# Project: Multilevel Monte Carlo

Paul-Antoine Leveilley
Ryan Timeus

January 12, 2025

## Contents

# 1 Introduction

In financial engineering, pricing financial contracts often involve estimating expectations of the type $\mathbb{E}[f(S_T)]$, where $(S_t)_{t\in[0,T]}$ is the solution the stochastic differential equation (SDE) :

$$dS_t = b(t, S_t)dt + \sigma(t, S_t)dW_t$$

And $f$ is a scalar function with lipschitz bounds. A simple way to estimate this quantity is to discretize the SDE and replace the expectation with an empirical mean. The discretization with time step $h = \frac{T}{M}$ consists in setting $\hat{S}_0 = S_0$ and for $j \in \{0, \dots, M-1\}$, computing :

$$\hat{S}_{j+1} = \hat{S}_j + b(t_j, \hat{S}_j)h + \sigma(t_j, \hat{S}_j)\Delta W_j$$

Where $(\Delta W_j)_{j\in\{0,\dots,M-1\}}$ is an i.i.d. sample of centered Gaussian random variables with standard deviation $\sqrt{h}$. An simple estimator for $\mathbb{E}[f(S_T)]$ is $\hat{Y} = \frac{1}{N}\sum_{i=1}^{N} f(\hat{S}_M^{(i)})$ where $(\hat{S}_M^{(i)})_{i\in\{1,\dots,N\}}$ are $N$ i.i.d. realizations of $\hat{S}_M$. We define the mean square error (MSE) of this estimator by :

$$MSE(\hat{Y}) = \mathbb{E}\left[\left(\hat{Y} - \mathbb{E}[f(S_T)]\right)^2\right]$$

The MSE can be decomposed in the following way :

$$\begin{aligned}
MSE(\hat{Y}) &= \mathbb{E}\left[\left(\hat{Y} - \mathbb{E}[f(\hat{S}_M)] + \mathbb{E}[f(\hat{S}_M)] - \mathbb{E}[f(S_T)]\right)^2\right] \\
&= \mathbb{E}\left[\left(\hat{Y} - \mathbb{E}[f(\hat{S}_M)]\right)^2\right] + \mathbb{E}\left[\left(\mathbb{E}[f(\hat{S}_M)] - \mathbb{E}[f(S_T)]\right)^2\right] \\
&+ 2\mathbb{E}\left[\left(\hat{Y} - \mathbb{E}[f(\hat{S}_M)]\right)\left(\mathbb{E}[f(\hat{S}_M)] - \mathbb{E}[f(S_T)]\right)\right] \\
&= \mathbb{E}\left[\left(\hat{Y} - \mathbb{E}[f(\hat{S}_M)]\right)^2\right] + \mathbb{E}\left[\left(\mathbb{E}[f(\hat{S}_M)] - \mathbb{E}[f(S_T)]\right)^2\right] \\
&+ 2\left(\mathbb{E}[f(\hat{S}_M)] - \mathbb{E}[f(S_T)]\right)\mathbb{E}\left[\left(\hat{Y} - \mathbb{E}[f(\hat{S}_M)]\right)\right] \\
&= \mathbb{E}\left[\left(\hat{Y} - \mathbb{E}[f(\hat{S}_M)]\right)^2\right] + \mathbb{E}\left[\left(\mathbb{E}[f(\hat{S}_M)] - \mathbb{E}[f(S_T)]\right)^2\right] \qquad (1)
\end{aligned}$$

Since $\mathbb{E}\left[\hat{Y} - \mathbb{E}[f(\hat{S}_M)]\right] = 0$. The first term on the right-hand side of (1) is the statistical error and the second term is the squared bias. As seen in the class (see [1, Section 3.4]), the statistical error is of order $\frac{1}{N}$ and the squared bias is of order $h^2$. In order to achieve a root mean square error (RMSE) of order $\varepsilon$, (i.e. a MSE of order $\varepsilon^2$), one need to choose $N = O(\varepsilon^{-2})$ and $h = O(\varepsilon)$ (i.e. $M = O(\varepsilon^{-1})$). For this choice of $M$ and $N$, the computational complexity of the naive Monte Carlo estimator $\hat{Y}$ is $N \times M = O(\varepsilon^{-3})$.

The multilevel method presented in [2] reduces this complexity to $O(\varepsilon^{-2}(\log\varepsilon)^2)$.

# 2 Multilevel Monte Carlo (MLMC) Method

The idea of the multilevel method is to approximate the solution of the SDE, on a sequence of grids with smaller and smaller time steps. Namely, consider the geometric sequence of time steps $h_l = \frac{T}{M^l}$,

$l = 0, \ldots, L$. Let $\hat{P}_l$ be the approximation of $f(S_T)$ using a grid with time step $h_l$. From the identity :

$$\mathbb{E}\left[\hat{P}_L\right] = \mathbb{E}\left[\hat{P}_0\right] + \sum_{l=1}^{L} \mathbb{E}\left[\hat{P}_l - \hat{P}_{l-1}\right]$$

We deduce the following multilevel estimator for $\mathbb{E}[\hat{P}_L]$ :

$$\hat{Y}_{ML} = \sum_{l=0}^{L} \hat{Y}_l$$

Where $\hat{Y}_0 = \frac{1}{N_0}\sum_{i=0}^{N_0} \hat{P}_l^i$ ($\hat{P}_l^i$ are i.i.d. realizations of $\hat{P}_l$) and $\hat{Y}_l = \frac{1}{N_l}\sum_{i=1}^{N_l} \hat{P}_l^i - \hat{P}_{l-1}^i$, $((\hat{P}_l^i - \hat{P}_{l-1}^i)_{i\in\{1,\ldots,N_l\}}$ are i.i.d.). The important point here is that $\hat{P}_l^i$ and $\hat{P}_{l-1}^i$ are computed on the approximation to the SDE with same Brownian path but with different time step. The computational cost of the multilevel estimator is of order $\sum_{l=0}^{L} N_l/h_l$. It's variance is $\mathrm{Var}(\hat{Y}_{ML}) = \sum_{l=0}^{L} V_l/N_l$ where $V_l = \mathrm{Var}(\hat{P}_l)$.

## 2.1 Heuristic argument for the optimal choice of $N_l$

We want to minimize the variance of $\hat{Y}_{ML}$, for fixed computational complexity. We want to solve the problem :

$$\begin{array}{ll} \min\limits_{(N_0,\ldots,N_L)\in\mathbb{N}^L} & \displaystyle\sum_{l=0}^{L} \frac{V_l}{N_l} \\ \text{u.c.} & \displaystyle\sum_{l=0}^{L} \frac{N_l}{h_l} = c \end{array}$$

Heuristically, we can treat $l$ like a continuous variable and replace the sums by integrals :

$$\begin{array}{ll} \min\limits_{(N_0,\ldots,N_L)\in\mathbb{N}^L} & \displaystyle\int_0^L \frac{V_l}{N_l} dl \\ \text{u.c.} & \displaystyle\int_0^L \frac{N_l}{h_l} dl = c \end{array}$$

To solve this optimization problem, we introduce the Lagrangian :

$$\mathcal{L}(N,\lambda) = \int_0^L \frac{V_l}{N_l} dl + \lambda\left(\int_0^L \frac{N_l}{h_l} dl - c\right)$$

Let us compute the derivative of $\mathcal{L}$ with respect to $N$, in the direction $n$.

$$\begin{aligned} \mathcal{L}(N+\varepsilon n,\lambda) &= \int_0^L \frac{V_l}{N_l + \varepsilon n_l} dl + \lambda\left(\int_0^L \frac{N_l + \varepsilon n_l}{h_l} dl - c\right) \\ &= \int_0^L \frac{V_l}{N_l} dl + \lambda\left(\int_0^L \frac{N_l}{h_l} dl - c\right) - \int_0^L \varepsilon n_l \frac{V_l}{N_l^2} dl + \int_0^L \varepsilon\lambda \frac{n_l}{h_l} dl + o(\varepsilon) \\ &= \mathcal{L}(N,\lambda) + \varepsilon \int_0^L n_l \left(\frac{\lambda}{h_l} - \frac{V_l}{N_l^2}\right) dl + o(\varepsilon) \end{aligned}$$

Hence, $\partial_N \mathcal{L}(n) = \int_0^L n_l\left(\frac{\lambda}{h_l} - \frac{V_l}{N_l^2}\right) dl$. At the optimum, this quantity should vanish and since $n$ is arbitrary, this implies that $\frac{\lambda}{h_l} = \frac{V_l}{N_l^2}$, or equivalently, $N_l \sim \sqrt{V_l h_l}$.

### 2.2 Error analysis

Let's consider the estimator $\hat{P}_l = f(\hat{S}_{M^l})$. From the theory developed in the class, (see [1, Section 3.3]) we know that there is weak convergence at the speed $O(h_l)$ i.e.

$$\left|\mathbb{E}\left[\hat{P}_l - P\right]\right| = O(h_l) \tag{2}$$

We also know that there is strong convergence ([1, Section 3.2]) at the speed $O(\sqrt{h_l})$ that is :

$$\mathbb{E}\left[|\hat{S}_{M^l} - S_T|^2\right] = O(h_l) \tag{3}$$

Equation (3) allows one to find an estimate for the variance of $\hat{Y}_l$. First we estimate $V_l = \mathrm{Var}(\hat{P}_l - P)$ :

$$\begin{aligned}\mathrm{Var}\left(\hat{P}_l - P\right) &\leqslant \mathbb{E}\left[(\hat{P}_l - P)^2\right] \\ &\leqslant c^2\mathbb{E}\left[|\hat{S}_{M^l} - S_T|^2\right]\end{aligned} \tag{4}$$

Where $c$ is the lipschitz modulus of $f$. To estimate $V_l$, we note that :

$$\begin{aligned}\mathrm{Var}\left(\hat{P}_l - \hat{P}_{l-1}\right) &= \mathrm{Var}\left((\hat{P}_l - P) - (\hat{P}_{l-1} - P)\right) \\ &= \mathrm{Var}\left(\hat{P}_l - P\right) + \mathrm{Var}\left(\hat{P}_{l-1} - P\right) - 2\,\mathrm{Cov}\left((\hat{P}_l - P)(\hat{P}_{l-1} - P),\right) \\ &\leqslant \mathrm{Var}\left(\hat{P}_l - P\right) + \mathrm{Var}\left(\hat{P}_{l-1} - P\right) + 2\sqrt{\mathrm{Var}\left(\hat{P}_l - P\right)}\sqrt{\mathrm{Var}\left(\hat{P}_{l-1} - P\right)} \\ &\leqslant \left(\sqrt{\mathrm{Var}\left(\hat{P}_l - P\right)} + \sqrt{\mathrm{Var}\left(\hat{P}_{l-1} - P\right)}\right)^2\end{aligned} \tag{5}$$

From (3), (4) and (5), we deduce that $V_l = O(h_l)$. Hence, the optimal choice for $N_l$ is asymptotically proportional to $\sqrt{V_l h_l} = O(h_l)$. We want to obtain a MSE of $O(\varepsilon^2)$. First, we deal with the variance. For that, we choose $N_l = O(\varepsilon^{-2} L h_l)$ (which is indeed asymptotically proportional to $h_l$). Since $\mathrm{Var}(\hat{Y}_{ML}) = \sum_{l=0}^{L} V_l/N_l$, we have $\mathrm{Var}(\hat{Y}_{ML}) = O\left(\varepsilon^2 L \sum_{l=0}^{L} h_l/h_l\right) = O(\varepsilon^2 L/L) = O(\varepsilon^2)$. To make the squared bias of order $\varepsilon^2$, we choose $L = -\frac{\log \varepsilon}{\log M} + O(1)$, so that $h_L = T/M^L = Te^{-L\log(M)} = Te^{\log\varepsilon + O(1)} = O(\varepsilon)$. We deduce from (2) the bias $|\mathbb{E}[\hat{P}_L - P]|$ is of order $\varepsilon$, hence the square bias is of order $\varepsilon^2$ and the MSE is of order $\varepsilon^2$. The computational complexity associated to this choice of $L$ and $(N_l)_{l\in\{0,\ldots,L\}}$ is $\sum_{l=0}^{L} N_l/h_l = O\left(\varepsilon^{-2} L \sum_{l=0}^{L} h_l/h_l\right) = O(\varepsilon^{-2}L^2) = O(\varepsilon^{-2}(\log\varepsilon)^2)$.

### 2.3 Numerical algorithm

#### 2.3.1 Heuristic termination criterion

Since we know how to chose each $N_l$, so as to achieve a desired accuracy for the variance, we wish to have a termination criterion that enables us to control the remaining bias. For that purpose, we can use the information given by the correction at level $l$ : $\mathbb{E}[\hat{P}_l - \hat{P}_{l-1}]$. Indeed, because of (2), there exists $c_1$ such that when $l \to \infty$ :

$$\mathbb{E}\left[\hat{P}_l - P\right] \simeq c_1 h_l$$

Moreover, since :

$$\mathbb{E}\left[\hat{P}_l - \hat{P}_{l-1}\right] = \mathbb{E}\left[\hat{P}_l - P\right] - \mathbb{E}\left[\hat{P}_{l-1} - P\right]$$

We have the following approximation :

$$\mathbb{E}\left[\hat{P}_l - \hat{P}_{l-1}\right] \simeq ch_l - ch_{l-1} = c(M-1)h_l$$
$$\simeq (M-1)\mathbb{E}\left[\hat{P}_l - P\right] \tag{6}$$

Because of this, a (heuristic) way to test if the remaining squared bias is less than $\varepsilon^2/2$ is to check if :

$$|\hat{Y}_L| < \frac{1}{\sqrt{2}}(M-1)\varepsilon$$

The same reasoning gives $\mathbb{E}\left[\hat{P}_{l-1} - \hat{P}_{l-2}\right] \simeq c(M^2-M)h_l \simeq M(M-1)\mathbb{E}[\hat{P}_l - P]$. So in order to be more cautious, we can additionally check whether we have : $|\hat{Y}_{L-1}|/M < \frac{1}{\sqrt{2}}(M-1)\varepsilon$. Hence the termination criterion used in the numerical procedure :

$$\max\{|\hat{Y}_{L-1}|/M, |\hat{Y}_L|\} < \frac{1}{\sqrt{2}}(M-1)\varepsilon \tag{7}$$

### 2.3.2 Richardson extrapolation

An other approach is to use Richardson extrapolation. Because, of the approximation $\mathbb{E}[\hat{P}_L - P] \simeq \mathbb{E}[\hat{P}_l - \hat{P}_{l-1}]/(M-1)$, an idea is to subtract the quantity $\hat{Y}_L/(M-1)$ to the combined estimator to form the new estimator :

$$\hat{Y}^R_{ML} = \hat{Y}_{ML} - \hat{Y}_L/(M-1)$$
$$= \frac{M}{M-1}\left(\hat{Y}_0 + \sum_{l=1}^{L}(\hat{Y}_l - \hat{Y}_{l-1}/M)\right)$$

This has the effect of canceling the leading order bias. Using the same ideas as in the previous paragraph, and assuming that the remaining bias is of order $h_l^2$ an appropriate termination criterion for this estimator is :

$$|\hat{Y}_L - \hat{Y}_{L-1}/M| < \frac{1}{\sqrt{2}}(M^2-1)\varepsilon \tag{8}$$

### 2.3.3 Optimal $M$

We mention that there is a way to choose $M$ in order to minimize the computational complexity. For more details, see [2, Section 4.1].

### 2.3.4 Algorithm

Putting everything discussed so far together, the numerical procedure used for the tests is as follows :

**Algorithm 1** Numerical procedure

1: Start with $L = 0$
2: Estimate $V_L$ with an initial set of $N_L = 10^4$ samples.
3: Define optimal $N_l$, $l \in \{0, \cdots, L\}$ using equation (9)
4: Estimate extra samples at each level as needed for new $N_l$.
5: If $L \geqslant 2$, test for convergence using equation (7) or equation (8).
6: If $L < 2$ or it is not converged, set $L = L + 1$ and go to 2.

The equation for the optimal $N_l$ is :

$$N_l = \left\lceil 2\varepsilon^{-2}\sqrt{V_l h_l}\sum_{l=0}^{L}\sqrt{V_l/h_l}\right\rceil \tag{9}$$

This choice makes the variance of the combined estimator less than $\varepsilon^2/2$ because :

$$\begin{aligned}
\mathrm{Var}(\hat{Y}_{ML}) &= \sum_{l=0}^{L}\frac{V_l}{N_l} \\
&= \sum_{l=0}^{L}\frac{V_l}{\left\lceil 2\varepsilon^{-2}\sqrt{V_l h_l}\sum_{l=0}^{L}\sqrt{V_l/h_l}\right\rceil} \\
&\leqslant \sum_{l=0}^{L}\frac{V_l}{2\varepsilon^{-2}\sqrt{V_l h_l}\sum_{l=0}^{L}\sqrt{V_l/h_l}} = \frac{\varepsilon^2}{2}
\end{aligned}$$

# 3 MLMC Numerical Results

We used the algorithm (1) to compute the prices of different options, in different models. In this section, we present our results. For each option, we produced 4 plots :

- The top left plot represents the logarithm base $M$ of the variance of $\hat{P}_l$ and $\hat{P}_{l-1} - \hat{P}_l$ plotted against the level $l$, estimated from a sample of $10^6$ paths.
- The top right plot represents the logarithm base $M$ of the absolute value of the mean of $\hat{P}_l$, $\hat{P}_{l-1} - \hat{P}_l$ and $\hat{Y}_l - \hat{Y}_{l-1}/M$ plotted against the level $l$. It is also based on a sample of $10^6$ paths.
- The bottom left plot represents the optimal values of $N_l$ obtained at the end of the algorithm, for different values of $\varepsilon$. The plot also includes optimal values of $N_l$ when Richardson extrapolation is used.
- The bottom right plot represents the computational complexity, multiplied by $\varepsilon^2$ for the different algorithms, plotted against $\varepsilon$. It includes the multilevel method, the multilevel method with Richardson extrapolation, the classical Monte-Carlo method with Richardson extrapolation and when this was possible (often it was not possible because it was too long to compute), it also includes the classical Monte Carlo method.

## 3.1 Computational complexity

The computational complexity of the multilevel estimator is defined by :

$$C = N_0 + \sum_{l=1}^{L} N_l(M^l + M^{l-1})$$

For the classical estimator, it is defined like this :

$$C^* = \sum_{l=0}^{L} N_l^* M^l$$

Where $N_l^* = 2\varepsilon^{-2}\,\mathrm{Var}(\hat{P}_l)$ so that the variance of the estimator is less than $\frac{\varepsilon^2}{2}$.

## 3.2 Results

For all plots, the options have maturity $T = 1$ and the initial value of the spot is $S_0 = 1$. As in [2], we used the value $M = 4$.

### 3.2.1 European call in the Black-Scholes model

For this plot and the following (except for the last one), we used the Black-Sholes model : $dS_t/S_t = rdt + \sigma dW_t$. With parameters $r = 0.05$ and $\sigma = 0.2$. The discounted payoff of an European option of maturity $T$ with strike $K$ is $g(S_T) = e^{-rT}(S_T - K)_+$. For this plot, we took $K = 1$. This plot is very

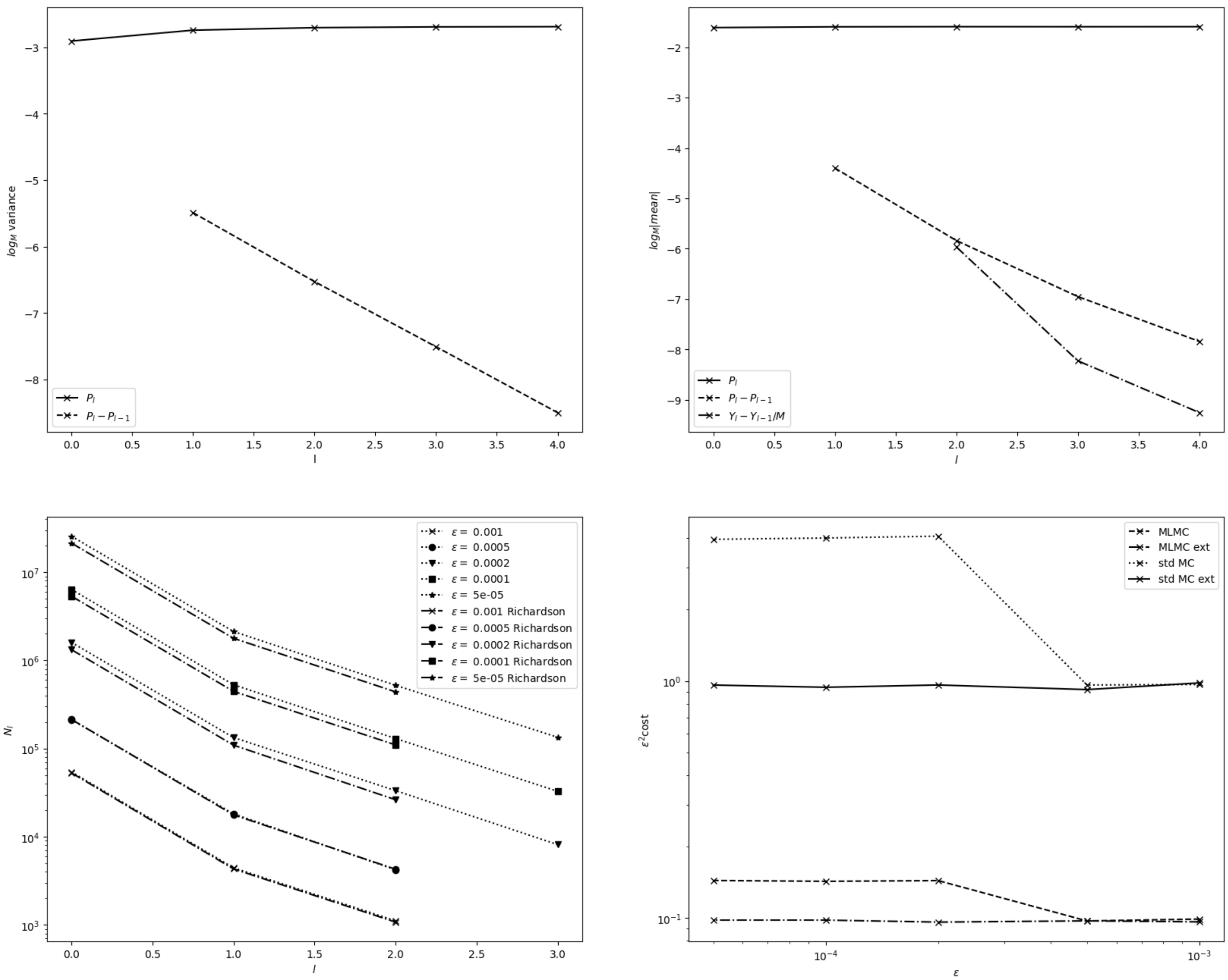


Figure 1: European call in the Black-Scholes model.

similar to figure 2 in [2] and we make the same comments as the authors. In particular, the logarithm of the variance of $\hat{P}_l - \hat{P}_{l-1}$ is a line with slope $-1$, indicating that $V_l$ is indeed $O(h_l)$. The bottom right plot clearly shows the superiority of the multilevel approach compared to the classical Monte Carlo approach. For example, for $\varepsilon \in \{2 \times 10^{-4}, 1 \times 10^{-4}, 5 \times 10^{-5}\}$, the multilevel method is about 20 times faster than the naive approach. With Richardson extrapolation, the multilevel approach is even 30 times less costly. The bottom left plot helps us understand how Richardson extrapolation improves the computational complexity of the multilevel algorithm. By canceling the leading-order bias, it allows the algorithm to achieve the desired accuracy with fewer levels.

### 3.2.2 Asian call in the Black-Scholes model

The discounted payoff of an Asian call with strike $K$ and maturity $T$ is $g(A_T) = e^{-rT}\,(A_T - K)_+$, where $A_T = T^{-1}\int_0^T S_t dt$. We used the following approximation for the integral : $T^{-1}\int_0^T S_t dt \simeq \frac{1}{M+1}\sum_{j=0}^{M} \hat{S}_j$ (Which is different from what was used in the [2]).

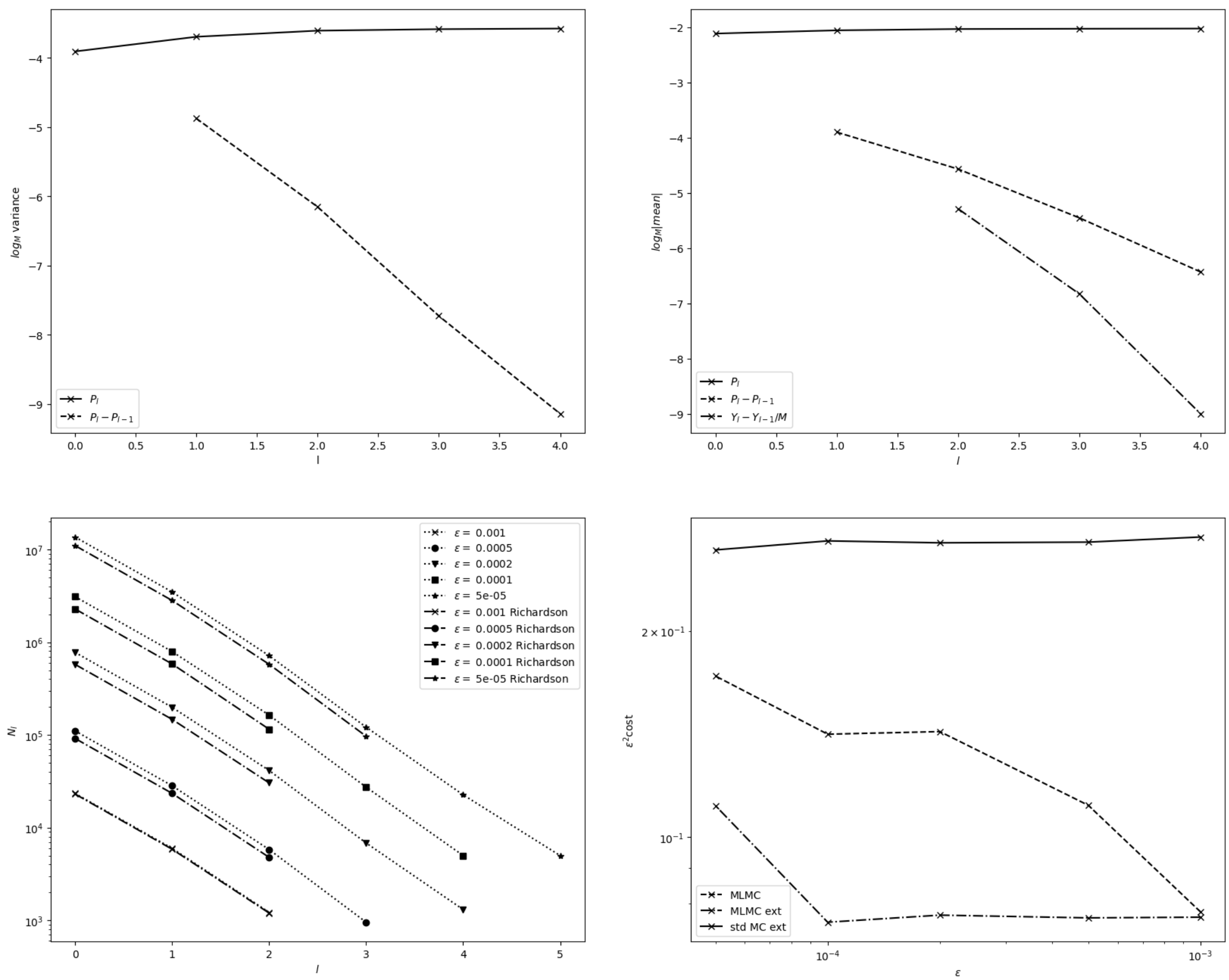


Figure 2: Asian call in the Black-Shcoles model

For the Asian call, our plots are slightly different than the ones in [2]. This is probably because we used a different numerical approximation for the integral. Surprisingly, our classical Monte Carlo with Richardson extrapolation seems quicker than what the authors of [2] obtained. That being said, the MLMC method is still quicker and MLMC method with Richardson is still the best performing method. It is also interesting to see that for our implementation of the Asian call, the multilevel method helped to reduce by two the total number of level needed, in comparison with the vanilla multilevel method, at least for the cases $\varepsilon \in \{2 \times 10^{-4}, 1 \times 10^{-4}, 5 \times 10^{-5}\}$.

### 3.2.3 Lookback option in the Black-Scholes model

A lookback option with maturity $T$ is an option whose discounted payoff is $g((S_t)_{t\in[0,T]}) = e^{-rT}(S_T - \min_{t\in[0,T]} S_t)$. As suggested in [2], we approximated the minimum value of $S_t$ along the path by $\hat{S}_{min,l} = \min_n \hat{S}_n(1-\beta^*\sigma h_l)$ with $\beta^* \simeq 0.5826$. For the lookback options, we did like the authors of

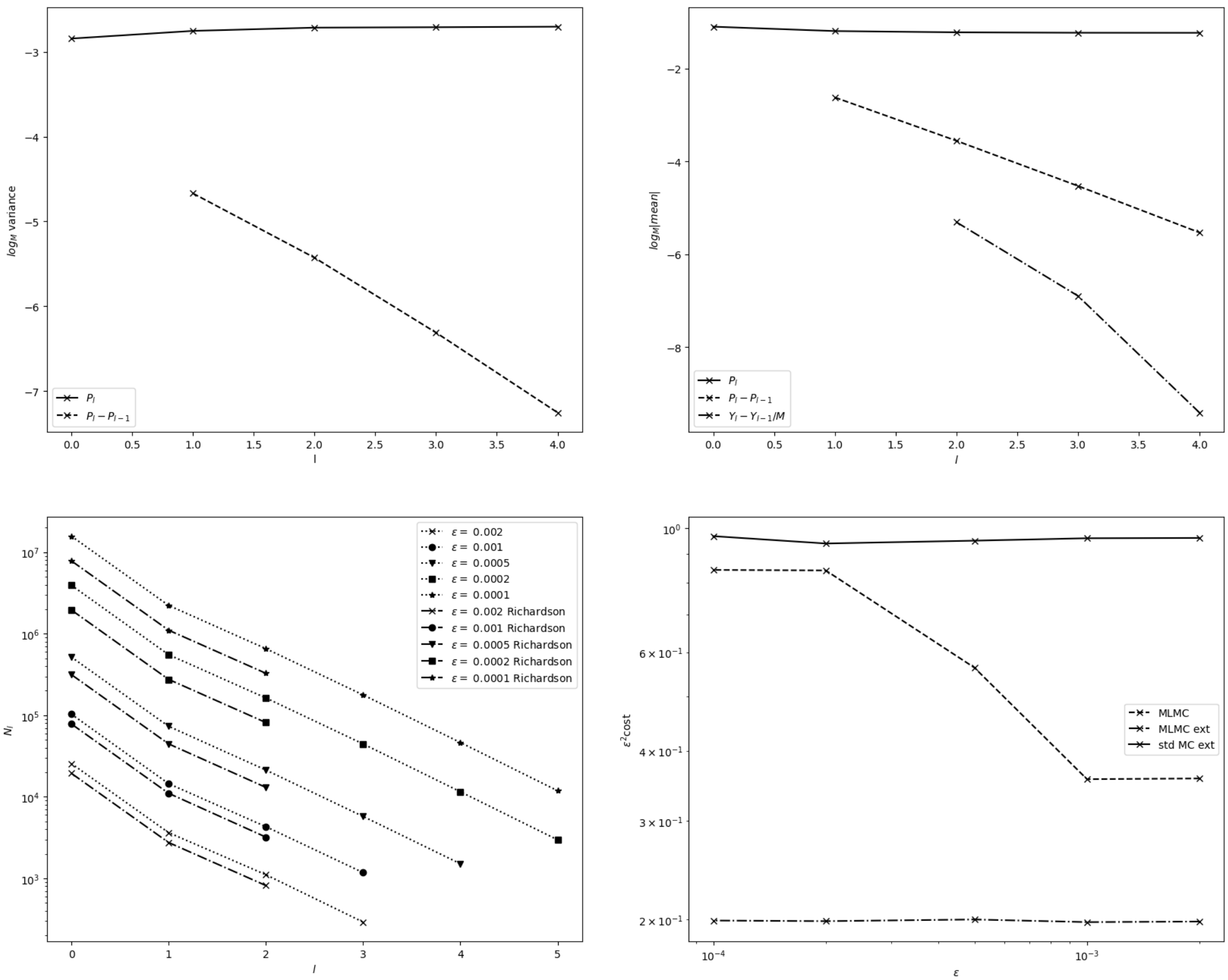


Figure 3: Lookback option in the Black-Scholes model.

[2], so we obtain comparable results and we make the same comments. It is interesting to note that for the smallest values of $\varepsilon$, the standard Monte Carlo method with Richardson extrapolation has a computational complexity close to the one of the standard multilevel method. However, the multilevel method with Richardson extrapolation clearly outperforms the standard multilevel method in this case.

### 3.2.4 Digital option in the Black-Scholes model

A digital option with maturity $T$ is an option whose discounted payoff is : $g(S_T) = e^{-rT}\mathbb{1}_{S_T \geqslant S_0}$. Here again we did just like in [2] and we obtain similar results. We remark that in this case, for $\varepsilon \in \{2\times 10^{-4}, 5\times 10^{-4}\}$, the standard method with Richardson extrapolation has a better computational complexity than the standard multilevel method.

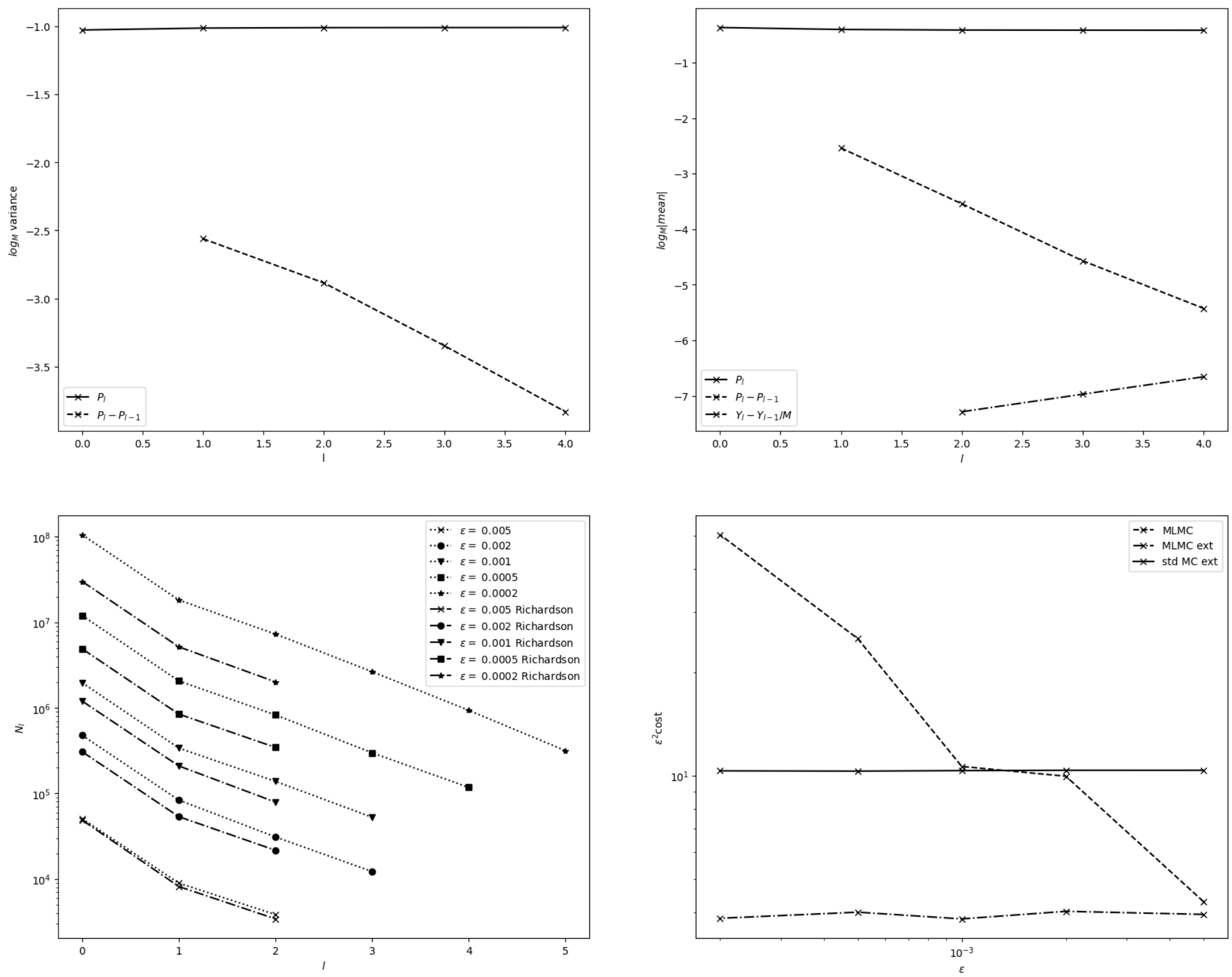


Figure 4: Digital option in the Black-Scholes model

### 3.2.5 European call in the Heston model

The Heston stochastic volatility model is the following :

$$\begin{cases} dS_t/S_t = rdt + \sqrt{V_t}dW_t^1 \\ dV_t = \lambda(\sigma^2 - V_t)dt + \xi\sqrt{V_t}dW_t^2 \end{cases} \tag{10}$$

Where $W^1$ and $W^2$ are two Brownian motions with correlation $\rho$. We used the following set of parameters : $r = 0.05$, $\sigma = 0.2$, $\lambda = 5$, $V_0 = 0.04$, $\xi = 0.25$ and $\rho = -0.5$. Unlike in [2], we approximated the solution to (10) with the Euler scheme for $S$ and $V$.

Although we used a different approach for the discretization of the SDE (10), we obtain results that are quite similar to the ones of [2]. The main difference being the plot of the variance of $\hat{P}_l - \hat{P}_{l-1}$. In our plot, $\log_M\left(\mathrm{Var}(\hat{P}_l - \hat{P}_{l-1})\right)$ is larger for smaller values of $l$. This comes from our choice of discretization of (10).

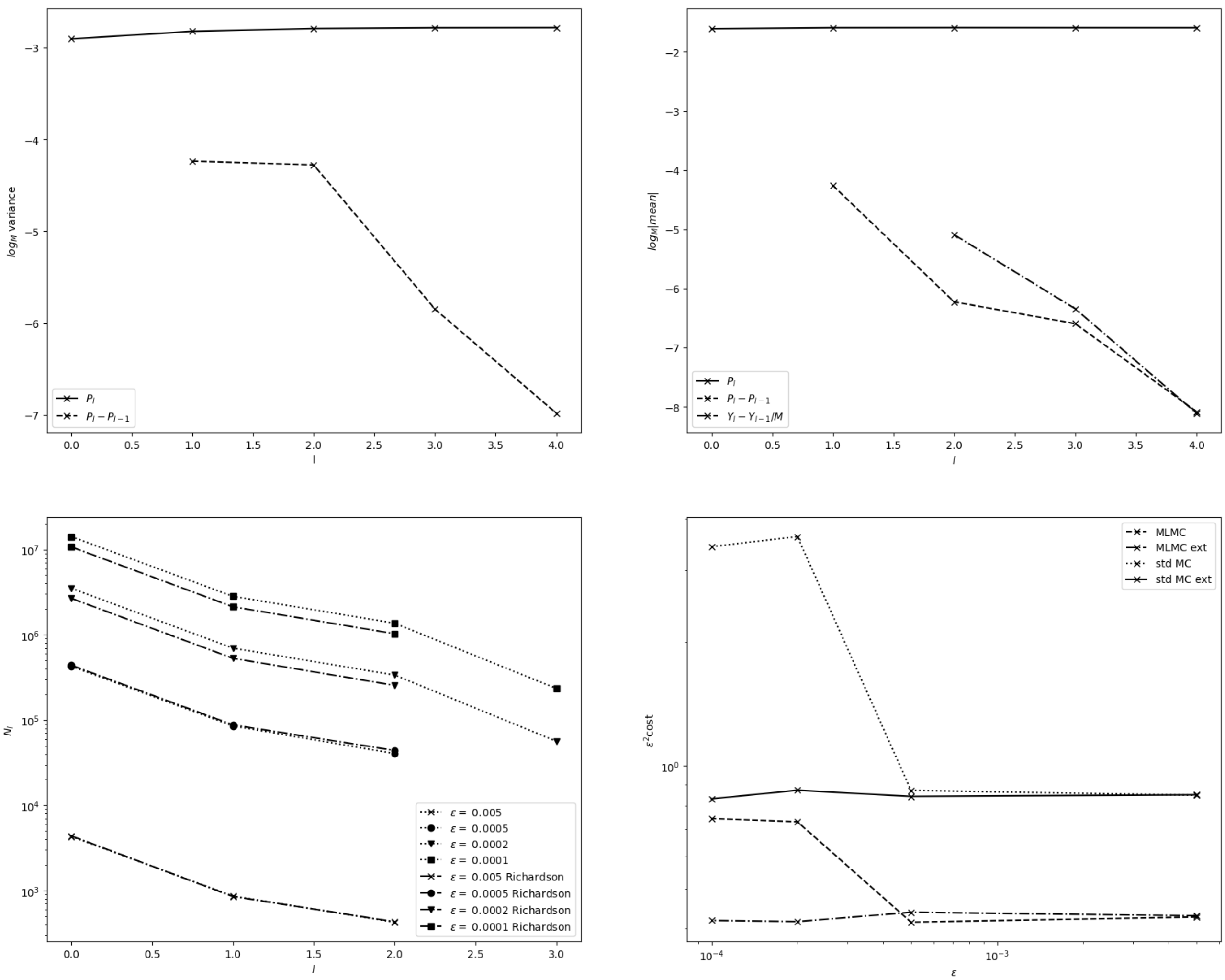


Figure 5: European call in the Heston model

# 4 Finite Difference Method

## 4.1 European options pricing

Under the risk-neutral measure $\mathbb{Q}$, the dynamics of the underlying asset price $S_t$ is given by the following SDE :

$$dS_t = rS_t dt + \sigma(t, S_t) dW_t.$$

Let $G = g(S_T)$ denote the payoff of a European option, which depends on the terminal value of the asset price $S_T$ at maturity $T$. According to the theory of financial derivatives, the option price $p(G)$ at the initial time $t = 0$ is computed as the discounted expected value of the payoff under the risk-neutral measure :

$$p(G) = p(0, S_0) = \mathbb{E}\left[e^{-rT} g(S_T)\right],$$

where $\mathbb{E}$ is the expectation operator under $\mathbb{Q}$.

**Derivation of the governing PDE**

More generally, the time-dependent price function $p(t,x)$, representing the option value at time $t$ given the asset price $S_t = x$, is given by :

$$p(t,x) = \mathbb{E}\left[e^{-r(T-t)}g(S_T) \,\middle|\, S_t = x\right].$$

This expression satisfies the terminal condition at maturity $p(T,x) = g(x)$ for all $x \in \mathbb{R}$. To derive the governing PDE for $p(t,x)$, we assume that $p$ is $\mathscr{C}^{1,2}([0,T[\times\mathbb{R},\mathbb{R})$. That is, applying Itô's lemma to the discounted process $e^{-rt}p(t,S_t)$, we obtain :

$$e^{-rt}p(t,S_t) = p(0,S_0) + \int_0^t e^{-ru}\left(\mathscr{L}p - rp\right)(u,S_u)du + \int_0^t e^{-ru}\sigma(u,S_u)\partial_x p(u,S_u)dW_u, \tag{11}$$

where the operator $\mathscr{L}$ is defined for every $(t,x) \in [0,T[\times\mathbb{R}$ as :

$$\mathscr{L}p(t,x) = \partial_t p(t,x) + rx\partial_x p(t,x) + \frac{1}{2}\sigma(t,x)^2\partial^2_{xx}p(t,x).$$

The term $e^{-rt}p(t,S_t) = \mathbb{E}\left[e^{-rT}g(S_T) \,\middle|\, S_t\right]$ is a martingale under the risk-neutral measure. Consequently, the drift term in (11) must vanish for all $t \in [0,T]$. This leads to the fundamental PDE for the price function $p(t,x)$ :

$$\left(\mathscr{L}p - rp\right)(t,S_t) = 0 \text{ a.s. a.e.} \implies \left(\mathscr{L}p - rp\right)(t,x) = 0 \quad \text{for all } (t,x) \in [0,T[\times\mathbb{R}.$$

In expanded form, the PDE becomes :

$$\begin{cases} \partial_t p(t,x) + rx\partial_x p(t,x) + \frac{1}{2}\sigma(t,x)^2\partial^2_{xx}p(t,x) - rp(t,x) = 0 & \text{in } [0,T) \times \mathbb{R} \\ p(T,x) = g(x) & \text{in } \mathbb{R}. \end{cases}$$

**Solving the PDE using the Finite Difference Method**

In practice, the domain $[0,T] \times \mathbb{R}$ must be truncated to a bounded computational domain due to numerical constraints. We therefore restrict our attention to the bounded domain $[0,T] \times (\alpha,\beta)$, where $\alpha < \beta$ are real numbers chosen such that the majority of the solution's behavior of interest lies within this interval. The PDE is then reformulated as :

$$\begin{cases} \mathscr{L}p - rp = 0 & \text{in } [0,T) \times (\alpha,\beta) \\ p = g & \text{in } \{T\} \times (\alpha,\beta) \\ p = \varphi & \text{in } [0,T) \times \{\alpha,\beta\} \end{cases}$$

where the boundary condition $p = \varphi$ at $x = \alpha$ and $x = \beta$ depends on the specific problem.

To approximate the solution numerically, we discretize the domain $[0,T] \times (\alpha,\beta)$ into a finite grid. Let $m > 0$ and $\ell \geqslant 0$ be given integers defining the number of time steps and spatial steps, respectively. We define the mesh as :

$$\left\{(t_n, x_i) := (n\Delta t, \alpha + i\Delta x) : 0 \leqslant n \leqslant m,\ 0 \leqslant i \leqslant \ell + 1\right\},$$

where $\Delta t = \frac{T}{m}$ and $\Delta x = \frac{\beta-\alpha}{l+1}$. At each grid point $(t_n, x_i)$, we approximate the value of $p(t,x)$ using a discrete approximation $(p_i^n)_{0\leqslant i\leqslant \ell+1, 0\leqslant n\leqslant m}$, where $p_i^n \approx p(t_n, x_i)$.

Using finite difference approximations for the derivatives, we have the following approximation :

$$\begin{aligned} rx_i\partial_x p(t_n,x_i) &+ \frac{1}{2}\sigma(t_n,x_i)^2\partial^2_{xx}p(t_n,x_i) - rp(t_n,x_i) \\ &\approx rx_i\left(\frac{p^n_{i+1}-p^n_{i-1}}{2\Delta x}\right) + \frac{1}{2}\sigma(t_n,x_i)^2\left(\frac{p^n_{i+1}+p^n_{i-1}-2p^n_i}{\Delta x^2}\right) - rp^n_i \\ &\approx \left(\frac{\sigma(t_n,x_i)^2}{2\Delta x^2} - \frac{rx_i}{2\Delta x}\right)p^n_{i-1} - \left(\frac{\sigma(t_n,x_i)^2}{\Delta x^2} + r\right)p^n_i + \left(\frac{\sigma(t_n,x_i)^2}{2\Delta x^2} + \frac{rx_i}{2\Delta x}\right)p^n_{i+1}. \end{aligned}$$

As $p(t,\cdot) = \varphi$ on $\{\alpha,\beta\}$, we approximate

$$\left(\left(\frac{\sigma(t_n,x_i)^2}{2\Delta x^2} - \frac{rx_i}{2\Delta x}\right)p^n_{i-1} - \left(\frac{\sigma(t_n,x_i)^2}{\Delta x^2} + r\right)p^n_i + \left(\frac{\sigma(t_n,x_i)^2}{2\Delta x^2} + \frac{rx_i}{2\Delta x}\right)p^n_{i+1}\right)_{1\leqslant i\leqslant l} \approx A^n p^n + f^n,$$

where $p^n = (p^n_i)_{1\leqslant i\leqslant l} \in \mathbb{R}^\ell$, $A^n \in \mathbb{R}^{\ell\times\ell}$ is a tridiagonal matrix given by

$$A^n_{i,i-1} = \frac{\sigma(t_n,x_i)^2}{2\Delta x^2} - \frac{rx_i}{2\Delta x}, \quad A^n_{i,i} = -\frac{\sigma(t_n,x_i)^2}{\Delta x^2} - r, \quad A^n_{i,i+1} = \frac{\sigma(t_n,x_i)^2}{2\Delta x^2} + \frac{rx_i}{2\Delta x},$$

and $f^n \in \mathbb{R}^\ell$ is such that $f^n_i = 0$ for $i \in \{2,\dots,l-1\}$ and

$$f^n_1 = \left(\frac{\sigma(t_n,x_1)^2}{2\Delta x^2} - \frac{rx_1}{2\Delta x}\right)\varphi(\alpha), \quad f^n_\ell = \left(\frac{\sigma(t_n,x_\ell)^2}{2\Delta x^2} + \frac{rx_\ell}{2\Delta x}\right)\varphi(\beta)$$

Finally, as constructed in the course (see [1, Section 7.1]), given $\theta \in [0,1]$, set $p^m = (g(x_i))_{1\leqslant i\leqslant l}$ and using the theta-scheme, we solve recursively for $n \in \{m-1,\dots,0\}$ :

$$\frac{p^{n+1}-p^n}{\Delta t} + \theta\left(A^n p^n + f^n\right) + (1-\theta)\left(A^{n+1}p^{n+1} + f^{n+1}\right) = 0.$$

In other words, solve

$$\left(I - \theta\Delta t A^n\right)p^n = \left(I + (1-\theta)\Delta t A^{n+1}\right)p^{n+1} + \Delta t\left(\theta f^n + (1-\theta)f^{n+1}\right). \tag{12}$$

## 4.2 FDM in the Black-Scholes model

In this section, we focus on the special case of the Black-Scholes model to investigate the performance of the FDM in solving the associated PDE. Specifically, we study European call and put options, as this setting provides a closed-form analytical solution. This allows for a detailed convergence analysis, enabling us to quantify the accuracy of the numerical approximations.

In the Black-Scholes framework, the volatility function is constant and proportional to the asset price, such that for all $t \in [0,T]$, $\sigma(t,S_t) = \sigma S_t$, where $\sigma > 0$ is the constant volatility parameter.

### European call option

For a European call option with payoff $G = (S_T - K)_+$, the boundary conditions for the FDM in this special case are as follows $p(t,0) = 0$ and $p(t,x) \sim x - Ke^{-r(T-t)}$ as $x \to \infty$ for every $t \in [0,T]$. Therefore, we set $\varphi(\alpha) = 0$ and $\varphi(\beta) = \beta - Ke^{-r(T-t)}$ and for the numerical experiments, the computational domain is chosen such that $\alpha = 0$ and $\beta = S_{\max}$, with $S_{\max} > 0$. The following parameter values are used : $S_0 = 1$, $r = 0.5$, $\sigma = 0.2$, $K = 1$, $T = 1$ and $S_{\max} = 2S_0$. These parameters define the domain and boundary conditions for the finite difference algorithm, implemented as described in (12).

### European put option

For a European put option, the payoff is given by $G = (K - S_T)_+$. Similar boundary conditions and domain truncations are applied : $p(t,x) \sim Ke^{-r(T-t)}$ as $x \to 0$ and $p(t,x) \sim 0$ as $x \to \infty$ for all $t \in [0,T]$, then, we set $\varphi(\alpha) = Ke^{-r(T-t)}$, $\varphi(\beta) = 0$ and the parameter setup remains consistent with the call option case.

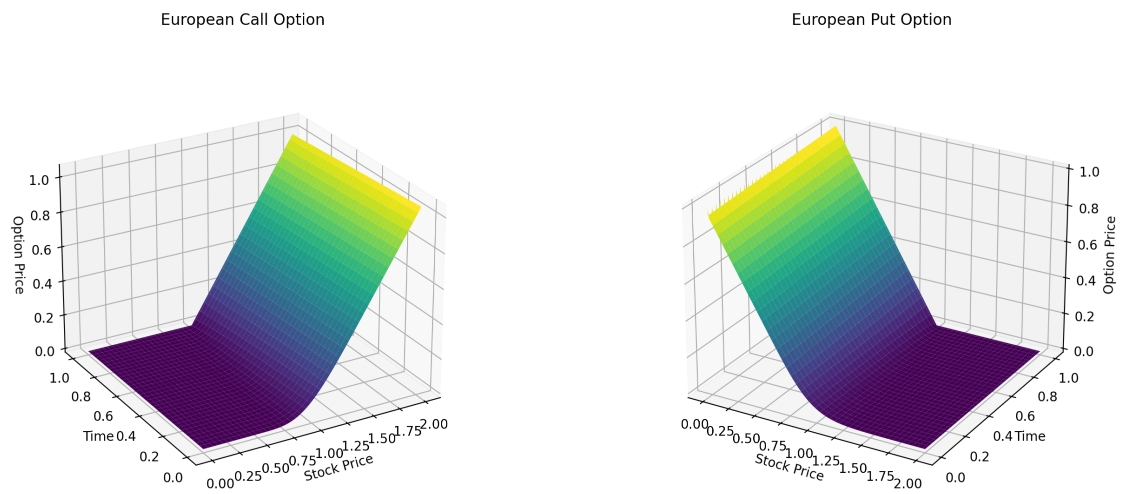


Figure 6: FDM using $\ell = 400$ and $m = 100$

### Convergence analysis

We now investigate the convergence properties of the theta-scheme, as presented in the course [1, Section 7.2]. The theoretical foundations of the theta-scheme, including its consistency and stability, have been rigorously established, guaranteeing its convergence under the standard criteria. To further validate these theoretical results, we provide numerical results specifically focused on the European call option using the implicit scheme ($\theta = 1$) and the Crank-Nicholson scheme ($\theta = 0.5$) :

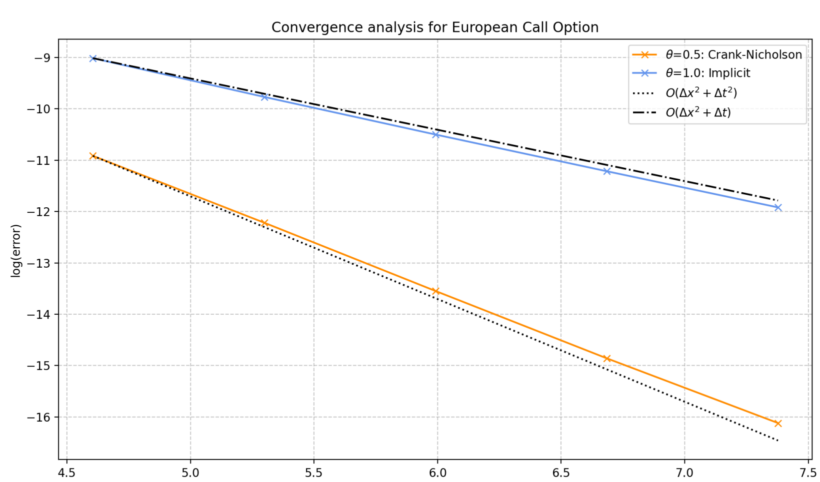


Figure 7: Convergence analysis at time $t = 0$

The convergence analysis shown in Figure 7 illustrates the error behavior of the FDM for a European call option at $t = 0$, as the spatial and temporal resolutions increase. Specifically:

- For the Crank–Nicholson scheme ($\theta = 0.5$), the numerical solution converges at the expected rate of $O(\Delta x^2 + \Delta t^2)$, as evidenced by the steeper slope in the log-log plot. This is consistent with the second-order accuracy of the scheme in both space and time.
- For the implicit scheme ($\theta = 1.0$), the convergence rate matches the theoretical rate of $O(\Delta x^2 + \Delta t)$, reflecting second-order accuracy in space and first-order accuracy in time.

The reference slopes in the figure confirm that the numerical experiments align well with theoretical expectations.

## References

[1] Julien Claisse. Monte carlo and finite difference methods with applications in finance, 2024. Lecture Notes.

[2] Michael B Giles. Multilevel monte carlo path simulation. *Operations research*, 56(3):607–617, 2008.