# Guaranteed Conservative Fixed Width Confidence Intervals Via Monte Carlo Sampling

Fred J. Hickernell, Lan Jiang, Yuewei Liu, and Art Owen

**Abstract** Monte Carlo methods are used to approximate the means, $\mu$, of random variables $Y$, whose distributions are not known explicitly. The key idea is that the average of a random sample, $Y_1, \ldots, Y_n$, tends to $\mu$ as $n$ tends to infinity. This article explores how one can reliably construct a confidence interval for $\mu$ with a prescribed half-width (or error tolerance) $\varepsilon$. Our proposed two-stage algorithm assumes that the *kurtosis* of $Y$ does not exceed some user-specified bound. An initial independent and identically distributed (IID) sample is used to confidently estimate the variance of $Y$. A Berry-Esseen inequality then makes it possible to determine the size of the IID sample required to construct the desired confidence interval for $\mu$. We discuss the important case where $Y = f(\boldsymbol{X})$ and $\boldsymbol{X}$ is a random $d$-vector with probability density function $\rho$. In this case $\mu$ can be interpreted as the integral $\int_{\mathbb{R}^d} f(\boldsymbol{x})\rho(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$, and the Monte Carlo method becomes a method for multidimensional cubature.

## 1 Introduction

Monte Carlo algorithms provide a flexible way to approximate $\mu = \mathbb{E}(Y)$ when one can generate samples of the random variable $Y$. For example, $Y$ might be the discounted payoff of some financial derivative, which depends on the future performance of assets that are described by a stochastic model. Then $\mu$ is the fair option

Fred J. Hickernell · Lan Jiang
Department of Applied Mathematics, Illinois Institute of Technology, Chicago, IL, 60616, USA
e-mail: hickernell@iit.edu; ljiang14@hawk.iit.edu

Yuewei Liu
School of Mathematics and Statistics, Lanzhou University, Lanzhou City, Gansu, China 730000
e-mail: lyw@lzu.edu.cn

Art Owen
Department of Statistics, Stanford University, Stanford, CA, 94305, USA
e-mail: owen@stanford.edu

price. The goal is to obtain a *confidence interval*

$$\Pr[|\mu - \hat{\mu}| \le \varepsilon] \ge 1 - \alpha, \tag{1}$$

where

- $\mu$ is approximated by the sample average of $n$ independent and identically distributed (IID) samples of $Y$,
$$\hat{\mu} = \hat{\mu}_n = \frac{1}{n} \sum_{i=1}^{n} Y_i, \tag{2}$$
- $\varepsilon$ is the half-width of the confidence interval, which also serves as an *error tolerance*, and
- $\alpha$ is the level of *uncertainty*, e.g., 1% or 0.1%, which is fixed in advance.

Often the sample size, $n$, is fixed in advance, and the central limit theorem (CLT) provides an approximate value for $\varepsilon$ in terms of $n$ and

$$\sigma^2 = \mathrm{Var}(Y) = \mathbb{E}[(Y - \mu)^2], \tag{3}$$

which itself may be approximated by the sample variance. The goal here is somewhat different. We want to fix $\varepsilon$ in advance and then determine how large the sample size must be to obtain a fixed width confidence interval of the form (1). Moreover, we want to make sure that our confidence interval is correct, not just approximately correct, or correct in the limit of vanishing $\varepsilon$. In this paper we present Algorithm 1 for obtaining such a fixed width confidence interval for the mean of a real random variable when one is performing Monte Carlo sampling.

Before presenting the method, we outline the reasons that existing fixed width confidence intervals are not suitable. In summary, there are two drawbacks of existing procedures. Much existing theory is *asymptotic*, i.e., the proposed procedure attains the desired coverage level in the limit as $\varepsilon \to 0$ but does not provide coverage guarantees for fixed $\varepsilon > 0$. We want such fixed $\varepsilon$ guarantees. A second drawback is that the theory may make distributional assumptions that are too strong. In Monte Carlo applications one typically does not have much information about the underlying distribution. The form of the distribution for $Y$ is generally not known, $\mathrm{Var}(Y)$ is generally not known, and $Y$ is not necessarily bounded. We are aiming to derive fixed width confidence intervals that do not require such assumptions.

The width (equivalently length) of a confidence interval tends to become smaller as the number $n$ of sampled function values increases. In special circumstances, we can choose $n$ to get a confidence interval of at most the desired length and at least the desired coverage level, $1 - \alpha$. For instance, if the variance, $\sigma^2 = \mathrm{Var}(Y)$, is known then an approach based on Chebychev's inequality is available, though the actual coverage will usually be much higher than the nominal level, meaning that much narrower intervals would have sufficed. Known variance in addition to a Gaussian distribution for $Y$ supports a fixed width confidence interval construction that is not too conservative. The CLT provides a confidence interval that is asymptotically correct, but our aim is for something that is definitely correct for finite sample sizes. Finally, conservative fixed width confidence intervals for means can be constructed for bounded random variables, by appealing to exponential inequalities such as Hoeffding's or Chernoff's inequality. Unfortunately, $Y$ is often unbounded, e.g., in the case where it represents the payoff of a call option.

If the relevant variance or bound is unknown, then approaches based on sequential statistics [24] may be available. In sequential methods one keeps increasing $n$ until the interval is narrow enough. Sequential confidence intervals require us to take account of the stopping rule when computing the confidence level. Unfortunately, all existing sequential methods are lacking in some aspects.

Serfling and Wackerly [21] consider sequential confidence intervals for the mean (alternatively for the median) in parametric distributions, symmetric about their center point. The symmetry condition is not suitable for general purpose Monte Carlo applications.

Chow and Robbins [2] develop a sequential sampling fixed width confidence interval procedure for the mean, but its guarantees are only asymptotic (as $\varepsilon \to 0$). Mukhopadhyay and Datta [14] give a procedure similar to Chow and Robbins', and it has similar drawbacks.

Bayesian methods can support a fixed width interval containing $\mu$ with $1-\alpha$ posterior probability, and Bayesian methods famously do not require one to account for stopping rules. They do however require strong distributional assumptions.

There is no assumption-free way to obtain exact confidence intervals for a mean, as has been known since Bahadur and Savage [1]. Some kind of assumption is needed to rule out settings where the desired quantity is the mean of a heavy tailed random variable in which rarely seen large values dominate the mean and spoil the estimate of the variance. The assumption we use is an upper bound on the modified kurtosis (normalized fourth moment) of the random variable $Y$:

$$\tilde{\kappa} = \frac{\mathbb{E}[(Y-\mu)^4]}{\sigma^4} \le \tilde{\kappa}_{\max}. \tag{4}$$

(The quantity $\tilde{\kappa}-3$ is commonly called the kurtosis.) Under such an assumption we present a two-stage algorithm: the first stage generates a conservative upper bound on the variance, and the second stage uses this variance bound and a Berry-Esseen Theorem, which can be thought of as a non-asymptotic CLT, to determine how large $n$ must be for the sample mean to satisfy confidence interval (1). Theorem 5 demonstrates the validity of the fixed width confidence interval, and Theorem 6 demonstrates that the cost of this algorithm is reasonable. These are our main new theoretical results.

Our procedure is a two-stage procedure rather than a fully sequential one. In this it is similar to the method of Stein [26, 27], except that the latter requires normally distributed data.

One might question whether assumption (4), which involves fourth moments of $Y$, is more reasonable than an assumption involving only the second moment of $Y$. For example, using Chebychev's inequality with the assumption

$$\sigma^2 \le \sigma^2_{\max} \tag{5}$$

also yields a fixed width confidence interval of the form (1). We would argue that (4) is indeed more reasonable. First, if $Y$ satisfies (4), then so does $cY$ for any nonzero $c$, however, the analog does not hold for (5). In fact, if $\sigma$ is nonzero, then (5) must be violated by $cY$ for $c$ sufficiently large. Second, making $\tilde{\kappa}_{\max}$ a factor of 10 or 100 larger than $\tilde{\kappa}$ does not significantly affect the total cost (number of samples required) of our two-stage Monte Carlo Algorithm 1 for a large range of values of $\sigma/\varepsilon$. However, the cost of our Monte Carlo algorithm, and indeed any Monte Carlo algorithm based on IID sampling is proportional to $\sigma^2$, so overestimating $\sigma^2$ by a factor of 10 or 100 or more to be safe increases the cost of the algorithm by that factor.

An important special case of computing $\mu = \mathbb{E}(Y)$ arises in the situation where $Y = f(\boldsymbol{X})$ for some function $f : \mathbb{R}^d \to \mathbb{R}$ and some random vector $\boldsymbol{X}$ with probability density function $\rho : \mathbb{R}^d \to [0,\infty)$. One may then interpret the mean of $Y$ as the multidimensional integral

$$\mu = \mu(f) = \mathbb{E}(Y) = \int_{\mathbb{R}^d} f(\boldsymbol{x})\rho(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}. \tag{6}$$

Note that unlike the typical probability and statistics setting, where $f$ denotes a probability density function, in this paper $f$ denotes an integrand, and $\rho$ denotes the probability density function. Given the problem of evaluating $\mu = \int_{\mathbb{R}^d} g(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$, one must choose a probability density function $\rho$ for which one can easily generate random vectors $\boldsymbol{X}$, and then set $f = g/\rho$. The quantities $\sigma^2$ and $\tilde{\kappa}$ defined above can be written in terms of weighted $\mathcal{L}_p$-norms of $f$:

$$\|f\|_p := \left\{\int_{\mathbb{R}^d} |f(\boldsymbol{x})|^p \rho(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}\right\}^{1/p}, \qquad \sigma^2 = \|f-\mu\|_2^2, \qquad \tilde{\kappa} = \frac{\|f-\mu\|_4^4}{\|f-\mu\|_2^4}. \tag{7}$$

For a given $g$, the choice of $\rho$ is not unique, and making an optimal choice belongs to the realm of *importance sampling*. The assumption of bounded kurtosis, (4), required by Algorithm 1, corresponds to an assumption that the integrand $f$ lies in the *cone* of functions

$$\mathcal{C}_{\tilde{\kappa}_{\max}} = \{f \in \mathcal{L}_4 : \|f-\mu(f)\|_4 \le \tilde{\kappa}_{\max}^{1/4} \|f-\mu(f)\|_2\}. \tag{8}$$

This is in contrast to a *ball* of functions, which would be the case if one was satisfying a bounded variance condition, (5).

From the perspective of numerical analysis, if $\rho$ has independent marginals, one may apply a product form of a univariate quadrature rule to evaluate $\mu$. However, this consumes a geometrically increasing number of samples as $d$ increases, and moreover, such methods often require rather strict smoothness assumptions on $f$.

If $f$ satisfies moderate smoothness conditions, then (randomized) quasi-Monte Carlo methods, or low discrepancy sampling methods for evaluating $\mu$ are more efficient than simple Monte Carlo [3, 9, 16, 25]. Unfortunately, practical error estimation remains a challenge for quasi-Monte Carlo methods. Heuristic methods have been proposed, but they lack theoretical justification. One such heuristic is used with reasonable success in the numerical examples of Section 4. Independent randomizations of quasi-Monte Carlo rules of fixed sample size can be used to estimate their errors, but they do not yet lead to guaranteed, fixed width confidence intervals.

Computational mathematicians have also addressed the problem of constructing automatic algorithms, i.e., given an error tolerance of $\varepsilon$, one computes an approximation, $\hat{\mu}$, based on $n$ evaluations of the integrand $f$, such that $|\mu - \hat{\mu}| \le \varepsilon$. For example, MATLAB [28], a popular numerical package, contains `quad`, an adaptive Simpson's rule for univariate quadrature routine developed by Gander and Gautschi [4]. Although `quad` and other automatic rules generally work well in practice, they do not have any rigorous guarantees that the error tolerance is met, and it is relatively simple to construct functions that fool them. This is discussed in Section 4. Since a random algorithm, like Monte Carlo, gives a random answer, any statements about satisfying an error criterion must be probabilistic. This leads us back to the problem of finding a fixed width confidence interval, (1).

An outline of this paper follows. Section 2 defines key terminology and provides certain inequalities used to construct our fixed width confidence intervals. The new two-stage Algorithm 1 is described in Section 3, where rigorous guarantees of its success and its cost are provided. Section 4 illustrates the challenges of computing $\mu$ to a guaranteed precision through several numerical examples. This paper ends with a discussion of our results and further work to be done.

## 2 Background probability and statistics

In our Monte Carlo applications, a quantity of interest is written as an expectation: $\mu = \mathbb{E}(Y)$, where $Y$ is a real valued random variable. As mentioned above, very often $Y = f(\boldsymbol{X})$ where $\boldsymbol{X} \in \mathbb{R}^d$ is a random vector with probability density function $\rho$. In other settings the random quantity $\boldsymbol{X}$ might have a discrete distribution or be infinite dimensional (e.g,. a Gaussian process) or both. For Monte Carlo estimation, we can work with the distribution of $Y$ alone. The Monte Carlo estimate of $\mu$ is the sample mean, as given in (2), where the $Y_i$ are IID random variables with the same distribution as $Y$.

### *2.1 Moments*

Our methods require conditions on the first four moments of $Y$ as described here. The variance of $Y$, as defined in (3), is denoted by $\sigma^2$, and its non-negative square root, $\sigma$, is the standard deviation of $Y$. Some of our expressions assume without stating it that $\sigma > 0$, and all will require $\sigma < \infty$. The skewness of $Y$ is $\gamma = \mathbb{E}[(Y - \mu)^3]/\sigma^3$, and the kurtosis of $Y$ is $\kappa = \tilde{\kappa} - 3 = \mathbb{E}[(Y - \mu)^4]/\sigma^4 - 3$ (see (4)). The mysterious 3 in $\kappa$ is there to make it zero for Gaussian random variables. Also, $\mu, \sigma^2, \gamma, \kappa$ are related to the first four cumulants [12, Chap. 2] of the distribution of $Y$, meaning that

$$\log(\mathbb{E}[\exp(tY)]) = \mu t + \frac{\sigma^2 t^2}{2} + \frac{\gamma \sigma^3 t^3}{3!} + \frac{\kappa \sigma^4 t^4}{4!} + o(t^4).$$

Our main results require a known upper bound for $\kappa$, which then implies that $\sigma$ and $\gamma$ are finite.

### 2.2 CLT intervals

A random variable $Z$ has the standard normal distribution, denoted by $\mathcal{N}(0,1)$, if

$$\Pr(Z \le z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} \exp(-t^2/2)\,\mathrm{d}t =: \Phi(z).$$

Under the central limit theorem, the distribution of $\sqrt{n}(\hat{\mu}_n - \mu)/\sigma$ approaches $\mathcal{N}(0,1)$ as $n \to \infty$, where $\hat{\mu}_n$ denotes the sample mean of $n$ IID samples. As a result

$$\Pr\bigl(\hat{\mu}_n - 2.58\sigma/\sqrt{n} \le \mu \le \hat{\mu}_n + 2.58\sigma/\sqrt{n}\bigr) \to 0.99 \tag{9}$$

as $n \to \infty$. We write the interval in (9) as $\hat{\mu}_n \pm 2.58\sigma/\sqrt{n}$. Equation (9) cannot be used when $\sigma^2$ is unknown, but the usual estimate

$$s_n^2 = \frac{1}{n-1} \sum_{i=1}^{n} (Y_i - \hat{\mu}_n)^2 \tag{10}$$

may be substituted, yielding the interval $\hat{\mu}_n \pm 2.58 s_n/\sqrt{n}$ which also satisfies the limit in (9) by Slutsky's theorem [8]. For an arbitrary confidence level $1-\alpha \in (0,1)$, we replace the constant 2.58 by $z_{\alpha/2} = \Phi^{-1}(1-\alpha/2)$. The width of this interval is $2z_{\alpha/2}s_n/\sqrt{n}$, and when $\mu$ is in the interval then the absolute error $|\mu - \hat{\mu}_n| \le \varepsilon := z_{\alpha/2}s_n/\sqrt{n}$.

The coverage level of the CLT interval is only asymptotic. In more detail, Hall [6, p. 948] shows that

$$\Pr\bigl(|\mu - \hat{\mu}_n| \le 2.58 s/\sqrt{n}\bigr) = 0.99 + \frac{1}{n}(A + B\gamma^2 + C\kappa) + O\Bigl(\frac{1}{n^2}\Bigr) \tag{11}$$

for constants $A$, $B$, and $C$ that depend on the desired coverage level (here 99%). Hall's theorem requires only that the random variable $Y$ has sufficiently many finite moments and is not supported solely on a lattice (such as the integers). It is interesting to note that the $O(1/n)$ coverage error in (11) is better than the $O(1/\sqrt{n})$ root mean squared error for the estimate $\hat{\mu}_n$ itself.

## 2.3 Standard Probability Inequalities

Here we present some well known inequalities that we will use. First, Chebychev's inequality ensures that a random variable (such as $\hat{\mu}_n$) is seldom too far from its mean.

**Theorem 1 (Chebychev's Inequality).** *[10, 6.1c, p. 52] Let Z be a random variable with mean $\mu$ and variance $\sigma^2 \geq 0$. Then for all $\varepsilon > 0$,*

$$\Pr[|Z-\mu| \geq \varepsilon] \leq \frac{\sigma^2}{\varepsilon^2}.$$

In some settings we need a one sided inequality like Chebychev's. We will use this one due to Cantelli.

**Theorem 2 (Cantelli's Inequality).** *[10, 6.1e, p. 53] Let Z be any random variable with mean $\mu$ and finite variance $\sigma^2$. For any $a \geq 0$, it follows that:*

$$\Pr[Z-\mu \geq a] \leq \frac{\sigma^2}{a^2+\sigma^2}.$$

Berry-Esseen type theorems govern the rate at which a CLT takes hold. We will use the following theorem which combines recent work on both uniform and non-uniform ($x$-dependent right hand side) versions.

**Theorem 3 (Berry-Esseen Inequality).** *Let $Y_1,\dots,Y_n$ be IID random variables with mean $\mu$, variance $\sigma^2 > 0$, and third centered moment $M_3 = E\left|Y_i-\mu\right|^3/\sigma^3 < \infty$. Let $\hat{\mu}_n = (Y_1+\cdots+Y_n)/n$ denote the sample mean. Then*

$$\left|\Pr\left[\frac{\hat{\mu}-\mu}{\sigma/\sqrt{n}} < x\right] - \Phi(x)\right|$$
$$\leq \Delta_n(x,M_3) := \frac{1}{\sqrt{n}}\min\left(A_1(M_3+A_2), \frac{A_3M_3}{1+|x|^3}\right) \qquad \forall x \in \mathbb{R},$$

*where $A_1 = 0.3328$ and $A_2 = 0.429$ [23], and $A_3 = 18.1139$ [15].*

The constants in the Berry-Esseen Inequality above have been an area of active research. We would not be surprised if there are further improvements in the near future.

Our method requires probabilistic bounds on the sample variance, $s_n^2$. For that, we will use some moments of the variance estimate.

**Theorem 4.** *[13, Eq. (7.16), p. 265] Let $Y_1,\dots,Y_n$ be IID random variables with variance $\sigma^2$ and modified kurtosis $\tilde{\kappa}$ defined in* (4)*. Let $s_n^2$ be the sample variance as defined in* (10)*. Then the sample variance is unbiased, $\mathbb{E}(s_n^2) = \sigma^2$, and its variance is*

$$\mathrm{Var}(s_n^2) = \frac{\sigma^4}{n}\left(\tilde{\kappa} - \frac{n-3}{n-1}\right).$$

## 3 Two-stage confidence interval

Our two-stage procedure works as follows. In the first stage, we take a sample of independent values $Y_1, \dots, Y_{n_\sigma}$ from the distribution of $Y$. From this sample we compute the sample variance, $s^2_{n_\sigma}$, according to (10) and estimate the variance of $Y_i$ by $\hat{\sigma}^2 = \mathfrak{C}^2 \hat{s}^2_{n_\sigma}$, where $\mathfrak{C}^2 > 1$ is a "variance inflation factor" that will reduce the probability that we have underestimated $\sigma^2 = \mathrm{Var}(Y)$. For the second stage, we use the estimate $\hat{\sigma}^2$ as if it were the true variance of $Y_i$ and use Berry-Esseen theorem to obtain a suitable sample size, $n_\mu$, for computing the sample average, $\hat{\mu}$, that satisfies the fixed width confidence interval (1).

The next two subsections give details of these two steps that will let us bound their error probabilities. Then we give a theorem on the method as a whole.

### *3.1 Conservative variance estimates*

We need to ensure that our first stage estimate of the variance $\sigma^2$ is not too small. The following result bounds the probability of such an underestimate.

**Lemma 1.** *Let $Y_1, \dots, Y_n$ be IID random variables with variance $\sigma^2 > 0$ and kurtosis $\kappa$. Let $s^2_n$ be the sample variance defined at* (10)*, and let $\tilde{\kappa} = \kappa + 3$. Then*

$$\Pr\left[s^2_n < \sigma^2 \left\{1 + \sqrt{\left(\tilde{\kappa} - \frac{n-3}{n-1}\right)\left(\frac{1-\alpha}{\alpha n}\right)}\right\}\right] \geq 1 - \alpha, \tag{12a}$$

$$\Pr\left[s^2_n > \sigma^2 \left\{1 - \sqrt{\left(\tilde{\kappa} - \frac{n-3}{n-1}\right)\left(\frac{1-\alpha}{\alpha n}\right)}\right\}\right] \geq 1 - \alpha. \tag{12b}$$

*Proof.* Applying Theorem 4 and choosing

$$a = \sqrt{\mathrm{Var}(s^2_n)\frac{1-\alpha}{\alpha}} = \sigma^2 \sqrt{\left(\tilde{\kappa} - \frac{n-3}{n-1}\right)\left(\frac{1-\alpha}{\alpha n}\right)} > 0,$$

it follows from Cantelli's inequality (Theorem 2) that

$$\begin{aligned}\Pr\left[s^2_n - \sigma^2 \geq \sigma^2 \sqrt{\left(\tilde{\kappa} - \frac{n-3}{n-1}\right)\left(\frac{1-\alpha}{\alpha n}\right)}\right] &= \Pr\left[s^2_n - \sigma^2 \geq a\right] \\ &\leq \frac{\mathrm{Var}(s^2_n)}{a^2 + \mathrm{Var}(s^2_n)} = \frac{\mathrm{Var}(s^2_n)}{\mathrm{Var}(s^2_n)\frac{1-\alpha}{\alpha} + \mathrm{Var}(s^2_n)} = \frac{1}{\left(\frac{1-\alpha}{\alpha}\right) + 1} = \alpha.\end{aligned}$$

Then (12a) follows directly. By a similar argument, applying Cantelli's inequality to the expression $\Pr\left[-s^2_n + \sigma^2 \geq a\right]$ implies (12b). □

Using Lemma 1 we can bound the probability that $\hat{\sigma}^2 = \mathfrak{C}^2 s_{n_\sigma}^2$ overestimates $\sigma^2$. Equation (12a) implies that

$$\Pr\left[\frac{s_{n_\sigma}^2}{1-\sqrt{\left(\tilde{\kappa}-\frac{n_\sigma-3}{n_\sigma-1}\right)\left(\frac{1-\alpha}{\alpha n_\sigma}\right)}} > \sigma^2\right] \geq 1-\alpha.$$

Thus, it makes sense for us to require the modified kurtosis, $\tilde{\kappa}$, to be small enough, relative to $n_\sigma$, $\alpha$, and $\mathfrak{C}$, in order to ensure that $\Pr(\hat{\sigma}^2 > \sigma^2) \geq 1-\alpha$. Specifically, we require

$$\frac{1}{1-\sqrt{\left(\tilde{\kappa}-\frac{n_\sigma-3}{n_\sigma-1}\right)\left(\frac{1-\alpha}{\alpha n_\sigma}\right)}} \leq \mathfrak{C}^2,$$

or equivalently,

$$\tilde{\kappa} \leq \frac{n_\sigma-3}{n_\sigma-1} + \left(\frac{\alpha n_\sigma}{1-\alpha}\right)\left(1-\frac{1}{\mathfrak{C}^2}\right)^2 =: \tilde{\kappa}_{\max}(\alpha, n_\sigma, \mathfrak{C}). \tag{13}$$

This condition is the explicit version of (4) mentioned in the introduction.

## 3.2 Conservative interval widths

Here we consider how to choose the sample size $n_\mu$ to get the desired coverage level from an interval with half-length at most $\varepsilon$. We suppose here that $\sigma$ is known. In practice we will use a conservative (biased high) estimate for $\sigma$.

First, if the CLT held exactly and not just asymptotically, then we could use a CLT sample size of

$$N_{\mathrm{CLT}}(\varepsilon, \sigma, \alpha) = \left\lceil \left(\frac{z_{\alpha/2}\sigma}{\varepsilon}\right)^2 \right\rceil$$

independent values of $Y_i$ in an interval like the one in (9).

Given knowledge of $\sigma$, but no assurance of a Gaussian distribution for $\hat{\mu}_n$, we could instead select a sample size based on Chebychev's inequality (Theorem 1). Taking

$$N_{\mathrm{Cheb}}(\varepsilon, \sigma, \alpha) = \left\lceil \frac{\sigma^2}{\alpha\varepsilon^2} \right\rceil \tag{14}$$

IID observations of $Y$ gives the confidence interval (1). Naturally $N_{\mathrm{Cheb}} \geq N_{\mathrm{CLT}}$.

Finally, we could use the non-uniform Berry-Esseen inequality from Theorem 3. This inequality requires a finite scaled third moment $M_3 = E\left|Y_i-\mu\right|^3/\sigma^3$. If $\hat{\mu}_n$ denotes a sample mean of $n$ IID random instances of $Y$, then the non-uniform Berry-Esseen inequality implies that

$$\begin{aligned}\Pr[|\mu - \hat{\mu}_n| \le \varepsilon] &= \Pr\left[\frac{\hat{\mu}_n - \mu}{\sigma/\sqrt{n}} \le \frac{\sqrt{n}\varepsilon}{\sigma}\right] - \Pr\left[\frac{\hat{\mu}_n - \mu}{\sigma/\sqrt{n}} < -\frac{\sqrt{n}\varepsilon}{\sigma}\right] \\ &\ge \left[\Phi(\sqrt{n}\varepsilon/\sigma) - \Delta_n(\sqrt{n}\varepsilon/\sigma, M_3)\right] \\ &\qquad - \left[\Phi(-\sqrt{n}\varepsilon/\sigma) + \Delta_n(-\sqrt{n}\varepsilon/\sigma, M_3)\right] \\ &= 1 - 2[\Phi(-\sqrt{n}\varepsilon/\sigma) + \Delta_n(\sqrt{n}\varepsilon/\sigma, M_3)], \end{aligned} \tag{15}$$

since $\Delta_n(-x, M_3) = \Delta_n(x, M_3)$. The probability of making an error no greater than $\varepsilon$ is bounded below by $1-\alpha$, i.e., the fixed width confidence interval (1) holds with $\hat{\mu} = \hat{\mu}_n$, provided $n \ge N_{\mathrm{BE}}(\varepsilon, \sigma, \alpha, M_3)$, where the Berry-Esseen sample size is

$$N_{\mathrm{BE}}(\varepsilon, \sigma, \alpha, M_3) := \min\left\{n \in \mathbb{N} : \Phi\left(-\sqrt{n}\varepsilon/\sigma\right) + \Delta_n(\sqrt{n}\varepsilon/\sigma, M_3) \le \frac{\alpha}{2}\right\}. \tag{16}$$

To compute $N_{\mathrm{BE}}(\varepsilon, \sigma, \alpha, M_3)$, we need to know $M_3$. In practice, substituting an upper bound on $M_3$ yields an upper bound on the necessary sample size.

Note that if the $\Delta_n$ term in (16) were absent, $N_{\mathrm{BE}}$ would correspond to the CLT sample size $N_{\mathrm{CLT}}$, and in general $N_{\mathrm{BE}} > N_{\mathrm{CLT}}$. It is possible that in some situations $N_{\mathrm{BE}} > N_{\mathrm{Cheb}}$ might hold, and in such cases we could use $N_{\mathrm{Cheb}}$ instead of $N_{\mathrm{BE}}$.

### *3.3 Algorithm and Proof of Its Success*

In detail, the two-stage algorithm works as described below.

**Algorithm 1 (Two Stage).** The user specifies four quantities:

- an initial sample size for variance estimation, $n_\sigma \in \{2, 3, \ldots\}$,
- a variance inflation factor $\mathfrak{C}^2 \in (1, \infty)$,
- an uncertainty $\alpha \in (0, 1)$, and,
- an error tolerance or confidence interval half-width, $\varepsilon > 0$.

At the first stage of the algorithm, $Y_1, \ldots, Y_{n_\sigma}$ are sampled independently from the same distribution as $Y$. Then the conservative variance estimate, $\hat{\sigma}^2 = \mathfrak{C}^2 s^2_{n_\sigma}$, is computed in terms of the sample variance, $s^2_{n_\sigma}$, defined by (10).

To prepare for the second stage of the algorithm we compute $\tilde{\alpha} = 1 - \sqrt{1-\alpha}$ and then $\tilde{\kappa}_{\max} = \tilde{\kappa}_{\max}(\tilde{\alpha}, n_\sigma, \mathfrak{C})$ using equation (13). The sample size for the second stage is

$$n_\mu = N_\mu(\varepsilon, \hat{\sigma}, \tilde{\alpha}, \tilde{\kappa}^{3/4}_{\max}), \tag{17}$$

where

$$N_\mu(\varepsilon, \sigma, \alpha, M) := \max\bigl(1, \min\bigl(N_{\mathrm{Cheb}}(\varepsilon, \sigma, \alpha), N_{\mathrm{BE}}(\varepsilon, \sigma, \alpha, M)\bigr)\bigr). \tag{18}$$

Recall that $N_{\mathrm{Cheb}}$ is defined in (14) and $N_{\mathrm{BE}}$ is defined in (16).

After this preparation, the second stage is to sample $Y_{n_\sigma+1}, \ldots, Y_{n_\sigma+n_\mu}$ independently from the distribution of $Y$, and independently of $Y_1, \ldots, Y_{n_\sigma}$. The algorithm then returns the sample mean,

$$\hat{\mu} = \frac{1}{n_\mu} \sum_{i=n_\sigma+1}^{n_\sigma+n_\mu} Y_i. \tag{19}$$

The success of this algorithm is guaranteed in the following theorem. The main assumption needed is an upper bound on the kurtosis.

**Theorem 5.** *Let $Y$ be a random variable with mean $\mu$, and either zero variance or positive variance with modified kurtosis $\tilde{\kappa} \le \tilde{\kappa}_{\max}(\tilde{\alpha}, n_\sigma, \mathfrak{C})$. It follows that Algorithm 1 above yields an estimate $\hat{\mu}$ given by* (19) *which satisfies the fixed width confidence interval condition*

$$\Pr(|\hat{\mu} - \mu| \le \varepsilon) \ge 1 - \alpha.$$

*Proof.* If $\sigma^2 = 0$, then $s_{n_\sigma}^2 = 0$, $n_\mu = 1$ and $\hat{\mu} = \mu$ with probability one. Now consider the case of positive variance. The first stage yields a variance estimate satisfying $\Pr(\hat{\sigma}^2 > \sigma^2) \ge 1 - \tilde{\alpha}$ by the argument preceding the kurtosis bound in (13) applied with uncertainty $\tilde{\alpha}$. The second stage yields $\Pr(|\hat{\mu} - \mu| \le \varepsilon) \ge 1 - \tilde{\alpha}$ by the Berry-Esseen result (15), so long as $\hat{\sigma} \ge \sigma$ and $M_3 \le \tilde{\kappa}_{\max}(\tilde{\alpha}, n_\sigma, \mathfrak{C})^{3/4}$. The second condition holds because $M_3 \le \tilde{\kappa}^{3/4}$ by Jensen's Inequality [10, 8.4.b]. Thus, in the two-stage algorithm we have

$$\begin{aligned} \Pr(|\hat{\mu} - \mu| \le \varepsilon) &= \mathbb{E}\big[\Pr(|\hat{\mu} - \mu| \le \varepsilon \mid \hat{\sigma})\big] \\ &\ge \mathbb{E}\left[(1-\tilde{\alpha})1_{\sigma \le \hat{\sigma}}\right] \\ &\ge (1-\tilde{\alpha})(1-\tilde{\alpha}) = 1 - \alpha. \qquad \square \end{aligned}$$

*Remark 1.* As pointed out earlier, the guarantees in this theorem require that the modified kurtosis of $Y$ not exceed the specified upper bound $\tilde{\kappa}_{\max}$. As it is presented, Algorithm 1 takes as inputs, $n_\sigma$, $\mathfrak{C}$, and $\alpha$, and uses these to compute $\tilde{\kappa}_{\max}$ according to (13). The reason for doing so is that one might have a better intuition for $n_\sigma$, $\mathfrak{C}$, and $\alpha$. Alternatively, one may specify $n_\sigma$ and $\tilde{\kappa}_{\max}$ and use (13) to compute $\mathfrak{C}$, or specify $\mathfrak{C}$ and $\tilde{\kappa}_{\max}$ and use (13) to compute $n_\sigma$. The issue of how one should choose $n_\sigma$, $\mathfrak{C}$, and $\tilde{\kappa}_{\max}$ in practice is discussed further in Section 5.

*Remark 2.* In this algorithm it is possible to choose $n_\mu$ much smaller than $n_\sigma$ if the sample variance is small. As a practical matter we suggest that if one is willing to invest $n_\sigma$ samples to estimate the variance then one should be willing to invest at least that many additional samples to estimate the mean. Therefore, in the numerical examples of Section 4 we use

$$N_\mu(\varepsilon, \sigma, \alpha, M) := \max\big(n_\sigma, \min\big(N_{\mathrm{Cheb}}(\varepsilon, \sigma, \alpha), N_{\mathrm{BE}}(\varepsilon, \sigma, \alpha, M)\big)\big) \tag{20}$$

instead of (18) to determine the sample size for the sample mean. Because the variance is typically harder to estimate accurately than the mean, one may wonder whether $n_\sigma$ should be chosen greater than $n_\mu$. However, for Monte Carlo simulation we only need the variance to one or two digits accuracy, whereas we typically want to know the mean to a much higher accuracy. By the error bound following from Chebychev's inequality (Theorem 1), the definition of $N_\mu$ in (20) means that the fixed width confidence interval constructed by Algorithm 1 also holds for any random variables, $Y$, with small variance, namely, $\sigma^2 \le \varepsilon^2 \alpha n_\sigma$, even if its kurtosis is arbitrarily large.

As mentioned in the introduction, one frequently encountered case occurs when $Y$ is a $d$-variate function of a random vector $\boldsymbol{X}$. Then $\mu$ corresponds to the multivariate integral in (6) and Theorem 5 may be interpreted as below:

**Corollary 1.** *Suppose that $\rho : \mathbb{R}^d \to \mathbb{R}$ is a probability density function, the integrand $f : \mathbb{R}^d \to \mathbb{R}$ has finite $\mathcal{L}_4$ norm as defined in* (7)*, and furthermore $f$ lies in the cone $\mathcal{C}_{\tilde{\kappa}_{\max}}$ defined in* (8)*, where $\tilde{\kappa}_{\max} = \tilde{\kappa}_{\max}(\tilde{\alpha}, n_\sigma, \mathfrak{C})$. It follows that Algorithm 1 yields an estimate, $\hat{\mu}$, of the multidimensional integral $\mu$ defined in* (6)*, which satisfies the fixed width confidence interval condition*

$$\Pr(|\hat{\mu} - \mu| \le \varepsilon) \ge 1 - \alpha.$$

## 3.4 Cost of the Algorithm

The number of function values required by the two-stage Algorithm 1 is $n_\sigma + n_\mu$, the sum of the initial sample size used to estimate the variance of $Y$ and the sample size used to estimate the mean of $Y$. Although $n_\sigma$ is deterministic, $n_\mu$ is a random variable, and so the cost of this algorithm might be best defined probabilistically. Moreover, the only random quantity in the formula for $n_\mu$ in (17) is $\hat{\sigma}^2$, the upper bound on variance. Clearly this depends on the unknown population variance, $\sigma^2$, and we expect $\hat{\sigma}^2$ not to overestimate $\sigma^2$ by much. Thus, the algorithm cost is defined below in terms of $\sigma^2$ and the error tolerance (interval half-width) $\varepsilon$. An upper bound on the cost is then derived in Theorem 6.

Let $A$ be any random algorithm that takes as its input, a method for generating random samples, $Y_1, Y_2, \ldots$ with common distribution function $F$ having variance $\sigma^2$ and modified kurtosis $\tilde{\kappa}$. Additional algorithm inputs are an error tolerance, $\varepsilon$, an uncertainty, $\alpha$, and a maximum modified kurtosis, $\tilde{\kappa}_{\max}$. The algorithm then computes $\hat{\mu} = A(F, \varepsilon, \alpha, \tilde{\kappa}_{\max})$, an approximation to $\mu = \mathbb{E}(Y)$, based on a total of $N_{\text{tot}}(\varepsilon, \alpha, \tilde{\kappa}_{\max}, F)$ samples. The probabilistic cost of the algorithm, with uncertainty $\beta$, for integrands of variance no greater than $\sigma^2_{\max}$ and modified kurtosis no greater than $\tilde{\kappa}_{\max}$ is defined as

$$N_{\text{tot}}(\varepsilon, \alpha, \beta, \tilde{\kappa}_{\max}, \sigma_{\max}) := \sup_{\substack{\tilde{\kappa} \le \tilde{\kappa}_{\max} \\ \sigma \le \sigma_{\max}}} \min\left\{N : \Pr[N_{\text{tot}}(\varepsilon, \alpha, \tilde{\kappa}_{\max}, F) \le N] \ge 1 - \beta\right\}.$$

Note that $\tilde{\kappa}_{\max}$ is an input to the algorithm, but $\sigma_{\max}$ is not. The cost of an arbitrary algorithm, $A$ may also depend on other parameters, such as $n_\sigma$ and $\mathfrak{C}$ in our Algorithm 1, which are related to $\tilde{\kappa}_{\max}$. However, this dependence is not shown explicitly to keep the notation simple.

The cost of the particular two-stage Monte Carlo algorithm defined in Algorithm 1 is

$$\sup_{\substack{\tilde{\kappa}\le\tilde{\kappa}_{\max}\\ \sigma\le\sigma_{\max}}} \min\left\{N : \Pr(n_\sigma + N_\mu(\varepsilon, \hat{\sigma}, \tilde{\alpha}, \tilde{\kappa}_{\max}^{3/4}) \le N) \ge 1-\beta\right\}.$$

Since $n_\sigma$ is fixed, bounding this cost depends on bounding $N_\mu(\varepsilon, \hat{\sigma}, \tilde{\alpha}, \tilde{\kappa}_{\max}^{3/4})$, which depends on $\hat{\sigma}$ as given by Algorithm 1. Moreover, $\hat{\sigma}$ can be bounded above using (12a) in Lemma 1. For $\tilde{\kappa} \le \tilde{\kappa}_{\max}$,

$$\begin{aligned}
1-\beta &\le \Pr\left[s_{n_\sigma}^2 < \sigma^2\left\{1+\sqrt{\left(\tilde{\kappa}-\frac{n_\sigma-3}{n_\sigma-1}\right)\left(\frac{1-\beta}{\beta n_\sigma}\right)}\right\}\right]\\
&\le \Pr\left[\hat{\sigma}^2 = \mathfrak{C}^2 s_{n_\sigma}^2 < \mathfrak{C}^2\sigma^2\left\{1+\sqrt{\left(\tilde{\kappa}_{\max}(n_\sigma,\tilde{\alpha},\mathfrak{C})-\frac{n_\sigma-3}{n_\sigma-1}\right)\left(\frac{1-\beta}{\beta n_\sigma}\right)}\right\}\right]\\
&= \Pr\left[\hat{\sigma}^2 < \sigma^2 v^2(\tilde{\alpha},\beta,\mathfrak{C})\right],
\end{aligned}$$

where

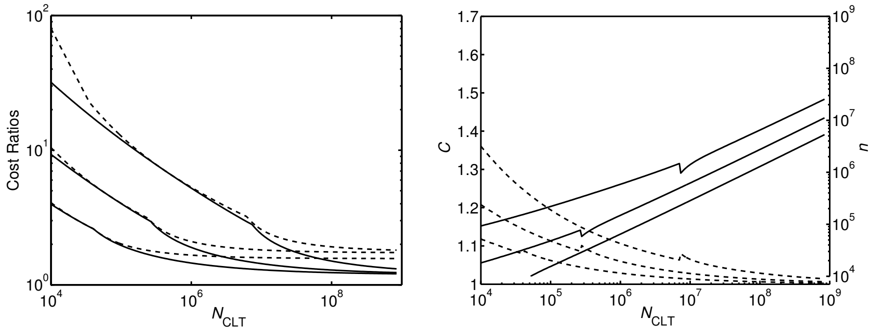
$$v^2(\tilde{\alpha},\beta,\mathfrak{C}) := \mathfrak{C}^2 + \left(\mathfrak{C}^2-1\right)\sqrt{\frac{\tilde{\alpha}(1-\beta)}{(1-\tilde{\alpha})\beta}} > 1.$$

Noting that $N_\mu(\varepsilon, \cdot, \tilde{\alpha}, \tilde{\kappa}_{\max}^{3/4})$ is a non-decreasing function allows one to derive the following upper bound on the cost of the adaptive Monte Carlo algorithm.

**Theorem 6.** *The two-stage Monte Carlo algorithm for fixed width confidence intervals based on IID sampling described in Algorithm 1 has a probabilistic cost bounded above by*

$$\begin{aligned}
&N_{\text{tot}}(\varepsilon,\alpha,\beta,\tilde{\kappa}_{\max},\sigma_{\max})\\
&\qquad \le N_{\text{up}}(\varepsilon,\alpha,\beta,\tilde{\kappa}_{\max},\sigma_{\max}) := n_\sigma + N_\mu(\varepsilon,\sigma_{\max}v(\tilde{\alpha},\beta,\mathfrak{C}),\tilde{\alpha},\tilde{\kappa}_{\max}^{3/4}).
\end{aligned}$$

Note that the Chebychev sample size, $N_{\text{Cheb}}$, defined in (14), the Berry-Esseen sample size, $N_{\text{BE}}$, defined in (16), and thus $N_\mu$ all depend on $\sigma$ and $\varepsilon$ through their ratio, $\sigma/\varepsilon$. Thus, ignoring the initial sample used to estimate the variance, $N_{\text{tot}}(\varepsilon,\alpha,\beta,\tilde{\kappa}_{\max},\sigma_{\max})$ is roughly proportional to $\sigma_{\max}^2/\varepsilon^2$, even though $\sigma_{\max}$ is not a parameter of the algorithm. Algorithm 1 *adaptively* determines the sample size, and thus the cost, to fit the unknown variance of $Y$. Random variables, $Y$, with small variances will require a lower cost to estimate $\mu$ with a given error tolerance than random variables with large variances.



(a) (b)

**Fig. 1** (a) The cost ratios of $N_{\mathrm{up}}(\varepsilon, 0.01, 0.01, \tilde{\kappa}_{\max}, \sigma)/N_{\mathrm{CLT}}(\varepsilon, \sigma, 0.01)$ for $\tilde{\kappa}_{\max} = 2, 10$, and 100, with $n_\sigma = 4000\tilde{\kappa}_{\max}$ (dashed) and $n_\sigma$ optimized (solid); (b) the optimal values of $n_\sigma$ (solid) and $\mathfrak{C}$ (dashed).

Figure 1(a) shows the ratio of the upper bound of the cost, $N_{\mathrm{up}}(\varepsilon, 0.01, 0.01, \tilde{\kappa}_{\max}, \sigma)$, to the ideal CLT cost, $N_{\mathrm{CLT}}(\varepsilon, \sigma, 0.01) = \lceil (2.58\sigma/\varepsilon)^2 \rceil$, for a range of $\sigma/\varepsilon$ ratios and for $\tilde{\kappa}_{\max} = 2, 10$, and 100. In these graphs the formula defining $N_{\mathrm{up}}$ in Theorem 6 uses the alternative and somewhat costlier formula for $N_\mu$ in (20). The dashed curves in Figure 1(a) show these cost ratios with $n_\sigma = 4000\tilde{\kappa}_{\max}$, which corresponds to $\mathfrak{C} \approx 1.1$. The solid curves denote the case where $n_\sigma$ and $\mathfrak{C}$ vary with $\sigma/\varepsilon$ to minimize $N_{\mathrm{up}}$. Figure 1(b) displays the optimal values of $n_\sigma$ (solid) and $\mathfrak{C}$ (dashed). In both figures, higher curves correspond to higher values of $\tilde{\kappa}_{\max}$.

Here, $N_{\mathrm{CLT}}$ denotes the ideal cost if one knew the variance of $Y$ a priori and knew that the distribution of the sample mean was close to Gaussian. The cost ratio is the penalty for having a guaranteed fixed width confidence interval in the absence of this knowledge about the distribution of $Y$. For smaller values of $N_{\mathrm{CLT}}$, equivalently smaller $\sigma/\varepsilon$, this cost ratio can be rather large. However the absolute effect of this large penalty is mitigated by the fact that the total number of samples needed is not much. For larger $N_{\mathrm{CLT}}$, equivalently larger $\sigma/\varepsilon$, the cost ratio approaches somewhat less than 1.4 in the case of optimal $n_\sigma$ and $\mathfrak{C}$, and somewhat less than 2 for $n_\sigma = 1000\tilde{\kappa}_{\max}$.

The discontinuous derivatives in the curves in Figure 1 arise from the minimum and maximum values arising in formulas (16) and (20) for $N_{\mathrm{BE}}$ and $N_\mu$, respectively. Taking the upper dashed curve in Figure 1(a) as an example, for $N_{\mathrm{CLT}}$ less than about $3.5 \times 10^4$, $N_\mu = n_\sigma$. For $N_{\mathrm{CLT}}$ from about $3.5 \times 10^4$ to about $6 \times 10^6$, $N_\mu$ corresponds to the second term in the minimum in the Berry-Esseen inequality, (16), i.e., the non-uniform term. For $N_{\mathrm{CLT}}$ greater than $6 \times 10^6$, $N_\mu$ corresponds to the first term in the minimum in the Berry-Esseen inequality, (16), i.e., the uniform term.

The ideal case of optimizing $n_\sigma$ and $\mathfrak{C}$ with respect to $\sigma/\varepsilon$ is impractical, since $\sigma$ is not known in advance. Our suggestion is to choose $\mathfrak{C}$ around 1.1, and then choose $n_\sigma$ as large as needed to ensure that $\tilde{\kappa}_{\max}$ is as large as desired. For example with $\mathfrak{C} = 1.1$ and $\tilde{\kappa}_{\max} = 2, 10$, and 100 we get $n_\sigma = 6593$, 59311, and 652417 respectively.

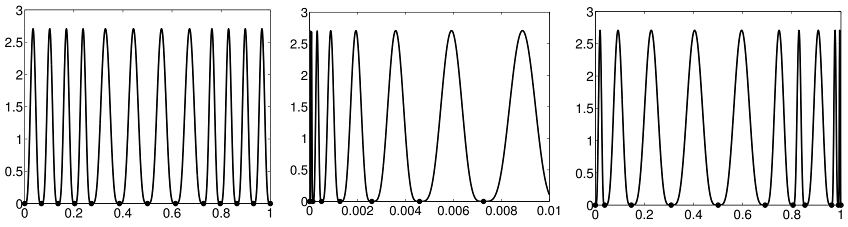
# 4 Numerical Examples

## *4.1 Univariate Fooling Functions for Deterministic Algorithms*

Several commonly used software packages have automatic algorithms for integrating functions of a single variable. These include

- `quad` in MATLAB [28], adaptive Simpson's rule based on `adaptsim` by Gander and Gautschi [4],
- `quadgk` in MATLAB [28], adaptive Gauss-Kronrod quadrature based on `quadva` by Shampine [22], and
- the `chebfun` [5] toolbox for MATLAB [28], which approximates integrals by integrating interpolatory Chebychev polynomial approximations to the integrands.

For these three automatic algorithms one can easily probe where they sample the integrand, feed the algorithms zero values, and then construct fooling functions for which the automatic algorithms will return a zero value for the integral. Figure 2 displays these fooling functions for the problem $\mu = \int_0^1 f(x)\,\mathrm{d}x$ for these three algorithms. Each of these algorithms is asked to provide an answer with an absolute error no greater than $10^{-14}$, but in fact the absolute error is 1 for these fooling functions. The algorithms `quad` and `chebfun` sample only about a dozen points before concluding that the function is zero, whereas the algorithm `quadgk` samples a much larger number of points (only those between 0 and 0.01 are shown in the plot).



`quad` `quadgk` `chebfun`

**Fig. 2** Plots of fooling functions, $f$, with $\mu = \int_0^1 f(x)\,\mathrm{d}x = 1$, but for which the corresponding algorithms return values of $\hat{\mu} = 0$.

### 4.2 Integrating a Single Hump

Accuracy and timing results have been recorded for the integration problem $\mu = \int_{[0,1]^d} f(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$ for a single hump test integrand

$$f(\boldsymbol{x}) = a_0 + b_0 \prod_{j=1}^{d} \left[ 1 + b_j \exp\left( -\frac{(x_j - h_j)^2}{c_j^2} \right) \right]. \tag{21}$$

Here $\boldsymbol{x}$ is a $d$ dimensional vector, and $a_0, b_0, \ldots, b_d, c_1, \ldots, c_d, h_1, \ldots, h_d$ are parameters. Figures 3 and 4 show the results of different algorithms being used to integrate 500 different instances of $f$. For each instance of $f$, the parameters are chosen as follows:

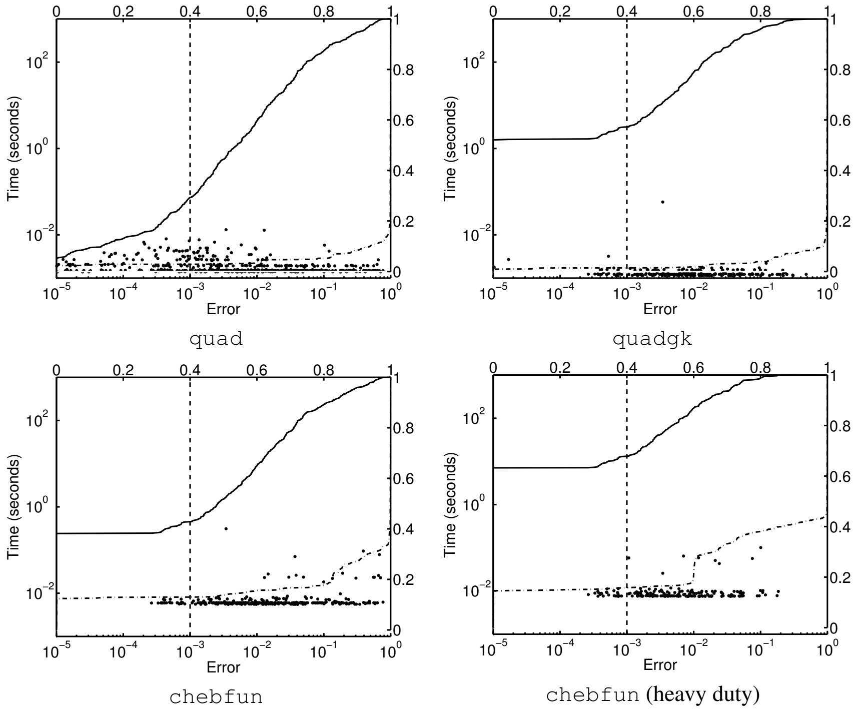
- $b_1, \ldots, b_d \in [0.1, 10]$ with $\log(b_j)$ being i.i.d. uniform,
- $c_1, \ldots, c_d \in [10^{-6}, 1]$ with $\log(c_j)$ being i.i.d. uniform,
- $h_1, \ldots, h_d \in [0, 1]$ with $h_j$ being i.i.d. uniform,
- $b_0$ chosen in terms of the $b_1, \ldots, b_d, c_1, \ldots, c_d, h_1, \ldots, h_d$ to make $\sigma^2 = \|f - \mu\|_2^2 \in [10^{-2}, 10^2]$, with $\log(\sigma)$ being i.i.d. uniform for each instance, and
- $a_0$ chosen in terms of the $b_0, \ldots, b_d, c_1, \ldots, c_d, h_1, \ldots, h_d$ to make $\mu = 1$.

These ranges of parameters are chosen so that the algorithms being tested fail to meet the error tolerance a significant number of times.

These 500 random constructions of $f$ with $d = 1$ are integrated using `quad`, `quadgk`, `chebfun`, Algorithm 1, and an automatic quasi-Monte Carlo algorithm that uses scrambled Sobol' sampling [3, 7, 11, 17, 18, 19]. For the Sobol' sampling algorithm the error is estimated by an inflation factor of 1.1 times the sample standard deviation of 8 internal replicates of one scrambled Sobol' sequence [20]. The sample size is increased until this error estimate decreases to no more than the tolerance. We have not yet found simple conditions on integrands for which this procedure is guaranteed to produce an estimate satisfying the error tolerance, and so we do not discuss it in detail. We are however, intrigued by the fact that it does seem to perform rather well in practice.

For all but `chebfun`, the specified absolute error tolerance is $\varepsilon = 0.001$. The algorithm `chebfun` attempts to do all calculations to near machine precision. The observed error and execution times are plotted in Figures 3 and 4. Whereas `chebfun` uses a minimum of $2^3 + 1 = 9$ function values, the figure labeled "`chebfun` (heavy duty)" displays the results of requiring `chebfun` to use at least $2^8 + 1 = 257$ function values. Algorithm 1 takes $\alpha = 0.01$, and $\mathfrak{C} = 1.1$. For the plot on the left, $n_\sigma = 2^{13} = 8192$, which corresponds to $\tilde{\kappa}_{\max} = 2.24$. For the heavy duty plot on the right, $n_\sigma = 2^{18} = 262144$, which corresponds to $\tilde{\kappa}_{\max} = 40.1$. The same initial sample sizes are used for the Sobol' sampling algorithm.

Figure 3 shows that `quad` and `quadgk` are quite fast, nearly always providing an answer in less than 0.01 seconds. Unfortunately, they successfully meet the error tolerance only about 30% of the time for `quad` and 50–60% of the time for `quadgk`. The difficult cases are those where $c_1$ is quite small, and these algorithms



quad

quadgk

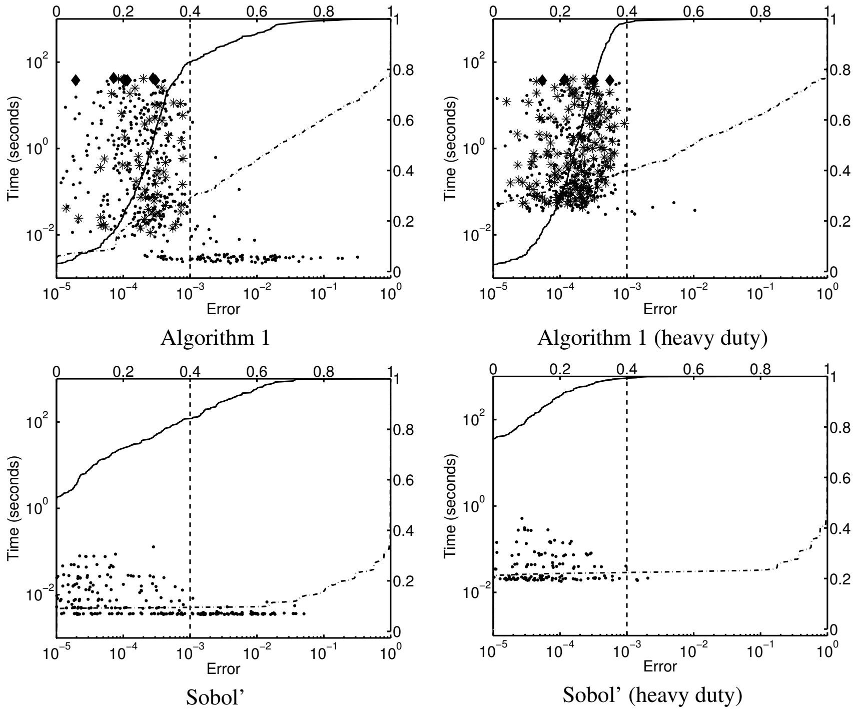
chebfun

chebfun (heavy duty)

**Fig. 3** Execution times and errors for test function (21) for $d = 1$ and error tolerance $\varepsilon = 10^{-3}$, and a variety of parameters giving a range of $\sigma$ and $\tilde{\kappa}$. Those points to the left/right of the dashed vertical line represent successes/failures of the automatic algorithms. The solid line shows that cumulative distribution of actual errors, and the dot-dashed line shows the cumulative distribution of execution times.

miss the sharp peak. The performance of chebfun is similar to that of quad and quadgk. The heavy duty version of chebfun fares somewhat better. For both of the chebfun plots there are a significant proportion of the data that do not appear because their errors are smaller than $10^{-5}$.

In the plots for Algorithm 1 in Figure 4 the alternative and somewhat costlier formula for $N_\mu$ in (20) is employed. An asterisk is used to label those points satisfying $\tilde{\kappa} \le \tilde{\kappa}_{\max}$, where $\tilde{\kappa}$ is defined in (7). All such points fall within the prescribed error tolerance, which is even better than the guaranteed confidence of 99%. For Algorithm 1 (heavy duty) $\tilde{\kappa}_{\max}$ is larger, so there are more points for which the guarantee holds. Those points labeled with a dot, are those for which $\tilde{\kappa} > \tilde{\kappa}_{\max}$, and so no guarantee holds. The points labeled with a diamond are those for which Algorithm 1 attempts to exceed the cost budget that we set, i.e., it wants to choose $n_\mu$ such that $n_\sigma + n_\mu > N_{\max} := 10^9$. In these cases $n_\mu$ is chosen as $\lfloor 10^9 - n_\sigma \rfloor$, which often is still large enough to get an answer that satisfies the error tolerance. Algorithm 1 performs somewhat more robustly than quad, quadgk, and chebfun, because it requires only a low degree of smoothness and takes a fairly large minimum sample.



Algorithm 1

Algorithm 1 (heavy duty)

Sobol'

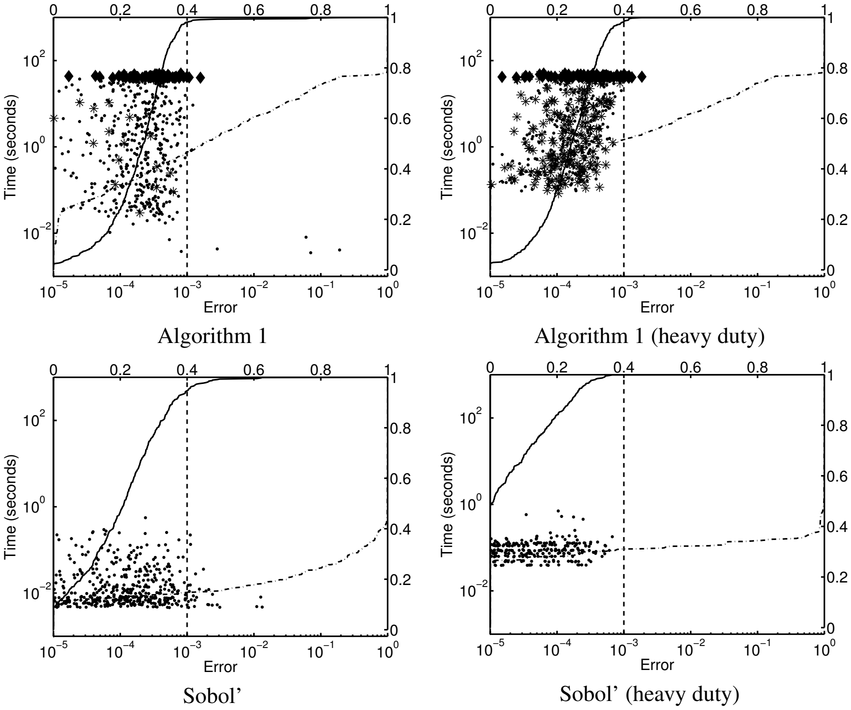
Sobol' (heavy duty)

**Fig. 4** Execution times and errors for test function (21) for $d = 1$ and error tolerance $\varepsilon = 10^{-3}$, and a variety of parameters giving a range of $\sigma$ and $\tilde{\kappa}$. Those points to the left/right of the dashed vertical line represent successes/failures of the automatic algorithms. The solid line shows that cumulative distribution of actual errors, and the dot-dashed line shows the cumulative distribution of execution times. For Algorithm 1 the points labeled * are those for which the Corollary 1 guarantees the error tolerance.

Algorithm 1 is generally much slower than the other algorithms because it does not assume any smoothness of the integrand. The more important point is that Algorithm 1 has a guarantee, whereas to our knowledge, the other routines do not.

From Figure 4, the Sobol' sampling algorithm is more reliable and takes less time than Algorithm 1. This is due primarily to the fact that in dimension one, Sobol' sampling is equivalent to stratified sampling, where the points are more evenly spread than IID sampling.

Figure 5 repeats the simulation shown in Figure 4 for the same test function (21), but now with $d = 2, \dots, 8$ chosen randomly and uniformly. For this case the univariate integration algorithms are inapplicable, but the multidimensional routines can be used. There are more cases where the Algorithm 1 tries to exceed the maximum sample size allowed, i.e., $(n_\sigma + n_\mu)d > N_{\max} := 10^9$, but the behavior seen for $d = 1$ still generally applies.



Algorithm 1

Algorithm 1 (heavy duty)

Sobol'

Sobol' (heavy duty)

**Fig. 5** Execution times and errors for test function (21) for $d = 2,\ldots,8$ and $\varepsilon = 10^{-3}$, with the rest of the parameters as in Figure 4.

## 4.3 Asian Geometric Mean Call Option Pricing

The next example involves pricing an Asian geometric mean call option. Suppose that the price of a stock $S$ at time $t$ follows a geometric Brownian motion with constant interest rate, $r$, and constant volatility, $v$. One may express the stock price in terms of the initial condition, $S(0)$, as

$$S(t) = S(0)\exp[(r - v^2/2)t + vB(t)], \qquad t \geq 0,$$

where $B$ is a standard Brownian motion. The discounted payoff of the Asian geometric mean call option with an expiry of $T$ years, a strike price of $K$, and assuming a discretization at $d$ times is

$$Y = \max\left( [\sqrt{S(0)}S(T/d)S(2T/d)\cdots S(T(d-1)/d)\sqrt{S(T)}]^{1/d} - K, 0 \right) \mathrm{e}^{-rT}. \tag{22}$$

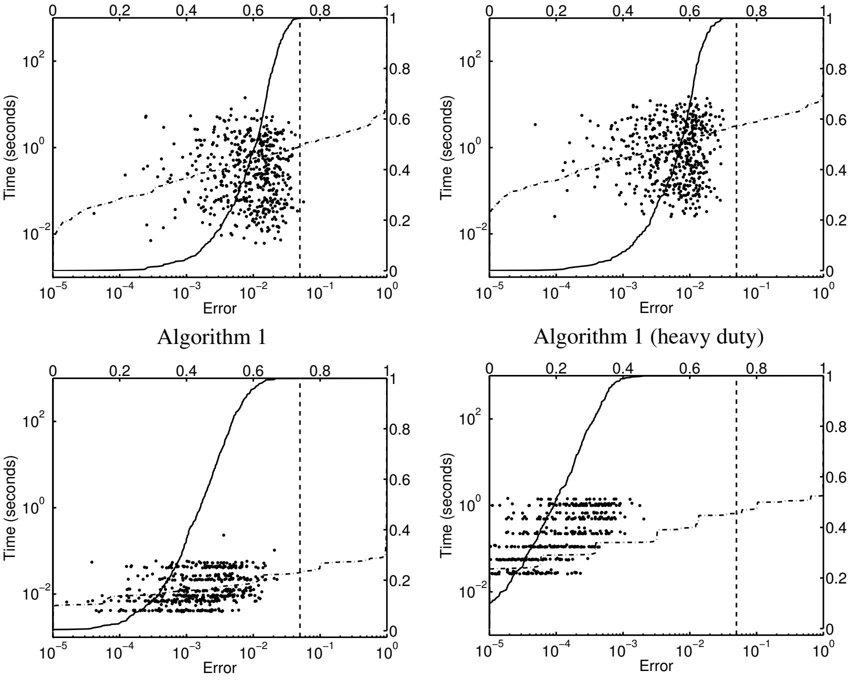
The fair price of this option is $\mu = \mathbb{E}(Y)$. One of our chief reasons for choosing this option for numerical experiments is that its price can be computed analytically, while the numerical computation is non-trivial.

In our numerical experiments, the values of the Brownian motion at different times required for evaluating the stock price, $B(T/d), B(2T/d), \ldots, B(T)$, are computed via a Brownian bridge construction. This means that for one instance of the Brownian motion we first compute $B(T)$, then $B(T/2)$, etc., using independent Gaussian random variables $X_1, \ldots, X_d$, suitably scaled. The Brownian bridge accounts for more of the low frequency motion of the stock price by the $X_j$ with smaller $j$, which allows the Sobol' sampling algorithm to do a better job.

The option price, $\mu = \mathbb{E}(Y)$, is approximated by Algorithm 1 and the Sobol' sampling algorithm using an error tolerance of $\varepsilon = 0.05$, and compared to the analytic value of $\mu$. The result of 500 replications is given in Figure 6. Some of the parameters are set to be fixed values, namely,

$$S(0) = K = 100, \qquad T = 1, \qquad r = 0.03.$$

The volatility, $v$, is drawn uniformly between 0.1 and 0.7. The number of time steps, $d$, is chosen to be uniform over $\{1, 2, 4, 8, 16, 32\}$. The true value of $\mu$ for these parameters is between about 2.8 and 14.



Algorithm 1

Algorithm 1 (heavy duty)

Sobol'

Sobol' (heavy duty)

**Fig. 6** Execution times and errors for the Asian geometric mean call option for $d = 1, 2, 4, 8, 16, 32$ and $\varepsilon = 0.05$.

For this example the true kurtosis of $Y$ is unknown. Both Algorithm 1 and the Sobol' sampling algorithm compute the option price to the desired error tolerance with high reliability. For the IID sampling Algorithm 1 and the ordinary Sobol' sampling algorithm it can be seen that some of the errors are barely under the error tolerance, meaning that the sample size is not chosen too conservatively. For the heavy duty Sobol' algorithm, the high initial sample size seems to lead to smaller than expected errors and larger than necessary computation times.

## 5 Discussion

Practitioners often construct CLT-based confidence intervals with the true variance estimated by the sample variance, perhaps multiplied by some inflation factor. Often, this approach works, but it has no guarantee of success. The two-stage algorithm presented here is similar to the approach just described, but it carries guarantees. These are derived by employing Cantelli's inequality to ensure a reliable variance upper bound, and by employing a Berry-Esseen inequality to ensure a large enough sample for the sample mean.

In certain cases our procedure multiplies the computational cost by a large factor such as 2 or 10 or even 100 compared to what one might spend based on the CLT with a known value of $\sigma$ (see Figure 1). While this seems inefficient, one should remember that the total elapsed time may still be well below several seconds. Furthermore, one typically does not know $\sigma$ in advance, and our adaptive algorithm estimates $\sigma$ and then an appropriate sample size $n_\mu$ from the data. Our algorithmic cost will be low when the unknown $\sigma$ is small and large when $\sigma$ is large.

Like any algorithm with guarantees, our algorithm does need to make assumptions about the random variable $Y$. We assume a known bound on the kurtosis of $Y$, either specified directly or implied by the user's choice of the sample size for estimating the variance, $n_\sigma$, and the variance inflation factor, $\mathfrak{C}^2$. This is a philosophical choice. We prefer not to construct an algorithm that assumes a bound on the variance of $Y$, because such an algorithm would not be guaranteed for $cY$ with $|c|$ large enough. If our algorithm works for $Y$, it will also work for $cY$, no matter how large $|c|$ is.

In practice the user may not know a priori if $\tilde{\kappa} \le \tilde{\kappa}_{\max}$ since it is even more difficult to estimate $\tilde{\kappa}$ from a sample than it is to estimate $\sigma^2$. Thus, the choice of $\tilde{\kappa}_{\max}$ relies on the user's best judgement. Here are a few thoughts that might help. One might try a sample of typical problems for which one knows the answers and use these problems to suggest an appropriate $\tilde{\kappa}_{\max}$. Alternatively, one may think of $\tilde{\kappa}_{\max}$ not as a parameter to be prescribed, but as a reflection of the robustness of one's Monte Carlo algorithm having chosen $\alpha$, $n_\sigma$ and $\mathfrak{C}$. The discussion at the end of Section 3.4 provides guidance on how to choose $n_\sigma$ and $\mathfrak{C}$ to achieve a given $\tilde{\kappa}_{\max}$ in a manner that minimizes total computational cost. Briefly, one should not skimp on $n_\sigma$, but choose $n_\sigma$ to be several thousand times $\tilde{\kappa}_{\max}$ and employ a $\mathfrak{C}$ that is relatively close to unity. Another way to look at the Theorem 5 is that, like a pathologist, it tells you what went wrong if the two-stage adaptive algorithm fails: the kurtosis of the random variable must have been too large. In any case, as one can see in Figure 1, in the limit of vanishing $\varepsilon/\sigma$, i.e., $N_{\text{CLT}} \to \infty$, the choice of $\tilde{\kappa}_{\max}$ makes a negligible contribution to the total cost of the algorithm. The main determinant of computational cost is $\varepsilon/\sigma$.

Bahadur and Savage [1] prove in Corollary 2 that it is *impossible* to construct exact confidence intervals for the mean of random variable whose distribution lies in a set satisfying a few assumptions. One of these assumptions is that the set of distributions is convex. This assumption is violated by our assumption of bounded kurtosis in Theorem 5. Thus, we are able to construct guaranteed confidence intervals.

Our algorithm is adaptive because $n_\mu$ is determined from the sample variance. Information-based complexity theory tells us that adaptive information does not help for the integration problem for symmetric, convex sets of integrands, $f$, in the worst case and probabilistic settings [29, Chapter 4, Theorem 5.2.1; Chapter 8, Corollary 5.3.1]. Here, in Corollary 1 the cone, $\mathcal{C}_{\tilde{\kappa}_{\max}}$, although symmetric, is not a convex set, so it is possible for adaption to help.

There are a couple of areas that suggest themselves for further investigation. One is relative error, i.e., a fixed width confidence interval of the form

$$\Pr[|\mu - \hat{\mu}| \le \varepsilon\,|\mu|] \ge 1 - \alpha.$$

Here the challenge is that the right hand side of the first inequality includes the unknown mean.

Another area for further work is to provide guarantees for automatic quasi-Monte Carlo algorithms. Here the challenge is finding reliable formulas for error estimation. Typical error bounds involve a semi-norm of the integrand that is harder to compute than the original integral. For randomized quasi-Monte Carlo an estimate of the variance of the sample mean using $n$ samples does not tell you much about the variance of the sample mean using a different number of samples.

## Acknowledgements

The first and second authors were partially supported by the National Science Foundation under DMS-0923111 and DMS-1115392. The fourth author was partially supported by the National Science Foundation under DMS-0906056.

The authors gratefully acknowledge discussions with Erich Novak and Henryk Woźniakowski, and the comments of the referees. The plots of the univariate fooling functions were prepared with the help of Nicholas Clancy and Caleb Hamilton. The first and fourth authors would like to express their thanks to the local organizers of the Tenth International Conference on Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing for hosting a wonderful conference.

## References

1. R. R. Bahadur and L. J. Savage, *The nonexistence of certain statistical procedures in nonparametric problems*, Ann. Math. Stat. **27** (1956), 1115–1122.
2. Y. S. Chow and H. Robbins, *On the asymptotic theory of fixed-width sequential confidence intervals for the mean*, Ann. Math. Stat. **36** (1965), 457–462.
3. J. Dick and F. Pillichshammer, *Digital nets and sequences: Discrepancy theory and quasi-Monte Carlo integration*, Cambridge University Press, Cambridge, 2010.
4. W. Gander and W. Gautschi, *Adaptive quadrature — revisited*, BIT **40** (2000), 84–101.
5. N. Hale, L. N. Trefethen, and T. A. Driscoll, *Chebfun version 4*, 2012.
6. P. Hall, *Theoretical comparisons of bootstrap confidence intervals*, Ann. Statist. **16** (1988), no. 3, 927–953.
7. H. S. Hong and F. J. Hickernell, *Algorithm 823: Implementing scrambled digital nets*, ACM Trans. Math. Software **29** (2003), 95–109.
8. E. L. Lehmann and J. P. Romano, *Testing statistical hypotheses*, third ed., Springer, New York, 2005.
9. C. Lemieux, *Monte Carlo and quasi-Monte Carlo sampling*, Springer Science+Business Media, Inc., New York, 2009.
10. Z. Lin and Z. Bai, *Probability inequalities*, Science Press and Springer-Verlag, Beijing and Berlin, 2010.
11. J. Matoušek, *On the $L_2$-discrepancy for anchored boxes*, J. Complexity **14** (1998), 527–556.
12. P. McCullagh, *Tensor methods in statistics*, Chapman and Hall, London, 1987.
13. R. Miller, *Beyond ANOVA, basics of applied statistics*, John Wiley & Sons, Inc., New York, 1986.
14. N. Mukhopadhyay and S. Datta, *On sequential fixed-width confidence intervals for the mean and second-order expansions of the associated coverage probabilities*, Ann. Inst. Statist. Math. **48** (1996), no. 3, 497–507.
15. Yu. S. Nefedova and I. G. Shevtsova, *On non-uniform convergence rate estimates in the central limit theorem*, Theory Probab. Appl. **57** (2012), 62–97.
16. H. Niederreiter, *Random number generation and quasi-Monte Carlo methods*, CBMS-NSF Regional Conference Series in Applied Mathematics, SIAM, Philadelphia, 1992.
17. A. B. Owen, *Randomly permuted $(t,m,s)$-nets and $(t,s)$-sequences*, Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing (H. Niederreiter and P. J.-S. Shiue, eds.), Lecture Notes in Statistics, vol. 106, Springer-Verlag, New York, 1995, pp. 299–317.
18. ———, *Monte Carlo variance of scrambled net quadrature*, SIAM J. Numer. Anal. **34** (1997), 1884–1910.
19. ———, *Scrambled net variance for integrals of smooth functions*, Ann. Stat. **25** (1997), 1541–1562.
20. ———, *On the Warnock-Halton quasi-standard error*, Monte Carlo Methods and Appl. **12** (2006), 47–54.
21. R. J. Serfling and D. D. Wackerly, *Asymptotic theory of sequential fixed-width confidence procedures*, J. Amer. Statist. Assoc. **71** (1976), no. 356, 949–955.
22. L. F. Shampine, *Vectorized adaptive quadrature in Matlab*, J. Comput. Appl. Math. **211** (2008), 131–140.
23. I. Shevtsova, *On the absolute constants in the Berry–Esseen type inequalities for identically distributed summands*, arXiv:1111.6554v1 [math.PR], 2011.
24. D. Siegmund, *Sequential analysis: Tests and confidence intervals*, Springer, New York, 1985.
25. I. H. Sloan and S. Joe, *Lattice methods for multiple integration*, Oxford University Press, Oxford, 1994.
26. C. Stein, *A two sample test for a linear hypothesis whose power is independent of the variance*, Ann. Math. Stat. **16** (1945), 243–258.
27. ———, *Some problems in sequential estimation*, Econometrica **17** (1949), 77–78.
28. The MathWorks, Inc., *MATLAB 7.12*, Natick, MA, 2012.
29. J. F. Traub, G. W. Wasilkowski, and H. Woźniakowski, *Information-based complexity*, Academic Press, Boston, 1988.