

Springer Proceedings in Mathematics & Statistics

Josef Dick
Frances Y. Kuo
Gareth W. Peters
Ian H. Sloan *Editors*

Monte Carlo and Quasi-Monte Carlo Methods 2012

 Springer

Springer Proceedings in Mathematics and Statistics

Volume 65

For further volumes:
<http://www.springer.com/series/10533>

Springer Proceedings in Mathematics and Statistics

This book series features volumes composed of selected contributions from workshops and conferences in all areas of current research in mathematics and statistics, including OR and optimization. In addition to an overall evaluation of the interest, scientific quality, and timeliness of each proposal at the hands of the publisher, individual contributions are all refereed to the high quality standards of leading journals in the field. Thus, this series provides the research community with well-edited, authoritative reports on developments in the most exciting areas of mathematical and statistical research today.

Josef Dick • Frances Y. Kuo • Gareth W. Peters
Ian H. Sloan
Editors

Monte Carlo and Quasi-Monte Carlo Methods 2012

 Springer

Editors

Josef Dick
Frances Y. Kuo
Gareth W. Peters
Ian H. Sloan
The University of New South Wales
School of Mathematics and Statistics
New South Wales
Sydney, Australia

ISSN 2194-1009

ISSN 2194-1017 (electronic)

ISBN 978-3-642-41094-9

ISBN 978-3-642-41095-6 (eBook)

DOI 10.1007/978-3-642-41095-6

Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2013956514

Mathematics Subject Classification (2010): Primary: 11K38, 11K45, 65-06, 65C05, 65C10, 65D30

Secondary: 11K38, 65D18, 65D32, 65R20, 91B25

© Springer-Verlag Berlin Heidelberg 2013

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed. Exempted from this legal reservation are brief excerpts in connection with reviews or scholarly analysis or material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use by the purchaser of the work. Duplication of this publication or parts thereof is permitted only under the provisions of the Copyright Law of the Publisher's location, in its current version, and permission for use must always be obtained from Springer. Permissions for use may be obtained through RightsLink at the Copyright Clearance Center. Violations are liable to prosecution under the respective Copyright Law.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

This volume represents the refereed proceedings of the Tenth International Conference on Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing, which was held at the University of New South Wales, Sydney, Australia, from 13 to 17 February 2012. It contains a limited selection of articles based on presentations given at the conference. The conference program was arranged with the help of an international committee consisting of:

- William Chen, *Macquarie University, Australia*
- Ronald Cools, *KU Leuven, Belgium*
- Josef Dick, *University of New South Wales, Australia* (Conference organizer)
- Henri Faure, *CNRS Marseille, France*
- Alan Genz, *Washington State University, USA*
- Mike Giles, *University of Oxford, UK*
- Paul Glasserman, *Columbia University, USA*
- Michael Gnewuch, *University of Kaiserslautern, Germany*
- Stefan Heinrich, *University of Kaiserslautern, Germany*
- Fred J. Hickernell, *Illinois Institute of Technology, USA*
- Aicke Hinrichs, *University of Rostock, Germany*
- Stephen Joe, *University of Waikato, New Zealand*
- Aneta Karaivanova, *Bulgarian Academy of Science, Bulgaria*
- Alexander Keller, *NVIDIA, Germany*
- Dirk P. Kroese, *University of Queensland, Australia*
- Frances Y. Kuo, *University of New South Wales, Australia* (Conference organizer)
- Gerhard Larcher, *Johannes Kepler University Linz, Austria*
- Pierre L'Ecuyer, *Université de Montréal, Canada*
- Christiane Lemieux, *University of Waterloo, Canada*
- Peter Mathé, *Weierstrass Institute Berlin, Germany*
- Makoto Matsumoto, *Hiroshima University, Japan*
- Kerrie Mengersen, *Queensland University of Technology, Australia*
- Thomas Müller-Gronbach, *University of Passau, Germany*
- Harald Niederreiter, *RICAM Linz and University of Salzburg, Austria*
- Erich Novak, *University of Jena, Germany*
- Art B. Owen, *Stanford University, USA*

- Gareth W. Peters, *University of New South Wales, Australia, and University College London, UK* (Conference organizer)
- Friedrich Pillichshammer, *Johannes Kepler University Linz, Austria*
- Leszek Plaskota, *University of Warsaw, Poland*
- Eckhard Platen, *University of Technology Sydney, Australia*
- Klaus Ritter, *University of Kaiserslautern, Germany*
- Gareth Roberts, *University of Warwick, UK*
- Wolfgang Ch. Schmid, *University of Salzburg, Austria*
- Nikolai Simonov, *Russian Academy of Sciences, Russia*
- Ian H. Sloan, *University of New South Wales, Australia* (Conference organizer)
- Ilya M. Sobol', *Russian Academy of Sciences, Russia*
- Jerome Spanier, *Claremont, California, USA*
- Shu Tezuka, *Kyushu University, Japan*
- Xiaoqun Wang, *Tsinghua University, China*
- Grzegorz W. Wasilkowski, *University of Kentucky, USA*
- Henryk Woźniakowski, *Columbia University, USA, and University of Warsaw, Poland*

This conference continued the tradition of biennial MCQMC conferences initiated by Harald Niederreiter, held previously at:

- University of Nevada in Las Vegas, Nevada, USA, in June 1994
- University of Salzburg, Austria, in July 1996
- Claremont Colleges in Claremont, California, USA, in June 1998
- Hong Kong Baptist University in Hong Kong, China, in November 2000
- National University of Singapore, Republic of Singapore, in November 2002
- Palais des Congrès in Juan-les-Pins, France, in June 2004
- Ulm University, Germany, in July 2006
- Université de Montréal, Canada, in July 2008
- University of Warsaw, Poland, in August 2010

The next conference will be held at the KU Leuven, Belgium, in April 2014.

The proceedings of these previous conferences were all published by Springer-Verlag, under the following titles:

- *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing* (H. Niederreiter and P.J.-S. Shiue, eds.)
- *Monte Carlo and Quasi-Monte Carlo Methods 1996* (H. Niederreiter, P. Hellekalek, G. Larcher, and P. Zinterhof, eds.)
- *Monte Carlo and Quasi-Monte Carlo Methods 1998* (H. Niederreiter and J. Spanier, eds.)
- *Monte Carlo and Quasi-Monte Carlo Methods 2000* (K.-T. Fang, F.J. Hickernell, and H. Niederreiter, eds.)
- *Monte Carlo and Quasi-Monte Carlo Methods 2002* (H. Niederreiter, ed.)
- *Monte Carlo and Quasi-Monte Carlo Methods 2004* (H. Niederreiter and D. Talay, eds.)
- *Monte Carlo and Quasi-Monte Carlo Methods 2006* (A. Keller, S. Heinrich, and H. Niederreiter, eds.)

- *Monte Carlo and Quasi-Monte Carlo Methods 2008* (P. L'Ecuyer and A. Owen, eds.)
- *Monte Carlo and Quasi-Monte Carlo Methods 2010* (L. Plaskota and H. Woźniakowski, eds.)

The program of the conference was rich and varied with over 140 talks being presented. Highlights were the invited plenary talks given by Pierre Del Moral (INRIA and University of Bordeaux 1), Mike Giles (Oxford University), Fred J. Hickernell (Illinois Institute of Technology), Aicke Hinrichs (University of Jena), Michael Lacey (Georgia Institute of Technology), Kerrie Mengersen (Queensland University of Technology), Andreas Neuenkirch (University of Kaiserslautern), Art B. Owen (Stanford University), Leszek Plaskota (University of Warsaw), and Eckhard Platen (University of Technology Sydney), and the tutorials given by Art B. Owen (Stanford University), Pierre Del Moral (INRIA and University of Bordeaux 1), Josef Dick (University of New South Wales), and Alex Keller (NVIDIA).

The papers in this volume were carefully screened and cover both the theory and the applications of Monte Carlo and quasi-Monte Carlo methods. We thank the anonymous reviewers for their reports and many others who contributed enormously to the excellent quality of the conference presentations and to the high standards for publication in these proceedings by careful review of the abstracts and manuscripts that were submitted.

We gratefully acknowledge generous financial support of the conference by the School of Mathematics and Statistics of the University of New South Wales, the Australian Mathematical Society (AustMS), the Australian and New Zealand Industrial and Applied Mathematics (ANZIAM), the Australian Mathematical Sciences Institute (AMSI), the Commonwealth Scientific and Industrial Research Organisation (CSIRO), and the National Science Foundation (NSF).

Finally, we want to express our gratitude to Springer-Verlag for publishing this volume.

Sydney, Australia
September 2013

Josef Dick
Frances Y. Kuo
Gareth W. Peters
Ian H. Sloan

Contents

Part I Invited Articles

| | |
|---|------------|
| Computing Functionals of Square Root and Wishart Processes Under the Benchmark Approach via Exact Simulation | 3 |
| Jan Baldeaux and Eckhard Platen | |
| The Supremum Norm of the Discrepancy Function: Recent Results and Connections | 23 |
| Dmitriy Bilyk and Michael Lacey | |
| An Introduction to Stochastic Particle Integration Methods: With Applications to Risk and Insurance | 39 |
| Pierre Del Moral, Gareth W. Peters, and Christelle Vergé | |
| Multilevel Monte Carlo Methods | 83 |
| Michael B. Giles | |
| Guaranteed Conservative Fixed Width Confidence Intervals via Monte Carlo Sampling | 105 |
| Fred J. Hickernell, Lan Jiang, Yuewei Liu, and Art B. Owen | |
| Discrepancy, Integration and Tractability | 129 |
| Aicke Hinrichs | |
| Noisy Information: Optimality, Complexity, Tractability | 173 |
| Leszek Plaskota | |

Part II Tutorial

| | |
|--|------------|
| Quasi-Monte Carlo Image Synthesis in a Nutshell | 213 |
| Alexander Keller | |

Part III Contributed Articles

| | |
|---|-----|
| Conditional Sampling for Barrier Option Pricing Under the Heston Model | 253 |
| Nico Achtsis, Ronald Cools, and Dirk Nuyens | |
| Probabilistic Star Discrepancy Bounds for Double Infinite Random Matrices | 271 |
| Christoph Aistleitner and Markus Weimar | |
| The L^2 Discrepancy of Irrational Lattices | 289 |
| Dmitriy Bilyk | |
| Complexity of Banach Space Valued and Parametric Integration | 297 |
| Thomas Daun and Stefan Heinrich | |
| Extended Latin Hypercube Sampling for Integration and Simulation | 317 |
| Rami El Haddad, Rana Fakhereddine, Christian Lécot, and Gopalakrishnan Venkiteswaran | |
| A Kernel-Based Collocation Method for Elliptic Partial Differential Equations With Random Coefficients | 331 |
| Gregory E. Fasshauer and Qi Ye | |
| Polynomial Accelerated MCMC and Other Sampling Algorithms Inspired by Computational Optimization | 349 |
| Colin Fox | |
| Antithetic Multilevel Monte Carlo Estimation for Multidimensional SDEs | 367 |
| Michael B. Giles and Lukasz Szpruch | |
| On the Convergence of Quantum and Sequential Monte Carlo Methods | 385 |
| François Giraud and Pierre Del Moral | |
| Lower Error Bounds for Randomized Multilevel and Changing Dimension Algorithms | 399 |
| Michael Gnewuch | |
| A Non-empirical Test on the Second to the Sixth Least Significant Bits of Pseudorandom Number Generators | 417 |
| Hiroshi Haramoto, Makoto Matsumoto, Takuji Nishimura, and Yuki Otsuka | |
| A Finite-Row Scrambling of Niederreiter Sequences | 427 |
| Roswitha Hofer and Gottlieb Pirsic | |

| | |
|--|-----|
| Reconstructing Multivariate Trigonometric Polynomials by Sampling Along Generated Sets | 439 |
| Lutz Kämmerer | |
| Bayesian Approaches to the Design of Markov Chain Monte Carlo Samplers | 455 |
| Jonathan M. Keith and Christian M. Davey | |
| Deterministic Consistent Density Estimation for Light Transport Simulation | 467 |
| Alexander Keller and Nikolaus Binder | |
| On Wavelet-Galerkin Methods for Semilinear Parabolic Equations with Additive Noise | 481 |
| Mihály Kovács, Stig Larsson, and Karsten Urban | |
| Component-by-Component Construction of Hybrid Point Sets Based on Hammersley and Lattice Point Sets | 501 |
| Peter Kritzer, Gunther Leobacher, and Friedrich Pillichshammer | |
| A QMC-Spectral Method for Elliptic PDEs with Random Coefficients on the Unit Sphere | 517 |
| Quoc Thong Le Gia | |
| Sampling and Low-Rank Tensor Approximation of the Response Surface | 535 |
| Alexander Litvinenko, Hermann G. Matthies, and Tarek A. El-Moselhy | |
| The Stochastic EM Algorithm for Censored Mixed Models | 553 |
| Ian C. Marschner | |
| Existence of Higher Order Convergent Quasi-Monte Carlo Rules via Walsh Figure of Merit | 569 |
| Makoto Matsumoto and Takehito Yoshiki | |
| ANOVA Decomposition of Convex Piecewise Linear Functions | 581 |
| Werner Römisch | |
| Hit-and-Run for Numerical Integration | 597 |
| Daniel Rudolf | |
| QMC Galerkin Discretization of Parametric Operator Equations | 613 |
| Christoph Schwab | |
| On the Choice of Weights in a Function Space for Quasi-Monte Carlo Methods for a Class of Generalised Response Models in Statistics | 631 |
| Vasile Sinescu, Frances Y. Kuo, and Ian H. Sloan | |

**Multi-level Monte Carlo Finite Difference and Finite Volume
Methods for Stochastic Linear Hyperbolic Systems** 649
Jonas Šukys, Siddhartha Mishra, and Christoph Schwab

Conference Participants 667

Index 685

Part I

Invited Articles

Guaranteed Conservative Fixed Width Confidence Intervals via Monte Carlo Sampling

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

Abstract Monte Carlo methods are used to approximate the means, μ , of random variables Y , whose distributions are not known explicitly. The key idea is that the average of a random sample, Y_1, \dots, Y_n , tends to μ as n tends to infinity. This article explores how one can reliably construct a confidence interval for μ with a prescribed half-width (or error tolerance) ε . 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 μ . We discuss the important case where $Y = f(X)$ and X is a random d -vector with probability density function ρ . In this case μ can be interpreted as the integral $\int_{\mathbb{R}^d} f(\mathbf{x})\rho(\mathbf{x}) \, d\mathbf{x}$, 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

F.J. Hickernell (✉) · L. Jiang

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

Y. Liu

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

A.B. Owen

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

performance of assets that are described by a stochastic model. Then μ is the fair option price. The goal is to obtain a *confidence interval*

$$\Pr[|\mu - \hat{\mu}| \leq \varepsilon] \geq 1 - \alpha, \quad (1)$$

where

- μ is approximated by the sample average of n independent and identically distributed (IID) samples of Y ,

$$\hat{\mu} = \hat{\mu}_n = \frac{1}{n} \sum_{i=1}^n Y_i, \quad (2)$$

- ε is the half-width of the confidence interval, which also serves as an *error tolerance*, and
- α 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 ε in terms of n and

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

which itself may be approximated by the sample variance. The goal here is somewhat different. We want to fix ε 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 ε . 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 $\varepsilon \rightarrow 0$ but does not provide coverage guarantees for fixed $\varepsilon > 0$. We want such fixed ε 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, $\text{Var}(Y)$ 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, $\sigma^2 = \text{Var}(Y)$, 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 $\varepsilon \rightarrow 0$). 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 μ 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 :

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

(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

$$\sigma^2 \leq \sigma_{\max}^2 \quad (5)$$

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 σ 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 σ/ε . However, the cost of our Monte Carlo algorithm, and indeed any Monte Carlo algorithm based on IID sampling is proportional to σ^2 , so overestimating σ^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 $Y = f(X)$ for some function $f : \mathbb{R}^d \rightarrow \mathbb{R}$ and some random vector X with probability density function $\rho : \mathbb{R}^d \rightarrow [0, \infty)$. One may then interpret the mean of Y as the multidimensional integral

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

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

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

For a given g , the choice of ρ 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

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

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 ρ has independent marginals, one may apply a product form of a univariate quadrature rule to evaluate μ . 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 μ 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 Sect. 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 ε , one computes an approximation, $\hat{\mu}$, based on n evaluations of the integrand f , such that $|\mu - \hat{\mu}| \leq \varepsilon$. 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 Sect. 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 Sect. 3, where rigorous guarantees of its success and its cost are provided. Section 4 illustrates the challenges of computing μ 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 $Y = f(X)$ where $X \in \mathbb{R}^d$ is a random vector with probability density function ρ . In other settings the random quantity 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 μ 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 σ^2 , and its non-negative square root, σ , 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 κ 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 κ , which then implies that σ and γ are finite.

2.2 CLT Intervals

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

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

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

$$\Pr(\hat{\mu}_n - 2.58\sigma/\sqrt{n} \leq \mu \leq \hat{\mu}_n + 2.58\sigma/\sqrt{n}) \rightarrow 0.99 \quad (9)$$

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

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

may be substituted, yielding the interval $\hat{\mu}_n \pm 2.58s_n/\sqrt{n}$ 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 μ is in the interval then the absolute error $|\mu - \hat{\mu}_n| \leq \varepsilon := z_{\alpha/2}s_n/\sqrt{n}$.

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

$$\Pr(|\mu - \hat{\mu}_n| \leq 2.58s/\sqrt{n}) = 0.99 + \frac{1}{n}(A + B\gamma^2 + C\kappa) + O\left(\frac{1}{n^2}\right) \quad (11)$$

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 μ 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 μ and finite variance σ^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 μ , variance $\sigma^2 > 0$, and third centered moment $M_3 = E|Y_i - \mu|^3 / \sigma^3 < \infty$. Let $\hat{\mu}_n = (Y_1 + \dots + Y_n)/n$ denote the sample mean. Then*

$$\left| \Pr\left[\frac{\hat{\mu}_n - \mu}{\sigma/\sqrt{n}} < x\right] - \Phi(x) \right|$$

$$\leq \Delta_n(x, M_3) := \frac{1}{\sqrt{n}} \min \left(A_1(M_3 + A_2), \frac{A_3 M_3}{1 + |x|^3} \right) \quad \forall x \in \mathbb{R},$$

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 σ^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*

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

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_σ} from the distribution of Y . From this sample we compute the sample variance, $s_{n_\sigma}^2$, according to (10) and estimate the variance of Y_i by $\hat{\sigma}^2 = \mathfrak{C}^2 \hat{s}_{n_\sigma}^2$, where $\mathfrak{C}^2 > 1$ is a “variance inflation factor” that will reduce the probability that we have underestimated $\sigma^2 = \text{Var}(Y)$. 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_μ , 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 σ^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 κ . Let s_n^2 be the sample variance defined at (10), and let $\tilde{\kappa} = \kappa + 3$. Then*

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

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

Proof. Applying Theorem 4 and choosing

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

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

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

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

Using Lemma 1 we can bound the probability that $\hat{\sigma}^2 = \mathfrak{C}^2 s_{n_\sigma}^2$ overestimates σ^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_σ , α , 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,

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

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_μ to get the desired coverage level from an interval with half-length at most ε . We suppose here that σ is known. In practice we will use a conservative (biased high) estimate for σ .

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

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

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

Given knowledge of σ , 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

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

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

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

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

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

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

To compute $N_{\text{BE}}(\varepsilon, \sigma, \alpha, M_3)$, 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 Δ_n term in (16) were absent, N_{BE} would correspond to the CLT sample size N_{CLT} , and in general $N_{\text{BE}} > N_{\text{CLT}}$. It is possible that in some situations $N_{\text{BE}} > N_{\text{Cheb}}$ might hold, and in such cases we could use N_{Cheb} instead of N_{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, $n_\sigma \in \{2, 3, \dots\}$,
- 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, \dots, Y_{n_σ} are sampled independently from the same distribution as Y . Then the conservative variance estimate, $\hat{\sigma}^2 = \mathfrak{C}^2 s_{n_\sigma}^2$, is computed in terms of the sample variance, $s_{n_\sigma}^2$, 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 Eq. (13). The sample size for the second stage is

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

where

$$N_\mu(\varepsilon, \sigma, \alpha, M) := \max(1, \min(N_{\text{Cheb}}(\varepsilon, \sigma, \alpha), N_{\text{BE}}(\varepsilon, \sigma, \alpha, M))). \quad (18)$$

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

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

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

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

$$\Pr(|\hat{\mu} - \mu| \leq \varepsilon) \geq 1 - \alpha.$$

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 $\Pr(\hat{\sigma}^2 > \sigma^2) \geq 1 - \tilde{\alpha}$ by the argument preceding the kurtosis bound in (13) applied with uncertainty $\tilde{\alpha}$. The second stage yields $\Pr(|\hat{\mu} - \mu| \leq \varepsilon) \geq 1 - \tilde{\alpha}$ by the Berry-Esseen result (15), so long as $\hat{\sigma} \geq \sigma$ and $M_3 \leq \tilde{\kappa}_{\max}(\tilde{\alpha}, n_\sigma, \mathfrak{C})^{3/4}$.

The second condition holds because $M_3 \leq \tilde{\kappa}^{3/4}$ by Jensen's Inequality [10, 8.4.b]. Thus, in the two-stage algorithm we have

$$\begin{aligned} \Pr(|\hat{\mu} - \mu| \leq \varepsilon) &= \mathbb{E}[\Pr(|\hat{\mu} - \mu| \leq \varepsilon \mid \hat{\sigma})] \\ &\geq \mathbb{E}[(1 - \tilde{\alpha})1_{\sigma \leq \hat{\sigma}}] \\ &\geq (1 - \tilde{\alpha})(1 - \tilde{\alpha}) = 1 - \alpha. \quad \square \end{aligned}$$

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_σ , \mathfrak{C} , and α , 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_σ , \mathfrak{C} , and α . Alternatively, one may specify n_σ 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_σ . The issue of how one should choose n_σ , \mathfrak{C} , and $\tilde{\kappa}_{\max}$ in practice is discussed further in Sect. 5.

Remark 2. In this algorithm it is possible to choose n_μ much smaller than n_σ if the sample variance is small. As a practical matter we suggest that if one is willing to invest n_σ 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 Sect. 4 we use

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

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_σ should be chosen greater than n_μ . 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_μ in (20) means that the fixed width confidence interval constructed by Algorithm 1 also holds for any random variables, Y , with small variance, namely, $\sigma^2 \leq \varepsilon^2 \alpha n_\sigma$, 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 X . Then μ corresponds to the multivariate integral in (6) and Theorem 5 may be interpreted as below:

Corollary 1. *Suppose that $\rho : \mathbb{R}^d \rightarrow \mathbb{R}$ is a probability density function, the integrand $f : \mathbb{R}^d \rightarrow \mathbb{R}$ 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 μ defined in (6), which satisfies the fixed width confidence interval condition*

$$\Pr(|\hat{\mu} - \mu| \leq \varepsilon) \geq 1 - \alpha.$$

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_σ is deterministic, n_μ 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_μ in (17) is $\hat{\sigma}^2$, the upper bound on variance. Clearly this depends on the unknown population variance, σ^2 , and we expect $\hat{\sigma}^2$ not to overestimate σ^2 by much. Thus, the algorithm cost is defined below in terms of σ^2 and the error tolerance (interval half-width) ε . 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, \dots with common distribution function F having variance σ^2 and modified kurtosis $\tilde{\kappa}$. Additional algorithm inputs are an error tolerance, ε , an uncertainty, α , 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 β , for integrands of variance no greater than σ_{\max}^2 and modified kurtosis no greater than $\tilde{\kappa}_{\max}$ is defined as

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

Note that $\tilde{\kappa}_{\max}$ is an input to the algorithm, but σ_{\max} is not. The cost of an arbitrary algorithm, A may also depend on other parameters, such as n_σ 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

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

Since n_σ 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 $\tilde{\kappa} \leq \tilde{\kappa}_{\max}$,

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

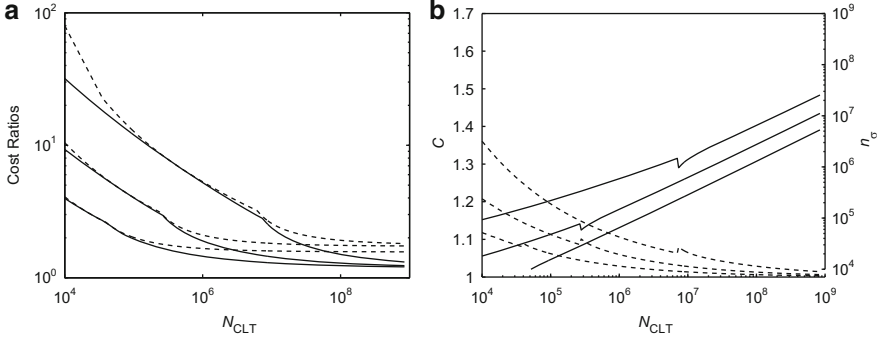


Fig. 1 (a) The cost ratios of $N_{up}(\varepsilon, 0.01, 0.01, \tilde{\kappa}_{\max}, \sigma)/N_{CLT}(\varepsilon, \sigma, 0.01)$ for $\tilde{\kappa}_{\max} = 2, 10$, and 100 , with $n_\sigma = 4,000 \tilde{\kappa}_{\max}$ (dashed) and n_σ optimized (solid); (b) the optimal values of n_σ (solid) and C (dashed).

$$= \Pr[\hat{\sigma}^2 < \sigma^2 v^2(\tilde{\alpha}, \beta, \mathfrak{C})],$$

where

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

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*

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

Note that the Chebychev sample size, N_{Cheb} , defined in (14), the Berry-Esseen sample size, N_{BE} , defined in (16), and thus N_μ all depend on σ and ε through their ratio, σ/ε . 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 σ_{\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 μ with a given error tolerance than random variables with large variances.

Figure 1a shows the ratio of the upper bound of the cost, $N_{up}(\varepsilon, 0.01, 0.01, \tilde{\kappa}_{\max}, \sigma)$, to the ideal CLT cost, $N_{CLT}(\varepsilon, \sigma, 0.01) = \lceil (2.58\sigma/\varepsilon)^2 \rceil$, for a range of σ/ε ratios and for $\tilde{\kappa}_{\max} = 2, 10$, and 100 . In these graphs the formula defining N_{up}

in Theorem 6 uses the alternative and somewhat costlier formula for N_μ in (20). The dashed curves in Fig. 1a show these cost ratios with $n_\sigma = 4,000 \tilde{\kappa}_{\max}$, which corresponds to $\mathfrak{C} \approx 1.1$. The solid curves denote the case where n_σ and \mathfrak{C} vary with σ/ε to minimize N_{up} . Figure 1b displays the optimal values of n_σ (solid) and \mathfrak{C} (dashed). In both figures, higher curves correspond to higher values of $\tilde{\kappa}_{\max}$.

Here, N_{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_{CLT} , equivalently smaller σ/ε , 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_{CLT} , equivalently larger σ/ε , the cost ratio approaches somewhat less than 1.4 in the case of optimal n_σ and \mathfrak{C} , and somewhat less than 2 for $n_\sigma = 1,000 \tilde{\kappa}_{\max}$.

The discontinuous derivatives in the curves in Fig. 1 arise from the minimum and maximum values arising in formulas (16) and (20) for N_{BE} and N_μ , respectively. Taking the upper dashed curve in Fig. 1a as an example, for N_{CLT} less than about 3.5×10^4 , $N_\mu = n_\sigma$. For N_{CLT} from about 3.5×10^4 to about 6×10^6 , N_μ corresponds to the second term in the minimum in the Berry-Esseen inequality, (16), i.e., the non-uniform term. For N_{CLT} greater than 6×10^6 , N_μ 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_σ and \mathfrak{C} with respect to σ/ε is impractical, since σ is not known in advance. Our suggestion is to choose \mathfrak{C} around 1.1, and then choose n_σ as large as needed to ensure that $\tilde{\kappa}_{\max}$ is as large as desired. For example with $\mathfrak{C} = 1.1$ and $\tilde{\kappa}_{\max} = 2, 10, \text{ and } 100$ we get $n_\sigma = 6,593, 59,311, \text{ and } 652,417$ 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.

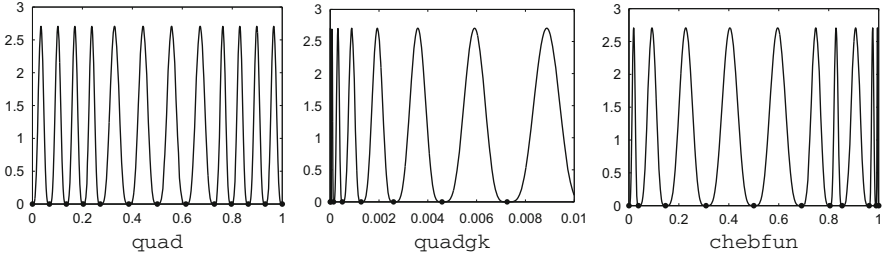


Fig. 2 Plots of fooling functions, f , with $\mu = \int_0^1 f(x) dx = 1$, but for which the corresponding algorithms return values of $\hat{\mu} = 0$.

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 $\mu = \int_0^1 f(x) dx$ 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).

4.2 Integrating a Single Hump

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

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

Here \mathbf{x} is a d dimensional vector, and $a_0, b_0, \dots, b_d, c_1, \dots, c_d, h_1, \dots, h_d$ 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:

- $b_1, \dots, b_d \in [0.1, 10]$ with $\log(b_j)$ being i.i.d. uniform,
- $c_1, \dots, c_d \in [10^{-6}, 1]$ with $\log(c_j)$ being i.i.d. uniform,
- $h_1, \dots, h_d \in [0, 1]$ with h_j being i.i.d. uniform,
- b_0 chosen in terms of the $b_1, \dots, b_d, c_1, \dots, c_d, h_1, \dots, h_d$ 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 $b_0, \dots, b_d, c_1, \dots, c_d, h_1, \dots, h_d$ to make $\mu = 1$.

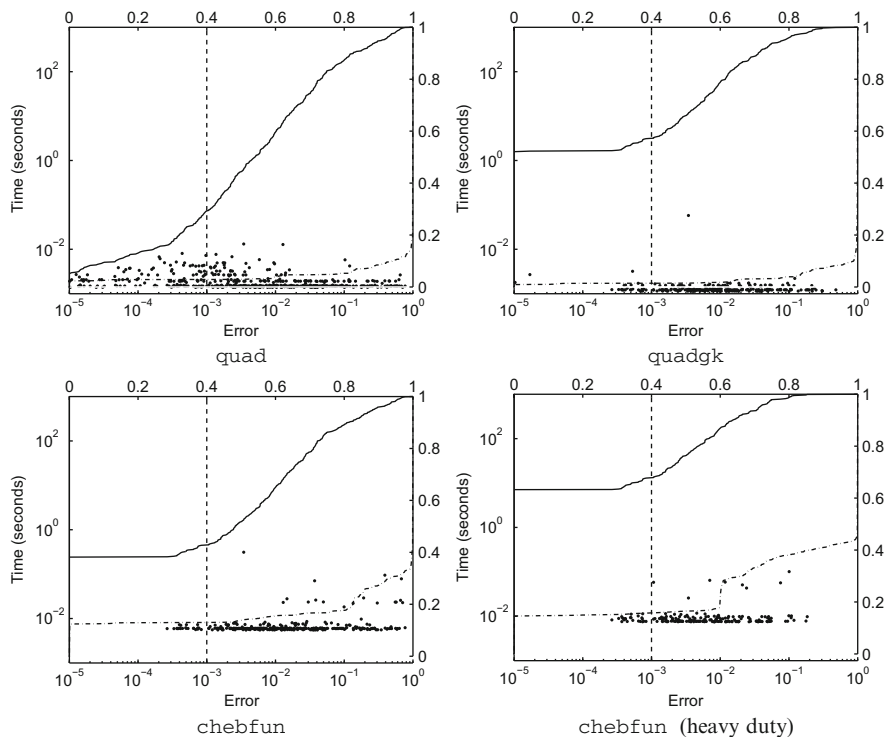


Fig. 3 Execution times and errors for test function (21) for $d = 1$ and error tolerance $\varepsilon = 10^{-3}$, and a variety of parameters giving a range of σ and $\tilde{\kappa}$. Those points to the left/right of the dashed vertical line represent successes/failures of the automatic algorithms.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

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, 17–19]. 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 $\epsilon = 0.001$. The algorithm `chebfun` attempts to do all calculations to near machine precision. The observed error and execution times are plotted in Figs. 3 and 4. Whereas

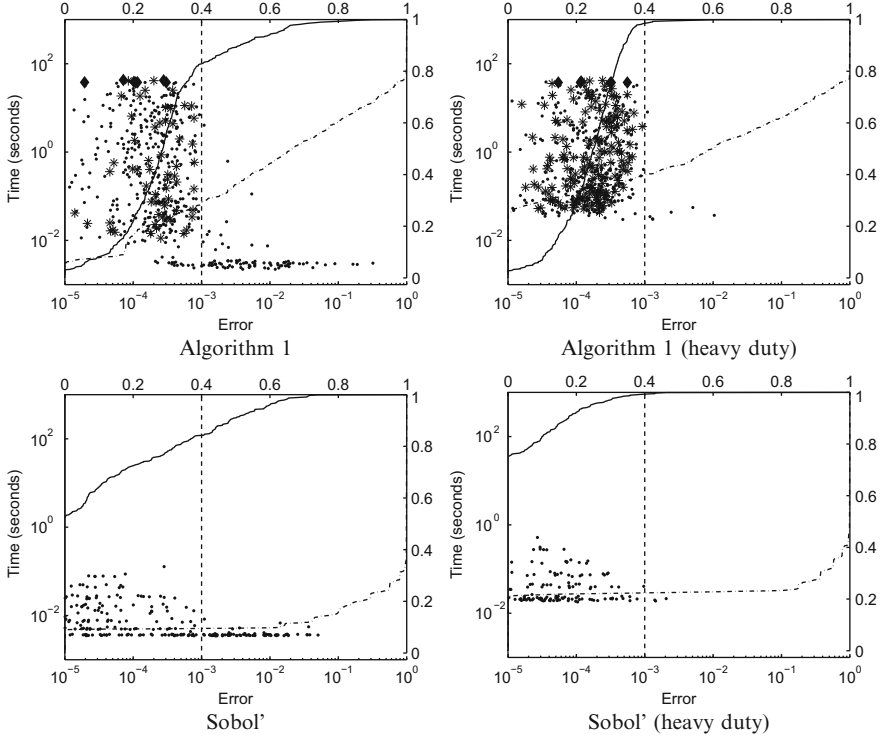


Fig. 4 Execution times and errors for test function (21) for $d = 1$ and error tolerance $\varepsilon = 10^{-3}$, and a variety of parameters giving a range of σ and $\tilde{\kappa}$. Those points to the left/right of the dashed vertical line represent successes/failures of the automatic algorithms. 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.

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, $n_\sigma = 2^{13} = 8,192$, which corresponds to $\tilde{\kappa}_{\max} = 2.24$. For the heavy duty plot on the right, $n_\sigma = 2^{18} = 262,144$, 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 s. 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 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 Fig. 4 the alternative and somewhat costlier formula for N_μ in (20) is employed. An asterisk is used to label those points satisfying $\tilde{\kappa} \leq \tilde{\kappa}_{\max}$, 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_μ such that $n_\sigma + n_\mu > N_{\max} := 10^9$. In these cases n_μ 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 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 Fig. 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 Fig. 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.

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

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

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

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

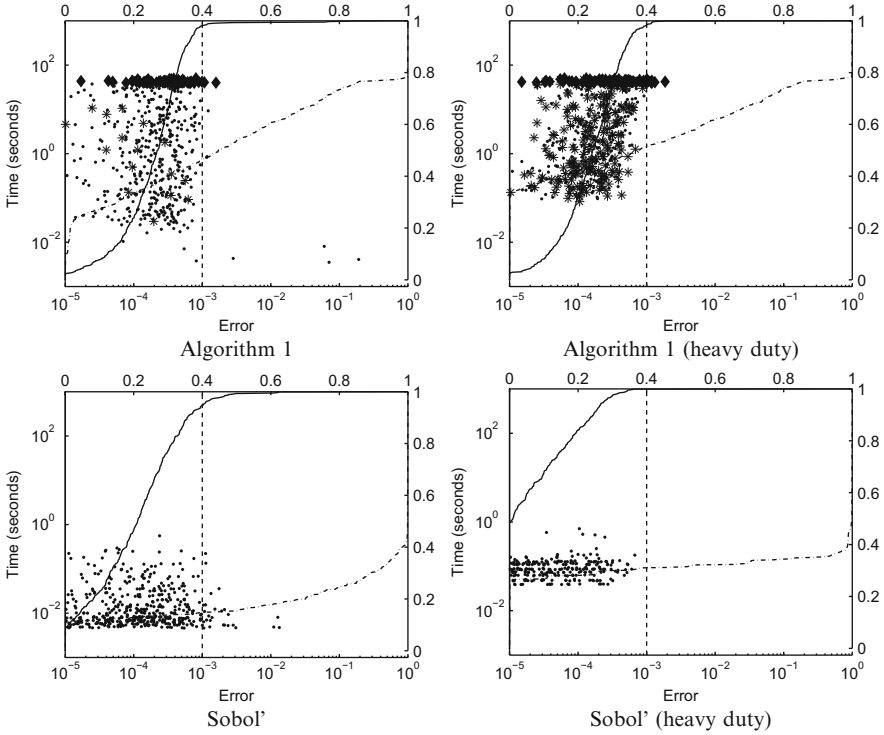


Fig. 5 Execution times and errors for test function (21) for $d = 2, \dots, 8$ and $\varepsilon = 10^{-3}$, with the rest of the parameters as in Fig. 4.

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, $B(T/d), B(2T/d), \dots, B(T)$, 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, \dots, 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 μ . The result of 500 replications is given in Fig. 6. Some of the parameters are set to be fixed values, namely,

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

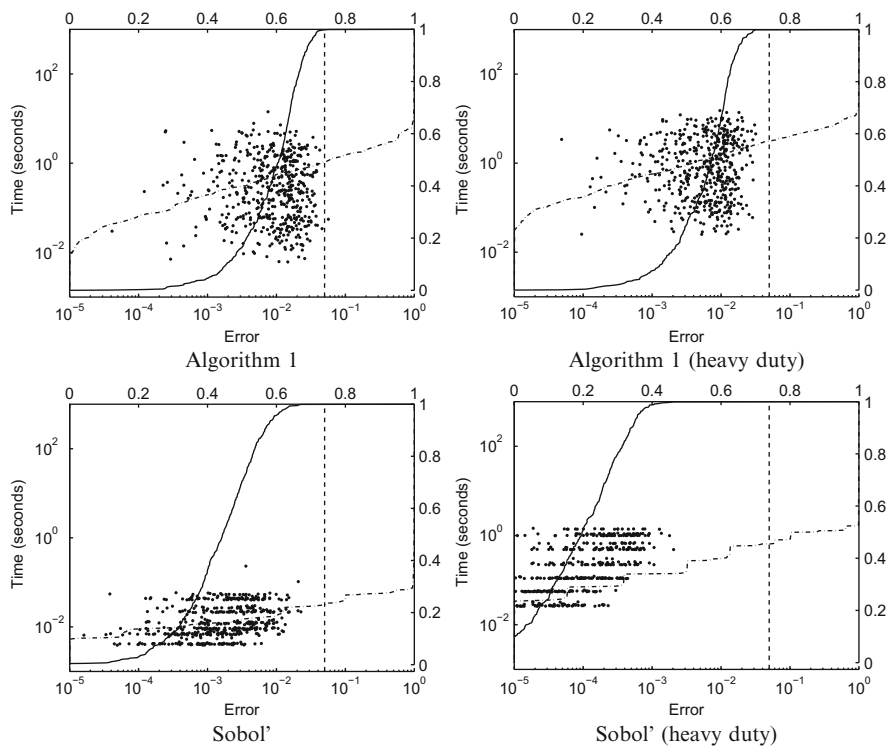


Fig. 6 Execution times and errors for the Asian geometric mean call option for $d = 1, 2, 4, 8, 16, 32$ and $\varepsilon = 0.05$.

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 μ for these parameters is between about 2.8 and 14.

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 σ (see Fig. 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 σ in advance, and our adaptive algorithm estimates σ and then an appropriate sample size n_μ from the data. Our algorithmic cost will be low when the unknown σ is small and large when σ 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_σ , 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 $\tilde{\kappa} \leq \tilde{\kappa}_{\max}$ since it is even more difficult to estimate $\tilde{\kappa}$ from a sample than it is to estimate σ^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 α , n_σ and \mathfrak{C} . The discussion at the end of Sect. 3.4 provides guidance on how to choose n_σ 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_σ , but choose n_σ 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 Fig. 1, in the limit of vanishing ε/σ , i.e., $N_{\text{CLT}} \rightarrow \infty$, the choice of $\tilde{\kappa}_{\max}$ makes a negligible contribution to the total cost of the algorithm. The main determinant of computational cost is ε/σ .

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_μ 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, Chap. 4, Theorem 5.2.1; Chap. 8, Corollary

5.3.1]. Here, in Corollary 1 the cone, $\mathcal{C}_{\kappa_{\max}}$, 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

$$\Pr[|\mu - \hat{\mu}| \leq \varepsilon |\mu|] \geq 1 - \alpha.$$

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. Bahadur, R.R., Savage, L.J.: The nonexistence of certain statistical procedures in nonparametric problems. *Ann. Math. Stat.* **27**, 1115–1122 (1956)
2. Chow, Y.S., Robbins, H.: On the asymptotic theory of fixed-width sequential confidence intervals for the mean. *Ann. Math. Stat.* **36**, 457–462 (1965)
3. Dick, J., Pillichshammer, F.: *Digital nets and sequences: discrepancy theory and quasi-Monte Carlo integration*. Cambridge University Press, Cambridge (2010)
4. Gander, W., Gautschi, W.: Adaptive quadrature—revisited. *BIT* **40**, 84–101 (2000)
5. Hale, N., Trefethen, L.N., Driscoll, T.A.: *Chebfun version 4* (2012)
6. Hall, P.: Theoretical comparisons of bootstrap confidence intervals. *Ann. Statist.* **16**, 927–953 (1988)
7. Hong, H.S., Hickernell, F.J.: Algorithm 823: implementing scrambled digital nets. *ACM Trans. Math. Software* **29**, 95–109 (2003)
8. Lehmann, E.L., Romano, J.P.: *Testing Statistical Hypotheses*, 3rd edn. Springer, New York (2005)
9. Lemieux, C.: *Monte Carlo and Quasi-Monte Carlo Sampling*. Springer Science+Business Media, Inc., New York (2009)
10. Lin, Z., Bai, Z.: *Probability Inequalities*. Science Press/Springer, Beijing/Berlin (2010)
11. Matoušek, J.: On the L_2 -discrepancy for anchored boxes. *J. Complexity* **14**, 527–556 (1998)
12. McCullagh, P.: *Tensor Methods in Statistics*. Chapman and Hall, London (1987)
13. Miller, R.: *Beyond ANOVA, Basics of Applied Statistics*. Wiley, New York (1986)

14. Mukhopadhyay, N., Datta, S.: On sequential fixed-width confidence intervals for the mean and second-order expansions of the associated coverage probabilities. *Ann. Inst. Stat. Math.* **48**, 497–507 (1996)
15. Nefedova, Yu.S., Shevtsova, I.G.: On non-uniform convergence rate estimates in the central limit theorem. *Theory Probab. Appl.* **57**, 62–97 (2012)
16. Niederreiter, H.: *Random Number Generation and Quasi-Monte Carlo Methods*. CBMS-NSF Regional Conference Series in Applied Mathematics. SIAM, Philadelphia (1992)
17. Owen, A.B.: Randomly permuted (t, m, s) -nets and (t, s) -sequences. In: Niederreiter, H., Shiue, P.J.-S. (eds.) *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*. Lecture Notes in Statistics, vol. 106, pp. 299–317. Springer, New York (1995)
18. Owen, A.B.: Monte Carlo variance of scrambled net quadrature. *SIAM J. Numer. Anal.* **34**, 1884–1910 (1997)
19. Owen, A.B.: Scrambled net variance for integrals of smooth functions. *Ann. Statist.* **25**, 1541–1562 (1997)
20. Owen, A.B.: On the Warnock-Halton quasi-standard error. *Monte Carlo Methods Appl.* **12**, 47–54 (2006)
21. Serfling, R.J., Wackerly, D.D.: Asymptotic theory of sequential fixed-width confidence procedures. *J. Amer. Statist. Assoc.* **71**, 949–955 (1976)
22. Shampine, L.F.: Vectorized adaptive quadrature in Matlab. *J. Comput. Appl. Math.* **211**, 131–140 (2008)
23. Shevtsova, I.: On the absolute constants in the Berry–Esseen type inequalities for identically distributed summands. arXiv:1111.6554v1 [math.PR] (2011)
24. Siegmund, D.: *Sequential analysis: tests and confidence intervals*. Springer, New York (1985)
25. Sloan, I.H., Joe, S.: *Lattice methods for multiple integration*. Oxford University Press, Oxford (1994)
26. Stein, C.: A two sample test for a linear hypothesis whose power is independent of the variance. *Ann. Math. Stat.* **16**, 243–258 (1945)
27. Stein, C.: Some problems in sequential estimation. *Econometrica* **17**, 77–78 (1949)
28. The MathWorks, Inc.: *MATLAB 7.12*, Natick (2012)
29. Traub, J.F., Wasilkowski, G.W., Woźniakowski, H.: *Information-based complexity*. Academic, Boston (1988)