Multilevel Monte Carlo methods using approximate distributions

Mike Giles
Oliver Sheridan-Methven

Mathematical Institute, University of Oxford

Algorithms and Complexity for Continuous Problems

Dagstuhl Seminar 19341

August 19 - 23, 2019

Monte Carlo

To estimate $\mathbb{E}[P]$, standard Monte Carlo simulation uses the average of N i.i.d. samples

$$N^{-1} \sum_{n=1}^{N} P^{(n)}$$

To achieve a RMS error of ε requires $N \approx \varepsilon^{-2} V$ samples, where $V = \mathbb{V}[P]$ is the variance.

If each sample costs C then the total cost is approximately $\varepsilon^{-2} V\, C$.

Two-level Monte Carlo

If $\widetilde{P} \approx P$, then since $\mathbb{E}[P] = \mathbb{E}[\widetilde{P}] + \mathbb{E}[P - \widetilde{P}]$ we can instead use

$$N_0^{-1} \sum_{n=1}^{N_0} \widetilde{P}^{(n)} + N_1^{-1} \sum_{n=1}^{N_1} (P^{(n)} - \widetilde{P}^{(n)}).$$

The cost of this estimator is $N_0 C_0 + N_1 C_1$, and the variance is $V_0/N_0 + V_1/N_1$, where $V_0 \equiv \mathbb{V}[\widetilde{P}]$, $V_1 \equiv \mathbb{V}[P - \widetilde{P}]$.

Minimising the cost subject to the same accuracy requirement gives the total cost

$$\varepsilon^{-2} (\sqrt{V_0 C_0} + \sqrt{V_1 C_1})^2.$$

If $C_0 = 10^{-1} C$, $C_1 = C$, $V_0 = V$, $V_1 = 10^{-3} V$ then the total cost is $0.121\, \varepsilon^{-2} V\, C$, a factor 8 savings compared to the original Monte Carlo.

Multilevel Monte Carlo

Given a sequence of increasingly accurate (and costly) approximations $\widehat{P}_0, \widehat{P}_1, \widehat{P}_2, \ldots \to P$, for example from the approximation of an SDE using 2^ℓ timesteps on level ℓ, then

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

and so the MLMC estimate is

$$N_0^{-1} \sum_{n=1}^{N_0} \widehat{P}_0^{(n)} + \sum_{\ell=1}^{L} N_\ell^{-1} \sum_{n=1}^{N_\ell} (\widehat{P}_\ell^{(\ell,n)} - \widehat{P}_{\ell-1}^{(\ell,n)}),$$

and the total cost ends up being approximately

$$\varepsilon^{-2} \left(\sum_{\ell=0}^{L} \sqrt{V_\ell C_\ell} \right)^2$$

Approximate random variables

In some applications, generating the random numbers can be a significant cost, especially with QMC when inverting the CDF.

e.g. Poisson distribution, increments of a Lévy process, non-central chi-squared distribution (CIR model)

Even with Normal random variables, cost of conversion from uniform r.v. to Normal is non-trivial for vector implementation.

This has led to previous research:

- Müller, Scheichl, Shardlow (2015) – 3 or 4-point approximations, leading to telescoping sum error which needs to be controlled
- Belomestny & Nagapetyan (2017) – different approximations on different levels
- G, Hefter, Mayer, Ritter (2019) – random bit approximations (IBC setting)

Approximate random variables

Simplest example: Euler-Maruyama approximation of a scalar SDE:

$$\widehat{X}_{t_{m+1}} = \widehat{X}_{t_m} + a(\widehat{X}_{t_m})\, h + b(\widehat{X}_{t_m})\sqrt{h}\, Z_m$$

where Z_m is a unit Normal r.v. generated as

$$Z_m = \Phi^{-1}(U_m)$$

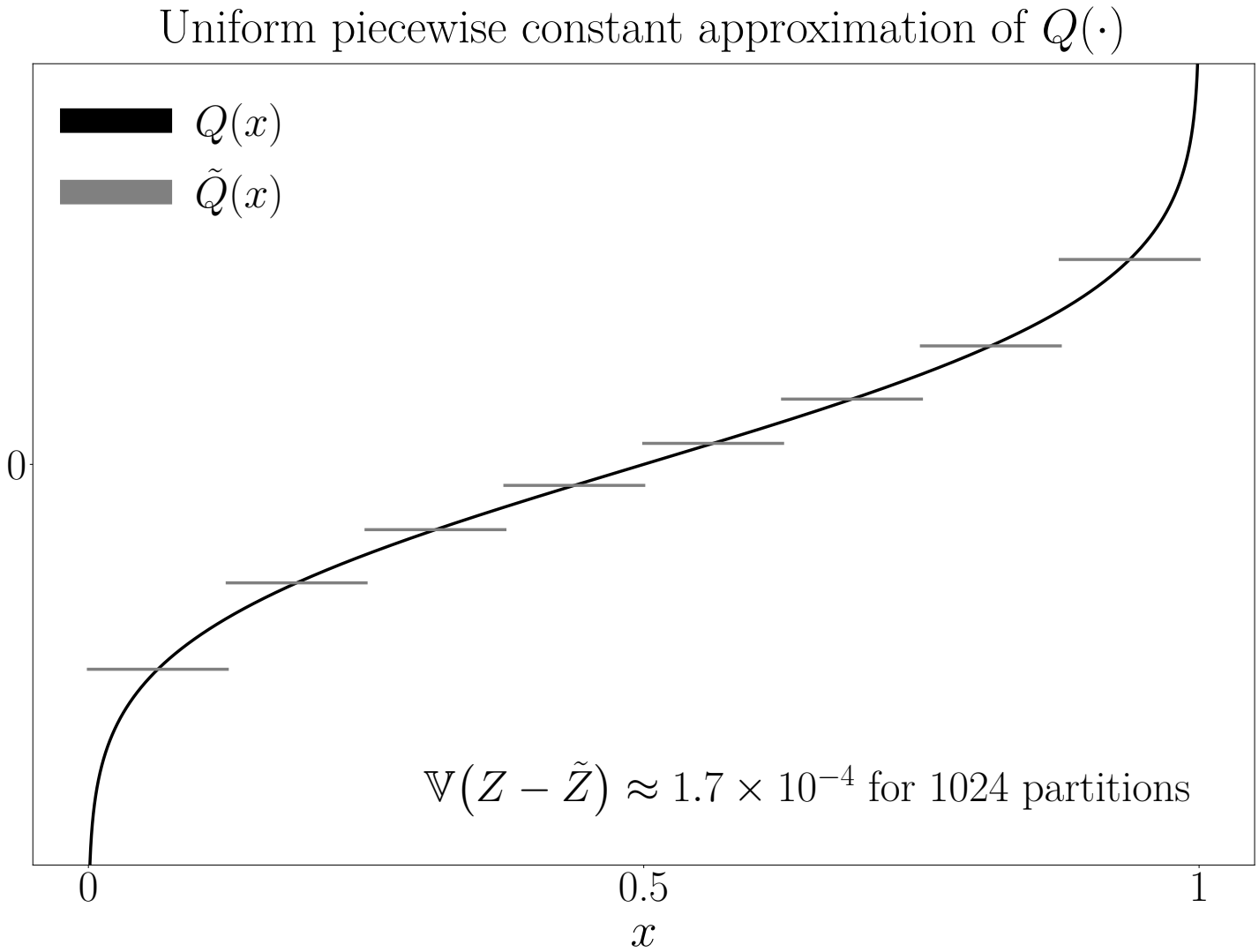
where U_m is a uniform $(0,1)$ r.v. and Φ^{-1} is the inverse Normal CDF.

Suppose instead we use approximate Normals $\widetilde{Z}_m$ generated by

$$\widetilde{Z}_m = \widetilde{Q}(U_m)$$

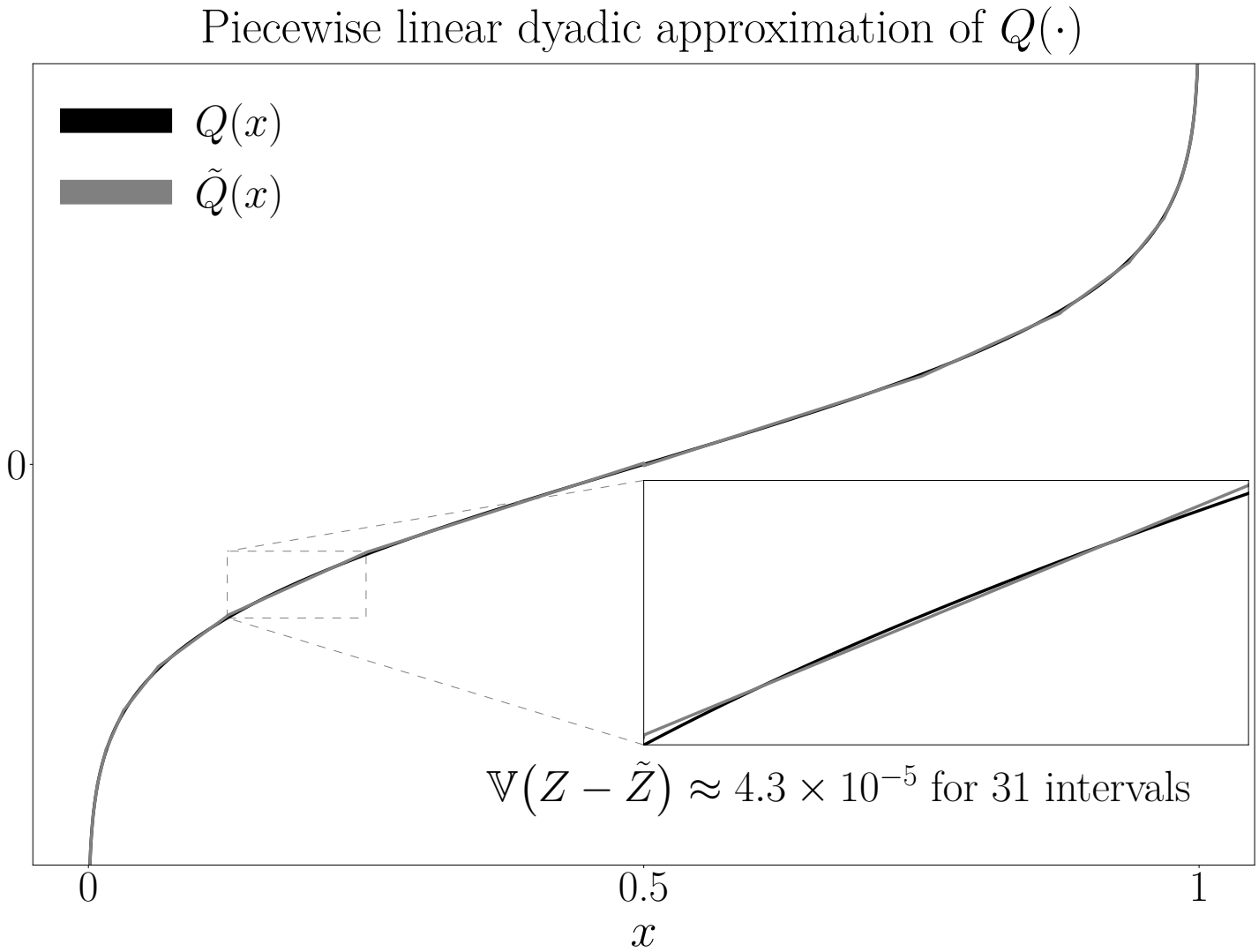
where $\widetilde{Q}$ is an approximation to Φ^{-1}.

Approximate random variables



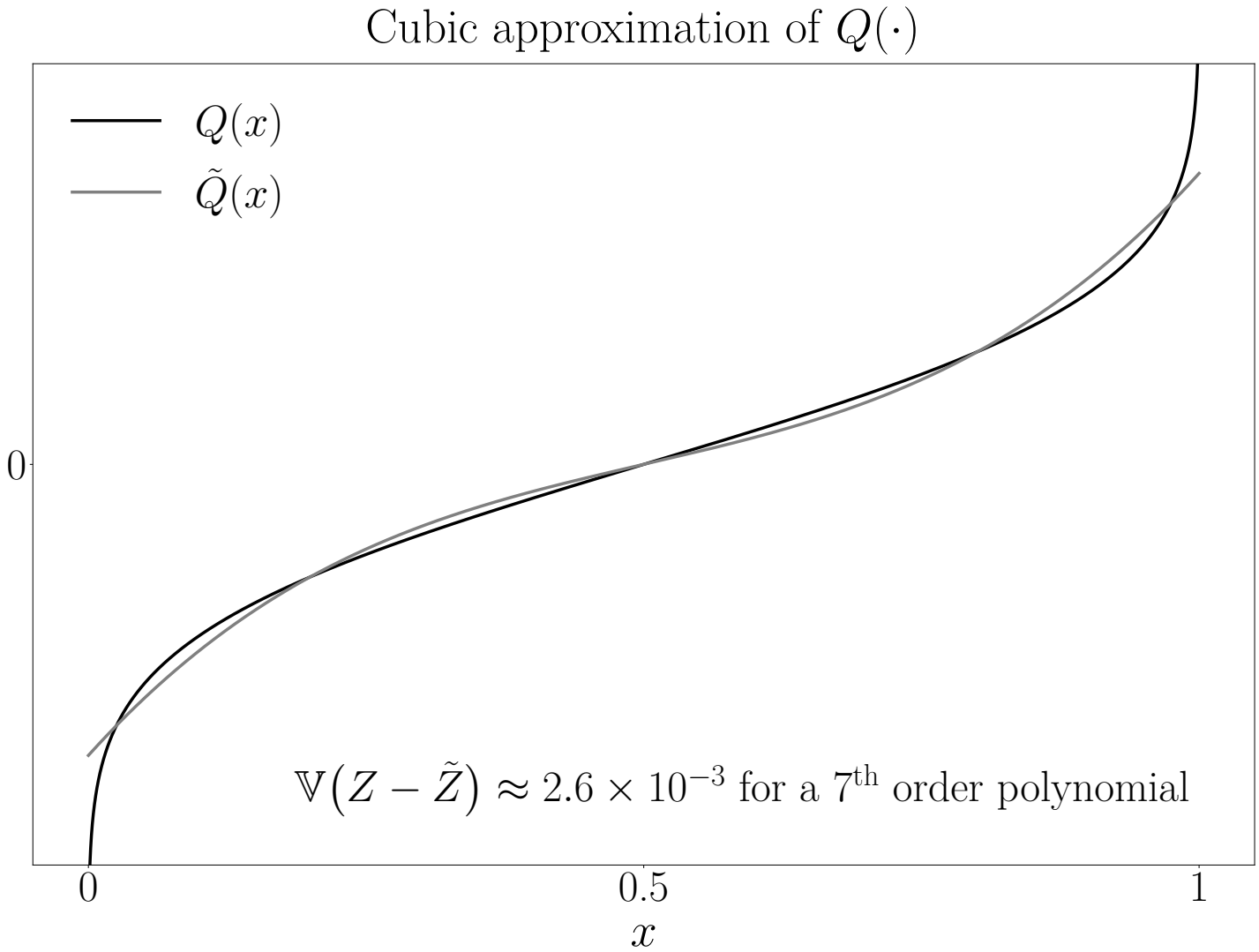
Good for scalar execution on CPUs – lookup in L1 cache

Approximate random variables



Good for vector execution on CPUs – lookup within a vector

Approximate random variables



Good for half-precision on GPUs – no lookup

Generation of uniform random variables

In randomised QMC, sometimes generate points as

$$U^{(m,n)} = S^{(m)} \wedge R^{(n)}$$

where $S^{(m)}$ is a set of Sobol points in $[0,1)^d$, $R^{(n)}$ is a set of pseudo-random points in $[0,1)^d$, and $\wedge$ is a bitwise exclusive-OR operator.

The same approach works for $S^{(m)}$ being pseudo-random points, and gives a point set $\{U^{(m,n)}\}$ with pairwise independence, which is all that is needed for standard variance analysis.

This effectively eliminates the cost of uniform random number generation – two sets of 1000 random numbers give rise to 10^6 random numbers at an average cost of less than 2 operations each.

This is a variant of a trick due to Bakhvalov (1964).

Approximate random variables

For an SDE approximation with a specified number of timesteps, a two-level MLMC approach gives

$$\mathbb{E}[\widehat{P}] \;=\; \mathbb{E}[\widetilde{P}] \;+\; \mathbb{E}[\widehat{P}-\widetilde{P}]$$

Analysis (G, Hefter, Mayer, Ritter, 2019) proves that for $P \equiv f(X_T)$ for a Lipschitz $f(x)$,

$$\mathbb{V}[\widehat{P}-\widetilde{P}] = O\left(\mathbb{E}[\,|\widetilde{Z}-Z|^2]\right)$$

so this can lead to significant savings if we have both

- $\mathbb{E}[\,|\widetilde{Z}-Z|^2] \ll 1$
- $\widetilde{Z}_m \equiv \widetilde{Q}(U_m)$ is much cheaper to evaluate than $Z_m \equiv \Phi^{-1}(U_m)$

Approximate random variables

How does this work in combination with standard timestepping MLMC?

Answer: nested MLMC

$$\begin{aligned}
\mathbb{E}[\widehat{P}_L] &= \mathbb{E}[\widehat{P}_0] + \sum_{\ell=1}^{L} \mathbb{E}[\widehat{P}_\ell - \widehat{P}_{\ell-1}] \\
&= \mathbb{E}[\widetilde{P}_0] + \mathbb{E}[\widehat{P}_0 - \widetilde{P}_0] \\
&+ \sum_{\ell=1}^{L} \left\{ \mathbb{E}[\widetilde{P}_\ell - \widetilde{P}_{\ell-1}] + \mathbb{E}\left[(\widehat{P}_\ell - \widehat{P}_{\ell-1}) - (\widetilde{P}_\ell - \widetilde{P}_{\ell-1}) \right] \right\}
\end{aligned}$$

The pair $(\widetilde{P}_\ell, \widetilde{P}_{\ell-1})$ are generated in the same way as $(\widehat{P}_\ell, \widehat{P}_{\ell-1})$, just replacing exact Z_m by approximate $\widetilde{Z}_m$, for same underlying U_m

MIMC compared to nested MLMC

A key aspect of the standard MLMC for SDEs is that the Brownian increment for a timestep of $2h$ is equal to the sum of Brownian increments for two timesteps of size h.

Another way of putting this is that if Z_1, Z_2 are unit Normal r.v.'s, then

$$\sqrt{h}\, Z_1 + \sqrt{h}\, Z_2 = \sqrt{2h}\, Z_3$$

where Z_3 is also a unit Normal r.v.

However, if $\widetilde{Z}_1, \widetilde{Z}_2$ are approximate unit Normal random variables, and

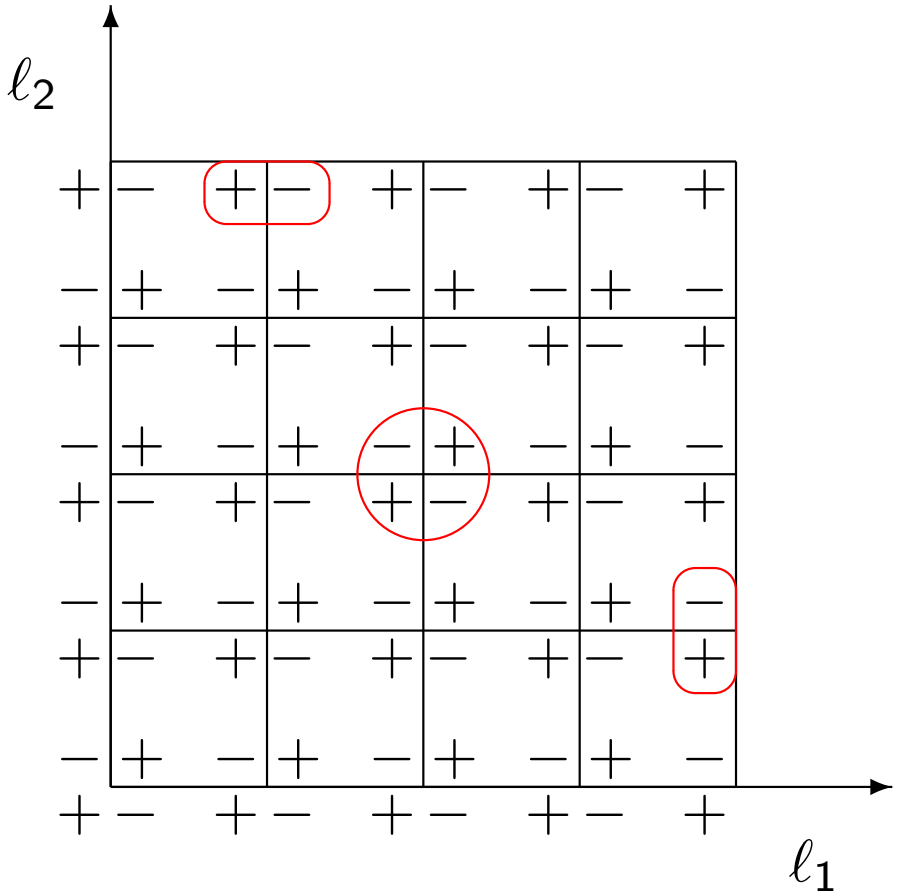
$$\sqrt{h}\, \widetilde{Z}_1 + \sqrt{h}\, \widetilde{Z}_2 = \sqrt{2h}\, \widetilde{Z}_3$$

then $\widetilde{Z}_3$ is an approximate Normal r.v. from a different distribution.

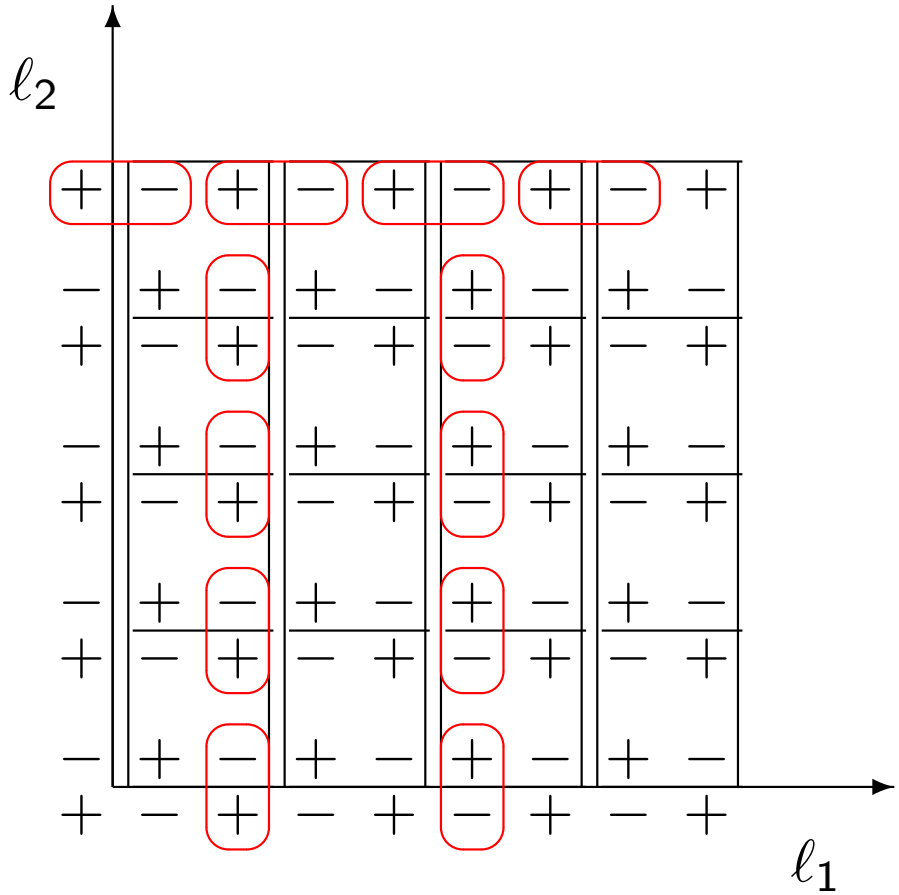
This is why we use the nested MLMC approach and not MIMC.

MIMC compared to nested MLMC

MIMC: 4-way cancellation



nested MLMC: 2-way cancellation



Numerical analysis

Assumption

The drift function $a : \mathbb{R} \to \mathbb{R}$ *and volatility function* $b : \mathbb{R} \to \mathbb{R}$ *are both* $C^1(\mathbb{R})$*, and both they and their derivatives are Lipschitz continuous so that there exist constants* L_a, L_b, L'_a, L'_b *such that*

$$|a(x)-a(y)| \le L_a|x-y|, \qquad |b(x)-b(y)| \le L_b|x-y|,$$
$$|a'(x)-a'(y)| \le L'_a|x-y|, \qquad |b'(x)-b'(y)| \le L'_b|x-y|.$$

Assumption

Random pairs $(Z, \widetilde{Z})$ *can be generated such that* $Z \sim N(0,1)$, $\mathbb{E}[\widetilde{Z}] = 0$, *and* $\mathbb{E}[\,|\widetilde{Z}-Z|^p] \le \mathbb{E}[\,|Z|^p]$, *for all* $p \ge 2$.

Numerical analysis

Lemma

At the terminal time T, for any $2 \le p < q$

$$\mathbb{E}\left[\left|\widehat{X}_\ell - \widehat{X}_{\ell-1}\right|^p\right] \prec h_\ell^{p/2}$$

$$\mathbb{E}\left[\left|\widetilde{X}_\ell - \widetilde{X}_{\ell-1}\right|^p\right] \prec h_\ell^{p/2}$$

$$\mathbb{E}\left[\left|\widetilde{X}_\ell - \widehat{X}_\ell\right|^p\right] \prec \mathbb{E}[\,|\widetilde{Z} - Z|^p]$$

$$\mathbb{E}\left[\left|(\widetilde{X}_\ell - \widetilde{X}_{\ell-1}) - (\widehat{X}_\ell - \widehat{X}_{\ell-1})\right|^p\right] \prec h_\ell^{p/2}\left(\mathbb{E}[\,|\widetilde{Z} - Z|^q]\right)^{p/q}$$

Proof: discrete time Burkholder-Davis-Gundy + Grönwall inequalities plus Mean Value Theorem and a generalisation

Numerical analysis

Lemma (Mean Value Theorem)

If $f : \mathbb{R} \to \mathbb{R}$ is $C^1(\mathbb{R})$, then there exists ξ which is a positively-weighted average of x_1, x_2 (i.e., $\xi = s\,x_1 + (1-s)x_2$ for some $0<s<1$) such that

$$f(x_1) - f(x_2) = (x_1 - x_2)\, f'(\xi).$$

Lemma

If $f : \mathbb{R} \to \mathbb{R}$ is $C^1(\mathbb{R})$, and f' is Lipschitz continuous with Lipschitz constant L'_f then there exists ξ which is a positively-weighted average of x_1, x_2, x_3, x_4 such that

$$f(x_1) - f(x_2) - f(x_3) + f(x_4) = (x_1 - x_2 - x_3 + x_4)\, f'(\xi) + R,$$

where

$$|R| \le \tfrac{1}{2} L'_f \left(|x_1 - x_2| + |x_3 - x_4|\right) \left(|x_1 - x_3| + |x_2 - x_4|\right).$$

Numerical analysis

If the output quantity of interest is $\mathbb{E}[f(X_T)]$, then the first result concerning the MLMC variance is the following.

Lemma

If $f : \mathbb{R} \to \mathbb{R}$ *is* $C^1(\mathbb{R})$ *and there are* $r, L_f, L'_f > 0$ *such that*

$$\begin{aligned} |f(x) - f(y)| &\leq L_f \left(1 + |x|^r + |y|^r\right) |x-y|, \\ |f'(x) - f'(y)| &\leq L'_f \left(1 + |x|^r + |y|^r\right) |x-y|, \end{aligned}$$

then for any $q > 2$

$$\mathbb{V}\left[(f(\widehat{X}_\ell) - f(\widehat{X}_{\ell-1})) - (f(\widetilde{X}_\ell) - f(\widetilde{X}_{\ell-1}))\right] \prec h_\ell \left(\mathbb{E}[\,|\widetilde{Z} - Z|^q]\right)^{2/q}.$$

However, financial put/call options are less regular, so for those ...

Numerical analysis

Lemma

If $f : \mathbb{R} \to \mathbb{R}$ *is* $C^1(\mathbb{R}\backslash K)$ *and there is are* $r, L_f, L'_f > 0$ *such*

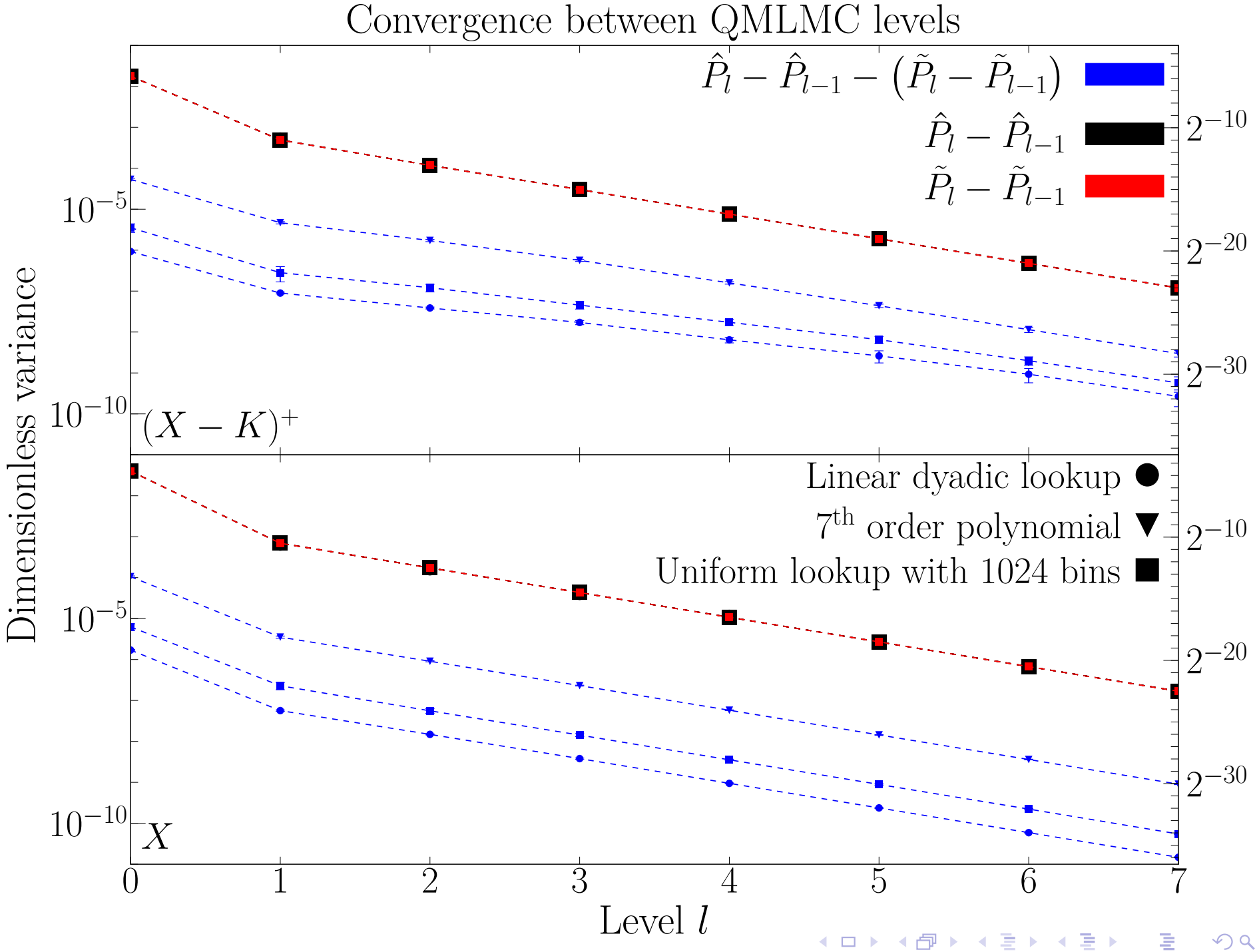
$$|f(x) - f(y)| \le L_f \left(1 + |x|^r + |y|^r\right) |x - y|, \qquad \text{for all } x, y$$
$$|f'(x) - f'(y)| \le L'_f \left(1 + |x|^r + |y|^r\right) |x - y|, \qquad \text{if } x > y > K \text{ or } x < y < K$$

and furthermore for all $D > 0$

$$\mathbb{P}[\,|X_T - K| < D] \;\prec\; D.$$

Then for any $q > 2$ *and any* $\delta > 0$,

$$\mathbb{V}\left[(f(\widehat{X}_\ell) - f(\widehat{X}_{\ell-1})) - (f(\widetilde{X}_\ell) - f(\widetilde{X}_{\ell-1}))\right]$$
$$\prec \quad \min\left\{ h_\ell \left(\mathbb{E}[\,|\widetilde{Z} - Z|^q]\right)^{(1-\delta)/(q+1)},\; h_\ell^{(1-\delta)/2 - 1/q} \left(\mathbb{E}[\,|\widetilde{Z} - Z|^q]\right)^{2/q} \right\}$$


Convergence between QMLMC levels
$\hat{P}_l - \hat{P}_{l-1} - (\tilde{P}_l - \tilde{P}_{l-1})$
$\hat{P}_l - \hat{P}_{l-1}$
$\tilde{P}_l - \tilde{P}_{l-1}$
$(X-K)^+$
X
Linear dyadic lookup
7^{th} order polynomial
Uniform lookup with 1024 bins
Dimensionless variance
10^{-5}
10^{-10}
2^{-10}
2^{-20}
2^{-30}
0
1
2
3
4
5
6
7
Level l

Reduced precision arithmetic

Further computational savings can be achieved by using reduced precision arithmetic.

Previous research at TU Kaiserslautern (Korn, Ritter, Wehn and others) and Imperial College (Luk and others) has used FPGAs with complete control over the precision, but we prefer GPUs.

In the latest NVIDIA GPUs, half-precision fp16 is twice as fast as single precision fp32 (which is 2-8 times faster than double precision fp64).

Future Intel CPUs will support bfloat16 at twice the performance of fp32, and 4 times the performance of fp64.

In most cases, single precision is perfectly sufficient for calculating $\widehat{P}$; half precision can be used for $\widetilde{P}$.

MC averaging should probably be done in double precision in both cases.

Reduced precision arithmetic

Very important: to ensure the telescoping sum is respected, must ensure that **exactly** the same computations are performed for $\widetilde{P}$ whether on its own or in calculating $\widehat{P} - \widetilde{P}$.

The effect of half-precision arithmetic can be modelled as

$$\widetilde{X}_{t_{m+1}} = \widetilde{X}_{t_m} + a(\widetilde{X}_{t_m})h + b(\widetilde{X}_{t_m})\sqrt{h}\,\widetilde{Z}_m + \delta\,\widetilde{X}_{t_m} V_m$$

where $\delta \approx 10^{-3}$ and the V_m are iid unit variance random variables.

Overall, leads to an $O(\delta^2/h)$ increase in the variance; if this increases it too much on finer levels, the reduced precision should not be used.

Conclusions / future work

- nested MLMC is similar to MIMC, but more general
- helpful in using approximate distributions and reduced precision
- offers significant computational savings in some situations
- in future, perform experiments with reduced precision, and extend to other distributions:
  - Poisson
  - binomial
  - non-central chi-squared (CIR process)
- also extend to MLQMC

References

MBG. 'Multilevel Monte Carlo methods'. *Acta Numerica*, 24:259-328, 2015.

MBG, F.Y. Kuo, I.H. Sloan. 'Combining sparse grids, multilevel MC and QMC for elliptic PDEs with random coefficients'. *Monte Carlo and Quasi-Monte Carlo Methods 2016*, Springer, 2018.

MBG, M. Hefter, L. Mayer, K. Ritter. 'Random bit quadrature and approximation of distributions on Hilbert spaces'. *Foundations of Computational Mathematics*, 19(1):205-238, 2019.

MBG, M. Hefter, L. Mayer, K. Ritter. 'Random bit multilevel algorithms for stochastic differential equations'. *Journal of Complexity*, 54:101395, 2019.