Numerical Methods for the Fokker-Planck Equation

Project Report
APMA E4301: Numerical Methods for PDEs

Yan Cheng (UNI: yc3855)

April 22, 2021

Abstract

In this project, we will perform some numerical studies on the one-dimensional Fokker-Planck equation. It is known that the solutions to the Fokker-Palnck equation can be approximated by Hermite polynomials, and this idea has been utilized to achieve agreeable results computationally. The main contents in this article is thus to explain and test this approach, and then compare it with the more familiar finte difference method.

1 Introduction

For my final project for the class, I want to apply what we have learn about the numerical solutions for PDE to the Fokker-Planck equation, which is an important equation in statistical physics. There is a vast amount of literature on this subject. The most comprehensive reference I found online is *The Fokker-Planck Equation* by Risken [2], but it is a little old and the notation is quite different from many other materials I found. A newer material is [3] by Pavliotis, in which it is suggested that there is some interesting development on the Hermite spectral methods (including but not limited to [4]). This new scheme is based on the finite difference method and the Hermite spectral method, both of which we will discuss in this article.

In its original setting when applied to Brownian motion, the Fokker-Planck equation describes the fluctuation of small but macroscopic partices in a fluid. The particles move around in an unpredictable way due to their collisions with the molecules of the fluid. It is only logical to assume the position of a particle in a region can only be determined up to a probability – the Fokker-Planck euqation is a model that describes the time evolution of the underlying probability density function.

In this project, we will focus on the one-dimensional Fokker-Planck equation

$$\frac{\partial}{\partial t}p(x,t) = -\frac{\partial}{\partial x}\left(\mu(x,t)p(x,t)\right) + \frac{\partial^2}{\partial x^2}\left(D(x,t)p(x,t)\right), \quad 0 \le x < L$$

$$p(x,0) = p_0(x), \quad p(0,t) = p(L,t) = 0.$$

It is a well-knwon fact that analytic solutions for Cauchy problem can be found, but the intitial-boundary value problems in general cannot be solved analytically. Hence, seeking effective numerical schemes has practical significance here too.

In the past, several methods had their success in approximating the solution to a Fokker-Planck equation numerically, including the finite difference method, the Galerkin method, the spectral method, etc. What we are interested in in this project is the performance of a certain type of spectral method known as the Hermite spectral method.

2 Finite Difference Method

In this section, we will use the finite difference method to solve a simple Fokker-Planck equation

$$\frac{\partial p}{\partial t} = \frac{\partial (xp)}{\partial x} + \frac{\partial^2 p}{\partial x^2}$$

with the initial condition (this example is taken from [6])

$$p(x,0) = e^{-x^2/2}\left(1 + \cos\left(\frac{\pi x}{2}\right) e^{\pi^2/8}\right).$$

The true solution to the unbounded problem is

$$p(x,t) = e^{-x^2/2}\left(1 + \cos\left(\frac{\pi x}{2} e^{-t}\right) e^{\pi^2/8} e^{-2t}\right).$$

When implementing the finite difference method, I noticed that the true solution vanish outside of $(-3,3)$ for all $t \geq 0$; this is due to the exponential decay property. Therefore, I took the domain to be some larger $[-L, L]$ and assumed we are working with the Dirichlet boundary condition. It goes without saying that as a result, the following numerical experiments are not so conclusive; however, this assumption does simplify the coding part a lot!

Using the product rule, we have

$$\frac{\partial p}{\partial t} = p + x\frac{\partial p}{\partial x} + \frac{\partial^2 p}{\partial x^2}.$$

Using the forward difference for the first derivative and central difference for the second derivative, we get

$$\frac{\partial p_j}{\partial t} = p_j + x_j \frac{p_{j+1} - p_j}{\Delta x} + \frac{p_{j+1} - 2p_j + p_{j-1}}{\Delta x^2}.$$

This can be written compactly as a system of ODEs

$$\frac{d\vec{p}}{dt} = A\vec{p}$$

where

$$A = I_n + \frac{1}{\Delta x}\mathrm{diag}(x_0, x_1, \ldots, x_n)\begin{bmatrix} -1 & 1 & & & & \\ & -1 & 1 & & & \\ & & -1 & 1 & & \\ & & \ddots & \ddots & & \\ & & & & -1 & 1 \\ & & & & & -1 \end{bmatrix} + \frac{1}{\Delta x^2}\begin{bmatrix} 2 & 1 & & & & \\ 1 & 2 & 1 & & & \\ & 1 & 2 & 1 & & \\ & & \ddots & \ddots & & \\ & & & 1 & 2 & 1 \\ & & & & 2 & 1 \end{bmatrix}$$

To solve the resulting ODE, I used the forward Euler and backward Euler. The results are as follows:

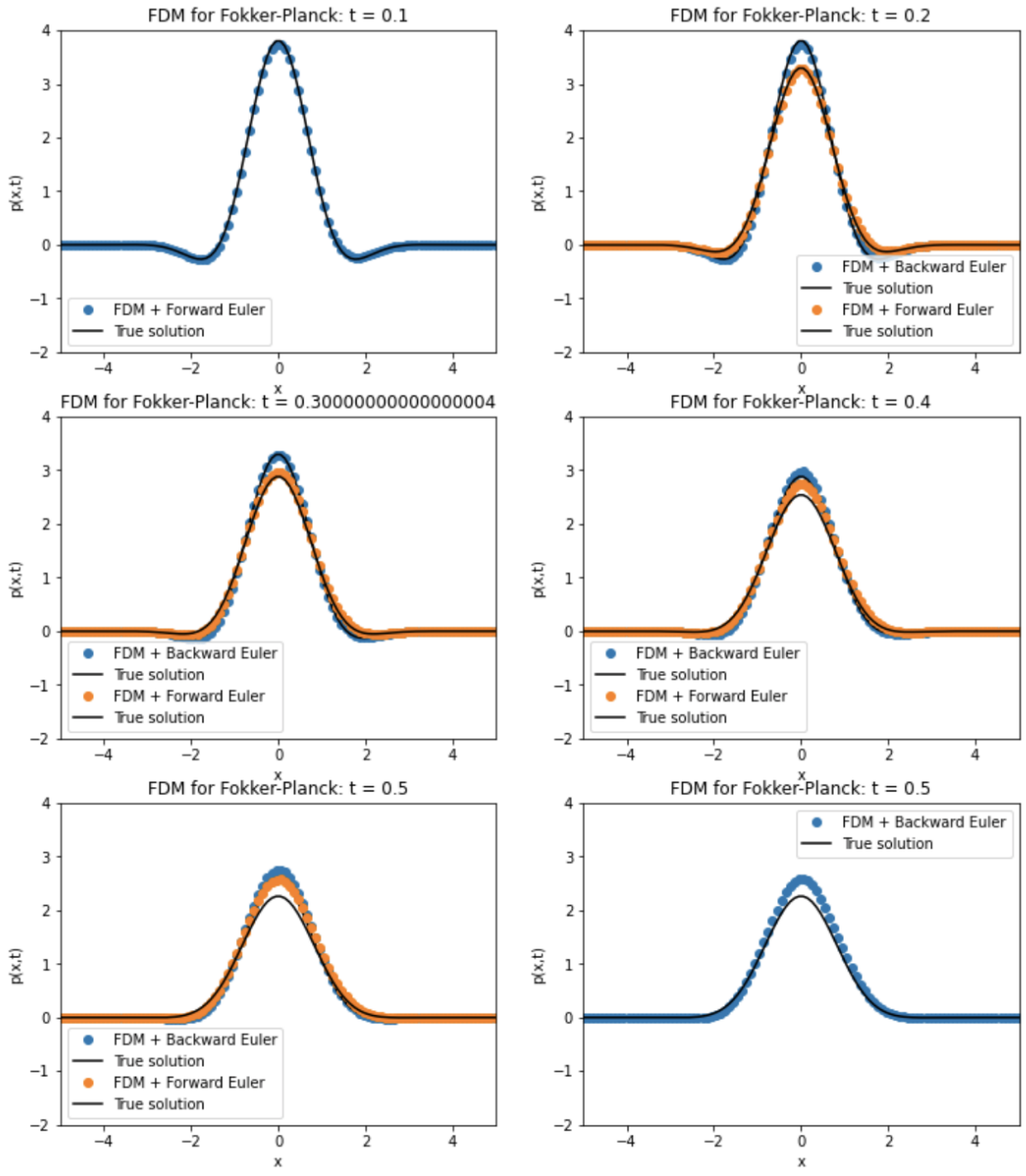

Figure 1: Using the finite difference method to solve the Fokker-Planck equation.

Strangely, the numerical results are not working so well. The reasons may be one or more of the following:

1. The Euler methods used to solve the ODE is not accurate enough

2. From the plots, we can tell that the source of error locates near the origin. The numerical scheme may not be stable in this region

3. The finite difference discretization in space somehow yields instability.

This evidence has encouraged us to look for other numerical schemes to better handle the Fokker-Planck equation. However, despite the failure, the finite difference method has an advantage that it is easy to be implemented. As we will see, the Hermite spectral method is much harder to implement.

3 Hermite Spectral Method

3.1 Theory

In the following, we will introduce a popular method for solving the Fokker-Planck equations using the Hermite polynomials. The solution that we hope to solve is a simple Fokker-Planck equation

$$\frac{\partial p}{\partial t} = \frac{\partial (xp)}{\partial x} + \frac{\partial^2 p}{\partial x^2} \tag{1}$$

with appropriate initial and boundary conditions.

It can be shown that the true solutions of the Fokker-Planck equation behaves like Gaussians at infinity. Therefore, it would be reasonable to expect that approximating the solutions by Hermite polynomials can yield good outcome. The n-th order Hermite polynomial H_n is defined[1] as

$$H_n(x) = (-1)^n e^{x^2} \frac{d^n}{dx^n} e^{-x^2}.$$

The Hermite polynomials are orthogonal to each other

$$\frac{1}{\sqrt{\pi} 2^n n!} \int_{-\infty}^{\infty} H_n(x) H_m(x) e^{-x^2} dx = \delta_{mn}, \quad m, n \geq 0.$$

It can be shown that the following recurrence relation holds:

$$H_{n+1} = 2xH_n(x) - 2nH_{n-1}(x), \quad n \geq 1.$$

Let

$$\varphi_n(x) = \left(2^n n! \sqrt{\pi}\right)^{-1/2} e^{-x^2/2} H_n.$$

Then $\{\varphi\}_{n=0}^{\infty}$ form an orthonormal basis in $L^2(\mathbb{R})$. Assume that the solution to (1) can be expressed as

$$p(x,t) = \sum_{n=0}^{\infty} c_n(t) \varphi_n(x).$$

These functions satisfy the recurrence relations

$$\varphi' = x\varphi_n = \sqrt{\frac{n}{2}} \varphi_{n-1} - \sqrt{\frac{n+1}{2}} \varphi_{n+1},$$

[1] We will be using what is called the "physicist's Hermite polynomials," which is what `numpy` uses.. It is a little different from the "probabilist's Hermite polynomials". Using diferent Hermite polymomials may lead to an error by some factor. This inconsistency is in fact a tricky issue when implementing the numerical method.

$$x\varphi_n' = \frac{\sqrt{n(n-1)}}{2}\varphi_{n-2} - \frac{1}{2}\varphi_n - \frac{\sqrt{(n+1)(n+2)}}{2}\varphi_{n+2},$$

and

$$\varphi_n'' = \frac{\sqrt{n(n-1)}}{2}\varphi_{n-2} - \left(n+\frac{1}{2}\right)\varphi_n - \frac{\sqrt{(n+1)(n+2)}}{2}\varphi_{n+2}.$$

Using these relations, we have

$$\begin{aligned} xp_x &= \sum_{n=0}^{\infty} \frac{\sqrt{n(n-1)}}{2} c_n(t)\varphi_{n-2}(x) - \sum_{n=0}^{\infty} c_n(t)\varphi_n(x) - \frac{\sqrt{(n+1)(n+2)}}{2} c_n(t)\varphi_{n+2}(x) \\ &= \sum_{n=0}^{\infty} \left[\frac{\sqrt{(n+1)(n+2)}}{2} c_{n+2} - \frac{1}{2}c_n - \frac{\sqrt{n(n-1)}}{2} c_n \right] \varphi_n \end{aligned}$$

if we set $c_{-1} = c_{-2} = 0$. By a smilar computation, we have

$$p_{xx} = \sum_{n=0}^{\infty} \left[\frac{\sqrt{(n+1)(n-+2)}}{2} c_{n+2} - \frac{2n+1}{2}c_n - \frac{\sqrt{n(n-2)}}{2} c_n \right] \varphi_n$$

Substituting them into (1) and rearranging, we get a family of ODEs

$$c_n' = -nc_n + \sqrt{(n+1)(n-+2)}c_{n+2}, \quad n = 0, 1, 2, \ldots$$

Let $d_j = c_{2k}$ and $e_k = c_{2k+1}$, $k = 0, 1, 2, \ldots$, then the above can be rewritten as

$$d_k' = -2kd_k + \sqrt{(2k+1)(2k+2)}d_{k+1}, \tag{2}$$

and

$$e_k' = -(2k+1)e_k + \sqrt{(2k+2)(2k+3)}e_{k+1}. \tag{3}$$

Suppose we approximate p by a truncated sum

$$p(x,t) \approx \sum_{n=0}^{2k+1} c_n(t)\varphi_n(x).$$

Let

$$D = [d_0, d_1, \ldots, d_k]^T, \quad E = [e_0, e_1, \ldots, e_k]^T.$$

Then (2) and (3) can be written compactly as

$$\frac{dD}{dt} = A_0 D, \quad \frac{dE}{dt} = A_1 E$$

where

$$A_0 = \begin{pmatrix} 0 & \sqrt{2} & 0 & 0 & \cdots \\ 0 & -2 & \sqrt{12} & 0 & \cdots \\ 0 & 0 & -4 & \sqrt{30} & \cdots \\ 0 & 0 & 0 & -6 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix}$$

and

$$A_1 = \begin{pmatrix} -1 & \sqrt{6} & 0 & 0 & \cdots \\ 0 & -3 & \sqrt{20} & 0 & \cdots \\ 0 & 0 & -5 & \sqrt{42} & \cdots \\ 0 & 0 & 0 & -7 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix}.$$

Since the eigenvalues of A_1 are all negative, we necessarily have

$$\lim_{t\to\infty} E(t) = 0.$$

It can be shown that $D(t)$ satisfies a similar decay result.

3.2 Discussion

To summarize our results from the previous subsection, the solution for (1) is approximated by

$$p_{2k}(x,t) = \sum_{n=0}^{2k+1} c_n(t)\varphi_n(x)$$

where

$$\varphi_n(x) = \frac{1}{\sqrt{2^n n!\sqrt{\pi}}} e^{-x^2/2} H_n, \quad n \geq 1.$$

Let $d_k = c_{2k}$, $e_k = c_{2k+1}$, $k = 0, 1, 2, \ldots$, then we can find $D = [d_0, d_1, \ldots, d_k]^T$, $\quad E = [e_0, e_1, \ldots, e_k]^T$ by solving two systems of ODEs

$$\frac{dD}{dt} = A_0 D, \quad \frac{dE}{dt} = A_1 E.$$

We have calculated the coefficient matrices A_0, A_1 for (1), but for the more general equation

$$\frac{\partial p}{\partial t} = \mu \frac{\partial (xp)}{\partial x} + \frac{\partial^2 p}{\partial x^2}$$

the recurrence relations lead to a couple of tridiagonal systems:

$$A_0 = \begin{pmatrix} \alpha_0 & \gamma_0 & & & \\ \beta_0 & \alpha_1 & \gamma_1 & & \\ 0 & \beta_1 & \alpha_2 & \gamma_2 & \\ & \ddots & \ddots & \ddots & \\ & & & \beta_{k-1} & \alpha_k \end{pmatrix}, \quad A_1 = \begin{pmatrix} a_0 & c_0 & & & \\ b_0 & a_1 & c_1 & & \\ 0 & b_1 & a_2 & c_2 & \\ & \ddots & \ddots & \ddots & \\ & & & b_{k-1} & a_k \end{pmatrix}$$

where

$$\alpha_i = \frac{\mu - 1}{2} - 2i$$

$$\beta_i = (1-\mu)\frac{\sqrt{(2k+1)(2k+2)}}{2}$$

$$\gamma_i = (1+\mu)\frac{\sqrt{(2k+1)(2k+2)}}{2}$$

and

$$a_i = \frac{\mu - 1}{2} - 2i$$

$$b_i = (1 - \mu)\frac{\sqrt{(2k+2)(2k+3)}}{2}$$

$$c_i = (1 + \mu)\frac{\sqrt{(2k+2)(2k+3)}}{2}.$$

I originally planned to solve

$$\frac{\partial p}{\partial t} = \frac{\partial (xp)}{\partial x} + \frac{\partial^2 p}{\partial x^2}$$

with the same initial condition as the one I used for the finite difference method, but unfortunately, I could not somehow get the code to work. The implementation is tricky, because the the initial conditions for the two resulting ODE systems is determined by $p(x, 0)$. Therefore, I tried to first use the Hermite interpolation to get the coefficients of $p(x, 0)$, and then artificially adjust the coefficient so that the basis functions are the *Hermite functions* φ_n.

Although my implementation was not successful, the numerical results for the same problem are given n [6]:

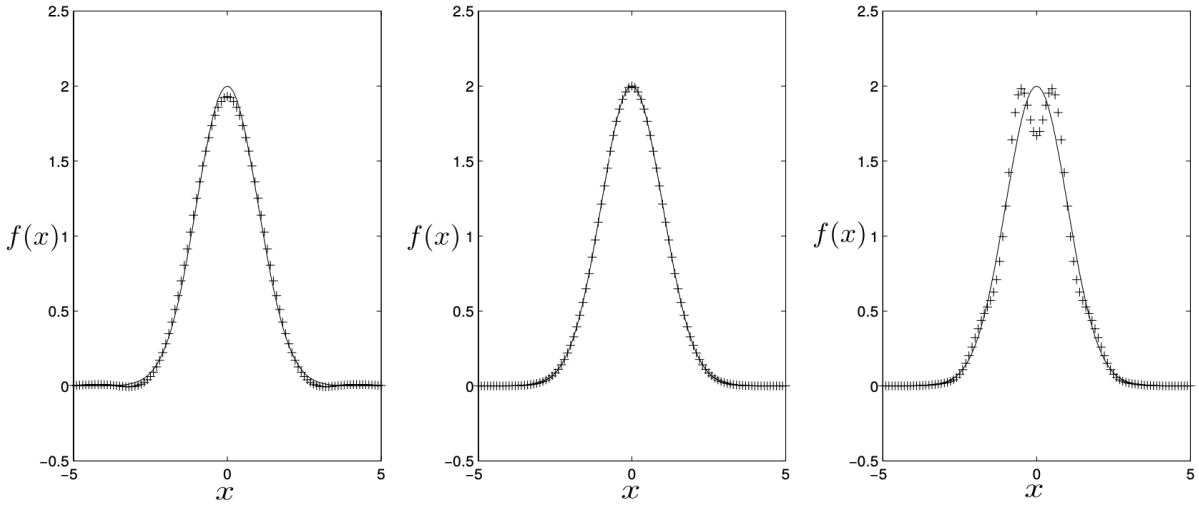

Figure 2: This is the numerical solution for $t = 10.0$.

It is also suggested by [6] that even if we are not able to approximate the initial condition very accruately, it is possible to get good numerical result for small time using the Hermite spectral method (the author did $t = 1.0$ and compared the result for using Hermite polynomials of maximum degrees 5 and 10).

4 Conclusion

In this project, I tried to solve the Fokker-Planck equation using the finite difference method, and I learned the Hermite spectral method that reduces the PDE problem to systems of ODEs. The latter

yields better performance, but is much harder to implement. Also, if we change the coefficients in the Fokker-Planck equations to be some complicated function of the space variable, the resulting ODE systems can be very different. Furthermore, the Hermite spectral method is perhaps more theoretically challenging. For instance, in [4], the authors did 20 pages of error analysis for their scheme, involving a series of complicated analytic estimates. This is also something I would like to explore more about in the future.

References

[1] Weinan E, Tiejun Li, and Eric Vanden-Eijnden. *Applied Stochastic Analysis.* Graduate Studies in Mathematics. Volume: 199; 2019.

[2] Hannes Risken, Till Frank. *The Fokker-Planck Equation: Methods of Solution and Applications.* Springer. 1996.

[3] Grigorios A. Pavliotis *Stochastic Processes and Applications: Diffusion Processes, the Fokker-Planck and Langevin Equations.* Springer. 2014.

[4] Johnson Fok, Benyu Guo, and Tao Tang. *Combined Hermite Spectral-Finite Difference Method for the Fokker-Planck Equation.* Mathematics of Computation. Volume 71, Number 240, Pages 1497–1528. Article electronically published on December 21, 2001

[5] Tianhong CHen *A Theoretical and Numerical Study for the Fokker-Planck Equation.* Master Thesis, Simon Fraser University, 1992.

[6] Masoumeh Mohammadi. *Analysis of discretization schemes for Fokker-Planck equations and related optimality systems.* PhD Thesis, 2014.