the domain of g. For example, our algorithm fails when $g(x) = \sqrt{x}$ since the JOA estimator may not always be non-negative. As remarked by several authors (Lyne et al., 2015), the domain problem is deeply connected with the sign problem in computational physics, which is NP-hard in its general form. Progress on the domain problem should not only let us improve our existing framework but also benefit both the statistics and physics communities. Lastly, the practical efficiency of the existing estimator (Algorithm 1) is still largely unexplored. In particular, empirical results suggest that the parameter p in Algorithm 1 significantly influences both the variance and the computation cost. Therefore, a strategy of optimizing the parameter p is an interesting open problem. Meanwhile, although

our existing estimator has already achieved the square-root convergence, a stratified version such as Vihola (2018) may still be able to further reduce variance and improve the constant before the rate.

Acknowledgments

Guanyang Wang gratefully acknowledges support by the National Science Foundation through grant DMS-2210849. We would like to thank the Editor, the Action Editor and three referees for their time and efforts in assessing the previous version of the manuscript and their constructive suggestions. Their insightful suggestions have significantly enhanced the quality and content of this paper.

Appendix A. Proofs

A.1 Auxiliary Lemmas

In this section we prove some auxiliary results that will be used throughout the technical proofs. We start (without proof) the well-known Marcinkiewicz-Zygmund inequality, and then prove two useful corollaries based on this inequality.

Lemma 5 (Marcinkiewicz-Zygmund inequality) *(Marcinkiewicz and Zygmund, 1937). If $X_1, \cdots, X_n$ are independent random variables with $\mathbb{E}[X_i] = 0$ and $\mathbb{E}\left[|X_i|^p\right] < \infty$ for some $p > 2$. Then,*

$$\mathbb{E}\left[\left|\sum_{i=1}^n X_i\right|^p\right] \leq C_p \mathbb{E}\left[\left(\sum_{i=1}^n |X_i|^2\right)^{p/2}\right],$$

where C_p is a constant that only depends on p.

One corollary of the Marcinkiewicz-Zygmund inequality is:

Corollary 6 *With all the assumptions as above, if we further assume that $X_1, \cdots, X_n$ are i.i.d.. Then,*

$$\mathbb{E}\left[\left|\frac{1}{n}\sum_{i=1}^n X_i\right|^p\right] \leq C_p \frac{\mathbb{E}|X_1|^p}{n^{p/2}}.$$

Proof [Proof of Corollary 6] Applying the Marcinkiewicz-Zygmund inequality on $(X_1/n, X_2/n, \ldots, X_n/n)$, we have:

$$\mathbb{E}\left[\left|\frac{1}{n}\sum_{i=1}^n X_i\right|^p\right] \leq C_p \mathbb{E}\left[\left(\sum_{i=1}^n |\frac{X_i}{n}|^2\right)^{p/2}\right].$$

Since $x^{p/2}$ is convex, we have

$$\left(\sum_{i=1}^n |\frac{X_i}{n}|^2\right)^{p/2} = \left(\frac{1}{n}\sum_{i=1}^n \frac{|X_i|^2}{n}\right)^{p/2} \leq \frac{1}{n}\sum_{i=1}^n \frac{|X_i|^p}{n^{p/2}}.$$

Taking expectation on both sides of the above inequality yields

$$\mathbb{E}\left[\left(\sum_{i=1}^{n}|\frac{X_i}{n}|^2\right)^{p/2}\right] \leq \frac{\mathbb{E}|X_1|^p}{n^{p/2}},$$

and our desired inequality follows. ■

The Marcinkiewicz-Zygmund inequality naturally generalizes to random vectors.

Corollary 7 (Multivariate Marcinkiewicz-Zygmund inequality) *Let $X_1, \cdots, X_n$ be i.i.d. random vectors in $\mathbb{R}^m$, with $\mathbb{E}[X_i] = \mathbf{0}$ and $\mathbb{E}[\|X_i\|_p^p] = \mathbb{E}[\sum_{j=1}^m |X_{i,j}|^p] < \infty$. Then*

$$\mathbb{E}\left[\|\frac{1}{n}\sum_{i=1}^{n} X_i\|_p^p\right] \leq C_p \frac{\mathbb{E}\left[\|X_1\|_p^p\right]}{n^{p/2}}.$$

Proof [Proof of Corollary 7] We know

$$\mathbb{E}\left[\|\frac{1}{n}\sum_{i=1}^{n} X_i\|_p^p\right] = \sum_{j=1}^{m}\mathbb{E}\left[|\frac{1}{n}\sum_{i=1}^{n} X_{i,j}|^p\right].$$

Applying Corollary 6 on each component of each X_i yields

$$\sum_{j=1}^{m}\mathbb{E}\left[|\frac{1}{n}\sum_{i=1}^{n} X_{i,j}|^p\right] \leq C_p \sum_{j=1}^{m}\frac{\mathbb{E}|X_{1,j}|^p}{n^{p/2}} = C_p\frac{\mathbb{E}\left[\|X_1\|_p^p\right]}{n^{p/2}},$$

as desired. ■

We also need the following inequality to compare $\|x\|_p$ and $\|x\|_q$ for $p \neq q$ and $x \in \mathbb{R}^m$. The proof follows directly from the Hölder's inequality, and we omit its proof here.

Lemma 8 *For any $x \in \mathbb{R}^m$ and $0 < p < q$, we have:*

$$\|x\|_p \leq m^{1/p-1/q}\|x\|_q.$$

A.2 Bounding $\mathbb{E}[|\Delta_n|^2]$

Recall that $\Delta_n = g\left(S_H(2^n)/2^n\right) - \frac{1}{2}\left(g\left(S_H^O(2^{n-1})/2^{n-1}\right) + g\left(S_H^E(2^{n-1})/2^{n-1}\right)\right)$, and the final estimator takes the form $\Delta_N/p_N + g(H_1)$. Therefore, understanding the theoretical properties of Δ_n is crucial for studying our estimator.

Proof [Proof of Lemma 2] For simplicity, we denote $m(\pi)$ by μ. By Assumption 3.3, there exists $\epsilon > 0$ such that g is α-Hölder continuous on $(\mu - \epsilon, \mu + \epsilon)$, we can then write Δ_n as:

$$|\Delta_n| = |\Delta_n|\mathbf{1}(A_1) + |\Delta_n|\mathbf{1}(A_2), \tag{2}$$

where A_1 is the event

$$\left\{\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}} - \mu\right\| < \epsilon\right\} \cap \left\{\left\|\frac{S_H^{\mathsf{E}}(2^{n-1})}{2^{n-1}} - \mu\right\| < \epsilon\right\},$$

and A_2 is the event

$$\left\{\max\left(\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}}-\mu\right\|,\left\|\frac{S_H^{\mathsf{E}}(2^{n-1})}{2^{n-1}}-\mu\right\|\right)\geq\epsilon\right\}$$

Under the event A_1, we have $\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}}-\mu\right\|<\epsilon$ and $\left\|\frac{S_H^{\mathsf{E}}(2^{n-1})}{2^{n-1}}-\mu\right\|<\epsilon$. This further implies

$$\left\|\frac{S_H(2^n)}{2^n}-\mu\right\|<\epsilon$$

by the triangle inequality and the fact $\frac{S_H(2^n)}{2^n}=\frac{1}{2}\left(\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}}+\frac{S_H^{\mathsf{E}}(2^{n-1})}{2^{n-1}}\right)$.

Then we can write Δ_n as:

$$\begin{aligned}
\Delta_n &= g\left(\frac{S_H(2^n)}{2^n}\right)-\frac{1}{2}\left(g\left(\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}}\right)+g\left(\frac{S_H^{\mathsf{E}}(2^{n-1})}{2^{n-1}}\right)\right)\\
&=\frac{1}{2}\left(g\left(\frac{S_H(2^n)}{2^n}\right)-g\left(\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}}\right)\right)+\frac{1}{2}\left(g\left(\frac{S_H(2^n)}{2^n}\right)-g\left(\frac{S_H^{\mathsf{E}}(2^{n-1})}{2^{n-1}}\right)\right)\\
&=\frac{1}{2}Dg(\xi_n^{\mathsf{O}})\left(\frac{S_H(2^n)}{2^n}-\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}}\right)+\frac{1}{2}Dg(\xi_n^{\mathsf{E}})\left(\frac{S_H(2^n)}{2^n}-\frac{S_H^{\mathsf{E}}(2^{n-1})}{2^{n-1}}\right)\\
&=\frac{1}{4}\left(Dg(\xi_n^{\mathsf{O}})-Dg(\xi_n^{\mathsf{E}})\right)\frac{S_H^{\mathsf{E}}(2^{n-1})-S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}},
\end{aligned}$$

where ξ_n^{O} is a convex combination of $\frac{S_H(2^n)}{2^n}$ and $\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}}$, ξ_n^{E} is a convex combination of $\frac{S_H(2^n)}{2^n}$ and $\frac{S_H^{\mathsf{E}}(2^{n-1})}{2^{n-1}}$ by the Multivariate Mean value Theorem. Under A_1, both ξ_n^{O} and ξ_n^{E} are within the ϵ-neighbor of μ, applying the α-Hölder continuous assumption yields

$$|\Delta_n|\leq c_1(\epsilon)\left\|\xi_n^{\mathsf{O}}-\xi_n^{\mathsf{E}}\right\|^{\alpha}\cdot\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})-S_H^{\mathsf{E}}(2^{n-1})}{2^{n-1}}\right\|\leq c_2(\epsilon)\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})-S_H^{\mathsf{E}}(2^{n-1})}{2^{n-1}}\right\|^{1+\alpha}.$$

Then,

$$\mathbb{E}\left[|\Delta_n|^2\mathbf{1}(A_1)\right]\leq c_2(\epsilon)\mathbb{E}\left[\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})-S_H^{\mathsf{E}}(2^{n-1})}{2^{n-1}}\right\|^{2(1+\alpha)}\right]. \tag{3}$$

Since $S_H^{\mathsf{O}}(2^{n-1})$ and $S_H^{\mathsf{E}}(2^{n-1})$ are vectors in $\mathbb{R}^m$, applying Lemma 8 on $p=2,q=2(1+\alpha)$ gives:

$$\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})-S_H^{\mathsf{E}}(2^{n-1})}{2^{n-1}}\right\|^{2(1+\alpha)}\leq m^{\alpha}\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})-S_H^{\mathsf{E}}(2^{n-1})}{2^{n-1}}\right\|_{2(1+\alpha)}^{2(1+\alpha)} \tag{4}$$

Since $S_H^{\mathsf{O}}(2^{n-1})-S_H^{\mathsf{E}}(2^{n-1})$ is the sum of 2^{n-1} i.i.d. random variables, each with the same distribution as H_2-H_1, applying the Multivariate Marcinkiewicz-Zygmund inequality

(Corollary 7) gives us:

$$\mathbb{E}\left[\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})-S_H^{\mathsf{E}}(2^{n-1})}{2^{n-1}}\right\|_{2(1+\alpha)}^{2(1+\alpha)}\right] \leq C_{2(1+\alpha)}\cdot\frac{\mathbb{E}\left[\|H_2-H_1\|_{2(1+\alpha)}^{2(1+\alpha)}\right]}{2^{(1+\alpha)(n-1)}} \tag{5}$$

$$\leq C_{2(1+\alpha)}\cdot 2^{3(1+\alpha)}\cdot\frac{\mathbb{E}\left[\|H_1\|_{2(1+\alpha)}^{2(1+\alpha)}\right]}{2^{(1+\alpha)n}}. \tag{6}$$

where the last step uses the inequality $(a+b)^p \leq 2^{p-1}(|a|^p+|b|^p)$ for $p\geq 2$. It is worth mentioning that the right hand side of (6) is finite as Assumption 3.4 guarantees H_1 has finite l-th moment with $l > 2+\alpha$. Combining (3), (4), and (6), we have:

$$\mathbb{E}\left[|\Delta_n|^2\mathbf{1}(A_1)\right] \leq C_1(m,\alpha,\epsilon)2^{-n(1+\alpha)}, \tag{7}$$

where $C_1(m,\alpha,\epsilon) = c_2(\epsilon)\cdot C_{2(1+\alpha)}\cdot 2^{3(1+\alpha)}\cdot\mathbb{E}\left[\|H_1\|_{2(1+\alpha)}^{2(1+\alpha)}\right]$ is a constant when Assumptions 3.1—3.4 are satisfied.

Under A_2, we have:

$$|\Delta_n|^2\mathbf{1}(A_2) \leq |\Delta_n|^2\mathbf{1}\left(\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}}-\mu\right\|>\epsilon\right)+|\Delta_n|^2\mathbf{1}\left(\left\|\frac{S_H^{\mathsf{E}}(2^{n-1})}{2^{n-1}}-\mu\right\|>\epsilon\right) \tag{8}$$

Now we upper bound the first term's expectation,

$$\mathbb{E}\left[|\Delta_n|^2\mathbf{1}\left(\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}}-\mu\right\|>\epsilon\right)\right] \leq \mathbb{E}[|\Delta_n|^{2s}]^{1/s}\mathbb{P}\left(\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}}-\mu\right\|>\epsilon\right)^{(s-1)/s} \tag{9}$$

$$\leq \mathcal{C}_s^{1/s}2^{-\alpha_s n/s}\mathbb{P}\left(\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}}-\mu\right\|>\epsilon\right)^{(s-1)/s} \tag{10}$$

$$\leq \mathcal{C}_s^{1/s}(\epsilon^{-l(s-1)/s})\cdot 2^{-\alpha_s n/s}\mathbb{E}\left[\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}}-\mu\right\|^l\right]^{(s-1)/s}. \tag{11}$$

Here (9) follows from the Hölder's inequality, (10) uses Assumption 3.5, and (11) follows from the Markov's inequality. Again, using Lemma 8 and Corollary 7, the term $\mathbb{E}\left[\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}}-\mu\right\|^l\right]$ can be upper bounded by:

$$\mathbb{E}\left[\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}}-\mu\right\|^l\right] \leq m^{l/2-1}\mathbb{E}\left[\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}}-\mu\right\|_l^l\right] \leq 2^{l/2}\cdot m^{l/2-1}\cdot C_l\cdot\frac{\mathbb{E}\left[\|H_1\|_l^l\right]}{2^{nl/2}}. \tag{12}$$

Combining (11) and (12), we have

$$\mathbb{E}\left[|\Delta_n|^2\mathbf{1}\left(\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}}-\mu\right\|>\epsilon\right)\right] \leq C_2(m,l,\epsilon,s)2^{-\alpha_s n/s}2^{-nl(s-1)/(2s)}$$

$$= C_2(m,l,\epsilon,s)2^{-n\left(\frac{\alpha_s}{s}+\frac{(s-1)l}{2s}\right)},$$

where $C_2(m,l,\epsilon,s) = \mathcal{C}_s^{1/s} \cdot \left(\epsilon^{-l} 2^{l/2} \cdot m^{l/2-1} \cdot C_l \cdot \mathbb{E}\left[\|H_1\|_l^l\right]\right)^{(s-1)/s}$ is a constant when Assumptions 3.1 - 3.5 are satisfied. Furthermore, by Assumption 3.5, $2\alpha_s + (s-1)l > 2s$. It is clear that $\frac{\alpha_s}{s} + \frac{(s-1)l}{2s} > 1$, and therefore

$$\mathbb{E}\left[|\Delta_n|^2 \mathbf{1}\left(\left\|\frac{S_H^{\mathsf{O}}(2^{n-1})}{2^{n-1}} - \mu\right\| > \epsilon\right)\right] \leq C_2(m,l,\epsilon,s) 2^{-(1+\tilde{\alpha})n}, \tag{13}$$

where $\tilde{\alpha} = \frac{\alpha_s}{s} + \frac{(s-1)l}{2s} - 1 > 0$. The same argument also shows

$$\mathbb{E}\left[|\Delta_n|^2 \mathbf{1}\left(\left\|\frac{S_H^{\mathsf{E}}(2^{n-1})}{2^{n-1}} - \mu\right\| > \epsilon\right)\right] \leq C_2(m,l,\epsilon,s) 2^{-(1+\tilde{\alpha})n}. \tag{14}$$

Combining (13), (14), and (8), we have

$$\mathbb{E}\left[|\Delta_n|^2 \mathbf{1}(A_2)\right] \leq 2C_2(m,l,\epsilon,s) 2^{-(1+\tilde{\alpha})n}. \tag{15}$$

Finally, taking $\gamma = \min\{\alpha, \tilde{\alpha}\}$, $C = C_1 + 2C_2$, and using (2), (7), and (15), we conclude:

$$\mathbb{E}[|\Delta_n|^2] \leq C 2^{-n(1+\gamma)}. \tag{16}$$

■

A.3 The Moment Assumption 3.4 and Markov Chain Mixing

In this subsection we discuss the relation between the Moment Assumption 3.4 and the mixing time of the underlying Markov chain. Throughout this subsection, the unbiased estimator H of $m(\pi)$ is assumed to be the JOA estimator $H_k(Y,Z)$ defined in Section 3.1.1, which also extends to $H_{k:m}(Y,Z) = (m-k+1)^{-1}\sum_{l=k}^m H_l(Y,Z)$ naturally.

Before giving a formal statement of Proposition 3, we first recall some definitions in Markov chain theory. We say a π-invariant, ϕ-irreducible and aperiodic Markov transition kernel P satisfies a geometric drift condition if there exists a measurable function $V : \Omega \to [1,\infty)$, $\lambda \in (0,1)$, and a measurable set $\mathcal{S}$ such that for all $x \in \Omega$:

$$\int P(x, \mathrm{d}y) V(y) \leq \lambda V(x) + b\mathbf{1}(x \in \mathcal{S}). \tag{17}$$

Moreover, the set $\mathcal{S}$ is called a small set if there exists a positive integer m, $\epsilon > 0$, and a probability measure ν on such that for every $x \in \mathcal{S}$:

$$P^m(x, \cdot) \geq \epsilon\mu(\cdot). \tag{18}$$

The technical definitions for irreducibility, aperiodicity and small sets can be found in Chapter 5 of Meyn and Tweedie (2012). The geometric drift condition is a key tool guaranteeing the geometric ergodicity of a Markov chain, meaning the Markov chain P converges to its stationary distribution π at a geometric rate. It is known that the geometric drift condition is satisfied for a wide family of Metropolis-Hastings algorithms. We refer the readers to Mengersen and Tweedie (1996); Roberts and Tweedie (1996b) for existing results.

Now we give a formal statement of Proposition 3.

Proposition 5 (Formal version of Proposition 3) *Suppose the Markov transition kernel described in Section 3.1.1 satisfies a geometric drift condition with a small set $\mathcal{S}$ of the form $\mathcal{S} = \{x : V(x) \leq L\}$ for $\lambda + b/(1+L) < 1$. Suppose there exists $\tilde{\epsilon} \in (0,1)$ such that*

$$\inf_{(x,y)\in\mathcal{S}\times\mathcal{S}} \bar{P}((x,y),\mathcal{D}) \geq \tilde{\epsilon},$$

where $\mathcal{D} := \{(x,x) : x \in \Omega\}$ is the diagonal of $\Omega \times \Omega$. Suppose also there exists $p > l$ and $D_p > 0$ such that $\mathbb{E}[\|f(Y_t)\|_p^p] < D_p$ for every t. Then $\mathbb{E}[\|H_k(Y,Z)\|_l^l] < \infty$ for every k.

The main ingredient in the proof of Proposition 5 is to control the tail probability of the meeting time τ. We say τ has a β-polynomial tail if there exists a constant $K_\beta > 0$ such that

$$\mathbb{P}(\tau > n) \leq K_\beta n^{-\beta}. \tag{19}$$

We say τ has an exponential tail if there exists a constant $K > 0$ and $\gamma \in (0,1)$ such that

$$\mathbb{P}(\tau > n) \leq K\gamma^n. \tag{20}$$

Our next result gives sufficient conditions to ensure Assumption 3.4.

Lemma 9 *Suppose one of the following holds:*

- *There exist $p > l$, $\beta > 0$, and $D_p > 0$ such that $\frac{1}{p} + \beta > \frac{1}{l}$; $\mathbb{E}[\|f(Y_t)\|_p^p] < D_p$ for every t, and τ has a β-polynomial tail;*
- *There exist $p > l$ and $D_p > 0$ such that $\mathbb{E}[\|f(Y_t)\|_p^p] < D_p$ for every t, and τ has an exponential tail.*

Then $\mathbb{E}[\|H_k(Y,Z)\|_l^l] < \infty$ for every k.

Proof [Proof of Lemma 9] We start with the first case. Without loss of generality, we assume $k = 0$ and the estimator $H_0(Y,Z) := f(Y_0) + \sum_{i=1}^{\tau-1}(f(Y_i) - f(Z_{i-1}))$ takes scalar value. Let $D_k := f(Y_k) - f(Z_{k-1})$ for $k \geq 1$, and $D_0 = f(Y_0)$, the estimator can be written as:

$$H_0(Y,Z) = \sum_{k=0}^{\infty} D_k \mathbf{1}(\tau > k).$$

The meeting time τ is almost surely (a.s.) finite by the β-polynomial assumption, therefore $H_0(Y,Z)$ is the limit of $H_0^n(Y,Z) := \sum_{k=0}^n D_k\mathbf{1}(\tau > k)$ in the a.s. sense. We will now prove $H_0^n(Y,Z) \to H_0(Y,Z)$ in L^l, which further implies $\mathbb{E}[|H_0(Y,Z)|^l] < \infty$.

By the Minkowski's inequality on the probability space $L^l(\Omega)$, we have

$$\left(\mathbb{E}[|H_0^n(Y,Z) - H_0(Y,Z)|^l]\right)^{1/l} = \left(\mathbb{E}[|\sum_{k=n+1}^{\infty} D_k\mathbf{1}(\tau > k)|^l]\right)^{1/l} \tag{21}$$

$$\leq \sum_{k=n+1}^{\infty} \left(\mathbb{E}[|D_k\mathbf{1}(\tau > k)|^l]\right)^{1/l}. \tag{22}$$

Every term in (22) can be upper bounded by the Hölder's inequality

$$\left(\mathbb{E}[|D_k\mathbf{1}(\tau > k)|^l]\right)^{1/l} \le \left(\mathbb{E}[|D_k|^p]\right)^{1/p}\left(\mathbb{P}(\tau > k)\right)^{1/q} \qquad \text{here } 1/q = 1/l - 1/p \tag{23}$$

$$\le (2D_p)^{1/p} K_\beta^{1/q} k^{-\beta/q} \tag{24}$$

$$= (2D_p)^{1/p} K_\beta^{1/q} k^{-\frac{\beta}{\frac{1}{l}-\frac{1}{p}}}. \tag{25}$$

Since $\beta > \frac{1}{l} - \frac{1}{p} > 0$, the right hand side of (24) is summable. Therefore we conclude

$$\sum_{k=n+1}^{\infty} \left(\mathbb{E}[|D_k\mathbf{1}(\tau > k)|^l]\right)^{1/l} \to 0$$

as $n \to \infty$, so $H_0^n(Y, Z) \to H_0(Y, Z)$ in L^l.

In the second case, exponential light tail implies β-polynomial tail for every $\beta > 0$, our result immediately follows from the first case.

■

The assumption $\mathbb{E}[\|f(Y_t)\|^p] < D_p$ in Lemma 9 is generally satisfied as long as f has p-th moment under the stationary distribution π. It remains to verify the tail conditions of τ, i.e., formula (19) or (20). The exponential tail (20) and polynomial tail (19) are closely related to the geometric ergodicity and polynomial ergodicity of the underlying marginal Markov chain P, respectively. For simplicity, we only give conditions for the exponential tail here, which is provided in Jacob et al. (2020). The sufficient conditions of polynomial tail of τ can be founded in Theorem 2 of Middleton et al. (2020).

Proposition 6 (Proposition 3.4 in Jacob et al. (2020)) *Suppose the Markov transition kernel described in Section 3.1.1 satisfies a geometric drift condition with a small set $\mathcal{S}$ of the form $\mathcal{S} = \{x : V(x) \le L\}$ for $\lambda + b/(1 + L) < 1$. Suppose there exists $\tilde{\epsilon} \in (0, 1)$ such that*

$$\inf_{(x,y)\in\mathcal{S}\times\mathcal{S}} \bar{P}((x, y), \mathcal{D}) \ge \tilde{\epsilon},$$

where $\mathcal{D} := \{(x, x) : x \in \Omega\}$ is the diagonal of $\Omega \times \Omega$. Then the meeting time τ has a exponential light tail.

Combining Lemma 9 and Proposition 6, the proof of Proposition 5 is immediate.

Proof [Proof of Proposition 5] By Proposition 6, we know τ has an exponential tail. Using the second case of Lemma 9, our result follows. ■

It is still possible to further strengthen Proposition 5 given extra assumptions on τ or f. For example, when τ has an exponential tail and $\mathbb{E}_\pi[e^{\theta f}] < \infty$ for a univariate f and some $\theta > 0$, one can then prove the JOA estimator also has an exponential moment, and thus has every finite-order moment. The existence of an exponential moment may help analyze the concentration properties of the JOA estimator.

A.4 Proof of Theorem 4

The proof of Theorem 4 is similar to Theorem 1. We present the detailed calculations here for concreteness.

We begin by introducing a lemma that serves as the counterpart to Lemma 2, specifically designed for nested expectations.

Lemma 10 *With all the assumptions in Theorem 4, we have* $\mathbb{E}[|\Delta_n|^2] = C2^{-(1+\gamma)n}$, *where* $\gamma = \min\{\alpha, \frac{\alpha_s}{s} + \frac{(s-1)l}{2s} - 1\}$, *and* $C = C(m, l, \epsilon, s, \alpha)$ *is a constant.*

Proof We denote $m(\pi(y|x))$ by $\mu(x)$. By Assumption 3.3, there exists $\epsilon > 0$ such that g_x is α-Hölder continuous on $(\mu(x) - \epsilon, \mu(x) + \epsilon)$, we can then write Δ_n as:

$$|\Delta_n| = |\Delta_n|\mathbf{1}(A_1) + |\Delta_n|\mathbf{1}(A_2), \tag{26}$$

where A_1 is the event

$$\left\{\left\|\frac{S^{\mathsf{O}}_{H(x)}(2^{n-1})}{2^{n-1}} - \mu(x)\right\| < \epsilon\right\} \cap \left\{\left\|\frac{S^{\mathsf{E}}_{H(x)}(2^{n-1})}{2^{n-1}} - \mu(x)\right\| < \epsilon\right\},$$

and A_2 is the event

$$\left\{\max\left(\left\|\frac{S^{\mathsf{O}}_{H(x)}(2^{n-1})}{2^{n-1}} - \mu(x)\right\|, \left\|\frac{S^{\mathsf{E}}_{H(x)}(2^{n-1})}{2^{n-1}} - \mu(x)\right\|\right) \geq \epsilon\right\}.$$

Under the event A_1, we have $\left\|\frac{S^{\mathsf{O}}_{H(x)}(2^{n-1})}{2^{n-1}} - \mu(x)\right\| < \epsilon$ and $\left\|\frac{S^{\mathsf{E}}_{H(x)}(2^{n-1})}{2^{n-1}} - \mu(x)\right\| < \epsilon$. This further implies

$$\left\|\frac{S_{H(x)}(2^n)}{2^n} - \mu(x)\right\| < \epsilon.$$

Then we can write Δ_n as:

$$\begin{aligned}
\Delta_n &= g_x\left(\frac{S_{H(x)}(2^n)}{2^n}\right) - \frac{1}{2}\left(g_x\left(\frac{S^{\mathsf{O}}_{H(x)}(2^{n-1})}{2^{n-1}}\right) + g_x\left(\frac{S^{\mathsf{E}}_{H(x)}(2^{n-1})}{2^{n-1}}\right)\right) \\
&= \frac{1}{2}\left(g_x\left(\frac{S_{H(x)}(2^n)}{2^n}\right) - g_x\left(\frac{S^{\mathsf{O}}_{H(x)}(2^{n-1})}{2^{n-1}}\right)\right) + \frac{1}{2}\left(g_x\left(\frac{S_{H(x)}(2^n)}{2^n}\right) - g_x\left(\frac{S^{\mathsf{E}}_{H(x)}(2^{n-1})}{2^{n-1}}\right)\right) \\
&= \frac{1}{2}Dg_x(\xi^{\mathsf{O}}_n)\left(\frac{S_{H(x)}(2^n)}{2^n} - \frac{S^{\mathsf{O}}_{H(x)}(2^{n-1})}{2^{n-1}}\right) + \frac{1}{2}Dg_x(\xi^{\mathsf{E}}_n)\left(\frac{S_{H(x)}(2^n)}{2^n} - \frac{S^{\mathsf{E}}_{H(x)}(2^{n-1})}{2^{n-1}}\right) \\
&= \frac{1}{4}\left(Dg_x(\xi^{\mathsf{O}}_n) - Dg_x(\xi^{\mathsf{E}}_n)\right)\frac{S^{\mathsf{E}}_{H(x)}(2^{n-1}) - S^{\mathsf{O}}_{H(x)}(2^{n-1})}{2^{n-1}},
\end{aligned}$$

where ξ^{O}_n is a convex combination of $\frac{S_{H(x)}(2^n)}{2^n}$ and $\frac{S^{\mathsf{O}}_{H(x)}(2^{n-1})}{2^{n-1}}$, ξ^{E}_n is a convex combination of $\frac{S_{H(x)}(2^n)}{2^n}$ and $\frac{S^{\mathsf{E}}_{H(x)}(2^{n-1})}{2^{n-1}}$ by the Multivariate Mean value Theorem. Under A_1, both ξ^{O}_n and ξ^{E}_n are within the ϵ-neighbor of $\mu(x)$, applying the α-Hölder continuous assumption yields

$$|\Delta_n| \le c_1(\epsilon) \left\|\xi_n^{\mathsf{O}} - \xi_n^{\mathsf{E}}\right\|^{\alpha} \cdot \left\| \frac{S_{H(x)}^{\mathsf{O}}(2^{n-1}) - S_{H(x)}^{\mathsf{E}}(2^{n-1})}{2^{n-1}} \right\|$$
$$\le c_2(\epsilon) \left\| \frac{S_{H(x)}^{\mathsf{O}}(2^{n-1}) - S_{H(x)}^{\mathsf{E}}(2^{n-1})}{2^{n-1}} \right\|^{1+\alpha}.$$

Then,

$$\mathbb{E}\left[|\Delta_n|^2 \mathbf{1}(A_1)\right] \le c_2(\epsilon) \mathbb{E}\left[\left\| \frac{S_{H(x)}^{\mathsf{O}}(2^{n-1}) - S_{H(x)}^{\mathsf{E}}(2^{n-1})}{2^{n-1}} \right\|^{2(1+\alpha)} \right]. \tag{27}$$

Since $S_{H(x)}^{\mathsf{O}}(2^{n-1})$ and $S_{H(x)}^{\mathsf{E}}(2^{n-1})$ are vectors in $\mathbb{R}^m$, applying Lemma 8 on $p = 2, q = 2(1+\alpha)$ gives:

$$\left\| \frac{S_{H(x)}^{\mathsf{O}}(2^{n-1}) - S_{H(x)}^{\mathsf{E}}(2^{n-1})}{2^{n-1}} \right\|^{2(1+\alpha)} \le m^{\alpha} \left\| \frac{S_{H(x)}^{\mathsf{O}}(2^{n-1}) - S_{H(x)}^{\mathsf{E}}(2^{n-1})}{2^{n-1}} \right\|_{2(1+\alpha)}^{2(1+\alpha)} \tag{28}$$

Since $S_{H(x)}^{\mathsf{O}}(2^{n-1}) - S_{H(x)}^{\mathsf{E}}(2^{n-1})$ is the sum of 2^{n-1} i.i.d. random variables, each with the same distribution as $H_2(x) - H_1(x)$, applying the Multivariate Marcinkiewicz-Zygmund inequality (Corollary 7) on the conditional distribution of $H_2 - H_1$ given x, then taking expectation over $\pi(x)$ gives us:

$$\mathbb{E}\left[\left\| \frac{S_{H(x)}^{\mathsf{O}}(2^{n-1}) - S_{H(x)}^{\mathsf{E}}(2^{n-1})}{2^{n-1}} \right\|_{2(1+\alpha)}^{2(1+\alpha)} \right] \le C_{2(1+\alpha)} \cdot \frac{\mathbb{E}\left[\|H_2 - H_1\|_{2(1+\alpha)}^{2(1+\alpha)}\right]}{2^{(1+\alpha)(n-1)}} \tag{29}$$
$$\le C_{2(1+\alpha)} \cdot 2^{3(1+\alpha)} \cdot \frac{\mathbb{E}\left[\|H_1\|_{2(1+\alpha)}^{2(1+\alpha)}\right]}{2^{(1+\alpha)n}}. \tag{30}$$

where the last step uses the inequality $(a+b)^p \le 2^{p-1}(|a|^p + |b|^p)$ for $p \ge 2$. It is worth mentioning that the right hand side of (30) is finite as Assumption 3.4 guarantees H_1 has finite l-th moment with $l > 2 + \alpha$. Combining (27), (28), and (30), we have:

$$\mathbb{E}\left[|\Delta_n|^2 \mathbf{1}(A_1)\right] \le C_1(m, \alpha, \epsilon) 2^{-n(1+\alpha)}, \tag{31}$$

where $C_1(m, \alpha, \epsilon) = c_2(\epsilon) \cdot C_{2(1+\alpha)} \cdot 2^{3(1+\alpha)} \cdot \mathbb{E}\left[\|H_1\|_{2(1+\alpha)}^{2(1+\alpha)}\right]$ is a constant when Assumptions 3.1—3.4 are satisfied.

Under A_2, we have:

$$|\Delta_n|^2 \mathbf{1}(A_2) \le |\Delta_n|^2 \mathbf{1}\left(\left\| \frac{S_{H(x)}^{\mathsf{O}}(2^{n-1})}{2^{n-1}} - \mu(x) \right\| > \epsilon \right) + |\Delta_n|^2 \mathbf{1}\left(\left\| \frac{S_{H(x)}^{\mathsf{E}}(2^{n-1})}{2^{n-1}} - \mu(x) \right\| > \epsilon \right) \tag{32}$$

Now we upper bound the first term's expectation,

$$\mathbb{E}\left[|\Delta_n|^2\mathbf{1}\left(\left\|\frac{S^{\mathsf{O}}_{H(x)}(2^{n-1})}{2^{n-1}}-\mu(x)\right\|>\epsilon\right)\right] \tag{33}$$

$$\leq \mathbb{E}[|\Delta_n|^{2s}]^{1/s}\mathbb{P}\left(\left\|\frac{S^{\mathsf{O}}_{H(x)}(2^{n-1})}{2^{n-1}}-\mu(x)\right\|>\epsilon\right)^{(s-1)/s} \tag{34}$$

$$\leq \mathcal{C}_s^{1/s}2^{-\alpha_s n/s}\mathbb{P}\left(\left\|\frac{S^{\mathsf{O}}_{H(x)}(2^{n-1})}{2^{n-1}}-\mu(x)\right\|>\epsilon\right)^{(s-1)/s} \tag{35}$$

$$\leq \mathcal{C}_s^{1/s}\cdot(\epsilon^{-l(s-1)/s})\cdot 2^{-\alpha_s n/s}\cdot\mathbb{E}\left[\left\|\frac{S^{\mathsf{O}}_{H(x)}(2^{n-1})}{2^{n-1}}-\mu(x)\right\|^l\right]^{(s-1)/s}. \tag{36}$$

Here (34) follows from the Hölder's inequality, (35) uses Assumption 3.5, and (36) follows from the Markov's inequality. Again, using Lemma 8 and Corollary 7 (again, on the conditional distribution of H given x, and then taking expectation over $\pi(x)$), the term $\mathbb{E}\left[\left\|\frac{S^{\mathsf{O}}_{H(x)}(2^{n-1})}{2^{n-1}}-\mu(x)\right\|^l\right]$ can be upper bounded by:

$$\mathbb{E}\left[\left\|\frac{S^{\mathsf{O}}_{H(x)}(2^{n-1})}{2^{n-1}}-\mu(x)\right\|^l\right]\leq m^{l/2-1}\mathbb{E}\left[\left\|\frac{S^{\mathsf{O}}_{H(x)}(2^{n-1})}{2^{n-1}}-\mu(x)\right\|_l^l\right] \tag{37}$$

$$\leq 2^{l/2}\cdot m^{l/2-1}\cdot C_l\cdot\frac{\mathbb{E}\left[\|H_1\|_l^l\right]}{2^{nl/2}}. \tag{38}$$

Combining (36) and (37), we have

$$\mathbb{E}\left[|\Delta_n|^2\mathbf{1}\left(\left\|\frac{S^{\mathsf{O}}_{H(x)}(2^{n-1})}{2^{n-1}}-\mu(x)\right\|>\epsilon\right)\right]\leq C_2(m,l,\epsilon,s)2^{-\alpha_s n/s}2^{-nl(s-1)/(2s)}$$

$$= C_2(m,l,\epsilon,s)2^{-n\left(\frac{\alpha_s}{s}+\frac{(s-1)l}{2s}\right)},$$

where $C_2(m,l,\epsilon,s)=\mathcal{C}_s^{1/s}\cdot\left(\epsilon^{-l}2^{l/2}\cdot m^{l/2-1}\cdot C_l\cdot\mathbb{E}\left[\|H_1\|_l^l\right]\right)^{(s-1)/s}$ is a constant when Assumptions 3.1 - 3.5 are satisfied. Furthermore, by Assumption 3.5, $2\alpha_s+(s-1)l>2s$. It is clear that $\frac{\alpha_s}{s}+\frac{(s-1)l}{2s}>1$, and therefore

$$\mathbb{E}\left[|\Delta_n|^2\mathbf{1}\left(\left\|\frac{S^{\mathsf{O}}_{H(x)}(2^{n-1})}{2^{n-1}}-\mu(x)\right\|>\epsilon\right)\right]\leq C_2(m,l,\epsilon,s)2^{-(1+\tilde{\alpha})n}, \tag{39}$$

where $\tilde{\alpha}=\frac{\alpha_s}{s}+\frac{(s-1)l}{2s}-1>0$. The same argument also shows

$$\mathbb{E}\left[|\Delta_n|^2\mathbf{1}\left(\left\|\frac{S^{\mathsf{E}}_{H(x)}(2^{n-1})}{2^{n-1}}-\mu(x)\right\|>\epsilon\right)\right]\leq C_2(m,l,\epsilon,s)2^{-(1+\tilde{\alpha})n}. \tag{40}$$

Combining (39), (40), and (32), we have

$$\mathbb{E}\left[|\Delta_n|^2\mathbf{1}(A_2)\right]\leq 2C_2(m,l,\epsilon,s)2^{-(1+\tilde{\alpha})n}. \tag{41}$$

Finally, taking $\gamma = \min\{\alpha, \tilde{\alpha}\}$, $C = C_1 + 2C_2$, and using (26), (31), and (41), we conclude:

$$\mathbb{E}[|\Delta_n|^2] \leq C2^{-n(1+\gamma)}. \tag{42}$$

■

Then the proof of Theorem 4 follows from a direct calculation.

Proof [Proof of Theorem 4] We will first show Statement 1 assuming Statement 2 holds. Then we show both Statement 2 and 3 holds.

Proof of Statement 1: We first argue $\mathbb{E}[\hat{\lambda} \mid x]$ is well-defined, and equals to $\lambda(x)$. Since Statement 2 implies $\hat{\lambda}$ has a finite second moment, we know the $\mathbb{E}[\hat{\lambda} \mid x]$ is well-defined (see Section 4.1 of Durrett (2019)). Meanwhile, $\hat{\lambda}(x)$ is unbiased for $\lambda(x)$ according to Theorem 1. Finally, it is clear by the law of iterated expectation that $\mathbb{E}[\hat{\lambda}] = \mathbb{E}[\mathbb{E}[\hat{\lambda} \mid x]] = \mathbb{E}_x[\lambda(x)] = \mathbb{E}_x[f(x, \mathbb{E}_y[\phi(x, y) \mid x])]$, as desired.

Proof of Statement 2: It suffices to show $\mathbb{E}[\Delta_N^2/p_N^2] < \infty$. We have $\mathbb{E}[\Delta_N^2/p_N^2] = \sum_{n=1}^{\infty} \mathbb{E}[\Delta_n^2](1-p)^{-n+1}p^{-1}$. By Lemma 10,

$$\begin{aligned}\mathbb{E}[\frac{\Delta_N^2}{p_N^2}] \leq Cp^{-1}(1-p)\sum_{n=1}^{\infty} 2^{-(1+\gamma)n}(1-p)^{-n} &= Cp^{-1}(1-p)\sum_{n=1}^{\infty}\left((1-p)2^{1+\gamma}\right)^{-n} \\ &= Cp^{-1}\frac{2^{-(1+\gamma)}}{1-\left((1-p)2^{1+\gamma}\right)^{-1}} < \infty,\end{aligned}$$

where the last inequality follows from $(1-p) > 2^{-(\gamma+1)}$.

Proof of Statement 3: Let C_H be the computation cost for implementing the unbiased MCMC subroutine $\mathcal{S}$ once. It is shown in Jacob et al. (2020) that $C_H < \infty$. The computation cost for implementing Algorithm 1 essentially comes from 2^N calls of the subroutine $\mathcal{S}$, where $N \sim \mathsf{Geo}(p)$. Therefore it suffices to show 2^N has a finite expectation. We calculate

$$\mathbb{E}[2^N] = \sum_{n=1}^{\infty} 2^n p(n) = \sum_{n=1}^{\infty} 2^n(1-p)^{n-1}p = \frac{2p}{2p-1} < \infty,$$

where the last inequality follows from $p > 1/2$. ■

A.5 Other Technical Proofs

A.5.1 Proof of Proposition 2

Proof We first show the unbiasedness of $\tilde{H}$. Notice that

$$\tilde{H} = H1_{\|H\| \geq \delta} + (H + (2\delta/\sqrt{d})\vec{1}B)1_{\|H\| < \delta}$$

where $B \sim \mathsf{U}\{-1, 1\}$ is independent with H. Therefore,

$$\mathbb{E}[\tilde{H}] = \mathbb{E}[H1_{\|H\| \geq \delta}] + \mathbb{E}[(H + (2\delta/\sqrt{d})\vec{1}B)1_{\|H\| < \delta}] = \mathbb{E}[H1_{\|H\| \geq \delta}] + \mathbb{E}[H1_{\|H\| < \delta}] = \mathbb{E}[H].$$

For the covariance, we can calculate the expectation of $\mathbb{E}[\tilde{H}_i\tilde{H}_j]$:

$$\begin{aligned}\mathbb{E}[\tilde{H}_i\tilde{H}_j] &= \mathbb{E}\left[\left(H_i 1_{\|H\|\geq\delta} + (H_i + \frac{2\delta}{\sqrt{d}}\vec{1}B)1_{\|H\|<\delta}\right)\left(H_j 1_{\|H\|\geq\delta} + (H_j + \frac{2\delta}{\sqrt{d}}\vec{1}B)1_{\|H\|<\delta}\right)\right] \\ &= \mathbb{E}[H_iH_j 1_{\|H\|\geq\delta}] + \mathbb{E}[H_iH_j 1_{\|H\|<\delta}] + \frac{4\delta^2}{d}\mathbb{P}[\|H\|\leq\delta] \\ &= \mathbb{E}[H_iH_j] + \frac{4\delta^2}{d}\mathbb{P}[\|H\|\leq\delta],\end{aligned}$$

the second to last equality follows from the fact that B has zero expectation and is independent with H. Therefore, we have

$$\mathsf{Cov}[\tilde{H}] = \mathsf{Cov}[H] + \frac{4\delta^2\mathbb{P}[\|H\|\leq\delta]}{d}I_d \preccurlyeq \mathsf{Cov}[H] + \frac{4\delta^2}{d}I_d,$$

as desired.

■

A.5.2 Proof of Corollary 3

Proof Let W be the estimator output from Algorithm 1. Let $\mathsf{Cost}(W)$ denote its expected computational cost. From Theorem 1, we know both $\mathsf{Var}(W)$ and $\mathsf{Cost}(W)$ is finite. For any fixed integer n, let $W_1, W_2, \ldots, W_n$ be the outputs of n independent calls of Algorithm 1, and let $\tilde{W} := \frac{\sum_{i=1}^n W_i}{n}$ be its average. It follow from the unbiasedness of each W_i that:

$$\mathbb{E}[(\tilde{W} - g(m(\pi)))^2] = \mathsf{Var}(\tilde{W}) = \frac{\mathsf{Var}(W)}{n}.$$

Taking $n = \mathsf{Var}(W)/\epsilon^2$, then the mean square error of $\tilde{W}$ will be no larger than ϵ^2, and the expected computational cost will be $n\mathsf{Cost}(W) = \mathsf{Var}(W)\mathsf{Cost}(W)/\epsilon^2 = \mathcal{O}(1/\epsilon^2)$. ■

A.6 Proof of Proposition 4

Proof Since we have more than $\mathsf{Var}_p[W]/\epsilon^2$ available processors, we can implement Algorithm 1 on each processor once, and average over all the results. The expected computing time per processor is then the expected time of implementing Algorithm 1 once. Theorem 1 shows the expected computing time per processor is $O(1)$ for our unbiased Monte Carlo method.

For the standard Monte Carlo estimator $S_{\mathrm{MC}}(k,n) := g(\sum_{i=k}^n f(X_i)/(n-k+1))$, we may first assume $k = 1$, which means no burn-in is used. Recall that the MSE of any estimator is no less than its squared bias. It follows from Geyer (2011) that the bias of $\sum_{i=1}^n f(X_i)/n$ (with respect to $\mathbb{E}_\pi[f]$) is of order $O(1/n)$. The smoothness assumptions on g show the bias of each Monte Carlo estimator $g(\sum_{i=k}^n f(X_i)/(n-k+1)$ is also of order $O(1/n)$. Since the bias is unchanged after averaging arbitrarily many *i.i.d.* estimators, users have to run their MCMC algorithm for at least $n = O(1/\epsilon)$ iterations to guarantee the squared bias

is of order $O(\epsilon^2)$. For each fixed k, the estimator $g(\sum_{i=k}^{n} f(X_i)/(n-k+1))$ have bias of $O(1/n)$. Therefore for each fixed k, users have to run their algorithm at least $O(1/\epsilon)$ steps. ■

References

AE Ades, G Lu, and K Claxton. Expected value of sample information calculations in medical decision modeling. *Medical decision making*, 24(2):207–227, 2004.

Sergios Agapiou, Gareth O Roberts, and Sebastian J Vollmer. Unbiased Monte Carlo: Posterior estimation for intractable/infinite-dimensional models. *Bernoulli*, 24(3):1726–1786, 2018.

Christophe Andrieu and Gareth O Roberts. The pseudo-marginal approach for efficient Monte Carlo computations. *The Annals of Statistics*, 37(2):697–725, 2009.

Denis Belomestny, Marcel Ladkau, and John Schoenmakers. Multilevel simulation based policy iteration for optimal stopping–convergence and complexity. *SIAM/ASA Journal on Uncertainty Quantification*, 3(1):460–483, 2015.

Charles H Bennett. Efficient estimation of free energy differences from Monte Carlo data. *Journal of Computational Physics*, 22(2):245–268, 1976.

Alexandros Beskos, Ajay Jasra, Kody Law, Raul Tempone, and Yan Zhou. Multilevel sequential Monte Carlo samplers. *Stochastic Processes and their Applications*, 127(5): 1417–1440, 2017.

Niloy Biswas and Lester Mackey. Bounding Wasserstein distance with couplings. *arXiv preprint arXiv:2112.03152*, 2021.

Niloy Biswas, Pierre E Jacob, and Paul Vanetti. Estimating convergence of Markov chains with L-lag couplings. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Niloy Biswas, Anirban Bhattacharya, Pierre E Jacob, and James E Johndrow. Coupling-based convergence assessment of some Gibbs samplers for high-dimensional bayesian regression with shrinkage priors. *Journal of the Royal Statistical Society: Series B (Statistical Methodology), to appear*, 2022.

Jose H. Blanchet and Peter W. Glynn. Unbiased Monte Carlo for optimization and functions of expectations via multi-level randomization. *2015 Winter Simulation Conference (WSC)*, pages 3656–3667, 2015.

Jose H Blanchet, Nan Chen, and Peter W Glynn. Unbiased monte carlo computation of smooth functions of expectations via taylor expansions. In *Winter Simulation Conference*, pages 360–367. IEEE, 2015.

Jose H. Blanchet, Peter W. Glynn, and Yanan Pei. Unbiased Multilevel Monte Carlo: Stochastic optimization, steady-state simulation, quantiles, and other applications. *arXiv preprint arXiv:1904.09929*, 2019.

Paul Dagum, Richard Karp, Michael Luby, and Sheldon Ross. An optimal algorithm for Monte Carlo estimation. *SIAM Journal on computing*, 29(5):1484–1496, 2000.

Tim J Dodwell, Christian Ketelsen, Robert Scheichl, and Aretha L Teckentrup. A hierarchical multilevel Markov chain Monte Carlo algorithm with applications to uncertainty quantification in subsurface flow. *SIAM/ASA Journal on Uncertainty Quantification*, 3 (1):1075–1108, 2015.

Charles R Doss, James M Flegal, Galin L Jones, and Ronald C Neath. Markov chain Monte Carlo estimation of quantiles. *Electronic Journal of Statistics*, 8(2):2448–2478, 2014.

Randal Douc, Pierre E Jacob, Anthony Lee, and Dootika Vats. Solving the poisson equation using coupled Markov chains. *arXiv preprint arXiv:2206.05691*, 2022.

Rick Durrett. *Probability: theory and examples*, volume 49. Cambridge university press, 2019.

Paul Fearnhead, Omiros Papaspiliopoulos, Gareth O Roberts, and Andrew Stuart. Random-weight particle filtering of continuous time processes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(4):497–512, 2010.

Gerald B Folland. *Real analysis: modern techniques and their applications*, volume 40. John Wiley & Sons, 1999.

Andrew Gelman and Xiao-Li Meng. Simulating normalizing constants: From importance sampling to bridge sampling to path sampling. *Statistical science*, pages 163–185, 1998.

Charles J Geyer. Introduction to Markov Chain Monte Carlo. *Handbook of markov chain monte carlo*, 20116022:45, 2011.

Michael B Giles. Multilevel Monte Carlo path simulation. *Operations Research*, 56(3): 607–617, 2008.

Michael B Giles. Multilevel Monte Carlo methods. *Acta Numer.*, 24:259–328, 2015.

Michael B Giles. MLMC for nested expectations. In *Contemporary Computational Mathematics-A Celebration of the 80th Birthday of Ian Sloan*, pages 425–442. Springer, 2018.

Michael B Giles and Takashi Goda. Decision-making under uncertainty: using MLMC for efficient estimation of EVPPI. *Statistics and Computing*, 29(4):739–751, 2019.

Michael B Giles and Lukasz Szpruch. Antithetic multilevel Monte Carlo estimation for multi-dimensional SDEs without Lévy area simulation. *The Annals of Applied Probability*, 24(4):1585–1620, 2014.

Peter W Glynn and Philip Heidelberger. Analysis of parallel replicated simulations under a completion time constraint. *ACM Transactions on Modeling and Computer Simulation*, 1(1):3–23, 1991.

Peter W Glynn and Chang-han Rhee. Exact estimation for Markov chain equilibrium expectations. *Journal of Applied Probability*, 51(A):377–389, 2014.

Peter W Glynn and Ward Whitt. The asymptotic efficiency of simulation estimators. *Operations research*, 40(3):505–520, 1992.

Takashi Goda, Tomohiko Hironaka, Wataru Kitade, and Adam Foster. Unbiased MLMC stochastic gradient-based optimization of Bayesian experimental designs. *SIAM Journal on Scientific Computing*, 44(1):A286–A311, 2022.

Stefan Heinrich. Lower bounds for the complexity of Monte Carlo function approximation. *Journal of Complexity*, 8(3):277–300, 1992.

Stefan Heinrich. Multilevel Monte Carlo methods. In *International Conference on Large-Scale Scientific Computing*, pages 58–67. Springer, 2001.

Jeremy Heng and Pierre E Jacob. Unbiased Hamiltonian Monte Carlo with couplings. *Biometrika*, 106(2):287–302, 2019.

Jeremy Heng, Jeremie Houssineau, and Ajay Jasra. On unbiased score estimation for partially observed diffusions. *arXiv preprint arXiv:2105.04912*, 2021.

Jeremy Heng, Ajay Jasra, Kody JH Law, and Alexander Tarakanov. On unbiased estimation for discretized models. *SIAM/ASA Journal on Uncertainty Quantification (to appear)*, 2023.

Tomohiko Hironaka and Takashi Goda. An efficient estimation of nested expectations without conditional sampling. *Journal of Computational and Applied Mathematics*, 421: 114811, 2023.

Viet Ha Hoang, Christoph Schwab, and Andrew M Stuart. Complexity analysis of accelerated MCMC methods for bayesian inversion. *Inverse Problems*, 29(8):085010, 2013.

Pierre E Jacob and Alexandre H Thiery. On nonnegative unbiased estimators. *The Annals of Statistics*, 43(2):769–784, 2015.

Pierre E Jacob, John O'Leary, and Yves F Atchadé. Unbiased Markov chain Monte Carlo methods with couplings. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 82(3):543–600, 2020.

Ajay Jasra, Kengo Kamatani, Kody JH Law, and Yan Zhou. A multi-index Markov chain Monte Carlo method. *International Journal for Uncertainty Quantification*, 8(1), 2018.

MS Keane and George L O'Brien. A bernoulli factory. *ACM Transactions on Modeling and Computer Simulation*, 4(2):213–219, 1994.

Roger Koenker and Kevin F Hallock. Quantile regression. *Journal of economic perspectives*, 15(4):143–156, 2001.

Daniel Levy, Yair Carmon, John C Duchi, and Aaron Sidford. Large-Scale Methods for Distributionally Robust Optimization. In *NeurIPS*, 2020.

Samuel Livingstone, Michael Betancourt, Simon Byrne, and Mark Girolami. On the geometric ergodicity of Hamiltonian Monte Carlo. *Bernoulli*, 25:3109–3138, 2019.

Anne-Marie Lyne, Mark Girolami, Yves Atchadé, Heiko Strathmann, and Daniel Simpson. On russian roulette estimates for bayesian inference with doubly-intractable likelihoods. *Statistical science*, 30(4):443–467, 2015.

Józef Marcinkiewicz and Antoni Zygmund. Quelques théoremes sur les fonctions indépendantes. *Fund. Math*, 29:60–90, 1937.

Delphine Maucort-Boulch, Silvia Franceschi, and Martyn Plummer. International correlation between human papillomavirus prevalence and cervical cancer incidence. *Cancer Epidemiology and Prevention Biomarkers*, 17(3):717–720, 2008.

Don McLeish. A general method for debiasing a Monte Carlo estimator. *Monte Carlo Methods and Applications*, 17(4):301–315, 2011.

Xiao-Li Meng and Wing Hung Wong. Simulating ratios of normalizing constants via a simple identity: a theoretical exploration. *Statistica Sinica*, pages 831–860, 1996.

Kerrie L Mengersen and Richard L Tweedie. Rates of convergence of the Hastings and Metropolis algorithms. *The Annals of Statistics*, 24(1):101–121, 1996.

Sean P Meyn and Richard L Tweedie. *Markov chains and stochastic stability*. Springer Science & Business Media, 2012.

Lawrence Middleton, George Deligiannidis, Arnaud Doucet, and Pierre E Jacob. Unbiased Markov chain Monte Carlo for intractable target distributions. *Electronic Journal of Statistics*, 14(2):2842–2891, 2020.

Şerban Nacu and Yuval Peres. Fast simulation of new coins from old. *The Annals of Applied Probability*, 15(1A):93–115, 2005.

Tin D Nguyen, Brian L Trippe, and Tamara Broderick. Many processors, little time: MCMC for partitions via optimal transport couplings. In *International Conference on Artificial Intelligence and Statistics*, pages 3483–3514. PMLR, 2022.

Erich Novak. *Deterministic and stochastic error bounds in numerical analysis*, volume 1349. Springer, 2006.

John O'Leary and Guanyang Wang. Metropolis-Hastings transition kernel couplings. *arXiv preprint arXiv:2102.00366*, 2021.

Omiros Papaspiliopoulos. A methodological framework for Monte Carlo probabilistic inference for diffusion processes. 2009.

Martyn Plummer. Cuts in Bayesian graphical models. *Statistics and Computing*, 25(1): 37–43, 2015.

James Gary Propp and David Bruce Wilson. Exact sampling with coupled Markov chains and applications to statistical mechanics. *Random Structures & Algorithms*, 9(1-2):223–252, 1996.

Tom Rainforth, Rob Cornish, Hongseok Yang, Andrew Warrington, and Frank Wood. On nesting Monte Carlo estimators. In *ICML*, 2018.

Chang-han Rhee and Peter W Glynn. Unbiased estimation with square root convergence for SDE models. *Operations Research*, 63(5):1026–1043, 2015.

Gareth O Roberts and Richard L Tweedie. Exponential convergence of langevin distributions and their discrete approximations. *Bernoulli*, pages 341–363, 1996a.

Gareth O Roberts and Richard L Tweedie. Geometric convergence and central limit theorems for multidimensional Hastings and Metropolis algorithms. *Biometrika*, 83(1):95–110, 1996b.

Yaniv Romano, Evan Patterson, and Emmanuel Candes. Conformalized quantile regression. *NeurIPS*, 32:3543–3553, 2019.

Jeffrey S Rosenthal. Faithful couplings of Markov chains: now equals forever. *Advances in Applied Mathematics*, 18(3):372–381, 1997.

Jeffrey S Rosenthal. Parallel computing and Monte Carlo algorithms. *Far East Journal of Theoretical Statistics*, 4(2):207–236, 2000.

Francisco JR Ruiz, Michalis K Titsias, Taylan Cemgil, and Arnaud Doucet. Unbiased gradient estimation for variational auto-encoders using coupled markov chains. *arXiv preprint arXiv:2010.01845*, 2020.

Hamza Ruzayqat, Neil K Chada, and Ajay Jasra. Unbiased Estimation using the Underdamped Langevin Dynamics. *arXiv preprint arXiv:2206.07202*, 2022.

Yuyang Shi and Rob Cornish. On Multilevel Monte Carlo Unbiased Gradient Estimation for Deep Latent Variable Models. In *AISTATS*, 2021.

Ichiro Takeuchi, Quoc V Le, Timothy D Sears, and Alexander J Smola. Nonparametric quantile estimation. *Journal of Machine Learning Research*, 7:1231–1264, 2006.

Matti Vihola. Unbiased estimators and multilevel Monte Carlo. *Operations Research*, 66 (2):448–462, 2018.

Wolfgang Wagner. Unbiased Monte Carlo evaluation of certain functional integrals. *Journal of Computational Physics*, 71(1):21–33, 1987.

Guanyang Wang. On the theoretical properties of the exchange algorithm. *Bernoulli*, 28 (3):1935–1960, 2022.

Guanyang Wang, John O'Leary, and Pierre Jacob. Maximal Couplings of the Metropolis-Hastings Algorithm. In *AISTATS*, pages 1225–1233. PMLR, 2021.

Zhengqing Zhou, Guanyang Wang, Jose H Blanchet, and Peter W Glynn. Unbiased optimal stopping via the muse. *Stochastic Processes and their Applications*, 2022.