ANZIAM J. 59(2017), 183–199
doi:10.1017/S1446181117000335

PRICING EUROPEAN OPTIONS ON REGIME-SWITCHING ASSETS: A COMPARATIVE STUDY OF MONTE CARLO AND FINITE-DIFFERENCE APPROACHES

X. C. ZENG[✉1], I. GUO[2] and S. P. ZHU[1]

(Received 4 May, 2016; accepted 26 April, 2017; first published online 23 October 2017)

Abstract

A numerical comparison of the Monte Carlo (MC) simulation and the finite-difference method for pricing European options under a regime-switching framework is presented in this paper. We consider pricing options on stocks having two to four volatility regimes. Numerical results show that the MC simulation outperforms the Crank–Nicolson (CN) finite-difference method in both the low-frequency case and the high-frequency case. Even though both methods have linear growth, as the number of regimes increases, the computational time of CN grows much faster than that of MC. In addition, for the two-state case, we propose a much faster simulation algorithm whose computational time is almost independent of the switching frequency. We also investigate the performances of two variance-reduction techniques: antithetic variates and control variates, to further improve the efficiency of the simulation.

2010 *Mathematics subject classification*: primary 65C05; secondary 65N06.

Keywords and phrases: option pricing, regime-switching model, finite-difference method, Monte Carlo simulation, variance-reduction technique, simulating total occupation time.

1. Introduction

Following the introduction of the Black–Scholes (BS) model [1] in 1973, analytical solutions for options pricing became a very popular research topic, since they were fast to calculate and easy to implement. However, cases with closed-form solutions are very rare and mostly happen in vanilla European options with path-independent payoff functions. Options without pricing formulas, including many exotic options as well as American options, have to be solved numerically. Much research has been

[1]School of Mathematics and Applied Statistics, University of Wollongong, NSW 2522, Australia; e-mail: xz379@uowmail.edu.au, spz@uow.edu.au.
[2]School of Mathematical Sciences, Clayton Campus, Monash University, VIC 3800, Australia; e-mail: ivan.guo@monash.edu.au.

done on comparing various numerical methods in the BS model for both European and American options (see, for instance, the books by Brandimarte [3] and Detemple [5]).

It is well known that the conventional BS model with a constant volatility fails to reflect the stochastic nature of financial markets. As a result, more realistic models which better reflect random market movements are needed. One of these extended models is the *regime-switching model*, in which the volatility term is driven by a hidden Markov chain and switches between a finite number of states. Since its introduction by Hamilton [11], growing evidence has suggested that this Markovian switching model can capture the time-series properties of several important financial variables. Analytical solutions have been discovered for the model with two states. First Guo [10] presented a closed-form formula with double integrals for European options under a two-state regime-switching framework. Fuh et al. [8] found an error in Guo's formula and presented a new formula in a very similar form. Later Zhu et al. [17] presented another closed-form solution which contains only a single integral by finding the Fourier inversion analytically. However, even though the pricing problem for European options in a two-state regime-switching model can be solved analytically, closed-form solutions for the models with more than two states have not been found yet. Numerical techniques thus continue to play an important role in pricing options with multi-state regime-switching models.

Research on the numerical methods for regime-switching models has been discussed in [2, 12–14, 16]. Here we focus on two basic and important methods, the Monte Carlo (MC) simulation and the finite-difference method, the performances of which have not yet been compared in the literature. Our paper provides such a comparison between these two methods, so that future researchers can better choose the more efficient one for pricing European-style options especially in the multi-state regime-switching model. We present a modified version of the algorithm of Lemieux [13], which is referred to as the fundamental MC simulation, and then investigate two variance-reduction techniques, antithetic variates and control variates. We also propose a new and much faster algorithm, which simulates the total occupation time instead of the trajectories for the two-state case. For the finite-difference method, we adopt the Crank–Nicolson (CN) scheme because of its unconditional stability and second-order convergence rate. Numerical analysis is given under the regime-switching framework with the number of regimes being up to four.

The rest of the paper is organized as follows. In Section 2, the model settings and notations are introduced. The MC simulation for regime-switching models is discussed in Section 3. The finite-difference method with CN schemes is presented in Section 4. Numerical results and comparisons are given in Section 5. Some concluding remarks are given in Section 6 and the Appendix comprises all the algorithms presented in this paper.

2. Model settings and notations

We first start with the introduction of regime-switching models. Let S_t be the price of an underlying asset in the market at time t. We further let the market have

https://doi.org/10.1017/S1446181117000335 Published online by Cambridge University Press

a nonconstant drift rate μ_t and a nonconstant volatility σ_t. Then in a regime-switching world where drift rates and volatilities are allowed to shift between different economic regimes, the fluctuation of an asset is assumed to follow the stochastic differential equation

$$dS_t = \mu(X_t)S_t\,dt + \sigma(X_t)S_t\,dW_t,$$

where X_t is a continuous-time Markov chain with K states and is independent of the standard Brownian motion W_t. Both the Markov chain and the standard Brownian motion are based on the probability triplet $\{\Omega, \mathcal{F}, \mathbb{P}\}$, where Ω is the set of all possible outcomes, $\mathcal{F}$ is the set of events, and $\mathbb{P}$ is the physical measure. For each state, the drift rate and the volatility are assumed to be constant and distinct, denoted by

$$\mu(X_t) = \begin{cases} \mu_1, & X_t = 1, \\ \mu_2, & X_t = 2, \\ \dots & \\ \mu_K, & X_t = K, \end{cases} \qquad \sigma(X_t) = \begin{cases} \sigma_1, & X_t = 1, \\ \sigma_2, & X_t = 2, \\ \dots & \\ \sigma_K, & X_t = K. \end{cases}$$

The generator of the Markov chain is

$$Q = \begin{pmatrix} \lambda_{11} & \lambda_{12} & \dots & \lambda_{1K} \\ \lambda_{21} & \lambda_{22} & \dots & \lambda_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_{K1} & \lambda_{K2} & \dots & \lambda_{KK} \end{pmatrix}$$

and, for each state, elements of the generator satisfy the equation $\lambda_{jj} + \sum_{i=1\, i\neq j}^{K} \lambda_{ji} = 0$.

Since another risk source X_t is introduced, the market becomes incomplete. As a result, there is no unique martingale measure. In this paper, we select the martingale measure presented by Elliott et al. [6] and assume that the interest rates are the same for all regimes. Thus, r is used to replace all μ_j under the risk-neutral measure for the rest of the paper. Under the risk-neutral measure, the price of a financial derivative with initial state j is obtained by

$$V_j = e^{-r(T-t)}\mathbb{E}(p(S_T)|\mathcal{F}_t, X_t = j), \tag{2.1}$$

where the function $p(\cdot)$ is the payoff function. For European options, substituting its payoff function into (2.1) and applying Itô's formula, the governing system of partial differential equations (PDEs) is given by (see the article by Buffington and Elliott [4])

$$\frac{\partial V_j}{\partial t} + \frac{1}{2}\sigma_j^2 S^2 \frac{\partial^2 V_j}{\partial S^2} + rS\frac{\partial V_j}{\partial S} - rV_j = \sum_{i=1\, i\neq j}^{K} \lambda_{ji}(V_j - V_i), \quad j = 1, 2, \dots, K. \tag{2.2}$$

Since the introduction of the regimes, (2.2) is a PDE system with a total of K equations which have to be solved simultaneously.

https://doi.org/10.1017/S1446181117000335 Published online by Cambridge University Press

3. Monte Carlo simulation

The MC simulations for regime-switching models can be found in [12, 13]. Lemieux [13] presented an algorithm to price vanilla European options in a K-state regime-switching model; however, this algorithm requires the estimation of the total amount of random numbers at the beginning. Since the number of exponential random numbers in each path is uncertain, such an estimation will lead to redundant random numbers, wasting some computational resources. Here we modify Lemieux's algorithm to get rid of the estimation. Hieber and Scherer [12] presented an efficient simulation method which was coupled with a few variance reductions for pricing barrier options in a two-state regime-switching model. The idea of their Brownian bridge approach is basically a special case of the simulated trajectories in Lemieux [13]. In this section, we present the modified version of Lemieux's algorithm, which this paper denotes as the fundamental MC, and two variance-reduction techniques: the antithetic variates and the control variates. In addition, we also propose a new algorithm to simulate the total occupation time, instead of the trajectory for the two-state case. The idea is based on the path-independent property of the vanilla European options, and we name it "simulating total occupation time".

3.1. Fundamental Monte Carlo simulation The theoretical framework for simulating trajectories is based on the fact that the holding time is exponentially distributed. Further, the probability of state j switching into state i can be estimated by $-\lambda_{ji}/\lambda_{jj}$, given the hypothesis that a switch has taken place. Our modified pricing of a vanilla European put option under a general K-state regime-switching model is presented in Algorithm 1. Simulations for European call options can be obtained by a simple change of payoff function.

Based on the fundamental MC, some variance-reduction techniques can be adopted to further improve the efficiency of the simulation, such as antithetic variates and control variates. In the next two subsections, we will introduce the two techniques; analysis of their numerical performances can be found in Section 5.

3.2. Antithetic variates Since the final stock prices used in the simulations are obtained using normal random variables, one way to improve the results is to reduce the variance of the normal random variables. The main idea of the antithetic variates [9] is that instead of estimating the expectation μ by averaging over N independent, identically distributed (iid) random variables, averaging over $N/2$ pairs of iid random variables which are denoted as Z_{ori}, and their negatives which are denoted as Z_{ant}. One natural way is to replace Step 6 in Algorithm 1 with the pair W and $-W$, since the standard normal distribution is symmetric around the y-axis. Thus, lines 37 to 40 are replaced by the steps as shown in Algorithm 2. The efficiency of the antithetic variates strongly depends on the negative correlation between $\max(E - Z_{\text{ori}}, 0)$ and $\max(E - Z_{\text{ant}}, 0)$.

https://doi.org/10.1017/S1446181117000335 Published online by Cambridge University Press

3.3. Control variates The control variates [9] method shares a common feature with the antithetic variates, which uses correlations to reduce variance. In this method, we need to find a variable C with a known mean which is related to the simulated model and is also correlated with the variable Y. In an options pricing problem, the underlying assets, that is, normally stocks, provide a universal set of control variates. This is because under the risk-neutral measure, the stock price at time t, is e^{rt} times the stock price today. To describe it in detail, given the risk-neutral measure with interest rate r (the same risk-neutral measure as we mentioned in Section 1, suppose that $S(t)$ is an asset price; then $e^{-rt}S(t)$ is a martingale. Suppose that we are pricing a contingent claim on S with the discounted payoff Y, which is a function of $S(t)$. From the independent implications of S_n, $n = 1, 2, \ldots, N$, we thus can form the estimator of control variates

$$\frac{1}{N}\sum_{n=1}^{N}[Y_n - b\{S_i(T) - e^{rT}S(0)\}],$$

where b is the coefficient which minimizes the variance of the estimator. To adopt the control variates, several steps must be added in between line 39 and line 40 of Algorithm 1. This is shown in Algorithm 3. The variance of the control variates estimator is $1 - \rho_{YZ}^2$. Stronger correlation between Y and Z means higher efficiency of the simulation with the control variates technique.

3.4. Simulating total occupation time Due to the Markovian property of the regime-switching model, the iterations of simulating trajectories can be avoided. This can be accomplished if we can simulate the total occupation time within each regime. As a result, a large amount of computational time can be saved. Theoretically, this can be achieved because the total occupation time is a random variable which has its own probability density function (PDF). We find that this is directly applicable to the two-state case since the analytical formula of the PDF of the total occupation time is available [8] for the two-state case. The PDF is shown in the following theorem.

THEOREM 3.1. *Assume that $T_{i|j}$ is the total occupation time that the Markov chain $X(t)$ spends in state i, given that the initial state is j, where $i, j \in \{1, 2\}$. Let $f_{i|j}$ be the* PDF *of $T_{i|j}$. Under the two-state Markov model, we have the following formulae:*

$$\begin{aligned}
f_{1|1}(t,T) &= e^{-\lambda_{12}T}\delta(T-t) + e^{-\lambda_{21}(T-t)-\lambda_{12}t}\Big[\lambda_{12}I_0(2(\lambda_{12}\lambda_{21}t(T-t))^{1/2}) \\
&\quad + \left(\frac{\lambda_{12}\lambda_{21}t}{T-t}\right)^{1/2} I_1(2(\lambda_{12}\lambda_{21}t(T-t))^{1/2})\Big], \\
f_{2|2}(t,T) &= e^{-\lambda_{21}T}\delta(T-t) + e^{-\lambda_{12}(T-t)-\lambda_{21}t}\Big[\lambda_{21}I_0(2(\lambda_{12}\lambda_{21}t(T-t))^{1/2}) \\
&\quad + \left(\frac{\lambda_{12}\lambda_{21}t}{T-t}\right)^{1/2} I_1(2(\lambda_{12}\lambda_{21}t(T-t))^{1/2})\Big], \\
f_{2|1}(t,T) &= f_{1|1}(T-t,T), \quad f_{1|2}(t,T) = f_{2|2}(T-t,T),
\end{aligned}$$

https://doi.org/10.1017/S1446181117000335 Published online by Cambridge University Press

where δ is a Dirac delta function and I_0, I_1 are modified Bessel functions such that

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

Proof of this theorem can be found in Fuh et al. [8].

Now we have obtained the PDF of the total occupation time and, further, we can solve for the cumulative probability distribution function (CDF) by numerical integration. Note that the calculation is quite straightforward everywhere apart from at $t = T$, which is a singular point and also contains a Dirac delta function. This can be interpreted as the total occupation time reaching a point mass at $t = T$. In other words, the Markov chain has remained in the same state throughout $[0, T]$. The probability of this occurring is given by $P(T_{i|i} = T) = e^{-\lambda_{ij}T}$, where $j = 3 - i$, which corresponds to a discontinuity in the CDF at $t = T$. In terms of the numerical integration, we first proceed as usual without the Dirac delta function and then add a jump of size $e^{-\lambda_{ij}T}$ to the CDF at $t = T$. A detailed algorithm of how to generate random numbers from the above PDF's is given in Algorithm 4.

Once having the simulation of the total occupation time, results for each path can be obtained by simple substitution, as we have shown in Algorithm 5. Note that since this is in the two-state case, the generator matrix contains four elements $\lambda_{11}, \lambda_{12}, \lambda_{21}, \lambda_{22}$. When we use λ_{ji} to represent them, a natural relation $i = 3 - j$ holds.

The core of Algorithm 5 is to generate the random variable following the PDF of the total occupation time. This algorithm is cheaper in computation than the fundamental MC because only two random numbers are generated in the algorithm, while a total of $n + 1$ random numbers must be generated in the fundamental MC, with n being the number of state changes. This can also explain why the computational time of the fundamental MC is a monotonically increasing function of the switching intensity and the time to expiry, while the computational time of this algorithm remains a constant and depends only on the number of paths.

In summary, the core of Algorithm 5 has two parts: one is to derive the PDF of the total occupation time and the other is to draw samples from this random variable. In theory, this algorithm can be extended to a general case with multi-state economy. Although the analytical formula of the PDF is not available in the multi-state case, a Fourier cosine expansion [7] can be applied to obtain a numerical distribution function. Drawing samples from this multi-variate random number can be done by building a very complex algorithm based on the acceptance–rejection method [9]. However, the multi-state extension is no longer fast and simple to implement, which was a crucial advantage of the total occupation time approach. Thus, this case will not be further explored in this paper.

4. Finite-difference method

Mielkie [14, 15] presented a detailed discussion of the CN finite-difference method to solve the coupled partial differential equations arising from a two-state regime-switching model. In contrast to the conventional BS model, the governing PDE problem

https://doi.org/10.1017/S1446181117000335 Published online by Cambridge University Press

of a regime-switching model contains a total of K coupled PDE's, each corresponding to one of the K different regimes. The coupled PDE's are solved simultaneously. In this section, we present the CN finite-difference method for a multi-state regime-switching model and show why the computation time of the finite-difference method grows linearly as the number of regimes increases.

We first introduce the transformation $\tau = T - t$ and $x = \log(S/E)$. Then, from (2.2),

$$\frac{\partial V_j}{\partial \tau} = \frac{1}{2}\sigma_j^2 \frac{\partial^2 V_j}{\partial x^2} + \left(r - \frac{1}{2}\sigma_j^2\right)\frac{\partial V_j}{\partial x} - rV_j - \sum_{\substack{i=1 \\ i\neq j}}^{K} \lambda_{ji}(V_j - V_i), \quad j = 1, 2, \ldots, K.$$

Here we discretize the region $[x_{\min}, x_{\max}] \times [0, T]$ into $(M+1)\times(N+1)$ grids, with $\Delta x = (x_{\max} - x_{\min})/M$ and $\Delta\tau = T/N$. We denote $V^n_{m,j}$ by $V_j(x_{\min} + m\Delta x, \Delta\tau)$. By applying the CN scheme,

$$\alpha_j V^{n+1}_{m-1,j} + \beta_j V^{n+1}_{m,j} + \gamma_j V^{n+1}_{m+1,j} + \sum_{i=1,\, i\neq j}^{K} \frac{\lambda_{ji}\Delta\tau}{2} V^{n+1}_{m,i} = f^n_{m,j}, \quad \text{where} \tag{4.1}$$

$$\alpha_j = -\left(\frac{\sigma_j^2\Delta\tau}{4\Delta x^2} - \frac{(r - \sigma_j^2/2)\Delta\tau}{4\Delta x}\right), \quad \beta_j = 1 + \frac{\sigma_j^2\Delta\tau}{2\Delta x^2} + \frac{r\Delta\tau}{2} - \frac{\lambda_{jj}\Delta\tau}{2},$$

$$\gamma_j = -\left(\frac{\sigma_j^2\Delta\tau}{4\Delta x^2} + \frac{(r - \sigma_j^2/2)\Delta\tau}{4\Delta x}\right),$$

$$f^n_{m,j} = -\alpha_j V^n_{m-1,j} + (2 - \beta_j)V^n_{m,j} - \gamma_j V^n_{m+1,j} - \sum_{\substack{i=1 \\ i\neq j}}^{K} \frac{\lambda_{ji}\Delta\tau}{2} V^n_{m,i}.$$

Writing (4.1) in matrix form,

$$\mathbf{P}_j \mathbf{V}^{n+1}_j + \sum_{i=1\, i\neq j}^{K} \mathbf{\Lambda}_{ji} \mathbf{V}^{n+1}_i = \mathbf{f}^n_j, \quad \text{where}$$

$$\mathbf{P}_j = \begin{pmatrix} \beta_j & \gamma_j & & & \\ \alpha_j & \beta_j & \gamma_j & & \\ & \ddots & \ddots & \ddots & \\ & & \alpha_j & \beta j & \gamma_j \\ & & & \alpha_j & \beta j \end{pmatrix}, \quad \mathbf{\Lambda}_{ji} = \begin{pmatrix} \lambda_{ji}\Delta\tau/2 & & & \\ & \lambda_{ji}\Delta\tau/2 & & \\ & & \ddots & \\ & & & \lambda_{ji}\Delta\tau/2 \end{pmatrix},$$

$$\mathbf{V}^n_j = \begin{pmatrix} V^n_{1,j} \\ V^n_{2,j} \\ \vdots \\ V^n_{N-2,j} \\ V^n_{N-1,j} \end{pmatrix}, \quad \mathbf{V}^n_i = \begin{pmatrix} V^n_{1,i} \\ V^n_{2,i} \\ \vdots \\ V^n_{N-2,i} \\ V^n_{N-1,i} \end{pmatrix}, \quad \mathbf{f}^n_j = \begin{pmatrix} f^n_{1,j} - \alpha_j V^{n+1}_{0,j} \\ f^n_{2,j} \\ \vdots \\ f^n_{N-2,j} \\ f^n_{N-1,j} - \gamma_j V^{n+1}_{N,j} \end{pmatrix}.$$

https://doi.org/10.1017/S1446181117000335 Published online by Cambridge University Press

Then the whole system at time $\tau = n\Delta\tau$ can be written as

$$\begin{cases} \mathbf{P}_1\mathbf{V}_1^n + \mathbf{\Lambda}_{12}\mathbf{V}_2^n + \mathbf{\Lambda}_{13}\mathbf{V}_3^n + \cdots + \mathbf{\Lambda}_{1K}\mathbf{V}_K^n = \mathbf{f}_1^n, \\ \mathbf{P}_2\mathbf{V}_2^n + \mathbf{\Lambda}_{21}\mathbf{V}_1^n + \mathbf{\Lambda}_{23}\mathbf{V}_3^n + \cdots + \mathbf{\Lambda}_{2K}\mathbf{V}_K^n = \mathbf{f}_2^n, \\ \vdots \\ \mathbf{P}_K\mathbf{V}_K^n + \mathbf{\Lambda}_{K1}\mathbf{V}_1^n + \mathbf{\Lambda}_{K2}\mathbf{V}_3^n + \cdots + \mathbf{\Lambda}_{KK-1}\mathbf{V}_{K-1}^n = \mathbf{f}_K^n. \end{cases} \tag{4.2}$$

The system (4.2) needs to be solved at each time step. To further clarify, we write this system in matrix form, assuming that

$$\mathbb{A} = \begin{pmatrix} \mathbf{P}_1 & \mathbf{\Lambda}_{12} & \dots & \mathbf{\Lambda}_{1K} \\ \mathbf{\Lambda}_{21} & \mathbf{P}_2 & \dots & \mathbf{\Lambda}_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{\Lambda}_{K1} & \mathbf{\Lambda}_{K2} & \dots & \mathbf{P}_K \end{pmatrix}, \quad \mathbb{V}^n = \begin{pmatrix} \mathbf{V}_1^n \\ \mathbf{V}_2^n \\ \vdots \\ \mathbf{V}_K^n \end{pmatrix}, \quad \mathbf{F}^n = \begin{pmatrix} \mathbf{f}_1^n \\ \mathbf{f}_2^n \\ \vdots \\ \mathbf{f}_K^n \end{pmatrix}.$$

Then we need only to solve the problem $\mathbb{A}\mathbb{V}^n = \mathbf{F}^n$ at each time step, and the option price can be obtained by interpolation when finishing the last step. It is worthwhile pointing out that the block matrix $\mathbb{A}$ is a sparse matrix, since $\mathbf{P}_j$ is a diagonal matrix and $\mathbf{\Lambda}_{ij}$ is an identity matrix. Thus, an iterative method can be adopted to solve the linear system. In addition, it is obvious that the size of the matrix $\mathbb{A}$ increases linearly as the number of regimes increases. Hence, the whole method grows rapidly according to the number of regimes, given the fact that the computational complexity of matrix inversion is at maximum $O((\mathrm{MNK})^3)$ in this problem. As a result, the CN tends to be far more expensive than the MC, when the two methods have a similar level of computational error in the cases where $K \geq 2$. We present the comparison in the next section.

5. Numerical performances and comparisons

In this section, we compare the MC and the CN methods. In addition, we test the fundamental MC against the two variance-reduction techniques and simulating the total occupation time. We present the numerical performances of each method in the models with two, three, and four regimes. The analytical solution by Zhu et al. [17] is adopted as the benchmark results for the two-state regime-switching model, while the trinomial tree method from Yuen and Yang [16] is applied for the multi-state cases.

5.1. MC vs CN We start with the comparison between the fundamental MC and CN. Since results from the simulations are random numbers instead of fixed values, we run the same simulations many times to obtain their 95% confidence intervals. The confidence intervals thus are considered as the accuracies of the simulation methods.

Our comparisons focus on regime-switching models with two, three, and four states, respectively. Two scenarios are considered for each model, a low-frequency scenario

https://doi.org/10.1017/S1446181117000335 Published online by Cambridge University Press

TABLE 1. MC vs CN in the low-frequency case. The parameters are $S_0 = 36$, $E = 40$, $T = 1$, $r = 0.1$. The volatilities are (0.15, 0.25), (0.15, 0.25, 0.35), (0.15, 0.25, 0.35, 0.45) for the two regimes, three regimes, four regimes, respectively. The jump intensities for all different models are $\lambda_{ji} = 1$, $j \neq i$.

	MC	CN	Benchmark
V_1^2	2.7022 ± 0.0015	2.7037 (0.0014)	2.7023
V_2^2	3.3203 ± 0.0017	3.3229 (0.0026)	3.3203
Time (sec)	12.4822	54.6678	
V_1^3	3.3562 ± 0.0016	3.3588 (0.0022)	3.3566
V_2^3	3.7653 ± 0.0018	3.7682 (0.0028)	3.7654
V_3^3	4.2508 ± 0.0020	4.2547 (0.0036)	4.2511
Time (sec)	19.0468	426.5641	
V_1^4	4.1021 ± 0.0023	4.106 (0.0033)	4.1032
V_2^4	4.3789 ± 0.0024	4.3834 (0.0037)	4.3797
V_3^4	4.7263 ± 0.0025	4.7316 (0.0043)	4.7273
V_4^4	5.1248 ± 0.0028	5.1306 (0.0049)	5.1257
Time (sec)	28.4594	938.2955	

and a high-frequency scenario, which aim at comparing the two methods with different parameters of the switching intensity, given the fact that the fundamental MC becomes more expensive as the switching intensity increases. To make a fair comparison, we manage to run both methods which give similar levels of error and then compare their computation times. As shown in Tables 1 and 2, results of MC simulations are shown with an average (AVE) over multiple runs, and the corresponding 95% confidence intervals (CON) are given in the form of AVE ± CON. The values in the parentheses in the column of CN are the difference between results from the finite-difference method and the benchmark. To obtain the benchmark results for the two-state regime-switching model, we apply the closed-form solution by Zhu et al. [17], while, for the three-state and four-state models, we use the trinomial tree method from Yuen and Yang [16] with 100 000 time steps. All the CN cases are run using 100 time steps and 2500 space steps, while each MC is run with 500 000 paths.

According to the two tables, each confidence interval of the MC is smaller than the error of the CN (except V_1^2 in Table 1, but the two values are rather close), while the CN spends several more times in computational time than the MC. Hence, the MC obviously outperforms the CN in both cases. The difference of the computation time becomes more significant as the number of the states increases. In addition, on one hand, in the high-frequency case, the computational time of the MC increases compared to the one in the low-frequency case. This can be explained as follows;

https://doi.org/10.1017/S1446181117000335 Published online by Cambridge University Press

TABLE 2. MC vs CN in the high-frequency case. All parameters are identical to the low-frequency case but the jump intensities now are $\lambda_{ji} = 100$, $j \neq i$.

	MC	CN	Benchmark
V_1^2	3.0577 ± 0.0016	3.0586 (0.0017)	3.0569
V_2^2	3.0646 ± 0.0017	3.0656 (0.0017)	3.0639
Time (sec)	12.4371	55.6734	
V_1^3	3.8696 ± 0.0024	3.8715 (0.0029)	3.8686
V_2^3	3.8733 ± 0.0024	3.8751 (0.0029)	3.8722
V_3^3	3.8787 ± 0.0024	3.8806 (0.0030)	3.8776
Time (sec)	31.6948	431.6442	
V_1^4	4.6810 ± 0.0023	4.6867 (0.0042)	4.6825
V_2^4	4.6835 ± 0.0023	4.6889 (0.0042)	4.6847
V_3^4	4.6902 ± 0.0023	4.6922 (0.0042)	4.6880
V_4^4	4.6945 ± 0.0024	4.6967 (0.0042)	4.6925
Time (sec)	109.1688	942.4624	

more iterations are involved in the simulating trajectories, as the average switching times are changed from once per year to 100 times per year. On the other hand, the computational time of the CN is independent of the jump intensity, because the computation cost, which is mainly solving linear algebraic systems, remains the same with the change of λ. Although MC is still much cheaper in the high-frequency case analysed here, it is worth mentioning that for sufficiently large λ, there will be a subset of parameter space in which CN will outperform MC. This is because the computation time of MC is increasing in λ, while the computation time of CN is not. However, at least with the other parameters selected in this study, the value of λ at which CN becomes competitive, with MC being too large to be realistic, is, for instance, $\lambda = 300$.

5.2. Comparison among simulations Since the simulation is cheaper than the finite-difference method, we would like to explore the performance of the simulations coupled with variance-reduction methods, which can be more efficient. For the two-state case, we also include the "simulating total occupation time" in the comparison. Before starting, we introduce an indicator to quantify the efficiency of MC from Lemieux [13] to simplify the comparison.

DEFINITION 5.1. The efficiency of an estimator $\hat{\mu}$ for a quantity μ is measured by the indicator

$$\text{Eff}(\hat{\mu}) = [\text{MSE}(\hat{\mu}) \times C(\hat{\mu})]^{-1},$$

https://doi.org/10.1017/S1446181117000335 Published online by Cambridge University Press

Table 3. The low-frequency two-state case.

	Fundamental MC	Antithetic variates	Control variates	Occupation time
V_1^2	2.7023 ± 0.0022	2.7019 ± 0.0013	2.7019 ± 0.0014	2.7032 ± 0.0011
V_2^2	3.3196 ± 0.0028	3.3196 ± 0.0016	3.3199 ± 0.0016	3.3197 ± 0.0010
Time (sec)	5.3670	2.9645	4.9100	0.0365
MSE	4.2666e−05	1.6323e−04	1.4110e−05	7.9885e−06
Efficiency	4.3670e+03	2.5909e+04	1.4434e+04	3.4330e+06

Table 4. The high-frequency two-state case.

	Fundamental MC	Antithetic variates	Control variates	Occupation time
V_1^2	3.0569 ± 0.0021	3.0573 ± 0.0015	3.0571 ± 0.0015	3.0574 ± 0.0012
V_2^2	3.0637 ± 0.0021	3.0642 ± 0.0015	3.0641 ± 0.0015	3.0636 ± 0.0011
Time (sec)	9.4756	5.0528	9.0112	0.0339
MSE	2.7902e−05	6.0305e−05	1.4990e−05	8.1818e−06
Efficiency	3.7823e+03	1.3128e+04	7.4030e+03	3.6071e+06

where $\mathrm{MSE}(\hat{\mu}) = \mathrm{Var}(\hat{\mu}) + B^2(\hat{\mu})$ is the mean square error of $\hat{\mu}$, $B(\hat{\mu}) = E(\hat{\mu}) - \mu$ is the bias of $\hat{\mu}$, and $C(\hat{\mu})$ is the expected computation time for $\hat{\mu}$.

As we see from the formula, the efficiency is inversely proportional to both the mean square error and the computational time. A larger Eff means a better estimator. We use the same parameters from the low-frequency case and the high-frequency case as we mentioned in Section 5.1. We start with the two-state case, in which the total occupation time algorithm is applicable.

According to Tables 3 and 4, each of the three techniques improves the simulation method in different degrees. The simulating total occupation time algorithm performs the best with a much higher efficiency. This is because the algorithm is much cheaper, even though its mean square error (MSE) is similar to that of the others. In addition, the computational time of the algorithm appears to be independent of the number of regimes, as the computational time in the low-frequency case is 0.0365 seconds against 0.0339 seconds in the high-frequency case. For the two variance-reduction techniques, the efficiencies of both methods decline from the low case to the high case. This is only because of more computation being involved, since the MSE's of the two methods are on the same level during the change of the cases. The antithetic variates estimator

https://doi.org/10.1017/S1446181117000335 Published online by Cambridge University Press

TABLE 5. The low-frequency multi-state case.

	Fundamental MC	Antithetic variates	Control variates
V_1^3	3.3555 ± 0.0022	3.3577 ± 0.0018	3.3568 ± 0.0016
V_2^3	3.7643 ± 0.0022	3.7664 ± 0.0019	3.7652 ± 0.0018
V_3^3	4.2496 ± 0.0022	4.2528 ± 0.0021	4.2518 ± 0.0019
Time (sec)	8.1438	4.5162	7.4225
MSE	3.8896e−05	1.4884e−04	1.3168e−05
Efficiency	3.1570e+03	1.3373e+04	1.0232e+04
V_1^4	4.1038 ± 0.0029	4.1035 ± 0.0021	4.1033 ± 0.0021
V_2^4	4.3808 ± 0.0031	4.3791 ± 0.0020	4.3800 ± 0.0021
V_3^4	4.7287 ± 0.0032	4.7275 ± 0.0020	4.7270 ± 0.0023
V_4^4	5.1273 ± 0.0033	5.1262 ± 0.0023	5.1253 ± 0.0025
Time (sec)	11.4680	6.7384	10.2069
MSE	3.9272e−05	2.2926e−04	1.6073e−05
Efficiency	1.4712e+03	1.0500e+04	6.0955e+03

outperforms the control variates estimator in efficiency, since it spends only roughly half of the computation time of the other two methods.

We put the multi-state cases together, including the regime-switching model with three states and four states. Again the same low case and high case are considered. Performances of the two variance-reduction methods are shown in Tables 5 and 6. According to these two tables, the same conclusion holds in the multi-state cases. Antithetic variates are the better choices for the variance-reduction technique with the higher efficiency in any situation. In addition, the MSE's of all of the three methods is independent of the parameter λ and the number of the states.

6. Conclusion

A comparative study of the MC and the CN for pricing European options with the regime-switching model is presented. Since the number of the PDEs within the governing system grows according to the number of regimes, solving the governing system becomes very inefficient as the number of regimes grows very large. Numerical results show that even for the two-state case, the fundamental MC is already more efficient than the CN in both the low-frequency case and the high-frequency case. The difference in efficiency becomes more severe as the number of regimes further increases. Such a finding suggests that future research on numerical techniques for regime-switching models should concentrate more on simulation-based methods.

https://doi.org/10.1017/S1446181117000335 Published online by Cambridge University Press

TABLE 6. The high-frequency multi-state case.

	Fundamental MC	Antithetic variates	Control variates
V_1^3	3.8642 ± 0.0023	3.8688 ± 0.0016	3.8692 ± 0.0016
V_2^3	3.8679 ± 0.0023	3.8715 ± 0.0016	3.8728 ± 0.0018
V_3^3	3.8733 ± 0.0023	3.8763 ± 0.0016	3.8783 ± 0.0019
Time (sec)	21.5188	11.4812	20.8839
MSE	2.9494e−05	1.0159e−04	1.5503e−05
Efficiency	1.5756e+03	7.7168e+03	3.0887e+03
V_1^4	4.6813 ± 0.0036	4.6822 ± 0.0023	4.6824 ± 0.0020
V_2^4	4.6835 ± 0.0036	4.6844 ± 0.0023	4.6846 ± 0.0020
V_3^4	4.6868 ± 0.0037	4.6878 ± 0.0024	4.6880 ± 0.0021
V_4^4	4.6913 ± 0.0037	4.6922 ± 0.0023	4.6923 ± 0.0021
Time (sec)	43.7829	23.4794	43.1711
MSE	4.3829e−05	2.8356e−04	1.3606e−05
Efficiency	5.2111e+02	2.4035e+03	1.7025e+03

We have also investigated two variance-reduction techniques: the antithetic variates and the control variates, to improve the efficiency of the simulation. Numerical performance shows that the antithetic variates are more efficient than the control variates with regime-switching models. Finally, we propose a much faster simulation algorithm for European options in the two-state regime-switching model. Computational time of the algorithm is independent of the switching frequency. However, such an algorithm is only applicable to the European options in a two-state regime-switching world. The algorithm for the multi-state cases is left for future research.

Acknowledgements

The first author would like to express his sincere gratitude to the Chinese Scholarship Council and the University of Wollongong (UOW) for providing funding supports that allowed him to carry out a PhD study at UOW. The third author would like to acknowledge the financial support from the Australian Research Council for funding the ARC(DP) Project (DP140102076), which has facilitated the collaboration among the three authors. In addition, the authors are grateful for the constructive comments and suggestions made by the two anonymous referees.

Appendix

Algorithm 1 Fundamental Monte Carlo

function FMC $(S_0, E, r, G, T, \sigma, j, N)$;

Require:

Initial stock price S_0;
Strike price E;
Interest rates r;
Generator matrix of the Markov chain $G = (\lambda_{mn}) \in \mathbb{R}^{K\times K}$;
Time to maturity T;
Volatilities vector $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_K)$;
Initial regime j $(j = 1, 2, \ldots, K)$;
Numbers of simulation paths N;

Ensure:

European put option price with respect to initial regime j;
for $n = 1$ to N **do**
  for $k = 1$ to K **do**
    $J[k] \leftarrow 0$;
  end for
  $t \leftarrow T$; $I \leftarrow j$; $Q \leftarrow 0$;
  while $t > 0$ **do**
    Generate $U_1, U_2 \sim$ uniform$(0, 1)$;
    $\tau \leftarrow \log(U_1)/\lambda_{II}$;
    $pr \leftarrow 0$; $m \leftarrow 1$;
    while $U_2 > pr$ **do**
      if $m \neq I$ **then**
        $pr \leftarrow pr - \lambda_{Im}/\lambda_{II}$;
      else
        $m \leftarrow m + 1$;
      end if
    end while
    if $\tau > t$ **then**
      $J[I] \leftarrow J[I] + t$;
    else
      $J[I] \leftarrow J[I] + \tau$;
    end if
    $t \leftarrow t - \tau$; $I \leftarrow m - 1$;
  end while
  for $k = 1$ to K **do**
    $Q \leftarrow Q + \sigma_k^2 J[k]$;
  end for
  Generate $W \sim$ normal$(0, 1)$;
  $Z[n] \leftarrow S_0 \exp\{(rT - Q/2) + \sqrt{Q}W\}$;
  $Y[n] \leftarrow e^{-rT} \max(E - Z[n], 0)$;
end for
$V = \frac{1}{N} \sum_{n=1}^{N} Y[n]$;
return V;

https://doi.org/10.1017/S1446181117000335 Published online by Cambridge University Press

Algorithm 2 Monte Carlo with antithetic variates

$Z_{\text{ori}}[n] \leftarrow S_0\exp\{(rT - Q/2) + \sqrt{Q}W\}$; $Z_{\text{ant}}[n] \leftarrow S_0\exp\{(rT - Q/2) - \sqrt{Q}W\}$
$Y[n] \leftarrow \frac{1}{2}e^{-rT}(\max(E - Z_{\text{ori}}[n], 0) + \max(E - Z_{\text{ant}}[n], 0))$;
$V = \frac{1}{N}\sum_{n=1}^{N} Y[n]$;
return V;

Algorithm 3 Monte Carlo with control variates

$\bar{Z} = \frac{1}{N}\sum_{n=1}^{N} Z[n]$; $\bar{Y} = \frac{1}{N}\sum_{n=1}^{N} Y[n]$;
$b = \dfrac{\sum_{n=1}^{N}(Z[n] - \bar{Z})(Y[n] - \bar{Y})}{\sum_{n=1}^{N}(Z[n] - \bar{Z})^2}$;
$V = \frac{1}{N}\sum_{n=1}^{N}(Y[n] - b(Z[n] - e^{rT}S_0))$;
return V;

Algorithm 4 Generating random numbers from the $f_{j|j}$

function GRN $(\lambda_{ji}, \lambda_{ij}, T, M)$;
Require:
Jump intensity λ_{ji}, λ_{ij};
Time to maturity T;
Numbers of time partitions M;
Ensure:
A random number following the pdf $f_{j|j}$;
$t[0] \leftarrow 0$;
$\Delta t = T/M$;
for $m = 1$ to M **do**
  $t[m] \leftarrow t[m-1] + \Delta t$;
  $f_j[m-1] \leftarrow e^{-\lambda_{ij}(T - t[m-1]) - \lambda_{ji}t[m-1]}[\lambda_{ji}I_0(2(\lambda_{ji}\lambda_{ij}t(T - t[m-1]))^{1/2})$
  $+\left(\dfrac{\lambda_{ji}\lambda_{ij}t}{T - t[m-1]}\right)^{1/2} I_1(2(\lambda_{ji}\lambda_{ij}t(T - t[m-1]))^{1/2})]$;
end for
$f_j[M] \leftarrow f_j[M-1] + e^{-\lambda_{ji}T}$;
$B \leftarrow 0$;
for $m = 0$ to M **do**
  $B \leftarrow B + f_j[m]$;
  $c_j[m] \leftarrow BT/M$;
end for
Generate $U \sim$ uniform$(0, 1)$;
$n \leftarrow 0$;
while $U > c_j[n]$ **do**
  $n \leftarrow n + 1$;
end while
return $t[n]$;

https://doi.org/10.1017/S1446181117000335 Published online by Cambridge University Press

Algorithm 5 Simulating total occupation time

function TOT $(S_0, E, r, G, T, \sigma, j, N, M)$;

Require:

Initial stock price S_0;
Strike price E;
Interest rates r;
Generator matrix of the Markov chain $G = (\lambda_{mn}) \in \mathbb{R}^{2\times 2}$;
Time to maturity T;
Volatilities vector $\sigma = (\sigma_1, \sigma_2)$;
Initial regime j $(j = 1, 2)$;
Numbers of simulation paths N;
Number of time partition for generating random variable from the pdf f_j M;

Ensure:

European put option price with respect to initial regime j;
for $n = 1$ to N **do**
    $J \leftarrow \mathrm{GRN}(\lambda_{ji}, \lambda_{ij}, T, M)$;
    $I \leftarrow T - J$;
    $Q \leftarrow \sigma_j^2 J + \sigma_i^2 I$;
    Generate $W \sim \mathrm{normal}(0, 1)$;
    $Z[n] = S_0 \exp\{(rT - U/2) + \sqrt{U}W\}$;
    $Y[n] = e^{-rT} \max(E - Z[n], 0)$;
end for
$V = \frac{1}{N} \sum_{n=1}^{N} Y[n]$;
return V;

References

[1] F. Black and M. Scholes, "The pricing of options and corporate liabilities", *J. Polit. Econ.* **81** (1973) 637–654; doi:10.1086/260062.

[2] N. P. B. Bollen, "Valuing options in regime-switching models", *J. Deriv.* **6** (1998) 38–49; doi:10.3905/jod.1998.408011.

[3] P. Brandimarte, *Numerical methods in finance and economics: a MATLAB-based introduction*, 2nd edn (John Wiley, Hoboken, NJ, 2006).

[4] J. Buffington and R. J. Elliott, "Regime switching and European options", in: *Stochastic theory control*, Volume 280 of *Lect. Notes in Control and Information Sciences* (ed. B. Pasik-Duncan), (Springer, Berlin–Heidelberg, 2002) 73–82.

[5] J. Detemple, *American-style derivatives: valuation and computation* (CRC Press, Boca Raton, FL, 2005).

[6] R. J. Elliott, L. Chan and T. K. Siu, "Option pricing and Esscher transform under regime switching", *Ann. Finance* **1** (2005) 423–432; doi:10.1007/s10436-005-0013-z.

[7] F. Fang and C. W. Oosterlee, "A novel pricing method for European options based on Fourier-cosine series expansions", *SIAM J. Sci. Comput.* **31** (2009) 826–848; doi:10.1137/080718061.

[8] C. D. Fuh, R. H. Wang and J. C. Cheng, "Option pricing in a Black–Scholes model with Markov switching", 2002, working paper, https://www.researchgate.net/publication/2880608_Option_Pricing_in_a_Black-Scholes_Model_with_Markov_Switching.

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

[10] X. Guo, "Information and option pricings", *Quant. Finance* **1** (2001) 38–44; doi:10.1080/713665550.

[11] J. D. Hamilton, "A new approach to the economic analysis of nonstationary time series and the business cycle", *Econometrica* **57** (1989) 357–384; doi:10.2307/1912559.
[12] P. Hieber and M. Scherer, "Efficiently pricing barrier options in a Markov-switching framework", *J. Comput. Appl. Math.* **235** (2010) 679–685; doi:10.1016/j.cam.2010.06.021.
[13] C. Lemieux, *Monte Carlo and quasi-Monte Carlo sampling*, *Springer Ser. Statist.* (Springer, New York, 2009).
[14] M. A. Mielkie, "Options pricing and hedging in a regime-switching volatility model", Ph.D. Thesis, The University of Western Ontario, 2014; http://ir.lib.uwo.ca/cgi/viewcontent.cgi?article=3581&context=etd.
[15] M. Mielkie and M. Davison, "Investigating the market price of volatility risk for options in a regime-switching market", *SSRN Electron. J.* (2013); doi:10.2139/ssrn.2326534.
[16] F. L. Yuen and H. Yang, "Option pricing with regime switching by trinomial tree method", *J. Comput. Appl. Math.* **233** (2010) 1821–1833; doi:10.1016/j.cam.2009.09.019.
[17] S. P. Zhu, A. Badran and X. Lu, "A new exact solution for pricing European options in a two-state regime-switching economy", *Comput. Math. Appl.* **64** (2012) 2744–2755; doi:10.1016/j.camwa.2012.08.005.