Multilevel Monte Carlo Path Simulation

Mike Giles

mike.giles@maths.ox.ac.uk

Oxford University Mathematical Institute

Oxford-Man Institute of Quantitative Finance

University of Trier, Sept 9th, 2009

SDEs in Finance

In computational finance, stochastic differential equations are used to model the behaviour of

- stocks
- interest rates
- exchange rates
- weather
- electricity/gas demand
- crude oil prices
- . . .

The stochastic term accounts for the uncertainty of unpredictable day-to-day events.

SDEs in Finance

These models are then used to calculate “fair” prices for a huge range of financial options:

- an option to sell a stock portfolio at a specific price in 2 years time
- an option to buy aviation fuel at a specific price in 6 months time
- an option to sell US dollars at a specific exchange rate in 3 years time

In most cases, the buyer of the financial option is trying to reduce their risk.

SDEs in Finance

Examples:

- Geometric Brownian motion (Black-Scholes model for stock prices)

$$\mathrm{d}S = r\, S\, \mathrm{d}t + \sigma\, S\, \mathrm{d}W$$

- Cox-Ingersoll-Ross model (interest rates)

$$\mathrm{d}r = \alpha(b-r)\,\mathrm{d}t + \sigma\,\sqrt{r}\,\mathrm{d}W$$

- Heston stochastic volatility model (stock prices)

$$\begin{aligned}\mathrm{d}S &= r\, S\, \mathrm{d}t + \sqrt{V}\, S\, \mathrm{d}W_1 \\ \mathrm{d}V &= \lambda\,(\sigma^2 - V)\,\mathrm{d}t + \xi\,\sqrt{V}\,\mathrm{d}W_2\end{aligned}$$

  with correlation ρ between $\mathrm{d}W_1$ and $\mathrm{d}W_2$

Generic Problem

Stochastic differential equation with general drift and volatility terms: SDE with general drift and volatility terms:

$$\mathrm{d}S(t) = a(S,t)\,\mathrm{d}t + b(S,t)\,\mathrm{d}W(t)$$

$W(t)$ is a Wiener variable with the properties that for any $q<r<s<t$, $W(t)-W(s)$ is Normally distributed with mean 0 and variance $t-s$, independent of $W(r)-W(q)$.

In many finance applications, we want to compute the expected value of an option dependent on the terminal state

$$P \equiv f(S(T))$$

with a uniform Lipschitz bound,

$$|f(U) - f(V)| \ \leq c\ \|U - V\|\,, \quad \forall\, U,V.$$

Standard MC Approach

Euler discretisation with timestep h:

$$\widehat{S}_{n+1} = \widehat{S}_n + a(\widehat{S}_n, t_n)\, h + b(\widehat{S}_n, t_n)\, \Delta W_n$$

where ΔW_n are Normal with mean 0, variance h.

Simplest estimator for expected payoff is an average of N independent path simulations:

$$\widehat{Y} = N^{-1} \sum_{i=1}^{N} \widehat{P}^{(i)}$$

where $\widehat{P} \equiv f(\widehat{S}_{T/h})$ is an approximation to $P \equiv f(S(T))$ for a given Brownian path $W(t)$.

Standard MC Approach

The mean square error is defined as

$$
\begin{aligned}
\mathbb{E}\left[\left(\widehat{Y} - \mathbb{E}[P]\right)^2\right] &= \mathbb{E}\left[\left(\widehat{Y} - \mathbb{E}[\widehat{P}] \;+\; \mathbb{E}[\widehat{P}] - \mathbb{E}[P]\right)^2\right] \\
&= \mathbb{E}\left[(\widehat{Y} - \mathbb{E}[\widehat{P}])^2\right] + (\mathbb{E}[\widehat{P}] - \mathbb{E}[P])^2 \\
&= N^{-1}\mathbb{V}[\widehat{P}] + \left(\mathbb{E}[\widehat{P}] - \mathbb{E}[P]\right)^2
\end{aligned}
$$

- first term is due to variance of estimator
- second term is due to bias due to finite timestep
  – weak convergence

Standard MC Approach

Weak convergence:

- error in the expected value, $\mathbb{E}[\widehat{P}] - \mathbb{E}[P]$
- most important error in most applications
- $O(h)$ for the Euler discretisation

Strong convergence:

- error in path approximation

$$\sqrt{\mathbb{E}\left[\left\|\widehat{S}_{T/h} - S(T)\right\|^2\right]} \quad \text{or} \quad \sqrt{\mathbb{E}\left[\max_{0<t<T}\left\|\widehat{S}(t) - S(t)\right\|^2\right]}$$

- usually not relevant, but important for multilevel method
- $O(h^{1/2})$ for the Euler discretisation

Standard MC Approach

Combined mean-square-error is $O(N^{-1} + h^2)$.

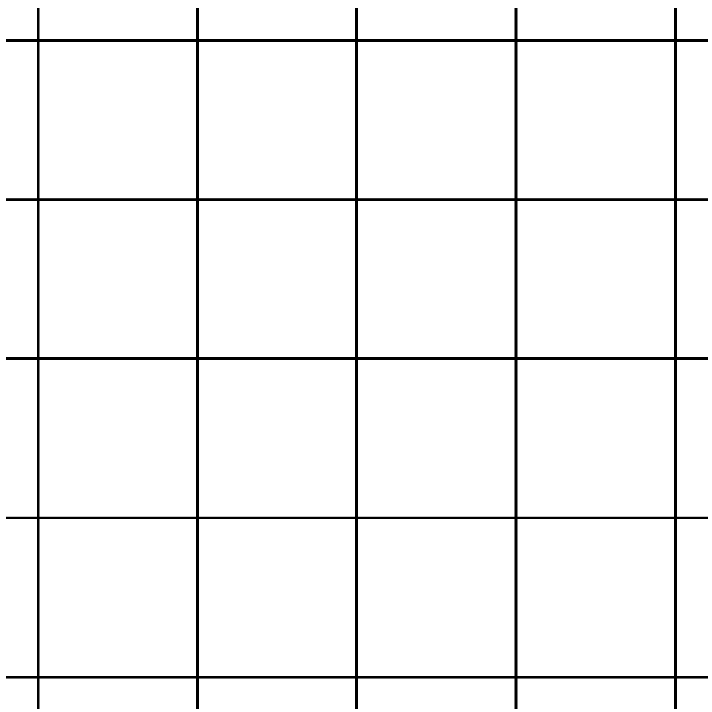
To make this equal to ε^2 requires

$$N = O(\varepsilon^{-2}), \quad h = O(\varepsilon) \quad \Longrightarrow \quad \text{cost} = O(N\,h^{-1}) = O(\varepsilon^{-3})$$

Aim is to improve this cost to $O\left(\varepsilon^{-2}(\log \varepsilon)^2\right)$, by combining simulations with different numbers of timesteps – same accuracy as finest calculations, but at a much lower computational cost.

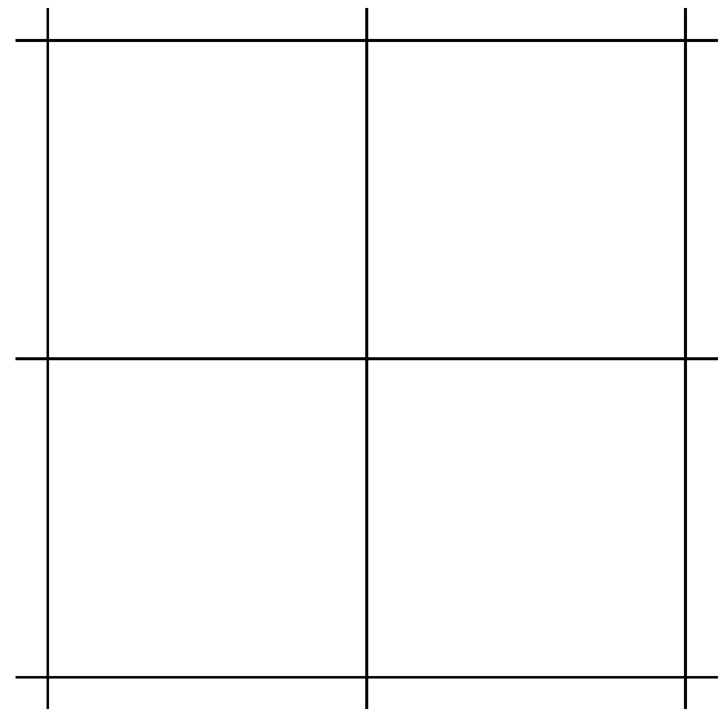
Multigrid

A powerful technique for solving PDE discretisations:



Fine grid

more accurate
more expensive



Coarse grid

less accurate
less expensive

Multigrid

Multigrid combines calculations on a nested sequence of grids to get the accuracy of the finest grid at a much lower computational cost.

We use a similar idea in Monte Carlo path calculations, combining simulations with different numbers of timesteps – same accuracy as finest calculations, but at a much lower computational cost.

Multilevel MC Approach

Consider multiple sets of simulations with different timesteps $h_l = 2^{-l}\,T,\ l=0,1,\ldots,L$, and payoff $\widehat{P}_l$

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

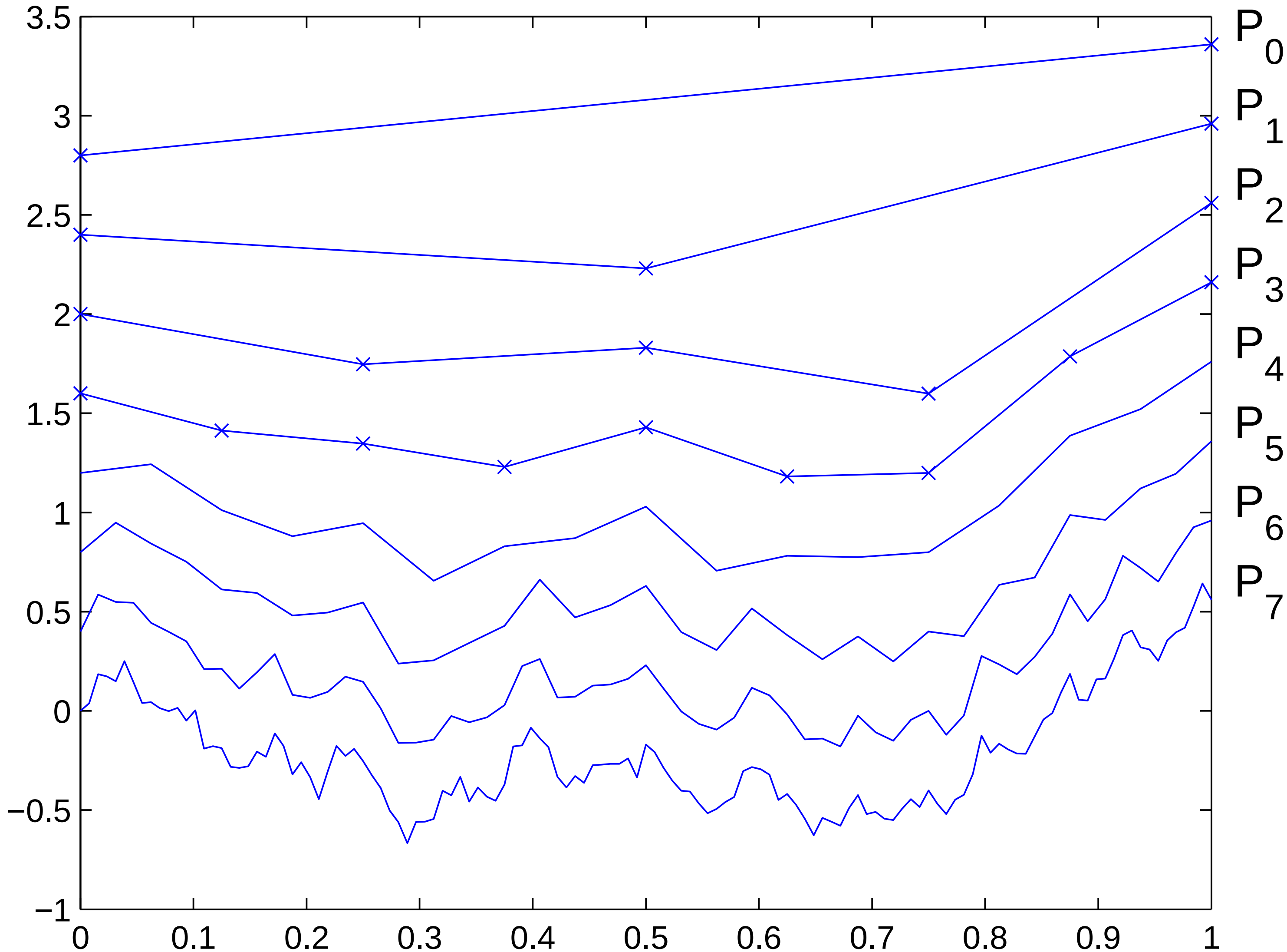
Expected value is same – aim is to reduce variance of estimator for a fixed computational cost.

Key point: approximate $\mathbb{E}[\widehat{P}_l - \widehat{P}_{l-1}]$ using N_l simulations with $\widehat{P}_l$ and $\widehat{P}_{l-1}$ obtained using <u>same</u> Brownian path.

$$\widehat{Y}_l = N_l^{-1} \sum_{i=1}^{N_l} \left(\widehat{P}_l^{(i)} - \widehat{P}_{l-1}^{(i)} \right)$$

Multilevel MC Approach

Discrete Brownian path at different levels



Multilevel MC Approach

- each level adds more detail to Brownian path
- $\mathbb{E}[\widehat{P}_l - \widehat{P}_{l-1}]$ reflects impact of that extra detail on the payoff
- different timescales handled by different levels – similar to different wavelengths being handled by different grids in multigrid

Multilevel MC Approach

Using independent paths for each level, the variance of the combined estimator is

$$\mathbb{V}\left[\sum_{l=0}^{L} \widehat{Y}_l\right] = \sum_{l=0}^{L} N_l^{-1} V_l, \qquad V_l \equiv \mathbb{V}[\widehat{P}_l - \widehat{P}_{l-1}],$$

and the computational cost is proportional to $\displaystyle\sum_{l=0}^{L} N_l\, h_l^{-1}$.

Hence, the variance is minimised for a fixed computational cost by choosing N_l to be proportional to $\sqrt{V_l\, h_l}$.

The constant of proportionality can be chosen so that the combined variance is $O(\varepsilon^2)$.

Multilevel MC Approach

For the Euler discretisation and the Lipschitz payoff function

$$\mathbb{V}[\widehat{P}_l - P] = O(h_l) \quad \Longrightarrow \quad \mathbb{V}[\widehat{P}_l - \widehat{P}_{l-1}] = O(h_l)$$

and the optimal N_l is asymptotically proportional to h_l.

To make the combined variance $O(\varepsilon^2)$ requires

$$N_l = O(\varepsilon^{-2} L\, h_l).$$

To make the bias $O(\varepsilon)$ requires

$$L = \log_2 \varepsilon^{-1} + O(1) \quad \Longrightarrow \quad h_L = O(\varepsilon).$$

Hence, we obtain an $O(\varepsilon^2)$ MSE for a computational cost which is $O(\varepsilon^{-2} L^2) = O(\varepsilon^{-2} (\log \varepsilon)^2)$.

Results

Geometric Brownian motion:

$$\mathrm{d}S = r\,S\,\mathrm{d}t + \sigma\,S\,\mathrm{d}W, \qquad 0 < t < T,$$

$T\!=\!1,\;\; S(0)\!=\!1,\;\; r\!=\!0.05,\;\; \sigma\!=\!0.2$

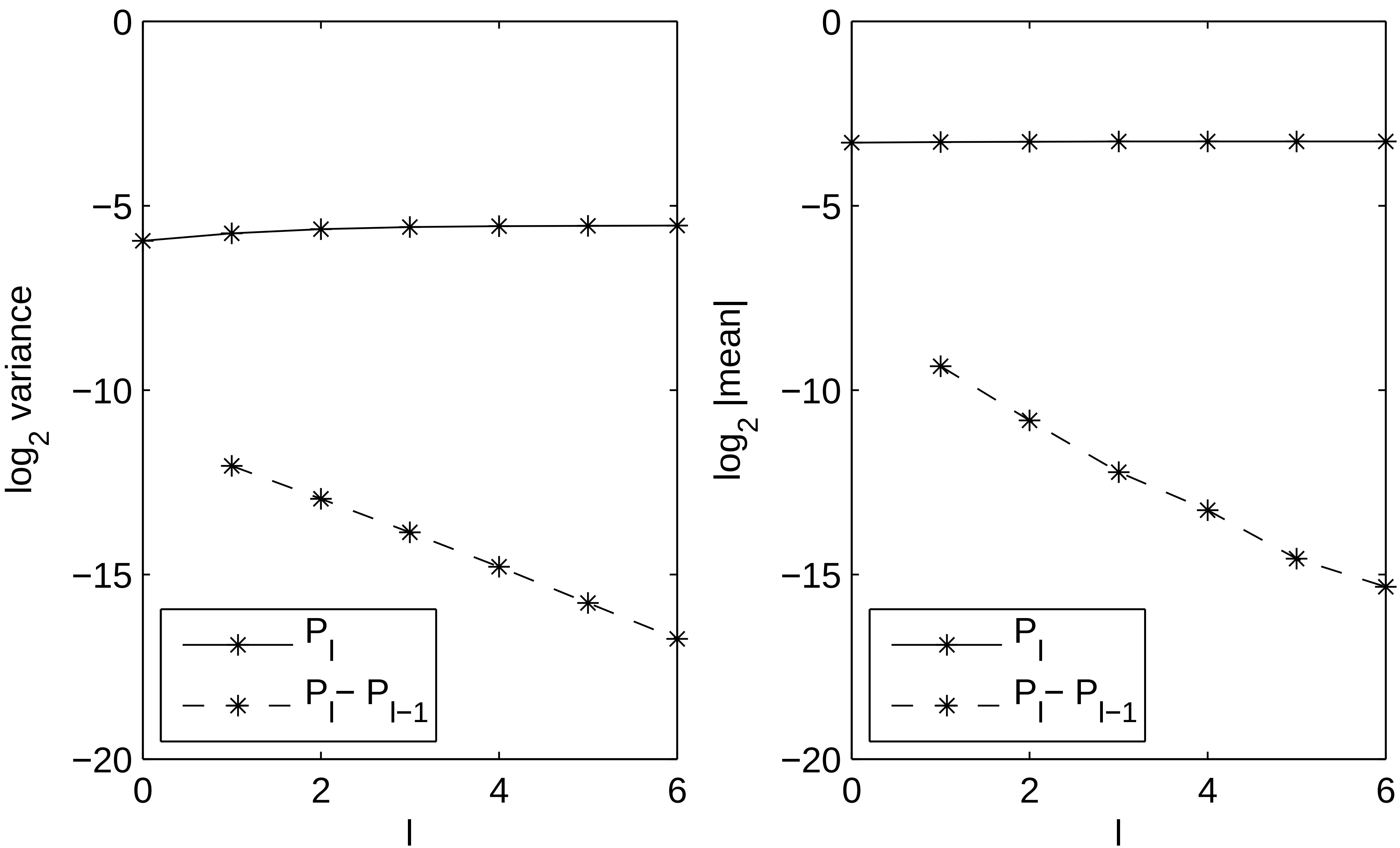
European call option with discounted payoff

$$\exp(-rT)\;\max(S(T)\!-\!K, 0)$$

with $K\!=\!1$.

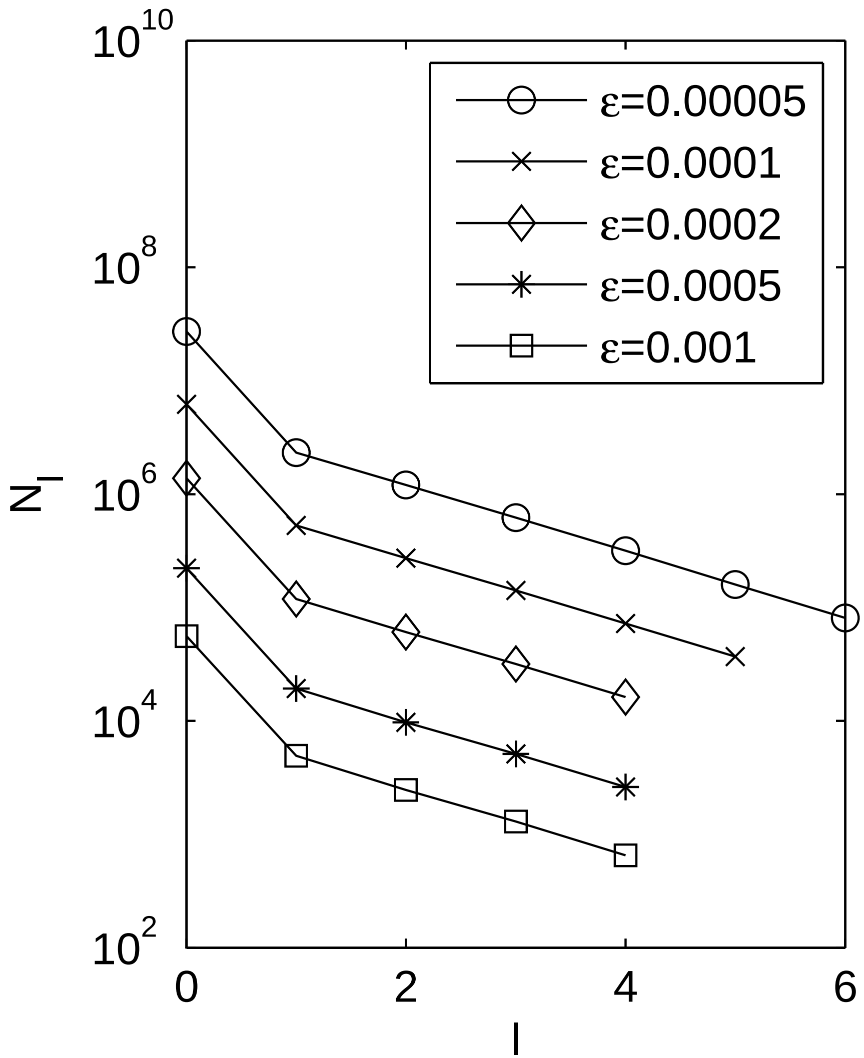
MLMC Results

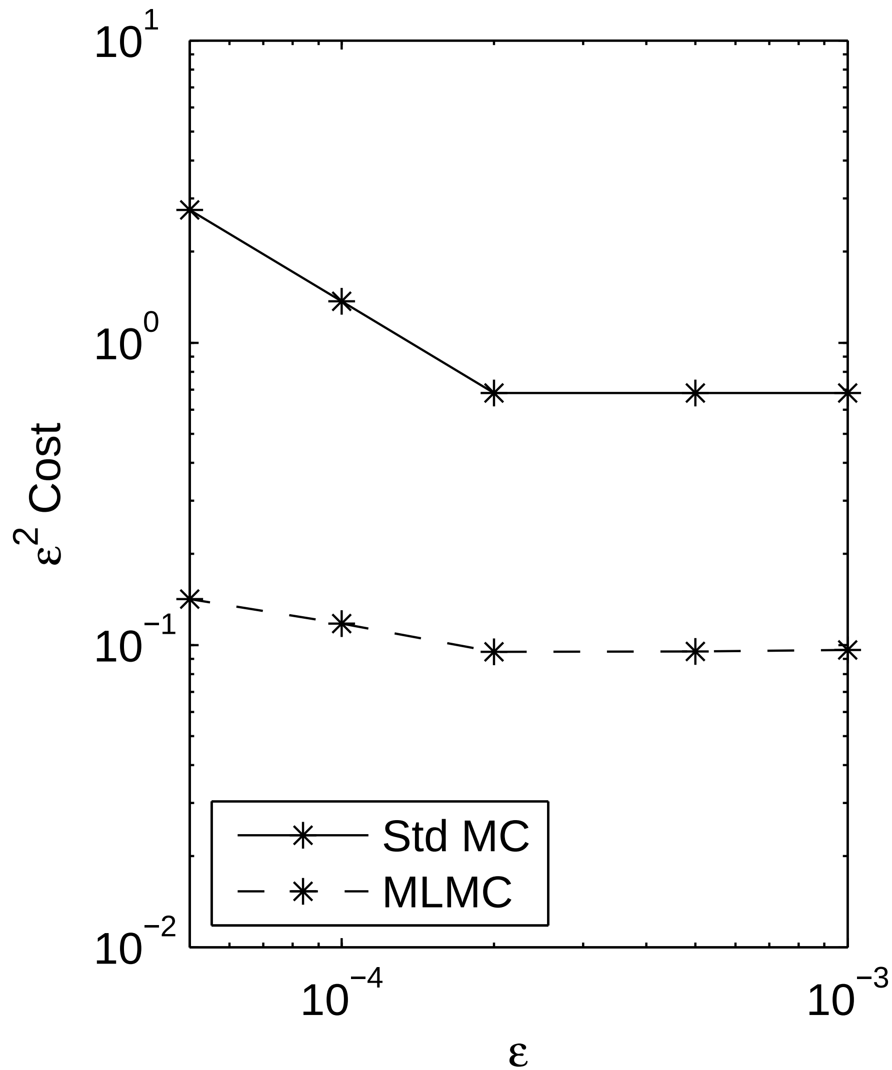
GBM: European call, $\exp(-rT)\ \max(S(T)-K,0)$



MLMC Results

GBM: European call, $\exp(-rT)\ \max(S(T)\!-\!K,0)$





Multilevel MC Approach

Theorem: *Let* P *be a functional of the solution of a stochastic o.d.e., and* $\widehat{P}_l$ *the discrete approximation using a timestep* $h_l = M^{-l}\, T$*.*

If *there exist independent estimators* $\widehat{Y}_l$ *based on* N_l *Monte Carlo samples, and positive constants* $\alpha \geq \frac{1}{2}, \beta, c_1, c_2, c_3$ *such that*

i) $\left| \mathbb{E}[\widehat{P}_l - P] \right| \leq c_1\, h_l^\alpha$

ii) $\mathbb{E}[\widehat{Y}_l] = \begin{cases} \mathbb{E}[\widehat{P}_0], & l = 0 \\ \mathbb{E}[\widehat{P}_l - \widehat{P}_{l-1}], & l > 0 \end{cases}$

iii) $\mathbb{V}[\widehat{Y}_l] \leq c_2\, N_l^{-1} h_l^\beta$

iv) C_l*, the computational complexity of* $\widehat{Y}_l$*, is bounded by*

$$C_l \leq c_3\, N_l\, h_l^{-1}$$

Multilevel MC Approach

then *there exists a positive constant* c_4 *such that for any* $\varepsilon < e^{-1}$ *there are values* L *and* N_l *for which the multi-level estimator*

$$\widehat{Y} = \sum_{l=0}^{L} \widehat{Y}_l,$$

has Mean Square Error $MSE \equiv E\left[\left(\widehat{Y} - E[P]\right)^2\right] < \varepsilon^2$

with a computational complexity C *with bound*

$$C \leq \begin{cases} c_4\, \varepsilon^{-2}, & \beta > 1, \\ c_4\, \varepsilon^{-2} (\log \varepsilon)^2, & \beta = 1, \\ c_4\, \varepsilon^{-2-(1-\beta)/\alpha}, & 0 < \beta < 1. \end{cases}$$

Milstein Scheme

The theorem suggests use of Milstein approximation
– better strong convergence, same weak convergence

Generic scalar SDE:

$$\mathrm{d}S(t) = a(S,t)\;\mathrm{d}t + b(S,t)\;\mathrm{d}W(t), \quad 0\!<\!t\!<\!T.$$

Milstein scheme:

$$\widehat{S}_{n+1} = \widehat{S}_n + a\,h + b\,\Delta W_n + \tfrac{1}{2}\,b'\,b\left((\Delta W_n)^2 - h\right).$$

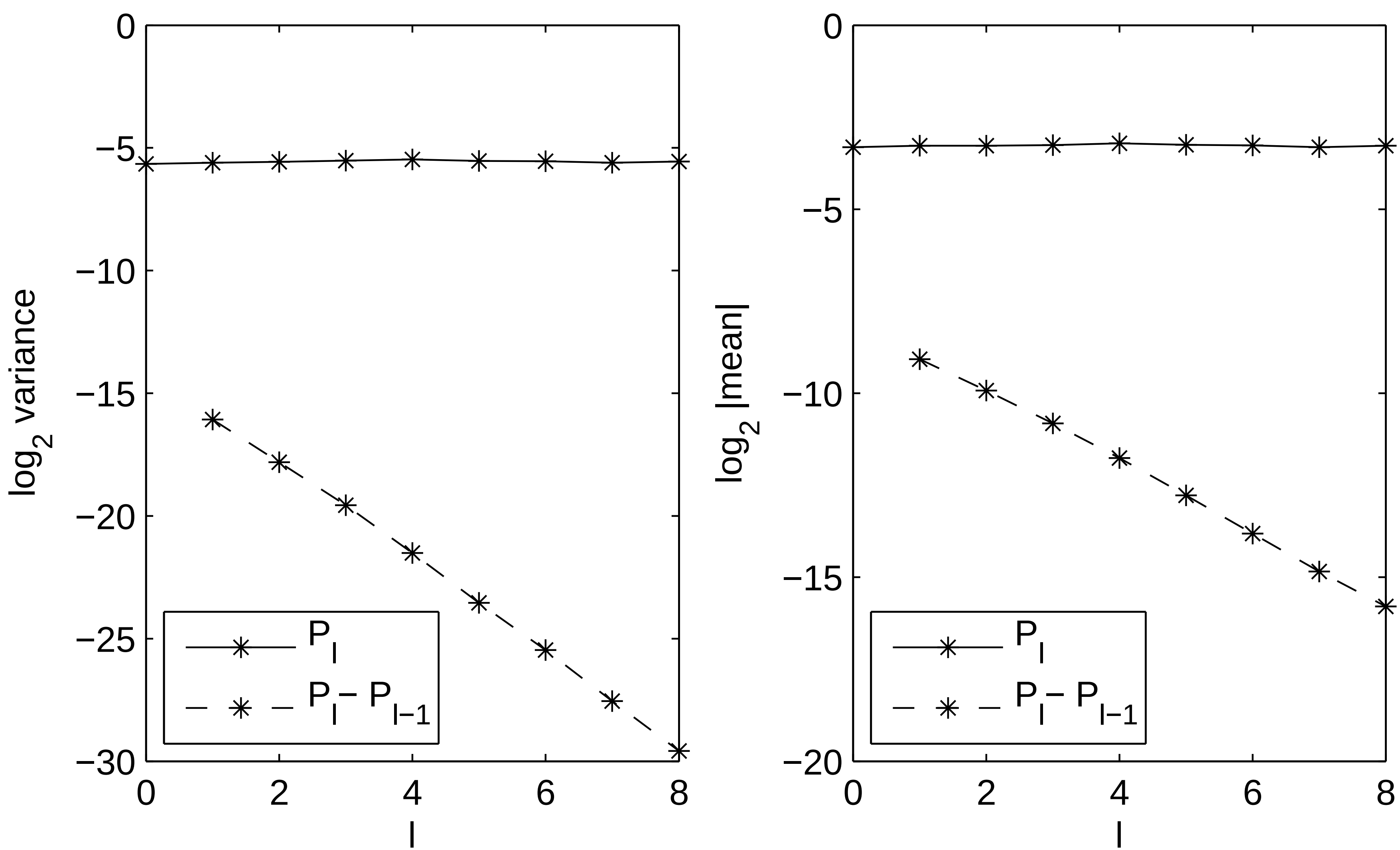
Milstein Scheme

In scalar case:

- $O(h)$ strong convergence
- $O(\varepsilon^{-2})$ complexity for Lipschitz payoffs – trivial
- $O(\varepsilon^{-2})$ complexity for more complex cases using carefully constructed estimators based on Brownian interpolation or extrapolation
    - digital, with discontinuous payoff
    - Asian, based on average
    - lookback and barrier, based on min/max

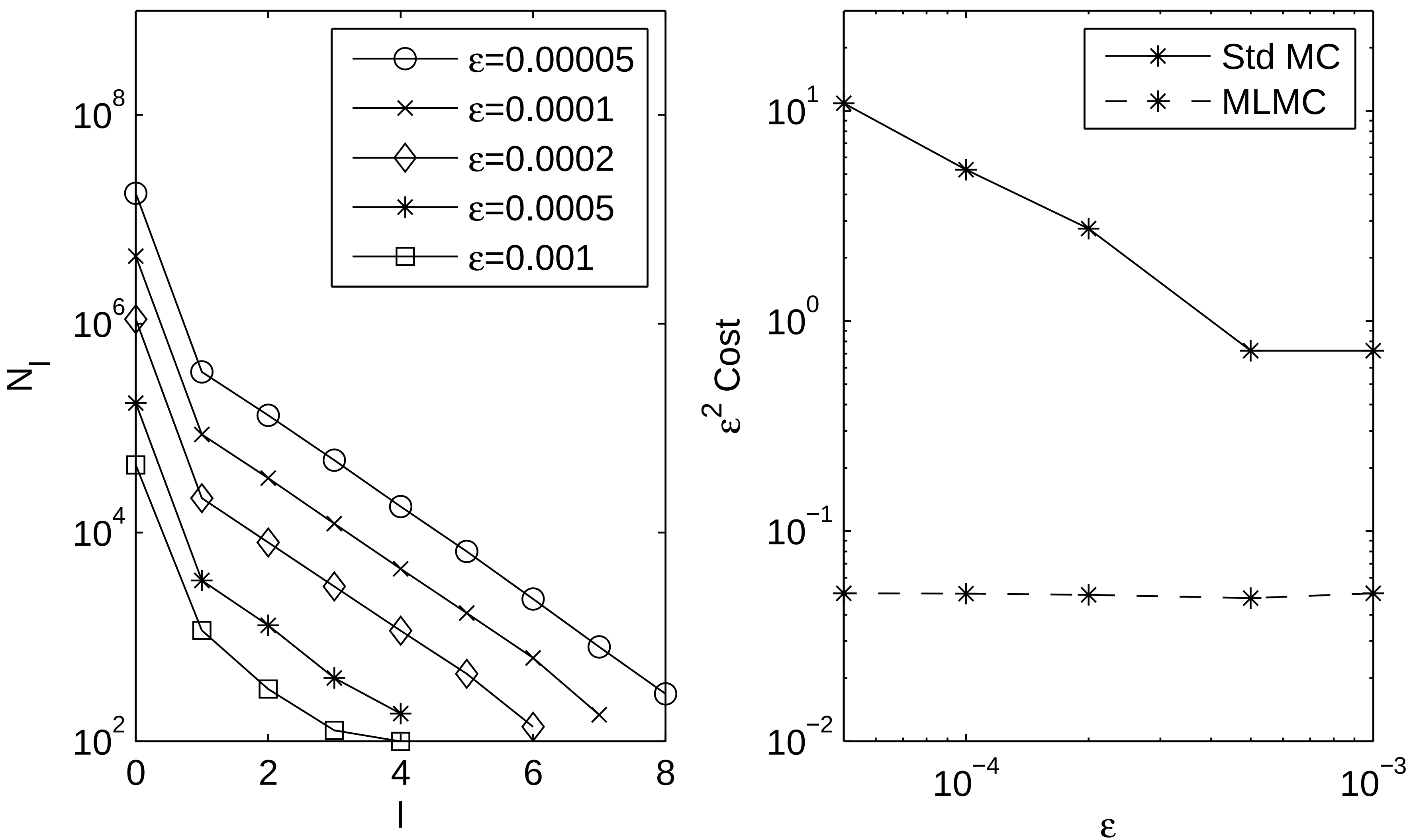
MLMC Results

GBM: European call, $\exp(-rT)\ \max(S(T)-K, 0)$



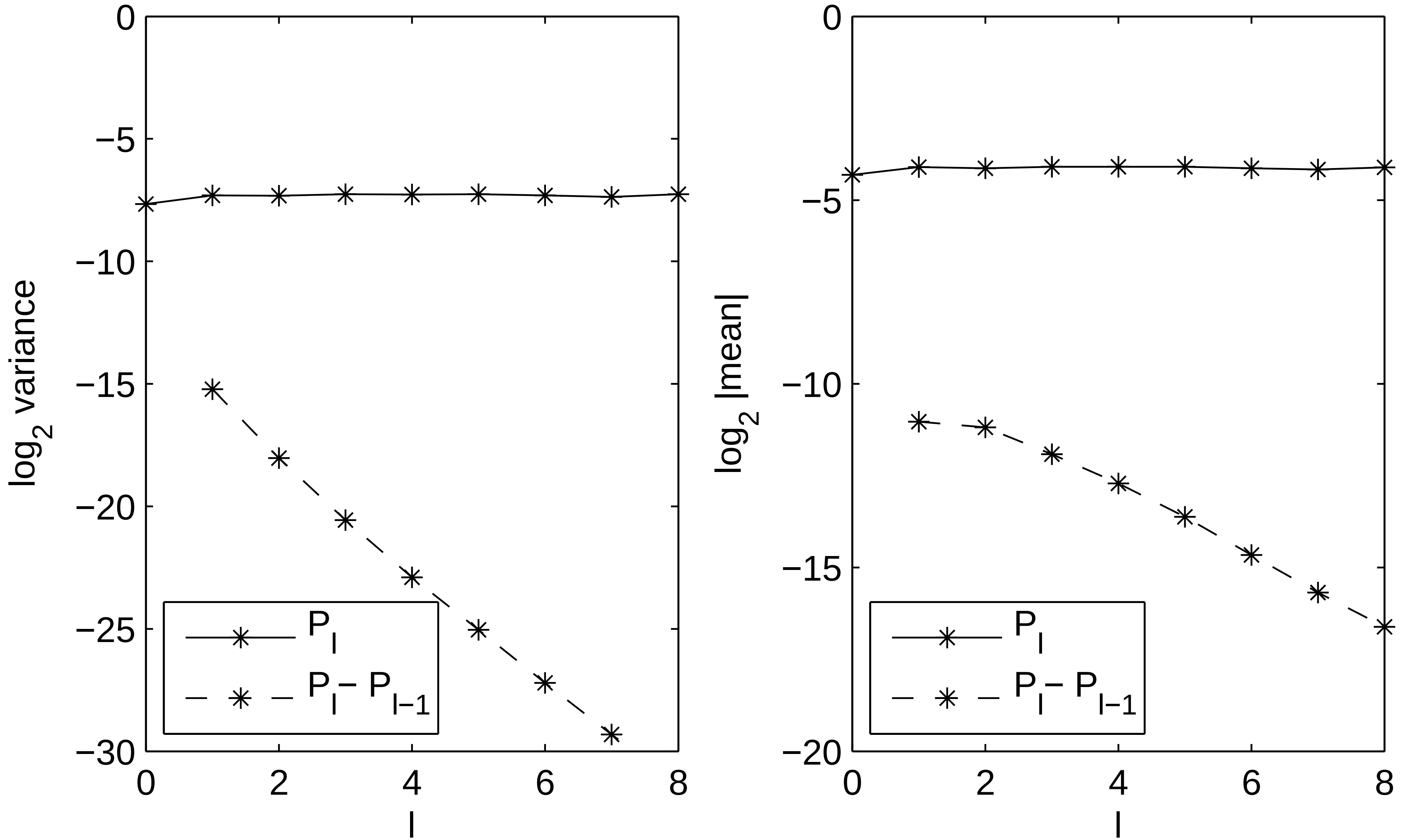
MLMC Results

GBM: European call, $\exp(-rT)\ \max(S(T)-K,0)$



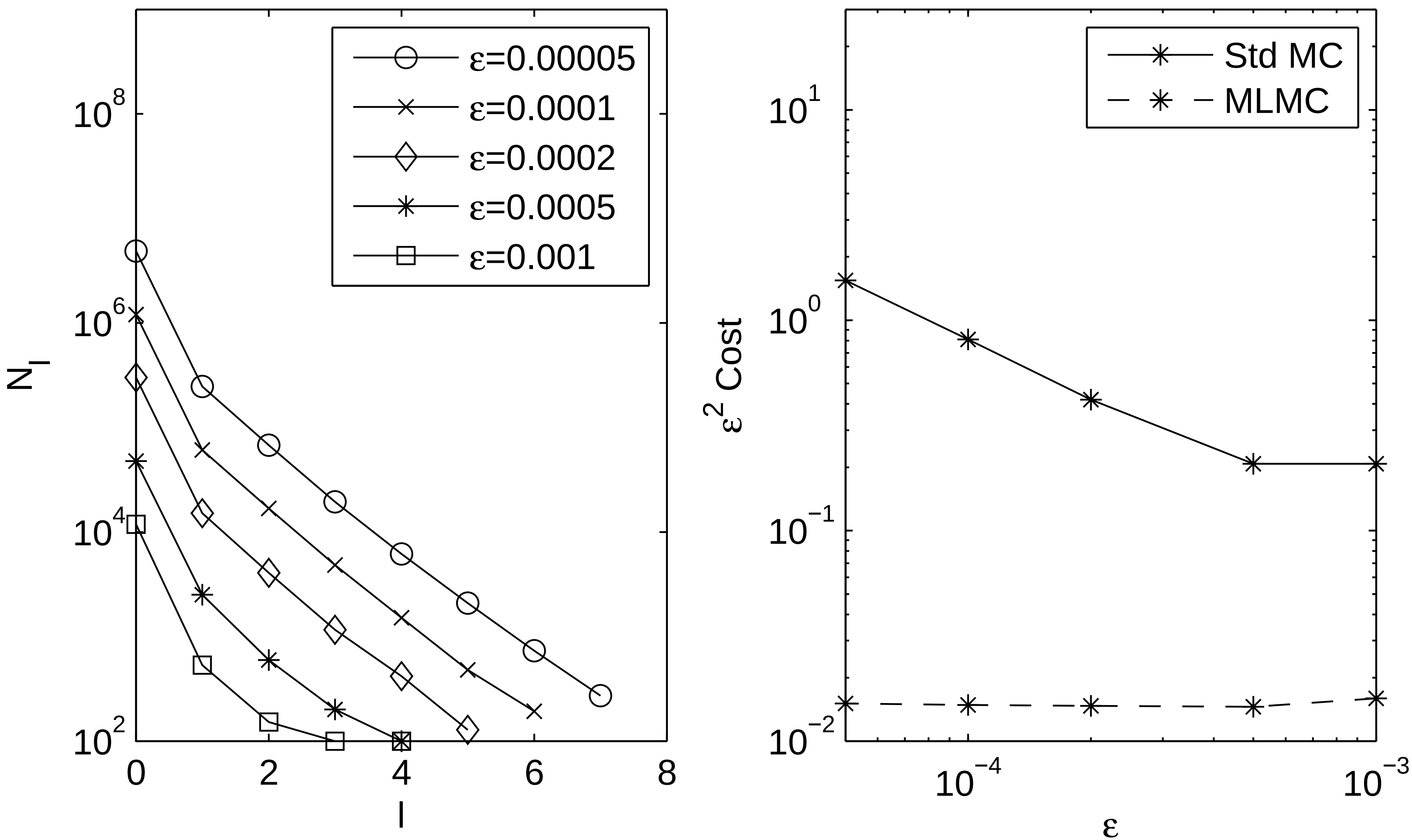
MLMC Results

GBM: Asian option, $\exp(-rT)\ \max(T^{-1}\int_0^T S(t)\,\mathrm{d}t - 1, 0)$



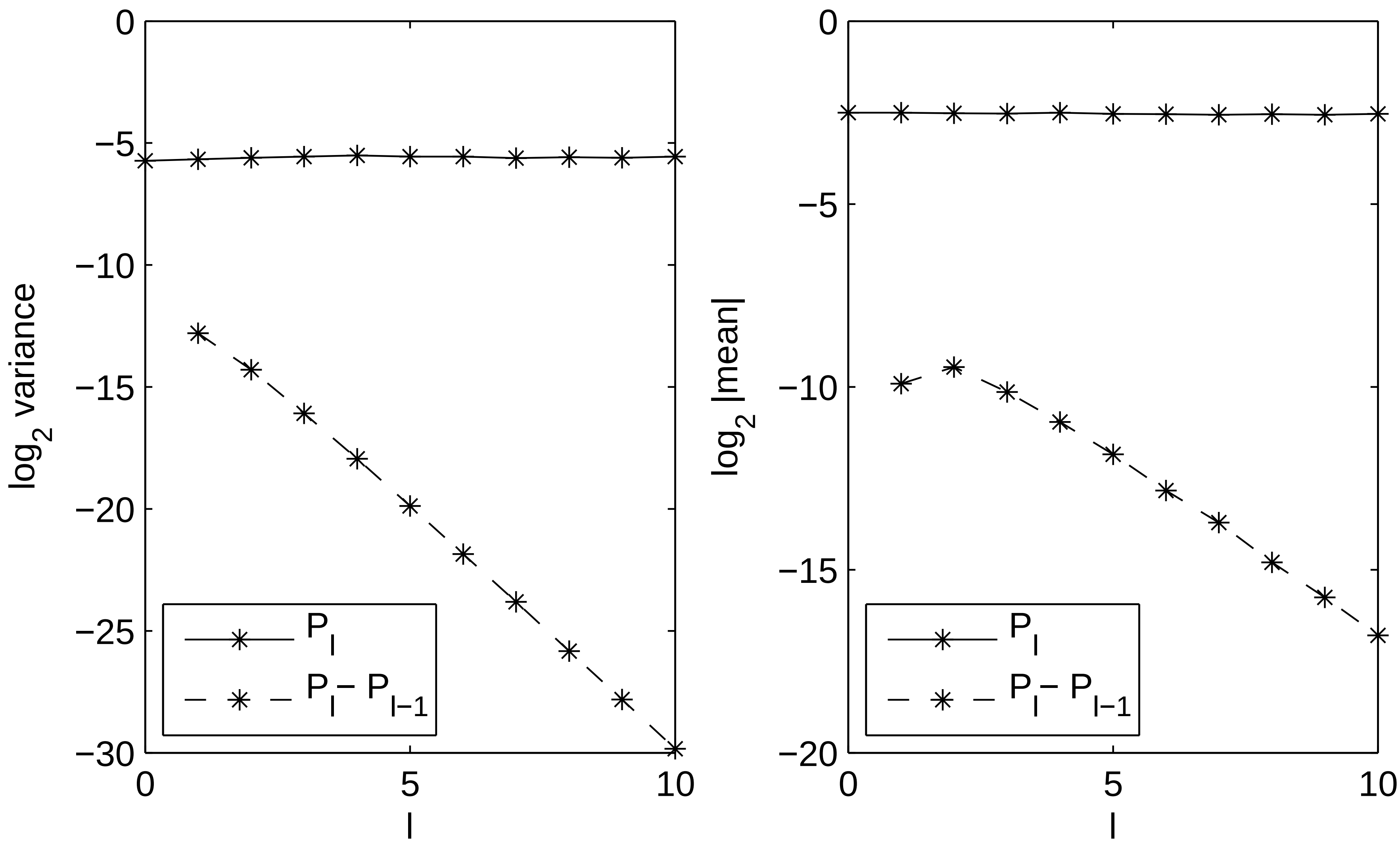
MLMC Results

GBM: Asian option, $\exp(-rT)\ \max(T^{-1}\int_0^T S(t)\,\mathrm{d}t - 1, 0)$



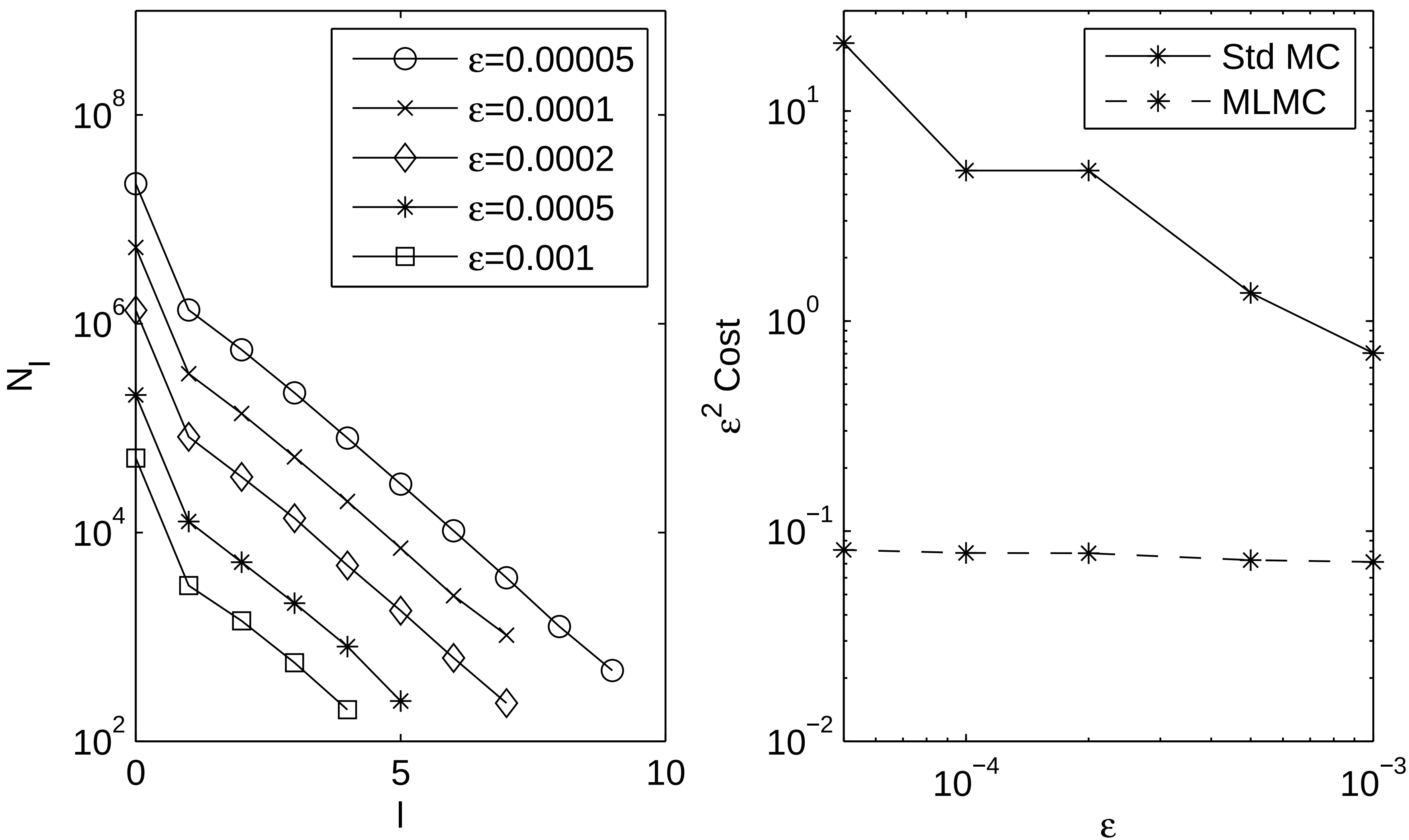
MLMC Results

GBM: lookback option, $\exp(-rT)\,(S(T) - \min_{0<t<T} S(t))$



MLMC Results

GBM: lookback option, $\exp(-rT)\,(S(T) - \min_{0<t<T} S(t))$



Milstein Scheme

Generic vector SDE:

$$\mathrm{d}S(t) = a(S,t)\,\mathrm{d}t + b(S,t)\,\mathrm{d}W(t), \quad 0<t<T,$$

with correlation matrix $\Omega(S,t)$ between elements of $\mathrm{d}W(t)$.

Milstein scheme:

$$\begin{aligned}\widehat{S}_{i,n+1} &= \widehat{S}_{i,n} + a_i\, h + b_{ij}\,\Delta W_{j,n} \\ &\quad + \tfrac{1}{2}\,\frac{\partial b_{ij}}{\partial S_l}\, b_{lk}\left(\Delta W_{j,n}\,\Delta W_{k,n} - h\,\Omega_{jk} - A_{jk,n}\right)\end{aligned}$$

with implied summation, and Lévy areas defined as

$$A_{jk,n} = \int_{t_n}^{t_{n+1}} (W_j(t)-W_j(t_n))\;\mathrm{d}W_k - (W_k(t)-W_k(t_n))\;\mathrm{d}W_j .$$

Milstein Scheme

In vector case:

- $O(h)$ strong convergence if Lévy areas are simulated correctly – expensive
- $O(h^{1/2})$ strong convergence in general if Lévy areas are omitted, except if a certain commutativity condition is satisfied (useful for a number of real cases)
- Lipschitz payoffs can be handled well using antithetic variables
- Other cases may require approximate simulation of Lévy areas

Results

Heston model:

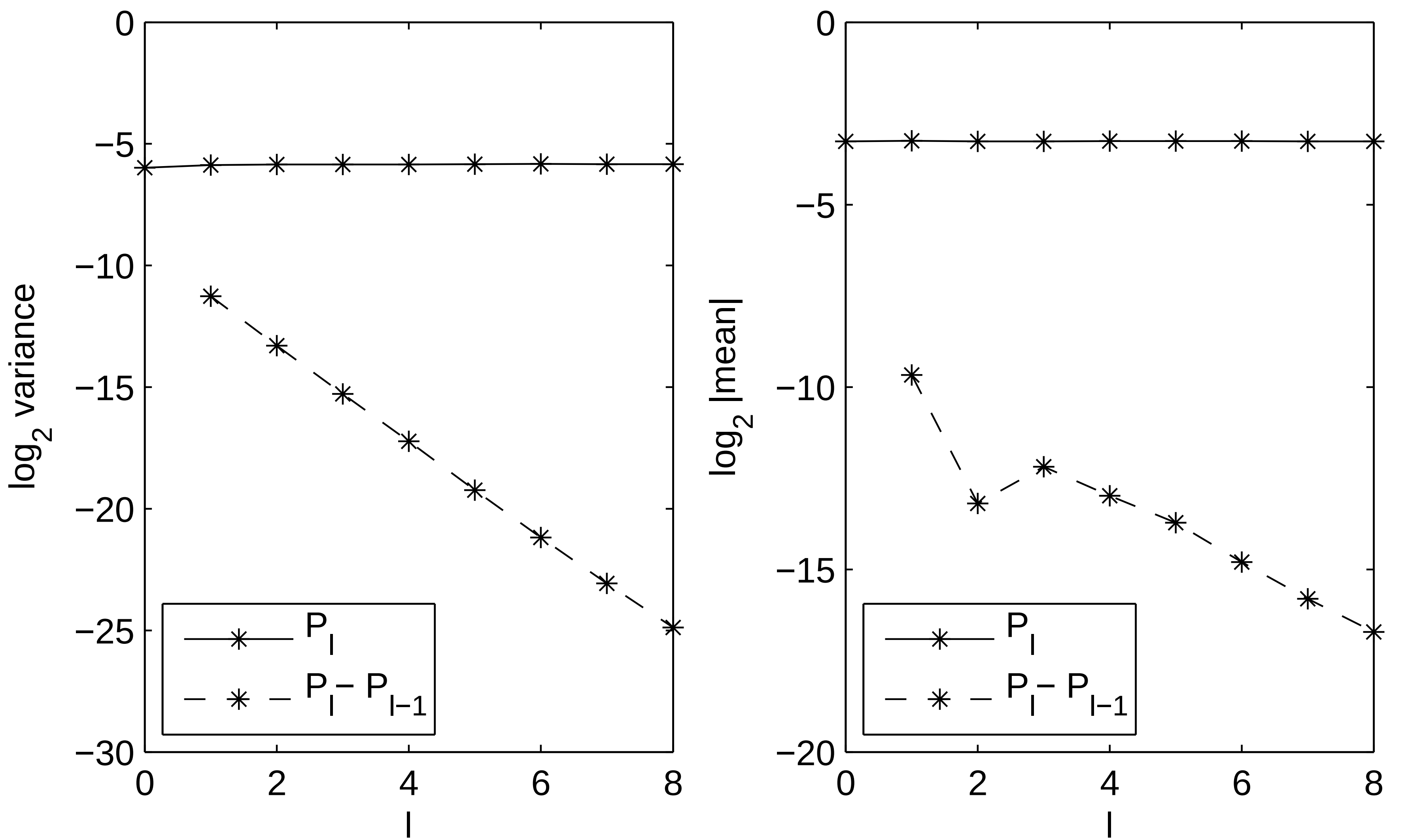
$$
\begin{aligned}
\mathrm{d}S &= r\,S\,\mathrm{d}t + \sqrt{V}\,S\,\mathrm{d}W_1, \qquad 0<t<T\\
\mathrm{d}V &= \lambda\,(\sigma^2 - V)\,\mathrm{d}t + \xi\,\sqrt{V}\;\mathrm{d}W_2,
\end{aligned}
$$

$T=1,\quad S(0)=1,\quad V(0)=0.04,\quad r=0.05,$

$\sigma=0.2,\quad \lambda=5,\quad \xi=0.25,\quad \rho=-0.5$

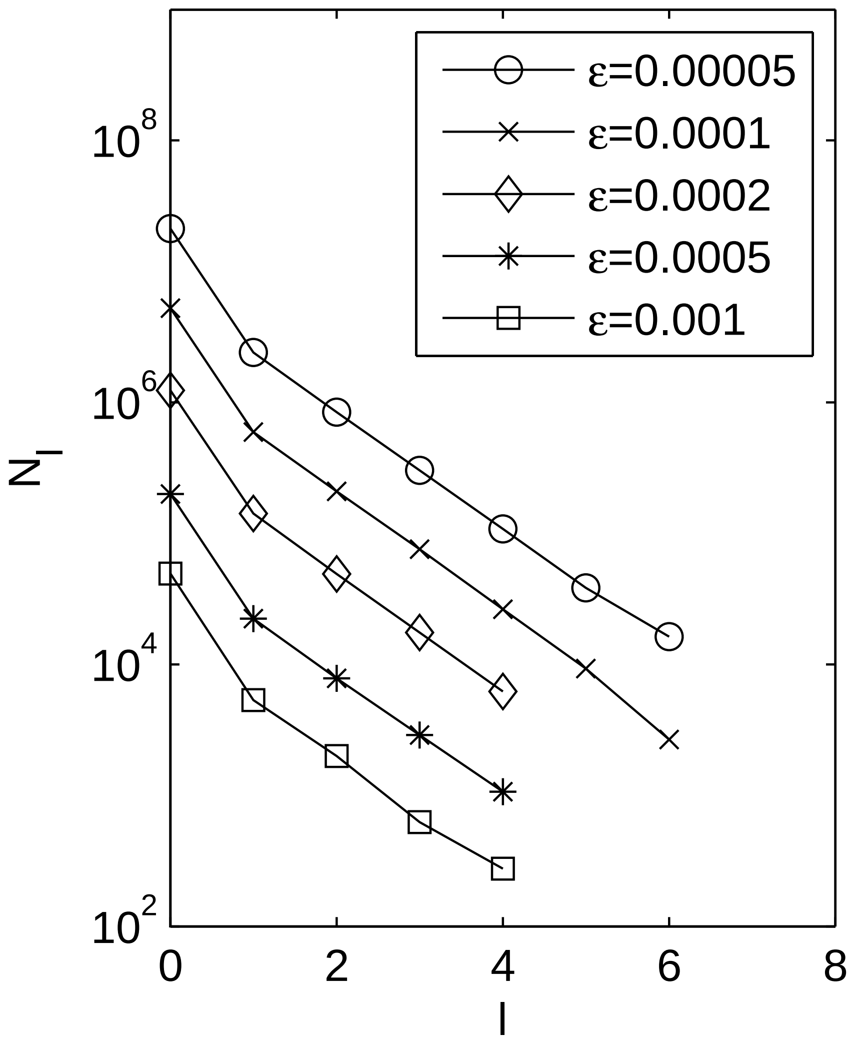
MLMC Results

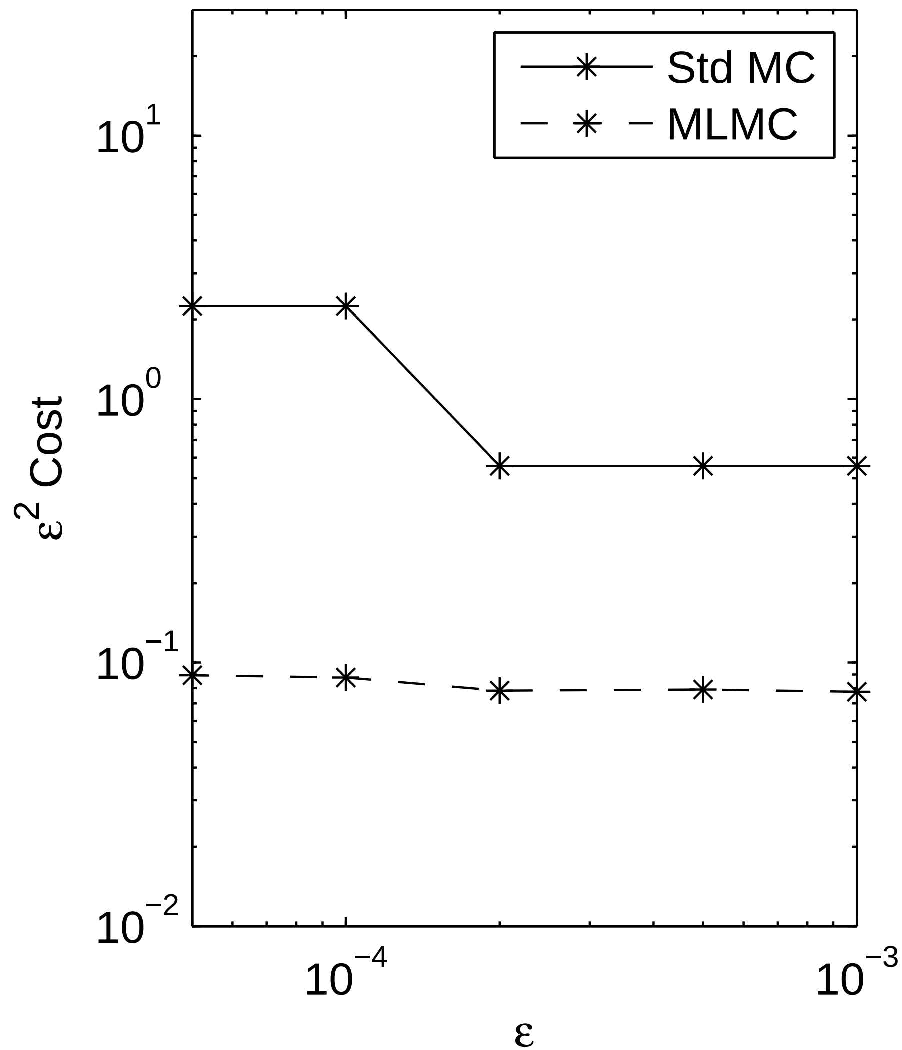
Heston model: European call



MLMC Results

Heston model: European call





Extensions

1) Quasi-Monte Carlo

- standard Monte Carlo has a random sampling error proportional to $N^{-1/2}$
- Quasi-Monte Carlo uses a deterministic choice of sample “points” to achieve an error which is nearly $O(N^{-1})$ in the best cases
- Not much applicable theory because financial payoffs don’t have required smoothness
- In practice, get great results using rank-1 lattice rules developed by Ian Sloan’s group at UNSW
- Haven’t yet tried Sobol sequences

Extensions

2) Numerical Analysis

	Euler		Milstein	
option	numerics	analysis	numerics	analysis
Lipschitz	$O(h)$	$O(h)$	$O(h^2)$	$O(h^2)$
Asian	$O(h)$	$O(h)$	$O(h^2)$	$O(h^2)$
lookback	$O(h)$	$O(h)$	$O(h^2)$	$o(h^{2-\delta})$
barrier	$O(h^{1/2})$	$o(h^{1/2-\delta})$	$O(h^{3/2})$	$o(h^{3/2-\delta})$
digital	$O(h^{1/2})$	$O(h^{1/2}\log h)$	$O(h^{3/2})$	$o(h^{3/2-\delta})$

Table: V_l convergence observed numerically (for GBM) and proved analytically (for more general SDEs) for both the Euler and Milstein discretisations. δ can be any strictly positive constant.

Extensions

Analysis for Euler discretisations:

- lookback and barrier options: Giles, Higham & Mao (*Finance & Stochastics, 2009*)
  - lookback analysis follows from strong convergence
  - barrier analysis shows dominant contribution comes from paths which are near the barrier; uses asymptotic analysis, first proving that "extreme" paths have negligible contribution
  - similar analysis for digital options gives $O(h^{1/2-\delta})$ bound instead of $O(h^{1/2}\log h)$
- digital options: Avikainen (*Finance & Stochastics, 2009*)
  - method of analysis is quite different

Extensions

Analysis for Milstein discretisations:

- work in progress by Giles, Debrabant & Rößler
- uses boundedness of all moments to bound the contribution to V_l from "extreme" paths (e.g. for which $\max_n |\Delta W_n| > h^{1/2-\delta}$ for some $\delta > 0$)
- uses asymptotic analysis to bound the contribution from paths which are not "extreme"

Extensions

3) "Greeks"

- this is the name given to derivatives such as $\dfrac{\partial}{\partial S_0}\,\mathbb{E}[P]$
- under certain circumstance, this is equal to $\mathbb{E}\left[\dfrac{\partial P}{\partial S_0}\right]$

  – this leads to the pathwise differentiation approach
- the multilevel approach should again work well but not tried yet
- can also incorporate the adjoint approach developed with Paul Glasserman – more efficient when many Greeks are wanted for one payoff function

Extensions

4) multivariate discontinuous payoffs

- problem with discontinuous payoffs is that small changes in path can lead to a big change in the payoff
- so far, have treated digital options using a “trick” in Paul Glasserman’s book, taking the conditional expectation one timestep before maturity, which effectively smooths the payoff
- the “vibrato” Monte Carlo idea generalises this to multivariate cases in which the conditional expectation is not known in closed form
- alternative is to use “splitting”, with multiple independent simulations of final timestep of each path

Extensions

5) American options

- with European options, the buyer can only exercise the option at maturity, the final time T
- with American options, the buyer can exercise at any time, leading to an optimal control problem
- in PDE approaches, this is solved using a linear complementarity approach which marches backwards in time
- modifying Monte Carlo methods is much harder – an active research topic
- I have some ideas on how to incorporate the multilevel approach – hope to start a project on this soon

Extensions

6) CUDA implementation on NVIDIA graphics cards

- advances in computer hardware/software are important as well as advances in mathematics
- graphics cards are very powerful parallel processors, with up to 240 cores per graphics chip (GPU)
- 2 years ago, NVIDIA introduced the CUDA development environment which uses minor extension to C/C++
- in 2007, a visiting student Xiaoke Su, achieved $100\times$ speedup on a LIBOR application using a 128-core GPU
- more recently, have been working with NAG to develop a random number generation library – freely available to academics

Extensions

Key issue in uniform random number generation:

- when generating 10M random numbers, might have 5000 threads and want each one to compute 2000 random numbers
- need a “skip-ahead” capability so that thread n can jump to the start of its “block” efficiently (usually $\log N$ cost to jump N elements)

Extensions

mrg32k3a (Pierre l'Ecuyer, '99, '02)

- popular generator in Intel MKL and ACML libraries
- pseudo-uniform output is $(x_{n,1} - x_{n,2} \mod m_1) \,/\, m_1$ where integers $x_{n,1}$, $x_{n,2}$ are defined by

$$\begin{aligned} x_{n,1} &= a_1\, x_{n-2,1} - b_1\, x_{n-3,1} \mod m_1 \\ x_{n,2} &= a_2\, x_{n-1,2} - b_2\, x_{n-3,2} \mod m_2 \end{aligned}$$

$a_1 = 1403580$, $b_1 = 810728$, $m_1 = 2^{32} - 209$,
$a_2 = 527612$, $b_2 = 1370589$, $m_2 = 2^{32} - 22853$.

Extensions

- Both recurrences are of the form

$$y_n = A\, y_{n-1} \mod m$$

where y_n is a vector $y_n = (x_n,\ x_{n-1},\ x_{n-2})^T$ and A is a 3×3 matrix. Hence

$$y_{n+2^k} = A^{2^k} y_n \mod m = A_k\, y_n \mod m$$

where A_k is defined by repeated squaring as

$$A_{k+1} = A_k\, A_k \mod m, \qquad A_0 \equiv A.$$

Can generalise this to jump N places in $O(\log N)$ operations.

Extensions

- output distributions:
  - uniform
  - exponential: trivial
  - Normal: Box-Muller or inverse CDF
  - Gamma: using “rejection” methods which require a varying number of uniforms and Normals to generate 1 Gamma variable
- producing Normals with **mrg32k3a**:
  - 2400M values/sec on a 216-core GTX260
  - 70M values/sec on a Xeon using Intel’s VSL
- have also implemented a **Sobol** generator to produce quasi-random numbers

Conclusions

Multilevel Monte Carlo method has already achieved

- improved order of complexity
- significant benefits for model problems

but more research is needed, both theoretical and applied.

Acknowledgements:

- Paul Glasserman, Mark Broadie, Terry Lyons for early discussions
- Microsoft, EPSRC, Oxford-Man Institute for funding