Numerical SDE: Exit problem

Yihang Chen 1700010780

December 29, 2019

Contents

1 Settings

Consider the following boundary value problem:

$$\begin{aligned} b\nabla u + \frac{1}{2}\Delta u &= f(x,y),\ \ (x,y) \in B \\ u &= \frac{1}{2} \text{ on } (x,y) \in \partial B \end{aligned} \tag{1}$$

, where $b = (x, y)$, $f(x, y) = x^2 + y^2 + 1$, $B = B_1(0)$. The exact solution $u(x, y) = (x^2 + y^2)/2$. We will solve this PDE numerically via the simulation of SDEs.

2 Converting PDEs to SDEs

2.1 General results

According to [Wikipedia, 2019] and Theorem 9.3.3 in [Øksendal, 2000], we can generally convert a semi-elliptic boundary value PDE to a stochastic process.

Let D be a domain in $\mathbb{R}^n$ and let L be a semi-elliptic differential operator on $C^2(\mathbb{R}^n;\mathbb{R})$ of the form:

$$L = \sum_{i=1}^{n} b_i(x)\frac{\partial}{\partial x_i} + \sum_{i,j=1}^{n} a_{ij}(x)\frac{\partial^2}{\partial x_i\, \partial x_j} \tag{2}$$

where the coefficients b_i and a_{ij} are continuous functions and all the eigenvalues of the matrix $\alpha(x) = a_{ij}(x)$ are non-negative. Let $f \in C(D;\mathbb{R})$ and $g \in C(\partial D;\mathbb{R})$. Consider the Poisson problem:

$$\begin{cases} -Lu(x) = f(x), & x \in D \\ \lim\limits_{y\to x} u(y) = g(x), & x \in \partial D \end{cases} \quad \text{(P1)}$$

The idea of the stochastic method for solving this problem is as follows. First, one finds an Itō diffusion X whose infinitesimal generator (stochastic processes)—infinitesimal generator A coincides with L on compact support—compactly-supported C^2 functions $f : \mathbb{R}^n \to \mathbb{R}$. For example, X can be taken to be the solution to the stochastic differential equation:

$$\mathrm{d}X_t = b(X_t)\,\mathrm{d}t + \sigma(X_t)\,\mathrm{d}B_t \tag{3}$$

where B is n-dimensional Brownian motion, b has components b_i as above, and the matrix σ is chosen so that:

$$\frac{1}{2}\sigma(x)\sigma(x)^\top = a(x), \quad \forall x \in \mathbb{R}^n \tag{4}$$

For a point $x \in \mathbb{R}^n$, let $\mathbb{P}^x$ denote the law of X given initial datum $X_0 = x$, and let $\mathbb{E}^x$ denote expectation with respect to $\mathbb{P}^x$. Let τ_D denote the first exit time of X from D.

In this notation, the candidate solution for (P1) is:

$$u(x) = \mathbb{E}^x \left[g\big(X_{\tau_D}\big) \cdot \chi_{\{\tau_D < +\infty\}} \right] + \mathbb{E}^X \left[\int_0^{\tau_D} f(X_t)\, \mathrm{d}t \right] \tag{5}$$

provided that g is a bounded function and that:

$$\mathbb{E}^x \left[\int_0^{\tau_D} \big| f(X_t) \big|\, \mathrm{d}t \right] < +\infty \tag{6}$$

It turns out that one further condition is required:

$$\mathbb{P}^x \big(\tau_D < \infty \big) = 1, \quad \forall x \in D \tag{7}$$

For all x, the process X starting at x almost surely leaves D in finite time. Under this assumption, the candidate solution above reduces to:

$$u(x) = \mathbb{E}^x \left[g\big(X_{\tau_D}\big) \right] + \mathbb{E}^x \left[\int_0^{\tau_D} f(X_t)\, \mathrm{d}t \right] \tag{8}$$

and solves (P1) in the sense that if $\mathcal{A}$ denotes the characteristic operator for X (which agrees with A on C^2 functions), then:

$$\begin{cases} -\mathcal{A}u(x) = f(x), & x \in D \\ \lim\limits_{t \uparrow \tau_D} u(X_t) = g\big(X_{\tau_D}\big), & \mathbb{P}^x\text{-a.s.}, \forall x \in D \end{cases} \quad \text{(P2)} \tag{9}$$

2.2 Specific results

We may rewrite the original PDE as:

$$\begin{aligned} Lu &= -g \text{ in } B \\ u &= \phi \text{ on } \partial B \end{aligned}$$

where $b = (x, y)^T$, $(a_{ij})_{2\times 2} = \frac{1}{2} I_2$, $g(x, y) = -(x^2 + y^2 + 1)$, $\phi = \frac{1}{2}$.

Consider an Ito diffusion$\{(X_t, Y_t)\}$ whose generator $\mathcal{A}$ coincides with L on $C_0^2(\mathbb{R}^2)$

$$d \begin{pmatrix} X_t \\ Y_t \end{pmatrix} = \begin{pmatrix} X_t \\ Y_t \end{pmatrix} dt + dW_t.$$

Denote τ as the exit time. Since we have the fact that $E^{(x,y)}(\tau) < \infty$ (as shown in the lecture notes) and $g(x, y) = -(x^2 + y^2 + 1)$ is bounded in B, it's then trivial to verify that $\mathbb{E} \int_0^{\tau_D} (X_s^2 + Y_s^2 + 1) ds \leq \sup q \ \mathbb{E}\tau_D < \infty$.

Hence, we can write the solution of the PDE 1 in

$$u(x,y) = \frac{1}{2} - \mathbb{E}^{x,y} \int_0^\tau (X_s^2 + Y_s^2 + 1) ds \tag{10}$$

2.3 Simulation Algorithms

We utilize the standard Euler-Maruyama scheme[1] to calculate the scheme to do the simulation of the value $u(x_0, y_0)$.

2.3.1 Euler-Maruyama scheme

Algorithm 1 EULER-MARUYAMA SCHEME

Input: (x_0, y_0), Δt: step size, N: number of workers
Output: $\hat{u}(x_0, y_0)$

```
sum ← 0  i ← 0  while i < N do
    X_0 ← x_0  Y_0 ← y_0  while True do
        X_{t+1} ← X_t + X_t Δt + N(0, Δt)  Y_{t+1} ← Y_t + Y_t Δt + N(0, Δt)  if X_{t+1}^2 + Y_{t+1}^2 > 1 then
            break
        end
    end
    Sum ← Sum + ∑_t (X_t^2 + Y_t^2 + 1)Δt
    i ← i + 1
end
return Sum/N
```

This is implemented in "euler.py".

2.3.2 Multilevel Monte-Carlo method

I try to apply multilevel Monte-Carlo method in this problem, but I fail. Since in the multilevel estimation, we need to use one Brownian motion to estimate two consecutive layers at one time. However, the exit time of the deeper layer is not known, which may not be a multiple of M. The error introduced here is not negligible and can not be easily overcome. However, since I have written the multilevel code according to the work [Giles, 2008]. I use a toy example here.

For the SDE

$$\mathrm{d}X_t = -\frac{1}{2} X_t \mathrm{d}t + \mathrm{d}W_t \tag{11}$$

Compute $u = \mathbb{E}X_1^2$. [2]

[1] For this specific problem, some variation, such as the Milstein methods and the basic Runge-Kutta method are equivalent to the standard Euler-Maruyama scheme.

[2] The exact solution is $1 - e^{-1}$.

I implement the MLMC in multilevel.py. The general function is function "mlmc". For specific requirement, we only need to modify "mlmc_euler", which serves as estimating the difference of two consecutive layers.
For example, we execute

```
mlmc(3, 0.005, 0.01, Richardson = True)
```

we get the error is 0.002(smaller than 0.005), and the variance is 0.8.

3 Numerical Results

3.1 Euler-Maruyama method

We evenly generate 20 points from the unit disk and use mean to approximate the expectation, and in the Euler–Maruyama code, we set the number of iteration is 3000. Furthermore, we set $\sqrt{\Delta t} = 0.005i,\ 1 \leq i \leq 30$.
Activate function "main_a". The result of the method follows in Fig(1). The score is 0.9971228350238678, indicating our estimation's accuracy, and the expected α is 0.47553515, we expect the ground-truth value is 0.5. To strengthen the results, we plot the relation Error-$\sqrt{\Delta t}$ in Figure 2, and disconvered that it is linear (regression score is 0.9928555690474009).

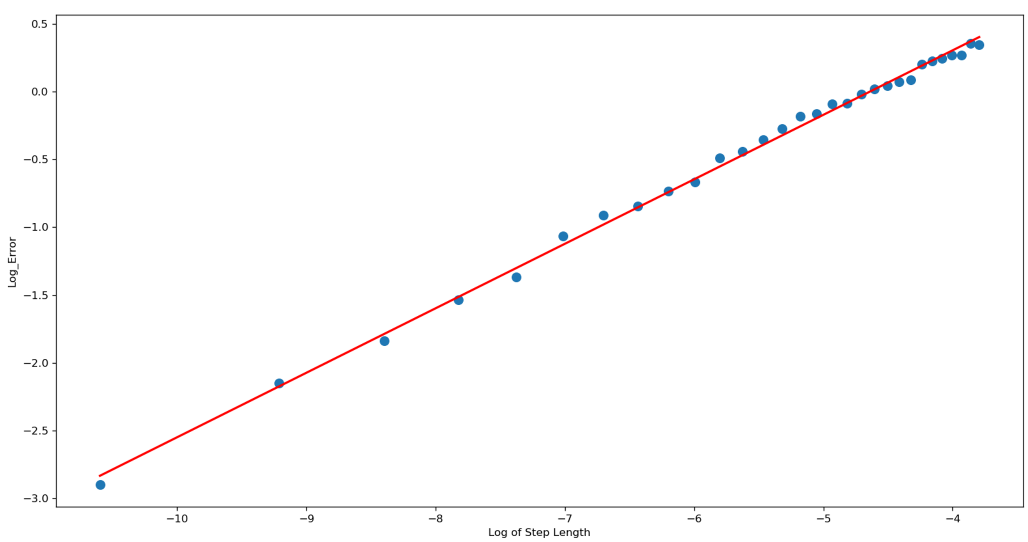

Figure 1: logError - $\log \Delta t$

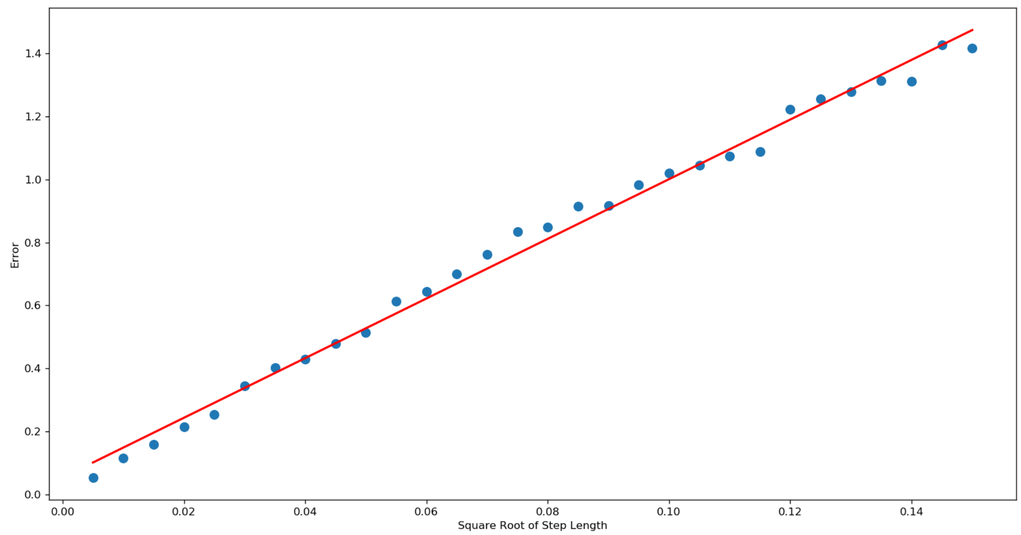

Figure 2: Error - $\sqrt{\Delta t}$

References

[Giles, 2008] Giles, M. B. (2008). Multilevel monte carlo path simulation. *Operations Research*, 56:607–617.

[Øksendal, 2000] Øksendal, B. (2000). Stochastic differential equations: An introduction with applications. Springer.

[Wikipedia, 2019] Wikipedia (2019). Stochastic processes and boundary value problems — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=Stochastic%20processes%20and%20boundary%20value%20problems&oldid=909492923. [Online; accessed 11-December-2019].