

Research Article

A Multilevel Monte Carlo Method for the Valuation of Swing Options

Hakimeh Ghodssi-Ghassemabadi  and **Gholam Hossein Yari** 

School of Mathematics, Iran University of Science and Technology, Tehran, Iran

Correspondence should be addressed to Gholam Hossein Yari; gholamhosseinyari91@gmail.com

Received 30 November 2019; Revised 22 September 2020; Accepted 4 December 2020; Published 19 January 2021

Academic Editor: Bekir Sahin

Copyright © 2021 Hakimeh Ghodssi-Ghassemabadi and Gholam Hossein Yari. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this study, we propose a novel approach for the valuation of swing options. Swing options are a kind of American options with multiple exercise rights traded in energy markets. Longstaff and Schwartz have suggested a regression-based Monte Carlo method known as the least-squares Monte Carlo (LSMC) method to value American options. In this work, first we introduce the LSMC method for the pricing of swing options. Then, to achieve a desired accuracy for the price estimation, we combine the idea of LSMC with multilevel Monte Carlo (MLMC) method. Finally, to illustrate the proper behavior of this combination, we conduct numerical results based on the Black–Scholes model. Numerical results illustrate the efficiency of the proposed approach.

1. Introduction

Swing options are American type contracts with multiple early exercise rights in energy markets, such as gas, oil, and electricity. In these markets, unknown future demands due to weather conditions, political situations, and unstorability of energy lead to price fluctuations. Swing contracts give the option holders multiple exercise rights in long or short positions up to the maturity time to protect themselves from fluctuations in commodity prices. A swing option gives its holder the right to change the volume of periodic deliveries of a particular commodity on certain dates in the future, within a given delivery period, at the prescribed strike price [1].

American option has no closed-form pricing formula, and consequently swing options have no explicit solutions; thus, numerical methods should be employed to price these options approximately. Numerical methods for pricing the swing options can be classified into the following three categories: lattice/tree methods, numerical methods for partial differential equations (PDEs), and Monte Carlo (MC) simulations. Thompson [2] was the first to present a paper on the pricing of swing options as take-or-pay options, in which he proposed a binomial tree algorithm to price a

version of swing options. Later, Lari-Lavassani et al. developed another binomial tree in which the spot price was a two-factor mean-reverting process [3]. Jaillet et al. applied multi-layered trinomial trees based on a dynamic programming approach for valuing swing options [4]. Indeed, they proposed a one-factor mean-reverting process for energy prices that explicitly incorporates seasonal effects. In the study by Shao and Xiang [5], the pricing problem was formulated as an optimal stochastic control problem, which was solved by the trinomial forest approach while the underlying asset followed a seasonal mean-reverting regime-switching model. Moreover, Kjaer in [6] implemented a finite difference method to solve PIDEs arising from a dynamic programming problem associated with the pricing of swing options. It is worth mentioning that due to easy implementation of PDE-based methods, they have become popular in financial problems in the last decades [6, 7].

Carmona and Touzi viewed swing options as optimal multiple stopping problems and solved the problem using the MC method and Malliavin calculus [8]. They used the theory of the Snell envelope to determine the optimal exercise boundaries. Zeghal and Mnif in [9] extended the latter method to Lévy processes for valuing swing options. Ben Latifa et al. in [10] valued swing options with the jump in

price process. They transformed optimal multiple stopping time problem into a sequence of ordinary stopping time problems and solved them as the unique viscosity solution of Hamilton–Jacobi–Bellman variational inequality. Moreover, Naouara and Trabelsi [11] proposed a generalization of swing options to the case of an optimal multiple stopping time problem with random number of exercise rights. They transformed the main problem into a sequence of standard optimal stopping time problems using dynamic programming.

The least-squares Monte Carlo (LSMC) method is a regression-based Monte Carlo simulation for valuing early exercise options. For the first time, Carriere [12] proposed the LSMC method, which was later used by Longstaff and Schwartz [13] for approximating the value of American options. These studies applied regression to approximate value functions in their backwards dynamic program. Figueroa [14] used an extension of the LSMC method for valuing swing options, as described in [15]. In these papers, regression was used to estimate continuation value (CV), and decisions for exercising rights were made based on the estimated CV. The pricing of American-type options via the MC method is a challenging task, as it requires the backward dynamic programming algorithm, which seems to be incompatible with the forward structure of MC methods [16]. Belomestny et al. in [16] implemented local and global regression methods to estimate the CV. They combined this idea with multilevel Monte Carlo (MLMC) method to price Bermudan options to reduce the complexity of the LSMC method.

In this paper, we focus on the pricing of swing options using regression-based methods. We combine the LSMC approach with the MLMC method to decrease the computational complexity of the MC method by varying the number of paths on different levels and using the variance reduction method (we refer the reader to [16, 17] for a survey), also to obtain more accurate approximation with a lower variance. To illustrate the methodology, we consider a numerical example based on the Black–Scholes model using different values of root mean square error (RMSE). Also, due to large computational times, we deal with a swing option with a small maximum number of exercise rights.

This paper is structured as follows: In Section 2, we introduce swing options. We present the LSMC method for the pricing of swing options in Section 3. In Section 4, we give a brief overview of the MLMC method and introduce its algorithm for the pricing of swing options. Section 5 provides a numerical example for pricing swing put options via the MLMC scheme. Also, in this section, we compare our pricing results with those obtained by Carmona and Touzi [8]. Finally, we conclude the paper in Section 6.

2. Swing Options

Swing options, with multiple exercise rights, play a major role in energy markets with limited and costly storage. These options offer flexibility regarding when and how much of a commodity is acquired and also provide protection against daily price fluctuations from the beginning until the expiration date of the contract. In other words, a swing option

provides the opportunity to repeatedly exercise the right of receiving variable amounts of the commodity at certain prices, regularly. The swing option could be considered as a generalization of the Bermudan option [4].

Swing options allow their holders to exercise one right in each predetermined exercise date from the beginning to the expiration date of the contract. Suppose that the lifetime of a swing contract is the interval $[0, T]$, and the possible exercise times are $t_1, t_2, \dots, t_n = T$, s.t. $t_i > 0$ for $i = 1, 2, \dots, n$, $\delta_i = t_i - t_{i-1}$ is the refraction time (i.e., the time between two successive exercise dates), and $t_0 = 0$. Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a complete probability space, with $\mathcal{F} = \{\mathcal{F}_t | 0 \leq t \leq T\}$ is the filtration with the σ -algebra \mathcal{F}_t at time t . Suppose that $\{S_t, 0 \leq t \leq T\}$ is the underlying price process defined on the filtered probability space. Consider $\Phi(S_t)$, K_t , and $\ell \in \mathbb{N}$ denote payoff function, strike price of the option at time t , and maximum number of exercise rights of the option, respectively. The strike price may be constant across all t or it may follow a stochastic process. The value function of a swing option with ℓ exercise rights at time 0 is calculated using the following formula:

$$v^{(\ell)}(0, s) = \sup_{\mathcal{T}_{0,T}^{(\ell)}} \mathbb{E} \left[\sum_{k=1}^{\ell} e^{-r t_{d_k}} \Phi(S_{t_{d_k}}) | S_{t_0} = s \right], \quad (1)$$

where $\mathcal{T}_{0,T}^{(\ell)} = \{(t_{d_1}, t_{d_2}, \dots, t_{d_\ell}) | 0 < t_{d_i} \leq T \text{ and } t_{d_i} - t_{d_{i-1}} \geq \delta_i \forall i = 1, 2, \dots, \ell\}$, with $\{d_1, d_2, \dots, d_\ell\} \subset \{1, 2, \dots, n\}$, and r is the risk-free interest rate.

3. LSMC Method for Pricing Swing Options

The LSMC algorithm for the pricing of swing options is based on stochastic dynamic programming (SDP) which approximates the conditional expectations within the SDP scheme using least-squares estimates. The idea is based on dynamic programming which begins from the maturity to the start time. At each time t_k , two quantities must be compared which are the payoff from immediate exercise and the CV. CV is the conditional expectation of the option payoff with respect to the risk-neutral pricing measure Q :

$$C_T(S_T) = 0, \quad (2)$$

$$C_t(S_t) = \mathbb{E}_Q[\nu_{t+1}(S_{t+1}) | S_t], \quad \text{for } t = t_{n-1}, \dots, t_1, \quad (3)$$

$$\nu_t(S_t) = \max\{\Phi(S_t), C_t(S_t)\}, \quad (4)$$

where $C_t(S_t)$ and $\nu_t(S_t)$, respectively, are the CV and option values at time t when the price is S_t . In order to approximate the CV in (3), the least-squares regression on a finite set of basis functions is utilized. Basis functions can be monomials, Laguerre polynomials, splines, and radial basis functions. By estimating the CV, one can find the optimal exercise strategy along each path on each exercise date.

Recursive equations (2)–(4) for swing options with ℓ remaining exercise rights are rewritten as follows:

$$C_T^\ell(S_T) = 0, \quad (5)$$

$$C_t^\ell(S_t) = \mathbb{E}[\nu_{t+1}^{(\ell)}(S_{t+1})|S_t], \quad \text{for } t = t_{n-1}, \dots, t_1, \quad (6)$$

$$\nu_t^{(\ell)}(S_t) = \max\{\Phi(S_t) + C_t^{\ell-1}(S_t), C_t^\ell(S_t)\}. \quad (7)$$

To use the LSMC algorithm for valuing swing options, we have to modify the original algorithm proposed in [13] with an extra dimension in cash flow matrices as described by Dörr in [15]. Below, we briefly present the steps of the LSMC algorithm for valuing swing options:

Step 1. A certain number of sample paths of the underlying asset is simulated and stored in a matrix S .

Step 2. Payoffs from the immediate exercise at each time step is stored in a matrix P .

Step 3. Cash flow (CF) matrices of the remaining exercise rights is determined at each time step where the number of determined CF matrices at each time step is equal to ℓ . The aforementioned matrices should be updated at each time step by moving backward in time from T . In CF matrices, the number of rows is equal to the number of sample paths, and the number of columns is equal to the number of exercise times. For obtaining CF matrices, following steps are needed:

- (i) According to equation (5), the value of CV equals zero at the maturity, and the CF matrix of the κ remaining exercise rights, $\kappa = \ell, \ell - 1, \dots, 1$, is evaluated by initializing the last κ columns of the CF matrix according to the last κ columns of the matrix P . The other columns of the CF matrix remain undefined.
- (ii) CF matrices at time $t < T$ in which $T - t < \ell$ are evaluated as described for the maturity time evaluation.
- (iii) For $t = t_{n-1}, \dots, t_1$ in which $T - t \geq \ell$, CF matrices should be updated according to the decision making in (7) i.e., Comparing two quantities: the values of CV and payoff from immediate exercise plus CV with one exercise right less than the original one:

For estimating the CV at time t with κ remaining exercise rights, first, we choose in-the-money (ITM) paths (i.e., paths at the t -th column of the matrix S with positive corresponding elements in the matrix P). Then, we regress discounted values of the CF matrix with κ exercise rights at time $t + 1$ onto the price vector at time t and estimate the CV vector. After estimating the CV, we compare quantities in (7) and make decisions. If it is optimal to exercise the right at time t and path u , the (u, t) entry of the CF matrix, i.e., the element of the u -th row and t -th column of CF, will be replaced with the (u, t) entry of the matrix P . The other paths' elements at time t are replaced with 0

and the next columns of CF are replaced with corresponding elements of the CF matrix at time $t + 1$ with one exercise right less. Otherwise, the CF matrix equals the same one at time $t + 1$.

Step 4. After determining the CF matrices at time t_1 and discounting them to the start time, t_0 , the option value will be evaluated by taking the average across paths.

4. Multilevel Monte Carlo Scheme for Pricing Swing Options

MLMC method was first introduced by Heinrich [18] for parametric integration. Giles proposed it for the estimation of an expected value arising from stochastic differential equations (SDEs) in finance [19]. Applying the MLMC method with Euler–Maruyama (EM) discretization reduces the computational complexity of MC simulations for payoff estimation from $O(\epsilon^{-3})$ to $O(\epsilon^{-2}(\log \epsilon)^2)$ to achieve a root mean square error (RMSE) of ϵ [17].

The MLMC method applies several levels of l , where $l = 0, 1, \dots, L$. $l = 0$ and $l = L$ correspond to the coarsest and finest levels, respectively. Each level uses steps of length $h_l = (T/M^l)$ for discretization to simulate paths in order to approximate the SDE, where M is the refinement factor and $M \geq 2$. The main idea of the MLMC method is to evaluate the expectation in the finest level as a telescopic sum and to apply the standard MC method to estimate each expectation as follows:

$$\mathbb{E}[\hat{P}_L] = \mathbb{E}[\hat{P}_0] + \sum_{l=1}^L \mathbb{E}[\hat{P}_l - \hat{P}_{l-1}], \quad (8)$$

where \hat{P}_l denotes the approximation of the payoff function on level l . Considering that \hat{Y}_0 and \hat{Y}_l , respectively, are the estimates for $\mathbb{E}[\hat{P}_0]$ and $\mathbb{E}[\hat{P}_l - \hat{P}_{l-1}]$ according to the MC method, we have

$$\begin{aligned} \hat{Y}_0 &= \frac{1}{N_0} \sum_{j=1}^{N_0} \hat{P}_0^{(j)}, \\ \hat{Y}_l &= \frac{1}{N_l} \sum_{j=1}^{N_l} (\hat{P}_l^{(j)} - \hat{P}_{l-1}^{(j)}), \end{aligned} \quad (9)$$

where the upper index j denotes the j th realization.

It is remarkable that the trajectories of simulating paths in the estimation of $\mathbb{E}[\hat{P}_l - \hat{P}_{l-1}]$ are the same. They are chosen from the finest level, though different step sizes h_l and h_{l-1} are applied. Let V_0 and V_l denote the variance of \hat{P}_0 and $(\hat{P}_l - \hat{P}_{l-1})$ for $l > 1$, respectively. V_l decreases with l , and after decreasing in variance, very few samples will be required on finer levels to estimate the expected values accurately [20].

Applying the MLMC method requires the discretization of SDE. The Euler–Maruyama (EM) scheme is one of the discretization methods. Consider an SDE of the usual form as follows:

$$\begin{aligned} dS_t &= \mu(S_t)dt + \sigma(S_t)dW_t, \\ S_0 &= s_0, \end{aligned} \quad (10)$$

where $\{W_t\}$ is a Wiener process. For SDE (10), the EM approximations of S_t (i.e., \hat{S}_{t_i} for $i \in \{0, 1, \dots, n\}$) are defined recursively through $\hat{S}_{t_0} = S_0$ and

$$\hat{S}_{t_{i+1}} = \hat{S}_{t_i} + \mu(\hat{S}_{t_i})\Delta t_{i+1} + \sigma(\hat{S}_{t_i})\Delta W_{i+1}, \quad (11)$$

with $\Delta t_{i+1} = t_{i+1} - t_i$, and $\Delta W_{i+1} = W_{t_{i+1}} - W_{t_i}$ for $i = 0, 1, \dots, n-1$.

The MLMC algorithm for pricing swing options is presented in which we choose N_l adaptively, which reduces the computational cost to achieve the desired root mean square error (Algorithm 1).

5. Numerical Result

In this section, we use the combination of the LSMC and MLMC methods to value a swing put option. We consider a swing put contract with maximum number of exercise rights $\ell = 5$. The price process S_t follows a geometric Brownian motion given as follows:

$$dS_t = rS_t dt + \sigma S_t dW_t, \quad (12)$$

where $\{W_t\}_{t \geq 0}$ is a Wiener process, r and σ are constant risk-free rate of interest and volatility, respectively. The parameter will be set as follows: $S_0 = 100$, $K = 100$, $\sigma = 0.3$, and $r = 0.05$. The life of the contract is in the interval $[0, 1]$, and the payoff function is $\Phi(S_t) = \max(K - S(t), 0)$. Furthermore, we consider $n = 5$.

To approximate the CV in (6), we choose a set of functions $\{1, x, x^2\}$ as basis functions for regression. All simulations are carried out on a computer with Intel (R) Core (TM) i5-4310U CPU @ 2.60 GHz processors with 16.0 GB RAM. To approximate SDE in formula (12), we use the EM discretization defined in formula (11) on the uniform time step $h_l = (T/M^l)$ on each level l in order to simulate the price paths. Here, we determine the value of M according to the number of exercise rights and exercise times considered in the problem. Due to the number of exercise rights in the problem, we have to choose $M \geq 5$ and also start the multilevel estimators in formula (8) from level 1:

$$\mathbb{E}[\hat{P}_L] = \mathbb{E}[\hat{P}_1] + \sum_{l=2}^L \mathbb{E}[\hat{P}_l - \hat{P}_{l-1}]. \quad (13)$$

In the MLMC method, we consider $M = 5$ and $N_l^{\text{init}} = 100$. For more details of implementing the MLMC method, we give a brief description of the MLMC algorithm for two levels:

- (i) For the first level, $l = 1$, we start with simulating $N_1^{\text{init}} = 100$ sample paths via the EM method and store them in a matrix S , after that, we use the least-squares regression method presented in Section 3 to estimate the values of \hat{P}_1 (there are N_1^{init} values of \hat{P}_1).

We evaluate the variance of this estimation as V_1 and utilize this value to determine the required optimal number of sample paths, N_1^{opt} , using the equation given in (iii) in Algorithm 1. By determining N_1^{opt} , we simulate $(N_1^{\text{opt}} - N_1^{\text{init}})$ new sample paths via the EM algorithm and use the total number of N_1^{opt} sample paths for estimating \hat{P}_1 . Lastly, we use the following formula in order to approximate \hat{Y}_1 :

$$\hat{Y}_1 = \frac{1}{N_1^{\text{opt}}} \sum_{j=1}^{N_1^{\text{opt}}} \hat{P}_1^{(j)}. \quad (14)$$

- (ii) For the second level, we implement the same method as described for the first level, just the difference is that, here, we need to approximate \hat{Y}_2 . Therefore, we should use N_2^{init} and N_2^{opt} number of sample paths for estimating both values of \hat{P}_1 and \hat{P}_2 and finally

$$\hat{Y}_2 = \frac{1}{N_2^{\text{opt}}} \sum_{j=1}^{N_2^{\text{opt}}} (\hat{P}_2^{(j)} - \hat{P}_1^{(j)}). \quad (15)$$

As a benchmark solution, we use the results of the MC-based method proposed by Carmona and Touzi in [8]. These swing option prices are $v_0^1 = 9.85$, $v_0^2 = 19.26$, $v_0^3 = 28.80$, $v_0^4 = 38.48$, and $v_0^5 = 48.32$. Swing option values using the MLMC method considering $\varepsilon \in \{1e-01, 5e-02, 1e-1.4\}$, $L = 4$, and absolute errors are reported in Table 1.

In Table 2, we compare the results of pricing the swing options using the MLMC method with $L = 3$ and $\varepsilon = 1e-1.7$ and the LSMC method using 500,000 sample paths. The number of sample paths in levels 1 to 3 for obtaining results of the MLMC method in Table 2 is 16085592, 5454601, and 3325175, respectively. Comparing the results of the MLMC and LSMC methods in these tables with the benchmark values reveals that the MLMC method is more accurate than the LSMC method for valuing swing options. It is remarkable that we obtain the results of the MLMC method in Table 2 for $L = 3$ (because of the lack of memory). Due to implementing regression, the run-time is high in both methods.

Figure 1 shows the optimal number of paths for each level according to the formula given in (iii) in Algorithm 1, for different values of $\varepsilon = 1e-01, 5e-02$, and $1e-1.4$. The optimal number of paths decreases in each level l in order to achieve the desired accuracy. Also, the required number of sample paths increase for obtaining more accurate approximation. In Figure 2, we report the variance reduction of the multilevel correction $(\hat{P}_l - \hat{P}_{l-1})$, which we denote by V_l . As shown in this figure, an increase in the number of levels leads to a reduction in the values of variance in the MLMC method. We compare this reduction with the variance of \hat{P}_l , which is the approximation of the option price for the LSMC method using the EM discretization with the time step

Input: desired accuracy ε , refinement factor M , expiration date T , number of levels L , and number of exercise rights ℓ .
Output: estimation of discounted payoff.

(1) **For** $l = 0, 1, \dots, L$ **do**:

 Choose initial number of sample paths, N_l^{init} .

If $l = 0$

 (i) Generate N_l^{init} sample paths according to a discretization scheme such as (11).

 (ii) Use the least-squares regression method to estimate \hat{P}_l , then determine V_l .

 (iii) Determine the optimal number of sample paths by the formula:

$$N_l^{\text{opt}} = \lceil 2\varepsilon^{-2} \sqrt{V_l h_l} \sum_{l=0}^L \sqrt{V_l / h_l} \rceil.$$

 (iv) Simulate extra samples, $(N_l^{\text{opt}} - N_l^{\text{init}})$, by the formula in (11).

 (v) Use the least-squares regression method on N_l^{opt} paths to estimate \hat{P}_l .

 (vi) Estimate \hat{Y}_0 by (9).

else

 (a) Generate N_l^{init} sample paths according to the EM scheme in (11), and use these paths for both levels l and $l-1$ (coarsen level l to the level $l-1$), we refer the interested readers to [21] for more details.

 (b) Use the least-squares regression method to estimate $\hat{P}_l - \hat{P}_{l-1}$, and also determine V_l using the following formula:

$$V_l = (1/N_l^{\text{init}} - 1) [\sum_{i=1}^{N_l^{\text{init}}} (\hat{P}_l - \hat{P}_{l-1})^2 - (1/N_l^{\text{init}}) (\sum_{i=1}^{N_l^{\text{init}}} (\hat{P}_l - \hat{P}_{l-1}))^2].$$

 (c) Determine the optimal number of sample paths using the equation in (iii).

 (d) Generate $N_l^{\text{opt}} - N_l^{\text{init}}$ sample paths by the EM scheme in (11) and use these paths for both levels l and $l-1$.

 (e) Use the least-squares regression method on N_l^{opt} paths to estimate $\hat{P}_l - \hat{P}_{l-1}$.

 (f) Estimate \hat{Y}_l by (9).

(2) **End For**

(3) Approximate the value by (8).

ALGORITHM 1: MLMC Algorithm for the pricing of swing options.

TABLE 1: Values of swing put option with $\ell = 5$, using the MLMC method ($L = 4$).

$\gamma_0^{(\ell)}$	$\varepsilon = 1e - 01$	Error	$\varepsilon = 5e - 02$	Error	$\varepsilon = 1e - 1.4$	Error
$\gamma_0^{(1)}$	9.78	(0.07)	9.63	(0.22)	9.48	(0.37)
$\gamma_0^{(2)}$	19.48	(0.22)	19.26	(0.00)	19.03	(0.23)
$\gamma_0^{(3)}$	29.31	(0.51)	28.87	(0.07)	28.68	(0.12)
$\gamma_0^{(4)}$	39.28	(0.80)	38.57	(0.09)	38.45	(0.03)
$\gamma_0^{(5)}$	49.29	(0.97)	48.60	(0.28)	48.38	(0.06)
CPU time (s)	545.8308		5618.0910		3232.4965	

TABLE 2: Value of swing put option with $\ell = 5$, using the MLMC method ($L = 3$, $\varepsilon = 1e - 1.7$) and LSMC method.

$\gamma_0^{(\ell)}$	Results of the MLMC	Error	Results of the LSMC	Error
$\gamma_0^{(1)}$	9.93	(0.08)	10.13	(0.28)
$\gamma_0^{(2)}$	19.68	(0.42)	19.76	(0.50)
$\gamma_0^{(3)}$	29.31	(0.51)	28.87	(1.07)
$\gamma_0^{(4)}$	38.98	(0.50)	37.41	(0.08)
$\gamma_0^{(5)}$	48.92	(0.60)	45.36	(2.96)
CPU time (s)	1119.0962		1165.0012	

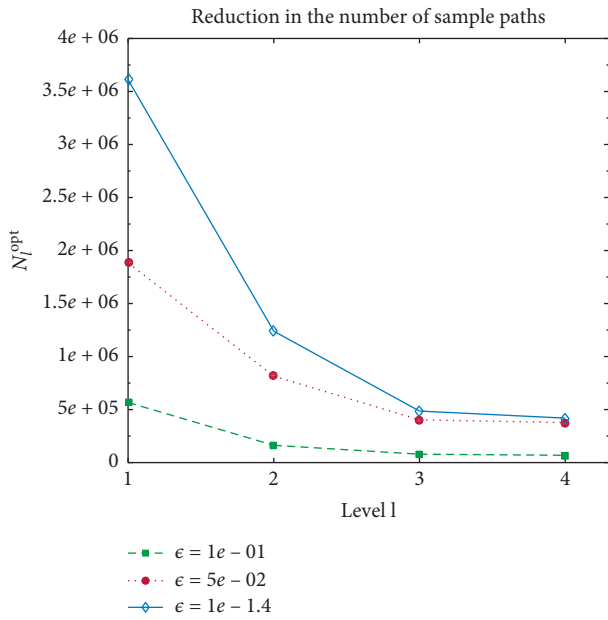


FIGURE 1: Optimal number of sample paths per level l , i.e., N_l^{opt} , in MLMC method for $\epsilon \in \{1e-01, 5e-02, 1e-1.4\}$ with $L = 4$.

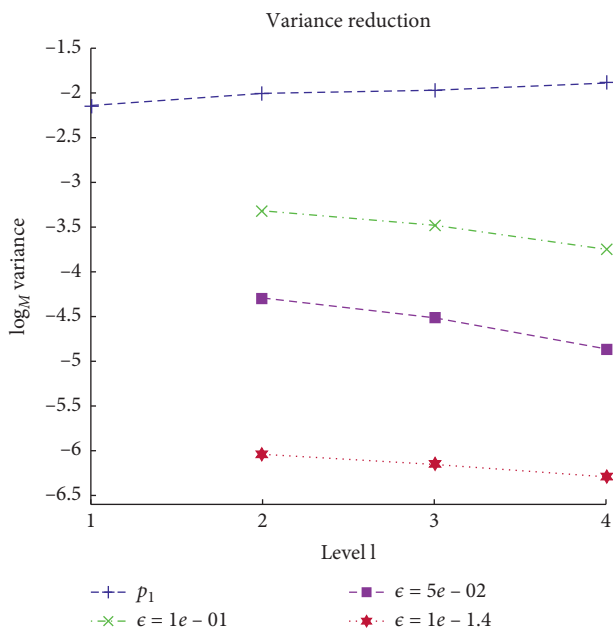


FIGURE 2: Level log variances: variance of multilevel correction, $P_l - P_{l-1}$, using the MLMC method for $\epsilon \in \{1e-01, 5e-02, 1e-1.4\}$, $L = 4$, and variance of P_l implementing the LSMC method, $M = 5$.

$$h_l = (T/M^l).$$

6. Conclusions

In this paper, we have studied the numerical pricing of the swing put option under a geometric Brownian motion model. To obtain a numerical solution to the problem, we have employed the least-squares regression method along

with the multilevel Monte Carlo method. We calculated the optimal number of samples for each level adaptively and ran the LSMC algorithm at each level. By increasing the number of levels, we observed a reduction in the optimal number of sample paths and the variance of multilevel corrections, which led to the accurate valuation of swing options with less computation. However, running the algorithm is hard-demanding due to the backward characteristic of the problem. Indeed, decreasing the optimal number of sample paths at each level is the key benefit of using the MLMC method, which leads to lower cost. To show the MLMC method's efficiency, we conducted numerical results with different targeted values of RMSE and compared the obtained results with the previous study and results from the LSMC method.

Several further extensions to this work could be (i) using different discretization methods with the higher-order convergence rate for simulation and investigating their effects on the approximated solutions and run-time of the algorithm and (ii) using different regression methods as represented in [16] and proposing the other types of basis functions in regression.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] D. Pilipovic and J. Wengler, "Getting into the swing," *Energy and Power Risk Management*, vol. 2, no. 10, pp. 22–24, 1998.
- [2] A. C. Thompson, "Valuation of path-dependent contingent claims with multiple exercise decisions over time: the case of take-or-pay," *The Journal of Financial and Quantitative Analysis*, vol. 30, no. 2, pp. 271–293, 1995.
- [3] A. Lari-Lavassani, M. Simchi, and A. Ware, "A discrete valuation of swing options," *Canadian Applied Mathematics Quarterly*, vol. 9, no. 1, pp. 35–74, 2001.
- [4] P. Jaillet, E. I. Ronn, and S. Tompaidis, "Valuation of commodity-based swing options," *Management Science*, vol. 50, no. 7, pp. 909–921, 2004.
- [5] L. Shao and K. Xiang, "Valuation of swing options under a regime-switching mean-reverting model," *Mathematical Problems in Engineering*, vol. 2019, Article ID 5796921, 14 pages, 2019.
- [6] M. Kjaer, "Pricing of swing options in a mean reverting model with jumps," *Applied Mathematical Finance*, vol. 15, no. 5–6, pp. 479–502, 2008.
- [7] M. C. Calvo-Garrido, M. Ehrhardt, and C. Vázquez, "Jump-diffusion models with two stochastic factors for pricing swing options in electricity markets with partial-integro differential equations," *Applied Numerical Mathematics*, vol. 139, pp. 77–92, 2019.
- [8] R. Carmona and N. Touzi, "Optimal multiple stopping and valuation of swing options," *Mathematical Finance*, vol. 18, no. 2, pp. 239–268, 2008.
- [9] A. B. Zeghal and M. Mnif, "Optimal multiple stopping and valuation of swing options in Lévy models," *International*

- Journal of Theoretical and Applied Finance*, vol. 9, no. 8, pp. 1267–1297, 2006.
- [10] I. Ben Latifa, J. F. Bonans, and M. Mnif, “Numerical methods for an optimal multiple stopping problem,” *Stochastics and Dynamics*, vol. 16, no. 5, pp. 715–739, 2016.
 - [11] N. J. B. Naouara and F. Trabelsi, “Generalization on optimal multiple stopping with application to swing options with random exercise rights number,” *Mathematical Control and Related Fields*, vol. 5, no. 4, pp. 807–826, 2015.
 - [12] J. F. Carriere, “Valuation of the early-exercise price for options using simulations and nonparametric regression,” *Insurance: Mathematics and Economics*, vol. 19, no. 1, pp. 19–30, 1996.
 - [13] F. A. Longstaff and E. S. Schwartz, “Valuing American options by simulation: a simple least-squares approach,” *The Review of Financial Studies*, vol. 14, no. 1, pp. 113–147, 2001.
 - [14] M. G. Figueroa, “Pricing multiple interruptible-swing contracts,” *Birbeck School of Economics, Mathematics and Statistics*, no. 606, 2006.
 - [15] U. Dörr, “Valuation of swing options and examination of exercise strategies by Monte Carlo techniques,” MSc. Thesis, University of Oxford, Oxford, UK, , 2003.
 - [16] D. Belomestny, F. Dickmann, and T. Nagapetyan, “Pricing Bermudan options via multilevel approximation methods,” *SIAM Journal on Financial Mathematics*, vol. 6, no. 1, pp. 448–466, 2015.
 - [17] M. B. Giles, “Multilevel Monte Carlo methods,” *Acta Numerica*, vol. 24, pp. 259–328, 2015.
 - [18] S. Heinrich, “Monte Carlo complexity of global solution of integral equations,” *Journal of Complexity*, vol. 14, no. 2, pp. 151–175, 1998.
 - [19] M. B. Giles, “Multilevel Monte Carlo path simulation,” *Operations Research*, vol. 56, no. 3, pp. 607–617, 2008.
 - [20] M. B. Giles and L. Szpruch, “Multilevel Monte Carlo methods for applications in finance,” *Recent Developments in Computational Finance*, pp. 3–47, World Scientific, Singapore, 2013.
 - [21] D. J. Higham, “An introduction to multilevel Monte Carlo for option valuation,” *International Journal of Computer Mathematics*, vol. 92, no. 12, pp. 2347–2360, 2015.