Monte Carlo Simulation Project

Jiajun Bao

University of Chicago

Abstract

This project explores the Metropolis Adjusted Langevin Algorithm (MALA) and compares it with the Random Walk Metropolis-Hastings (RWMH) in approximating Gaussian, multimode, and high-dimensional probability distribution settings. The project also studies the importance of optimal discretization step size in MALA.

1 Introduction and Overview

This project aims to introduce and explore the Metropolis Adjusted Langevin Algorithm (MALA) and compare it with Random Walk Metropolis-Hastings (RWMH). MALA is the metropolized version Unadjusted Langevin Algorithm (ULA). ULA can be viewed as the Euler discretization of the continuous Langevin dynamics without Metropolis-Hastings steps. MALA incorporates gradient information into the proposal, aiming for a more efficient exploration of the target distribution. RWMH is a version of Metropolis-Hastings algorithms, where proposal distributions are centered at the current state, leading to a random walk. These Monte Carlo Markov Chain(MCMC) algorithms serve as important tools for generating samples from complex, high-dimensional probability distributions, which are widely used in various fields, including statistics, chemistry, etc.

We first introduce the theoretical backgrounds, particularly in Stochastic Differential Equations, diffusion processes, Fokker-Planck Equation, and Euler-Maruyama method. We present the MALA algorithm and provide examples demonstrating its capacity to approximate Gaussian, multimodal, and high-dimensional distributions. An important point of our study is the determination of an optimal discretization step size (Δt) for MALA, as it is crucial to the efficiency and efficacy of the algorithm. Through experimentation and comparison, this research aims to provide comprehensive insights into the advantages and limitations of the MALA method.

2 Theoretical Background

2.1 Stochastic Differential Equations (SDEs) and Diffusion Processes

Stochastic Differential Equations (SDEs) are differential equations in which one or more of the terms is a stochastic process. The Langevin diffusion is described by the following stochastic differential equation:[3]

$$dX(t) = -\nabla V(X(t))dt + \sqrt{2\beta^{-1}}dW(t), X(0) = x_0 \tag{2.1}$$

where W(t) is a standard Wiener process, and V represents the potential energy for a particle system. This equation is often interpreted as a physical system of particles in which the noise comes from fluctuations due to a solvent. We set $\beta = 1$ in this project to simplify notation. Under mild regularity conditions on $V(x)$ (e.g., $V(x)$ grows to ∞ as $|x|$ approaches to ∞), this diffusion process is ergodic and has an invariant distribution $\pi(x) \propto \exp(-V(x))$.[5]. This indicates regardless of the initial distribution π_0, for large t, $X(t) \overset{approx.}{\sim} \pi(x)$. This suggests a natural way to approximate expectations with respect to π:

$$I_\pi[h] = \int_{\mathbb{R}^d} h(x)\pi(x)dx \approx \frac{1}{T}\int_0^T h(x(t))dt \tag{2.2}$$

2.2 Fokker-Planck Equation and Euler-Maruyama Method

The Fokker-Planck equation provides a deterministic evolution of the probability density function of a diffusion process and describes how the distribution evolves over time using a partial differential equation with the initial distribution, which has the form:

$$\partial_t \pi_t = \nabla \cdot (\pi_t \nabla V) + \Delta \pi_t, \pi(0) = \pi_0 \tag{2.3}$$

However, it is very difficult to solve (2.3) analytically. In practice, to simulate the continuous-time diffusion (2.1), a numerical integration scheme is needed. Here, we introduce the Euler-Maruyama Method, which is an extension of the Euler method for ordinary differential equations to stochastic differential equations, to generate sample paths and approximate the solutions to SDEs. The Euler-Muruyama discretization of the Langevin diffusion takes forms as follows:

$$X(t_k + \Delta t) = X(t_k) + \Delta t \nabla log f(x(t_k)) + \sqrt{(2\Delta t)}\xi^{t_k}, \xi^{t_k} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_d). \tag{2.4}$$

Similarly to what we observed in Hamiltonian Monte Carlo algorithm with the Leap Frog Integrator, the discretization of a continuous-time SDE with numerical integrators like Euler-Muruyama might break the ergodicity of the original system. Therefore the acceptance/rejection step is used to correct this error, which leads to the discussion in Metropolis Hastings Algorithm in Section 3. As we see later on, the discretization step size (Δt) for the Metropolis Adjusted Langevin Algorithm plays a very important role of its effectiveness.

3 Markov Chain Monte Carlo Development

3.1 The Metropolis Hastings Algorithm and Random Walk Metropolis Hastings

Before we introduce the Metropolis Adjusted Langevin Algorithm (MALA), we first briefly restate the Metropolis Hastings Algorithm and Random Walk Metropolis Hastings(RWMH) defined in [7], as the MALA algorithm is based on Metropolis Hasting. The probability of accepting a proposed move from $x \in E$ to $Z \in E$ is defined by

$$a(x, z) = min\{1, \frac{f(z)}{f(x)}\frac{q(z, x)}{q(x, z)}\} \tag{3.1}$$

Algorithm 3.1 The Metropolis Hastings Algorithm

```
1: **Input**: Target distribution $f$, initial distribution $\pi_0$, proposal Markov kernel $q(x, z)$
2: **Initial draw**: Sample $X^{(0)} \sim \pi_0$
3: **Subsequent Samples**: For n = 0, ..., N-1 do:
4: **Step 1**: Sample $Y^* \sim q(X^{(n)}, \cdot)$
5: **Step 2**:
       Update
       $$x^{(n+1)} = \begin{cases} Z^* & \text{w.p. a}(X^{(n)}, Z^*) \\ X^{(n)} & \text{w.p. 1-a}(X^{(n)}, Z^*) \end{cases}$$
6: **Output**: Samples $X^{(1)}, ..., X^{(N)}$
```

The Random Walk Metropolis Hastings algorithm (RWMH) is a type of Metropolis Hastings algorithm where the proposal kernel is chosen to be of the form: $q_{RWMH}(x, z) = g(z - x)$ for some distribution g such as normal, t-distribution, and uniform, which proposals are made according to a random walk.

3.2 Metropolis Adjusted Langevin Algorithm (MALA)

As discussed in Part 2.2, the Euler-Muruyama method provides a simple yet powerful numerical discretization scheme to generate a proposal for the next state of the Langevin diffusion. However, since the discretization step introduces bias into the proposed transitions, and this might break the ergodicity of the system, we need to apply the Metropolis-Hastings correction to decide whether to accept the proposal or stay at the current state. In here, the Markov kernel $q_{MALA}(x, z) = \mathcal{N}(z; x + \Delta t \nabla log f(x), 2\Delta t I_d)$ follows naturally from (2.4).

Algorithm 3.2 Metropolis Adjusted Langevin Algorithm (MALA)

```
Input: Target distribution $f$, initial distribution $\pi_0$, step size $\Delta t$
Initial draw: Sample $X^{(0)} \sim \pi_0$
Subsequent Samples: For n = 0, ..., N-1 do:
Proposal step:
        Generate a proposal for the next state using the Euler-Muruyama discretization
    of the Langevin diffusion:
        $X(t_k + \Delta t) = X(t_k) + \Delta t \nabla log f(x(t_k)) + \sqrt{(2\Delta t)}\xi^{t_k}$, $\xi^{t_k} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_d)$.
Acceptance step:
        Correct via the Metropolis Hastings Algorithm (Algorithm 3.1) with the pro-
    posal Markov kernel: $q_{MALA}(x, z) = \mathcal{N}(z; x + \Delta t \nabla log f(x), 2\Delta t I_d)$
Output: Samples $X^{(1)}, ..., X^{(N)}$
```

4 Algorithm Implementation and Results

4.1 Case Study 1: MALA and RWMH for Standard Gaussian Target Distribution

We initiate the exploration by considering a simple setup. The target distribution f is a standard normal distribution, i.e., $f = \mathcal{N}(0, 1)$. We generated 10,000 samples using both MALA and RWMH, based on Algorithms 3.1 and 3.2 we introduced earlier with the same initial state to ensure a fair comparison between the two methods.

For MALA, as illustrated in Figure 1, we investigated how varying the step size Δt affects the algorithm's performance. The step size in MALA is a crucial parameter determining each proposed move's magnitude in the state space. A small step size means the algorithm explores the target distribution with small steps, which may lead to an efficient exploration of a high-density area but a slow exploration of the overall state space. On the other hand, a large step size could facilitate broader exploration but also increase the chance of proposal rejection. Hence, understanding the impact of varying Δt allows us to tune this parameter for the optimal trade-off between exploration efficiency and acceptance probability.

For the RWMH algorithm, as illustrated in Figure 2, we varied the standard deviation parameter, sigma in proposal generation, $g_\sigma = \mathcal{N}(0, \sigma^2)$. The choice of sigma plays a vital role in the performance of RWMH. If sigma is small, the proposals tend to be close to the current state, potentially leading to a high acceptance rate but also possibly resulting in slow explorations. If sigma is large, the proposed state can be quite far from the current state, which might facilitate exploration of the state space, but also may result in a lower acceptance rate.

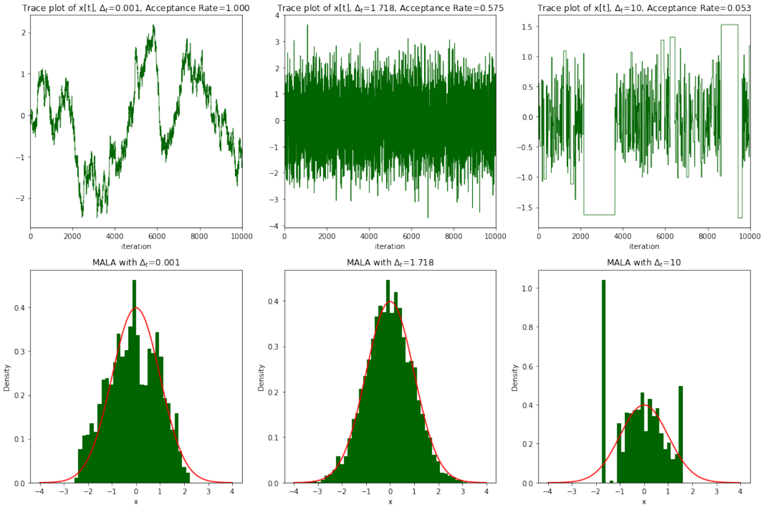

Figure 1 MALA for Standard Gaussian Target Distribution

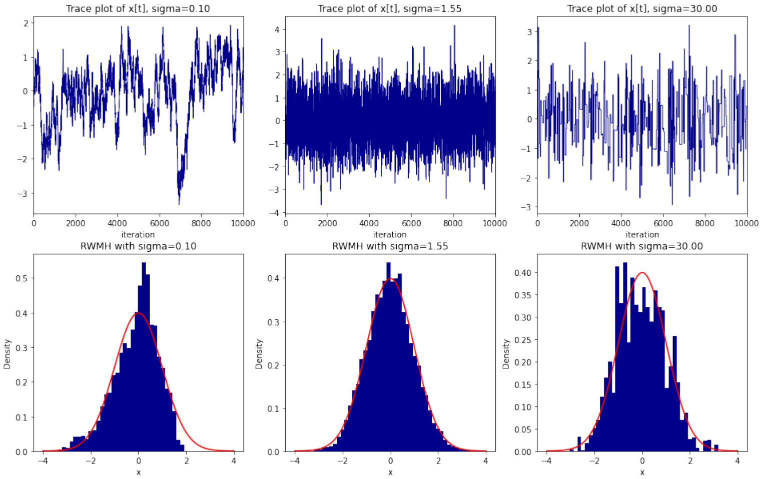

Figure 2 RWMH for Standard Gaussian Target Distribution

As explained in [4], the optimal acceptance rate for MALA is 0.574. To achieve this rate, we determine the optimal Δ_t value, as shown in Figure 3, which is approximately 1.718. This is also the value we used to generate the middle plot of Figure 1. As discussed in [6][7], In the case of a basic scenario where the target follows a normal distribution and the proposal is a normally distributed random walk, the optimal variance σ^2 can be determined by minimizing the integrated autocorrelation time of the associated chains, and the optimal value for σ^2 is 2.4. the estimation with the optimal value is presented in the middle plot of Figure 2, which corresponds to an acceptance rate of 0.44.

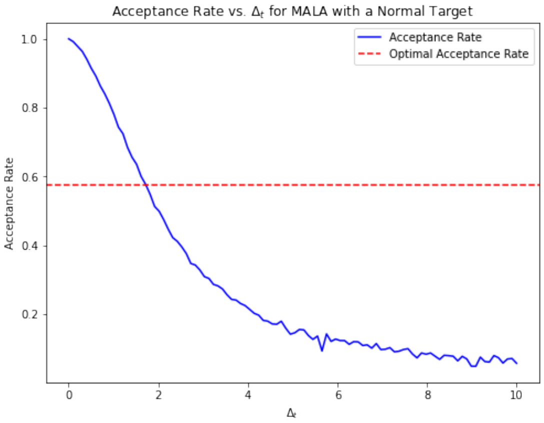

Figure 3 Acceptance Rate vs Δ_t for MALA with Normal Target

4.2 Case Study 2: MALA and RWMH for Multi-Mode Target Distribution

In this case study, we evaluated the MALA and RWMH algorithms' performance against a multi-modal (2 modes) target distribution $\pi(x) \propto \exp(-V(x))$, where $V(x) = x^4 - 3x^2 + 2$. We found both methods to be effective when parameters are well tuned. When the step size, Δt, was optimized in MALA, the algorithm demonstrated a balanced exploration strategy, with particles spending roughly equal amounts of time exploring each mode. However, when Δt was not optimally set, particularly when it was too small, MALA exhibited restricted exploration, tending to remain in one mode. This shows the importance of correctly tuning the step size for MALA to ensure comprehensive sampling. This is particularly crucial in multi-modal distributions.

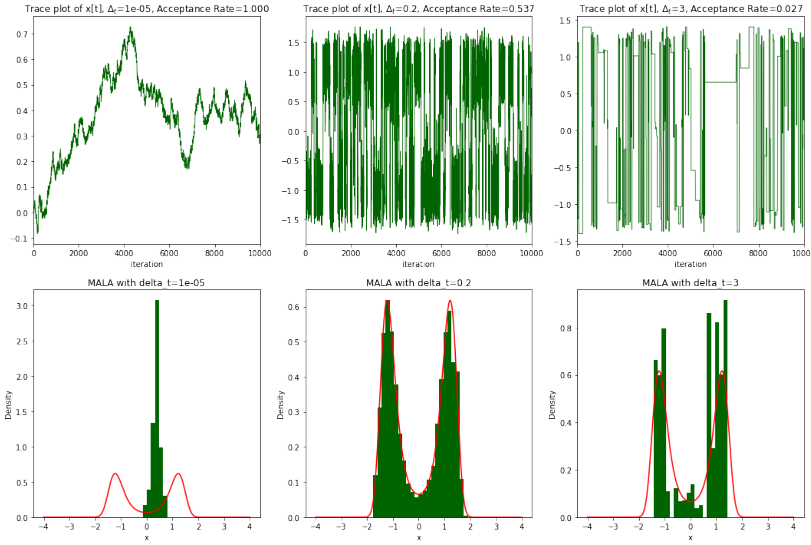

Figure 4 MALA for Target f with 2 Modes

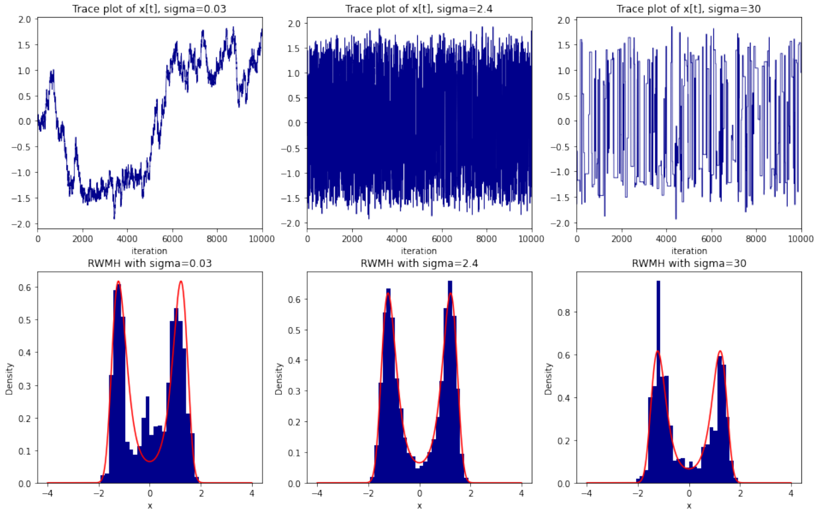

Figure 5 RMWH for Target f with 2 Modes

4.3 Case Study 3: MALA and RWMH for High-dimensional Target Distribution

In this case, we examine the effectiveness of MALA and RWMH in a high dimensional distribution setting. We analyze in the setting where the target f is a 10-dimensional Gaussian distribution given by: $f = \mathcal{N}(\mathbf{0}, \mathbf{I_d})$, where $\mathbf{x}$ is a 10-dimensional vector, $\mathbf{0}$ is a 10-dimensional zero vector, and $\mathbf{I}$ is the 10x10 identity matrix.

We first found the optimal Δ_t for MALA that has optimal acceptance rate for MALA is 0.574. Similarly, we found the optimal value for σ^2 that corresponds to the its optimal acceptance probability for higher dimension, which is roughly 0.234, as discussed in [4].

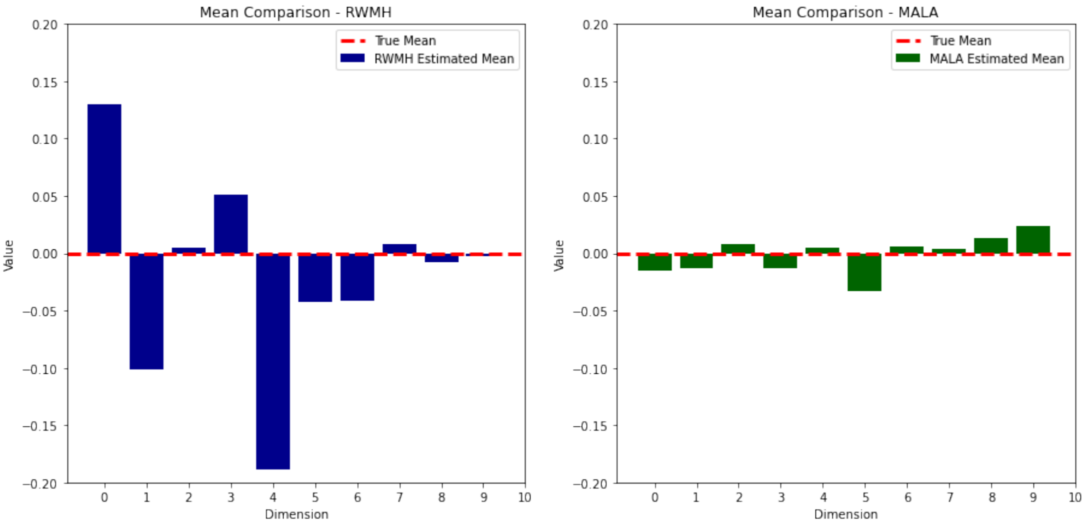

Figure 6 MALA and RWMH Mean Estimate for 10-Dimensional Gaussian Target f

Under identical settings and optimal parameters for each, as demonstrated in Figure 6, MALA demonstrates better performance to RWMH when estimating parameters of multi-dimensional distributions, such as a 10-dimensional standard Gaussian. The advantage of MALA arises from its use of gradient information in generating proposal states, which allows for more efficient exploration of the parameter space and accelerates convergence to the target distribution. This is especially advantageous in high-dimensional scenarios. However, similar to the Hamiltonian Monte Carlo algorithm, MALA necessitates the evaluation of the gradient of V, potentially leading to higher computational costs.

5 Discussion and Bibliography

This project presented a comprehensive comparison between MALA and RWMH. We started with a theoretical background on stochastic differential equations, diffusion processes, Fokker-Planck Equation, and Euler-Maruyama methods. Then, we introduced the algorithms and explored their implementations in various scenarios.

Three case studies were conducted, each focusing on a different type of target distribution: standard Gaussian, multi-modal, and high-dimensional Gaussian. We found that, MALA showed superior performance in dealing with high-dimensional distributions due to its gradient-based proposal mechanism. However, evaluating the gradient of the potential function can be more computationally expensive or not available in practical applications.

Future research, beyond the scope of this current project due to time constraints, could investigate the higher order discretization of the continuous Langevin dynamics, which are discussed in: [1] [2]; the extension a more efficient sampling method based on stochastic gradients, Stochastic Gradient Langevin Dynamics (SGLD), for Bayesian learning from large scale datasets. [8]

References

[1] A. G. A. Eberle and R. Zimmer. Couplings and quantitative contraction rates for langevin dynamics. *The Annals of Probability*, 47(4), 2018.

[2] A. S. Dalalyan and L. Riou-Durand. On sampling from a log-concave density using kinetic langevin diffusions. *Bernoulliy*, 26(3), 2020.

[3] C. Gardiner. *Stochastic methods : a handbook for the natural and social sciences*. Springer, 2009.

[4] G. O. Roberts and J. S. Rosenthal. Optimal scaling for various Metropolis-Hastings algorithms. *Statistical science*, 16(4):351–367, 2001.

[5] G. O. Roberts and R. L. Tweedie. Exponential convergence of langevin distributions and their discrete approximations. *Bernoulli*, 2(4):341–363, 1996.

[6] G. O. Roberts, A. Gelman, and W. R. Gilks. Weak convergence and optimal scaling of random walk Metropolis algorithms. *The annals of applied probability*, 7(1): 110–120, 1997.

[7] D. Sanz-Alonso. *Monte Carlo Simulation STAT 31511 Lecture Notes*. 2023.

[8] M. Welling and Y. W. Teh. Bayesian learning via stochastic gradient Langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 681–688, 2011.