Multilevel Quasi-Monte Carlo

Mike Giles

giles@comlab.ox.ac.uk

Oxford University Computing Laboratory

Acknowledgment: research funding from Microsoft and EPSRC

and collaboration with Ian Sloan & Frances Kuo, UNSW

Outline

Objective is faster Monte Carlo simulation of path dependent options to estimate values and Greeks.

Several ingredients, not yet all combined:

- quasi-Monte Carlo (not new)
- multilevel method (new)
- adjoint pathwise Greeks (newish)

- multicore processing (work-in-progress)
  96-way parallel processing on plug-in cards

Generic Problem

Stochastic differential equation with general drift and volatility terms:

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

We want to compute the expected value of an option dependent on $S(t)$. In the simplest case of European options, it is a function of the terminal state

$$P = f(S(T))$$

with a uniform Lipschitz bound,

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

Simplest MC Approach

Euler discretisation with timestep h:

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

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

$$\widehat{Y} = N^{-1} \sum_{i=1}^{N} f(\widehat{S}^{(i)}_{T/h})$$

- weak convergence – $O(h)$ error in expected payoff
- strong convergence – $O(h^{1/2})$ error in individual path

Simplest MC Approach

Mean Square Error is $O\left(N^{-1} + h^2\right)$

- first term comes from variance of estimator
- second term comes from bias due to weak convergence

To make this $O(\varepsilon^2)$ 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 $O\left(\varepsilon^{-p}\right)$, with p as small as possible, ideally close to 1.

Note: for a relative error of $\varepsilon = 0.001$, the difference between ε^{-3} and ε^{-1} is huge.

Standard MC Improvements

- variance reduction techniques (e.g. control variates, stratified sampling) improve the constant factor in front of ε^{-3}, sometimes spectacularly
- improved second order weak convergence (e.g. through Richardson extrapolation) leads to $h = O(\sqrt{\varepsilon})$, giving $p=2.5$
- Quasi-Monte Carlo reduces the number of samples required, at best leading to $N \approx O(\varepsilon^{-1})$, giving $p \approx 2$ with first order weak methods

Multilevel method gives $p=2$ without QMC, and at best $p \approx 1$ with QMC.

Quasi-Monte Carlo

- well-established technique for approximating high-dimensional integrals
- for finance applications see papers by l'Ecuyer and book by Glasserman
- Sobol sequences are perhaps most popular; I use lattice rules (Sloan & Kuo)
- two important ingredients for success:
  - randomized QMC for confidence intervals
  - good identification of “dominant dimensions”

Quasi-Monte Carlo

If $Z_n \sim U[0,1]$ then $\Phi^{-1}(Z_n) \sim N(0,1)$. Hence, the expected value from a path discretization based on d unit Normals can be expressed as:

$$\int_{[0,1]^d} f(S(Z))\, \mathrm{d}Z$$

- standard MC uses random points; average of N samples is an unbiased estimator with $O(N^{-1/2})$ r.m.s. error and a computable confidence interval
- QMC uses special points with a uniform distribution; average of N samples has an error which at best is $O(N^{-1})$ but without a computable confidence interval

Quasi-Monte Carlo

To regain confidence intervals when using lattice rules use multiple “sets” of QMC points, each with a random offset:

$$\widehat{Z}_{m,n} = Z_n + r_m \mod 1$$

- average of each set is a random variable; its mean is unbiased and has a computable confidence interval.
- how many sets? – a tradeoff between efficiency and accuracy of confidence interval.
- some use as few as 10 sets – I prefer 32.

(For Sobol sequences, more common to use digital scrambling to maintain certain desirable properties.)

Quasi-Monte Carlo

Observation: QMC points are very uniform in leading few dimensions, so give very accurate results if integrand depends primarily on first few dimensions.

Consequence: QMC efficiency is greatly enhanced by changing how the Z_n are used in the path simulation

- Brownian Bridge:
  Z_1 used to compute $W(T)$,
  Z_2 used to compute $W(T/2)$, conditional on $W(T)$
  Z_3, Z_4 used to compute $W(T/4), W(3T/4)$, etc.
- Principal Component Analysis (PCA):
  similar idea, essential when $S(t)$ is multi-dimensional

Multilevel MC Approach

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

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

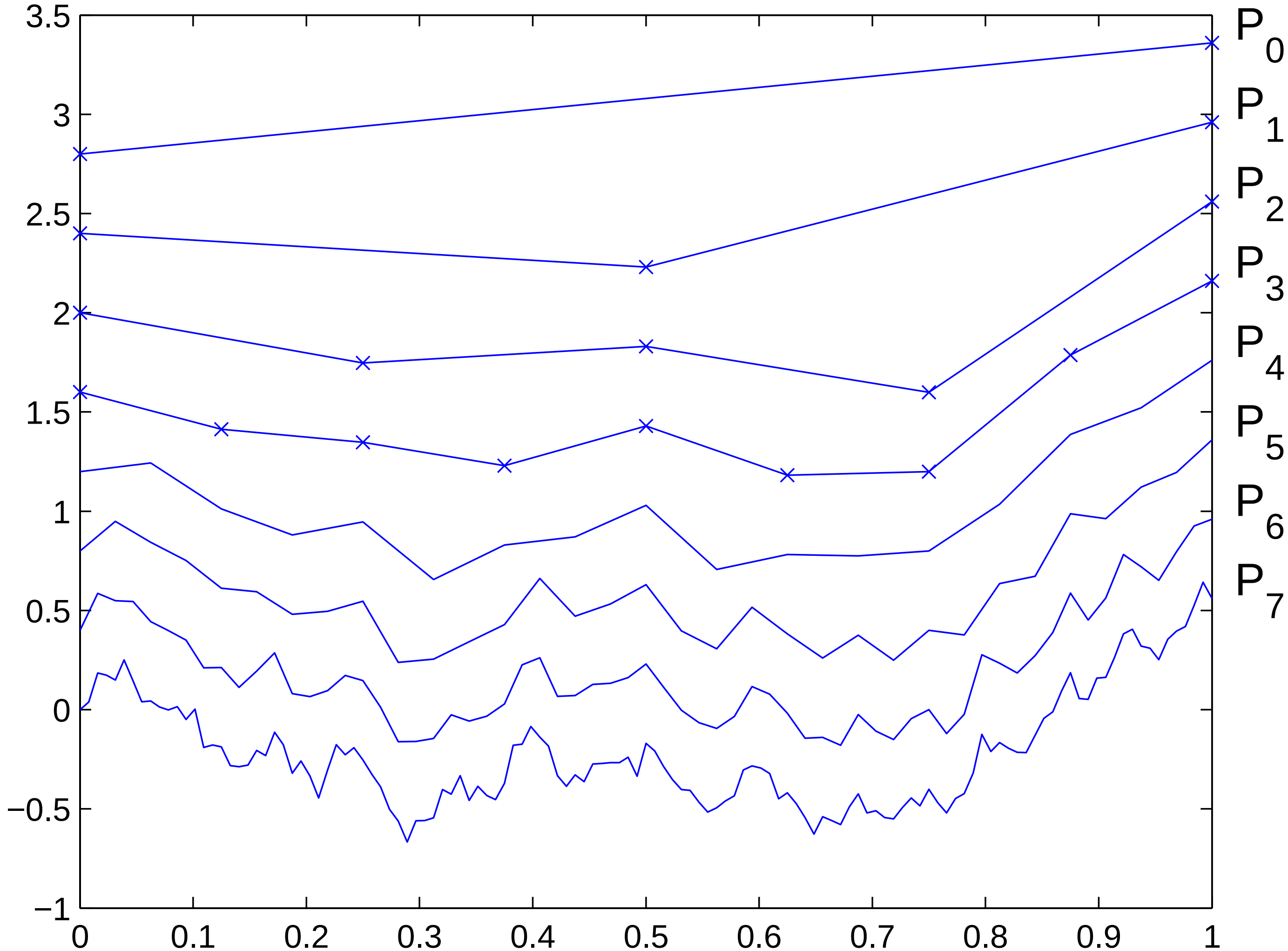
Expected value is same – aim is to reduce variance of estimator for a fixed computational cost.

Key point: approximate $E[\widehat{P}_l - \widehat{P}_{l-1}]$ 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

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

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

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 a Lipschitz payoff function

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

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:

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

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

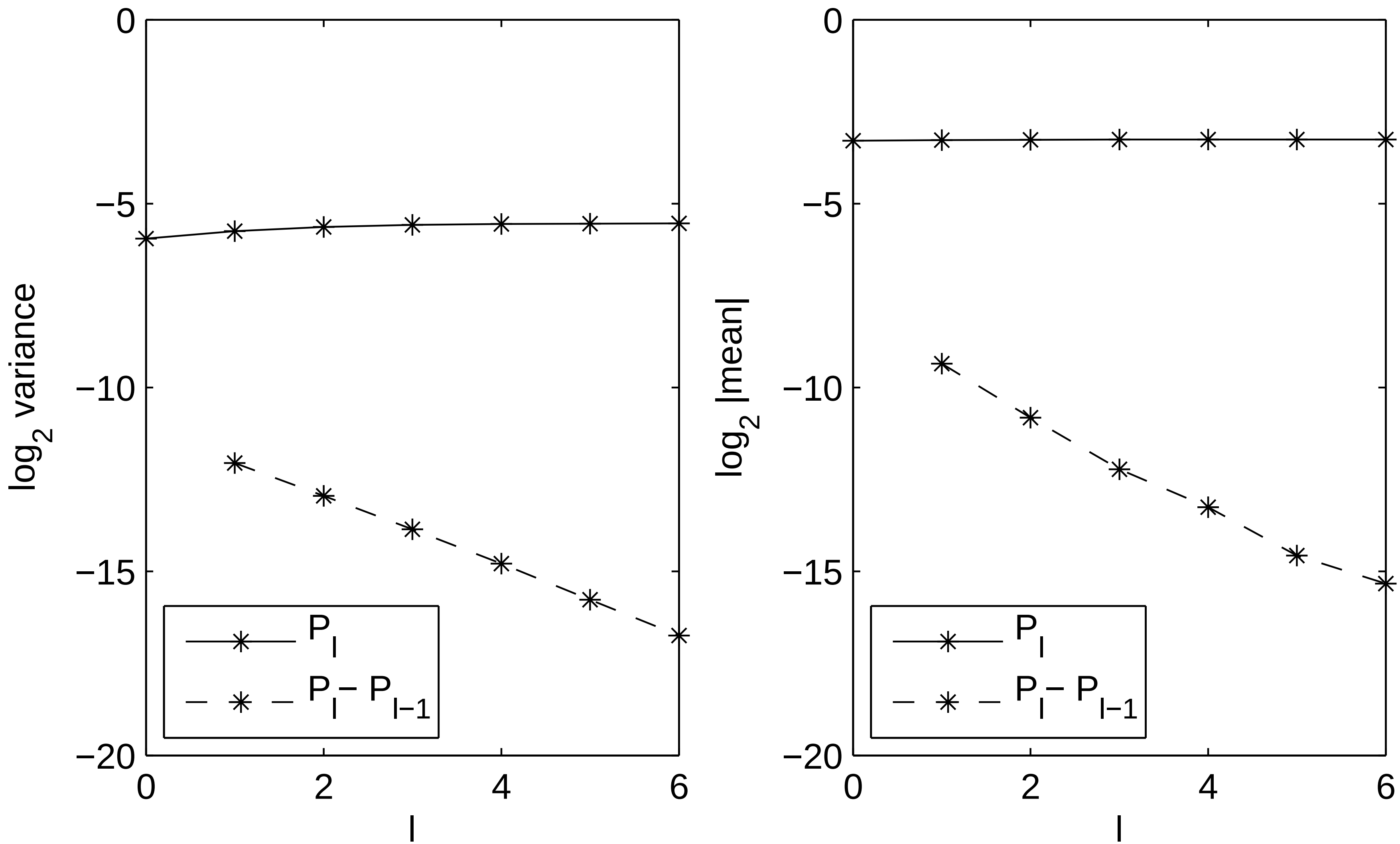
European call option with discounted payoff ($K\!=\!1$)

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

Down-and-out barrier option: same provided $S(t)$ stays above $B\!=\!0.9$

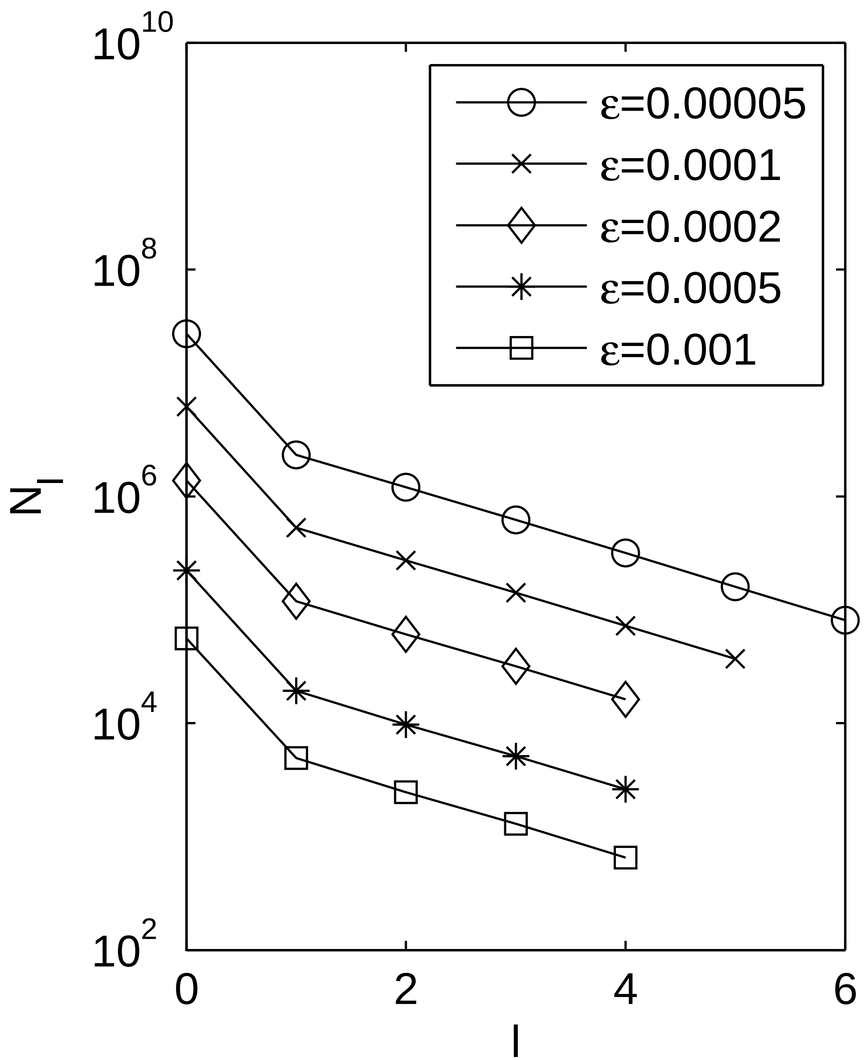
Results

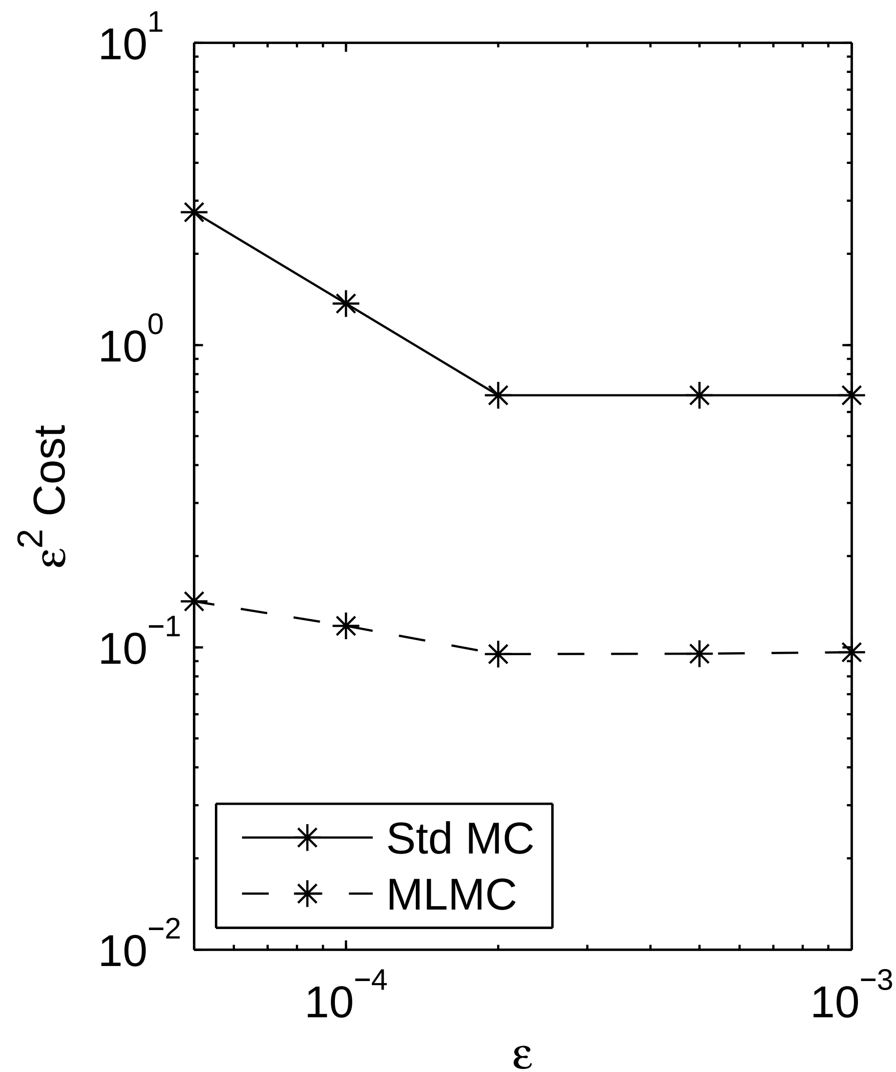
GBM: European call



Results

GBM: European call





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) $E[\widehat{P}_l - P] \leq c_1\, h_l^\alpha$

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

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

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

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:

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

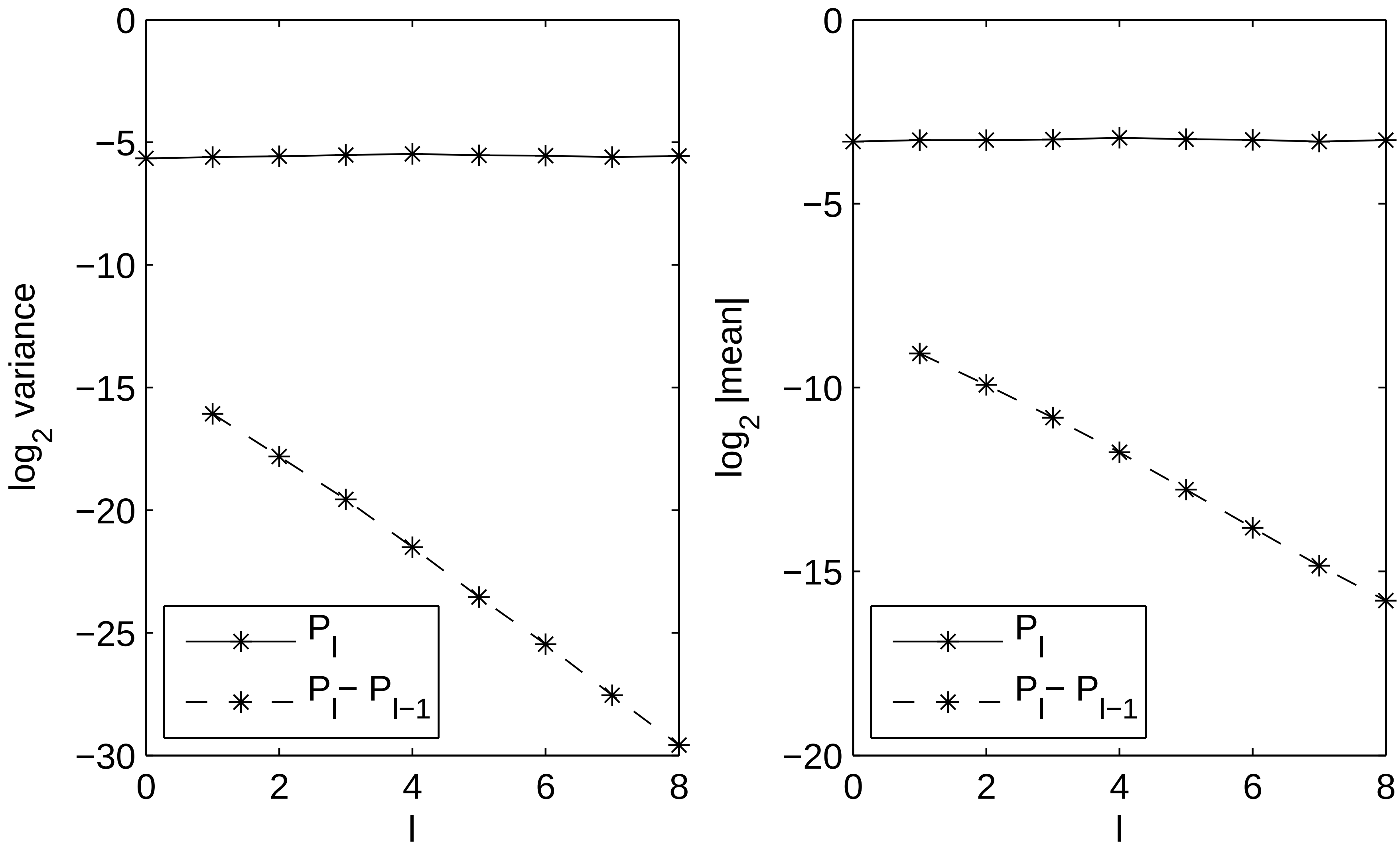
Milstein Scheme

In scalar case:

- $O(h)$ strong convergence
- $O(\varepsilon^{-2})$ complexity for Lipschitz payoffs – trivial
- $O(\varepsilon^{-2})$ complexity for Asian, lookback, barrier and digital options using carefully constructed estimators, based on Brownian interpolation
- key idea: within each timestep, model the behaviour as simple Brownian motion conditional on the two end-points – analytic results exist for distribution of min/max/average

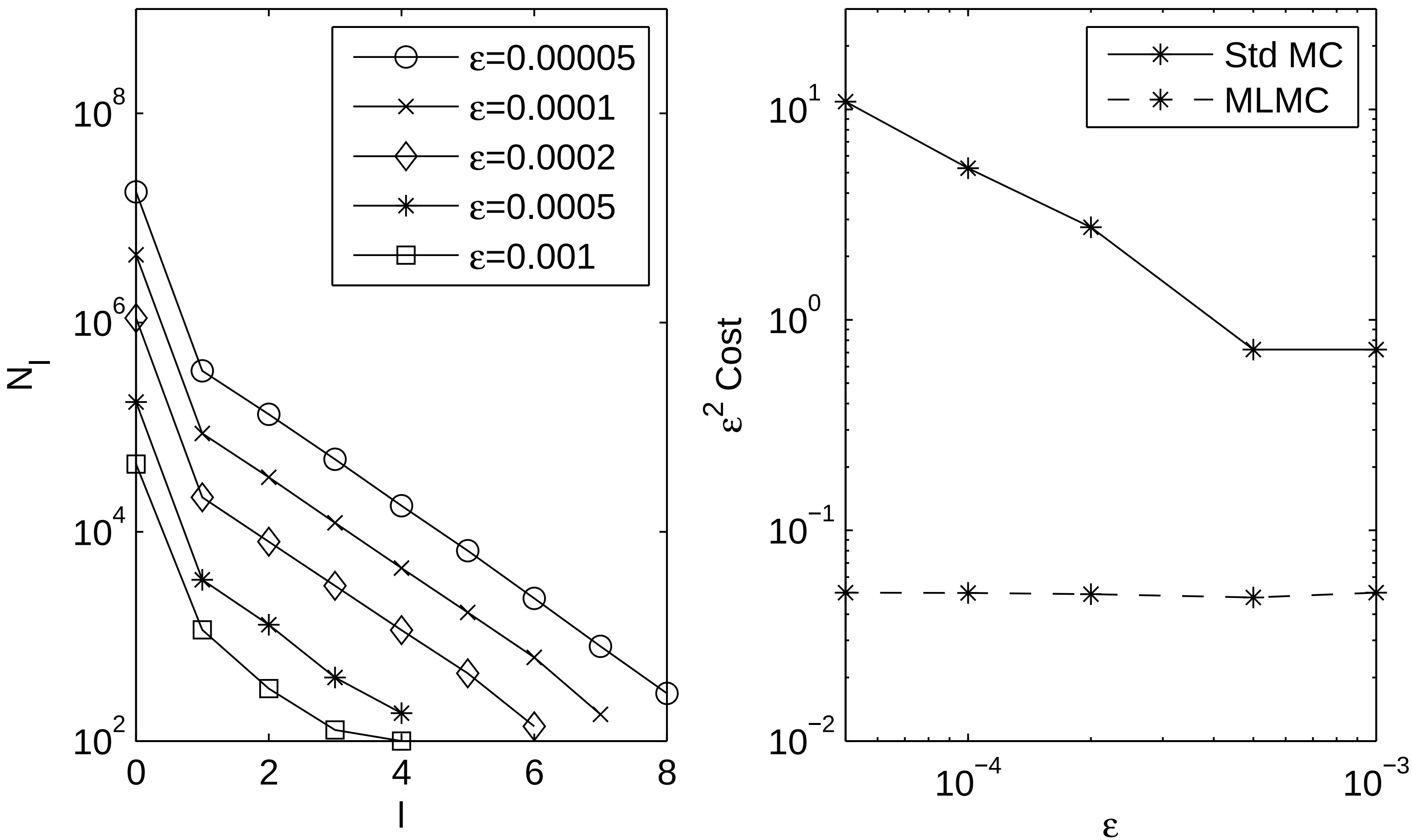
MLMC Results

GBM: European call



MLMC Results

GBM: European call

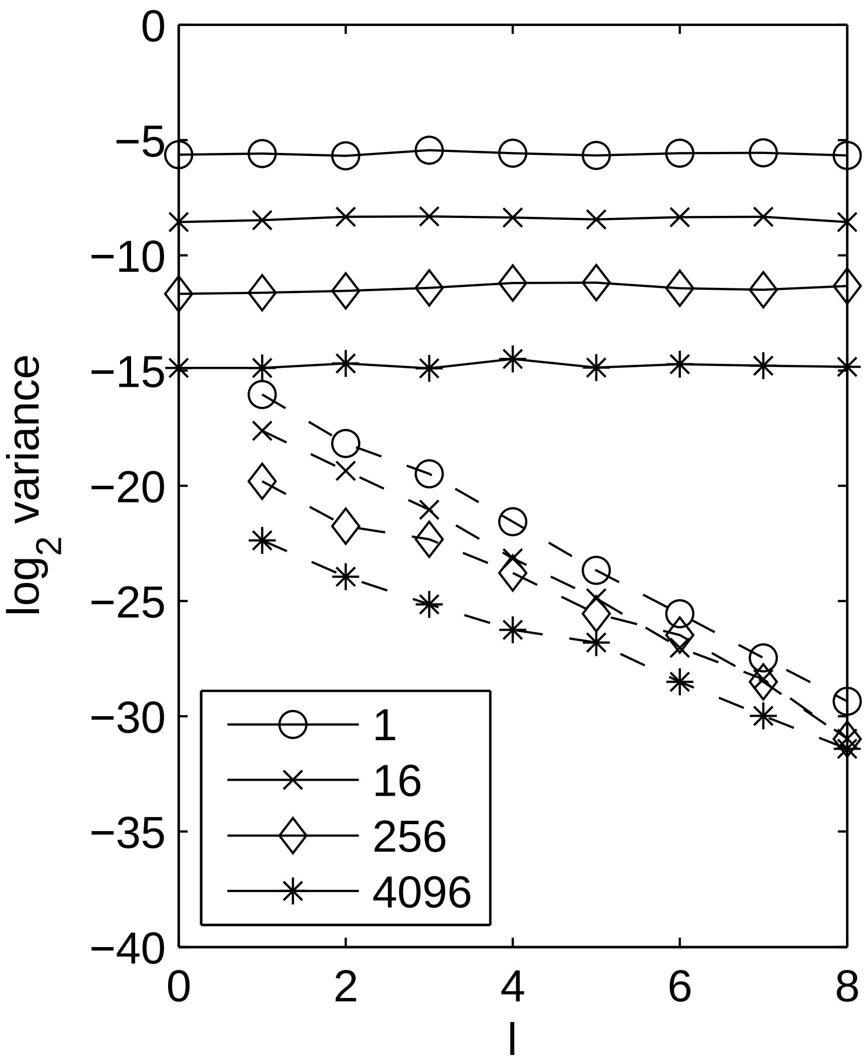


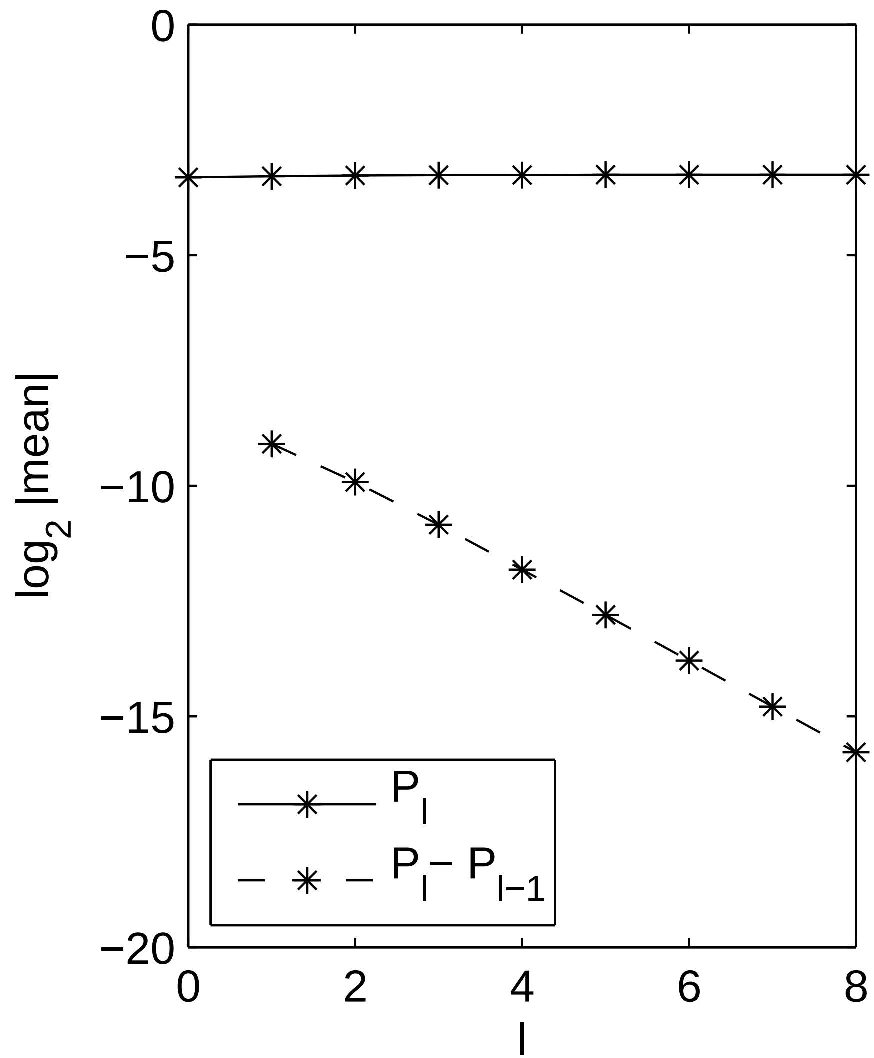
Multilevel QMC

- rank-1 lattice rule developed by Sloan, Kuo & Waterhouse at UNSW
- 32 randomly-shifted sets of QMC points
- number of points in each set increased as needed to achieved desired accuracy, based on confidence interval estimate
- results show QMC to be particularly effective on lowest levels with low dimensionality

MLQMC Results

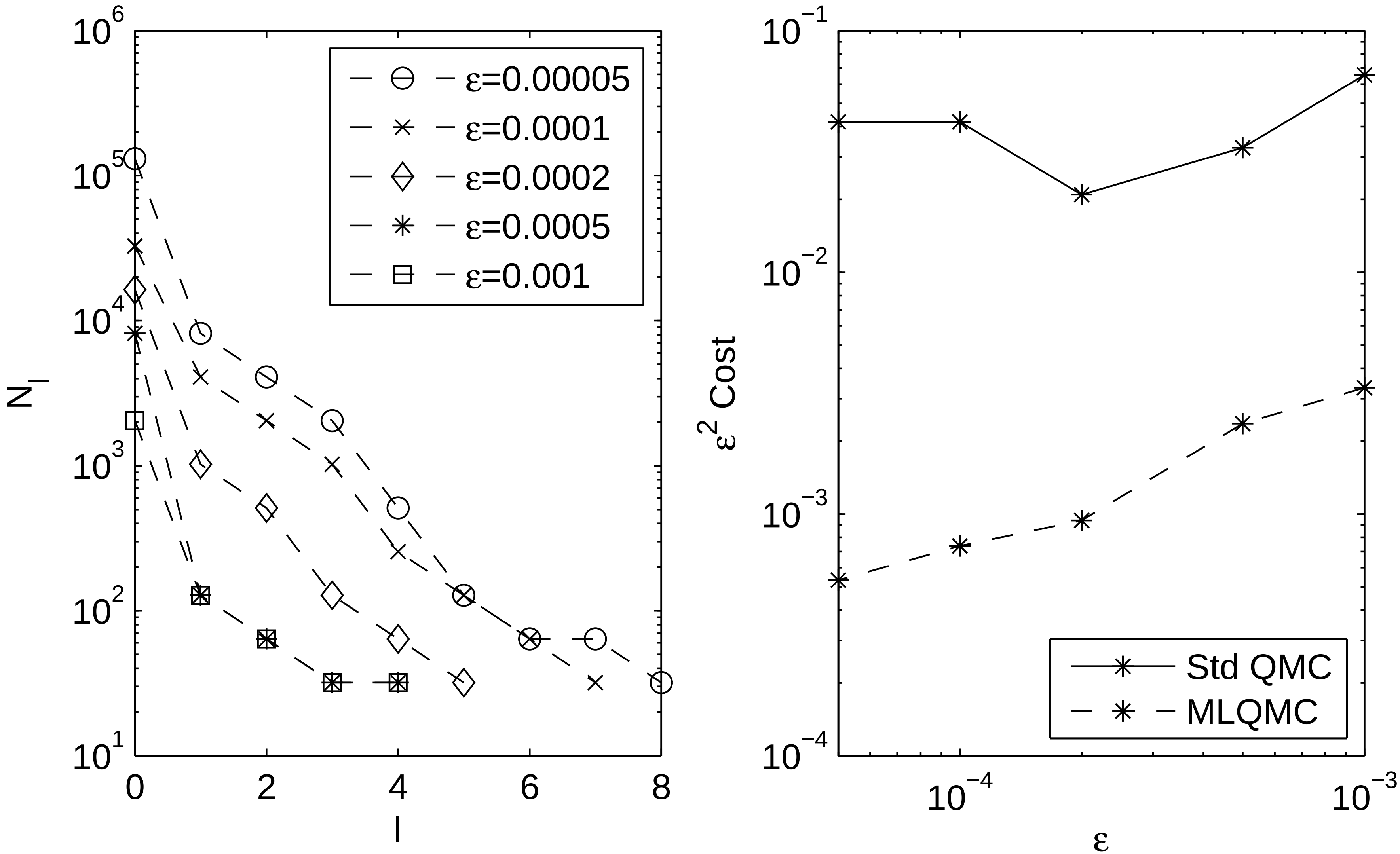
GBM: European call





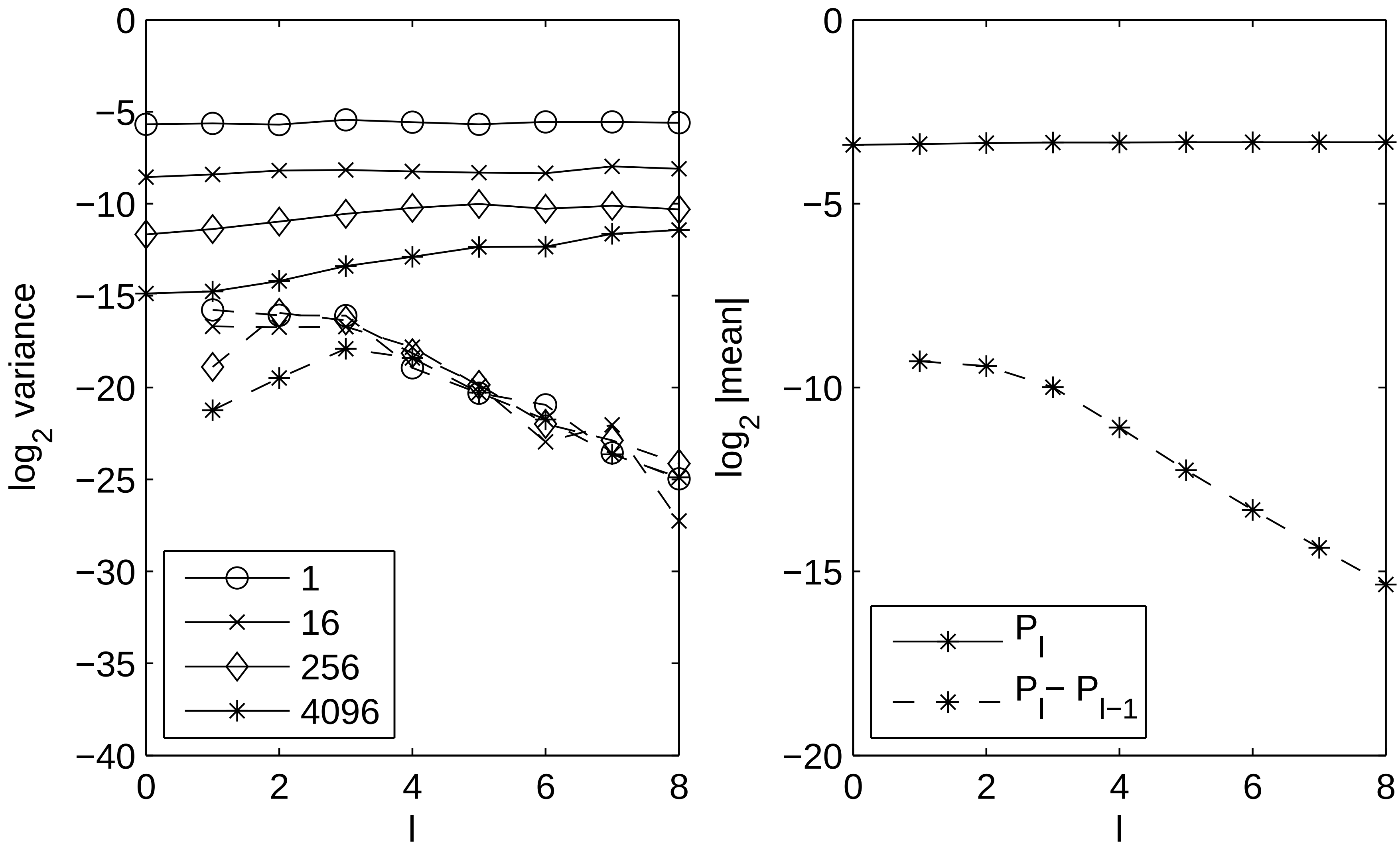
MLQMC Results

GBM: European call



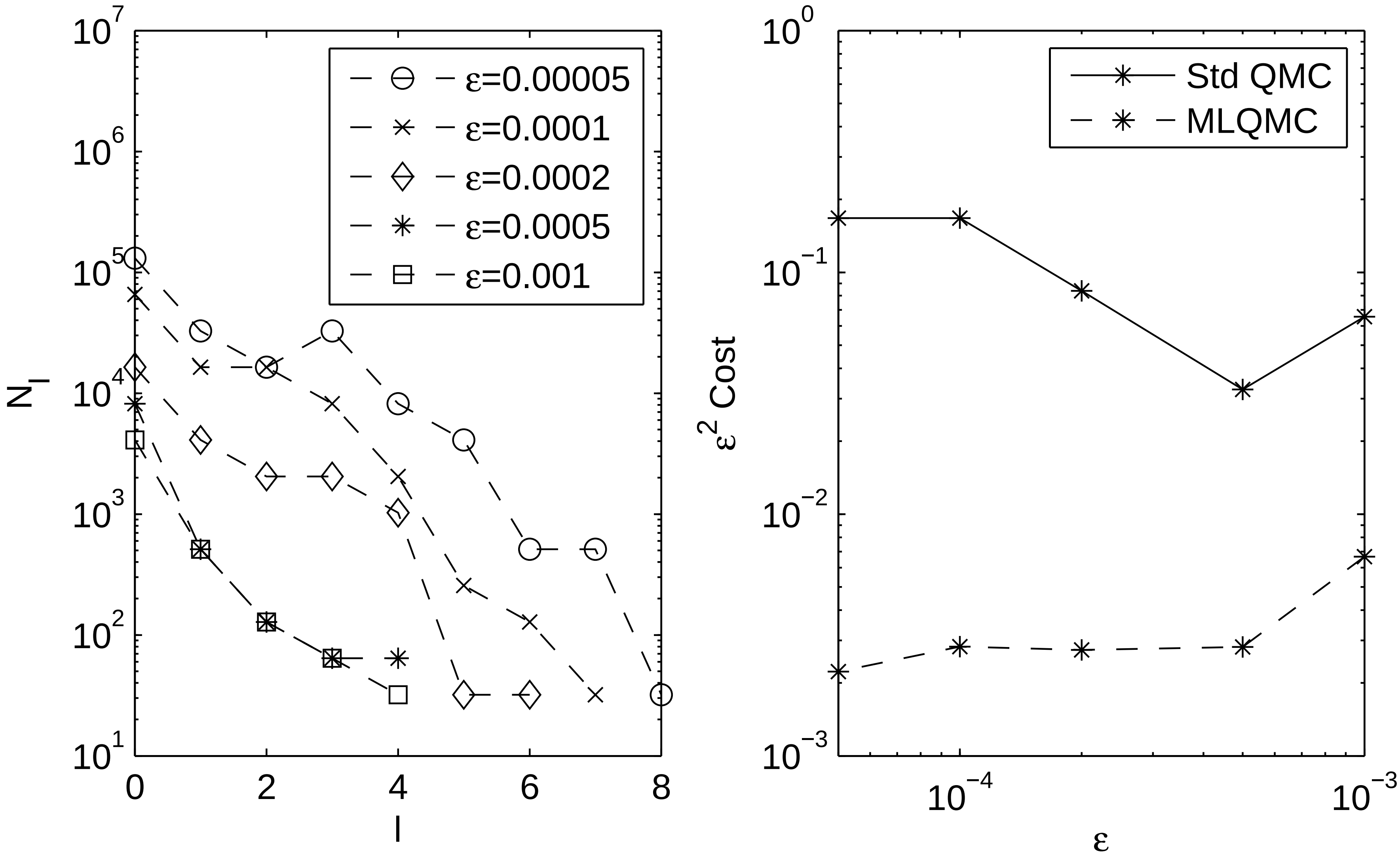
MLQMC Results

GBM: barrier option



MLQMC Results

GBM: barrier option



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 – future challenge

Results

Heston model:

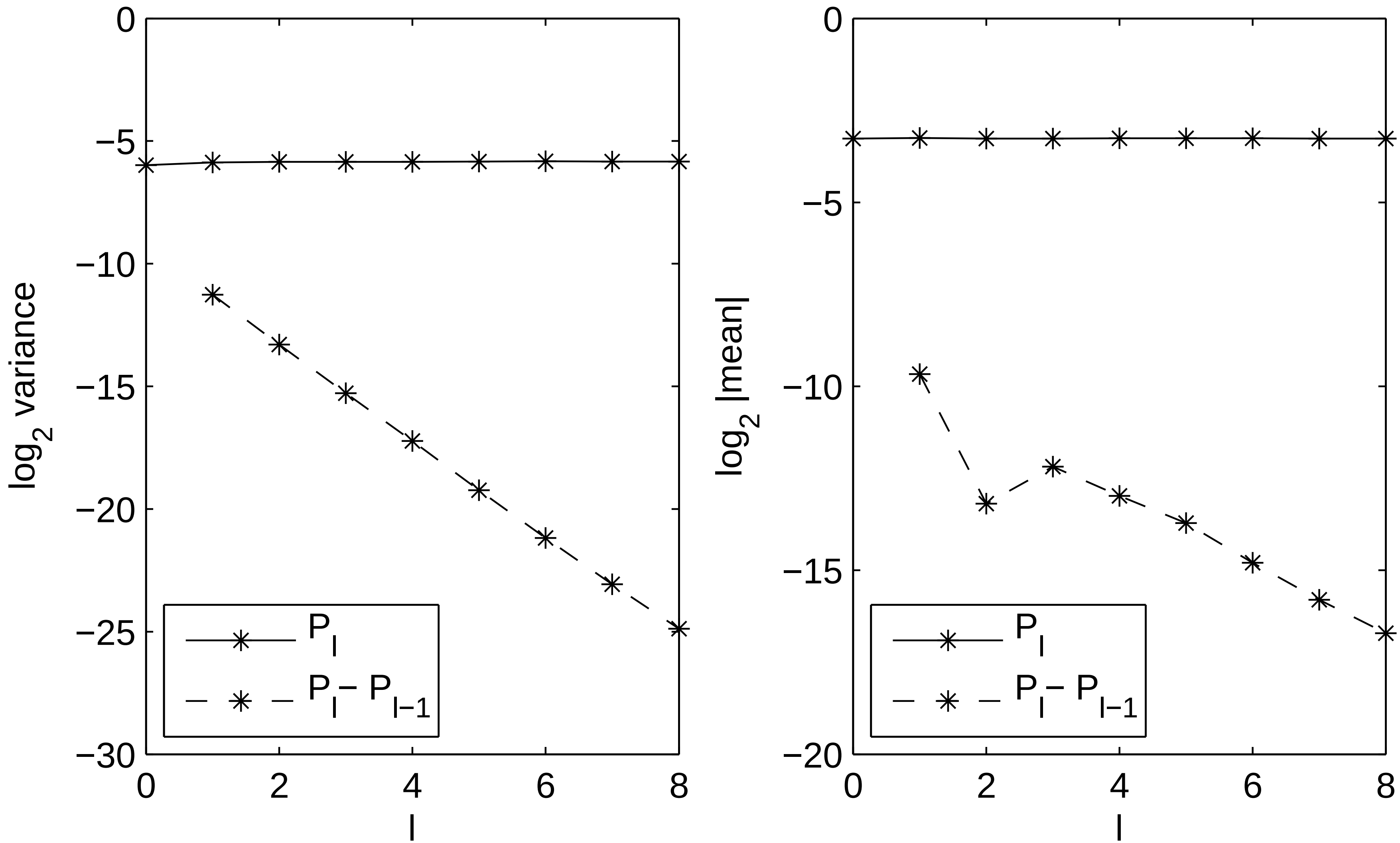
$$\begin{aligned} \mathrm{d}S &= r\,S\,\mathrm{d}t + \sqrt{V}\,S\,\mathrm{d}W_1, \qquad 0<t<T \\ \mathrm{d}V &= \lambda\,(\sigma^2 - V)\,\mathrm{d}t + \xi\,\sqrt{V}\,\mathrm{d}W_2, \end{aligned}$$

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

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

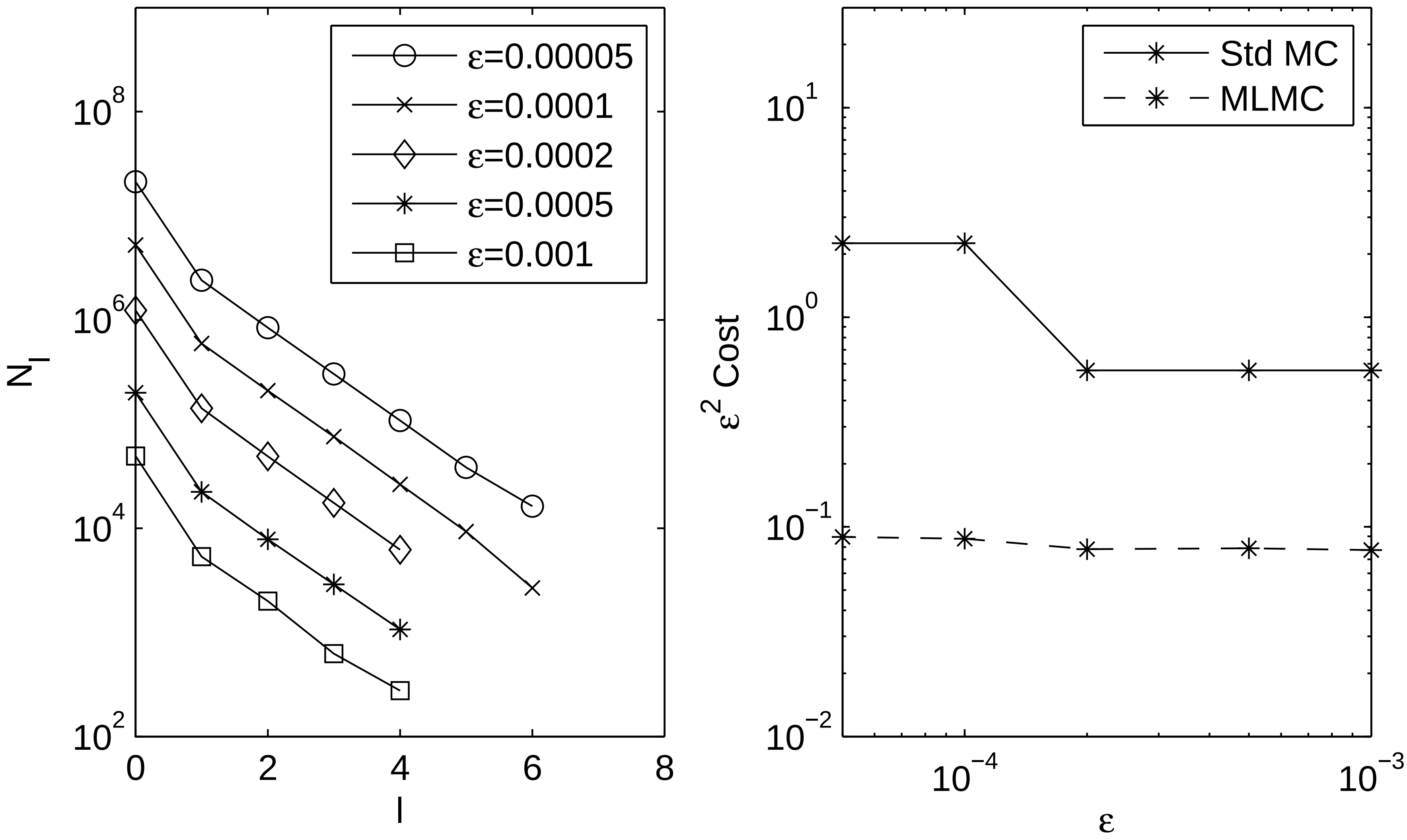
Results

Heston model: European call



Results

Heston model: European call



Greeks

As well as estimating the price, we also need to estimate various Greeks:

$$\Delta = \frac{\partial V}{\partial S_0}, \quad \Gamma = \frac{\partial^2 V}{\partial S_0^2}, \quad \text{Vega} = \frac{\partial V}{\partial \sigma}.$$

Various approaches:

- finite differences: simple but expensive and potentially inaccurate
- LRM: expensive since variance often increases as $h \to 0$
- Malliavin: complex but maybe good in some cases
- pathwise differentiation: most efficient, especially when using adjoints, but restricted to differentiable payoffs

Greeks

Under certain conditions (e.g. $f(S), a(S,t), b(S,t)$ all continuous and piecewise differentiable) the derivative with respect to some arbitrary input parameter θ is

$$\frac{\partial}{\partial \theta}\, E[f(S(T))] = E\left[\frac{\partial f(S(T))}{\partial \theta}\right] = E\left[\frac{\partial f}{\partial S}\ \frac{\partial S(T)}{\partial \theta}\right].$$

The discrete estimator therefore has the form

$$\widehat{Y}_\theta = N^{-1} \sum_{i=1}^{N} \frac{\partial f}{\partial S}(\widehat{S}^{(i)}_{T/h})\ \frac{\partial \widehat{S}}{\partial \theta}^{(i)}_{T/h}.$$

Greeks

If the discrete path evolution for a given set of Wiener increments is written as

$$\widehat{S}_{n+1} = F_n(\widehat{S}_n),$$

then differentiating it to calculate Δ gives

$$\frac{\partial \widehat{S}_{n+1}}{\partial S_0} \; = \; \frac{\partial F_n}{\partial S_n} \, \frac{\partial \widehat{S}_n}{\partial S_0} \; \equiv \; D(n) \, \frac{\partial \widehat{S}_n}{\partial S_0}$$

and so

$$\frac{\partial f}{\partial S} \, \frac{\partial \widehat{S}_{T/h}}{\partial S_0} = \frac{\partial f}{\partial S} \, D_{T/h} \ldots D_2 \; D_1 \; D_0.$$

Greeks

$$\frac{\partial f}{\partial S}\,\frac{\partial \widehat{S}_{T/h}}{\partial S_0} = \frac{\partial f}{\partial S}\; D_{T/h} \ldots D_2\; D_1\; D_0.$$

- in multi-dimensional cases, the D_n are matrices and $\dfrac{\partial f}{\partial S}$ is a row vector
- multiplying from right to left is the standard approach, and involves matrix-matrix multiplies
- multiplying from left to right is the adjoint approach, and involves vector-matrix products which are much cheaper
- this extends naturally to other Greeks, at a cost 2-3 times the original path calculation *independent of the number of first order Greeks being calculated*

Greeks

- combining adjoint Greeks with multilevel Monte Carlo is fine in principle, but not yet tested
- first order Greeks are one degree less smooth than payoffs, so Delta of European call is similar to a digital option, and can't do second order Greeks without smoothing
- big challenge is the need for payoff differentiability — new "vibrato" Monte Carlo idea combines adjoint pathwise sensitivity for path calculation with LRM for payoff evaluation, and eases implementation too

Conclusions

Results so far:

- (much) improved order of complexity
- (fairly) easy to implement
- significant benefits for model problems

However:

- lots of scope for further development
  - multi-dimensional SDEs needing Lévy areas
  - combining adjoint Greeks and multilevel MC
  - “vibrato” Monte Carlo
  - numerical analysis of algorithms
- need to test ideas on real finance applications

Papers

M.B. Giles, “Multilevel Monte Carlo path simulation”, to appear in *Operations Research*

M.B. Giles, “Improved multilevel convergence using the Milstein scheme”, to appear in proceedings of *MCQMC06*

M.B. Giles & P. Glasserman, “Smoking Adjoints: fast Monte Carlo Greeks”, *Risk*, January 2006.

`www.comlab.ox.ac.uk/mike.giles/finance.html`

Email: `giles@comlab.ox.ac.uk`