General multilevel Monte Carlo methods for pricing discretely monitored Asian options

Nabil Kahalé *

April 22, 2020

Abstract

We describe general multilevel Monte Carlo methods that estimate the price of an Asian option monitored at m fixed dates. For a variety of processes that can be simulated exactly, we prove that, for the same computational cost, our method yields an unbiased estimator with variance lower than the variance of the standard Monte Carlo estimator by a factor of order m. We show how to combine our approach with the Milstein scheme for processes driven by scalar stochastic differential equations, and with the Euler scheme for processes driven by multidimensional stochastic differential equations. Numerical experiments confirm that our method outperforms the conventional Monte Carlo algorithm by a factor proportional to m.

Keywords: finance, simulation, Asian option, multilevel Monte Carlo method, variance reduction

1 Introduction

Asian options are financial derivatives whose payoff depends on the arithmetic average of an underlying during a specific time-period. Asian options are useful to corporations which are exposed to average exchange rates or commodity prices over a certain period of time. Pricing Asian options has been the subject of many studies. Under the Black-Scholes model, the price of a continuously sampled Asian option can be expressed as an infinite series (Linetsky 2004). Discretely monitored Asian options under Lévy processes are valued by recursive integration in (Fusai and Meucci 2008) and via a fast Fourier transform algorithm in (Černỳ and Kyriakou 2011). Expansions for discretely sampled Asian options are derived in (Cai, Li and Shi 2013) for diffusion models and in (Shiraya and Takahashi 2018) for diffusion processes with jumps. Bandi and Bertsimas (2014) price Asian options via robust linear optimization. Asian options under Markov processes can be priced via transform based methods (Cai, Song and Kou 2015, Cui, Lee and Liu 2018). Fusai and Kyriakou (2016) give optimized bounds on Asian option prices for a wide range of processes. Kahalé (2017) describes a convex programming method that computes optimal model-independent bounds on Asian option prices. Corsaro, Kyriakou, Marazzina and Marino (2019) present a transform-based algorithm that prices discretely monitored Asian options in a general stochastic volatility framework that includes Lévy processes. (Gambaro, Kyriakou and Fusai 2020) propose a discrete-time approach to pricing Asian options. Monte Carlo methods can price Asian options under various models, but conventional Monte Carlo algorithms have a high computational cost, which motivates the need to improve the efficiency of such methods. Control variate techniques (Kemna and Vorst 1990, Dingeç and Hörmann 2012, Dingec, Sak and Hörmann 2014, Shiraya and Takahashi 2017), path adjustments methods (see (Duan and Simonato 1998) and (Glasserman 2004, Section 4.5.1)), importance sampling

*ESCP Business School, 75011 Paris, France; e-mail: nkahale@escp.eu.

algorithms (Glasserman, Heidelberger and Shahabuddin 1999, Genin and Tankov 2018), and Quasi-Monte Carlo methods (e.g. (Wang and Sloan 2011)) are common variance reduction techniques used to price Asian options. When the underlying follows a stochastic differential equation (SDE) satisfying certain conditions, the multilevel Monte Carlo method (MLMC) described in (Giles 2008) estimates the price of a continuously monitored Asian option with mean square error ϵ^2 in $O((\ln(\epsilon))^2\epsilon^{-2})$ time using the Euler discretization scheme, for $\epsilon > 0$. This computational cost is reduced to $O(\epsilon^{-2})$ time using the Milstein scheme for scalar SDEs (Giles, Debrabant and Rössler 2019) and multi-dimensional SDEs (Giles, Szpruch et al. 2014) satisfying certain conditions. For a broad class of pure-jump exponential Lévy processes, Giles and Xia (2017) estimate the price of a continuously monitored Asian option with mean square error ϵ^2 in $O(\epsilon^{-2})$ time via the MLMC method. Kebaier and Lelong (2018) show how to combine the MLMC method with importance sampling. Randomized multilevel Monte Carlo methods (RMLMC) that produce efficient and unbiased estimators of expectations of functionals arising in SDEs are given in (Rhee and Glynn 2015, Vihola 2018). Exact simulation algorithms, which exist for several financial models (see (Glasserman 2004, Section 3)), also yield unbiased estimators for prices of derivatives. Recent exact simulation methods have been developed for Heston's stochastic volatility model (Broadie and Kaya 2006, Glasserman and Kim 2011), jump-diffusion processes (Giesecke and Smelov 2013), the SABR model (Cai, Song and Chen 2017), and the Ornstein–Uhlenbeck driven stochastic volatility model (Li and Wu 2019).

Consider now an Asian option with a given maturity monitored at m fixed dates. When the underlying price process at the m dates can be simulated exactly in $\Theta(m)$ time, as in the Black-Scholes model for instance, the time required to estimate the option price with variance $O(\epsilon^2)$ is $\Theta(m\epsilon^{-2})$ under the conventional Monte Carlo method, assuming the payoff variance is upper and lower bounded by constants independent of m. This is because the price process needs to be simulated $\Theta(\epsilon^{-2})$ times to achieve such accuracy.

This paper describes a general multilevel framework to price an Asian option monitored at m dates. Our approach does not make any assumptions on the nature of the stochastic process driving the underlying. It however assumes the existence of a linear relationship between the underlying and forward prices, that the underlying price is square-integrable, and makes certain assumptions on the running time required to simulate the underlying on a discrete time grid with a given precision. The latter condition is satisfied in any model where the underlying price process can be simulated exactly at m' fixed dates in $O(m')$ expected time. Using the Milstein scheme, it is also satisfied, under certain conditions and for suitable discretization grids, by processes driven by scalar SDEs. Our approach yields unbiased estimators with variance $O(\epsilon^2)$ for the Asian option price in $O(m + \epsilon^{-2})$ expected time for a variety of processes including the Black-Scholes model, Merton's jump-diffusion model, the Square-Root model, Kou's double exponential jump-diffusion model, the variance gamma and the normal inverse Gaussian model (NIG) exponential Lévy processes and, using the Milstein scheme, processes driven by scalar SDEs satisfying certain conditions. Our method is also applicable with the same performance guarantees if the underlying is the average of assets that follow a multi-dimensional geometric Brownian motion. Using the Euler scheme, our approach produces a (usually biased) estimator with mean square error $O(\epsilon^2)$ in $O(m + (\ln(\epsilon))^2\epsilon^{-2})$ expected time for processes driven by one-dimensional or multidimensional SDEs satisfying certain conditions. We are not aware of any previous Monte Carlo, MLMC or RMLMC method that provably achieves such tradeoffs between the running time and target accuracy, even under the Black-Scholes model. Our method has the following features:

1. It is simple to implement and is based on the martingale property of forward prices. It is provably efficient for a wide range of processes, including a class of pure-diffusion, jump-diffusion and pure-jump exponential Lévy models. It does not make any assumptions on the dates at which the option is monitored. It assumes that the sum of the absolute values of the weights associated with the monitoring dates is upper-bounded by a constant

independent of m, but makes no assumptions on the sign or order of magnitude of these weights. Our approach thus applies to average price and average strike options.

2. It prices Asian options monitored at m dates with target accuracy $O(\epsilon)$ in $O(m+\epsilon^{-2})$ or $O(m+(\ln(\epsilon))^2\epsilon^{-2})$ expected time, depending on the assumptions satisfied by the diffusion process. Note that, for general weights and dates, the time to read the input is of order m. Previous multilevel methods have focused on continuously monitored Asian options. Giles, Szpruch et al. (2014) and Giles, Debrabant and Rössler (2019) mention that their methods can price Asian options monitored at m dates, but do not analyse the performance of their algorithms in terms of m. The tradeoff between the running time and target accuracy of our unbiased price estimator is similar to that of the randomized dimension reduction technique for Monte Carlo simulations, that is used in (**?**) to reduce the variance in Markov chains simulations.

The rest of the paper is organized as follows. Section 2 describes the modelling framework and recalls the MLMC and RMLMC methods. Section 3 presents our algorithms for Asian options pricing, provides examples, and shows how to combine our approach with the Euler and Milstein schemes. Section 4 gives numerical simulations. Section 5 contains concluding remarks. Omitted proofs are contained in the appendix. Further numerical experiments, a detailed description of our algorithms, and a combination of our approach with a martingale-based control variate technique are presented in the appendix. Our approach can in principle be combined with alternative control variate, path adjustments and Quasi-Monte Carlo methods. The appendix also shows how to adapt our method to continuously monitored Asian options.

2 Preliminaries

2.1 The modelling framework

Assume that interest rates are deterministic. Let T be a fixed maturity and m a positive integer. Denote by $F(t)$ the forward price of an underlying calculated at time t for maturity T. For $0 \leq j \leq m$, let $F_j := F(t_j)$, where $t_0 < \cdots < t_m$, with $t_0 = 0$ and $t_m = T$. Note that F_m is the underlying price at T. Let $A := \sum_{j=1}^{m} w_j F_j$ be a linear combination of the forward prices, where the w_j's are non-zero signed weights whose absolute values sum up to 1. Consider an Asian option with payoff $f(A)$ at maturity T, where f is a κ-Lipschitz real-valued function of one variable. Such a payoff can model Asian options that arise in a broad range of situations. For instance, the payoff of an average price call with strike K and maturity T on futures prices maturing at T is equal to $f(A)$, with $f(x) = \max(x-K, 0)$ and $w_1 = \cdots = w_m = 1/m$. This is because forward prices are equal to futures prices when interest rates are deterministic. Similarly, the payoff of an average strike call with maturity T on futures prices maturing at T is equal to $f(A)$, where $f(x) = 2\max(x, 0)$ and $w_1 = \cdots = w_{m-1} = -(m-1)^{-1}/2$, with $w_m = 1/2$. In the same vein, average price and average strike options have a payoff equal to $f(A)$ for a suitable choice of f and of the weights w_j's if the underlying is a stock that pays deterministic dividends, or an index with a deterministic and continuous dividend rate, or an exchange rate. This is due to the existence of a deterministic linear relationship between the forward price and the underlying price (see (Hull 2014, Chap. 5)).

We assume the existence of a risk-neutral probability Q such that the sequence (F_j), $0 \leq j \leq m$, is a martingale under Q, and the price of the option at time 0 is $e^{-rT}\mathbb{E}(f(A))$, where r is the risk-free rate at time 0 for maturity T. The existence of Q can be shown under no-arbitrage conditions (see (Glasserman 2004, Section 1.2.2)). All expectations in this paper are taken with respect to Q. We assume that F_m is square-integrable. By (Revuz and Yor 1999, Corollary 1.6, p. 53), this implies that F_j is square-integrable for $1 \leq j \leq m$. We also assume that κ is upper-bounded by a constant independent of m. Throughout the rest of the paper, the running time refers to the number of arithmetic operations. Denote by $\mathbb{N}$ the set of non-negative integers.

2.2 The MLMC method

The MLMC method described in (Giles 2008) efficiently estimates the expectation of a random variable Y_L that is approximated with increasing accuracy by random variables Y_l, $0 \le l \le L-1$, for some integer L. For $0 \le l \le L$, denote by C_l the expected cost of computing $Y_l - Y_{l-1}$, with $Y_{-1} := 0$. Assume that Y_l, $0 \le l \le L$, are square-integrable. For $0 \le l \le L$, let $\bar{Y}_l$ be the average of n_l independent copies of $Y_l - Y_{l-1}$, where n_l is a positive integer. Assume that the estimators $\bar{Y}_0, \dots, \bar{Y}_L$ are independent. Following the analysis in (Giles 2008), $\bar{Y} := \sum_{l=0}^{L} \bar{Y}_l$ is an unbiased estimator of $\mathbb{E}(Y_L)$, and

$$\mathrm{Var}(\bar{Y}) = \sum_{l=0}^{L} \frac{\mu_l}{n_l}, \tag{1}$$

where $\mu_l := \mathrm{Var}(Y_l - Y_{l-1})$ for $0 \le l \le L$. Let $\bar{C} := \sum_{l=0}^{L} n_l C_l$ be the expected cost of computing $\bar{Y}$. It is observed in (Giles 2008) that the work-normalized variance $\bar{C}\mathrm{Var}(\bar{Y})$ is minimized when n_l is proportional to $\sqrt{\mu_l / C_l}$, ignoring integrality constraints. The work-normalized variance of an unbiased estimator is defined as the product of the variance and expected running time. Glynn and Whitt (1992) show that the efficiency of an unbiased estimator is inversely proportional to the work-normalized variance.

2.3 The RMLMC method

We now recall a RMLMC method of Rhee and Glynn (2015) that efficiently estimates the expectation of a random variable Y that is approximated by random variables Y_l, $l \ge 0$. As in Section 2.2, denote by C_l the expected cost of computing $Y_l - Y_{l-1}$, for $l \ge 0$, with $Y_{-1} := 0$. Assume that Y and Y_l, $l \ge 0$, are square-integrable. Let (p_l), $l \ge 0$, be a probability distribution such that $p_l > 0$ for $l \ge 0$. Let $N \in \mathbb{N}$ be an integral random variable independent of $(Y_l : l \ge 0)$ such that $\Pr(N = l) = p_l$ for $l \ge 0$. Set $Z := (Y_N - Y_{N-1})/p_N$, with $Y_{-1} := 0$. For a square-integrable random variable X, let $||X|| := \sqrt{\mathbb{E}(X^2)}$. The following result is due to Rhee and Glynn (2015) (see also (Vihola 2018, Theorem 2)).

Theorem 2.1 (Rhee and Glynn (2015))**.** *Assume that $||Y_l - Y||$ converges to 0 as l goes to infinity. If $\sum_{l=0}^{\infty} ||Y_l - Y_{l-1}||^2 / p_l$ is finite then Z is square-integrable, $\mathbb{E}(Z) = \mathbb{E}(Y)$, and*

$$||Z||^2 = \sum_{l=0}^{\infty} \frac{||Y_l - Y_{l-1}||^2}{p_l}.$$

Denote by C be the expected cost of computing Z. Propositions 2.1 and 2.2 are in the same spirit as results previously obtained in (Giles 2008, Theorem 3.1) and (Rhee and Glynn 2015). For completeness, we give their proof in the appendix. Proposition 2.1 shows that, under certain conditions on Y_l and C_l, the sequence (p_l), $l \ge 0$, can be chosen so that both $||Z||$ and C are finite.

Proposition 2.1. *Assume that $||Y_0||^2 \le \nu$ and that, for $l \ge 0$,*

$$||Y_l - Y||^2 \le \nu 2^{-\beta l} \tag{2}$$

and $C_l \le c2^l$, where c, ν and β are positive constants, with $\beta \in (1, 2]$. If, for $l \ge 0$,

$$p_l = (1 - 2^{-(\beta+1)/2}) 2^{-(\beta+1)l/2}, \tag{3}$$

then Z is square-integrable, $\mathbb{E}(Z) = \mathbb{E}(Y)$, and

$$||Z||^2 \le \frac{20\nu}{1 - 2^{-(\beta-1)/2}}. \tag{4}$$

Furthermore,

$$C \leq \frac{c}{1 - 2^{-(\beta-1)/2}}. \tag{5}$$

If we relax (2), Proposition 2.2 shows how to construct a biased estimator Z_I of Y, for any positive integer I, with expected cost and variance bounded by a linear function of I, and a bias that decreases geometrically with I.

Proposition 2.2. *Assume that* $||Y_0||^2 \leq \nu$ *and that, for* $l \geq 0$*,*

$$||Y_l - Y||^2 \leq \nu 2^{-l} \tag{6}$$

and $C_l \leq c2^l$*, where* ν *and* c *are positive constants. Let* $p_l = 2^{-(l+1)}$ *for* $l \geq 0$*. Fix a positive integer* I *and set* $Z_I := (Y_N - Y_{N-1})\mathbf{1}_{\{N \leq I\}}/p_N$*. Then* Z_I *is square-integrable,*

$$(\mathbb{E}(Z_I - Y))^2 \leq \nu 2^{-I}, \tag{7}$$

and

$$||Z_I||^2 \leq 12\nu(I+1). \tag{8}$$

Furthermore, the expected cost of computing Z_I *is at most* cI*.*

More sophisticated versions of the RMLMC method can be found in (Rhee and Glynn 2015, Vihola 2018).

3 Multilevel algorithms for Asian options

We construct multilevel approximations of A in Section 3.1 and use them in Sections 3.2 and 3.3 to build estimators of the Asian option price. Section 3.2 considers the case where forward prices can be simulated exactly, while Section 3.3 treats the case where forward prices can be simulated approximately. Set $\alpha := f((\sum_{j=1}^m w_j)F_0)$ and $U := f(A) - \alpha$.

3.1 Multilevel approximations of A

Here we construct an increasing sequence of subsets of $\{1, \ldots, m\}$ and show that A is approximated, with increasing accuracy, by linear combinations of forward prices corresponding to these subsets. For integers i and j with $1 \leq i \leq m$ and $0 \leq j \leq m$, let

$$W(i,j) := \sum_{k=i}^{j} w_k \text{ and } W'(i,j) := \sum_{k=i}^{j} |w_k|.$$

By convention, $W(i,j) = W'(i,j) := 0$ if $j < i$. Define the subsets J_l of $\{1, \ldots, m\}$, for $l \geq 0$, as follows. Set $L := \lceil \log_2 m \rceil$ and $J_l := \{1, \ldots, m\}$ for $l \geq L$. For $0 \leq l \leq L-1$, let

$$J_l := \{j \in \{1, \ldots, m\} : 2^l W'(1, j-1) < \lfloor 2^l W'(1,j) \rfloor\}, \tag{9}$$

where $\lfloor x \rfloor$ denotes the largest integer upper-bound by x. Note that $J_0 = \{m\}$. Roughly speaking, J_l consists of the indices j where the sequence $W'(1,j)$ "jumps" over a multiple of 2^{-l}. Proposition 3.1 shows that the sequence (J_l), $l \geq 0$, is increasing and that the size of J_l is at most $2^l + 1$.

Proposition 3.1. *For* $l \geq 0$*,*

$$|J_l| \leq 2^l + 1 \tag{10}$$

and

$$J_l \subseteq J_{l+1}. \tag{11}$$

Proposition 3.1 implies that, for $0 \le l \le L-1$,

$$J_l = \{j \in J_{l+1} : 2^l W'(1, j-1) < \lfloor 2^l W'(1,j) \rfloor\}. \tag{12}$$

For $l \ge 0$, define the following trapezoidal approximation of A:

$$A_l := \sum_{j \in J_l} w_j F_j + \frac{1}{2} \sum_{(i,k) \in \mathcal{P}_l} W(i+1, k-1)(F_i + F_k), \tag{13}$$

where $\mathcal{P}_l$ is the set of pairs of consecutive of elements of the set $\{0\} \cup J_l$. Thus A_l is obtained from A by replacing each F_j with $(F_i + F_k)/2$ for each pair $(i,k) \in \mathcal{P}_l$ and each integer j with $i < j < k$. By construction, A_l is a deterministic linear function of (F_j), $j \in J_l$. Note that $A_l = A$ for $l \ge L$. Theorem 3.1 gives a bound on the L^2-distance between A_0 and $W(1,m)F_0$ on one hand, and between A_l and A on the other hand.

Theorem 3.1. $||A_0 - W(1,m)F_0||^2 \le \mathrm{Var}(F_m)$ *and, for* $l \ge 0$,

$$||A_l - A||^2 \le 2^{-2l}\mathrm{Var}(F_m). \tag{14}$$

Algorithm M below calculates the coefficients $W(i+1, k-1)$ in (13), for $0 \le l \le L-1$ and $(i,k) \in \mathcal{P}_l$, in $O(m)$ total time, using the following steps.

1. Calculate recursively $W(1,j)$ and $W'(1,j)$ for $1 \le j \le m$.

2. Construct by backward induction the subsets J_l, for $0 \le l \le L$, using (12). This takes $O(m)$ total time because $|J_{l+1}| \le 1 + 2^{l+1}$ for $l \in \{0, \ldots, L-1\}$, and so J_l can be constructed in $O(2^l)$ time.

3. For $l \in \{0, \ldots, L-1\}$ and each pair $(i,k) \in \mathcal{P}_l$, calculate $W(i+1, k-1)$ via the relation $W(i+1, k-1) = W(1, k-1) - W(1, i)$. For each level l, this takes $O(2^l)$ time, and so this step takes $O(m)$ total time.

3.2 The exact simulation case

Assumption 1 (A1). There is a constant c independent of m such that, for any subset J of $\{1, \ldots, m\}$, the expectation of the time required to simulate the vector (F_j), $j \in J$, is at most $c|J|$.

A1 holds if the expectation of the time to simulate the forward price process on a discrete time grid of size n is $O(n)$. Theorem 3.2 shows how to construct an unbiased estimator of the Asian option price under A1 using the RMLMC method. The p_l's are chosen by setting $\beta = 2$ in (3), as suggested by the proof of the theorem.

Theorem 3.2. *Suppose A1 holds. Let* $N \in \mathbb{N}$ *be an integral random variable independent of* $(F_j : 1 \le j \le m)$ *such that* $\Pr(N = l) = p_l$ *for non-negative integer* l*, where* $p_l := (1 - 2^{-3/2})2^{-3l/2}$*. Set* $V := (U_N - U_{N-1})/p_N$*, where* $U_l := f(A_l) - \alpha$ *for* $l \ge 0$ *and* $U_{-1} := 0$*. Then* V *is square-integrable,*

$$\mathbb{E}(f(A)) = \mathbb{E}(V) + \alpha, \tag{15}$$

and

$$\mathrm{Var}(V) \le 70\kappa^2 \mathrm{Var}(F_m). \tag{16}$$

Furthermore, the expectation of the time required to simulate V *is upper-bounded by a constant independent of* m.

Proof. As $|J_l| \leq 2^l + 1$, the expectation of the time to simulate the vector (F_j), $j \in J_l$, is at most $c2^{l+1}$. Together with (13), this implies the existence of a constant c' independent of m such that, for $l \geq 0$, the expectation of the time to simulate $U_l - U_{l-1}$ is at most $c'2^l$. Since $A_L = A$, we have $U_L = U$. By (13), A_l is square-integrable for $l \geq 0$ and, because f is κ-Lipschitz, so are U_l and U. As $|U_0| \leq \kappa |A_0 - W(1,m)F_0|$, Theorem 3.1 implies that

$$||U_0||^2 \leq \kappa^2 \mathrm{Var}(F_m). \tag{17}$$

Similarly, as $|U_l - U| \leq \kappa |A_l - A|$ for $l \geq 0$, by Theorem 3.1,

$$||U_l - U||^2 \leq \kappa^2 2^{-2l} \mathrm{Var}(F_m). \tag{18}$$

The conditions of Proposition 2.1 are thus met for $Y = U$ and $Y_l = U_l$ for $l \geq 0$, with $\nu = \kappa^2 \mathrm{Var}(F_m)$, $\beta = 2$ and $c = c'$. By (5), the expectation of the time required to simulate $V = Z$ is at most $4c'$. Furthermore, V is square-integrable with $\mathbb{E}(V) = \mathbb{E}(U)$, which yields (15). Similarly, (16) follows from (4) as $20/(1 - 2^{-1/2}) \approx 68.28$. $\square$

Theorem 3.2 shows that $e^{-rT}(V + \alpha)$ is an unbiased estimator of the Asian option price that can be simulated in constant time with variance bounded by a constant independent of m. Simulating $\lceil \epsilon^{-2} \rceil$ independent copies of V yields an unbiased estimator of the option price with variance $O(\epsilon^2)$ in $O(m + \epsilon^{-2})$ expected time, including the $O(m)$ preprocessing cost of Algorithm M. Assuming that the variance of $f(A)$ is lower bounded by a constant independent of m, our estimator outperforms the conventional Monte Carlo method by a factor of order m. More precisely, simulating m independent copies of V has the same expected cost, up to a constant, as a single iteration of the standard Monte Carlo algorithm, but produces an unbiased estimator of the Asian option price with $O(1/m)$ variance. Theorem 3.3 shows how to construct another unbiased estimator of the Asian option price under A1 using the MLMC method.

Theorem 3.3. *Suppose A1 holds. Define U_l, $l \geq -1$, as in Theorem 3.2 and, for $0 \leq l \leq L$, let $\mu_l := \mathrm{Var}(U_l - U_{l-1})$ and*

$$n_l := \lfloor 1 + \frac{m\sqrt{\mu_l/|J_l|}}{\sum_{l'=0}^{L} \sqrt{\mu_{l'}|J_{l'}|}} \rfloor. \tag{19}$$

For $0 \leq l \leq L$, let $\bar{U}_l$ be the average of n_l independent copies of $U_l - U_{l-1}$. Assume that the estimators $\bar{U}_0, \ldots, \bar{U}_L$ are independent. Set $\bar{U} := \sum_{l=0}^{L} \bar{U}_l$. Then

$$\mathbb{E}(f(A)) = \mathbb{E}(\bar{U}) + \alpha, \tag{20}$$

and

$$m\mathrm{Var}(\bar{U}) \leq 240\kappa^2 \mathrm{Var}(F_m). \tag{21}$$

Furthermore, the expectation of the time required to simulate $\bar{U}$ is $O(m)$.

Assuming the variances μ_l, $0 \leq l \leq L$, are known, Theorem 3.3 shows that $e^{-rT}(\bar{U} + \alpha)$ is an unbiased estimator of the Asian option price that can be simulated in $O(m)$ time with variance $O(1/m)$. Once again, assuming that the variance of $f(A)$ is lower bounded by a constant independent of m, this estimator outperforms the conventional Monte Carlo estimator by a factor of order m. Simulating $\lceil \epsilon^{-2}/m \rceil$ independent copies of $\bar{U}$ yields an unbiased estimator of the option price with variance $O(\epsilon^2)$ in $O(m + \epsilon^{-2})$ expected time. The variances μ_l can be estimated by Monte Carlo simulation. Below are examples where A1 holds.

3.2.1 The Black-Scholes model

In this model, $F(t)$ satisfies the SDE

$$dF(t) = \sigma F(t) dW \tag{22}$$

on $[0, T]$, where σ is a constant volatility and W is a one-dimensional Brownian motion under Q. Given $J \subseteq \{1, \ldots, m\}$, let $n = |J|$, and let $0 = \tau_0 < \tau_1 < \cdots < \tau_n$ be the elements of the time grid $G = \{0\} \cup \{t_j : j \in J\}$, sorted in increasing order. Let $X_1, \ldots, X_n$ be independent standard Gaussian random variables. We simulate the forward prices on G in $O(n)$ time using the following recursive procedure (Glasserman 2004, Section 3.2.1):

$$F(\tau_k) = F(\tau_{k-1}) \exp(-\sigma^2 \frac{\tau_k - \tau_{k-1}}{2} + \sigma \sqrt{\tau_k - \tau_{k-1}} X_k),$$

$1 \leq k \leq n$. Then, for $j \in J$, we set $F_j = F(\tau_k)$, where k is the index such that $\tau_k = t_j$. Thus A1 holds for the Black-Scholes model. Furthermore, it is well-known that the forward price is square-integrable at any fixed date in this model.

3.2.2 Merton's jump-diffusion model

Here, the risk-neutral process for the forward price satisfies the jump-diffusion SDE (see (Merton 1976)):

$$\frac{dF(t)}{F(t-)} = -\lambda \mu dt + \sigma dW(t) + dJ(t) \tag{23}$$

on $[0, T]$, where W is a Brownian motion, $J(t) := \sum_{j=1}^{N(t)} (Y_j - 1)$, and $N(t)$ is a Poisson process with rate λ. If a jump occurs at time τ_j, then $S(\tau_j+) = S(\tau_j-) Y_j$, where $\ln(Y_j)$ is a Gaussian random variable with mean β and standard deviation γ. The model parameters satisfy the equation: $\mu + 1 = \exp(\beta + \gamma^2/2)$. We assume that W, N and the Y_j's are independent. An algorithm that simulates the forward price process on a discrete time grid of size n in $O(n)$ expected time is given in (Glasserman 2004, Section 3.5.1). Thus A1 holds for Merton's jump-diffusion model. A classical calculation based on (Glasserman 2004, Section 3.5.1) shows that the forward price is square-integrable at any fixed date in this model.

3.2.3 The Square-Root diffusion model

Here we assume that $F(t)$ satisfies the following SDE:

$$dF(t) = \sigma \sqrt{F(t)} dW(t)$$

on $[0, T]$, where W is a Brownian motion under Q, and $\sigma > 0$. The Square-Root diffusion model, introduced in (Cox and Ross 1976), is a special case of the CEV model. The appendix describes an algorithm that simulates the forward price on a discrete time grid of size n in $O(n)$ expected time and shows that F_m is square-integrable. Thus A1 holds for the Square-Root diffusion model.

It is well-known that the standard Euler scheme is not defined for Square-Root diffusions because it may produce negative forward prices. The related Cox-Ingersoll-Ross process has an implicit Euler scheme with a strong convergence of order 1 (see (Alfonsi 2015, Section 3.2)) under certain assumptions on the model parameters, but we are not aware of discretization schemes with positive strong order of convergence for Square-Root diffusions. Thus, previous MLMC methods based on the Euler or Milstein schemes are inapplicable to this process.

3.2.4 Kou's double exponential jump-diffusion model

The risk-neutral process for the forward price in this model (see (Kou 2002)) is given by (23) on $[0, T]$, where W is a Brownian motion, $J(t) := \sum_{j=1}^{N(t)}(Y_j - 1)$, and $N(t)$ is a Poisson process with rate λ. The Y_j's are i.i.d. positive random variables such that $X := \ln(Y_1)$ has density function

$$f_X(x) = p\eta_1 e^{-\eta_1 x}\mathbf{1}_{\{x\geq 0\}} + (1-p)\eta_2 e^{\eta_2 x}\mathbf{1}_{\{x<0\}},$$

where $\eta_1 > 1$, $\eta_2 > 0$, $0 \leq p \leq 1$. We assume that W, N and the Y_j's are independent. The martingale condition implies that

$$\mu + 1 = \mathbb{E}(Y_1) = p\frac{\eta_1}{\eta_1 - 1} + (1-p)\frac{\eta_2}{\eta_2 + 1}.$$

An algorithm that simulates the forward price process on a discrete time grid of size n in $O(n)$ expected time is given in (Glasserman 2004, Section 3.5.1). If $\eta_1 > 2$, a simple calculation shows that the Y_j's are square integrable which, by (Glasserman 2004, Eq. 3.81), guarantees that the second moment of $F(T)$ is finite.

3.2.5 The variance gamma model

Here we assume that $F(t) = F_0 \exp(\omega t + X(t))$, where $X(t) = X(t; \sigma, \nu, \theta)$ is a variance gamma process under Q (Madan, Carr and Chang 1998) with parameters $\sigma, \nu > 0$, and θ, where $\theta\nu + \sigma^2\nu/2 < 1$ and

$$\omega = \frac{1}{\nu}\ln(1 - \theta\nu - \sigma^2\nu/2).$$

The process $X(t)$ starts at 0, has independent increments, and can be simulated recursively at dates $\tau_0 = 0 < \tau_1 < \cdots < \tau_n$ as follows (Glasserman 2004, Section 3.5.2):

$$X(\tau_{i+1}) := X(\tau_i) + \theta Y + \sigma\sqrt{Y}Z, \tag{24}$$

where $Z \sim N(0,1)$ (i.e. Z has the standard normal distribution), and Y is independent of Z and has distribution $\text{Gamma}((\tau_{i+1} - \tau_i)/\nu, \nu)$. As a gamma random variable can be simulated in unit expected time (Devroye 1986), A1 holds for the variance gamma model. It follows from (24) that

$$\mathbb{E}(\exp(2X(T))) = \mathbb{E}(\exp(2(\theta + \sigma^2)Y)),$$

where $Y \sim \text{Gamma}(T/\nu, \nu)$. By a standard calculation, this implies that $F(T)$ is square integrable if $2(\theta + \sigma^2)\nu < 1$.

3.2.6 The NIG model

The inverse Gaussian distribution with parameters $\delta, \gamma > 0$ has density

$$f_{IG}(x) := \frac{\delta e^{\delta\gamma}}{\sqrt{2\pi}}x^{-3/2}\exp(-\frac{1}{2}(\delta^2 x^{-1} + \gamma^2 x)),\ x > 0.$$

The NIG model assumes that $F(t) = F_0 \exp(\omega t + X(t))$, where $X(t)$ is a normal inverse Gaussian process under Q (Glasserman 2004, Section 3.5.2) with parameters $\delta, \gamma > 0$ and β, where $2\beta + 1 < \gamma^2$ and

$$\omega := -\delta(\gamma - \sqrt{\gamma^2 - 2\beta - 1}). \tag{25}$$

The process $X(t)$ starts at 0, has independent increments, and can be simulated recursively at dates $\tau_0 = 0 < \tau_1 < \cdots < \tau_n$ as follows (Glasserman 2004, Section 3.5.2):

$$X(\tau_{i+1}) := X(\tau_i) + \beta Y + \sqrt{Y}Z,\ Z \sim N(0,1),$$

where Y is independent of Z and has an inverse Gaussian distribution with parameters $\delta(\tau_{i+1} - \tau_i)$ and γ. Hence

$$\mathbb{E}(\exp(X(T))) = \mathbb{E}(\exp((\beta + 1/2)Y)),$$

where Y has an inverse Gaussian distribution with parameters δT and γ. By (Cont and Tankov 2004, Table 4.4), for $u < \gamma^2/2$,

$$\mathbb{E}(e^{uY}) = \exp(\delta T(\gamma - \sqrt{\gamma^2 - 2u})),$$

and so (25) guarantees that $(F(t))$, $0 \le t \le T$, is a martingale. A similar calculation shows that $F(T)$ is square integrable if $4(\beta + 1) < \gamma^2$. The inverse Gaussian distribution can be simulated in unit expected time (Glasserman 2004, Section 3.5.2), and so A1 holds for the NIG model.

3.2.7 Multi-dimensional geometric Brownian motion

A1 also holds if the underlying is the average of assets that follow a multi-dimensional geometric Brownian motion. An algorithm that jointly simulates such assets is given in (Glasserman 2004, Section 3.2.3).

3.3 The approximate simulation case

For $J \subseteq \{1, \ldots, m\}$, let $\mathbb{R}^J$ denote the set of vectors of dimension $|J|$, indexed by the elements of J.

Assumption 2 (A2). There are constants c_1, c_2 and $\beta \in [1, 2]$ such that, for $l \ge 0$ and $J \subseteq \{1, \ldots, m\}$, there is a random vector $\hat{F} = \hat{F}(J, l) \in \mathbb{R}^J$ such that $||\hat{F}_j - F_j||^2 \le c_2 2^{-\beta l}$ for any $j \in J$. For $l \ge 1$ and $J' \subseteq J \subseteq \{1, \ldots, m\}$, the expected time required to simulate the vector $(\hat{F}(J', l-1), \hat{F}(J, l))$ is at most $c_1(|J| + 2^l)$.

The first condition in A2 says that, for $l \ge 0$ and $J \subseteq \{1, \ldots, m\}$, the forward price F_j is approximated by $\hat{F}_j$ with "mean square error" at most $c_2 2^{-\beta l}$ for any $j \in J$. The second condition gives an upper bound on the expected time to jointly simulate $\hat{F}(J', l-1)$ and $\hat{F}(J, l)$. It is shown in Sections 3.3.2 and 3.3.3 that, under certain conditions and using suitable discretization grids, A2 holds when the Euler or Milstein schemes are used to approximately simulate forward prices.

Assume now that A2 holds. For $l \ge 0$, let $\hat{F}^l := \hat{F}(J_l, l)$ and

$$\hat{A}_l := \sum_{j \in J_l} w_j \hat{F}^l_j + \frac{1}{2} \sum_{(i,k) \in \mathcal{P}_l} W(i+1, k-1)(\hat{F}^l_i + \hat{F}^l_k). \tag{26}$$

Thus $\hat{A}_l$ is obtained from A by replacing each F_j with $\hat{F}^l_j$ if $j \in J_l$ and by $(\hat{F}^l_i + \hat{F}^l_k)/2$ if $(i, k) \in \mathcal{P}_l$ and $i < j < k$. Note that $\hat{A}_l$ is a deterministic linear function of the vector $\hat{F}^l$. Proposition 3.2 gives a bound on the L^2-distance between $\hat{A}_0$ and $W(1, m)F_0$ on one hand, and between $\hat{A}_l$ and A on the other hand.

Proposition 3.2. *If A2 holds then $||\hat{A}_0 - W(1, m)F_0||^2 \le c_3$ and $||\hat{A}_l - A||^2 \le c_3 2^{-\beta l}$ for $l \ge 0$, where $c_3 = 2(c_2 + \mathrm{Var}(F_m))$.*

Theorem 3.4 shows how to construct an unbiased estimator of the Asian option price under A2, with $\beta > 1$. The case $\beta = 1$ will be considered in Theorem 3.5.

Theorem 3.4. *Suppose A2 holds with $\beta > 1$. Let $N \in \mathbb{N}$ be an integral random variable independent of $(\hat{F}(J_l, l) : l \ge 0)$ such that $\Pr(N = l) = p_l$ for non-negative integer l, where p_l*

is given by (3). *Let* $\hat{U}_l := f(\hat{A}_l) - \alpha$ *for* $l \geq 0$, *and let* $\hat{V} := (\hat{U}_N - \hat{U}_{N-1})/p_N$, *where* $\hat{U}_{-1} := 0$. *Then* $\hat{V}$ *is square-integrable and*

$$\mathbb{E}(f(A)) = \mathbb{E}(\hat{V}) + \alpha. \tag{27}$$

Furthermore, $\mathrm{Var}(\hat{V})$ *and the expectation of the time required to simulate* $\hat{V}$ *are upper-bounded by constants independent of* m.

As per the discussion following Theorem 3.2, Theorem 3.4 shows that $e^{-rT}(\hat{V}+\alpha)$ is an unbiased estimator of the Asian option price that can be simulated in constant time and with variance bounded by a constant independent of m. Independent $\lceil \epsilon^{-2} \rceil$ runs of this estimator yield an unbiased estimator of the Asian option price with variance $O(\epsilon^2)$ in $O(m+\epsilon^{-2})$ expected time.

Theorem 3.5 constructs an estimator of the option price with an arbitrarily small bias when A2 holds with $\beta = 1$.

Theorem 3.5. *Suppose A2 holds with* $\beta = 1$. *Fix* $\epsilon \in (0, 1/2)$ *and set* $I := \lceil 2\log_2(1/\epsilon) \rceil$. *Let* $N \in \mathbb{N}$ *be an integral random variable independent of* $(\hat{F}(J_l, l) : l \geq 0)$ *such that* $\Pr(N = l) = 2^{-(l+1)}$ *for* $l \in \mathbb{N}$. *Let* $\hat{U}_l := f(\hat{A}_l) - \alpha$ *for* $l \geq 0$, *and let*

$$\hat{V} := \frac{\hat{U}_N - \hat{U}_{N-1}}{p_N} \mathbf{1}_{\{N \leq I\}},$$

where $\hat{U}_{-1} := 0$. *Then* $\hat{V}$ *is square-integrable and*

$$(\mathbb{E}(\hat{V}) + \alpha - \mathbb{E}(f(A)))^2 \leq c_3 \kappa^2 \epsilon^2, \tag{28}$$

where c_3 *is defined as in Proposition 3.2. Furthermore, there are constants* c_4 *and* c_5 *independent of* m *and of* ϵ *such that* $\mathrm{Var}(\hat{V}) \leq c_4 \ln(1/\epsilon)$ *and the expectation of the time required to simulate* $\hat{V}$ *is upper-bounded by* $c_5 \ln(1/\epsilon)$.

Under the assumptions of Theorem 3.5, the Asian option price can be calculated with $O(\epsilon^2)$ mean square error in $O(m + \epsilon^{-2} \ln^2(1/\epsilon))$ expected time as follows. We simulate n independent copies of $\hat{V}$, where $n = \lceil \ln(1/\epsilon)\epsilon^{-2} \rceil$, and take their average $\hat{V}_n$. Since $\mathrm{Var}(\hat{V}_n) = \mathrm{Var}(\hat{V})/n$, we have $\mathrm{Var}(\hat{V}_n) \leq c_4 \epsilon^2$. Furthermore, as $\mathbb{E}(\hat{V}_n) = \mathbb{E}(\hat{V})$, it follows from (28) that

$$(\mathbb{E}(\hat{V}_n) + \alpha - \mathbb{E}(f(A)))^2 \leq c_3 \kappa^2 \epsilon^2.$$

Since the mean square error is the sum of the variance and squared bias, we conclude that

$$||\hat{V}_n + \alpha - \mathbb{E}(f(A))||^2 \leq (c_4 + c_3\kappa^2)\epsilon^2.$$

Thus $e^{-rT}(\bar{V} + \alpha)$ is an estimate of the Asian option price $e^{-rT}\mathbb{E}(f(A))$ with mean square error $O(\epsilon^2)$. The total expected time to simulate $\hat{V}_n$ is $O(m + \ln^2(\epsilon)\epsilon^{-2})$, including the cost of Algorithm M.

The rest of this section shows that A2 holds when the forward price follows a continuous diffusion process satisfying certain conditions. For a vector or a matrix z, denote by $|z|$ the square root of the sum of the squared entries of z.

3.3.1 The Euler scheme

We recall here the Euler scheme applied to SDEs. Consider a d-dimensional process $(H(t))$, $0 \leq t \leq T$, which is a strong solution to the SDE

$$dH(t) = a(H(t), t)\, dt + b(H(t), t)\, dW, \ H(0) = H_0, \tag{29}$$

where $a : \mathbb{R}^d \times \mathbb{R}^+ \to \mathbb{R}^d$ is a d-dimensional vector function, $b : \mathbb{R}^d \times \mathbb{R}^+ \to \mathbb{R}^{d\times d'}$ is a $d \times d'$ matrix function, and W is a d'-dimensional Brownian motion. Consider a deterministic time-grid $G = \{\tau_0, \tau_1, \dots, \tau_n\}$, with $0 = \tau_0 < \tau_1 < \cdots < \tau_n = T$, and let $\delta := \max_{0\le k\le n-1}(\tau_{k+1} - \tau_k)$. The Euler scheme can be used to approximate the discrete process $(H(t))$, $t \in G$, by the random sequence $\tilde{H} = \tilde{H}(G)$ defined recursively as follows: $\tilde{H}_0 := H_0$ and, for $0 \le k \le n-1$,

$$\tilde{H}_{k+1} := \tilde{H}_k + a(\tilde{H}_k, \tau_k)(\tau_{k+1} - \tau_k) + b(\tilde{H}_k, \tau_k)(W(\tau_{k+1}) - W(\tau_k)). \tag{30}$$

Suppose that

$$|a(x,t) - a(y,t)| + |b(x,t) - b(y,t)| \le K_1|x - y| \text{ (Lipschitz condition)}, \tag{31}$$

$$|a(x,t)| + |b(x,t)| \le K_2(1 + |x|) \text{ (Linear Growth condition)}, \tag{32}$$

and

$$|a(x,t) - a(x,s)| + |b(x,t) - b(x,s)| \le K_3(1 + |x|)|t - s|^{1/2} \tag{33}$$

for all $s,t \in [0,T]$ and $x,y \in \mathbb{R}^d$, where K_1, K_2 and K_3 are constants. Then it follows from the proof of (Kloeden and Platen 1992, Theorem 10.2.2) that

$$\mathbb{E}(\max_{0\le k\le n} |\tilde{H}_k - H(\tau_k)|^2) \le K_1'\delta, \tag{34}$$

where K_1' is a constant.

3.3.2 The Euler scheme and Assumption A2

Assume that $F(t)$ is the first component of a d-dimensional process $(H(t))$, $0 \le t \le T$, that is a strong solution to the SDE (29), and that W is a d'-dimensional Brownian motion under Q. Suppose that the conditions (31), (32) and (33) hold and that, for $(x,t) \in \mathbb{R}^d \times [0,T]$, $a(x,t)$ and $b(x,t)$ can be calculated in constant time. For $J \subseteq \{1,\dots,m\}$ and $l \ge 0$, let

$$G(J,l) := \{t_j : j \in J\} \cup \{i2^{-l}T : 0 \le i \le 2^l\}.$$

Let $0 = \tau_0 < \tau_1 < \cdots < \tau_n = T$ be the elements of $G(J,l)$ sorted in increasing order. Note that $n \le |J| + 2^l$ and that the maximum distance between two consecutive elements of $G(J,l)$ is at most $2^{-l}T$. Construct $\tilde{H} = \tilde{H}(G(J,l))$ via (30), and define $\hat{F} = \hat{F}(J,l) \in \mathbb{R}^J$ as follows. For $j \in J$, let $\hat{F}_j$ be the first component of $\tilde{H}_k$, where k is the index such that $\tau_k = t_j$. In other words, $\hat{F}$ is the vector of first components of the "restriction" of $\tilde{H}$ to the dates corresponding to J.

It follows from (34) that

$$\mathbb{E}(\max_{0\le k\le n} |\tilde{H}_k - H(\tau_k)|^2) \le K_1' 2^{-l}T,$$

where K_1' is a constant, and so $||\hat{F}_j - F_j||^2 \le K_1' 2^{-l}T$ for $j \in J$. Furthermore, for $l \ge 1$ and $J' \subseteq J \subseteq \{1,\dots,m\}$, the grid $G(J', l-1)$ is contained in $G(J,l)$. The vector $(\hat{F}(J', l-1), \hat{F}(J,l))$ can thus be simulated in at most $c_1(|J| + 2^l)$ time, where c_1 is a constant independent of m, by first simulating W on the elements of $G(J,l)$ and then using the same W to calculate recursively $\tilde{H}(G(J,l))$ and $\tilde{H}(G(J', l-1))$ via (30). Thus A2 holds with $\beta = 1$ for the Euler scheme described above.

3.3.3 The one-dimensional Milstein scheme

Assume now that $F(t)$ is a strong solution to the SDE

$$dF(t) = b(F(t), t)\, dW,$$

where $b \in \mathcal{C}^{3,1}(\mathbb{R} \times \mathbb{R}^+)$ is a real-valued function and W is a one-dimensional Brownian motion under Q. Suppose that, for $(x,t) \in \mathbb{R} \times [0,T]$, $b(x,t)$ can be calculated in constant time. For $J \subseteq \{1,\dots,m\}$ and $l \geq 0$, define recursively the sequence $F^* = F^*(J,l)$ via the Milstein scheme (Kloeden and Platen 1992, p. 345) as follows: set $F_0^* := F_0$ and

$$F_{k+1}^* := F_k^* + b(F_k^*, \tau_k)(\Delta W) + \frac{1}{2} b(F_k^*, \tau_k) \frac{\partial b}{\partial x}(F_k^*, \tau_k)((\Delta W)^2 - (\tau_{k+1} - \tau_k)), \tag{35}$$

$0 \leq k \leq n-1$, where $\Delta W := W(\tau_{k+1}) - W(\tau_k)$ and $\tau_0, \dots, \tau_n$ are defined as in Section 3.3.2. Kloeden and Platen (1992, Theorem 10.6.3) show that, under certain conditions on b,

$$\mathbb{E}(\max_{0 \leq k \leq n} (F_k^* - F(\tau_k))^2) \leq K_2' \delta^2,$$

where K_2' is a constant, and $\delta := \max_{0 \leq k \leq n-1}(\tau_{k+1} - \tau_k)$. These conditions include the existence of the derivatives $\partial^i b/\partial x^i$, $1 \leq i \leq 3$, $\partial b/\partial t$, and $\partial^2 b/\partial x \partial t$. For $j \in J$, set $\hat{F}_j := F_k^*$, where k is the index such that $\tau_k = t_j$. As for the Euler scheme, the vector $(\hat{F}(J', l-1), \hat{F}(J,l))$ can be simulated in at most $c_1(|J| + 2^l)$ time, where c_1 is a constant independent of m, by first simulating W on the elements of $G(J,l)$ and then using the same W to calculate recursively $F^*(J,l)$ and $F^*(J', l-1)$ via (35). Thus A2 holds with $\beta = 2$ for the Milstein scheme described above.

4 Numerical experiments

The simulation experiments were performed on a desktop PC with an Intel Pentium 2.90 GHz processor and 4 GB of RAM. The codes were written in the C++ programming language. Our experiments assume that interest rates are constant and equal to r. We have implemented the RMLMC method of Theorem 3.2, and the MLMC method of Theorem 3.3, but replaced m with $30m$ in (19) in order to mitigate the rounding effect and achieve greater efficiency. The variances μ_l were estimated by Monte Carlo simulation using 10^3 independent runs. The RMLMC method based on the Milstein scheme (RMLMC-Milstein) was implemented for the Black-Scholes model as described in Theorem 3.4, with $\beta = 2$, without solving explicitly (22). In Tables 1 through 6, "Price" is the estimated Asian option price obtained via n independent replications, and "Std" is the estimated price standard error. Note that the variance of the estimated price is equal to the variance of a single run divided by n. The variable "Cost" refers to the total number of simulated underlying prices throughout the n replications. Thus, Cost $\times$ Std^2 is an estimate of the work-normalized variance. As per the discussions following Theorems 3.2, 3.3 and 3.4, n is set to a constant independent of m for the RMLMC and RMLMC-Milstein algorithms, and is inversely proportional to m for the MLMC algorithm. The constants were chosen so that the variable "Cost" has the same order of magnitude for the studied algorithms. The variable "Time" refers to the total running time in seconds of the n replications and includes, for the MLMC algorithm, the time needed to estimate the μ_l's. As the variance of a single run of the standard Monte Carlo estimator is $e^{-2rT}\mathrm{Var}(f(A))$, we measure the performance of a method through the following factors, in line with (Glynn and Whitt 1992):

$$\mathrm{Eff}_{\mathrm{cost}} := \frac{m e^{-2rT} \mathrm{Var}(f(A))}{\mathrm{Cost} \times \mathrm{Std}^2},$$

Table 1: Pricing at the money average price calls in the Black-Scholes model

m	Method	n	Price	Std	Cost	Cost $\times$ Std^2	Time	Eff$_{\text{cost}}$	Eff$_{\text{time}}$
125	RMLMC	1×10^8	0.3523	1.5×10^{-4}	2.10×10^8	4.5	24	12	8
	MLMC	8×10^4	0.3523	1.5×10^{-4}	2.11×10^8	4.5	21	12	10
	RMLMC-Mil.	1×10^8	0.3522	1.4×10^{-4}	3.30×10^8	6.4	36	8	6
250	RMLMC	1×10^8	0.3512	1.5×10^{-4}	2.13×10^8	4.7	24	24	17
	MLMC	4×10^4	0.3508	1.5×10^{-4}	2.18×10^8	4.7	22	23	19
	RMLMC-Mil.	1×10^8	0.3511	1.4×10^{-4}	3.33×10^8	6.6	36	17	13
500	RMLMC	1×10^8	0.3506	1.5×10^{-4}	2.15×10^8	4.8	25	45	31
	MLMC	2×10^4	0.3504	1.5×10^{-4}	2.18×10^8	4.9	22	44	35
	RMLMC-Mil.	1×10^8	0.3506	1.4×10^{-4}	3.35×10^8	6.8	36	32	23

Table 2: Pricing average strike calls in the Black-Scholes model

m	Method	n	Price	Std	Cost	Cost $\times$ Std^2	Time	Eff$_{\text{cost}}$	Eff$_{\text{time}}$
$m = 125$	RMLMC	1×10^8	0.3632	2.0×10^{-4}	1.42×10^8	5.4	18	17	11
	MLMC	8×10^4	0.3632	1.4×10^{-4}	2.08×10^8	3.9	21	23	19
	RMLMC-Mil.	1×10^8	0.3631	2.0×10^{-4}	2.62×10^8	10	30	9	6
$m = 250$	RMLMC	1×10^8	0.3627	2.0×10^{-4}	1.42×10^8	5.6	18	34	21
	MLMC	4×10^4	0.3628	1.4×10^{-4}	2.03×10^8	4	20	47	38
	RMLMC-Mil.	1×10^8	0.3628	2.0×10^{-4}	2.62×10^8	10	30	18	13
$m = 500$	RMLMC	1×10^8	0.3626	2.0×10^{-4}	1.42×10^8	5.7	18	61	38
	MLMC	2×10^4	0.3626	1.4×10^{-4}	2.08×10^8	4.2	21	83	66
	RMLMC-Mil.	1×10^8	0.3626	2.0×10^{-4}	2.62×10^8	11	30	32	23

and

$$\mathrm{Eff}_{\text{time}} := \frac{\mathrm{Time(MC)}\, e^{-2rT}\mathrm{Var}(f(A))}{\mathrm{Time} \times \mathrm{Std}^2},$$

where "Time(MC)" is the running in seconds of a single run of the standard Monte Carlo method. Here $\mathrm{Var}(f(A))$ is estimated via 10^5 independent samples of A. The payoff of an average price call with strike K is $\max(m^{-1}(\sum_{i=1}^{m} S_i) - K, 0)$, while the payoff of an average strike call is $\max(S_m - (m-1)^{-1}(\sum_{i=1}^{m-1} S_i), 0)$, where S_i is the underlying price at $t_i := iT/m$.

4.1 The Black-Scholes model

In our experiments, the underlying is a stock S with no dividends, and the model parameters are $S_0 = 2$, $\sigma = 50\%$, $r = 5\%$, and $T = 2$. These values are taken from (Linetsky 2004). Table 1 gives our results for average price calls with $K = 2$ and selected values of m. For the RMLMC and RMLMC-Milstein algorithms, the cost of $n = 10^8$ independent replications, and the standard error of the estimated price, are roughly independent of m. These results are consistent with Theorems 3.2 and 3.4. Likewise, for the MLMC algorithm, the cost of $n = 10^7/m$ independent replications and the standard error of the estimated price are roughly independent of m. This is consistent with Theorem 3.3, which implies that the expected cost (resp. variance) of a single run of this algorithm is $O(m)$ (resp. $O(m^{-1})$). For the RMLMC, MLMC and RMLMC-Milstein algorithms, Eff$_{\text{cost}}$ and Eff$_{\text{time}}$ are roughly proportional to m. Thus, these algorithms outperform the standard Monte Carlo algorithm by a factor of order m. Table 2 reports similar results for average strike calls. In Table 1, the RMLMC and MLMC methods have a similar performance, as indicated by Eff$_{\text{cost}}$ and Eff$_{\text{time}}$. In Table 2, the MLMC method slightly outperforms the RMLMC method. This can be explained by observing that the frequencies n_l in Theorem 3.3 are near-optimal, which is not always the case for the probabilities p_l in Theorem 3.2. In both tables, the RMLMC method outperforms the RMLMC-Milstein algorithm. In the Black-Scholes model, the RMLMC-Milstein algorithm has no advantages over RMLMC because the forward price process can be simulated exactly. Table 3 reports prices of Asian options produced by the RMLMC algorithm with a very large value of m. The cost and standard error of the estimated price in Table 3 are essentially the same as the corresponding values in Tables 1 and 2. The price of the average price call in Table 3 is very close to the price of the continuously monitored average price call given in (Linetsky 2004), which is 0.350095.

Table 3: Randomized multilevel Monte Carlo pricing of Asian calls in the Black-Scholes model. The average price call at the money.

	m	n	Price	Std	Cost	Cost $\times$ Std^2	Time
Average price	10^7	10^8	0.3501	1.4×10^{-4}	2.21×10^8	5.1	26
Average strike	10^7	10^8	0.3625	2.1×10^{-4}	1.42×10^8	6.0	19

4.2 Merton's jump-diffusion model

In our experiments, the underlying is an index with constant dividend yield q. The model parameter values used are $S_0 = 2$, $\sigma = 17.65\%$, $r = 5.59\%$, $q = 1.14\%$, $\lambda = 8.90\%$, $\beta = -88.98\%$, and $\gamma = 45.05\%$. Except for the spot price, these values are taken from (Andersen and Andreasen 2000), where they were obtained by fitting option prices with maturities ranging from one month to ten years. We set $T = 2$. Table 4 gives prices of average price calls using the RMLMC and MLMC algorithms. Here again, the cost and the standard error of the estimated price of 10^8 (resp. $10^7/m$) independent replications of the RMLMC (resp. MLMC) algorithm are roughly independent of m. For both algorithms, $\mathrm{Eff}_{\mathrm{cost}}$ and $\mathrm{Eff}_{\mathrm{time}}$ are roughly proportional to m. The RMLMC and MLMC methods have a similar performance.

Table 4: Pricing at the money average price calls in Merton's jump-diffusion model

m	Method	n	Price	Std	Cost	Cost $\times$ Std^2	Time	$\mathrm{Eff}_{\mathrm{cost}}$	$\mathrm{Eff}_{\mathrm{time}}$
$m = 125$	RMLMC	1×10^8	0.19306	5.0×10^{-5}	2.10×10^8	0.53	63	13	12
	MLMC	8×10^4	0.19303	5.0×10^{-5}	2.16×10^8	0.53	60	13	13
$m = 250$	RMLMC	1×10^8	0.19242	5.1×10^{-5}	2.13×10^8	0.55	63	26	23
	MLMC	4×10^4	0.19244	5.0×10^{-5}	2.22×10^8	0.56	62	25	24
$m = 500$	RMLMC	1×10^8	0.19208	5.1×10^{-5}	2.15×10^8	0.56	65	50	44
	MLMC	2×10^4	0.19208	5.0×10^{-5}	2.27×10^8	0.57	63	50	47

4.3 The Square-Root diffusion model

The model parameter values in our experiments are $S_0 = 2$, $r = 5\%$, $\sigma = 0.4$, and $T = 2$. Table 5 gives prices of average price calls using the RMLMC and MLMC algorithms. Once again, the cost and the standard error of the estimated price of the n independent replications of the RMLMC and MLMC algorithms are roughly independent of m. The efficiency of both algorithms is roughly proportional to m. The RMLMC and MLMC methods have a similar performance.

Table 5: Pricing at the money average price calls in the Square-Root diffusion model

m	Method	n	Price	Std	Cost	Cost $\times$ Std^2	Time	$\mathrm{Eff}_{\mathrm{cost}}$	$\mathrm{Eff}_{\mathrm{time}}$
$m = 125$	RMLMC	1×10^8	0.21832	6.3×10^{-5}	2.10×10^8	0.82	158	13	13
	MLMC	8×10^4	0.21842	6.2×10^{-5}	2.15×10^8	0.82	157	13	13
$m = 250$	RMLMC	1×10^8	0.21768	6.3×10^{-5}	2.13×10^8	0.85	160	26	26
	MLMC	4×10^4	0.21772	6.2×10^{-5}	2.22×10^8	0.85	164	26	26
$m = 500$	RMLMC	1×10^8	0.21724	6.4×10^{-5}	2.15×10^8	0.87	162	50	50
	MLMC	2×10^4	0.21729	6.4×10^{-5}	2.21×10^8	0.90	162	49	49

4.4 The variance gamma model

The model parameter values in our experiments are $S_0 = 2$, $r = 5\%$, $\sigma = 0.1213$, $\nu = 0.1686$, $\theta = -0.1436$, and $T = 2$. The values of ν, σ and θ are taken from (Madan, Carr and Chang 1998). The simulation results, reported in Table 6, are similar in nature to those in Tables 1, 4 and 5. In particular, the efficiency of the RMLMC and MLMC algorithms is roughly proportional to m.

Table 6: Pricing at the money average price calls in the variance gamma model

m	Method	n	Price	Std	Cost	Cost $\times$ Std^2	Time	Eff$_{\text{cost}}$	Eff$_{\text{time}}$
$m = 125$	RMLMC	1×10^8	0.13832	3.1×10^{-5}	2.10×10^8	0.20	69	14	13
	MLMC	8×10^4	0.13825	3.1×10^{-5}	2.21×10^8	0.21	67	14	13
$m = 250$	RMLMC	1×10^8	0.13782	3.1×10^{-5}	2.13×10^8	0.21	64	27	21
	MLMC	4×10^4	0.13771	3.1×10^{-5}	2.21×10^8	0.22	61	26	22
$m = 500$	RMLMC	1×10^8	0.13757	3.2×10^{-5}	2.15×10^8	0.22	68	53	42
	MLMC	2×10^4	0.13753	3.1×10^{-5}	2.21×10^8	0.22	67	52	43

5 Conclusion

We have described a general MLMC framework to estimate the price of an Asian option monitored at m dates. We assume the existence of a linear relation between the underlying and forward prices, and that the underlying price is square-integrable at maturity T. Our approach yields unbiased estimators with variance $O(\epsilon^2)$ in $O(m + \epsilon^{-2})$ expected time for a variety of processes that can be simulated exactly and, via the Milstein scheme, processes driven by scalar SDEs satisfying certain conditions. Using the Euler scheme, our approach estimates the Asian option price with mean square error $O(\epsilon^2)$ in $O(m + (\ln(\epsilon))^2\epsilon^{-2})$ expected time for processes driven by multidimensional SDEs satisfying certain conditions. Numerical experiments confirm that our approach outperforms the conventional Monte Carlo method by a factor proportional to m.

A direction for future research is to extend our approach to models beyond those studied in this paper. Simulating the Heston and SABR models, whose dynamics violate the Lipschitz condition (31), with the standard Euler and Milstein schemes can lead to erratic results (Glasserman and Kim 2011, Cai, Song and Chen 2017). Combining our techniques with provably efficient simulation methods for these models is left for future work.

Acknowledgments

This research has been presented at the 35th Spring International Conference of the French Finance Association, May 2018. The author thanks conference participants, the editor and four reviewers for helpful comments and suggestions. This work was achieved through the Laboratory of Excellence on Financial Regulation (Labex ReFi) under the reference ANR-10-LABX-0095.

A Proof of Proposition 2.1

As $(x + x')^2 \leq 2(x^2 + x'^2)$ for any real numbers x and x', if X and X' are square-integrable random variables,

$$||X + X'||^2 \leq 2(||X||^2 + ||X'||^2). \tag{36}$$

For $l \geq 1$, by applying (36) with $X = Y_l - Y$ and $X' = Y_{l-1} - Y$, it follows that

$$||Y_l - Y_{l-1}||^2 \leq 2(||Y_l - Y||^2 + ||Y_{l-1} - Y||^2). \tag{37}$$

Since $||Y_{l-1} - Y||^2 \leq 4\nu 2^{-\beta l}$ by (2), it follows that from (37) that

$$||Y_l - Y_{l-1}||^2 \leq 10\nu 2^{-\beta l}. \tag{38}$$

As $||Y_0||^2 \leq \nu$, (38) holds also for $l = 0$. Thus, as $p_l \geq 2^{-1-(\beta+1)l/2}$,

$$\begin{aligned} \sum_{l=0}^{\infty} \frac{||Y_l - Y_{l-1}||^2}{p_l} &\leq 20\nu \sum_{l=0}^{\infty} 2^{-(\beta-1)l/2} \\ &= \frac{20\nu}{1 - 2^{-(\beta-1)/2}}. \end{aligned}$$

By Theorem 2.1, we conclude that Z is square-integrable with $\mathbb{E}(Z) = \mathbb{E}(Y)$, and that (4) holds.

We now prove (5). As observed in (Rhee and Glynn 2015), $C = \sum_{l=0}^{\infty} p_l C_l$. Since $p_l \leq 2^{-(\beta+1)l/2}$,

$$C \leq c \sum_{l=0}^{\infty} 2^{-(\beta-1)l/2},$$

which concludes the proof. □

B Proof of Proposition 2.2

We apply Theorem 2.1 to the sequence $(Y_{\min(l,I)} : l \geq 0)$ and Y_I. Thus $Z = Z_I$, and so Z_I is square-integrable, $\mathbb{E}(Z_I) = \mathbb{E}(Y_I)$, and

$$||Z_I||^2 = \sum_{l=0}^{I} \frac{||Y_l - Y_{l-1}||^2}{p_l}.$$

Hence

$$\begin{aligned} (\mathbb{E}(Z_I - Y))^2 &= (\mathbb{E}(Y_I - Y))^2 \\ &\leq ||Y_I - Y||^2, \end{aligned}$$

which yields (7). On the other hand, for $l \geq 1$, as $||Y_{l-1} - Y||^2 \leq 2\nu 2^{-l}$ by (6), it follows from (37) that

$$||Y_l - Y_{l-1}||^2 \leq 6\nu 2^{-l}. \tag{39}$$

Since $||Y_0||^2 \leq \nu$, (39) also holds for $l = 0$. Hence,

$$\sum_{l=0}^{I} \frac{||Y_l - Y_{l-1}||^2}{p_l} \leq 12\nu(I+1),$$

which implies (8). Finally, the expected cost of computing Z_I is $\sum_{l=0}^{I} p_l C_l$, which is upper-bounded by cI since $p_l C_l \leq c/2$. □

C Proof of Proposition 3.1

We first show (10). As this equation clearly holds for $l \geq L$, we assume that $0 \leq l \leq L-1$. Let j and j' be two elements of J_l, with $j < j'$. As $j \leq j' - 1$,

$$\begin{aligned} \lfloor 2^l W'(1,j) \rfloor &\leq \lfloor 2^l W'(1,j'-1) \rfloor \\ &\leq 2^l W'(1,j'-1) \\ &< \lfloor 2^l W'(1,j') \rfloor, \end{aligned}$$

where the last equation follows from (9). Thus the map $j \mapsto \lfloor 2^l W'(1,j) \rfloor$ from J_l to $\{0,\ldots,2^l\}$ is strictly increasing. This implies (10).

We now show (11). As this relation is obvious when $l \geq L-1$, assume that $0 \leq l \leq L-2$. Since $2\lfloor x \rfloor \leq \lfloor 2x \rfloor$ for $x \in \mathbb{R}$, for any an element j of J_l,

$$2^{l+1} W'(1,j-1) < 2\lfloor 2^l W'(1,j) \rfloor \leq \lfloor 2^{l+1} W'(1,j) \rfloor,$$

where the first equation follows from (9). Thus, $j \in J_{l+1}$. This implies (11). □

D Proof of Theorem 3.1

Proposition D.1 proves standard properties of square-integrable martingales.

Proposition D.1. *For* $0 \le i \le j \le k \le m$,

$$\mathbb{E}(F_i(F_k - F_j)) = 0, \tag{40}$$

and

$$||F_i|| \le ||F_j||. \tag{41}$$

Moreover,

$$||F_j - F_i||^2 \le ||F_k||^2 - ||F_i||^2. \tag{42}$$

Proof. Let $\mathcal{F} = (\mathcal{F}_i)$, $0 \le i \le m$, be the natural filtration of the random process (F_i), $0 \le i \le m$. By the tower law,

$$\begin{aligned}
\mathbb{E}(F_i(F_k - F_j)) &= \mathbb{E}(\mathbb{E}(F_i(F_k - F_j)|\mathcal{F}_j)) \\
&= \mathbb{E}(F_i \mathbb{E}(F_k - F_j|\mathcal{F}_j)) \\
&= 0.
\end{aligned}$$

The last equation follows from the fact that (F_i), $0 \le i \le m$, is a martingale with respect to $\mathcal{F}$. This implies (40). In particular, $\mathbb{E}(F_i(F_j - F_i)) = 0$. As $F_j = (F_j - F_i) + F_i$,

$$||F_j||^2 = ||F_j - F_i||^2 + ||F_i||^2,$$

which proves (41). The inequality $||F_j|| \le ||F_k||$ then implies (42). □ □

We next prove the following proposition.

Proposition D.2. *For* $l \ge 0$, *if* $(i, k) \in \mathcal{P}_l$ *then* $W'(i+1, k-1) \le 2^{-l}$.

Proof. The desired inequality clearly holds if $k = i+1$. Assume that $k > i+1$. Thus $l \le L-1$. For any integer j in $[i+1, k-1]$, since $j \notin J_l$, we have $2^l W'(1, j-1) \ge \lfloor 2^l W'(1, j) \rfloor$, and so

$$\lfloor 2^l W'(1, j-1) \rfloor = \lfloor 2^l W'(1, j) \rfloor. \tag{43}$$

Hence

$$\begin{aligned}
2^l W'(1, k-1) - 1 &\le \lfloor 2^l W'(1, k-1) \rfloor \\
&= \lfloor 2^l W'(1, i) \rfloor \\
&\le 2^l W'(1, i).
\end{aligned}$$

The second equation follows from (43). As $W'(i+1, k-1) = W'(1, k-1) - W'(1, i)$, this completes the proof. □ □

We now prove Theorem 3.1. By (13) and the relation $J_0 = \{m\}$,

$$A_0 = w_m F_m + \frac{1}{2} W(1, m-1)(F_0 + F_m).$$

As $W(1, m) = W(1, m-1) + w_m$, it follows that

$$A_0 - W(1, m) F_0 = (\frac{1}{2} W(1, m-1) + w_m)(F_m - F_0),$$

and so $||A_0 - W(1, m) F_0|| \le ||F_m - F_0||$. As $\mathbb{E}(F_m) = F_0$, this implies the desired bound on $||A_0 - W(1, m) F_0||^2$.

Fix now $l \geq 0$. For $(i,k) \in \mathcal{P}_l$, let

$$B_i := \sum_{j=i+1}^{k-1} w_j(F_j - F_i) \text{ and } B'_i := \sum_{j=i+1}^{k-1} w_j(F_j - F_k).$$

Rewriting (13) as

$$A_l = \sum_{j \in J_l} w_j F_j + \frac{1}{2} \sum_{(i,k)\in\mathcal{P}_l} \sum_{j=i+1}^{k-1} w_j(F_i + F_k),$$

and noting that

$$A = \sum_{j \in J_l} w_j F_j + \sum_{(i,k)\in\mathcal{P}_l} \sum_{j=i+1}^{k-1} w_j F_j,$$

it follows that

$$A - A_l = \frac{1}{2} \sum_{(i,k)\in\mathcal{P}_l} (B_i + B'_i).$$

Hence, by the triangular inequality,

$$||A - A_l|| \leq \frac{1}{2} || \sum_{(i,k)\in\mathcal{P}_l} B_i|| + \frac{1}{2} || \sum_{(i,k)\in\mathcal{P}_l} B'_i||. \tag{44}$$

We bound each of the two terms in the RHS of (44) separately. First observe that if (i,k) and (i',k') are two distinct elements of $\mathcal{P}_l$ with $i < i'$, then

$$\begin{aligned} \mathbb{E}(B_i B_{i'}) &= \sum_{j=i+1}^{k-1} \sum_{j'=i'+1}^{k'-1} w_j w_{j'} \mathbb{E}((F_j - F_i)(F_{j'} - F_{i'})) \\ &= 0, \end{aligned}$$

where the second equation follows from (40). Thus

$$|| \sum_{(i,k)\in\mathcal{P}_l} B_i||^2 = \sum_{(i,k)\in\mathcal{P}_l} ||B_i||^2.$$

On the other hand, for $(i,k) \in \mathcal{P}_l$, by the triangular inequality,

$$\begin{aligned} ||B_i|| &\leq \sum_{j=i+1}^{k-1} |w_j| \, ||F_j - F_i|| \\ &\leq W'(i+1, k-1)\sqrt{||F_k||^2 - ||F_i||^2}, \end{aligned}$$

where the second equation follows from (42). Using Proposition D.2, it follows that

$$\begin{aligned} \sum_{(i,k)\in\mathcal{P}_l} ||B_i||^2 &\leq 2^{-2l} \sum_{(i,k)\in\mathcal{P}_l} (||F_k||^2 - ||F_i||^2) \\ &= 2^{-2l}(||F_m||^2 - ||F_0||^2) \\ &= 2^{-2l}\mathrm{Var}(F_m). \end{aligned}$$

We conclude that

$$|| \sum_{(i,k)\in\mathcal{P}_l} B_i|| \leq 2^{-l}\mathrm{Std}(F_m).$$

The same upper bound on $|| \sum_{(i,k)\in\mathcal{P}_l} B'_i||$ can be shown in a similar way. Hence

$$||A - A_l|| \leq 2^{-l}\mathrm{Std}(F_m).$$

This concludes the proof. $\square$

E Proof of Theorem 3.3

The proof is similar to that of Theorem 3.2. We apply the results in Section 2.2 with $Y_l = U_l$ for $0 \le l \le L$, and replace $\bar{Y}_l$ with $\bar{U}_l$ and $\bar{Y}$ with $\bar{U}$. As $U_L = U$, the analysis of Section 2.2 shows that $\mathbb{E}(\bar{U}) = \mathbb{E}(U) = \mathbb{E}(f(A)) - \alpha$. This implies (20). Let

$$\bar{m} := \frac{m}{\sum_{l=0}^{L} \sqrt{\mu_l |J_l|}}.$$

Since $n_l \ge \bar{m}\sqrt{\mu_l/|J_l|}$ for $0 \le l \le L$, it follows from (1) that

$$\begin{aligned} \mathrm{Var}(\bar{U}) &\le \bar{m}^{-1}(\sum_{l=0}^{L} \sqrt{\mu_l |J_l|}) \\ &= \frac{(\sum_{l=0}^{L} \sqrt{\mu_l |J_l|})^2}{m}. \end{aligned} \tag{45}$$

As $\mu_0 = \mathrm{Var}(U_0)$, by (17), we have $\mu_0 \le \kappa^2 \mathrm{Var}(F_m)$. By arguments similar to those leading to (37), for $l \ge 1$,

$$||U_l - U_{l-1}||^2 \le 2(||U_l - U||^2 + ||U_{l-1} - U||^2).$$

Since $||U_{l-1} - U||^2 \le 4\kappa^2 2^{-2l} \mathrm{Var}(F_m)$ by (18),

$$||U_l - U_{l-1}||^2 \le 10\kappa^2 2^{-2l} \mathrm{Var}(F_m).$$

We conclude that $\mu_l \le 10\kappa^2 2^{-2l} \mathrm{Var}(F_m)$ for $0 \le l \le L$. Since $|J_l| \le 2^{l+1}$, it follows from (45) that

$$m\mathrm{Var}(\bar{U}) \le \frac{20\kappa^2 \mathrm{Var}(F_m)}{(1 - 2^{-1/2})^2},$$

which implies (21), as $20(1 - 2^{-1/2})^{-2} \approx 233.14$.

Denote by C_l the expectation of the time to simulate $U_l - U_{l-1}$, for $0 \le l \le L$, and let $\bar{C} := \sum_{l=0}^{L} n_l C_l$ be the expected cost of computing $\bar{U}$. As in the proof of Theorem 3.2, it can be shown that there is a constant c' independent of m such that $C_l \le c'|J_l|$ for $0 \le l \le L$. As $n_l \le 1 + \bar{m}\sqrt{\mu_l/|J_l|}$,

$$\bar{C} \le c' \sum_{l=0}^{L} |J_l| + c'\bar{m} \sum_{l=0}^{L} \sqrt{\mu_l |J_l|}.$$

Since $|J_l| \le 2^{l+1}$ for $l \ge 0$, it follows that $\bar{C} \le c' 2^{L+2} + c'm \le 9c'm$. □

F Proof of Proposition 3.2

By (13) and (26),

$$\hat{A}_l - A_l = \sum_{j \in J_l} w_j (\hat{F}_j^l - F_j) + \frac{1}{2} \sum_{(i,k) \in \mathcal{P}_l} W(i+1, k-1)((\hat{F}_i^l - F_i) + (\hat{F}_k^l - F_k)).$$

Hence

$$||\hat{A}_l - A_l|| \le \sum_{j \in J_l} w_j ||\hat{F}_j^l - F_j|| + \frac{1}{2} \sum_{(i,k) \in \mathcal{P}_l} W(i+1, k-1)(||\hat{F}_i^l - F_i|| + ||\hat{F}_k^l - F_k||).$$

As $||\hat{F}_j^l - F_j|| \le \sqrt{c_2 2^{-\beta l}}$ for $j \in J_l$ and

$$\sum_{j \in J_l} w_j + \sum_{(i,k) \in \mathcal{P}_l} W(i+1, k-1) = 1,$$

it follows that $||\hat{A}_l - A_l|| \le \sqrt{c_2 2^{-\beta l}}$. Together with (14) and (36), this shows that $||\hat{A}_l - A||^2 \le c_3 2^{-\beta l}$. Similarly, as $||A_0 - W(1,m)F_0||^2 \le \mathrm{Var}(F_m)$, we have $||\hat{A}_0 - W(1,m)F_0||^2 \le c_3$. □

G Proof of Theorem 3.4

The proof is similar to that of Theorem 3.2. By A2 and (10), the vector $(\hat{F}^{l-1}, \hat{F}^l)$ can be simulated in $O(2^l)$ expected time for $l \geq 1$. Hence, by (26), there is a constant c' independent of m such that, for $l \geq 0$, the expectation of the time to simulate $\hat{U}_l - \hat{U}_{l-1}$ is at most $c'2^l$. As $|\hat{U}_0| \leq \kappa|\hat{A}_0 - W(1,m)F_0|$, Proposition 3.2 implies that $||\hat{U}_0||^2 \leq c_3\kappa^2$, where c_3 is defined as in Proposition 3.2. Similarly, for $l \geq 0$, as $|\hat{U}_l - U| \leq \kappa|\hat{A}_l - A|$, Proposition 3.2 shows that $||\hat{U}_l - U||^2 \leq c_3\kappa^2 2^{-\beta l}$. The conditions of Proposition 2.1 are thus met for $Y = U$ and $Y_l = \hat{U}_l$ for $l \geq 0$, with $\nu = c_3\kappa^2$ and $c = c'$. Thus, $\hat{V} = Z$ is square-integrable with $\mathbb{E}(\hat{V}) = \mathbb{E}(U)$. This implies (27). By (4),

$$||\hat{V}||^2 \leq \frac{20c_3\kappa^2}{1 - 2^{-(\beta-1)/2}},$$

and so $\mathrm{Var}(\hat{V})$ is upper-bounded by a constant independent of m. By (5), the expectation of the time to simulate $\hat{V}$ is at most $c'/(1 - 2^{-(\beta-1)/2})$. This completes the proof. $\square$

H Proof of Theorem 3.5

By arguments similar to those used in the proof of Theorem 3.4, there is a constant c' independent of m and of ϵ such that the expected cost of computing $\hat{U}_l - \hat{U}_{l-1}$ is at most $c'2^l$ for $l \geq 0$. Also, $||\hat{U}_0||^2 \leq c_3\kappa^2$ and, for $l \geq 0$,

$$||\hat{U}_l - U||^2 \leq c_3\kappa^2 2^{-l}.$$

The conditions of Proposition 2.2 are thus met for $Y = U$ and $Y_l = \hat{U}_l$ for $l \geq 0$, with $\nu = c_3\kappa^2$ and $c = c'$. By (7), $\hat{V} = Z_I$ is square-integrable and $(\mathbb{E}(\hat{V} - U))^2 \leq c_3\kappa^2\epsilon^2$. This implies (28). Similarly, (8) implies that

$$\mathrm{Var}(\hat{V}) \leq 48c_3\kappa^2 \log_2(1/\epsilon).$$

Furthermore, the expectation of the time required to simulate $\hat{V}$ is at most $4c' \log_2(1/\epsilon)$. $\square$

I Simulation of Square-Root diffusions

Proposition I.1 shows how to sample $F(t)$, for $t \in [0, T]$. Proposition I.1 and its proof are inspired from the analysis of the Cox-Ingersoll-Ross process in (Glasserman 2004, Section 3.4.1).

Proposition I.1. *Let N be a Poisson random variable with mean $2F_0/(\sigma^2 t)$. For integer $k \geq 1$, let χ_k^2 be a Chi-Square random variable with k degrees of freedom independent of N, and let $\chi_0^2 = 0$. Then $F(t)$ has the same distribution as $(\sigma^2 t/4)\chi_{2N}^2$. Furthermore, $F(t)$ is square-integrable.*

Proof. For $t \in [0, T]$, let $X(t) = 4F(t)/\sigma^2$, and let $x = X(0)$. Then

$$X(t) = x + 2\int_0^t \sqrt{X(s)}\, dW(s). \tag{46}$$

Hence X is a squared Bessel process of dimension 0. Such a process is a martingale (Jeanblanc, Yor and Chesney 2009, p. 339), and so $\int_0^t X(s)\, ds$ has finite expectation. By (46) and the isometry of stochastic integrals (Jeanblanc, Yor and Chesney 2009, Section 1.5.1), it follows that $X(t)$ is square-integrable. By (Jeanblanc, Yor and Chesney 2009, p. 344), for $t > 0$, we have $\Pr(X(t) = 0) = e^{-x/(2t)}$ and $X(t)$ has density

$$q_t(x, y) = \frac{1}{2t}\sqrt{\frac{x}{y}} \exp(-\frac{x+y}{2t}) I_1(\frac{\sqrt{xy}}{t})$$

at $y > 0$, where I_1 is the modified Bessel function with index 1 defined for $z > 0$ by

$$I_1(z) = \sum_{k=0}^{\infty} \frac{(z/2)^{2k+1}}{k!(k+1)!}.$$

For $y > 0$ and $k \geq 1$,

$$\Pr(\chi^2_{2k} \geq y) = \frac{1}{2} \int_y^\infty e^{-z/2} \frac{(z/2)^{k-1}}{(k-1)!}\, dz.$$

Thus,

$$\Pr(t\chi^2_{2k} \geq y) = \frac{1}{2t} \int_y^\infty \exp(-\frac{z}{2t})(\frac{z}{2t})^{k-1} \frac{1}{(k-1)!}\, dz.$$

Since $\mathbb{E}(N) = x/(2t)$, we have

$$\Pr(N = k) = \exp(-\frac{x}{2t})(\frac{x}{2t})^k \frac{1}{k!},$$

and so

$$\begin{aligned}
\Pr(t\chi^2_{2N} \geq y) &= \sum_{k=1}^{\infty} \Pr(N = k) \Pr(t\chi^2_{2k} \geq y) \\
&= \frac{1}{2t} \int_y^\infty \exp(-\frac{x+z}{2t}) \sum_{k=1}^{\infty} (\frac{x}{2t})^k (\frac{z}{2t})^{k-1} \frac{1}{(k-1)!k!}\, dz \\
&= \int_y^\infty q_t(x, z)\, dz \\
&= \Pr(X(t) \geq y).
\end{aligned}$$

Thus, $X(t)$ has the same as $t\chi^2_{2N}$. This concludes the proof. □ □

Consider now a time grid G consisting of $n+1$ dates $0 = \tau_0 < \tau_1 < \cdots < \tau_n$. We can use Proposition I.1 to recursively sample $F(\tau_k)$ for $1 \leq k \leq n$, and thereby simulate the forward price process on G in $O(n)$ expected time. Algorithms that simulate in unit expected time Poisson and Chi-Square random variables are given in (Devroye 1986). In our experiments, though, we have used generators from the standard C++ library.

J Detailed implementation of algorithms

This section presents a detailed implementation of the RMLMC, MLMC and RMLMC-Milstein algorithms. It assumes that Algorithm M has been executed and uses the same notation as Section 3.

J.1 The exact simulation case

This section assume that A1 holds.

J.1.1 Generating $U_l - U_{l-1}$

The function GDU takes as input a non-negative integer l and generates an instance of $U_l - U_{l-1}$ in $O(2^l)$ expected time.

J.1.2 Algorithm RMLMC

The algorithm RMLMC generates in constant expected time a random variable V such that the price of the Asian option with payoff $f(A)$ at T is equal to $e^{-rT}(\mathbb{E}(V) + \alpha)$. The variance of V is upper-bounded by a constant independent of m.

Algorithm 1 The function GDU

function GDU(l)
  if $l > L$ **then**
    return 0
  end if
  Simulate (F_j), $j \in J_l$
  $A_l \leftarrow \sum_{j\in J_l} w_j F_j + \frac{1}{2}\sum_{(i,k)\in\mathcal{P}_l} W(i+1,k-1)(F_i+F_k)$
  if $l = 0$ **then**
    return $f(A_0) - f(W(1,m)F_0)$
  end if
  $A_{l-1} \leftarrow \sum_{j\in J_{l-1}} w_j F_j + \frac{1}{2}\sum_{(i,k)\in\mathcal{P}_{l-1}} W(i+1,k-1)(F_i+F_k)$
  return $f(A_l) - f(A_{l-1})$
end function

Algorithm 2 The algorithm RMLMC

Simulate a random variable N such that $\Pr(N = l) = p_l$ for $l \in \mathbb{N}$, where $p_l := (1-2^{-3/2})2^{-3l/2}$
return GDU(N)$/p_N$

J.1.3 Algorithm MLMC

The algorithm MLMC generates a random variable $\bar{U}$ such that the price of the Asian option with payoff $f(A)$ at T is equal to $e^{-rT}(\mathbb{E}(\bar{U}) + \alpha)$. The expected time to generate $\bar{U}$ is $O(m)$ and the variance of $\bar{U}$ is $O(1/m)$.

Algorithm 3 The algorithm MLMC

for $l \leftarrow 0, L$ **do**
  Estimate $\mu_l \leftarrow \mathrm{Var}(U_l - U_{l-1})$ via 10^3 independent runs of GDU(l)
end for
for $l \leftarrow 0, L$ **do**
  $n_l \leftarrow \left\lfloor 1 + \frac{30m\sqrt{\mu_l/|J_l|}}{\sum_{l'=0}^{L}\sqrt{\mu_{l'}|J_{l'}|}} \right\rfloor$
  Let $\bar{U}_l$ be the average of n_l independent copies of $U_l - U_{l-1}$, simulated by calling GDU(l)
end for
return $\bar{U} \leftarrow \sum_{l=0}^{L} \bar{U}_l$

J.2 The approximate simulation case

This section, based on the Milstein scheme, make the same assumptions as Section 3.3.3.

J.2.1 Generating $\hat{U}_l - \hat{U}_{l-1}$

The function GDU-Milstein takes as input a non-negative integer l and generates an instance of $\hat{U}_l - \hat{U}_{l-1}$ in $O(2^l)$ expected time.

J.2.2 Algorithm RMLMC-Milstein

The algorithm RMLMC-Milstein generates in constant expected time a random variable $\hat{V}$ such that the price of the Asian option with payoff $f(A)$ at T is equal to $e^{-rT}(\mathbb{E}(\hat{V}) + \alpha)$. The variance of $\hat{V}$ is upper-bounded by a constant independent of m.

Algorithm 4 The function GDU-Milstein

function GDU-MILSTEIN(l)

Let $0 = \tau_0 < \tau_1 < \cdots < \tau_n = T$ be the elements of the set

$$\{t_j : j \in J_l\} \cup \{i2^{-l}T : 0 \le i \le 2^l\}$$

sorted in increasing order

Simulate $(W(\tau_1), \ldots, W(\tau_n))$

$F_0^* \leftarrow F_0$

for $k \leftarrow 0, n-1$ **do**

$\Delta W \leftarrow W(\tau_{k+1}) - W(\tau_k)$

$F_{k+1}^* \leftarrow F_k^* + b(F_k^*, \tau_k)(\Delta W) + \frac{1}{2} b(F_k^*, \tau_k) \frac{\partial b}{\partial x}(F_k^*, \tau_k)((\Delta W)^2 - (\tau_{k+1} - \tau_k))$

end for

for $j \in J_l$ **do**

$\hat{F}_j^l \leftarrow F_k^*$, where k is the index such that $\tau_k = t_j$

end for

$\hat{A}_l \leftarrow \sum_{j \in J_l} w_j \hat{F}_j^l + \frac{1}{2} \sum_{(i,k) \in \mathcal{P}_l} W(i+1, k-1)(\hat{F}_i^l + \hat{F}_k^l)$

if $l = 0$ **then**

return $f(\hat{A}_0) - f(W(1, m)F_0)$

end if

Let $0 = \tau'_0 < \tau'_1 < \cdots < \tau'_{n'} = T$ be the elements of the set

$$\{t_j : j \in J_{l-1}\} \cup \{i2^{-l+1}T : 0 \le i \le 2^{l-1}\}$$

sorted in increasing order

$F'_0 \leftarrow F_0$

for $k \leftarrow 0, n'-1$ **do**

$\Delta W \leftarrow W(\tau'_{k+1}) - W(\tau'_k)$

$F'_{k+1} \leftarrow F'_k + b(F'_k, \tau'_k)(\Delta W) + \frac{1}{2} b(F'_k, \tau'_k) \frac{\partial b}{\partial x}(F'_k, \tau'_k)((\Delta W)^2 - (\tau'_{k+1} - \tau'_k))$

end for

for $j \in J_{l-1}$ **do**

$\hat{F}_j^{l-1} \leftarrow F'_k$, where k is the index such that $\tau'_k = t_j$

end for

$\hat{A}_{l-1} \leftarrow \sum_{j \in J_{l-1}} w_j \hat{F}_j^{l-1} + \frac{1}{2} \sum_{(i,k) \in \mathcal{P}_{l-1}} W(i+1, k-1)(\hat{F}_i^{l-1} + \hat{F}_k^{l-1})$

return $f(\hat{A}_l) - f(\hat{A}_{l-1})$

end function

Algorithm 5 The algorithm RMLMC-Milstein

Simulate a random variable N such that $\Pr(N = l) = p_l$ for $l \in \mathbb{N}$, where $p_l := (1 - 2^{-3/2}) 2^{-3l/2}$

return GDU-Milstein(N)$/p_N$

Table 7: Pricing average price calls in the Black-Scholes model with $m = 500$ and different values of n

Method	n	Price	Std	Cost	Cost $\times$ Std^2	Time	Eff$_{\text{cost}}$	Eff$_{\text{time}}$
RMLMC	6.25×10^6	0.3499	6.0×10^{-4}	1.34×10^7	4.8	2	45	31
	2.5×10^7	0.3509	3.0×10^{-4}	5.38×10^7	4.8	6	45	31
	1×10^8	0.3506	1.5×10^{-4}	2.15×10^8	4.8	25	45	31
	4×10^8	0.3506	7.5×10^{-5}	8.61×10^8	4.8	99	45	32
MLMC	1.25×10^3	0.3511	5.9×10^{-4}	1.36×10^7	4.7	1	46	34
	5×10^3	0.3507	3.0×10^{-4}	5.45×10^7	4.8	5	45	36
	2×10^4	0.3504	1.5×10^{-4}	2.18×10^8	4.9	22	44	35
	8×10^4	0.35066	7.5×10^{-5}	8.73×10^8	4.9	87	44	36
RMLMC-Mil.	6.25×10^6	0.3507	5.7×10^{-4}	2.09×10^7	6.8	2	32	23
	2.5×10^7	0.3508	2.8×10^{-4}	8.37×10^7	6.8	9	32	23
	1×10^8	0.3506	1.4×10^{-4}	3.35×10^8	6.8	36	32	23
	4×10^8	0.35061	7.1×10^{-5}	1.34×10^9	6.8	136	32	25

Table 8: Pricing average price calls in the Black-Scholes model via multilevel and control variate methods

m	Method	n	Price	Std	Cost	Cost $\times$ Std^2	Time	Eff$_{\text{cost}}$	Eff$_{\text{time}}$
125	CV	1×10^6	0.3522	2.2×10^{-4}	1.25×10^8	6.1	11	9	9
	RMLMC+CV	1×10^8	0.35228	6.1×10^{-5}	2.1×10^8	0.78	24	70	49
	MLMC+CV	8×10^4	0.35221	5.7×10^{-5}	2.27×10^8	0.72	22	75	62
	RMLMC-Mil.+CV	1×10^8	0.35219	6.5×10^{-5}	3.3×10^8	1.4	36	40	29
250	CV	1×10^6	0.3510	2.2×10^{-4}	2.5×10^8	12	21	9	9
	RMLMC+CV	1×10^8	0.35113	6.2×10^{-5}	2.13×10^8	0.81	25	136	96
	MLMC+CV	4×10^4	0.35115	5.8×10^{-5}	2.29×10^8	0.76	22	145	122
	RMLMC-Mil.+CV	1×10^8	0.35105	6.5×10^{-5}	3.33×10^8	1.4	36	78	59
500	CV	1×10^6	0.3505	2.2×10^{-4}	5×10^8	24	42	9	8
	RMLMC+CV	1×10^8	0.35059	6.2×10^{-5}	2.15×10^8	0.83	25	259	181
	MLMC+CV	2×10^4	0.35061	5.8×10^{-5}	2.36×10^8	0.8	23	269	222
	RMLMC-Mil.+CV	1×10^8	0.35055	6.5×10^{-5}	3.35×10^8	1.4	37	153	112

K Further numerical experiments and combination with a control variate method

We report here additional numerical experiments and show how to combine our approach with a control variate method. We use the Black-Scholes model in our numerical experiments. Sections K.1 and K.2 assume that $S_0 = K = 2$, $\sigma = 50\%$, $r = 5\%$ and $T = 2$.

K.1 Varying the number of simulations

Table 7 gives prices of average price calls for the RMLMC, MLMC and RMLMC-Milstein algorithms with different values of n. As expected, for each algorithm, the variable "Cost" (resp. "Std") is roughly proportional to n (resp. $n^{-1/2}$). Similarly, the variables Cost $\times$ Std^2, "Eff$_{\text{cost}}$" and "Eff$_{\text{time}}$" are roughly independent of n. The RMLMC and MLMC methods have a similar performance and outperform the RMLMC-Milstein algorithm.

K.2 Using A as control variate

Given a real number Δ, define the real-valued function f_Δ of one variable as follows: $f_\Delta(x) := f(x) - \Delta(x - W(1,m)F_0)$. Note that f_Δ is $(\kappa + \Delta)$-Lipschitz and, by the martingale property, that $\mathbb{E}(f_\Delta(A)) = \mathbb{E}(f(A))$. Thus, the Asian options with payoff $f(A)$ and $f_\Delta(A)$ at maturity T have the same price at time 0. Hence, under A1, a control variate method (CV) can price the Asian option with payoff $f(A)$ by discounting the average of $f_\Delta(A)$ over n independent simulated paths. We can combine the CV method with the pricing algorithms in this paper, under A1 or A2, by applying them to f_Δ rather than f. In order to minimize $\mathrm{Var}(f_\Delta(A))$, we set

$$\Delta = \frac{\mathrm{Cov}(f(A), A)}{\mathrm{Var}(A)}$$

Table 9: Efficiency of pricing algorithms for average price calls in the Black-Scholes model, with $S_0 = 2$, $\sigma = 50\%$, $r = 5\%$, $T = 2$, $m = 500$ and various values of K

Method	$K = 1.4$	$K = 1.6$	$K = 1.8$	$K = 2$	$K = 2.2$	$K = 2.4$	$K = 2.6$
RMLMC	51	49	47	45	43	41	39
MLMC	50	48	46	44	42	39	38
RMLMC-Mil.	35	34	33	32	31	30	29
CV	59	26	14	9	6	5	4
RMLMC+CV	1413	723	416	259	177	132	103
MLMC+CV	1776	811	431	269	184	132	104
RMLMC-Mil.+CV	695	387	236	153	108	83	67

Table 10: Efficiency of pricing algorithms for average price calls in the Black-Scholes model, with $S_0 = K = 2$, $r = 5\%$, $T = 2$, $m = 500$ and various values of σ

Method	$\sigma = 0.1$	$\sigma = 0.2$	$\sigma = 0.3$	$\sigma = 0.4$	$\sigma = 0.5$	$\sigma = 0.6$	$\sigma = 0.7$
RMLMC	54	51	49	47	45	43	40
MLMC	52	49	47	46	44	41	38
RMLMC-Mil.	37	35	34	33	32	30	27
CV	11	8	7	8	9	10	12
RMLMC+CV	314	226	222	236	259	290	331
MLMC+CV	315	231	227	246	269	308	350
RMLMC-Mil.+CV	213	155	150	152	153	151	146

in our numerical experiments (Glasserman 2004, Section 4.1.1). We estimate Δ by Monte Carlo simulation using 10^4 independent runs. Table 8 gives prices of average price calls using the CV method and its combination with the RMLMC, MLMC and RMLMC-Milstein methods. For each value of m, the CV method is about 9 times more efficient than standard Monte Carlo, according to the variables "Eff_{cost}" and "Eff_{time}". Moreover, a comparison between Tables 1 and 8 shows that, combining the CV method with the RMLMC, MLMC, and RMLMC-Milstein algorithms improves their efficiency by about a factor of 6, 6, and 5, respectively. In line with Theorems 3.2, 3.3 and 3.4, the efficiency of the RMLMC, MLMC and RMLMC-Milstein algorithms, combined with the CV method, is roughly proportional to m.

K.3 Varying the strike and the volatility

Table 9 (resp. 10) gives Eff_{cost} for the pricing algorithms considered previously applied to average price calls with various values of K (resp. σ). In accordance with previous experiments, we set $n = 10^8$ for the RMLMC and RMLMC-Milstein algorithms and their combination with the CV method, $n = 2 \times 10^4$ for the MLMC algorithm and its combination with the CV method, and $n = 10^6$ for the CV method. Table 9 shows that, as K increases, the efficiency of the RMLMC, MLMC and RMLMC-Milstein algorithms slightly decreases, while that of the CV method and its combination with RMLMC, MLMC and RMLMC-Milstein strongly decreases. It is well-known that the efficiency of a control variate method can vary widely with the parameters of a problem (Glasserman 2004, Section 4.1.1). In contrast, in Table 10, the efficiency of the pricing algorithms varies mildly with σ, with no clear tendency in the dependence on σ.

L Generalization to continuously monitored Asian options

Denote by $\mathcal{B}([0,T])$ the Borel σ-algebra on $[0,T]$. This section assumes that $(\Omega, \mathcal{F})$ is a measurable space and that $(\omega, t) \mapsto F(t)(\omega)$ is a measurable function on the product space $(\Omega, \mathcal{F}) \times ([0,T], \mathcal{B}([0,T]))$, where $F(t)(\omega) = F(t)$ is the forward price at time t. Let w be a measurable function on $[0,T]$ such that $|w(t)| \leq T^{-1}$ for $0 \leq t \leq T$, and let w^* be a real number with $|w^*| \leq 1/2$. This section studies a continuously monitored Asian option with payoff $f(A^c)$

at maturity T, where f is a κ-Lipschitz real-valued function of one variable and

$$A^c := w^*F(T) + \int_0^T w(t)F(t)\,dt.$$

Here again, we assume the existence of a risk-neutral probability Q such that the stochastic process (F_t), $0 \le t \le T$, is a Q-martingale, and the price of the option at time 0 is $e^{-rT}\mathbb{E}(f(A^c))$, where r is the risk-free rate at time 0 for maturity T. We also assume that $F(T)$ is square-integrable and that, for $l \ge 0$ and $0 \le i \le 2^l - 1$,

$$W^{(l)}(i) := \int_{i2^{-l}T}^{(i+1)2^{-l}T} w(t)\,dt$$

can be calculated in constant time. By the Cauchy-Schwartz inequality,

$$\begin{aligned}\left(\int_0^T w(t)F(t)\,dt\right)^2 &\le \left(\int_0^T w(t)^2\,dt\right)\left(\int_0^T F(t)^2\,dt\right)\\ &\le T^{-1}\int_0^T F(t)^2\,dt.\end{aligned}$$

A continuous-time version of (41) implies that $||F^2(t)||^2 \le ||F(T)||^2$ for $0 \le t \le T$. Thus

$$\begin{aligned}\mathbb{E}\left(\left(\int_0^T w(t)F(t)\,dt\right)^2\right) &\le T^{-1}\mathbb{E}\left(\left(\int_0^T F(t)^2\,dt\right)^2\right)\\ &= T^{-1}\int_0^T ||F(t)||^2\,dt\\ &\le ||F(T)||^2,\end{aligned}$$

where the second equation follows from Fubini's Theorem. Hence A^c is square-integrable. For $l \ge 0$, define the following trapezoidal approximation of A:

$$A_l^c := w^*F(T) + \frac{1}{2}\sum_{i=0}^{2^l-1} W^{(l)}(i)(F(i2^{-l}T) + F((i+1)2^{-l}T)).$$

The following is a continuous-time version of Theorem 3.1.

Theorem L.1. *$||A_0^c - (w^* + W^{(0)}(0))F_0||^2 \le \mathrm{Var}(F(T))$ and, for $l \ge 0$,*

$$||A_l^c - A^c||^2 \le 2^{-2l}\mathrm{Var}(F(T)).$$

Proof. The proof is similar to that of Theorem 3.1. By construction,

$$A_0^c = w^*F(T) + \frac{1}{2}W^{(0)}(0)(F(0) + F(T)).$$

Hence

$$A_0^c - (w^* + W^{(0)}(0))F_0 = (\frac{1}{2}W^{(0)}(0) + w^*)(F(T) - F_0),$$

and so $||A_0^c - (w^* + W^{(0)}(0))F_0|| \le ||F(T) - F_0||$. This implies the desired bound on $||A_0^c - (w^* + W^{(0)}(0))F_0||$.

Fix now $l \ge 0$ and set $\theta_i := i2^{-l}T$. For $0 \le i \le 2^l - 1$, set

$$B_i := \int_{\theta_i}^{\theta_{i+1}} w(t)(F(t) - F(\theta_i))\,dt \text{ and } B_i' := \int_{\theta_i}^{\theta_{i+1}} w(t)(F(t) - F(\theta_{i+1})\,dt.$$

Thus,

$$A^c - A_l^c = \frac{1}{2}\sum_{i=0}^{2^l-1}(B_i + B_i').$$

For $0 \le i \le 2^l - 1$, by the Cauchy-Schwartz inequality,

$$\begin{aligned} B_i^2 &\le \left(\int_{\theta_i}^{\theta_{i+1}} w(t)^2\, dt\right)\left(\int_{\theta_i}^{\theta_{i+1}} (F(t) - F(\theta_i))^2\, dt\right) \\ &\le 2^{-l}T^{-1}\int_{\theta_i}^{\theta_{i+1}} (F(t) - F(\theta_i))^2\, dt. \end{aligned}$$

The proof of (42) shows that, for $0 \le u \le s \le t \le T$,

$$||F(s) - F(u)||^2 \le ||F(t)||^2 - ||F(u)||^2.$$

Hence, by Fubini's theorem,

$$\begin{aligned} ||B_i||^2 &\le 2^{-l}T^{-1}\int_{\theta_i}^{\theta_{i+1}} ||F(t) - F(\theta_i)||^2\, dt. \\ &\le 2^{-2l}(||F(\theta_{i+1})||^2 - ||F(\theta_i)||^2). \end{aligned}$$

A similar calculation together with the inequality $2xy \le x^2 + y^2$ shows that the random variable

$$\int_{(t,t')\in[\theta_i,\theta_{i+1}]\times[\theta_{i'},\theta_{i'+1}]} |F(t) - F(\theta_i)|\,|F(t') - F(\theta_{i'})|\, dt\, dt'$$

has a finite expectation. On the other hand, the proof of (40) shows that, for $0 \le u \le s \le t \le T$,

$$\mathbb{E}(F(u)(F(t) - F(s))) = 0.$$

By Fubini's theorem, it follows that, for $0 \le i < i' \le 2^l - 1$,

$$\begin{aligned} \mathbb{E}(B_i B_{i'}) &= \int_{(t,t')\in[\theta_i,\theta_{i+1}]\times[\theta_{i'},\theta_{i'+1}]} w(t)w(t')\mathbb{E}((F(t) - F(\theta_i))(F(t') - F(\theta_{i'})))\, dt\, dt' \\ &= 0. \end{aligned}$$

The remainder of the proof is similar to that of Theorem 3.1. □ □

Theorem L.1 can be used to build estimators for the continuously monitored Asian option price under Assumptions A1 or A2. For instance, a proof similar to that of Theorem 3.2 implies the following.

Theorem L.2. *Suppose A1 holds. Let $N \in \mathbb{N}$ be an integral random variable independent of $(F_j : 1 \le j \le m)$ such that $\Pr(N = l) = p_l$ for non-negative integer l, where $p_l := (1 - 2^{-3/2})2^{-3l/2}$. Set $V^c := (f(A_N^c) - f(A_{N-1}^c))/p_N$, where $A_{-1}^c := (w^* + W^{(0)}(0))F_0$. Then V^c is square-integrable,*

$$\mathbb{E}(f(A)) = \mathbb{E}(V) + f(A_{-1}^c),$$

and

$$\mathrm{Var}(V^c) \le 70\kappa^2\mathrm{Var}(F(T)).$$

Furthermore, the expectation of the time required to simulate V^c is finite.

References

Alfonsi, A. (2015). *Affine diffusions and related processes: simulation, theory and applications*, Springer, Cham.

Andersen, L. and Andreasen, J. (2000). Jump diffusion models: volatility smile fitting and numerical methods for pricing, *Review of derivatives research* **4**: 231–262.

Bandi, C. and Bertsimas, D. (2014). Robust option pricing, *European Journal of Operational Research* **239**(3): 842–853.

Broadie, M. and Kaya, Ö. (2006). Exact simulation of stochastic volatility and other affine jump diffusion processes, *Operations Research* **54**(2): 217–231.

Cai, N., Li, C. and Shi, C. (2013). Closed-form expansions of discretely monitored Asian options in diffusion models, *Mathematics of Operations Research* **39**(3): 789–822.

Cai, N., Song, Y. and Chen, N. (2017). Exact simulation of the SABR model, *Operations Research* **65**(4): 931–951.

Cai, N., Song, Y. and Kou, S. (2015). A general framework for pricing Asian options under Markov processes, *Operations Research* **63**(3): 540–554.

Černỳ, A. and Kyriakou, I. (2011). An improved convolution algorithm for discretely sampled Asian options, *Quantitative Finance* **11**(3): 381–389.

Cont, R. and Tankov, P. (2004). *Financial modelling with jump processes*, CRC press, London.

Corsaro, S., Kyriakou, I., Marazzina, D. and Marino, Z. (2019). A general framework for pricing Asian options under stochastic volatility on parallel architectures, *European Journal of Operational Research* **272**(3): 1082–1095.

Cox, J. C. and Ross, S. A. (1976). The valuation of options for alternative stochastic processes, *Journal of financial economics* **3**(1-2): 145–166.

Cui, Z., Lee, C. and Liu, Y. (2018). Single-transform formulas for pricing Asian options in a general approximation framework under Markov processes, *European Journal of Operational Research* **266**(3): 1134–1139.

Devroye, L. (1986). *Non-Uniform Random Variate Generation*, Springer-Verlag, New-York.

Dingeç, K. D. and Hörmann, W. (2012). A general control variate method for option pricing under Lévy processes, *European Journal of Operational Research* **221**(2): 368–377.

Dingec, K. D., Sak, H. and Hörmann, W. (2014). Variance reduction for Asian options under a general model framework, *Review of Finance* **19**(2): 907–949.

Duan, J.-C. and Simonato, J.-G. (1998). Empirical martingale simulation for asset prices, *Management Science* **44**(9): 1218–1233.

Fusai, G. and Kyriakou, I. (2016). General optimized lower and upper bounds for discrete and continuous arithmetic Asian options, *Mathematics of Operations Research* **41**(2): 531–559.

Fusai, G. and Meucci, A. (2008). Pricing discretely monitored Asian options under Lévy processes, *Journal of Banking & Finance* **32**(10): 2076–2088.

Gambaro, A. M., Kyriakou, I. and Fusai, G. (2020). General lattice methods for arithmetic Asian options, *European Journal of Operational Research* **282**(3): 1185 – 1199.

Genin, A. and Tankov, P. (2018). Optimal importance sampling for Lévy processes, *Stochastic Processes and their Applications* . In press.

Giesecke, K. and Smelov, D. (2013). Exact sampling of jump diffusions, *Operations Research* **61**(4): 894–907.

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

Giles, M. B., Debrabant, K. and Rössler, A. (2019). Analysis of multilevel Monte Carlo path simulation using the Milstein discretisation, *Discrete & Continuous Dynamical Systems-B* pp. 381–401.

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

Giles, M. B. and Xia, Y. (2017). Multilevel Monte Carlo for exponential Lévy models, *Finance and Stochastics* **21**(4): 995–1026.

Glasserman, P. (2004). *Monte Carlo methods in financial engineering*, Springer, New York.

Glasserman, P., Heidelberger, P. and Shahabuddin, P. (1999). Asymptotically optimal importance sampling and stratification for pricing path-dependent options, *Mathematical finance* **9**(2): 117–152.

Glasserman, P. and Kim, K.-K. (2011). Gamma expansion of the Heston stochastic volatility model, *Finance and Stochastics* **15**(2): 267–296.

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

Hull, J. (2014). *Options, Futures and Other Derivatives*, 9th edn, Prentice-Hall, Upper Saddle River, NJ.

Jeanblanc, M., Yor, M. and Chesney, M. (2009). *Mathematical methods for financial markets*, Springer, London.

Kahalé, N. (2017). Superreplication of financial derivatives via convex programming, *Management Science* **63**(7): 2323–2339.

Kebaier, A. and Lelong, J. (2018). Coupling importance sampling and Multilevel Monte Carlo using sample average approximation, *Methodology and Computing in Applied Probability* **20**(2): 611–641.

Kemna, A. G. and Vorst, A. (1990). A pricing method for options based on average asset values, *Journal of Banking & Finance* **14**(1): 113–129.

Kloeden, P. E. and Platen, E. (1992). *Numerical solution of stochastic differential equations*, Springer-Verlag, New York.

Kou, S. G. (2002). A jump-diffusion model for option pricing, *Management Science* **48**(8): 1086–1101.

Li, C. and Wu, L. (2019). Exact simulation of the Ornstein–Uhlenbeck driven stochastic volatility model, *European Journal of Operational Research* **275**(2): 768–779.

Linetsky, V. (2004). Spectral expansions for Asian (average price) options, *Operations Research* **52**(6): 856–867.

Madan, D. B., Carr, P. P. and Chang, E. C. (1998). The variance gamma process and option pricing, *Review of Finance* **2**(1): 79–105.

Merton, R. C. (1976). Option pricing when underlying stock returns are discontinuous, *Journal of financial economics* **3**(1): 125–144.

Revuz, D. and Yor, M. (1999). *Continuous martingales and Brownian motion*, third edn, Springer-Verlag, Berlin.

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

Shiraya, K. and Takahashi, A. (2017). A general control variate method for multi-dimensional SDEs: An application to multi-asset options under local stochastic volatility with jumps models in finance, *European Journal of Operational Research* **258**(1): 358–371.

Shiraya, K. and Takahashi, A. (2018). Pricing average and spread options under local-stochastic volatility jump-diffusion models, *Mathematics of Operations Research* **44**(1): 303–333.

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

Wang, X. and Sloan, I. H. (2011). Quasi-Monte Carlo methods in financial engineering: An equivalence principle and dimension reduction, *Operations Research* **59**(1): 80–95.