Multilevel quasi-Monte Carlo path simulation

Mike Giles
mike.giles@maths.ox.ac.uk

Oxford University Mathematical Institute
Oxford-Man Institute of Quantitative Finance

Acknowledgments: Frances Kuo, Ian Sloan, Ben Waterhouse (UNSW – 2007)
and Adrien Grumberg (Oxford – 2015)
Objective of this research was faster Monte Carlo simulation of path dependent options to estimate values and Greeks.

Several separate ingredients:
- multilevel method
- quasi-Monte Carlo
- adjoint pathwise Greeks
- parallel computing on NVIDIA graphics cards

Emphasis in this presentation was on multilevel QMC
Generic Problem

Stochastic differential equation with general drift and volatility terms:

\[ dS(t) = a(S, t) \, dt + b(S, t) \, dW(t) \]

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 $h$:

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

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

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

- 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 \( O(N^{-1} + h^2) \)
- first term comes from variance of estimator
- second term comes from bias due to weak convergence

To make this \( O(\varepsilon^2) \) requires

\[
N = O(\varepsilon^{-2}), \quad h = O(\varepsilon) \quad \Rightarrow \quad \text{cost} = O(N h^{-1}) = O(\varepsilon^{-3})
\]

Aim is to improve this cost to \( O(\varepsilon^{-p}) \), with \( p \) as small as possible, ideally close to 1.

Note: for a relative error of \( \varepsilon = 0.001 \), the difference between \( \varepsilon^{-3} \) and \( \varepsilon^{-1} \) is huge.
Standard MC Improvements

- variance reduction techniques (e.g. control variates, stratified sampling) improve the constant factor in front of $\varepsilon^{-3}$, 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.
MLMC Approach

Consider multiple sets of simulations with different timesteps $h_\ell = 2^{-\ell} T$, $\ell = 0, 1, \ldots, L$, and payoff $\hat{P}_\ell$

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

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

Key point: approximate $\mathbb{E}[\hat{P}_\ell - \hat{P}_{\ell-1}]$ using $N_\ell$ simulations with $\hat{P}_\ell$ and $\hat{P}_{\ell-1}$ obtained using same Brownian path.

$$
\hat{Y}_\ell = N_\ell^{-1} \sum_{i=1}^{N_\ell} \left( \hat{P}_\ell^{(i)} - \hat{P}_{\ell-1}^{(i)} \right)
$$
MLMC Approach

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

$$\mathbb{V} \left[ \sum_{\ell=0}^{L} \hat{Y}_\ell \right] = \sum_{\ell=0}^{L} N_\ell^{-1} V_\ell, \quad V_\ell \equiv \mathbb{V}[\hat{P}_\ell - \hat{P}_{\ell-1}],$$

and the computational cost is proportional to $$\sum_{\ell=0}^{L} N_\ell h_\ell^{-1}.$$ 

Hence, the variance is minimised for a fixed computational cost by choosing $$N_\ell$$ to be proportional to $$\sqrt{V_\ell h_\ell}.$$ 

The constant of proportionality can be chosen so that the combined variance is $$O(\varepsilon^2).$$
Theorem: Let $P$ be a functional of the solution of a stochastic o.d.e., and $\hat{P}_\ell$ the discrete approximation using a timestep $h_\ell = M^{-\ell} T$.

If there exist independent estimators $\hat{Y}_\ell$ based on $N_\ell$ Monte Carlo samples, and positive constants $\alpha \geq \frac{1}{2}, \beta, c_1, c_2, c_3$ such that

i) $\mathbb{E}[\hat{P}_\ell - P] \leq c_1 h_\ell^\alpha$

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

iii) $\nabla[\hat{Y}_l] \leq c_2 N_\ell^{-1} h_\ell^\beta$

iv) $C_\ell$, the computational complexity of $\hat{Y}_\ell$, is bounded by

$$C_\ell \leq c_3 N_\ell h_\ell^{-1}$$
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

$$\hat{Y} = \sum_{\ell=0}^{L} \hat{Y}_\ell,$$

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

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 scheme — better strong convergence, same weak convergence

Generic scalar SDE:

\[ dS(t) = a(S, t) \, dt + b(S, t) \, dW(t), \quad 0 < t < T. \]

Milstein scheme:

\[ \hat{S}_{n+1} = \hat{S}_n + ah + b \Delta W_n + \frac{1}{2} b' b \left( (\Delta W_n)^2 - h \right). \]
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 or extrapolation
Milstein Scheme

Brownian interpolation: within each timestep, model the behaviour as simple Brownian motion conditional on the two end-points

\[
\hat{S}(t) = \hat{S}_n + \lambda(t)(\hat{S}_{n+1} - \hat{S}_n) \\
+ b_n \left( W(t) - W_n - \lambda(t)(W_{n+1} - W_n) \right),
\]

where

\[
\lambda(t) = \frac{t - t_n}{t_{n+1} - t_n}
\]

There then exist analytic results for the distribution of the min/max/average over each timestep.
Milstein Scheme

Brownian extrapolation for final timestep:

\[ \hat{S}_N = \hat{S}_{N-1} + a_{N-1} h + b_{N-1} \Delta W_N \]

– considering all possible \( \Delta W_N \) gives Gaussian distribution, for which a digital option has a known conditional expectation (Glasserman)

This payoff smoothing can be generalised to multivariate cases, and leads to a “vibrato” Monte Carlo technique which is suitable for both efficient multilevel analysis and the computation of Greeks
Results

Geometric Brownian motion:

\[ dS = r \, S \, dt + \sigma \, S \, dW, \quad 0 < t < T, \]

with parameters \( T = 1, \quad S(0) = 1, \quad r = 0.05, \quad \sigma = 0.2 \)

- **European call option**: \( \exp(-rT) \, \max( S(T) - 1, 0 ) \)
- **European digital call**: \( \exp(-rT) \, 1_{S(T) > 1} \)
- **Down-and-out barrier option**: same as call provided \( S(t) \) stays above \( B = 0.9 \)
MLMC Results

GBM: European call

![Graphs showing log2 variance and log2 |mean| vs level l for European call option pricing using MLMC methods. The graphs display decreasing trends with increasing level l, indicating the variance and mean decrease as the level increases.](Multilevel Monte Carlo – p. 16)
MLMC Results

GBM: European call

![Graph showing MLMC results for GBM European call with different ε values.]
GBM: digital call

Multilevel Monte Carlo – p. 18
MLMC Results

GBM: digital call

\[ N_l \]

\[ \varepsilon = 0.0001 \]
\[ \varepsilon = 0.0002 \]
\[ \varepsilon = 0.0005 \]
\[ \varepsilon = 0.001 \]
\[ \varepsilon = 0.002 \]

\[ \varepsilon^2 \text{Cost} \]

\[ \text{Std MC} \]
\[ \text{MLMC} \]
MLMC Results

GBM: barrier option

![Graphs showing log2 variance and log2 |mean| for GBM barrier option](image)

- **Graph 1:** Log2 variance vs. level (l) with lines for $P_l$ and $P_l - P_{l-1}$.
- **Graph 2:** Log2 |mean| vs. level (l) with lines for $P_l$ and $P_l - P_{l-1}$.

Multilevel Monte Carlo – p. 20
MLMC Results

GBM: barrier option

![Graph showing MLMC results for GBM barrier option with different ε values. The x-axis represents the level (l), the y-axis represents the number of simulations (N), and the ε^2 cost is shown on a separate axis.](image)
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; we used rank-1 lattice rules (Sloan & Kuo)
- two important ingredients for success:
  - randomized QMC for confidence intervals
  - good identification of “dominant dimensions” (Brownian Bridge and/or PCA)
Quasi-Monte Carlo

Approximate high-dimensional hypercube integral

\[ \int_{[0,1]^d} f(x) \, dx \]

by

\[ \frac{1}{N} \sum_{i=0}^{N-1} f(x^{(i)}) \]

where

\[ x^{(i)} = \left[ \frac{i}{N} z \right] \]

and \( z \) is a \( d \)-dimensional “generating vector”.

Multilevel Monte Carlo – p. 23
Quasi-Monte Carlo

In the best cases, error is $O(N^{-1})$ instead of $O(N^{-1/2})$ but without a confidence interval.

To get a confidence interval, let

$$x^{(i)} = \left[ \frac{i}{N} z + x_0 \right].$$

where $x_0$ is a random offset vector.

Using 32 different random offsets gives a confidence interval in the usual way.
Quasi-Monte Carlo

For the path discretisation we can use

$$\Delta W_n = \sqrt{h} \Phi^{-1}(x_n),$$

where $\Phi^{-1}$ is the inverse cumulative Normal distribution.

Much better to use a Brownian Bridge construction:

- $x_1 \rightarrow W(T)$
- $x_2 \rightarrow W(T/2)$
- $x_3, x_4 \rightarrow W(T/4), W(3T/4)$
- \ldots and so on by recursive bisection
Multilevel QMC

- rank-1 lattice rule developed by Sloan, Kuo & Waterhouse at UNSW
- 32 randomly-shifted sets of QMC points gives unbiased estimate and confidence interval for multilevel correction
- MLQMC algorithm uses same heuristic as MLMC algorithm to estimate weak error and choose optimal number of levels
- MLQMC algorithm repeatedly doubles the number of points on the level with greatest variance/cost ratio, until desired accuracy is achieved
- results show QMC to be particularly effective on lowest levels with low dimensionality
MLQMC Results

GBM: European call

![Graph showing log2 variance and log2 mean vs. I for different simulations, including 1, 16, 256, and 4096.](image)

Multilevel Monte Carlo – p. 27
MLQMC Results

GBM: European call

![Graph showing MLQMC results for GBM European call]

- $\epsilon = 0.00005$
- $\epsilon = 0.0001$
- $\epsilon = 0.0002$
- $\epsilon = 0.0005$
- $\epsilon = 0.001$

Cost

$\epsilon^2$ Cost

- Std QMC
- MLQMC

Multilevel Monte Carlo – p. 28
MLQMC Results

GBM: barrier option

\begin{align*}
\log_2 \text{variance} &\quad \log_2 |\text{mean}| \\
\text{1} &\quad \text{16} &\quad \text{256} &\quad \text{4096}
\end{align*}

Multilevel Monte Carlo – p. 29
MLQMC Results

GBM: barrier option

![Graph showing GBM barrier option results with various ε values and MLQMC versus Std QMC comparisons.](image)
This research (and almost all of the presentation) comes from 2007.

In May-June 2015, Adrien Grumberg repeated everything using Sobol points with Matousek–Owen digital scrambling.

The results were very similar – if anything, the Sobol results were slightly better.


people.maths.ox.ac.uk/gilesm/files/Adrien_Grumberg.pdf
Conclusions

- Initial MLQMC research came very soon after MLMC.
- Numerical results were very encouraging, but there was no supporting numerical analysis.
- Frances Kuo and Ian Sloan later developed MLQMC for SPDEs, in collaboration with Rob Scheichl, Christoph Schwab, and others – also made good progress on the numerical analysis.

Future:
- More experiments
- New applications (e.g. continuous-time Markov processes)
- More numerical analysis


people.maths.ox.ac.uk/gilesm/mlmc.html
people.maths.ox.ac.uk/gilesm/acta/
people.maths.ox.ac.uk/gilesm/mlmc_community.html