Optimal Variance Reduction via Multilevel Monte Carlo (MLMC) Methods

Gary Pai

December 20, 2025

1 Introduction

Multilevel Monte Carlo (MLMC) is a variance reduction strategy designed to estimate the expected value of a random variable P when the exact simulation is computationally expensive. Instead of sampling only from a high-fidelity model, MLMC uses a hierarchy of models with increasing accuracy and cost. Multilevel Monte Carlo (MLMC) was significantly popularized by Giles [1], who demonstrated that for path-dependent options, the computational complexity could be reduced from $O(\epsilon^{-3})$ to $O(\epsilon^{-2})$.

2 The Telescoping Sum

The fundamental identity of MLMC is the telescoping sum. Given a sequence of approximations $P_0, P_1, \ldots, P_L$ with increasing levels of refinement (where P_L is the finest/most accurate), the expectation $E[P_L]$ is decomposed as:

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

Here, $E[P_0]$ is the base estimate on a coarse grid, and each term $E[P_\ell - P_{\ell-1}]$ represents a **correction** that accounts for the bias between levels.

3 The Logic of the Correction

The "Estimate of the Correction" is the core innovation of MLMC. In your example:

- **Level 0 (P_0):** 10-step simulation (Cheap, High Bias).
- **Level 1 (P_1):** 1000-step simulation (Expensive, Low Bias).

By **coupling** the paths—using the same random underlying increments for both the 10-step and 1000-step versions— the difference $P_1 - P_0$ becomes highly correlated.

3.1 Variance Reduction

Because the paths are coupled, the variance of the difference, $Var(P_\ell - P_{\ell-1})$, is significantly smaller than the variance of the individual level $Var(P_\ell)$. Since the required number of samples N to achieve a target error ϵ is proportional to the variance, we only need a very small number of expensive samples at level L to estimate the correction accurately.

4 Optimal Sample Allocation

To minimize the total computational cost for a fixed variance, we use the Neyman Allocation to determine the number of samples N_ℓ at each level:

$$N_\ell \propto \sqrt{\frac{V_\ell}{C_\ell}} \tag{2}$$

Where:

- V_ℓ is the variance of the correction at level ℓ.
- C_ℓ is the cost per sample at level ℓ.

5 Comparison Summary

Feature	**Standard Monte Carlo**	**Multilevel Monte Carlo**
Strategy	Single high-resolution level	Hierarchy of resolutions
Primary Cost	High (many high-res samples)	Low (most samples are coarse)
Variance	$Var(P_L)$	$Var(P_\ell - P_{\ell-1}) \to 0$ as $\ell \to L$
Efficiency	$O(\epsilon^{-3})$ for paths	$O(\epsilon^{-2})$ (typically)

6 The Path Coupling

In a standard Monte Carlo simulation, a path is generated using independent random increments. In the Multilevel Monte Carlo (MLMC) framework, the "Estimate of the Correction" relies on **Coupling**. This ensures that the coarse model (P_0) and the fine model (P_1) are driven by the same underlying stochastic process.

7 The Simulation Mechanism

Consider the simulation of a Stochastic Differential Equation (SDE) over a time interval $[0, T]$.

7.1 The Fine Path (P_1)

For the fine path, we divide the time interval into $M_{fine} = 1000$ steps. We generate a sequence of independent normal random variables:

$$\delta W_j \sim N(0, \Delta t_{fine}), \quad j = 1, \ldots, 1000 \tag{3}$$

The fine path P_1 is constructed by iterating through these 1000 small increments.

7.2 The Coarse Path (P_0)

For the coarse path, we use $M_{coarse} = 10$ steps. Instead of generating new randomness, we **couple** it to the fine path by summing the fine increments. Each coarse step ΔW_k is the sum of 100 fine increments:

$$\Delta W_k = \sum_{j=100(k-1)+1}^{100k} \delta W_j, \quad k = 1, \ldots, 10 \tag{4}$$

8 Estimating the Correction

The correction term is defined as the difference between the two estimators:

$$L_1 = E[P_1 - P_0] \tag{5}$$

Because the paths are coupled, they "track" each other throughout the simulation. If the fine path experiences a spike due to a specific sequence of δW_j, the coarse path will experience the same spike because it incorporates those exact same increments.

9 Statistical Implications

The power of this method lies in the variance of the difference. Let $V(X)$ denote variance:

- **Independent Sampling:** If P_1 and P_0 were independent, $V(P_1 - P_0) = V(P_1) + V(P_0)$, which is very large.
- **Coupled Sampling:** Because of high correlation, $V(P_1 - P_0) = V(P_1) + V(P_0) - 2Cov(P_1, P_0)$. As the paths converge, this variance approaches zero.

10 Computational Advantage

In this specific 10 vs. 1000 step example, the MLMC estimator allows us to:

1. Use a massive number of samples (e.g., 10^6) for the 10-step path to capture the general expectation.
2. Use a very small number of samples (e.g., 10^2) for the 1000-step correction to capture the discretization refinement.

The resulting estimate has the accuracy of a 1000-step simulation but the computational cost is dominated by the 10-step simulation.

11 Optimal Resource Allocation and Complexity Analysis in MLMC

To optimize a Multilevel Monte Carlo simulation, we must solve a constrained optimization problem: *How can one achieve the lowest possible variance for a fixed computational budget (time or money)?*

In the specific case of a 10-step (P_0) versus a 1000-step (P_1) simulation, the sample sizes are not chosen arbitrarily. Instead, they are determined by the **Neyman Allocation** (also known as the optimal allocation).

12 The Optimal Allocation Formula

To achieve a target accuracy ϵ, the optimal number of samples N_ℓ at each level ℓ is given by:

$$N_\ell = \mu \sqrt{\frac{V_\ell}{C_\ell}} \tag{6}$$

Where:

- V_ℓ: The **Variance** of the correction (the fluctuations of the difference between levels).
- C_ℓ: The **Cost** per sample (computational time required for one sample).
- μ: A constant (Lagrange multiplier) determined by the total accuracy requirement.

13 Numerical Application: 10-step vs. 1000-step

Consider a scenario with the following hypothetical data obtained from a preliminary test run:

- **Level 0 (Base, 10 steps):** $C_0 = 1$ second, $V_0 = 100$ (High variance as it is the raw value).
- **Level 1 (Correction, 10 vs 1000 steps):** $C_1 = 100$ seconds, $V_1 = 0.01$ (Low variance due to path coupling).

Calculating the ratio of samples (N_1/N_0):

$$\frac{N_1}{N_0} = \frac{\sqrt{V_1/C_1}}{\sqrt{V_0/C_0}} = \sqrt{\frac{0.01/100}{100/1}} = \sqrt{\frac{0.0001}{100}} = \sqrt{0.000001} = 0.001 \tag{7}$$

Conclusion: For every 1,000 inexpensive simulations performed at Level 0, only 1 expensive correction simulation is required at Level 1 to maintain the desired precision.

14 Visualizing the Work Distribution

In standard Monte Carlo, the computational effort is represented as a large rectangle (high volume at high cost). In MLMC, the effort follows a **pyramid structure**: the base level contains the vast majority of samples (high volume, low cost), while the top level consists of very few samples (low volume, high cost).

15 Complexity Shift

The computational complexity of standard Monte Carlo for path simulations is typically $O(\epsilon^{-3})$. MLMC significantly improves this scaling:

- **Standard MC:** $O(\epsilon^{-3})$
- **MLMC:** $O(\epsilon^{-2})$ or $O(\epsilon^{-2}(\log \epsilon)^2)$

If the required accuracy is increased by a factor of 10 ($\epsilon \to 0.1\epsilon$):

- Standard Monte Carlo takes **1,000x** longer.
- MLMC takes only **100x** longer.

16 Summary

The "estimate of the correction" is powerful because it allows us to shift the bulk of the computational burden to the coarsest possible model. We only use the "expensive" model to figure out the bias of the cheap model, not to explore the entire randomness of the system.

17 Reference

References

[1] M. B. Giles. *Multilevel Monte Carlo Path Simulation*. Operations Research, 56(3):607–617, 2008.