Multilevel Monte Carlo (MLMC)

May 16, 2025

1 Introduction

Monte Carlo (MC) methods are widely used to estimate expectations $\mathbb{E}[P]$ when the underlying distribution is complicated or unknown analytically. However, standard MC can be computationally expensive, especially when high accuracy is required.

Multilevel Monte Carlo (MLMC) methods, introduced by Mike Giles in 2008, improve the efficiency of Monte Carlo simulations by combining multiple levels of discretization, significantly reducing the computational cost while maintaining a given accuracy.

2 Principle of MLMC

Instead of approximating $\mathbb{E}[P]$ directly, MLMC decomposes it into a sum of corrections between levels:

$$\mathbb{E}[P_L] = \mathbb{E}[P_0] + \sum_{l=1}^{L} \mathbb{E}[P_l - P_{l-1}] \tag{1}$$

where:

- P_l is the estimator at level l (typically, finer discretization means larger l).
- P_0 is a very coarse, cheap approximation.
- $P_l - P_{l-1}$ is a "correction" between two consecutive levels.

Each term is estimated independently with a Monte Carlo estimator.

3 Variance and Cost Balancing

The key idea is to:

- Use **many cheap samples** at coarse levels (where variance is high but cost is low), and
- Use **fewer expensive samples** at fine levels (where variance is small but cost is high).

Thus, the MLMC estimator is:

$$\hat{P}^{\text{MLMC}} = \sum_{l=0}^{L} \frac{1}{N_l} \sum_{i=1}^{N_l} (P_l^{(i)} - P_{l-1}^{(i)}) \tag{2}$$

where N_l is the number of samples at level l.

4 MLMC Complexity

Let ε is the target root mean square (RMS) error. Suppose:

- The bias $\mathbb{E}[P_L - P] \sim \mathcal{O}(2^{-\alpha L})$,
- The variance $\mathbb{V}[P_l - P_{l-1}] \sim \mathcal{O}(2^{-\beta l})$,
- The cost per sample $C_l \sim \mathcal{O}(2^{\gamma l})$.

Then the total cost to achieve RMS error ε satisfies:

$$\mathrm{Cost}_{\mathrm{MC}} \sim \varepsilon^{-3}, \quad \mathrm{Cost}_{\mathrm{MLMC}} \sim \begin{cases} \varepsilon^{-2} & \text{if } \beta > \gamma, \\ \varepsilon^{-2}(\log \varepsilon)^2 & \text{if } \beta = \gamma, \\ \varepsilon^{-2-\frac{\gamma-\beta}{\alpha}} & \text{if } \beta < \gamma. \end{cases} \tag{3}$$

Typically, for SDE discretizations (Euler scheme), $\alpha = 1$, $\beta = 1$, $\gamma = 1$ and MLMC reduces cost from ε^{-3} to ε^{-2}.

5 Algorithm Overview

1. Set a minimum level $L_{\min}$, maximum level $L_{\max}$, initial samples N_0, and desired tolerance ε.
2. Initialize sample statistics for each level: sums of Y_l and Y_l^2.
3. Estimate variances V_l and costs C_l for each level.
4. Allocate number of samples N_l optimally based on V_l, C_l, and ε.
5. If the bias (estimated from $\mathbb{E}[P_L - P_{L-1}]$) is too large, increase L.
6. Repeat until both the variance and bias criteria are satisfied.

6 Advantages of MLMC

- **Massive speedup**: MLMC drastically reduces computational cost for small tolerances.
- **Flexibility**: MLMC applies to any problem where coarse and fine simulations can be coupled.
- **Error control**: MLMC naturally balances bias and variance.

7 Example: European Call Option under Black-Scholes

- Under the risk-neutral measure, the asset price S_t follows a Geometric Brownian Motion:

  $$\mathrm{d}S_t = rS_t\,\mathrm{d}t + \sigma S_t\,\mathrm{d}W_t \tag{4}$$

  where:

  - r is the risk-free rate,
  - σ is the volatility,
  - W_t is a standard Brownian motion.

- The European call option payoff is: $P = e^{-rT}(S_T - K)^+$.
- MLMC estimates $\mathbb{E}[P]$ by simulating S_T under multiple time discretizations and computing corrections between coarse and fine approximations of the payoff.

Using MLMC, the cost to achieve an RMS error can be reduced by up to a factor of **10** compared to plain Monte Carlo. While the optimal complexity is $\mathcal{O}(\varepsilon^{-2})$ when the variance decays faster than the cost increases ($\beta > \gamma$), in the typical case where $\beta = \gamma = 1$, the complexity becomes $\mathcal{O}(\varepsilon^{-2}(\log \varepsilon)^2)$ — still significantly better than standard Monte Carlo's $\mathcal{O}(\varepsilon^{-3})$.

8 Conclusion

Multilevel Monte Carlo is a powerful method that intelligently reduces the variance and computational cost by mixing simulations at different accuracies.
It is particularly useful for financial engineering, uncertainty quantification, and SDE simulations.