Probabilistic schemes for semi-linear nonlocal diffusion equations with application in predicting runaway electron dynamics

by

Minglei Yang

A dissertation submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Auburn, Alabama
December 14, 2019

Keywords: Partial integro-differential equation, Feynman-Kac theory, Sparse grids, Runaway electrons, Fokker-Planck equation, Delaunay triangulation.

Approved by

Yanzhao Cao, Chair, Professor of Mathematics and Statistics
Guannan Zhang, Co-Chair, Research Scientist of Mathematics Division Oak Ridge National Laboratory
Ming Liao, Professor of Mathematics and Statistics
Junshan Lin, Associate Professor of Mathematics and Statistics
Hans-Werner van Wyk, Assistant Professor of Mathematics and Statistics
Anh Nguyen, Assistant Professor of Computer Science and Software Engineering

Abstract

In this work, we focus on developing and analyzing novel probabilistic numerical approaches for solving several types of semi-linear nonlocal diffusion equations in both unbounded and bounded high dimensional spaces. First, we propose probabilistic schemes for solving the partial integro-differential equation (PIDE) with Fokker-Planck operator related to a jump-diffusion process in the unbounded domain $\mathbb{R}^d$. Under a given probability space, we exploit the probabilistic representation of the solution of PIDE to construct both temporal discrete and temporal-spatial discrete schemes of the solution of PIDE. The rigorous error analysis is provided to prove that the temporal discrete scheme can achieve first-order convergence. To add in spatial discretization, the temporal-spatial discrete scheme incorporates with the high-order piecewise polynomial interpolation that leads to high order convergence with respect to spatial mesh size Δx. Next, we consider another typical nonlocal diffusion equation, the fractional Laplacian equations. Due to work studied by Serge Cohen and Jan Rosiéski (2007), and S. Asmussen and Jan Rosiéski (2001), stable processes can be simulated by Lévy processes that consist of the compound Poisson processes and appropriate Brownian motions. Hence the fractional Laplacian operator can be approximated by the second partial integro-differential operator. Our probabilistic schemes for the PIDE model introduced in Chapter 3 can give a novel numerical approach for solving the fractional Laplacian equation. Third, we impose volume constraints into PIDEs model and consider the initial-boundary value partial integro-differential equations. The key idea is to exploit the regularity of the solution $u(t, x)$ to avoid direct approximation of the random exit time corresponding to the boundary conditions. The error from the exit time decays sub-exponentially with respect to temporal mesh Δt when all interior grid points are sufficiently far from the boundary. Moreover, our numerical methods lead to an overall first-order convergence rate with respect to Δt and achieve high order convergence with respect to Δx. Last, we introduce one application of the initial-boundary value PIDE problem,

the approximation of the runaway probability of electrons in fusion tokamak simulation. Runaway electrons (REs) generated during magnetic disruptions present a major threat to the safe operation of fusion tokamak. A critical aspect of understanding REs dynamics is to calculate runaway probabilities, i.e., the probability of an electron in the phase space will runaway on, or before, a time $t > 0$. Mathematically, such probability can be obtained by solving the adjoint equation of the underlying Fokker-Planck equation that controls the electron dynamics. In this work, we present a sparse-grid probabilistic scheme for computing runaway probability. The key ingredient of our approach is to represent the solution of the adjoint equation as a conditional expectation, such that discretizing the differential operator becomes approximating a set of integrals. The sparse grid interpolation is utilized to approximate the runaway probability, and adaptive refinement is also exploited to handle the sharp transition layer between the runaway and non-runaway region.

Acknowledgments

There are pivotal contributions made by a lot of people to this dissertation. I would first and foremost like to thank my Ph. D. advisor, Dr. Yanzhao Cao for giving me the opportunity to study at Auburn University. He gave me endless support and pushed me ever forward towards success.

I am also grateful to my Ph. D. co-advisor, Dr. Guannan Zhang, who leads me to extend my research area and offered me an excellent opportunity to join the Oak Ridge National Lab as an intern. His guidance helped me all the time in research.

I want to thank my dissertation committee members and university reader: Dr. Ming Liao, Dr. Junshan Lin, Dr. Hans-Werner van Wyk, Dr. Anh Nguyen, and Dr. Xing Fang, for their time, excellent supports, and valuable advices. I enjoyed and learned a lot from Dr. Liao and Dr. Lin's graduate courses.

Amid the many peers that assisted me, Huan Xu, you are always there when I had questions or needed a listening ear.

Last but not least, I would like to express my deepest gratitude to my families and friends. No one achieves anything alone.

Table of Contents

Abstract ii

Acknowledgments iv

1 Introduction 1

2 Problem definitions and mathematical preliminaries 7

2.1 Probability space, random variables and stochastic processes 7

2.2 FBSDEs and PIDEs 8

2.3 Sparse grid quadrature 11

2.4 High interpolation on Delaunay triangulation 13

3 Probabilistic schemes for the partial integro-differential equations in unbounded domains 16

3.1 Problem setting 16

3.2 The probabilistic numerical schemes 17

3.2.1 The probabilistic representation 18

3.2.2 The temporal discrete scheme 20

3.2.3 The temporal-spatial discrete scheme 21

3.3 Error estimates 25

3.3.1 Error estimate of the temporal discrete scheme 26

3.3.2 Error estimate of the temporal-spatial discrete scheme 31

3.4 Numerical examples 39

4 Probabilistic schemes for the fractional Laplacian equation 44
4.1 Problem setting 44
4.2 Gaussian approximation of Lévy processes 45
4.3 Numerical examples 49

5 The PIDEs with Volume Constraints 52
5.1 Problem setting 52
5.1.1 The non-divergence form of the PIDE 52
5.2 The new probabilistic numerical scheme for the PIDEs 53
5.2.1 Mesh generation 53
5.2.2 The probabilistic representation of the PIDE solution 54
5.2.3 Approximation of the backward stochastic process 55
5.2.4 Temporal discretization of the probabilistic representation 57
5.2.5 Approximation of the conditional expectation 57
5.2.6 The fully discrete scheme 58
5.3 Numerical examples 60

6 Runaway electrons model 62
6.1 Problem setting 62
6.2 A sparse-grid probabilistic method for the adjoint equation 65
6.2.1 Temporal discretization 66
6.2.2 Sparse-grid interpolation for spatial discretization 67
6.2.3 Quadrature for the conditional expectation 71
6.3 Numerical examples 73
6.3.1 Example 1: Escape probability of a Brownian motion 73
6.3.2 The runaway probability of the three-dimensional RE model 75
6.4 Concluding remarks 78

7 Summary and Future Work 80

7.1 Summary 80

7.2 Future work 81

References 82

Appendix 88

7.3 Construction of backward filtration 88

7.4 Ito formula for backward stochastic differential equation 89

7.5 River models in numerical examples 90

List of Figures

2.1 Quadratic and Cubic elements in Tetrahedron 14

3.1 Brownian motions simulation 19

3.2 Spatial domains 40

3.3 Errors and convergence rates with respect to Δx in Example 1 42

3.4 The evolution of hydraulic head $u(t, x)$ in Example 2 43

4.1 Gaussian approximation of Lévy process 47

4.2 Errors and convergence rates with respect to Δx in Example 3 51

5.1 Errors and convergence rates with respect to Δx in Example 4 61

6.1 Left: linear hierarchical basis; Middle: quartic hierarchical basis where the quadratic polynomials appear since level 2; Right: cubic hierarchical basis where the cubic polynomials appear since level 3. 68

6.2 The relative error of the approximate escape probability of the standard Brownian motion. 75

6.3 The error distribution in the spatial domain $[0, 5] \times [0, 5]$ for the three cases considered in Fig 6.2 for $t = 0.5, 1.0$ and 2.0. 76

6.4 Cross sections of the runaway probability P_{RE} as well as the corresponding adaptive sparse grids at three time instances $t = 24, 60$ and 120. 77

6.5 Comparison between the our approach and the direct MC for pitch angle $\theta = 15^{\circ}$ and minor radius $r = 0.5$. 78

7.1 River Model 1 90

7.2 River Model 2 91

7.3 River Model 3 92

List of Tables

2.1 Gauss Rules Summary 13

3.1 Errors and convergence rates with respect to Δt in Example 1 41

4.1 Errors and convergence rates with respect to Δt in Example 3 50

5.1 Errors and convergence rates with respect to Δt in Example 4, where T=1, $\Delta x = \frac{1}{32}$ 61

Chapter 1

Introduction

Mathematical modeling plays an essential role in predicting behavior problems in business, physical and biological science. The physical and mathematical models developed are often highly simplified versions of reality. Nonetheless, we gained great insights in understanding the underlying physical mechanisms through these models.

In many physical applications, we aim to describe how one quantity changes in relation to another quantity. For instance, the distribution of some quantities evolves over time in a solid medium. Such a relationship can be represented mathematically by derivatives. It should be reasonable to express such principles in terms of differential equations. Differential equations can be divided into several types based on their own properties. The diffusion equation is a type of partial differential equations (PDEs) relevant to the Markov process. Diffusion equations can be formally classified into two types: local diffusion equations and nonlocal diffusion equations. From the stochastic perspective, the diffusion is deemed nonlocal whenever the associated underlying stochastic process is given by a non-Brownian motion process, i.e., a process without independent increments. The feature of nonlocal diffusion can be applied in a wide variety of fields of physics, finance, and insurance, such as contaminant flow in groundwater, the dynamics of financial markets and risk measures [20, 26, 29]. The existence and uniqueness of solutions of diffusion equations have been proved in [5, 41], but obtaining the analytical solutions of such problems is typically difficult. Hence numerical solutions are highly desired in applications. Various methods exist to build numerical solutions of diffusion equations, e.g., meshless methods [3], finite-element-type methods [11] and continuous-time

random walk (CTRW) methods [10, 29]. For the first two methods (can be classified as deterministic methods), the nonlocal operator may lead to a severe computational cost due to nonsparsity of the underlying linear or nonlinear systems and the CTRW methods as the typical stochastic methods suffer the slow convergence and requiring a large number of data samples to guarantee small errors. The key idea of our numerical approach is to exploit the probabilistic representation of the solution of the differential equation, which can achieve stable and high order numerical schemes.

A recently developed probabilistic numerical approach is exploited to provide a variational analysis for a general class of nonlinear parabolic partial differential equations with Dirichlet boundary conditions in $\mathbb{R}^d$. The probabilistic approach connects the nonlinear parabolic partial differential equations with Dirichlet boundary conditions with the decoupled forward-backward stochastic differential equations (FBSDEs) with random terminal time. Peng and Pardoux [35] have proved the existence and uniqueness of backward stochastic differential equations (BSDEs) with a fixed terminal time under some standard conditions. Moreover, the extension of BSDEs with a random terminal time has been studied in [9, 34]. We refer to some literature [6, 30–32] that have been devoted to numerical methods for the solutions of BSDEs. One typical technique to deal with the random exit time is approximating exit time [6], but it suffers a low convergence rate. Another numerical approach is to use probabilistic representations of their solution. An approximation solution of the Dirichlet problem has the form of expectation of a functional of the chain trajectory, which involved by the Monte Carlo (MC) technique and linear interpolation skills.

Due to errors in the measurements and inherent randomness of real dynamics, coefficients and the forcing term of differential equations can be defined by a family of random functions with a set of parameters. In reality, some initial and boundary conditions are often unknown, and we are only given specific ranges of some values. Hence they are also suitable to be treated as random variables or random processes. White noise is a typical stochastic process with an independent and identical distribution, which is the generalized derivative of the Wiener process or Brownian motion. Incorporating random processes into traditional ordinary differential equations (ODEs) or partial differential equations, we get stochastic differential equations

(SDEs) or stochastic partial differential equations (SPDEs). The conventional approach for approximating the action of random elements in the solution of ODEs or PDEs is the Monte Carlo method [14]. Monte Carlo methods are a broad class of computational algorithms that rely on repeated random sampling to obtain numerical results. The fundamental concept is to use randomness to solve problems that might be deterministic in principle. However, the MC method is not efficient in solving large-scale mathematical models.

Lévy processes form a significant class of stochastic processes that mainly contain both Brownian motion and the Poisson process (finite /infinite jump amplitude). The stable process that has paths of infinite variation is widely used in modeling complex physical and economics phenomena. The simulation of a stable process has been investigated in a vast literature [4, 8, 22, 24]. Asmussen and Rosiński [4] showed that a Brownian motion could simulate the remainder with a small variance in most cases. In the multidimensional space, the first technical issue is how to choose a compound Poisson process such that it is easy to simulate and the choice also determines the form of the remainder process, which we want to approximate by a Brownian motion. Serge Cohen and Jan Rosiński [8] conducted the theoretical analysis and obtained the necessary and sufficient conditions for Gaussian approximations in multidimensional spaces. The symmetric α stable (SαS) process as a particular case of stable processes with infinity variation is the corresponding Lévy process of the fractional Laplacian operator. In general, SαS process with ($1 \leq \alpha \leq 2$) is different from the Gaussian process because it does not have joint probability density functions and it is usually expressed by the characteristic function.

Next, we look at one application of PIDEs with volume constraints in plasma physics. In magnetically confined fusion plasma, runaway electrons (REs) can be generated during magnetic disruptions due to the strong electric field resulting from the rapid cooling of the plasma. At high enough velocities, the drag force on a particle due to Coulomb collisions in plasma decreases as the particle velocity increases. As a result, in the presence of a strong enough parallel electric field, fast electrons can "runaway" and be continuously accelerated. Understanding this phenomenon has been an area of significant interest because of the potential impact that REs can have to the safe operation of ITER. In particular, if not avoided or mitigated, REs can

severely damage plasma facing components Ref. [13, 21, 28]. We propose a sparse-grid probabilistic scheme to study RE dynamics in phase space.

The outline of this work is as follows: In Chapter 2, we introduce some important mathematical concepts and preliminaries related to probability space, stochastic differential equations, some fundamental knowledge about sparse grids quadrature and piecewise interpolation polynomials in high dimensional irregular domains.

In Chapter 3, we propose novel numerical schemes for approximating the solution of PIDE in unbounded domains based on the works [50, 54]. We derive the explicit probabilistic representation of the solution of the initial PIDE associated with a *backward* jump-diffusion process based on the nonlinear Feynman-Kac theory. The Feynman-Kac theory establishes the relationship between the PIDEs and a certain class of stochastic differential equations with jumps. We propose both temporal discrete scheme and temporal-spatial discrete scheme taking advantage of the probabilistic representation. High-order convergence in spatial discretization for the general PIDEs and the corresponding error analysis are still missing in the literature. We exploit the Markovian property of the jump-diffusion process to prove that the temporal-spatial discrete scheme that can obtain the high-order convergence with respect to Δx to fill the gap. Specifically, temporal integrals in the probabilistic representation of the solution of PIDE are discretized using the implicit Euler method so that the temporal discrete scheme is stable and achieves first-order convergence in the weak sense. High-order numerical methods for discretizing the probabilistic representation, such as the Crank-Nicolson scheme, can also be used, but our goal is to develop stable and effective schemes for PIDEs that work in complicate domains rather than seeking high-order convergence numerical schemes. In the temporal-spatial discrete scheme, building composite quadrature rules is critical to approximate the conditional mathematical expectations with respect to both Brownian motion and the compound Poisson process. In particular, the integrals only with respect to Brownian motion are estimated by Gauss-Hermite rule and a specific form, e.g., Gauss-Legendre, Gauss-Jacobi and Newton-Cotes rules, for approximating the integrals with respect to the compound Poisson process can be determined based on the regularities of the kernel and the forcing term. Since our examples are high-dimensional cases, the tensor products of quadrature rules will bring

about the heavy computational cost. In this case, we use sparse-grid quadrature rules [7, 18, 19] to alleviate the explosion of computational cost from the curse of dimensionality. Moreover, to avoid the explosion of the total number of quadrature points with the number of time steps, we construct the piecewise pth Lagrange polynomials on a pre-determined spatial mesh to evaluate the integrands at quadrature points.

Chapter 4 is dedicated to describing the fractional Laplacian equations (partial differential equations with fractional Laplacian operator) and the corresponding numerical schemes based on the Gaussian approximation [4, 8]. As well known, the fractional Laplacian operator $(-\Delta)^{\alpha/2}$ is the infinitesimal generator of the symmetric α stable process for a fixed $\alpha \in (0, 2]$. And due to the Gaussian approximation method for stable processes, the stable processes can be approximated by an appropriate Lévy processes, i.e., a combination of an appropriate compound Poisson process and a Brownian motion with small variance, which is related to PIDEs studied by Chapter 3. The main contributions of this chapter are as follows: we simulate two symmetric α stable processes with two different $\alpha = 0.5, 1.5$ in one-dimensional space by Lévy processes to illustrate the accuracy for Gaussian approximation. Next, we construct the probabilistic schemes for the PIDE model which is the approximation of the fraction Laplacian equation in three-dimensional space. We aim to give a novel numerical approach for approximating the solution of the fractional Laplacian equation.

In Chapter 5, we research the PIDE with volume constraints, which is a natural extension, to the nonlocal case, of boundary conditions for local PDEs. Compared with the PIDEs model in the unbounded domain $\mathbb{R}^d$, the main issue in solving the PIDEs with volume constraint is the low accuracy for approximating the probabilistic representation of the solution near the Dirichlet boundary. To get the discretization approximation of the probabilistic representation of the solution, we divide the probabilistic representation into two parts based on two event subsets, i.e., one describes the event that the state of the underlying stochastic process remains in bounded domain after a fixed time s, another one describes the event that the state of underlying has exited before or on a fixed time s. In order to avoid accurately approximating the exit probability, the key is to set all interior grid points are sufficiently far from the boundary such that the underlying stochastic process starting from any interior node of will have a tiny exit

probability, we prove that given the temporal-spatial mesh $\mathcal{T} \times \mathcal{K}_{\Delta x}$, if the distance between the boundary and all the interior grid points is on the order of $\mathcal{O}((\Delta t)^{\frac{1}{2}-\varepsilon})$, then the exit probability is on the order of $\mathcal{O}((\Delta t)^{\varepsilon} \exp(-1/(\Delta t)^{2\varepsilon}))$. For instance, we set the mesh size Δx on the order of $\mathcal{O}(\sqrt{\Delta t})$, so that the process $\overleftarrow{X}_s^{t_{n+1},x}$ starting from any interior node of $\mathcal{K}_{\Delta x}$ will have a very small exit probability. In this case, piecewise cubic interpolation with $\mathcal{O}((\Delta x)^4)$ is needed to recover a $\mathcal{O}((\Delta t)^2)$ local error.

In Chapter 6, we focus on the RE dynamics in 3-dimensional space with coordinates (p, ξ, r), where p denotes the magnitude of the relativistic momentum and ξ the cosine of the pitch angle θ, i.e., the angle between the electron's velocity and the magnetic field, and r the minor radius. In this case, the dynamics of the distribution of electrons is determined by the Fokker-Planck (FP) equation describing the competition between the electric field acceleration, Coulomb collisions, synchrotron radiation damping, and sources describing the second generation of RE due to head-on collisions [40].The method we are proposing is different from those based on the solution of the Fokker-Planck equation, e.g., Ref. [25,27], and also different from the direct Monte-Carlo simulations. Instead, our approach is based on the Feynman-Kac formula relating the solution of the adjoint equation and the corresponding system of stochastic differential equations (SDEs). Specifically, we first represent the solution of the adjoint equation as a conditional expectation with respect to the underlying SDEs that describe the dynamics of the electrons. As such, the task of discretizing the differential operator becomes approximating the conditional expectation, which includes a quadrature rule for numerical integration and an interpolation strategy for evaluating the integrand at quadrature points. In this work, we use local hierarchical sparse grid methods [7,15,17,23,37] to handle the interpolation for two reasons. First, the terminal condition of the adjoint equation is discontinuous, and the adaptive refinement strategy of sparse grids can effectively capture such irregularity as well as control the growth of the total number of grid points. Second, the problem is considered in a high-dimensional space. The sparse gird can alleviate the computational cost.

Chapter 2

Problem definitions and mathematical preliminaries

2.1 Probability space, random variables and stochastic processes

Having stated mathematical models we plan to deal with, we need to find reasonable probabilistic concepts associated with the quantities mentioned and mathematical models. In this section, we will discuss

- Random Variables.
- Independence.
- Families of random variables.

Before defining those probabilistic concepts, we first introduce the definition of probability space. Assuming that sampling space Ω consists of all the possible outcomes of some experiments, i.e., the elementary events. Then the σ-algebra of subsets in Ω has the following properties:

1. $\emptyset \in \mathcal{F}$, where $\emptyset$ is the empty set in Ω.
2. $A \in \mathcal{F} \Rightarrow A^C \in \mathcal{F}$, where A^C is the complement of A in Ω, i.e., $A^C = \Omega \backslash A$.
3. $A_1, A_2, \ldots \in \mathcal{F} \Rightarrow \cup_{i=1}^{\infty} A_i \in \mathcal{F}$.

The triple $(\Omega, \mathcal{F}, P)$ is called a probability space, where the probability measure P is a map: $\mathcal{F} \rightarrow [0, 1]$. The study of probability spaces is often restricted to the study of complete probability space, i.e., spaces which are right continuous and contain all P-null sets.

Two sets A, $B \in \mathcal{F}$ are said to be independent if $P(A \cap B) = P(A)P(B)$. Two σ-algebras $\mathcal{F}_1$ and $\mathcal{F}_2$ of $\mathcal{F}$ are called independent if

$$P(A_1 \cap A_2) = P(A_1)P(A_2), \quad \forall A_1 \in \mathcal{F}_1, A_2 \in \mathcal{F}_2.$$

A random variable, usually written X, is a variable whose possible values are numerical outcomes of a random phenomenon. There are two types of random variables: discrete and continuous. And a stochastic or random process is a mathematical object usually defined as a family of random variables.

2.2 FBSDEs and PIDEs

In this section, we first introduce the decoupled forward-backward stochastic differential equations (FBSDEs) with jumps. Under the stochastic basis described in section 2.1, we assume that the filtration is generated by two independent stochastic processes: one is a standard Brownian motion $\{B_t\}_{t\geq 0}$ and the other is a Poisson random measure μ on $E \times [0,T]$, where the space $E = \mathbb{R}^d \setminus \{0\}$ is equipped with its Borel field $\mathcal{E}$, the compensator of μ is given as $\nu(de, dt) = \lambda(de)dt$, such that $\{\tilde{\mu}(A \times [0,t]) = (\mu - \nu)(A \times [0,t])\}_{0\leq t\leq T}$ is a martingale for all $A \in \mathcal{E}$. The measure λ is assumed to be σ-finite satisfying

$$\int_E (1 \wedge |e|^2)\lambda(de) < +\infty.$$

We introduce the following FBSDE with jumps

$$\begin{cases} X_t = X_0 + \int_0^t b(x, X_s)ds + \int_0^t \sigma(x, X_s)dW_s + \int_0^t \int_E c(s, X_{s-}, e)\tilde{\mu}(de, ds), \\ Y_t = \xi + \int_t^T f(s, X_s, Y_s, Z_s, \Gamma_s)ds - \int_t^T Z_s dW_s - \int_t^T U_s(e)\tilde{\mu}(de, ds), \end{cases} \tag{2.2.1}$$

where the drift coefficient $b : [0.T] \times \mathbb{R}^d \to \mathbb{R}^d$ and the local diffusion coefficient $\sigma : [0.T] \times \mathbb{R}^d \to \mathbb{R}^{d\times d}$ are assumed to be globally Lipschitz, and let the jump coefficient c be measurable

and such that for all $e \in E$,

$$|c(t,x,e)| \leq C(1 \wedge |e|), x \in \mathbb{R}^d,$$

$$|c(t,x,e) - c(t,x',e)| \leq C|x-x'|(1 \wedge |e|), x, x' \in \mathbb{R}^d,$$

and the process Γ_s is defined by $\Gamma_s = \int_E U_s(e)\eta(e)\lambda(de)$ for a bounded function η. For a fixed terminal time $T > 0$, we define some sets of random processes, we denote S^2 as the set of $\mathcal{F}_t$-adapted c$\grave{a}$dl$\grave{a}$g processes $\{Y_t, 0 \leq t \leq T\}$ such that

$$\|Y\|_{S^2}^2 := \mathbb{E}\left[\left(\sup_{0\leq t\leq T}|Y_t|\right)^2\right] < \infty,$$

$L^2(W)$ be the set of F_t-progressively measurable processes $\{Z_t, 0 \leq t \leq T\}$ which are such that

$$\|Z\|_{L^2(W)}^2 := \mathbb{E}\left[\int_0^T |Z_t|^2 dt\right] < \infty,$$

and by $L^2(\tilde{\mu})$ we denote the set of mapping $U : \Omega \times [0,T] \times E \to \mathbb{R}$ which are $\mathcal{P} \bigotimes \mathcal{E}$ measurable, where $\mathcal{P}$ denotes the σ-algebra of $\mathcal{F}_t$ predictable subsets of $\Omega \times [0,T]$, and such that

$$\|U\|_{L^2(\tilde{\mu})}^2 := \mathbb{E}\left[\int_0^T \int_E U_t(e)^2 \lambda(de)dt\right] < \infty.$$

Assuming the given functions f, φ are under standard assumptions stated in Theorem 2.1 in [5], the FBSDE (2.2.1) has unique solution $(Y_t, Z_t, U_t) \in S^2 \times L^2(W) \times L^2(\tilde{\mu})$.

Next, we give a short presentation of the nonlinear Feynman-Kac theory and the corresponding partial integro-differential equation. The extension of the nonlinear Feynman-Kac thoery states that the adapted solution (Y_t, Z_t, U_t) can be related to the unique viscosity solution $u(t,x) \in \mathcal{C}([0,T] \times \mathbb{R}^d)$ of the nonlinear PIDE with a second-order integral-differential operator, such a PIDE typically involves a local convection term and a nonlocal integral.

Let $[0,T]$ with $T > 0$ be a time interval, we consider the system of partial integro-differential equations of parabolic type,

$$\begin{cases} \dfrac{\partial u}{\partial t}(t,x) - \mathcal{K}[u](t,x) - f(t,x,u,\sigma\nabla u, \mathcal{B}[u]) = 0, & \forall (t,x) \in [0,T]\times\mathbb{R}^d, \\ u(T,x) = \varphi(x), & \forall x \in \mathbb{R}^d, \end{cases} \tag{2.2.2}$$

where the second-order integral-differential operator $\mathcal{K}$ is defined of the form

$$\begin{aligned} \mathcal{K}[u](t,x) =& \sum_{i=1}^{d} b_i \frac{\partial u}{\partial x_i}(t,x) + \frac{1}{2}\sum_{i,j=1}^{d} \sigma\sigma^\top \frac{\partial^2 u}{\partial x_i \partial x_j}(t,x) \\ &+ \int_E \left[u(t,x+c(t,x,e)) - u(t,x) - \sum_{i=1}^{d} \frac{\partial u}{\partial x_i}(t,x) c(t,x,e) \right] \lambda(e) de, \end{aligned} \tag{2.2.3}$$

and the operator $\mathcal{B}$ is an integral operator

$$\mathcal{B} = \int_E [u(t,x+c(t,x,e)) - u(t,x)].$$

We define all functions b, σ, f, φ and c the same way as in the FBSDE (2.2.1). Under the initial condition that $X_t = x$ for a fixed $t \in [0,T)$, the triple $(Y_s^{t,x}, Z_s^{t,x}, U_s^{t,x})$ for $t \le s \le T$ can be represented by

$$\begin{cases} Y_s^{t,x} = u(s, X_s^{t,x}), \\ Z_s^{t,x} = \sigma(s, X_s^{t,x})\nabla u(s, X_s^{t,x}), \\ U_s^{t,x} = u\left(s, X_{s-}^{t,x} + c(s-, X_{s-}^{t,x}, e)\right) - u(s, X_{s-}^{t,x}). \end{cases} \tag{2.2.4}$$

Some physics models use Fokker-Planck operator $\mathcal{J}$ as the second order integro-differential operator, which is the adjoint of the operator $\mathcal{K}$. Assume that the amplitude $c(t,x,e)$ is monotonic x, let the post-jump state value be $y = x + c(t,x,e)$ for each fixed (t,e) with inverse written as $x = y - \bar{c}(t,y,e)$ relating the pre-jump state to the post-jump state where $\bar{c}(t,y,e) = c(t,x,e)$. For a fixed (t,e), if $y = (I+c)x$, take inverse matrix to each side gets $x = (I+c)^{-1}y = (I-\bar{c})y$, thus $\bar{c} =: I - (I+c)^{-1}$ and $dx = (1 - \bar{c}_y(t,y,e))dy$, where $(1-\bar{c}_y(t,y,e))$ is the Jacobian of the inverse transformation, and the Fokker-Planck operator

is then defined as

$$\begin{aligned}\mathcal{F}[\phi](t,x) =& \frac{1}{2}\sum_{i,j=1}^{q}\frac{\partial^2(\sigma_i\phi\sigma_j^\top)}{\partial x_i\partial x_j}(t,x) - \sum_{i=1}^{q}\frac{\partial(b_i\phi)}{\partial x_i}(t,x) - \lambda\phi(t,x) \\ &+ \int_E\left[\sum_{i=1}^{q}\frac{\partial(\phi c)}{\partial x_i}(t,x,e) + \phi(x-\bar{c}(t,x,e))\left|1-\bar{c}_x(t,x,e)\right|\right]\lambda(de),\end{aligned} \tag{2.2.5}$$

where the Jacobian is

$$\left|1-\bar{c}_x(t,x,e)\right| = \left|1-\det\left[\frac{\partial\bar{c}_i(t,x,e)}{\partial x_j}\right]_{d\times d}\right|. \tag{2.2.6}$$

2.3 Sparse grid quadrature

The full tensor-product method for approximating multivariate functions and multidimensional integrals are difficult to implement due to the curse of dimensionality, i.e., computational complexity and storage dimensionality. The computational cost grows exponentially as the dimensionality of the problem increases, therefore, it is necessary to overcome this computational issue when the number of dimensional is large. Smolyak [42] first proposed the sparse tensor product approximation, which expects to preserve a high accuracy of the relative error with reducing the number of grid points compared to the full tensor-product rule. Sparse grid (SG) is a method for approximating multidimensional functions and integrals, where the approximation methods are constructed by using certain combinations of tensor products of one-dimensional rules. We give a brief framework of sparse grid quadrature which will be used in the following chapter and review some existing algorithms for the numerical integration of multivariate functions defined on d-dimensional cubes introduced in [47]. For simplicity, we consider the multidimensional integral of the function $g(x)$ defined on the d-dimensional hypercube, i.e., $x \in [-1,1]^d := \mathbb{D}$, linear transformation can be applied to translate $[-1,1]$ to any arbitrary interval $[a,b]$, and simple linear transformation will not affect the grid mesh. The multidimensional integral

$$I^d g := \int_{\mathbb{D}} g(x)dx,$$

which is approximated by a sequence

$$Q[g] := \sum_{i=1}^{N} w_i g(x_i),$$

where the points $\{x_i\}_{i=1}^{N}$ and the weights $\{w_i\}_i^N$ depend on the one dimensional rule and selected tensors.

Trapezoidal rule: The weights of the Newton-Cotes formulas is determined by integration of the Lagrange polynomial based on the equidistant abscissas, which includes closed version and open version. Sufficient accuracy requires large numbers of points, but it will lead to the formulas numerically instable, therefore iterated versions, e.g., the iterated trapezoidal rule, are commonly used. The trapezoidal rule is based on a piecewise linear approximation of function g, which is defined as

$$U_n g = \frac{1}{n+1}\left(\frac{3}{2}g\left(\frac{1}{n+1}\right) + \sum_{i=2}^{n-1} g\left(\frac{i}{n+1}\right) + \frac{3}{2}g\left(\frac{n}{n+1}\right)\right), \tag{2.3.1}$$

and for any function $g \in C^r([0,1])$, the error bounded by

$$\left|U_n g - I^1 g\right| = O(n^{-r}),$$

where the notion of regularity C^r is defined by [47]

$$C^r := \left\{ g : \mathbb{D} \to \mathbb{R}, \|\frac{\partial^s g}{\partial x^s}\|_\infty < \infty, s \leq r \right\}.$$

Clenshaw-Curtis rule: The Clenshaw-Curtis formulas is a stable and interpolatory integration formula and use the abscissas given as the extreme points of Chebyshev polynomials. For the quadrature formula U_n with degree fo U_n is $n-1$, the error bound is given as

$$\left|U_n g - I^1 g\right| = O(n^{-r}),$$

Table 2.1: Gauss Rules Summary

Name	Generalized Integral
Gauss-Patterson (nested rule):	$\int_{-a}^{b} g(x)dx$
Gauss-Legendre:	$\int_{-a}^{b} g(x)dx$
Gauss-Chebyshev type 1:	$\int_{-a}^{b} g(x)(b-x)^{-0.5}(x-a)^{-0.5}dx$
Gauss-Chebyshev type 2:	$\int_{-a}^{b} g(x)(b-x)^{0.5}(x-a)^{0.5}dx$
Gauss-Gegenbauer:	$\int_{-a}^{b} g(x)(b-x)^{\alpha}(x-a)^{\alpha}dx$
Gauss-Jacobi	$\int_{-a}^{b} g(x)(b-x)^{\alpha}(x-a)^{\beta}dx$
Gauss-Laguerre	$\int_{0}^{\infty} g(x)(x-a)^{\alpha}e^{-b(x-a)}dx$
Gauss-Hermite	$\int_{\infty}^{\infty} g(x)(x-a)^{\alpha}e^{-b(x-a)^2}dx$

the abscissas are given by

$$x_i = \frac{1}{2}\left(1 - cos\left(\frac{\pi i}{n+1}\right)\right),$$

for $i = 1, \ldots, n$, and the weights for add n are obtained

$$w_i = \frac{2}{n+1}\sin\left(\frac{\pi i}{n+1}\right)\sum_{j=1}^{(n+1)/2}\frac{1}{2j-1}\sin\left(\frac{(2j-1)\pi i}{n+1}\right).$$

Gauss formulas: Gauss formulas can achieve the maximum possible polynomial degree of exactness of $2n-1$, Gauss-Patterson is the only nested rule, and for the case of the unit weight function, the abscissas are the zeros of the Legendre polynomial and the weights are computed by integrating the associated Lagrange polynomials. We list some Gauss rules and the corresponding generalized integrals Table 2.1, studying details refers to [43].

2.4 High interpolation on Delaunay triangulation

To introduce Delaunay triangulation (DT), we recall some definitions studied by [36],

- **simplex** is the convex hull of $d+1$ affinely independent points in $\mathcal{R}^d$.
- **circumsphere** is the sphere through the vertices of a simplex.
- **flat** is an affine subspace of dimension k, where $k < d$.

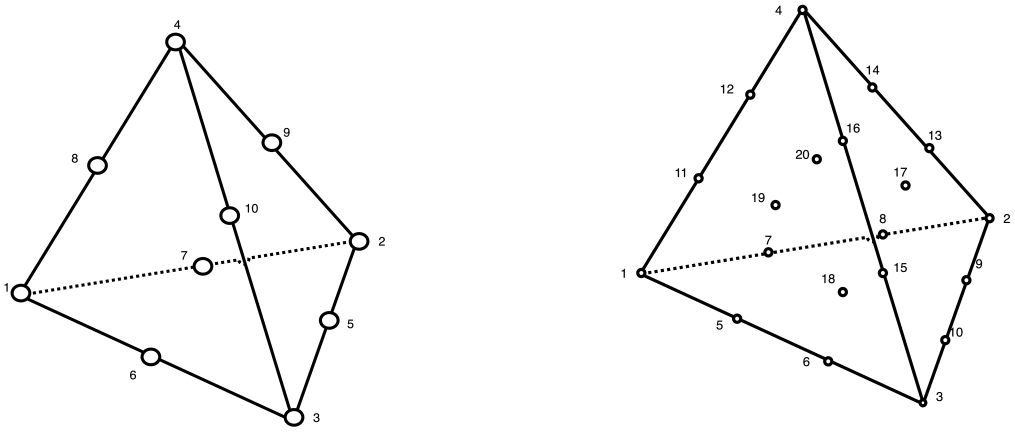

Figure 2.1: Quadratic and Cubic elements in Tetrahedron

In 3D space, the convex hull of a given set P of n points is decomposed into tetrahedra, Delaunay triangulation is the result from the Lawson flip algorithm such that the circumsphere of every d-simplex is empty, i.e., it does not contain any the given points in its interior and the amount of tetrahedra is not known in advance and can vary from $\mathcal{O}(n)$ to $\mathcal{O}(n^2)$. There are some properties of DT as follows [16]:

- DT is unique and no $d+2$ points lie on a same d-sphere and no $k+2$ points lie on a common k-flat in non-degenerate position.
- DT includes $\mathcal{O}(n^{\frac{d}{2}})$ simplices at most.
- DT has the minimizes the maximum radius of any simplex enclosing sphere.

High order elements: Quadratic and Cubic tetrahedron elements are 10 and 20 nodes shown in figure 2.4, respectively. The quadratic element consists of six nodes at the middle of the edges of the tetrahedron element and four-nodal, the corresponding shape functions based

on the barycentric coordinates $L_i, i = 1, 2, 3, 4$, are given as follows:

$$\begin{cases} N_i = (2L_i - 1)L_i, \quad i = 1, 2, 3, 4, \\ N_5 = 4L_2L_3, \quad N_6 = 4L_1L_3, \\ N_7 = 4L_1L_2, \quad N_8 = 4L_1L_4, \\ N_9 = 4L_2L_4, \quad N_{10} = 4L_3L_4. \end{cases} \tag{2.4.1}$$

The 20-node tetrahedron element is a cubic element consisting of four-nodal, two additional nodes are added even on each edge of the element, and four-node control-face nodes given as:

$$\begin{cases} N_i = \dfrac{1}{2}(3L_i - 1)L_i, \quad i = 1, 2, 3, 4 \\ N_5 = \dfrac{9}{2}(3L_1 - 1)L_1L_3, \quad N_6 = \dfrac{9}{2}(3L_3 - 1)L_1L_3, \\ N_7 = \dfrac{9}{2}(3L_1 - 1)L_1L_2, \quad N_8 = \dfrac{9}{2}(3L_2 - 1)L_1L_2, \\ N_9 = \dfrac{9}{2}(3L_2 - 1)L_2L_3, \quad N_{10} = \dfrac{9}{2}(3L_3 - 1)L_2L_3, \\ N_{11} = \dfrac{9}{2}(3L_1 - 1)L_1L_4, \quad N_{12} = \dfrac{9}{2}(3L_1 - 1)L_1L_4, \\ N_{13} = \dfrac{9}{2}(3L_2 - 1)L_2L_4, \quad N_{14} = \dfrac{9}{2}(3L_4 - 1)L_2L_4, \\ N_{15} = \dfrac{9}{2}(3L_3 - 1)L_3L_4, \quad N_{16} = \dfrac{9}{2}(3L_4 - 1)L_3L_4, \\ N_{17} = 27L_2L_3L_4, \quad N_{18} = 27L_1L_2L_3, \\ N_{19} = 27L_1L_3L_4, \quad N_{20} = 27L_1L_2L_4. \end{cases} \tag{2.4.2}$$

Chapter 3

Probabilistic schemes for the partial integro-differential equations in unbounded domains

3.1 Problem setting

We introduce in this section the partial integro-differential equation (PIDE) in the temporal-spatial domain $[0,T]\times\mathbb{R}^d$ with $T>0$, $d=1,2,3$, i.e.,

$$\begin{aligned}\frac{\partial u}{\partial t}(t,x)-\mathcal{L}[u](t,x)&=f(t,x,u),\quad \forall (t,x)\in[0,T]\times\mathbb{R}^d,\\ u(0,x)&=\varphi(x),\qquad \forall x\in\mathbb{R}^d,\end{aligned}\tag{3.1.1}$$

where f is the forcing and φ is the initial condition. The partial integro-differential operator $\mathcal{L}$ is of the form

$$\begin{aligned}\mathcal{L}[u](t,x)=&-\sum_{i=1}^{d}\frac{\partial}{\partial x_i}[b_i(t,x)u(t,x)]+\sum_{i,j=1}^{d}\frac{\partial^2}{\partial x_i\partial x_j}[K_{ij}(t,x)u(t,x)]\\ &+\int_E[u(t,x+c(t,e))-u(t,x)]\gamma(e)de,\end{aligned}\tag{3.1.2}$$

where $b(t,x)\in\mathbb{R}^d$ is the drift vector, $K(t,x)\in\mathbb{R}^{d\times d}$ is the local diffusion tensor, the nonlocal kernel $\gamma(e)$ is assumed to be *symmetric* and *integrable*, i.e.,

$$\gamma(e)\geq 0\ \ \forall e\in\mathbb{R}^d\ \text{ and }\ \lambda=\int_{\mathbb{R}^d}\gamma(e)de<\infty,\ \ \rho(e)=\gamma(e)/\lambda,\tag{3.1.3}$$

and $c(t,e) \in \mathbb{R}^d$ with $E := \mathbb{R}^d \backslash \{0\}$ is the jump coefficient. For the sake of simplicity, here we assume the jump coefficient $c(t,e)$ satisfies

$$\int_E c(t,e)\gamma(e)de = 0.$$

The existence and uniqueness of the viscosity solution $u(t,x)$ of PIDE (3.1.1) was proved in [5]. In this work, we are interested in the numerical approximation of the viscosity solution $u(t,x) \in C([0,T] \times \mathbb{R}^d)$. Since our numerical scheme is developed based on the probabilistic representation of the viscosity solution, we now briefly recall the connection between the PIDE and a stochastic differential equation (SDE) driven by Lèvy processes. Let $(\Omega, \mathcal{G}, (\mathcal{G}_t)_{)\leq t\leq T}, \mathbb{P})$ be a stochastic basis satisfying the completeness and right continuity, where the filtration $(\mathcal{G}_t)_{0\leq t\leq T}$ is generated by two mutually independent random processes, i.e., a d-dimensional Brownian motion B_t and a Poisson random measure $\pi(A,t)$ defined on $E \times [0,T]$. The compensator of π and the compensated Poisson random measure are denoted by $z(de,dt) = \gamma(e)de\,dt$ and $\tilde{\pi}(de,dt) = \pi(de,dt) - z(de,dt)$, respectively, such that $\{\tilde{\pi}(A \times [0,t]) = (\pi - z)(A \times [0,t])\}_{0\leq t\leq T}$ is a martingale for all $A \in \mathcal{E}$ with $\mathcal{E}$ being the Borel field of $E\backslash\{0\}$. Then, the operator $\mathcal{L}$ in (3.1.2) is the Fokker-Planck operator of the SDE defined in $(\Omega, \mathcal{G}, (\mathcal{G}_t)_{)\leq t\leq T}, \mathbb{P})$, i.e.,

$$V_t = V_0 + \int_0^t b(s,V_s)ds + \int_0^t \sigma(s,V_s)dB_s - \int_0^t \int_E c(s,e)\tilde{\pi}(de,ds), \tag{3.1.4}$$

where b and c are defined in (3.1.2), σ is the diffusion coefficient such that the local diffusion tensor $K(t,x)$ in (3.1.2) satisfies $K = \frac{1}{2}\sigma\sigma^\top$.

3.2 The probabilistic numerical schemes

The nonlinear Feynman-Kac theory studied by [5, 50] establishes a link between the partial differential equations (PDEs) and stochastic processes. Based on that relationship we propose the probabilistic numerical schemes of the viscosity solution $u(t,x)$ of PIDE (3.1.1). Specifically, we derive the probabilistic representation of u in section 3.2.1 and propose the numerical

scheme in temporal discretization in section 3.2.2 and the temporal-spatial discrete scheme is constructed in section 3.2.3.

3.2.1 The probabilistic representation

To build the explicit probabilistic representation of $u(t,x)$ of initial value PIDE (3.1.1) in arbitrary time interval $[0,T]$, we consider the underlying jump-diffusion process being *backward* so that the probabilistic representation is forward. In order to avoid the construction of *backward* filtration in the whole time interval $[0,T]$ that is an uncommon definition in stochastic theory, we consider the PIDE (3.1.1) in each subinterval $[t_n, t_{n+1}]$ independently. To proceed it, we introduce a uniform partition for the time interval $[0,T]$, i.e.,

$$\mathcal{T} = \{0 = t_0 \leq \cdots \leq t_N = T\},$$

with $\Delta t = t_i - t_{i-1}$ for $1 \leq i \leq N$. It is known that the non-divergence structure of the partial intego-differential operator is a prerequisite for applying the nonlinear Feynman-Kac theory, we hence transform the partial integro-differential operator $\mathcal{L}$ (3.1.2) into the non-divergence form and define new drift functions

$$\beta_i(t,x) := 2\sum_{j=1}^{q} \frac{\partial K_{ij}(t,x)}{\partial x_j}(t,x) - b_i(t,x)$$

the PIDE (3.1.1) in $[t_n, t_{n+1}]$ can be rewritten as

$$\begin{cases} \dfrac{\partial u}{\partial t}(t,x) - \mathcal{L}^*[u](t,x) = g(t,x,u), \forall (t,x) \in [t_n, t_{n+1}] \times \mathbb{R}^d, \\ u(t_n,x) = \varphi(t_n,x), \forall x \in \mathbb{R}^q, \end{cases} \tag{3.2.1}$$

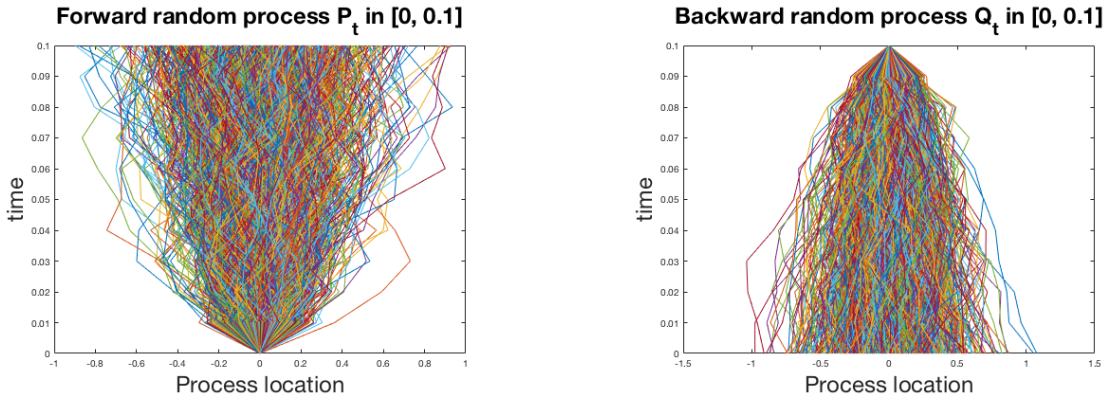

Figure 3.1: Brownian motions simulation

where the non-divergence form operator $\mathcal{L}^*$ and forcing g are expressed as

$$\begin{aligned}\mathcal{L}^*[u](t,x) :=& \sum_{i=1}^{d} \beta_i(t,x)\frac{\partial u}{\partial x_i}(t,x) + \sum_{i,j=1}^{d} K_{ij}(t,x)\frac{\partial^2 u}{\partial x_i x_j}(t,x) \\ &+ \int_E \left[u(t,x+c(t,e)) - u(t,x)\right]\gamma(e)de, \qquad (3.2.2)\\ g(t,x,u) :=& f(t,x,u) + \left[\sum_{i,j=1}^{d}\frac{\partial^2 K_{ij}}{\partial x_i \partial x_j}(t,x) - \sum_{i=1}^{d}\frac{\partial b_i}{\partial x_i}(t,x)\right] u(t,x).\end{aligned}$$

The operator $-L^*[u](t,x)$ is the infinitesimal generator of the *backward* stochastic process $\overleftarrow{X}_s^{t_{n+1},x}$ for $s \in [t_n, t_{n+1}]$, under the initial condition that $\overleftarrow{X}_{t_{n+1}} = x$, which is given by

$$\overleftarrow{X}_s^{t_{n+1},x} = x - \int_{t_{n+1}}^{s} \beta(t, \overleftarrow{X}_t^{t_{n+1},x})dt + \int_{t_{n+1}}^{s} \sigma(t, \overleftarrow{X}_t^{t_{n+1},x})d\overleftarrow{W}_t + \int_{t_{n+1}}^{s}\int_E c(t,e)\tilde{\mu}(de,dt), \quad (3.2.3)$$

where $\overleftarrow{W}_t$ is the *backward* standard Brownian motion and $\tilde{\mu}$ is a compensated Poisson random measure on E that has the same jump intensity with $\tilde{\pi}$. Details about *backward* filtration and stochastic processes are attached in Appendix 7.3.

To make it easy to understand the difference between *Forward* random process and *backward* random process, we simulate two typical random processes in $[0, 0.1]$, i.e., $dP_t = dB_t$ and $dQ_t = d\overleftarrow{W}_t$. See Figure 3.1.

To derive the probabilistic representation of the viscosity solution $u(t,x)$, it is essential to introduce the mathematical expectation notation, i.e., $\mathbb{E}^x_{t_{n+1}}[\cdot]$, which is crucial in stochastic

area and will be used quite frequently. Also denote, for $0 \leq n \leq N-1$,

$$\mathbb{E}^x_{t_{n+1}}[\cdot] := \mathbb{E}[\cdot|\overleftarrow{X}_{t_{n+1}} = x] \tag{3.2.4}$$

defined on the complete stochastic space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t_n \leq t \leq t_{n+1}}, \mathbb{P})$, where the filtration $(\mathcal{F}_t)_{t_n \leq t \leq t_{n+1}}$ is generated by *backward* random processes $\overleftarrow{W}_t$ and $\mu(A,t)$ in $t \in [t_n, t_{n+1}]$. Since the *backward* random process $\{\overleftarrow{X}_s^{t_{n+1},x}, s \in [t_n, t_{n+1}]\}$ is only defined on $[t_n, t_{n+1}]$, the initial condition $\overleftarrow{X}_{t_{n+1}} = x$ is independent of filtration $(\mathcal{F}_t)_{t_{n+1} \leq t \leq T}$. Based on the construction of $\mathbb{E}^x_{t_{n+1}}[\cdot]$ and the Feynaman-Kac theory introduced in [5], the probabilistic representation of $u(t_{n+1}, x)$ is derived as

$$u(t_{n+1}, x) = \mathbb{E}^x_{t_{n+1}}[u(t_n, \overleftarrow{X}_{t_n}^{t_{n+1},x})] - \int_{t_{n+1}}^{t_n} \mathbb{E}^x_{t_{n+1}}\left[g(t, \overleftarrow{X}_t^{t_{n+1},x}, u(t, \overleftarrow{X}_t^{t_{n+1},x})\right] dt. \tag{3.2.5}$$

The construction of probabilistic representation of $u(t_{n+1}, x)$ (3.2.5) is given in Appendix 7.4.

3.2.2 The temporal discrete scheme

Many classical numerical schemes have been proposed to accurately approximate the stochastic integrals in the jump-diffusion process (3.2.3), i.e., the Euler scheme studied by [51, 52], the Milstein scheme, the Itô-Taylor type weak and strong schemes of order β [24, 38]. The explicit Euler method is used in our numerical schemes to approximate the temporal integrals that avoid solving the underlying nonlinear systems. Under the initial condition that $\overleftarrow{X}_{t_{n+1}} = x$ in $[t_n, t_{n+1}]$, the approximation X^n is satisfying

$$\begin{aligned} X^n =& x - \beta(t_{n+1}, x) \int_{t_{n+1}}^{t_n} dt + \sigma(t_{n+1}, x) \int_{t_{n+1}}^{t_n} d\overleftarrow{W}_t + \int_E c(t_{n+1}, e) \int_{t_{n+1}}^{t_n} \tilde{\mu}(de, dt) \\ =& x + \beta(t_{n+1}, x)\Delta t + \sigma(t_{n+1}, x)\Delta\overleftarrow{W}_{t_{n+1}} + \sum_{k=1}^{N_{\Delta t}} c(t_{n+1}, e_k), \end{aligned} \tag{3.2.6}$$

where $N_{\Delta t}$, i.e., the number of jumps in time interval Δt, is the underling *backward* Poisson process with intensity λ and the *backward* Wiener process $\Delta\overleftarrow{W}_{t_{n+1}} := \overleftarrow{W}_{t_n} - \overleftarrow{W}_{t_{n+1}}$. Note that the integrand $\mathbb{E}^x_{t_{n+1}}[g(t, \overleftarrow{X}_t^{t_{n+1},x}, u(t, \overleftarrow{X}_t^{t_{n+1},x}))]$ in equation (3.2.5) is a deterministic function

of $t \in [t_n, t_{n+1}]$, the implicit Euler scheme can be applied that guarantees stability and achieves overall first order convergence with respect Δt, i.e.

$$u(t_{n+1}, x) = \mathbb{E}^x_{t_{n+1}}[u(t_n, \overleftarrow{X}^{t_{n+1},x}_{t_n})] + \Delta t g(t_{n+1}, x, u(t_{n+1}, x)) + R_{n+1}, \tag{3.2.7}$$

where the residual R_{n+1} is equal to

$$R_{n+1} = -\int_{t_{n+1}}^{t_n} \mathbb{E}^x_{t_{n+1}}[g(t, \overleftarrow{X}^{t_{n+1},x}_t, u(t, \overleftarrow{X}^{t_{n+1},x}_t))]dt - \Delta t g(t_{n+1}, x, u(t_{n+1}, x)). \tag{3.2.8}$$

We combine the approximation of the jump-diffusion process (3.2.6) and the temporal approximation of the probabilistic representation (3.2.7) to propose the temporal discrete scheme for the viscosity solution $u(t_{n+1}, x)$ as follows:

Given the initial random variable u_n, we regard $u_{n+1}(x)$ as the approximation of $u(t_{n+1}, x)$ at the time level $t = t_{n+1}, \forall n = 0, \cdots, N-1$, which satisfies

$$\begin{cases} X^n = x + \beta(t_{n+1}, x)\Delta t + \sigma(t_{n+1}, x)\Delta \overleftarrow{W}_{t_{n+1}} + \sum_{k=1}^{N_{\Delta t}} c(t_{n+1}, e_k), \\ u_{n+1}(x) = \mathbb{E}^x_{t_{n+1}}[u_n(X^n)] + \Delta t g(t_{n+1}, x, u_{n+1}(x)). \end{cases} \tag{3.2.9}$$

Actually, the generalized θ schemes studied by [52, 53] are also applicable for estimating the time integral in (3.2.5), e.g., $\theta = \frac{1}{2}$ leads to the overall second-order Crank-Nicolson scheme incorporating with high order numerical schemes for approximating the jump-diffusion process, e.g., the Itô-Taylor type weak schemes of order $\beta = 2$. We focus on building stable and accurate numerical schemes for solving the PIDE (3.1.1) in this thesis, high order numerical schemes will be studied in the future work.

3.2.3 The temporal-spatial discrete scheme

Before studying the temporal-spatial discrete scheme, we first define the intensity of Poisson jumps $\lambda(de)$ and the probability measure of the jump height $\rho(e)de$. We define $\lambda(de) := \gamma(e)de$ for $e \in E$, which is σ-finite due to the integrability assumption of $\gamma(e)$ in (3.1.3) and $\rho(e)de$ being the probability measure of the jump height $c(t, e)$ satisfies $\int_E \rho(e)de = 1$.

The temporal discrete scheme (3.2.9) is not computationally feasible because the involved mathematical expectation $\mathbb{E}^x_{t_{n+1}}[\cdot]$ is defined in the whole $\mathbb{R}^d$, thus an effective spatial discretization is necessary to estimate $\mathbb{E}^x_{t_{n+1}}[\cdot]$, we partition the d-dimensional Euclidean space $\mathbb{R}^d$ by a tensor-product grid

$$\mathcal{K} = \mathcal{K}^1 \times \mathcal{K}^2 \times \times \cdots \times \mathcal{K}^d,$$

where $\mathcal{K}^l$ for $l = 1, \ldots, d$ is a uniform partition of the one-dimensional real space $\mathbb{R}$, i.e.,

$$\mathcal{K}^l = \{x^l_j | x^l_j \in R, j \in Z, x^l_j < x^l_{j+1}, \lim_{j\to+\infty} x^l_j = +\infty, \lim_{j\to-\infty} x^l_j = -\infty\}.$$

Based on the tensor grid $\mathcal{K}$, we turn to develop quadrature rules to estimate the mathematical expectation $\mathbb{E}^x_{t_{n+1}}[\cdot]$ in (3.2.9) at $(t_{n+1}, x_j) \in \mathcal{T} \times \mathcal{K}$. It is observed that $\mathbb{E}^{x_j}_{t_{n+1}}[\cdot]$ is defined with respect to the probability measure of the *backward* incremental stochastic process $\Delta\overleftarrow{X}^n_j := X^n_j - x_j = \Phi(t_{n+1}, t_n, x_j)$ that is determined by the explicit Euler scheme, i.e.,

$$\Phi(t_{n+1}, t_n, x_j) = X^n_j - x_j = \beta(t_{n+1}, x_j)\Delta t + \sigma(t_{n+1}, x_j)\Delta\overleftarrow{W}_{t_{n+1}} + \sum_{k=1}^{N_{\Delta t}} c(t_{n+1}, e_k), \quad (3.2.10)$$

and the number of jumps of $\overleftarrow{X}_t$ within $[t_n, t_{n+1})$, denoted by N_t, follows a Poisson distribution with intensity λ. The size of each jump $c(t_{n+1}, e)$ follows the distribution $\rho(e)de$. Next we observe that $\Delta\overleftarrow{W}^l_{t_{n+1}} = \sqrt{\Delta t}\xi^l$ for $l = 1, 2, \ldots, d$, where random variable ξ^l follows the standard normal distribution $N(0, 1)$, thus we denote by $\varrho(\xi) := (\varrho(\xi^1), \ldots, \varrho(\xi^d))$ the probability density function of $\xi := (\xi^1, \ldots, \xi^d)$, the mathematical expectation $\mathbb{E}^{x_j}_{t_{n+1}}[u^n]$ can be represented

as

$$
\begin{aligned}
\mathbb{E}_{t_{n+1}}^{x_j}[u^n] =& \sum_{m=0}^{\infty} P(N_{\Delta t}=m)\mathbb{E}\Bigg[u^n\Bigg(x_j+\beta(t_{n+1},x_j)\Delta t+\sigma(t_{n+1},x_j)\sqrt{\Delta t}\xi \\
&+\sum_{k=1}^{m} c(t_{n+1},e_k)\Bigg)\Bigg] \\
=& \sum_{m=0}^{\infty} e^{-\lambda\Delta t}\frac{(\lambda\Delta t)^m}{m!}\mathbb{E}\Bigg[u^n\Bigg(x_j+\beta(t_{n+1},x_j)\Delta t+\sigma(t_{n+1},x_j)\sqrt{\Delta t}\xi \\
&+\sum_{k=1}^{m} c(t_{n+1},e_k)\Bigg)\Bigg] \\
=& e^{-\lambda\Delta t}\int_{\mathbb{R}^d} u^n\big(x_j+\beta(t_{n+1},x_j)\Delta t+\sigma(t_{n+1},x_j)\sqrt{\Delta t}\xi\big)\varrho(\xi)d\xi \\
&+\sum_{m=1}^{\infty} e^{-\lambda\Delta t}\frac{(\lambda\Delta t)^m}{m!}\int_{\mathbb{R}^d}\int_E\cdots\int_E u^n\Bigg(x_j+\beta(t_{n+1},x_j)\Delta t \\
&+\sigma(t_{n+1},x_j)\sqrt{\Delta t}\xi+\sum_{k=1}^{m} c(t_{n+1},e_k)\Bigg)\varrho(\xi)\prod_{k=1}^{m}\rho(e_k)de_1\cdots de_m d\xi,
\end{aligned}
\tag{3.2.11}
$$

where $c(t_{n+1},e_k)$ for $k=1,\ldots,m$ is the height of the kth jump. We observe that the probability of taking first m jumps in Δt, i.e., $P(N_{\Delta t}=m)$ is of order $\mathcal{O}((\lambda\Delta t)^m)$. Given a prescribed accuracy, the mathematical expectation $\mathbb{E}_{t_{n+1}}^{x_j}[u^n]$ can be approximated by the sum of a finite sequence by remaining finite number of jumps. For instance, if we expect a first-order convergence with respect to Δt, i.e., the local truncation error R_{n+1} should be of order $\mathcal{O}((\lambda\Delta t)^2)$, the first two terms should be retained assuming λ is of order $\mathcal{O}(1)$. We denote the approximation of $\mathbb{E}_{t_{n+1}}^{x_j}[u^n]$ by retaining the first M_u+1 terms by $\mathbb{E}_{t_{n+1},M_u}^{x_j}[u^n]$, where M_u means the number of jumps in $\mathbb{E}_{t_{n+1}}^{x_j}[u^n]$. Next, we propose appropriate quadrature rules to approximate multiple integrals in equation (3.2.11). To proceed, we first introduce two fundamental types of quadrature formulas aiming to the normal distribution and the specified distribution of the jump variable with compactly supported, respectively.

1. For a d-dimensional function $k(x)$, the Gauss-Hermite quadrature formula is

$$
\int_{\mathbb{R}^d} k(x)e^{-x^2}dx \approx \sum_{i=1}^{L} w_i k(a_i). \tag{3.2.12}
$$

where the quadrature weights $\{w_i\}_{i=1}^L$ and the quadrature root $\{a_i\}_{i=1}^L$ of the Gauss-Hermite quadrature can be found in [1],

2. For a d-dimensional function $k(x)$, the selected quadrature rule based on the probability density function $\varrho(\xi)$, e.g., Gauss-Legendre is available when $\varrho(\xi)$ follows a uniform distribution, is generalized as

$$\int_E k(\xi)\,\varrho(\xi)d\xi \approx \sum_{q=1}^{Q} \tilde{w}_q^k \left(\tilde{a}_q^k\right), \tag{3.2.13}$$

the set $\{\tilde{w}_q\}_{q=1}^Q$, $\{\tilde{a}_q\}_{q=1}^Q$ denote quadrature weights and quadrature points, respectively.

We need to approximate the $m \times d + d$ dimensional integrals in (3.2.11) by combining the above two basic quadrature rules depending on the properties of $\rho(e)$, $\varrho(\xi)$ and the smoothness of u^n. Without loss of generality, for $m = 0, \ldots, M_u$, we denote by $\{w_i, a_i\}$ and $\{\tilde{w}_q^m, \tilde{a}_q^{k,m}\}$, the set of weights and points of Gauss-Hermite rule and selected quadrature rule for estimating the integrals with respect to ξ and $\{e_1, \ldots, e_k\}$ in (3.2.11), where k represents kth jump for $1 \le k \le m$. Then the approximation of $\mathbb{E}_{t_{n+1}}^{x_j}[u^n]$, denoted by $\hat{\mathbb{E}}_{t_{n+1},M_u}^{x_j}[u^n]$ is given by

$$\begin{aligned}\hat{\mathbb{E}}_{t_{n+1},M_u}^{x_j}[u^n] =& e^{-\lambda\Delta t}\sum_{i=1}^{L} w_j u^n\Big(x_j + \beta(t_{n+1},x_j)\Delta t + \sigma(t_{n+1},x_j)\sqrt{2\Delta t}a_i\Big) \\ &+ \sum_{m=1}^{M_u} e^{-\lambda\Delta t}\frac{(\lambda\Delta t)^m}{m!}\sum_{i=1}^{L}\sum_{q=1}^{Q_m} w_i\tilde{w}_q^m u^n\Big(x_j + \beta(t_{n+1},x_j)\Delta t \\ &+ \sigma(t_{n+1},x_j)\sqrt{2\Delta t}a_i + \sum_{k=1}^{m} c(t_{n+1},\tilde{a}_q^{k,m})\Big).\end{aligned} \tag{3.2.14}$$

The quadrature points $\left\{x_j + \beta(t_{n+1},x_j)\Delta t + \sigma(t_{n+1},x_j)\sqrt{\Delta t}a_i + \sum_{k=1}^m c(t_{n+1},\tilde{a}_q^{k,m})\right\}_{i=1,q=1}^{L,Q}$ in (3.2.14) may not belong to the spatial grid $\mathcal{K}$, constructing a pth order piecewise Lagrange polynomial based on the spatial grid $\mathcal{K}$ is needed to compute u^n. We denote the pth order piecewise Lagrange polynomial by $u^{n,p}$, i.e., $u^{n,p}(x) := \sum_{i\in\mathbb{Z}^d} u_i^{n,p}\psi_i(x)$, where ψ_i is the pth order Lagrange nodal basis function and $u_i^{n,p}$ is an approximation of $u^n(x_i)$ studied by [46,50].

Based on the above discussion, given the random variables $u^n(x_j)$ for $j \in \mathbb{Z}$, n=1, 2, ...,

N-1, the temporal-spatial discrete scheme is

$$\begin{cases} X_j^n = x_j + \beta(t_{n+1}, x_j)\Delta t + \sigma(t_{n+1}, x_j)\Delta \overleftarrow{W}_{t_{n+1}} + \sum_{k=1}^{N_{\Delta t}} c(t_{n+1}, e_k), \\ u_j^{n+1,p} = \hat{\mathbb{E}}^{x_j}_{t_{n+1},M_u}[u^{n,p}] + \Delta t g(t_{n+1}, x_j, u_j^{n+1,p}), \\ u^{n+1}(x) = \sum_{i=1}^{p+1} \left\{ \hat{\mathbb{E}}^{x_{j_i}}_{t_{n+1},M_u}[u^{n,p}] + \Delta t g(t_{n+1}, x_{j_i}, u_{j_i}^{n+1,p}) \right\} \psi_{j_i}(x), \end{cases} \tag{3.2.15}$$

where the interpolation points $\{x_{j_i}\}_{i=1}^{p+1} \in \mathcal{K}$ are the closet $p+1$ neighboring points of x, the quantitiy M_u denote the number of Poisson jumps in the approximation of $\mathbb{E}^{x_{j_i}}_{t_{n+1}}[u]$ and $\{\psi_{j_i}(x)\}_{i=1}^{p+1}$ is the set of p-th order piecewise Lagrange basis function associated with $\{x_{j_i}\}_{i=1}^{p+1}$. Since our scheme is an explicit time-stepping scheme, i.e., $u^{n+1}(x)$ at each time point, which can avoid the difficulty of solving the linear or nonlinear systems.

3.3 Error estimates

To analyze the relative errors of numerical schemes at temporal layer $t = T$ starting from initial point $t = 0$, all stochastic processes should be considered under a complete stochastic space in $[0, T]$. To proceed, we redefine some concepts and notations introduced in section 3.2, which will not affect the structure of proposed numerical schemes. For a fixed $T \in [0, \infty)$, we consider the *backward* jump-diffusion process $\overleftarrow{X}_t$ (3.2.3) in the stochastic space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{0\le t\le T}, \mathbb{P})$, where $\mathcal{F}_t$ is generated by two *backward* random processes $\overleftarrow{W}_t$ and μ on $E\times[0,T]$, i.e., $\mathcal{F}_t := \mathcal{F}^{\overleftarrow{W}}_{t,T} \vee \mathcal{F}^{\mu}_{t,T}$, where for any process $\overleftarrow{\eta}_t$, $\mathcal{F}^{\overleftarrow{\eta}}_{t,T} := \sigma\{\overleftarrow{\eta}_s - \overleftarrow{\eta}_t; t \le s \le T\}$. Note that the decreasing collection $\{\mathcal{F}_t, t \in [0,T]\}$ is a decreasing filtration. In this case, the *bakward* Brownian motion $\{\overleftarrow{W}_t\}_{0\le t\le T}$ and the compensator $\{\tilde{\mu}\}_{0\le t\le T}$ are martingales with respect to $(\mathcal{F}_t)_{0\le t\le T}$, where the initial condition $\overleftarrow{X}_{t_{n+1}} = x$ in $[t_n, t_{n+1}]$ is dependent with filtration $\mathcal{F}_{t_{n+1}}$, and the conditional mathematical expectation notation $\mathbb{E}^x_{t_{n+1}}[\cdot]$ is $\mathcal{F}_{t_{n+1}}$ measurable. Moreover, we assume the terminal value of $\overleftarrow{X}_T$ is $\mathcal{F}_T$ measurable and $\mathbb{E}[|\overleftarrow{X}_T|^2] \le \infty$.

3.3.1 Error estimate of the temporal discrete scheme

In this section, we analyze the temporal numerical scheme (3.2.9). The explicit Euler scheme of the *backward* jump-diffusion (3.2.6) achieves first-order convergence that has been discussed in the literature [38] and our strategy for estimating truncation error of u_{n+1} follows the approach of the previous work [50, 54], i.e., construct the upper bound of the global error by accumulating all local truncation errors recursively.

We set $x = X^{n+1}$ in (3.2.9) and define the errors of $u_{n+1}(X^{n+1})$ and $g^{n+1} := g(t_{n+1}, X^{n+1}, u_{n+1}(X^{n+1}))$, respectively, by

$$\begin{aligned} e_u^{n+1} &:= u(t_{n+1}, X^{n+1}) - u_{n+1}(X^{n+1}), \\ e_g^{n+1} &:= g(t_{n+1}, X^{n+1}, u(t_{n+1}, X^{n+1})) - g^{n+1}, \end{aligned} \tag{3.3.1}$$

where $u(t_{n+1}, X^{n+1}) = \mathbb{E}[u(t_{n+1}, X_{t_{n+1}})|X_{t_{n+1}} = X^{n+1}]$. It should be noted that $u(t_n, \overleftarrow{X}_{t_n}^{t_{n+1}, X^{n+1}})$ and $u(t_n, X^n)$ for $0 \le n \le N-1$ are usually different stochastic processes because $\overleftarrow{X}_{t_n}^{t_{n+1}, X^{n+1}}$ and X^n are two different processes obtained by (3.2.3) and (3.2.6), respectively. As such, the residual $\mathcal{R}_u^{n+1}$ is introduced as

$$\mathcal{R}_u^{n+1} = \mathbb{E}_{t_{n+1}}^{X^{n+1}}[u(t_n, \overleftarrow{X}_{t_n}^{t_{n+1}, X^{n+1}}) - u(t_n, X^n)], \tag{3.3.2}$$

and the following properties of the explicit Euler scheme (3.2.6) used in estimating residual were studied by [38]:

1. Stability: for an integer $r > 0$, there exists a constant $C \in (0, \infty)$ such that

$$\max_{0 \le n \le N} \mathbb{E}[|X^n|^r] \le C(1 + [|X_T|^r]). \tag{3.3.3}$$

2. Approximation error: there exist a positive real number r such that for any function $g \in \mathcal{C}_P^4$, we have

$$|\mathbb{E}_{t_{n+1}}^{X^{n+1}}[g(\overleftarrow{X}_{t_n}^{t_{n+1}, X^{n+1}}) - g(X^n)]| \le C(1 + |X^{n+1}|^{2r})(\Delta t)^2, \tag{3.3.4}$$

where $\mathcal{C}_p^4$ denotes the set of fourth order continuously differentiable functions which together with their derivatives of order up to fourth have at most polynomial growth. One more estimate for the *backward* jump-diffusion process (3.2.3) requires in the following arguments is that under Assumption 4.1 in [54], if $\mathbb{E}[|X_T|^{2m}] < \infty$ for some integer $m \geq 1$, the solution of the jump-diffusion in (3.2.3) has the estimate

$$\mathbb{E}_{t_{n+1}}^{X^{n+1}}[|\overleftarrow{X}_s^{t_{n+1},X^{n+1}}|^{2m}] \leq (1 + \mathbb{E}_{t_{n+1}}^{X^{n+1}}[|X^{n+1}|^{2m}])e^{C(t_{n+1}-s)}, \tag{3.3.5}$$

where $s \in [0, t_{n+1}]$ and C is a constant.

In the rest of section 3.3, we only analyze numerical schemes in one-dimensional case $(d = 1)$, and results can be extended to high-dimensional cases naturally. In addition to global Lipschitz continuity on forcing term g, we need to impose some regularities on coefficients in equation (3.2.1) to obtain error estimates. To proceed, we first introduce the notation of regularity:

$\mathcal{C}_b^{(k_1,\ldots,k_J)}(D_1 \times \cdots \times D_J) = \Big\{\phi : \prod_{j=1}^J D_j \to \mathbb{R} | \frac{\partial^{\alpha_1}\ldots\partial^{\alpha_J}\phi}{\partial^{\alpha_1}x_1\ldots\partial^{\alpha_J}x_J}$ is bounded and continuous for $0 \leq \alpha_j \leq k_j, j = 1, \ldots, J$, where $\alpha = (\alpha_1, \ldots, \alpha_J)$, where $J \in \mathbb{N}^+$.$\Big\}$

Based on the above assumptions and properties, we give the estimates of truncation errors R_{n+1} and $\mathcal{R}_u^{n+1}$ in the following Lemma.

Lemma 3.1. *Under Assumption 4.1 in [54], if* $\varphi(x) \in \mathcal{C}_b^{6+\alpha}(\mathbb{R})$ *with* $\alpha \in (0, 1)$, $\beta(t, x) \in \mathcal{C}_b^{2,4}([0, T] \times \mathbb{R})$, $g(t, x, y) \in \mathcal{C}_b^{2,4,4}([0, T] \times \mathbb{R}^2)$, $\sigma(t, x) \in \mathcal{C}_b^{2,4}([0, T] \times \mathbb{R})$ *and* $c(t, e) \in \mathcal{C}_b^{2,\infty}([0, T] \times \mathbb{R})$, *then*

$$\mathbb{E}[|R_{n+1}|^2] \leq C(\Delta t)^4, \quad \mathbb{E}[|\mathcal{R}_u^{n+1}|^2] \leq C(\Delta t)^4, \tag{3.3.6}$$

where C *is a positive constant depending on the upper bounds of* $\overleftarrow{X}_T$ *and derivatives of* β, σ, c, φ, g *and the jump intensity* λ.

Proof. For notational simplicity, we omit the superscripts of $\overleftarrow{X}_t^{t_{n+1},X^{n+1}}$ and define $F(t,\overleftarrow{X}_t) := g(t,\overleftarrow{X}_t,u(t,\overleftarrow{X}_t))$, then $F \in \mathcal{C}_b^{2,4}([0,T]\times\mathbb{R})$. We define three partial integro-differential operators,

$$\begin{aligned} L^0F(t,x) &:= \frac{\partial F}{\partial t}(t,x) - \beta(t,x)\frac{\partial F}{\partial x}(t,x) - K(t,x)\frac{\partial^2 F}{\partial x^2}(t,x) \\ &\quad - \int_E [F(t,x+c(t,e)) - F(t,x)]\lambda(de), \\ L^1F(t,x) &:= \sigma(t,x)\frac{\partial F}{\partial x}(t,x), \\ L^{-1}F(t,x) &:= F(t,x+c(t,e)) - F(t,x). \end{aligned} \tag{3.3.7}$$

Applying the Itô formula to $F(s,\overleftarrow{X}_s)$ for $t_n \le s \le t_{n+1}$, under the condition $\overleftarrow{X}_{t_{n+1}} = X^{n+1}$, we have

$$\begin{aligned} F(s,\overleftarrow{X}_s) =& F(t_{n+1},X^{n+1}) + \int_{t_{n+1}}^s L^0F(t,\overleftarrow{X}_t)dt + \int_{t_{n+1}}^s L^1F(t,\overleftarrow{X}_{t-})d\overleftarrow{W}_t \\ &+ \int_{t_{n+1}}^s \int_E L^{-1}F(t,\overleftarrow{X}_{t-})\tilde{\mu}(de,dt). \end{aligned} \tag{3.3.8}$$

Then substituting the above $F(s,\overleftarrow{X}_s)$ into the integral $\int_{t_{n+1}}^{t_n} \mathbb{E}_{t_{n+1}}^{X^{n+1}}[F(s,\overleftarrow{X}_s)]ds$, due to the martingale property, we get

$$\begin{aligned} &\int_{t_{n+1}}^{t_n} \mathbb{E}_{t_{n+1}}^{X^{n+1}}[F(s,\overleftarrow{X}_s)]ds \\ &= \int_{t_{n+1}}^{t_n} \mathbb{E}_{t_{n+1}}^{X^{n+1}}\Big[F(t_{n+1},X^{n+1}) + \int_{t_{n+1}}^s L^0F(t,\overleftarrow{X}_t)dt + \int_{t_{n+1}}^s L^1F(t,\overleftarrow{X}_{t-})d\overleftarrow{W}_t \\ &\quad + \int_{t_{n+1}}^s \int_E L^{-1}F(t,\overleftarrow{X}_{t-})\tilde{\mu}(de,dt)\Big]ds \\ &= \int_{t_{n+1}}^{t_n} \mathbb{E}_{t_{n+1}}^{X^{n+1}}[F(t_{n+1},X^{n+1})]ds + \int_{t_{n+1}}^{t_n}\int_{t_{n+1}}^s \mathbb{E}_{t_{n+1}}^{X^{t_{n+1}}}[L^0F(t,X_t)]dtds \\ &= -F(t_{n+1},X^{n+1})\Delta t + \int_{t_{n+1}}^{t_n}\int_{t_{n+1}}^s \mathbb{E}_{t_{n+1}}^{X^{n+1}}[L^0F(t,\overleftarrow{X}_t)]dtds \\ &= -F(t_{n+1},X^{n+1})\Delta t + \int_{t_{n+1}}^{t_n}\int_{t_{n+1}}^s \mathbb{E}_{t_{n+1}}^{X^{n+1}}\Big[L^0F(t_{n+1},X^{n+1}) \\ &\quad + \int_{t_{n+1}}^t L^0L^0F(z,\overleftarrow{X}_z)dz\Big]dtds \\ &= -F(t_{n+1},X^{n+1})\Delta t + \frac{1}{2}(\Delta t)^2 L^0F(t_{n+1},X^{n+1}) \\ &\quad + \int_{t_{n+1}}^{t_n}\int_{t_{n+1}}^s\int_{t_{n+1}}^t \mathbb{E}_{t_{n+1}}^{X^{n+1}}[L^0L^0F(z,\overleftarrow{X}_z)]dzdtds. \end{aligned} \tag{3.3.9}$$

Thus, Eq. (3.2.8) equals

$$\begin{aligned} R_{n+1} =& -\int_{t_{n+1}}^{t_n} \mathbb{E}_{t_{n+1}}^{X^{n+1}}[F(s,\overleftarrow{X}_s)]ds - \Delta t F(t_{n+1}, X^{n+1}) \\ =& -\frac{1}{2}(\Delta t)^2 L^0 F(t_{n+1}, X^{n+1}) - \int_{t_{n+1}}^{t_n}\int_{t_{n+1}}^{s}\int_{t_{n+1}}^{t} \mathbb{E}_{t_{n+1}}^{X^{n+1}}[L^0L^0F(z,\overleftarrow{X}_z)]dzdtds. \end{aligned} \tag{3.3.10}$$

Take the expectation $\mathbb{E}[\cdot]$ to R_{n+1}, by the Cauchy-Schwarz inequality, we have that

$$\begin{aligned} \mathbb{E}[|R_{n+1}|] \leq& \mathbb{E}\left[\left|\frac{1}{2}(\Delta t)^2 L^0 F(t_{n+1}, X^{n+1})\right|\right] \\ &+ \mathbb{E}\left[\left|\int_{t_{n+1}}^{t_n}\int_{t_{n+1}}^{s}\int_{t_{n+1}}^{t} \mathbb{E}_{t_{n+1}}^{X^{n+1}}[L^0L^0F(z,\overleftarrow{X}_z)]dzdtds\right|\right], \\ \leq& \frac{1}{2}(\Delta t)^2 \mathbb{E}[|L^0F(t_{n+1}, X^{n+1})|] \\ &+ (\Delta t)^{\frac{3}{2}}\sqrt{\int_{t_{n+1}}^{t_n}\int_{t_{n+1}}^{s}\int_{t_{n+1}}^{t} \mathbb{E}[|L^0L^0F(z,\overleftarrow{X}_z)|^2]dzdtds}, \\ \leq& \frac{1}{2}(\Delta t)^2 \mathbb{E}[|L^0F(t_{n+1}, X^{n+1})|] + \sup_{0\leq t\leq T}\sqrt{\mathbb{E}[|L^0L^0F(z,\overleftarrow{X}_z)|^2]}(\Delta t)^3, \\ \leq& C(\Delta t)^2, \end{aligned} \tag{3.3.11}$$

where the constant C depends on T, K and the upper bounds of derivatives of β, σ, c, φ, g and the jump intensity λ. From the definitions of $\mathcal{R}_u^{n+1}$, we easily get these following error estimates by using properties (3.3.3) and (3.3.4)

$$\begin{aligned} &\mathbb{E}[|X^{n+1}|^2] \leq C(1+\mathbb{E}[|X_T|^2]), \\ &\mathbb{E}[|\mathcal{R}_u^{n+1}|] \leq C(1+\mathbb{E}[|X^{n+1}|^{2r}])(\Delta t)^2 \leq C(1+\mathbb{E}[|X_T|^{2r}])(\Delta t)^2. \end{aligned} \tag{3.3.12}$$

□

Once we had Lemma 3.1, the convergence rate of the temporal discrete scheme follows from the Theorem 3.2.

Theorem 3.2. *Assume* $u(t_{n+1}, X^{n+1})$ *be the exact solution satisfying equation* (3.2.7) *and* $u_{n+1}(X^{n+1})$ *be the solution of the temporal discrete scheme* (3.2.9)*, respectively, where* $n =$

$0, 2, \ldots, N-1$, *then we have*

$$\mathbb{E}[|u(t_{n+1}, X^{n+1}) - u_{n+1}(X^{n+1})|] \leq C\Delta t, \tag{3.3.13}$$

where C is a constant defined in Lemma 3.1.

Proof. Subtracting (3.2.9) from (3.2.7), we have

$$\begin{aligned} e_u^{n+1} =& u(t_{n+1}, X^{n+1}) - u_{n+1}(X^{n+1}) \\ =& \mathbb{E}_{t_{n+1}}^{X^{n+1}}[u(t_n, \overleftarrow{X}_{t_n}^{t_{n+1}, X^{n+1}}) - u_n(X^n)] + \Delta t e_g^{n+1} + R_{n+1} \\ =& \mathbb{E}_{t_{n+1}}^{X^{n+1}}[u(t_n, \overleftarrow{X}_{t_n}^{t_{n+1}, X^{n+1}}) - u(t_n, X^n) + u(t_n, X^n) - u_n(X^n)] \\ & + \Delta t e_g^{n+1} + R_{n+1} \\ =& \mathbb{E}_{t_{n+1}}^{X^{n+1}}[e_u^n] + \Delta t e_g^{n+1} + \mathcal{R}_u^{n+1} + R_{n+1}. \end{aligned} \tag{3.3.14}$$

Thus

$$|e_u^{n+1}| \leq |\mathbb{E}_{t_{n+1}}^{X^{n+1}}[e_u^n]| + L\Delta t|e_u^{n+1}| + |\mathcal{R}_u^{n+1}| + |R_{n+1}|, \tag{3.3.15}$$

where L is the Lipschitz constant of function g with respect of u. From Lemma 3.1 we have

$$\mathbb{E}[|R_{n+1}|] \leq C(\Delta t)^2, \quad \mathbb{E}[|\mathcal{R}_u^{n+1}|] \leq C(\Delta t)^2, \tag{3.3.16}$$

hence the estimate for $\mathbb{E}[|e_u^n|]$ can be written as:

$$\mathbb{E}[|e_u^{n+1}|] \leq \mathbb{E}[|e_u^n|] + L\Delta t\mathbb{E}[|e_u^{n+1}|] + C(\Delta t)^2, \tag{3.3.17}$$

which is equivalent to

$$\mathbb{E}[|e_u^{n+1}|] \leq \frac{1}{1 - L\Delta t}\mathbb{E}[|e_u^n|] + \frac{C(\Delta t)^2}{1 - L\Delta t}. \tag{3.3.18}$$

We obtain, by induction, an upper bound of $\mathbb{E}[|e_u^{n+1}|]$, i.e.,

$$\mathbb{E}[|e_u^{n+1}|] \leq C\Delta t. \tag{3.3.19}$$

□

3.3.2 Error estimate of the temporal-spatial discrete scheme

In this section, we focus on analyzing the error estimate for the temporal-spatial discrete scheme (3.2.15). In order to avoid the Lebesgue constant of the pth order interpolant in error analysis under L^∞ that causes globally error blows up, the relative errors are considered under a designated norm that associate with transition probability density function of the backward process $\overleftarrow{X}_t$. To control the error coming from the quadrature rule involved in $\hat{\mathbb{E}}^{x_j}_{t_{n+1},M_u}[\cdot]$, we give a generalized Assumption 3.3, which can be specified according to the quadrature rule used in the numerical scheme (3.2.15).

Assumption 3.3. *For $m = 1, \ldots, M_u$, the error of the quadrature rule used in* (3.2.14) *is controlled by*

$$\begin{aligned}
&\left| \int_{\mathbb{R}} \int_E \cdots \int_E u^n \left(x_j + \beta(t_{n+1}, x_j)\Delta t + \sigma(t_{n+1}, x_j)\sqrt{\Delta t}\xi + \sum_{k=1}^{m} c(t_{n+1}, e_k) \right) \right. \\
&\varrho(\xi) \prod_{k=1}^{m} \rho(e_k) de_1 \cdots de_m d\xi - \sum_{i=1}^{L} \sum_{q=1}^{Q_m} w_j \tilde{w}_q^m u^n \left(x_j + \beta(t_{n+1}, x_j)\Delta t \right. \\
&\left. \left. + \sigma(t_{n+1}, x_j)\sqrt{2\Delta t} a_i + \sum_{k=1}^{m} c(t_{n+1}, \tilde{a}_q^{k,m}) \right) \right| \leq CQ^{-r},
\end{aligned} \tag{3.3.20}$$

where $r \geq 0$ is the convergence rate, C is a constant determined by the smoothness of u^n and kernel ρ.

For the case m=0, i.e., only Brownian motion is involved in $\hat{\mathbb{E}}^{x_j}_{t_{n+1},M_u}[\cdot]$, the quadrature error is bounded by [39]

$$\begin{aligned}
&\left| \int_{\mathbb{R}^d} u^n \big(x_j + \beta(t_{n+1}, x_j)\Delta t + \sigma(t_{n+1}, x_j)\sqrt{\Delta t}\xi \big) \varrho(\xi) d\xi \right. \\
&\left. - \sum_{i=1}^{L} w_j u^n \Big(x_j + \beta(t_{n+1}, x_j)\Delta t + \sigma(t_{n+1}, x_j)\sqrt{2\Delta t} a_i \Big) \right| \leq C \frac{L!}{2^L (2L)!} (\Delta t)^L,
\end{aligned} \tag{3.3.21}$$

where the constant C is independent with L and Δt and 3-points Guassian-Hermite quadrature rule can provide the second order convergence of local truncation.

Now, we introduce the designate norm that we use to measure the temporal-spatial approximation error, i.e.,

$$||v(t_{n+1},\cdot)|| := \sup_{\mu\in[a_0,b_0]} \int_{\mathbb{R}} |v(t_{n+1},x)|\overleftarrow{\rho}_{t_{n+1}|T}(x|\mu)dx, \tag{3.3.22}$$

where $[a_0, b_0]$ is any bounded sub-domain in $\mathbb{R}$, and $\overleftarrow{\rho}_{t_{n+1}|t_T}(\cdot|\mu)$ is the transition probability density function of *backward* process $\overleftarrow{X}_t$ in (3.2.3) under the condition that $X_T = \mu \in [a_0, b_0]$. In this effort, function is only measured in the bounded domain $[a_0, b_0]$ even variable x is defined in $\mathbb{R}$. Note that, due to the Markovian property of $X_T = \mu \in [a_0, b_0]$, we have

$$\overleftarrow{\rho}_{t_n|t_{n+1}}(\zeta|x)\overleftarrow{\rho}_{t_{n+1}|T}(x|\mu) = \overleftarrow{\rho}_{t_n,t_{n+1}|T}(\zeta,x|\mu). \tag{3.3.23}$$

The probabilistic representation of the exact value $u(t_{n+1}, x)$ of (3.1.1) at the time-space point $(t_{n+1}, x) \in [0, T] \times \mathbb{R}^d$, from (3.2.7), is given as

$$u(t_{n+1},x) = \mathbb{E}^x_{t_{n+1}}[u(t_n, \overleftarrow{X}^{t_{n+1},x}_{t_n})] + \Delta t g(t_{n+1}, x, u(t_{n+1}, x)) + \mathcal{O}(\Delta t)^2, \tag{3.3.24}$$

Subtracting (3.2.15) from (3.3.24), and denoting $e^{n+1}(x) = u(t_{n+1}, x) - u^{n+1}(x)$, we have

$$\begin{aligned}
e^{n+1}(x) =& \mathbb{E}^x_{t_{n+1}}[u(t_n, \overleftarrow{X}^{t_{n+1},x}_{t_n})] + \Delta t g(t_{n+1}, x, u(t_{n+1}, x)) + \mathcal{O}(\Delta t)^2 \\
&- \sum_{i=1}^{p+1} \left\{ \hat{\mathbb{E}}^{x_{j_i}}_{t_{n+1},M_u}[u^{n,p}] + \Delta t g(t_{n+1}, x_{j_i}, u^{n+1,p}_{j_i}) \right\} \psi_{j_i}(x) \\
=& \left\{ \mathbb{E}^x_{t_{n+1}}[u(t_n, \overleftarrow{X}^{t_{n+1},x}_{t_n})] - \sum_{i=1}^{p+1} \hat{\mathbb{E}}^{x_{j_i}}_{t_{n+1},M_u}[u^{n,p}]\psi_{j_i}(x) \right\} \\
&+ \Delta t \left\{ g(t_{n+1}, x, u(t_{n+1}, x)) - \sum_{i=1}^{p+1} g(t_{n+1}, x_{j_i}, u^{n+1,p}_{j_i})\psi_{j_i}(x) \right\} \\
&+ \mathcal{O}(\Delta t)^2 \\
=& e^{n+1}_{\mathrm{I}}(x) + \Delta t\, e^{n+1}_{\mathrm{II}}(x) + \mathcal{O}(\Delta t)^2.
\end{aligned} \tag{3.3.25}$$

In order to estimate $e_{\mathrm{I}}^{n+1}(x)$, we decompose it as follows,

$$
\begin{aligned}
e_{\mathrm{I}}^{n+1}(x) &= \mathbb{E}_{t_{n+1}}^{x}[u(t_n, \overleftarrow{X}_{t_n}^{t_{n+1},x})] - \sum_{i=1}^{p+1} \hat{\mathbb{E}}_{t_{n+1},M_u}^{x_{j_i}}[u^{n,p}]\psi_{j_i}(x) \\
&= \underbrace{\mathbb{E}_{t_{n+1}}^{x}[u(t_n, \overleftarrow{X}_{t_n}^{t_{n+1},x})] - \mathbb{E}_{t_{n+1}}^{x}[u(t_n, X^n)]}_{e_{\mathrm{I.1}}^{n+1}} + \underbrace{\mathbb{E}_{t_{n+1}}^{x}[u(t_n, X^n)] - \mathbb{E}_{t_{n+1}}^{x}[u^n(X^n)]}_{e_{\mathrm{I.2}}^{n+1}} \\
&\quad + \underbrace{\mathbb{E}_{t_{n+1}}^{x}[u^n(X^n)] - \mathbb{E}_{t_{n+1},M_u}^{x}[u^n(X^n)]}_{e_{\mathrm{I.3}}^{n+1}} + \underbrace{\mathbb{E}_{t_{n+1},M_u}^{x}[u^n(X^n)] - \hat{\mathbb{E}}_{t_{n+1},M_u}^{x}[u^n(X^n)]}_{e_{\mathrm{I.4}}^{n+1}} \\
&\quad + \underbrace{\hat{\mathbb{E}}_{t_{n+1},M_u}^{x}[u^n(X^n)] - \sum_{i=1}^{p+1} \hat{\mathbb{E}}_{t_{n+1},M_u}^{x_{j_i}}[u^{n,p}]\psi_{j_i}(x)}_{e_{\mathrm{I.5}}^{n+1}},
\end{aligned}
$$

where X^n is the the explicit Euler scheme for *backward* process $\overleftarrow{X}_{t_n}^{t_{n+1},x}$ given in (3.2.6). Similarly, we decompose $e_{\mathrm{II}}^{n+1}(x)$ as

$$
\begin{aligned}
e_{\mathrm{II}}^{n+1}(x) =& g(t_{n+1}, x, u(t_{n+1}, x)) - \sum_{i=1}^{p+1} g(t_{n+1}, x_{j_i}, u_{j_i}^{n+1,p})\psi_{j_i}(x) \\
=& \underbrace{g(t_{n+1}, x, u(t_{n+1}, x)) - g(t_{n+1}, x, u^{n+1}(x))}_{e_{\mathrm{II.1}}^{n+1}} \\
&+ \underbrace{g(t_{n+1}, x, u^{n+1}(x)) - \sum_{i=1}^{p+1} g(t_{n+1}, x_{j_i}, u_{j_i}^{n+1,p})\psi_{j_i}(x)}_{e_{\mathrm{II.2}}^{n+1}}.
\end{aligned}
$$

We apply the norm defined in (3.3.22) on both sides of the error in (3.3.25) to botain

$$
\begin{aligned}
\|e^{n+1}(x)\| =& \|e_{\mathrm{I.1}}^{n+1}(x) + e_{\mathrm{I.2}}^{n+1}(x) + e_{\mathrm{I.3}}^{n+1}(x) + e_{\mathrm{I.4}}^{n+1}(x) + e_{\mathrm{I.5}}^{n+1}(x) + e_{\mathrm{II.1}}^{n+1}(x) \\
&+ e_{\mathrm{II.2}}^{n+1}(x) + \mathcal{O}(\Delta t)^2\| \\
\leq& \|e_{\mathrm{I.1}}^{n+1}(x)\| + \|e_{\mathrm{I.2}}^{n+1}(x)\| + \|e_{\mathrm{I.3}}^{n+1}(x)\| + \|e_{\mathrm{I.4}}^{n+1}(x)\| + \|e_{\mathrm{I.5}}^{n+1}(x)\| \\
&+ \|e_{\mathrm{II.1}}^{n+1}(x)\| + \|e_{\mathrm{II.2}}^{n+1}(x)\| + \mathcal{O}(\Delta t)^2.
\end{aligned}
\tag{3.3.26}
$$

The errors on the right side of the above inequality come from different numerical approximations, which we analyze individually. For the error $e_{\text{I.1}}^{n+1}$, we have

$$\begin{aligned}
\|e_{\text{I.1}}^{n+1}\| =& \|\mathbb{E}_{t_{n+1}}^{x}[u(t_n, \overleftarrow{X}_{t_n}^{t_{n+1},x})] - \mathbb{E}_{t_{n+1}}^{x}[u(t_n, X^n)]\| \\
=& \sup_{\mu\in[a_0,b_0]} \int_{\mathbb{R}} \left|\mathbb{E}_{t_{n+1}}^{x}[u(t_n, \overleftarrow{X}_{t_n}^{t_{n+1},x})] - \mathbb{E}_{t_{n+1}}^{x}[u(t_n, X^n)]\right| \overleftarrow{\rho}_{t_{n+1}|T}(x|\mu)dx \\
\leq& \sup_{\mu\in[a_0,b_0]} \int_{\mathbb{R}} C(1+|x|^{2r})(\Delta t)^2 \overleftarrow{\rho}_{t_{n+1}|T}(x|\mu)dx \\
=& C(\Delta t)^2 \sup_{\mu\in[a_0,b_0]} \mathbb{E}_T^{\mu}[1+|x|^{2r}] \\
\leq& C(\Delta t)^2 \sup_{\mu\in[a_0,b_0]} (1+|\mu|^{2r}),
\end{aligned} \tag{3.3.27}$$

where the first inequality is coming from the result of the error estimate of residual $\mathcal{R}_u^{n+1}$ in Lemma 3.1 and the last inequality comes from equation (3.3.5). For $e_{\text{I.2}}^{n+1}$, we have

$$\begin{aligned}
\|e_{\text{I.2}}^{n+1}\| =& \|\mathbb{E}_{t_{n+1}}^{x}[u(t_n, X^n)] - \mathbb{E}_{t_{n+1}}^{x}[u^n(X^n)]\| \\
=& \sup_{\mu\in[a_0,b_0]} \int_{\mathbb{R}} \left|\mathbb{E}_{t_{n+1}}^{x}[u(t_n, X_x^n)] - \mathbb{E}_{t_{n+1}}^{x}[u^n(X_x^n)]\right| \overleftarrow{\rho}_{t_{n+1}|T}(x|\mu)dx \\
=& \sup_{\mu\in[a_0,b_0]} \int_{\mathbb{R}} \left| \int_{\mathbb{R}} [u(t_n,\zeta) - u^n(\zeta)] \overleftarrow{\rho}_{n|n+1}(\zeta|x)d\zeta \right| \overleftarrow{\rho}_{t_{n+1}|T}(x|\mu)dx \\
=& \sup_{\mu\in[a_0,b_0]} \int_{\mathbb{R}} \left| \int_{\mathbb{R}} [u(t_n,\zeta) - u^n(\zeta)] \big(\overleftarrow{\rho}_{n|n+1}(\zeta|x) - \overleftarrow{\rho}_{t_n|t_{n+1}}(\zeta|x) \right. \\
& \left. + \overleftarrow{\rho}_{t_n|t_{n+1}}(\zeta|x)\big)d\zeta \right| \overleftarrow{\rho}_{t_{n+1}|T}(x|\mu)dx \\
\leq& \sup_{\mu\in[a_0,b_0]} \int_{\mathbb{R}} \left| \int_{\mathbb{R}} [u(t_n,\zeta) - u^n(\zeta)] \left[\overleftarrow{\rho}_{n|n+1}(\zeta|x) - \overleftarrow{\rho}_{t_n|t_{n+1}}(\zeta|x)\right] d\zeta \right| \overleftarrow{\rho}_{t_{n+1}|T}(x|\mu)dx \\
& + \sup_{\mu\in[a_0,b_0]} \int_{\mathbb{R}} \left| \int_{\mathbb{R}} [u(t_n,\zeta) - u^n(\zeta)] \overleftarrow{\rho}_{t_n|t_{n+1}}(\zeta|x) \overleftarrow{\rho}_{t_{n+1}|T}(x|\mu)d\zeta \right| dx \\
\leq& \sup_{\mu\in[a_0,b_0]} \int_{\mathbb{R}} \left| \int_{\mathbb{R}} [u(t_n,\zeta) - u^n(\zeta)] \left[\overleftarrow{\rho}_{n|n+1}(\zeta|x) - \overleftarrow{\rho}_{t_n|t_{n+1}}(\zeta|x)\right] d\zeta \right| \overleftarrow{\rho}_{t_{n+1}|T}(x|\mu)dx \\
& + \sup_{\mu\in[a_0,b_0]} \int_{\mathbb{R}} \int_{\mathbb{R}} |u(t_n,\zeta) - u^n(\zeta)| \overleftarrow{\rho}_{t_n,t_{n+1}|T}(\zeta,x|\mu)d\zeta dx \\
=& \sup_{\mu\in[a_0,b_0]} \int_{\mathbb{R}} \left| \int_{\mathbb{R}} [u(t_n,\zeta) - u^n(\zeta)][\overleftarrow{\rho}_{n|n+1}(\zeta|x) - \rho_{t_n|t_{n+1}}(\zeta|x)]d\zeta \right| \overleftarrow{\rho}_{t_{n+1}|T}(x|\mu)dx \\
& + ||e^n||,
\end{aligned} \tag{3.3.28}$$

where $\overleftarrow{\rho}_{n|n+1}$ is the transition probability density function of discrete *backward* process X^n defined in (3.2.6). To approximate the first term on the right side, we first explicitly express the Approximation error property of the explicit Euler scheme (3.3.4) as that there exist positive real number r such that for any function $g \in \mathcal{C}_P^4$, we have

$$\begin{aligned} |\mathbb{E}_{t_{n+1}}^x[g(\overleftarrow{X}_{t_n}^{t_{n+1},x}) - g(X^n)]| &= \left| \int_{\mathbb{R}} g(\zeta)\overleftarrow{\rho}_{t_n|t_{n+1}}(\zeta|x)d\zeta - \int_{\mathbb{R}} g(\zeta)\overleftarrow{\rho}_{n|n+1}(\zeta|x)d\zeta \right| \\ &= \left| \int_{\mathbb{R}} g(\zeta)\left[\overleftarrow{\rho}_{t_n|t_{n+1}}(\zeta|x) - \overleftarrow{\rho}_{n|n+1}(\zeta|x)\right] d\zeta \right| \\ &\leq C(1+|x|^{2r})(\Delta t)^2. \end{aligned} \tag{3.3.29}$$

According to the previous assumptions of coefficients β, σ, c and forcing g, it is easy to prove $u(t,x) - u^n(x) \in \mathcal{C}_P^4$. Then based on the equation (3.3.30), we have

$$\left| \int_{\mathbb{R}} u(t_n,\zeta) - u^n(\zeta)][\overleftarrow{\rho}_{t_n|t_{n+1}}(\zeta|x) - \overleftarrow{\rho}_{n|n+1}(\zeta|x)]d\zeta \right| \leq C(1+|x|^{2r})(\Delta t)^2. \tag{3.3.30}$$

Plugging equation (3.3.30) into equation (3.3.28), the truncation error $\|e_{\text{I.2}}^{n+1}\|$ can be estimated by

$$\begin{aligned} \|e_{\text{I.2}}^{n+1}\| \leq& \|e^n\| + \sup_{\mu\in[a_0,b_0]} \int_{\mathbb{R}} C(1+|x|^{2r})(\Delta t)^2 \overleftarrow{\rho}_{t_{n+1}|T}(x|\mu)dx \\ \leq& \|e^n\| + C(\Delta t)^2 \sup_{\mu\in[a_0,b_0]} \mathbb{E}_T^{\mu}[1+|x|^{2r}] \\ \leq& \|e^n\| + C(\Delta t)^2 \sup_{\mu\in[a_0,b_0]} (1+|\mu|^{2r}). \end{aligned} \tag{3.3.31}$$

Also, for the error $e_{\mathrm{I}.3}^{n+1}$, we deduce that

$$\begin{aligned}
\|e_{\mathrm{I}.3}^{n+1}\| =&\|\mathbb{E}_{t_{n+1}}^{x}[u^n(X^n)] - \mathbb{E}_{t_{n+1},M_u}^{x}[u^n(X^n)]\| \\
=& \sup_{\mu\in[a_0,b_0]} \int_{\mathbb{R}} \left|\mathbb{E}_{t_{n+1}}^{x}[u^n(X^n)] - \mathbb{E}_{t_{n+1},M_u}^{x}[u^n(X^n)]\right| \overleftarrow{\rho}_{t_{n+1}|T}(x|\mu)dx \\
=& \sup_{\mu\in[a_0,b_0]} \int_{\mathbb{R}} \Bigg| \sum_{m=M_u+1}^{\infty} e^{-\lambda\Delta t}\frac{(\lambda\Delta t)^m}{m!} \int_{\mathbb{R}}\int_{E}\cdots\int_{E} u^n\Bigg(x+\beta(t_{n+1},x)\Delta t \\
&+\sigma(t_{n+1},x)\sqrt{\Delta t}\xi + \sum_{k=1}^{m} c(t_{n+1},e_k)\Bigg)\varrho(\xi)\prod_{k=1}^{m}\rho(e_k)de_1\cdots de_m d\xi\Bigg| \\
&\cdot \overleftarrow{\rho}_{t_{n+1}|T}(x|\mu)dx \\
\leq& C(\lambda\Delta t)^{M_u+1},
\end{aligned} \tag{3.3.32}$$

where the above inequality is due to the Assumption 3.3 and constant C depends on the upper bounds of u^n. For the error $e_{\mathrm{I}.4}^{n+1}$, we have

$$\begin{aligned}
\|e_{\mathrm{I}.4}^{n+1}\| =&\|\mathbb{E}_{t_{n+1},M_u}^{x}[u^n(X^n)] - \hat{\mathbb{E}}_{t_{n+1},M_u}^{x}[u^n(X^n)]\| \\
=&\Bigg\| e^{-\lambda\Delta t}\Bigg[\int_{\mathbb{R}} u^n\Bigg(x+\beta(t_{n+1},x)\Delta t+\sigma(t_{n+1},x)\sqrt{\Delta t}\xi\Bigg)\varrho(\xi)d\xi - \sum_{i=1}^{K} w_i u^n\Bigg(x \\
&+\beta(t_{n+1},x)\Delta t+\sigma(t_{n+1},x)\sqrt{\Delta t}a_i\Bigg)\Bigg]\Bigg\| + \Bigg\| \sum_{m=1}^{M_u} e^{-\lambda\Delta t}\frac{(\lambda\Delta t)^m}{m!}\Bigg[\int_{\mathbb{R}}\int_{E}\cdots \\
&\int_{E} u^n\Bigg(x+\beta(t_{n+1},x)\Delta t+\sigma(t_{n+1},x)\sqrt{\Delta t}\xi+\sum_{k=1}^{m}c(t_{n+1},e_k)\Bigg)\varrho(\xi)\prod_{k=1}^{m}\rho(e_k) \\
&de_1\cdots de_m d\xi - \sum_{i=1}^{K}\sum_{q=1}^{Q_m} w_i\tilde{w}_q^m u^n\Bigg(x+\beta(t_{n+1},x)\Delta t+\sigma(t_{n+1},x)\sqrt{\Delta t}a_i \\
&+\sum_{k=1}^{m} c(t_{n+1},\tilde{a}_q^{k,m})\Bigg)\Bigg]\Bigg\| \\
:=&A+B.
\end{aligned} \tag{3.3.33}$$

The first term A can be controlled by

$$\begin{aligned} A =& \left\| e^{-\lambda\Delta t}\left[\int_{\mathbb{R}} u^n\left(x+\beta(t_{n+1},x)\Delta t+\sigma(t_{n+1},x)\sqrt{\Delta t}\xi\right)\varrho(\xi)d\xi\right.\right. \\ & \left.\left. -\sum_{i=1}^{K} w_j u^n\left(x+\beta(t_{n+1},x)\Delta t+\sigma(t_{n+1},x)\sqrt{\Delta t}a_i\right)\right]\right\| \\ \leq & ||e^{-\lambda\Delta t}C(\Delta t)^K|| \leq C(\Delta t)^K. \end{aligned} \tag{3.3.34}$$

Under the Assumption 3.3, the second term B can be controlled by

$$B \leq \sum_{m=1}^{M_u} e^{-\lambda\Delta t}\frac{(\lambda\Delta t)^m}{m!}(CQ^{-r}) \leq C\lambda\Delta t Q^{-r}, \tag{3.3.35}$$

thus,

$$\|e_{\mathrm{I.4}}^{n+1}\| \leq C(\Delta t)^K + C\lambda\Delta t Q^{-r}. \tag{3.3.36}$$

For $e_{\mathrm{I.5}}^{n+1}$, based on the classic error bound of pth Lagrange interpolation, we have

$$||e_{\mathrm{I.5}}^{n+1}|| = \left\| \hat{\mathbb{E}}^{x}_{t_{n+1},M_u}[u^n(X^n)] - \sum_{i=1}^{p+1} \hat{\mathbb{E}}^{x_{j_i}}_{t_{n+1},M_u}[u^{n,p}]\psi_{j_i}(x) \right\| \leq C(\Delta x)^{p+1}. \tag{3.3.37}$$

For $e_{\mathrm{II.1}}^{n+1}$, due to the Lipschitz continuity of forcing g with respect to u, we obatin

$$\begin{aligned} \|e_{\mathrm{II.1}}^{n+1}\| &= \|g(t_{n+1},x,u(t_{n+1},x)) - g(t_{n+1},x,u^{n+1}(x))\| \\ &= \sup_{\mu\in[a_0,b_0]}\int_{\mathbb{R}} |g(t_{n+1},x,u(t_{n+1},x)) - g(t,x,u^{n+1}(x))|\overleftarrow{\rho}_{t_{n+1}|T}(x|\mu)dx \\ &\leq L \sup_{\mu\in[a_0,b_0]}\int_{\mathbb{R}} |u(t_{n+1},x) - u^{n+1}(x)|\overleftarrow{\rho}_{t_{n+1}|T}(x|\mu)dx \\ &= L\|e^{n+1}\|, \end{aligned} \tag{3.3.38}$$

where L is the Lipschitz constant and for $e_{\text{II.2}}^{n+1}$,

$$\begin{aligned}\|e_{\text{II.2}}^{n+1}\| =& \left\| g(t_{n+1}, x, u^{n+1}(x)) - \sum_{i=1}^{p+1} g(t_{n+1}, x_{j_i}, u_{j_i}^{n+1,p})\psi_{j_i}(x) \right\| \\ =& \sup_{\mu\in[a_0,b_0]} \int_{\mathbb{R}} \left| g(t_{n+1}, x, u^{n+1}(x)) - \sum_{i=1}^{p+1} g(t_{n+1}, x_{j_i}, u_{j_i}^{n+1,p})\psi_{j_i}(x) \right| \\ & \cdot \overleftarrow{\rho}_{t_{n+1}|T}(x|\mu)dx \\ \leq& C(\Delta x)^{p+1}.\end{aligned} \tag{3.3.39}$$

Putting all estimates together, we get

$$\begin{aligned}\|e^{n+1}\| \leq& \|e^n\| + C(\Delta t)^2 \sup_{\mu\in[a_0,b_0]} (1+|\mu|^{2r}) + C(\lambda\Delta t)^{M_u+1} + C(\Delta t)^K \\ & + C\lambda\Delta t Q^{-r} + C(\Delta x)^{p+1} + \Delta t\left(L\|e^{n+1}\| + C(\Delta x)^{p+1}\right).\end{aligned} \tag{3.3.40}$$

Using induction technique and for sufficiently small Δt, we have

$$\|e^{n+1}\| \leq C\left[\Delta t + (\lambda\Delta t)^{M_u} + (\Delta t)^{K-1} + \lambda Q^{-r} + \frac{(\Delta x)^{p+1}}{\Delta t}\right]. \tag{3.3.41}$$

Summarize the above inequalities, we get the following theorem for the error estimate of temporal-spatial scheme.

Theorem 3.4. *let $u(t_{n+1}, x)$ be the exact solution of the PIDEs* (3.1.1) *at (t_{n+1}, x) and u^{n+1} be the solution of temporal-spatial discrete scheme* (3.2.15)*. Under Assumption 3.3, the error $e_{n+1}(x) = u(t_{n+1}, x) - u^{n+1}$, where $n = 0, 1, \ldots, N-1$, can be estimated by*

$$||e^n|| \leq C\left[\Delta t + (\lambda\Delta t)^{M_u} + (\Delta t)^{K-1} + \lambda Q^{-r} + \frac{(\Delta x)^{p+1}}{\Delta t}\right], \tag{3.3.42}$$

where the constant C is depending on the X_T, the jump intensity λ, the upper bounds of derivatives of g, β, σ and φ.

3.4 Numerical examples

In this section, we perform one three-dimensional synthetic example and one two-dimensional example in the groundwater flow area to verify the theoretical analysis. Specifically, Example 1 with general stochastic parameters b, σ, c demonstrates the accuracy and effectiveness of the temporal-spatial discrete scheme (3.2.15) for handling PIDE (3.1.1), we denote Δt and Δx as the temporal and spatial step sizes, respectively. Based on uniform partitions in both temporal and spatial domains and we test the convergence rate of our scheme (3.2.15) with respect to Δt and Δx. To get the second-order convergence rate with respect to Δx, we construct the cubic interpolation based on Delaunay Triangulation. In this case, we are facing a large amount of six-dimensional integration problems. The sparse grid quadrature rule can be used to decrease the numbers of quadrature points. In the example 2, we set up a $2D$ groundwater flow model and do simulate the evolution of hydraulic head instead of solving the relative error convergence, where the dynamic is driven by Brownian motion and Poisson jump process, separately.

Example 1. We consider the following three-dimensional nonlocal PIDE

$$\begin{cases} \dfrac{\partial u}{\partial t}(t,x) - \mathcal{L}[u](t,x) = f(t,x,u), \forall (t,x) \in [0,T) \times \mathbb{R}^3, \\ u(0,x) = \varphi(x), \forall x \in \mathbb{R}^3. \end{cases} \tag{3.4.1}$$

The operator $\mathcal{L}$ is defined in (3.1.2) and parameters are given as:

- the drift coefficient: $b(t,x) = x$,
- the jump height function: $c(t,e) = e$,
- the symmetric kernel: $\gamma(e) = 10^3 \times 1_{|e| \leq 0.1}$,
- the diffusion coefficient:

$$\sigma(t,x) = \begin{bmatrix} sin(x_1) & 0 & 0 \\ 0 & cos(x_3) & 0 \\ 0 & 0 & sin(x_2) \end{bmatrix}.$$

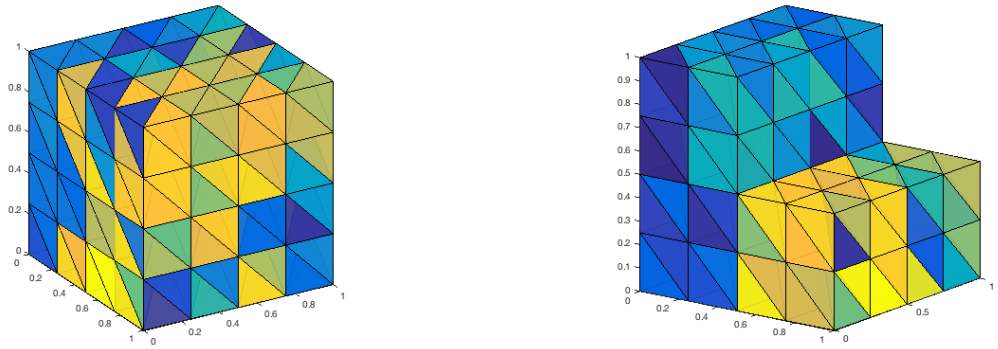

Figure 3.2: Spatial domains

We choose the exact solution to be

$$u(t,x) = \sin\left(\frac{t}{2}\right)x_1^4 + x_2x_3,$$

so that the forcing term f is given by

$$\begin{aligned} f(t,x) = &\left(\sin^2(x_1) - \cos^2(x_1) - 3\right)u(t,x) + \frac{x_1^4}{2}\cos\left(\frac{t}{2}\right) - 2x_2x_3 \\ &+ \left(4x_1^4 - 4x_1^3\sin(2x_1) - 6x_1^2\sin^2(x_1) - \frac{2\pi x_1^2}{125} - \frac{4\pi}{350000}\right)\sin\left(\frac{t}{2}\right) \end{aligned} \tag{3.4.2}$$

and the initial condition $\varphi(x)$ is equal to $u(0,x)$. We solve the PIDE (3.4.1) in two different domains, i.e., the first one is a cube $[0,1]^3$, the second one is an incomplete cube, domains are shown in Figure 3.2. To test the convergence rate with respect to Δt, we set $\Delta x = \frac{1}{64}$ and use linear interpolation with Delaunay Triangulation, which is enough to guarantee the first order convergence. Since the symmetric kernel $\gamma(e)$ is uniform with the support $|e| \leq 0.1$ and Brownian motion is also involved in the conditional expectation $\mathbb{E}_{t_{n+1}}^{x_j}[\cdot]$, we use sparse grids based on Gauss-Legendre quadrature rule and Gauss-Hermite quadrature rule with enough high level such that the truncation error of interpolation can be ignored, we set $M_u = 1$, i.e., one jump in conditional expectation $\hat{\mathbb{E}}_{t_{n+1},M_u}^{x_j}[u^n]$ and divide the time interval $[0,1]$ into N_t with $\Delta t = 2^{-2}, 2^{-3}, 2^{-4}, 2^{-5}, 2^{-6}$, the error is measured in two different measure norms L^∞ and L^2. Results are listed in table 3.1. As expected, the convergence rate with respect to Δt is first order in both spatial domains.

Table 3.1: Errors and convergence rates with respect to Δt in Example 1

Δt	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	CR
$\|u-u\|_{L^\infty}$						
cubic	4.6977e-01	2.0351e-01	8.2775e-02	3.2263e-02	1.4464e-02	1.2700
in-cubic	4.0173e-01	1.4239e-01	5.3069-02	2.5996e-02	1.2886e-02	1.2378
$\|u-u\|_{L^2}$						
cubic	1.6159e-01	7.5064e-02	3.4977e-02	1.6026e-02	7.2867e-03	1.1170
in-cubic	1.1603e-01	5.0939e-02	2.3613e-02	1.0961e-02	5.0264e-03	1.1274

Next, we test the convergence rate with respect to Δx, we observe in Theorem 3.4 that to test the convergence rate with respect to Δx, the quantity $\frac{1}{\Delta t}$ should be in a reasonable range, thus balancing Δt and Δx is needed here:

- To get the first order convergence rate with respect to Δx, we use linear interpolation and set $\Delta t = \Delta x$ with $T = 0.5$.
- To get the second order convergence rate with respect to Δx, we use cubic interpolation and set $\Delta t = (\Delta x)^2$ with $T = 0.25$.

To access the cubic interpolation on Delaunay Triangulation, we need to bulid the finite element approximation in each tetrahedron written as customary form, i.e.,

$$U(x_1, x_2, x_3) = \sum_{i=1}^{N} c_i u_i(x_1, x_2, x_3),$$

where N is the number of nodes, i.e., $N = 20$ for cubic interpolation, c_i is the shape function and $u_i(x_1, x_2, x_3)$ is the interpolant at ith node, $i = 1, 2, \ldots, N$ studied by [33, 55]. Compared to the linear interpolation on Delaunay triangulation, the cubic interpolation requires much more node points which aggravates the computational cost in high dimensional cases. Sparse grid interpolation [7, 37, 43, 45] could be used to alleviate this explosion that will be considered in our future works. To test the first order convergence with respect to Δx, we divide the spatial domain D into N_x with $\Delta x = 2^{-2}, 2^{-3}, 2^{-4}, 2^{-5}\ 2^{-6}$, and set $\Delta t = \Delta x$. To test the second order convergence with respect to Δx, we divide the spatial domain D into N_x with $\Delta x = 2^{-1}, 2^{-2}, 2^{-3}, 2^{-4}$ and set $\Delta t = (\Delta x)^2$. In Figure 3.3, we can see that the numerical results verify the theoretical analysis.

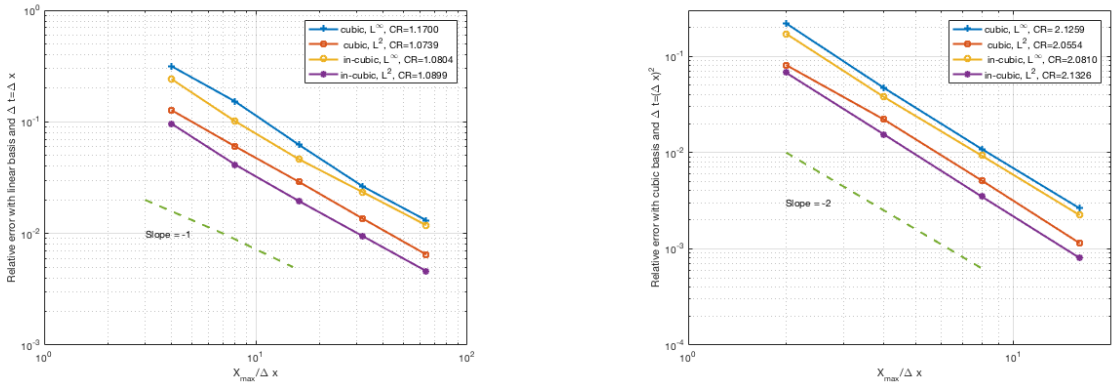

Figure 3.3: Errors and convergence rates with respect to Δx in Example 1

Example 2. We present the evolution of the 2D hydraulic head model given as

$$\begin{cases} \frac{\partial u}{\partial t}(t,x) - \mathcal{L}[u](t,x) = 0, & (t,x) \in [0,T] \times D, \\ u(t,x) = 10[(1+\frac{1}{2}\sin t)x_2 + 5x_1], & (t,x) \in (0,T] \times \{\partial D_{top} \cup \partial D_{bot} \cup \partial D_{right}\}, \\ u(t,x) = \sin t + 2 + 4x_1, & (t,x) \in (0,T] \times \partial D_{left}, \\ u(t,x) = 0, & (t,x) \in 0 \times \bar{D}, \end{cases} \tag{3.4.3}$$

where domain is given in Figure 7.3 in Appendix 7.5. And the partial integro-differential operator $\mathcal{L}$ is determined by the underlying process, here we consider the dynamic system is driven by Poisson jump process and Brownian motion process, respectively.

1. When the Lévy process does only consist of by Poisson jump process, based on the Feynman-Kac theory, the infinitesimal operator $\mathcal{L}$ is a nonlocal operator given as

$$\mathcal{L}[u] = \int_E \Big(u(t,x+e) - u(t,x)\Big)\gamma(e)de,$$

we set parameters as $T = 10\pi$, $\Delta t = \frac{\pi}{16}$ and the kernel is defined by

$$\gamma(e) = \begin{cases} 4, & |e| \le \delta, \\ 0, & otherwise. \end{cases} \tag{3.4.4}$$

The larger jump horizon δ within reasonable range leads to a faster evolution of hydraulic head u, such phenomenon is investigated by setting $\delta = 0.1$ and 0.25, the results are shown in Figure 3.4.

2. We consider the the Lévy process does only consist of by Brownian motion to compare with the nonlocal operator. In this case, $\mathcal{L}$ is the Laplace operator, i.e.,

$$\mathcal{L}[u] = \frac{1}{2}\sum_{i=1}^{2}\frac{\partial^2 u}{\partial x_i^2}(t,x),$$

since the heat equation allows an infinite speed of propagation, we consider this dynamic system in a small time range, i.e., $T = \frac{\pi}{16}$ with $\Delta t = \frac{\pi}{2048}$. The evolution of u is shown in Figure 3.4.

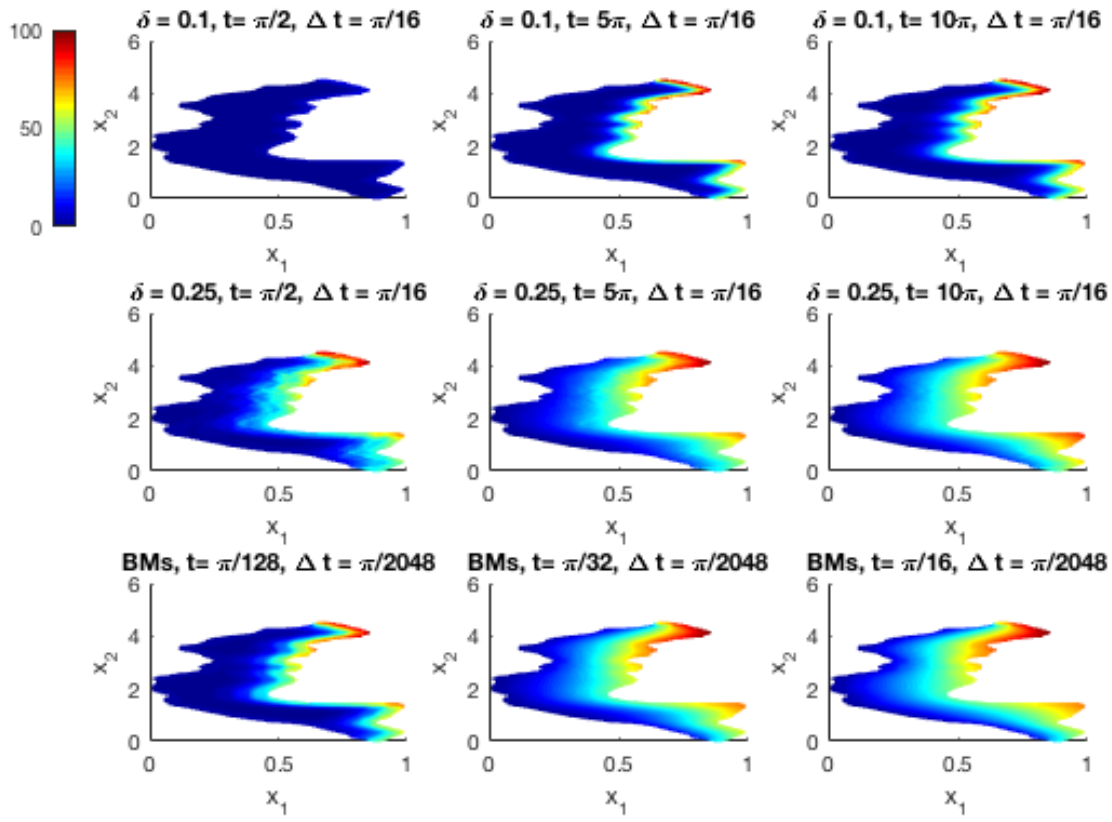

Figure 3.4: The evolution of hydraulic head $u(t,x)$ in Example 2

Chapter 4

Probabilistic schemes for the fractional Laplacian equation

4.1 Problem setting

We first describe the fractional Laplacian equation that is the focus of Chapter 4 and demonstrate that our probabilistic schemes in Chapter 3 provide a novel perspective to solve the fractional Laplacian diffusion equation based on the study by [4, 8]. To proceed, we use a one-dimensional numerical example to verify the Gaussian Approximation Theorem introduced by J. Rosiński and S. Cohen [8], i.e., the Gaussian approximation of Lévy processes can simulate a kind of stable processes. The second part is that we numerically test errors convergence rate of the proposed schemes for estimating $u(t,x)$ of the PIDE related to the Lévy process coming from Gaussian approximation. In particular, we obtain the convergence rates with respect to both Δt and Δx for two different α cases, i.e, $\alpha = 0.5$ and 1.5.

Consider the following PDE with fractional Laplacian operator:

$$\begin{cases} \dfrac{\partial v}{\partial t}(t,x) + (-\Delta)^{\alpha/2} v(t,x) = f(t,x), \forall (t,x) \in [0,T] \times \mathbb{R}^d, \\ v(0,x) = \varphi(x), \forall x \in \mathbb{R}^d, \end{cases} \tag{4.1.1}$$

where the fractional Laplacian operator is defined by

$$(-\Delta)^{\alpha/2} v(t,x) = C_{d,\alpha} \int_{\mathbb{R}^d} \frac{v(t,x) - v(t,y)}{|x-y|^{d+\alpha}} dy, \tag{4.1.2}$$

with $C_{d,\alpha}$ given as

$$C_{d,\alpha} = \frac{\alpha 2^{\alpha-1} \Gamma(\frac{\alpha+d}{2})}{\pi^{d/2} \Gamma(\frac{2-\alpha}{2})}.$$

According to the Lévy-Khintchine lemma [2], the stochastic process corresponding to fractional operator $(-\Delta)^{\alpha/2}$ is the symmetric α-stable process (SαS) $X = \{X_t, 0 \leq t \leq T\}$ with characteristic function being

$$\phi(t, w) = \mathbb{E}[e^{iwX_t}] = \exp\{-e|w|^{\alpha}\},$$

where $w \in \mathbb{R}^d$, $\alpha \in (0, 2]$, i is the imaginary unit and the Lévy measure ν is defined by

$$\nu(dz) = C_{d,\alpha} \frac{1}{|z|^{d+\alpha}} dz \quad \forall z \in \mathbb{R}^d.$$

Note that $(-\Delta)^{\alpha/2}$ is the infinitesimal generator of that SαS process for a fixed $\alpha \in (0, 2]$.

4.2 Gaussian approximation of Lévy processes

To approximate a Lévy process that has paths of infinite variation by an appropriate compound Poisson process leads to a significant error. S. Asmussen and J. Rosiński [4] proved that the remainder process could be approximated by one Brownian motion with a small variance. In other words, combining such a small variance Brownian motion with an appropriate Poisson process can improve the approximation accuracy. In high-dimensional space $\mathbb{R}^d$, we define a Lévy process $X_\epsilon := \{X_\epsilon(t) : t \geq 0\}$, for every $\epsilon \in (0, 1]$, with characteristic function of the form

$$\mathbb{E}[e^{i\langle y, X_\epsilon(t)\rangle}] = exp\left\{t \int_{\mathbb{R}^d} \left[e^{i\langle y,x\rangle} - 1 - i\langle y, x\rangle\right] \nu_\epsilon(dx)\right\}, \tag{4.2.1}$$

where the Poisson random measure satisfies

$$\int_{\mathbb{R}^d} \|x\|^2 \nu_\epsilon(dx) < \infty.$$

Then Lévy process $X_\epsilon(t)$ has zero mean and covariance $\mathbb{E}[X_\epsilon(t)X_\epsilon(t)^\top] = t\Sigma_\epsilon$, where $\Sigma_\epsilon = \int_{\mathbb{R}^d} x \cdot x^\top \nu_\epsilon(dx)$. Theorem 2.2 in [4] states that suppose Σ_ϵ is nonsingular for $\epsilon \in (0, 1]$, then as $\epsilon \to 0$, we have

$$\Sigma_\epsilon^{-1/2} X_\epsilon \xrightarrow{d} B \tag{4.2.2}$$

if and only if for every $\kappa > 0$

$$\int_{\langle \Sigma_\epsilon^{-1} x, x\rangle > \kappa} \langle \Sigma_\epsilon^{-1} x, x\rangle \nu_\epsilon(dx),$$

where B denotes a standard Brownian motion and " $\xrightarrow{d}$ " stands for the weak convergence of processes in the space $D([0,T), \mathbb{R}^d)$ of càdlàg functions with the Skorohod topology. Right now we can do the approximate simulation of multivariate $S\alpha S$ processes based on the above technology for approximating the remainder process X_ϵ.

We give a decomposition of Lévy measure ν as

$$\nu = \nu_\epsilon + \nu^\epsilon,$$

where $\nu^\epsilon(\mathbb{R}^d) < \infty$. The $S\alpha S$ process $X = \{X(t), 0 \le t \le T\}$ without drift term can be decomposed as

$$X \stackrel{d}{=} X_\epsilon + N^\epsilon = \nu_{\{\|x\| \le \epsilon\}} + \nu_{\{\|x\| \ge \epsilon\}}, \tag{4.2.3}$$

where $N^\epsilon = \{N^\epsilon(t), 0 \le t \le T\}$ is a compound Poisson process with the jumps measure ν^ϵ and since the equation (4.2.2), we have

$$X \xrightarrow{d} \Sigma_\epsilon^{1/2} B + N^\epsilon \tag{4.2.4}$$

Theorem 3.1 in [8] provides a theoretical basis of the approximation formula (4.2.4) and we numerically verify the accuracy of that theorem by two α cases, i.e., $\alpha = 1.5$ and 0.5 in one-dimensional space. We choose an appropriate parameter ϵ and results are shown in Figure 4.1. In each case, the first subplot is plotting the compound Poisson process N^ϵ. The second subplot is the Brownian motion $\Sigma_\epsilon^{1/2} B$. We compare the composite processes $N^\epsilon + \Sigma_\epsilon^{1/2} B$, in the third subplot, with the analytic probability density function of the $S\alpha S$ process X. In the third subplot, we see that, with an appropriate number ϵ, the composite processes $N^\epsilon + \Sigma_\epsilon^{1/2} B$ provides a reasonable approximation for the SαS process X.

The SαS process is the underling stochastic process of the fractional Laplacian equation

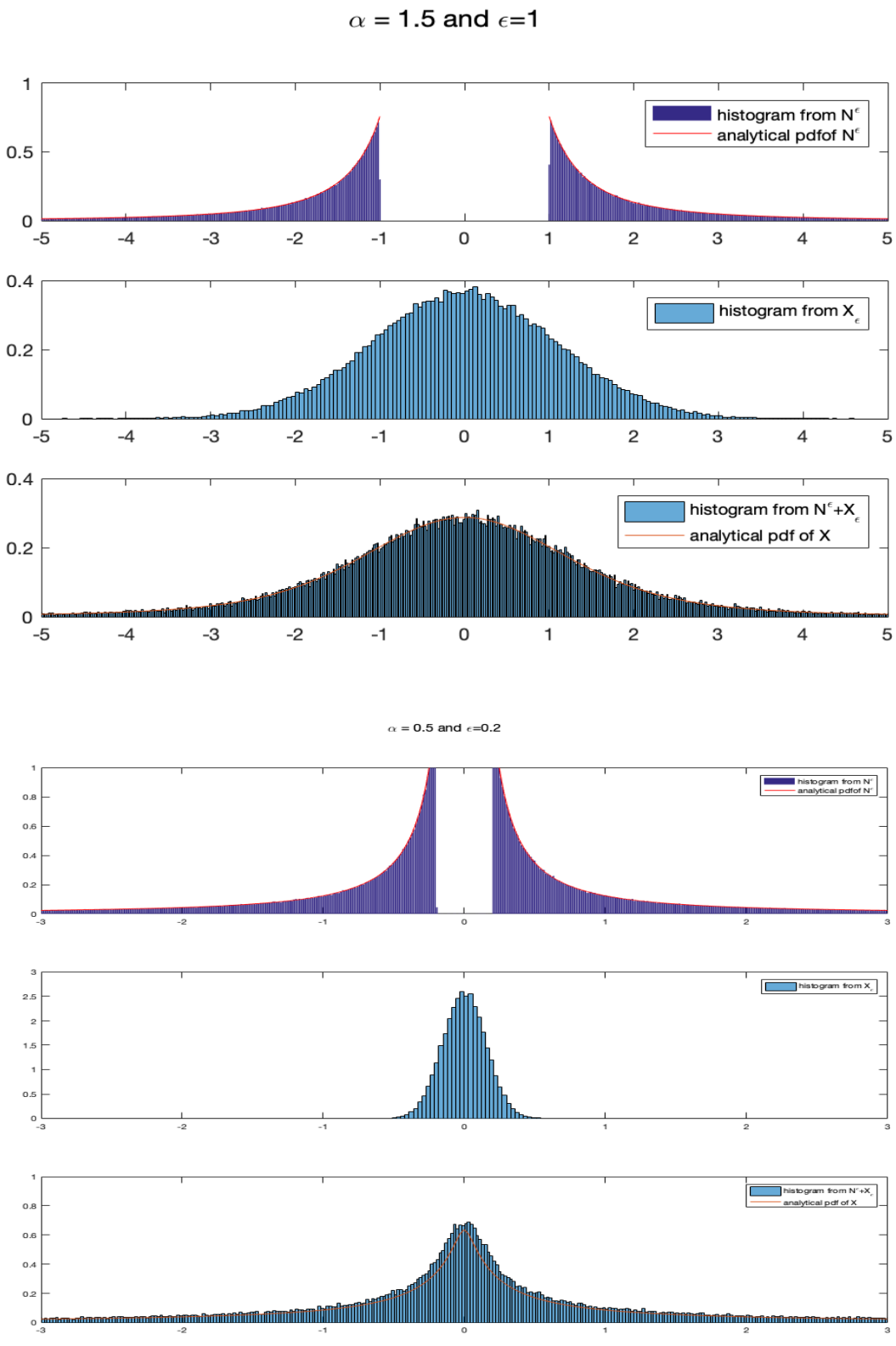

Figure 4.1: Gaussian approximation of Lévy process

and the Gaussian approximation-Lévy process is associated with PIDE introduced in Chapter 3. In the stochastic perspective, the Gaussian approximation gives us a new approach to approximate the solution $v(t,x)$ in fractional Laplacian equation (4.1.1) by a solution $u(t,x)$ of PIDE with second order integro-differential operator. Here we focus on constructing the numerical schemes for the solution $u(t,x)$. The theoretical analysis for estimating the relative error between $u(t,x)$ and $v(t,x)$ refers to literature [4, 8]. The underlying corresponding to (4.1.1) is given as

$$X_t^\alpha = X_0^\alpha + \int_0^t \int_E \tilde{\nu}(de, ds), \tag{4.2.5}$$

where ν is the random measure of the SαS process, superscript $0 < \alpha \leq 2$ is the fraction Laplacian order, then X_t^α can be approximated by

$$X_t^\alpha = X_0^\alpha + \int_0^t \Sigma_\epsilon^{1/2} dB_t + \int_0^t \int_E e\tilde{\nu}^\epsilon(de, ds), \tag{4.2.6}$$

where the Poisson process measure $\tilde{\nu}^\epsilon$ is constructed according to the measure decomposition (4.2.3). We know that the SDE (4.2.6) is the underlying stochastic process of the probabilistic representation of the solution $u(t,x)$ of the following PIDE in $[0,T]$:

$$\begin{cases} \dfrac{\partial u}{\partial t}(t,x) - \mathcal{L}^*[u](t,x) = f(t,x), \quad \forall (t,x) \in (0,T] \times \mathbb{R}^d, \\ u(0,x) = \varphi(x), \quad \forall x \in \mathbb{R}^d, \end{cases} \tag{4.2.7}$$

where $\mathcal{L}^*[u](t,x)$, the infinitesimal generator of the random process $\Sigma_\epsilon^{1/2} B + N^\epsilon$, is defined as

$$\begin{aligned} \mathcal{L}[u](t,x) =& \frac{1}{2} \sum_{i,j=1}^d (\Sigma_\epsilon^{1/2})(\Sigma_\epsilon^{1/2})^\top \frac{\partial^2 u}{\partial x_i \partial x_j}(t,x) \\ &+ \int_{\mathbb{R}^d} \left(u(t,x+e) - u(t,x) - \sum_{i=1}^d \frac{\partial u}{\partial x_i} e \right) \lambda^\epsilon(de), \end{aligned} \tag{4.2.8}$$

where $\lambda^\epsilon(de)$ is the jump intensity of random process measure which is defined as

$$\nu^\epsilon(de) = \begin{cases} C_{d,\alpha} \dfrac{1}{|e|^{d+\alpha}} de, \quad \epsilon \leq |e|, \\ 0, \quad otherwise. \end{cases} \tag{4.2.9}$$

We add an outer boundary ϵ_b to kernel $\nu^\epsilon(de)$, i.e., $\epsilon \leq |e| \leq \epsilon_b$ so that it will be much easier to select a quadrature rule for approximating spatial integrals with respect to jump variable.

4.3 Numerical examples

Example 3. We consider the fractional Laplacian equation (4.1.1) through solving the PIDE (4.2.7) in $\mathbb{R}^3$. We choose the exact solution to be $u(t,x) = (t+x_1)^2 + x_2x_3$, and consider two different α cases.

Case 1: When $\alpha = 0.5$, we choose $\epsilon = 0.2$ and outside boundary $\epsilon_b = 0.3$, then the variance Σ_ϵ is solved as

$$\Sigma_\epsilon = \frac{0.2^{1.5}8\pi}{9} \cdot C_{3,0.5}\mathbf{I}_{3\times 3}$$

and the forcing term is given as

$$f(t,x) = 2(t+x_1) - \frac{0.25^{1.5}8\pi}{9}C_{3,0.5}.$$

The initial condition $\varphi(x)$ can be gotten from $u(t,x)$, the jump intensity

$$\lambda = \int_E \nu^\epsilon(de) = 8\pi C_{3,0.5}(0.2^{-0.5} - 0.3^{-0.5})$$

and the probability measure

$$\rho(e)de = \frac{1}{8\pi(0.2^{-0.5} - 0.3^{-0.5})|e|^{3.5}}1_{0.2\leq|e|\leq 0.3}de.$$

Case 2: When $\alpha = 1.5$, we choose $\epsilon = 0.2$ and outside boundary $\epsilon_b = 0.3$, then the variance Σ_ϵ is

$$\Sigma_\epsilon = \frac{8\sqrt{0.2}\pi C_{3,1.5}}{3}\mathbf{I}_{3\times 3}$$

and the forcing term is given by

$$f(t,x) = 2(t+x_1) - \frac{8\pi C_{3,1.5}\sqrt{0.3}}{3}.$$

Table 4.1: Errors and convergence rates with respect to Δt in Example 3

$\alpha = 0.5$ \ Δt	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	CR
$\|u - u\|_{L^\infty}$						
cubic	2.2916e-01	1.1179e-01	5.4899e-02	2.7300e-02	1.4526e-02	0.9993
in-cubic	2.2825e-01	1.1172e-01	5.4341e-02	2.6725e-02	1.4026e-02	1.0113
$\|u - u\|_{L^2}$						
cubic	1.7846e-01	8.2772e-02	3.8682e-02	1.8766e-02	9.9604e-03	1.0468
in-cubic	1.6972e-01	7.7344e-02	3.5876e-02	1.6972e-02	8.6047e-03	1.0792
$\alpha = 1.5$ \ Δt	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	CR
$\|u - u\|_{L^\infty}$						
cubic	1.6356	8.1391e-01	4.4791e-01	2.3528e-01	1.1458e-01	0.9461
in-cubic	1.3570	7.2056e-01	3.0769e-01	1.6742e-01	7.4434e-02	1.0482
$\|u - u\|_{L^2}$						
cubic	8.6992e-01	5.1089e-01	2.6265e-01	1.2385e-01	5.7766e-02	0.9870
in-cubic	7.1458e-01	3.9250e-01	1.8760e-01	8.3060e-02	3.6301e-02	1.0838

The initial condition $\varphi(x)$ can be gotten from $u(t, x)$, the jump intensity

$$\lambda = \int_E \nu^\epsilon(de) = \frac{8\pi}{3} C_{3,1.5}(0.2^{-1.5} - 0.3^{-1.5})$$

and the probability measure is given as

$$\rho(e) = \frac{3}{8\pi(0.2^{-1.5} - 0.3^{-1.5})|e|^{4.5}} 1_{0.2 \le |e| \le 0.3}.$$

We expect the first order convergence with respect to both Δt and Δx for $\alpha = 0.5$ and 1.5. Temporal and spatial parameters are given as same as example 1 and results are shown in Table 4.1 and Figure 4.2. We see that both temporal discretization and spatial discretization errors decay at the first order rate that verify the theoretical analysis in Section 3.3.

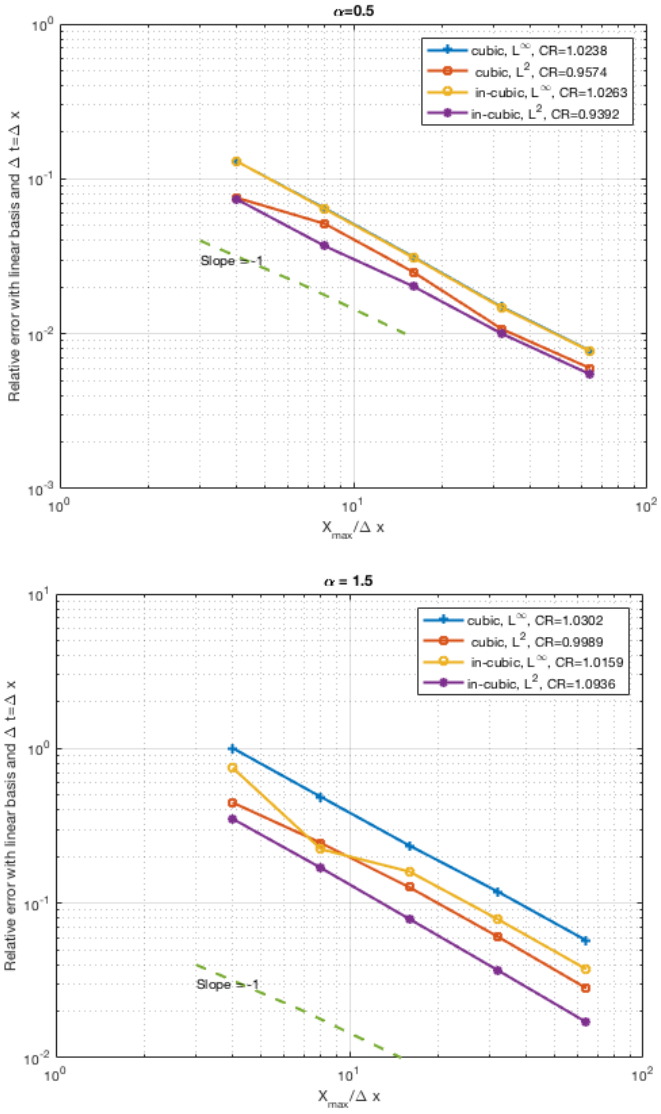

Figure 4.2: Errors and convergence rates with respect to Δx in Example 3

Chapter 5

The PIDEs with Volume Constraints

5.1 Problem setting

Let $\mathscr{D} \in \mathbb{R}^d$ denote a bounded open polygonal domain, we consider the following initial-boundary value problem in $[0,T] \times (\mathscr{D} \cup \mathscr{D}_b)$ where $T > 0$ and $\mathscr{D} \cup \mathscr{D}_b \subset \mathbb{R}^d$ with $d = 1, 2, 3$, i.e.,

$$\begin{aligned} \frac{\partial u}{\partial t}(t,x) - \mathcal{L}[u](t,x) &= f(t,x,u), \quad \forall (t,x) \in (0,T] \times \mathscr{D}, \\ u(0,x) &= u_0(x), \qquad \forall x \in \mathscr{D} \cup \mathscr{D}_b, \\ u(t,x) &= u_b(t,x), \qquad \forall (t,x) \in (0,T] \times \mathscr{D}_b, \end{aligned} \tag{5.1.1}$$

where f is the forcing, u_0 is an initial condition and u_b is an volume constraint acting on a volume $\mathscr{D}_b$ that is disjoint from $\mathscr{D}$. The volume constraint is a natural extension, to the nonlocal case, of boundary conditions for local PDEs. The partial integro-differential operator $\mathcal{L}$ is defined in (3.1.2). The well-posedness of PIDE (5.1.1) with random terminal time has been studied by [9, 34].

5.1.1 The non-divergence form of the PIDE

We assume the coefficients of $\mathcal{L}$ in (3.1.2) have sufficient regularities, such that we can rewrite the PIDE in (3.1.1) in a *non-divergence* form, i.e.,

$$\begin{aligned} \frac{\partial u}{\partial t}(t,x) - \mathcal{L}^*[u](t,x) &= g(t,x,u) \quad \forall (t,x) \in (0,T] \times \mathscr{D}, \\ u(0,x) &= u_0(x), \qquad \forall x \in \mathscr{D} \cup \mathscr{D}_b, \\ u(t,x) &= u_b(t,x), \qquad \forall (t,x) \in (0,T] \times \mathscr{D}_b, \end{aligned} \tag{5.1.2}$$

where the non-divergence form operator $\mathcal{L}^*$ is defined by

$$\begin{aligned}\mathcal{L}^*[u](t,x) := &\sum_{i=1}^{d} \beta_i(t,x)\frac{\partial u}{\partial x_i}(t,x) + \sum_{i,j=1}^{d} K_{ij}(t,x)\frac{\partial^2 u}{\partial x_i x_j}(t,x) \\ &+ \int_E \left[u(t,x+c(t,e)) - u(t,x)\right]\gamma(e)de,\end{aligned} \tag{5.1.3}$$

with the new drift coefficient β_i being

$$\beta_i(t,x) := 2\sum_{j=1}^{d} \frac{\partial K_{ij}(t,x)}{\partial x_j}(t,x) - b_i(t,x).$$

The new forcing term g is defined by

$$g(t,x,u) := f(t,x,u) + \left[\sum_{i,j=1}^{d} \frac{\partial^2 K_{ij}}{\partial x_i \partial x_j}(t,x) - \sum_{i=1}^{d} \frac{\partial b_i}{\partial x_i}(t,x)\right] u(t,x).$$

Hereafter, we focus on solving the PIDE in (5.1.2).

5.2 The new probabilistic numerical scheme for the PIDEs

In this section, we move on proposing the probabilistic numerical schemes. The nonlinear Feynman-Kac theory studied by [5] establishes a probabilistic representation of the solution in PIDE (5.1.2) that is the same expression as the case of PIDE in the unbounded domain $\mathbb{R}^d$. Next, we consider the way to formulate the random time of the underlying *backward* stochastic process $\overleftarrow{X}_s$ in (3.2.3) exits the bounded domain $\mathscr{D}$.

5.2.1 Mesh generation

We construct the temporal-spatial meshes for approximating the PIDE (3.2.1). We use the same temporal mesh $\mathcal{T}$ and remark that the temporal mesh $\mathcal{T}$ is not adapted to the jump instances of the stochastic process defined in (3.2.3). In the spatial domain, we define a triangular or tetrahedral mesh, denoted by $\mathcal{K}_{\Delta x}$, for the closed domain $\overline{\mathscr{D}}$, where Δx indicates the maximum size of the elements. We denote by $\mathcal{J}$ the index set for the grid points of $\mathcal{K}_{\Delta x}$. Since the grid points can be divided into interior nodes $\{x_i \in \mathcal{K}_{\Delta x} \cap \mathscr{D}\}$ and boundary nodes $\{x_i \in$

$\mathcal{K}_{\Delta x} \cap \partial\mathscr{D}\}$, we denote by $\mathcal{J}_{\text{interior}}$ and $\mathcal{J}_{\text{boundary}}$ the index sets of interior and boundary nodes, respectively.

5.2.2 The probabilistic representation of the PIDE solution

We now restrict the PIDE in (5.1.2) within one time step $[t_n, t_{n+1}]$ and derive a probabilistic representation of $u(t_{n+1}, x)$ using $u(t_n, x)$ and $g(t, x, u(t, x))$ for $(t, x) \in [t_n, t_{n+1}] \times (\mathscr{D} \cup \mathscr{D}_b)$. According to Dynkin's formula [12], the operator $-\mathcal{L}^*$ in (5.1.3) is the *infinitesimal generator* of $\overleftarrow{X}_s^{t_{n+1},x}$ for $s \in [t_n, t_{n+1}]$, so that we can exploit the Feynman-Kac theory to derive a probabilistic representation to the solution of the initial-boundary value problem in (5.1.2). For any test function $u(t, x) \in C^1([t_n, t_{n+1}]) \times C^2(\mathbb{R}^d)$, we apply the Itô formula to $u(s, \overleftarrow{X}_s^{t_{n+1},x})$ for $s \in [t_n, t_{n+1}]$, and take mathematical expectation of the expansion, then we have

$$u(t_{n+1}, x) = \mathbb{E}\left[u(s, \overleftarrow{X}_s^{t_{n+1},x}) - \int_{t_{n+1}}^{s} \left(\frac{\partial u}{\partial t} - \mathcal{L}^*[u]\left(t, \overleftarrow{X}_t^{t_{n+1},x}\right)\right) dt\right], \tag{5.2.1}$$

where the operator $\mathcal{L}^*$ is defined in (5.1.3). If $u(t, x)$ is a strong solution of the PIDE in (5.1.2), then we have a probabilistic representation of $u(t_{n+1}, x)$, i.e.,

$$u(t_{n+1}, x) = \mathbb{E}\left[u(s, \overleftarrow{X}_s^{t_{n+1},x}) - \int_{t_{n+1}}^{s} g\left(t, \overleftarrow{X}_t^{t_{n+1},x}, u(t, \overleftarrow{X}_t^{t_{n+1},x})\right) dt\right], \tag{5.2.2}$$

where $s \in [t_n, t_{n+1}]$ and $g(t, x, u)$ is the forcing term in (5.1.2).

The boundary condition of the problem in (5.1.2) can be incorporated into the probabilistic representation in (5.2.2) by defining an exit time of $\overleftarrow{X}_t^{t_{n+1},x}$, i.e.,

$$\tau_{t_{n+1},x} := \sup\left\{s < t_{n+1} \,\middle|\, \overleftarrow{X}_s^{t_{n+1},x} \notin \mathscr{D} \cup \mathscr{D}_b, x \in \mathscr{D}\right\},$$

which indicates the *first* exit time of the process $\overleftarrow{X}_s^{t_{n+1},x}$ defined in (3.2.3). Then, all the possible paths of the process $\overleftarrow{X}_s^{t_{n+1},x}$ within $[t_n, t_{n+1}]$ can be divided into two subsets, denoted by $\mathcal{S}_1$ and $\mathcal{S}_2$, i.e.,

$$\mathcal{S}_1 := \left\{\tau_{t_{n+1},x} \le t_n\right\} \quad \text{and} \quad \mathcal{S}_2 := \left\{\tau_{t_{n+1},x} > t_n\right\}, \tag{5.2.3}$$

where $\mathcal{S}_1$ describes the event that the state of $\overleftarrow{X}_s^{t_{n+1},x}$ remains in $\mathscr{D} \cup \mathscr{D}_b$ for any $s \in [t_n, t_{n+1}]$, and $\mathcal{S}_2$ describes the event that $\overleftarrow{X}_s^{t_{n+1},x}$ will exit the domain $\mathscr{D} \cup \mathscr{D}_b$ at some time instance within $[t_n, t_{n+1}]$. It is easy to see that $\mathbb{P}(\mathcal{S}_1) + \mathbb{P}(\mathcal{S}_2) = 1$. We can divide the expectation in (5.2.2) into two parts based on the subsets $\mathcal{S}_1$ and $\mathcal{S}_2$, i.e.,

$$
\begin{aligned}
& u(t_{n+1}, x) \\
=& \, \mathbb{P}(\mathcal{S}_1)\, \mathbb{E}\left[u(t_n, \overleftarrow{X}_{t_n}^{t_{n+1},x}) - \int_{t_{n+1}}^{t_n} g\left(t, \overleftarrow{X}_t^{t_{n+1},x}, u(t, \overleftarrow{X}_t^{t_{n+1},x})\right) dt \right] \\
& + \, \mathbb{P}(\mathcal{S}_2)\, \mathbb{E}\left[u_b(\tau_{t_{n+1},x}, \overleftarrow{X}_{\tau_{t_{n+1},x}}^{t_{n+1},x}) - \int_{t_{n+1}}^{\tau_{t_{n+1},x}} g\left(t, \overleftarrow{X}_t^{t_{n+1},x}, u(t, \overleftarrow{X}_t^{t_{n+1},x})\right) dt \right].
\end{aligned}
\tag{5.2.4}
$$

Even though we defined a temporal-spatial mesh in Section 5.2.1, we have not yet conducted any discretization of the PIDE in (5.1.2). The discretization includes four steps, i.e., (i) approximating the dynamic of the backward stochastic process $\overleftarrow{X}_{t_n}^{t_{n+1},x}$ on the temporal-spatial mesh, (ii) discretizing the temporal integrals in (5.2.4), (iii) approximating the expectation operator $\mathbb{E}[\cdot]$, and (iv) reconstructing the solution $u(t, x)$. Those four steps will be described in Section 5.2.3, 5.2.4, 5.2.5 and 5.2.6, respectively.

5.2.3 Approximation of the backward stochastic process

One of the challenges in discretizing (5.2.4) is to accurately approximate the second term of the right-hand side, i.e., the exit probability $\mathbb{P}(\mathcal{S}_2)$ and corresponding conditional expectation. Standard strategies can only provide $\mathcal{O}(\sqrt{\Delta t})$, which, if used, will deteriorate the overall convergence rate of our method. To circumvent such challenge, we introduce additional truncation and discretization to the Euler scheme in (3.2.6), such that the probability $\mathbb{P}(\mathcal{S}_2)$ determined by the final discretized $\overleftarrow{X}_s^{t_{n+1},x}$ will be on the order of $\mathcal{O}((\Delta t)^2)$. If achieved, the second term on the right-hand side of (5.2.4) can be neglected without affecting the desired convergence rate.

Specifically, we truncate the number of jumps within $[s, t_{n+1}]$, controlled by $N_{t_{n+1}-s}$, to maximumly one jump, i.e., replacing $N_{t_{n+1}-s}$ in (3.2.6) with $N_{t_{n+1}-s} \vee 1 = \min(N_{t_{n+1}-s}, 1)$. Such truncation is based on the fact that the probability of having more than one jumps within Δt is on the order of $\mathcal{O}((\lambda \Delta t)^2)$. Moreover, we have another two critical observations: (a) the further the starting point x away from the boundary $\partial\mathscr{D}$, the smaller the exit probability $\mathbb{P}(\mathcal{S}_2)$;

(b) We only need to set the starting point x of $\overleftarrow{X}_s^{t_{n+1},x}$ on the grid points of the spatial mesh $\mathcal{K}_{\Delta x}$ defined in Section 5.2.1, in order to compute the approximation of $u(t_{n+1},x)$ in (5.2.4) on the grid points, and the value of $u(t_{n+1},x)$ at other locations will be approximated by interpolation. Based on such two observations, we set the mesh size Δx on the order of $\mathcal{O}(\sqrt{\Delta t})$, so that the process $\overleftarrow{X}_s^{t_{n+1},x}$ starting from any interior node of $\mathcal{K}_{\Delta x}$ will have a very small exit probability. In this case, piecewise cubic interpolation with $\mathcal{O}((\Delta x)^4)$ is needed to recover a $\mathcal{O}((\Delta t)^2)$ local error. As such the final discretization scheme of $\overleftarrow{X}_s^{t_{n+1},x}$ in (3.2.3) is given by

$$\widehat{X}_s^{t_{n+1},x_i} := x_i + b(t_{n+1},x_i)(t_{n+1}-s) + \sigma(t_{n+1},x_i)\Delta\overleftarrow{W}_{t_{n+1}-s} + \sum_{k=0}^{N_{t_{n+1}-s}\vee 1} c(t_{n+1},e_k), \quad (5.2.5)$$

for all interior grid points $x_i \in \mathscr{D}\cap\mathcal{K}_{\Delta x}$.

Lemma 5.1. *Given the temporal-spatial mesh $\mathcal{T}\times\mathcal{K}_{\Delta x}$, if all the interior grid points $x_i \in \mathscr{D}\cap\mathcal{K}_{\Delta x}$ satisfies*

$$\text{dist}(x_i,\partial\mathscr{D}) > \bar{b}\Delta t + \bar{\sigma}(\Delta t)^{\frac{1}{2}-\varepsilon}, \quad (5.2.6)$$

with $\bar{b}$ and $\bar{\sigma}$ the upper bounds of $|b|,|\sigma|$ in $[0,T]\times\mathscr{D}\cup\mathscr{D}_b$. Then for any positive constant $\varepsilon>0$, there exists a sufficiently small Δt such that

$$\mathbb{P}(\widehat{\tau}_{t_{n+1},x_i} > t_n) \le C(\Delta t)^{\varepsilon}\exp\left(-\frac{1}{(\Delta t)^{2\varepsilon}}\right),$$

where $\widehat{\tau}_{t_{n+1},x_i}$ is the exit time of the approximation $\widehat{X}_s^{t_{n+1},x_i}$ in (5.2.5)*, and the constant $C>0$ is independent of Δt.*

The above lemma follows the approach from Theorem 3.2 in J.Yang [48]. When substituting the approximation $\widehat{X}_s^{t_{n+1},x_i}$ into the probabilistic representation in (5.2.4), the second term on the right hand side can be viewed as an $\mathcal{O}((\Delta t)^2)$ truncation error, such that (5.2.4) can be rewritten as

$$u(t_{n+1},x_i) = \mathbb{E}\left[u(t_n,\widehat{X}_{t_n}^{t_{n+1},x_i}) - \int_{t_{n+1}}^{t_n} g\left(t,\widehat{X}_t^{t_{n+1},x_i},u(t,\widehat{X}_t^{t_{n+1},x_i})\right)dt\right] + R_{n+1}, \quad (5.2.7)$$

where $x_i\in\mathscr{D}\cap\mathcal{K}_{\Delta x}$, and R_{n+1} denotes the approximation error between (5.2.4) and (5.2.7).

5.2.4 Temporal discretization of the probabilistic representation

Now we discretize the temporal integral in (5.2.7). The implicit Euler scheme can be applied that guarantees stability and achieves desired first order convergence with respect to Δt. Specifically, for any interior grid point $x_i \in \mathscr{D} \cap \mathcal{K}_{\Delta x}$, we have

$$u(t_{n+1}, x_i) = \mathbb{E}\left[u(t_n, \widehat{X}_{t_n}^{t_{n+1},x_i})\right] + \Delta t\, g(t_{n+1}, x_i, u(t_{n+1}, x_i)) + R_{n+1}, \tag{5.2.8}$$

where the residual R_{n+1} is assumed to absorb the truncation error from the implicit Euler scheme. The advantage of using the probabilistic representation is that

$$\mathbb{E}\left[g\left(t_{n+1}, \widehat{X}_{t_{n+1}}^{t_{n+1},x_i}, u(t_{n+1}, \widehat{X}_{t_{n+1}}^{t_{n+1},x_i})\right)\right] = g\left(t_{n+1}, x_i, u(t_{n+1}, x_i)\right),$$

due to the Markovian property of $\widehat{X}_{t_{n+1}}^{t_{n+1},x_i}$. As such, the value of $u(t_{n+1}, x_i)$ can be computed independently without knowing $u(t_{n+1}, x_j)$ at any other locations x_j, when neglecting the residual terms in (5.2.8). Such decoupling helps avoid solving linear systems with possibly dense matrices for the nonlocal problem under consideration.

5.2.5 Approximation of the conditional expectation

This subsection is to construct a quadrature rule for discretizing the conditional expectation in (5.2.4). To proceed, we write out the expression of the expectation as

$$\begin{aligned}
&\mathbb{E}\left[u(t_n, \widehat{X}_{t_n}^{t_{n+1},x_i})\right] \\
=&\,\mathbb{P}(N_{\Delta t}=0)\,\mathbb{E}\left[u\left(t_n,\, x_i + b(t_{n+1},x_i)\Delta t + \sigma(t_{n+1},x_i)\Delta\tilde{W}\right)\right] \\
&+ \mathbb{P}(N_{\Delta t}=1)\,\mathbb{E}\left[u\left(t_n,\, x_i + b(t_{n+1},x_i)\Delta t + \sigma(t_{n+1},x_i)\Delta\tilde{W}\right) + c(t_{n+1},e)\right] \\
=&\,\mathrm{e}^{-\lambda\Delta t}\left[\int_{\mathbb{R}^d} u\big(t_n, x_i + b(t_{n+1},x_i)\Delta t + \sigma(t_{n+1},x_i)\sqrt{\Delta t}\xi\big)\varrho(\xi)d\xi\right. \\
&\left.+ \lambda\Delta t\int_{\mathbb{R}^d}\int_E u\big(x_i + b(t_{n+1},x_i)\Delta t + \sigma(t_{n+1},x_i)\sqrt{\Delta t}\xi + c(t_{n+1},e)\big)\varrho(\xi)\rho(e)dqd\xi\right],
\end{aligned} \tag{5.2.9}$$

where $\varrho(\xi)$ is the standard Gaussian probability density for $\Delta\check{W}$, $\rho(e)$, defined in (3.1.3), is the probability density function for the jump amplitude $c(t_{n+1}, e)$, and $\mathbb{P}(N_{\Delta t} = 0) = \mathrm{e}^{-\lambda\Delta t}$, $\mathbb{P}(N_{\Delta t} = 1) = \mathrm{e}^{-\lambda\Delta t}\lambda\Delta t$ with λ the jump intensity given in (3.1.3).

The two integrals in (5.2.9) can be approximated by Gaussian quadrature rules. The integral with respect to $\varrho(\xi)$ can be approximated by the Gauss-Hermite rule. It is proved in Section 3.3 that the 3-point Gauss-Hermite rule (3^d total) is sufficient to achieve a $\mathcal{O}((\Delta t)^2)$ local truncation error. The integral with respect to the jump amplitude e can be approximated by either Newton-Cotes or Gaussian quadrature rules, depending on the nonlocal kernel $\gamma(e)$ under consideration.

Without loss of generality, we denote by $\{w_m, \xi_m\}_{m=0}^{M}$ the Hermite quadrature weights and abscissae, and by $\{v_l, q_l\}_{l=1}^{L}$ the selected rule for estimating the integrals in (5.2.9) with respect to e. As such, the approximate expectation, denoted by $\widehat{\mathbb{E}}[\cdot]$, is defined as

$$\begin{aligned}
&\widehat{\mathbb{E}}[u(t_n, \widehat{X}_{t_n}^{t_{n+1},x_i})] \\
&= \mathrm{e}^{-\lambda\Delta t}\Bigg[\sum_{m=1}^{M} w_m u\Big(t_{n+1}, x_i + b(t_{n+1}, x_i)\Delta t + \sigma(t_{n+1}, x_i)\sqrt{2\Delta t}\,\xi_m\Big) \\
&\quad + \lambda\Delta t \sum_{m=1}^{M}\sum_{l=1}^{L} w_m v_l\, u\Big(t_n, x_i + b(t_{n+1}, x_i)\Delta t + \sigma(t_{n+1}, x_i)\sqrt{2\Delta t}\,\xi_m + c(t_{n+1}, q_l)\Big)\Bigg].
\end{aligned} \tag{5.2.10}$$

Substituting (5.2.10) into (5.2.8), we obtain an updated approximation scheme

$$u(t_{n+1}, x_i) = \widehat{\mathbb{E}}\left[u(t_n, \widehat{X}_{t_n}^{t_{n+1},x_i})\right] + \Delta t\, g(t_{n+1}, x_i, u(t_{n+1}, x_i)) + R_{n+1}, \tag{5.2.11}$$

where the residual R_{n+1} is assumed to absorb one additional error from (5.2.10).

5.2.6 The fully discrete scheme

It is easy to see that a temporal-spatial discretization scheme can be defined based on (5.2.11) by neglecting the residual term R_{n+1}. Nevertheless, the quadrature points used in (5.2.10) may not be the grid points on the spatial mesh $\mathcal{K}_{\Delta x}$, such that approximating $u(t_{n+1}, x_i)$ only at the interior grid points $x_i \in \mathscr{D} \cap \mathcal{K}_{\Delta x}$ is not sufficient to move forward to the next time step. Thus,

we need to reconstruct $u(t_{n+1}, x)$ on the closed domain $\overline{\mathscr{D}}$ based on the approximate nodal values of $u(t_{n+1}, x_i)$ and the volume constraint $u_b(t, x)$. In this work, the approximation of $u(t_{n+1}, x_i)$, denoted by $u^{n,p}(x)$, is defined by p-th order piecewise Lagrange interpolation on the spatial mesh $\mathcal{K}_{\Delta x}$, i.e.,

$$u^{n+1,p}(x) = \sum_{i \in \mathcal{J}_{\text{interior}}} u_i^{n+1}\, \psi_i(x) + \sum_{i \in \mathcal{J}_{\text{boundary}}} u_b(t_{n+1}, x_i)\, \psi_i(x), \tag{5.2.12}$$

where u_i^{n+1} denotes the nodal approximation of $u(t_{n+1}, x)$ at x_i for $i \in \mathcal{J}_{\text{interior}}$, u_b is the boundary condition and ψ_i is the nodal basis function associated with x_i. We combine all the discretization schemes discussed in Section 5.2.3, 5.2.4, 5.2.5 and 5.2.6 into a fully discrete scheme, presented in Algorithm 1.

Algorithm 1: *The fully discrete scheme*

Input: u_0, u_b, g, $\mathcal{L}^*$, $\mathscr{D}$, $\mathscr{D}_b$

Output: The approximation $u^{n,p}(x)$ for $n = 1, \ldots, N_t$;

1: Generate the temporal and spatial meshes $\mathcal{T}_{\Delta t}$ and $\mathcal{K}_{\Delta x}$;
2: **for** $n = 0, \ldots, N_t - 1$ **do**
3: **for** $i \in \mathcal{J}_{\text{interior}}$ **do**
4: • Generate quadrature points used in (5.2.10) for $m = 1, \ldots, M, l = 1, \ldots, L$

$$\widetilde{X}_{i,m} = x_i + b(t_{n+1}, x_i)\Delta t + \sigma(t_{n+1}, x_i)\sqrt{2\Delta t}\,\xi_m,$$
$$\bar{X}_{i,l,m} = x_i + b(t_{n+1}, x_i)\Delta t + \sigma(t_{n+1}, x_i)\sqrt{2\Delta t}\,\xi_m + c(t_{n+1}, q_l);$$

5: • Evaluate the interpolant $u^{n,p}(x)$ to obtain $u^{n,p}(\widetilde{X}_{i,m})$ and $u^{n,p}(\bar{X}_{i,l,m})$;
6: • Compute $\widehat{\mathbb{E}}[u^{n,p}(x_i, \widehat{X}_{t_n}^{t_{n+1}, x_i})]$ by substituting $u^{n,p}(\widetilde{X}_{i,m})$ and $u^{n,p}(\bar{X}_{i,l,m})$ into the quadrature rule in (5.2.10);
7: • Compute the nodal value u_i^{n+1} by solving a nonlinear equation based on (5.2.11)

$$u_i^{n+1} = \widehat{\mathbb{E}}\left[u^{n,p}(x_i, \widehat{X}_{t_n}^{t_{n+1}, x_i})\right] + \Delta t\; g(t_{n+1}, x_i, u_i^{n+1});$$

8: **end for**
9: Construct the interpolant $u^{n+1,p}(x)$ using u_i^{n+1} and $u_b(t, x)$ via (5.2.12);
10: **end for**

5.3 Numerical examples

Example 4. We consider the following three-dimensional nonlocal diffusion problem with Dirichlet boundary condition,

$$\begin{cases} \dfrac{\partial u}{\partial t}(t,x) - \displaystyle\int_E \Big(u(t,x+e) - u(t,x)\Big)\gamma(e)de = f(t,x,u), \forall (t,x) \in [0,T] \times \mathscr{D}, \\ u(0,x) = \varphi(x), \forall x \in \mathscr{D} \cup \mathscr{D}_b, \\ u(t,x) = x^2 exp(-\dfrac{t}{10}), \forall (t,x) \in [0,T] \times \mathscr{D}_b, \end{cases} \tag{5.3.1}$$

where the symmetric kernel $\gamma(e) = 8^2 \times 1_{|e| \leq \frac{1}{8}}$ and we consider the domain $\mathscr{D}_{total} := \mathscr{D} \cup \mathscr{D}_b$ in two cases, one is a cube $[0,1]^3$ and another one is a three-dimensional river model that is shown in Figure 7.1 in Appendix 7.5. The interior region $\mathscr{D}$ consists of all grid points such that their all quadrature points fall inside the union region $\mathscr{D}_{total}$. We choose the exact solution to be

$$u(t,x) = x^2 exp(-\frac{t}{10}),$$

so that the forcing term f is given by

$$f(t,x) = -\frac{u(t,x)}{10} - \frac{\pi}{640} exp(-\frac{t}{10}).$$

And the initial condition $\varphi(x)$ can be solved directly from u. To test the convergence rate with respect to Δt, we set $\Delta x = \frac{1}{32}$ and use linear interpolation with Delaunay triangulation which is enough to guarantee $\mathcal{O}(\Delta t)$. Since the symmetric kernel $\gamma(e)$ is uniform with the support $|e| \leq \frac{1}{8}$, we use the sparse grids based on Gauss-Legendre quadrature rules with enough high level and set $M_u = 1$, the time interval $[0,1]$ is divided into N_t with $\Delta t = 2^{-1}, 2^{-2}, 2^{-3}, 2^{-4}, 2^{-5}$. The errors of inner grid points are measured in two different measure norms L^∞ and L^2. Results are shown in Table 4. As expected, the convergence rate with respect to Δt is first order in both space domains. Next, we test the convergence rate with respect to Δx, we consider a small river model that is shown in 7.2, results are listed in Figure 4. As Theorem 3 in stated [50], using linear interpolation with Delaunay Triangulation can get

Table 5.1: Errors and convergence rates with respect to Δt in Example 4, where T=1, $\Delta x = \frac{1}{32}$

Δt	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	CR
$\|u-u\|_{L^\infty}$						
cubic	1.1046e-01	5.5788e-02	2.7572e-02	1.3262e-02	6.0030e-03	1.0476
river	4.9923e-01	2.6885e-01	1.3963e-01	7.0783e-02	3.5222e-02	0.9576
$\|u-u\|_{L^2}$						
cubic	5.1275e-02	2.6937e-02	1.3507e-02	6.4277e-03	2.7976e-03	1.0459
river	2.3797e-01	1.2670e-01	6.5254e-02	3.2838e-02	1.6176e-02	0.9706

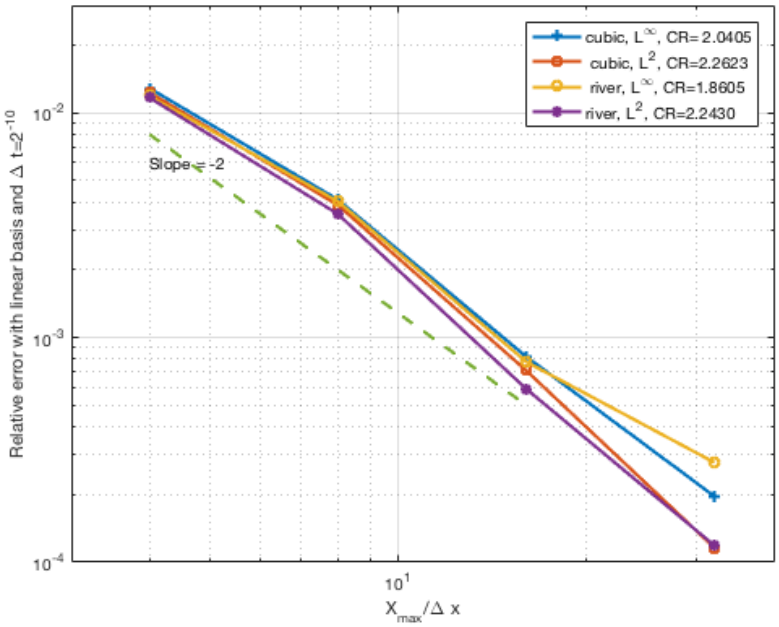

Figure 5.1: Errors and convergence rates with respect to Δx in Example 4

the second order convergence rate with respect to Δx.

Chapter 6

Runaway electrons model

6.1 Problem setting

We consider a three-dimensional runaway electrons model describing the dynamics of the magnitude of the relativistic momentum, denoted by p, the cosine of the pitch angle θ, denoted by $\xi = \cos\theta$, and the minor radius, denoted by r. The relativistic momentum p is normalized using the thermal momentum and the time is normalized using the thermal collisional frequency. That is, if $\hat{p}$ and $\hat{t}$ denote the dimensional variables, then $p = \hat{p}/(mv_T)$ and $t = \nu_{ee}\hat{t}$, where $v_T = \sqrt{2T/m}$ is the thermal speed with T the plasma temperature and m the electron mass, and the thermal collision frequency is $\nu_{ee} = e^4 n \ln n\Lambda/(4\pi\varepsilon_0 m^2 v_T^3)$ with e the absolute value of the electron charge, ε_0 is the vacuum permittivity and Λ the Coulomb logarithm. Specifically, those dynamics are defined by the following stochastic differential equations

$$\begin{cases} dp = \left[E\xi - \dfrac{\gamma p}{\tau}(1-\xi^2) - C_F + \dfrac{1}{p^2}\dfrac{\partial}{\partial p}\left(p^2 C_A\right)\right] dt + \sqrt{2C_A}\, dB_p, \\ d\xi = \left[\dfrac{E\,(1-\xi^2)}{p} - \dfrac{\xi(1-\xi^2)}{\tau\gamma} - 2\xi\dfrac{C_B}{p^2}\right] dt + \dfrac{\sqrt{2C_B}}{p}\sqrt{1-\xi^2}\, dB_\xi, \\ dr = \sqrt{2D_r} dB_r, \end{cases} \tag{6.1.1}$$

where B_p, B_ξ and B_r are the standard Brownian motions, E is the electric field, and the functions C_A, C_B, C_F and D_r are defined by

$$\begin{aligned} C_A(p) &= \bar{\nu}_{ee}\,\bar{v}_T^2\,\frac{\psi(y)}{y}, \\ C_B(p) &= \frac{1}{2}\,\bar{\nu}_{ee}\,\bar{v}_T^2\,\frac{1}{y}\left[Z+\phi(y)-\psi(y)+\frac{y^2}{2}\delta^4\right], \\ C_F(p) &= 2\,\bar{\nu}_{ee}\,\bar{v}_T\,\psi(y), \\ D_r(p) &= D_0\exp(-(p/\Delta p)^2), \end{aligned}$$

$$\phi(y)=\frac{2}{\sqrt{\pi}}\int_0^y e^{-s^2}ds\,, \qquad \psi(y)=\frac{1}{2y^2}\left[\phi(y)-y\frac{d\phi}{dy}\right],$$

$$y=\frac{p}{\gamma}, \qquad \gamma=\sqrt{1+(\delta p)^2}, \qquad \delta=\frac{v_T}{c}=\sqrt{\frac{2T}{mc^2}},$$

with Z, c denoting the ion effective charge and the speed of light, respectively.

The problem we want to address is the computation of the probability that an electron with coordinates (p,ξ,r) will runaway at, or before, a prescribed time. By "runaway" we mean that, as a result of the electric field acceleration, the electron will reach a prescribed momentum, $p_{\max}$. The dependence of the runaway probability on $p_{\max}$ becomes negligible for large enough $p_{\max}$, which is the reason why this dependence is not usually accounted for explicitly. More formally, for a given $(t,p,\xi,r)\in[0,T_{\max}]\times[p_{\min},p_{\max}]\times[-1,1]\times[0,1]$, where $p_{\min}$ is a lower momentum boundary, the runaway probability, $P_{\mathrm{RE}}(t,p,\xi,r)$, is defined as the probability that an electron located at (p,ξ,r) at the initial time instant $t_0=0$ will acquire a momentum $p_{\max}$ on, or before $t>0$.

Mathematically, the runaway probability can be described as the escape probability of a stochastic dynamical systems. For notational simplicity, we define

$$X_t := (p,\xi,r),$$

and rewrite the SDE in (6.1.1) using X_t, i.e.,

$$X_t = X_0 + \int_0^t b(X_s)ds + \int_0^t \sigma(X_s)dB_s \quad \text{with } X_0 \in \mathcal{D} \subset \mathbb{R}^3, \tag{6.1.2}$$

where $\mathcal{D} = [p_{\min}, p_{\max}] \times [-1, 1] \times [0, 1]$, and the drift b and the diffusivity σ can be easily defined based on Eq. (6.1.1). In the following sections, we will use (6.1.2) to introduce our probabilistic scheme and will come back to Eq. (6.1.1) in the section of numerical examples.

We divide the boundary of $\mathcal{D}$ into three parts $\partial\mathcal{D}_1$, $\partial\mathcal{D}_2$ and $\partial\mathcal{D}_3$, defined by

$$\begin{aligned}
\partial\mathcal{D}_1 &:= \{p = p_{\max}\} \cap \partial\mathcal{D}, \\
\partial\mathcal{D}_2 &:= (\{p = p_{\min}\} \cup \{r = 1\}) \cap \partial\mathcal{D}, \\
\partial\mathcal{D}_3 &:= (\{\xi = -1\} \cup \{\xi = 1\} \cup \{r = 0\}) \cap \partial\mathcal{D},
\end{aligned}$$

such that $\partial\mathcal{D}_1 \cup \partial\mathcal{D}_2 \cup \partial\mathcal{D}_3 = \partial\mathcal{D}$. The boundary $\partial\mathcal{D}_1$ represents the runaway boundary. To give a formal definition of the runaway probability, we denote the runaway time of X_t by

$$\tau := \inf\left\{t > 0 \,|\, X_t \in \partial\mathcal{D}_1\right\},$$

which represents the earliest escape time of the process X_t that initially starts from $X_0 = x \in \mathcal{D}$. Then, the runaway probability can be formally defined by

$$P_{\mathrm{RE}}(t, x) = \mathbb{P}\left\{\tau \leq t \,|\, X_0 = x \in \mathcal{D}\right\}. \tag{6.1.3}$$

For a fixed $T \in [0, T_{\max}]$, the probability $P_{\mathrm{RE}}(T, x)$ can be represented by the solution of the adjoint equation of the Fokker-Planck equation based on (6.1.2). Such adjoint equation is a

backward parabolic terminal boundary value problem, i.e.,

$$\begin{aligned}
\frac{\partial u(t,x)}{\partial t} + \mathcal{L}^*(t,x)[u(t,x)] = 0 \quad &\text{for} \quad x \in \mathcal{D}, t < T, \\
u(t,x) = 1 \quad &\text{for} \quad x \in \partial\mathcal{D}_1, t \leq T, \\
u(t,x) = 0 \quad &\text{for} \quad x \in \partial\mathcal{D}_2, t \leq T, \\
\nabla u(t,x) = 0 \quad &\text{for} \quad x \in \partial\mathcal{D}_3, t \leq T, \\
u(T,x) = 0 \quad &\text{for} \quad x \in \mathcal{D},
\end{aligned} \tag{6.1.4}$$

where the operator $\mathcal{L}^*(t,x)$ is the adjoint of the Fokker-Planck operator, defined by

$$\mathcal{L}^*(x)[u] := \sum_{i=1}^{d} b_i \frac{\partial u}{\partial x^i} + \frac{1}{2} \sum_{i,j=1}^{d} (\sigma\sigma^\top)_{i,j} \frac{\partial^2 u}{\partial x^i x^j},$$

with b_i the i-th component of the drift $b(x)$, $(\sigma\sigma^\top)_{i,j}$ the (i,j)-th entry of $\sigma\sigma^\top$ and x^i the i-th component of x. It is easy to see that $P_{\mathrm{RE}}(T,x)$ can be represented by

$$P_{\mathrm{RE}}(T,x) = u(0,x). \tag{6.1.5}$$

It should be noted that the runaway probability at each time T requires a solution of the adjoint equation in (6.1.4), such that recovering the entire dynamic of P_{RE} in $[0, T_{\max}]$ requires a sequence of PDE solutions. However, due to the time *independence* of b and σ in (6.1.2) considered in this work, the dynamic of $P_{\mathrm{RE}}(t,x)$ for $(t,x) \in [0, T_{\max}] \times \mathcal{D}$ can be represented by

$$P_{\mathrm{RE}}(t,x) = u(T_{\max} - t, x) \quad \text{for} \;\; t \in [0, T_{\max}], \tag{6.1.6}$$

where u is the solution of (6.1.4) with $T = T_{\max}$.

6.2 A sparse-grid probabilistic method for the adjoint equation

The theoretical foundation of our probabilistic scheme is the Feynman-Kac theory that links the SDE in equation (6.1.2) to the adjoint problem in equation (6.1.4). This section focuses

on solving the adjoint equation in equation (6.1.4). The probabilistic representation of $u(t,x)$ and the temporal discretization is given in Section 6.2.1. The spatial discretization including a special treatment of the involved random escape time τ is provided in Section 6.2.2.

6.2.1 Temporal discretization

To write out the probabilistic representation of $u(t,x)$ in Eq. (6.1.4), we need to rewrite the SDE in Eq. (6.1.2) as the conditional form, i.e.,

$$X_s^{t,x} = x + \int_t^s b(X_{\bar{t}}^{t,x})d\bar{t} + \int_t^s \sigma(X_{\bar{t}}^{t,x})dB_{\bar{t}} \quad \text{for} \;\; s \geq t, \tag{6.2.1}$$

where the superscript ^{t,x} indicates the condition that $X_s^{t,x}$ starts from $(t,x) \in [0, T_{\max}] \times \mathcal{D}$. Accordingly, we can define the *conditional escape time*

$$\tau_{t,x} := \min(\tau_{t,x}^1, \tau_{t,x}^2) \tag{6.2.2}$$

with

$$\tau_{t,x}^1 := \inf\{s > t \,|\, X_s^{t,x} \in \partial\mathcal{D}_1\}, \;\; \tau_{t,x}^2 := \inf\{s > t \,|\, X_s^{t,x} \in \partial\mathcal{D}_2\}, \tag{6.2.3}$$

such that the probabilistic representation of the solution $u(t,x)$ of the adjoint equation in (6.1.4) is given by

$$u(t,x) = \mathbb{E}\left[u\left(s \wedge \tau_{t,x}, X_{s\wedge\tau_{t,x}}^{t,x}\right)\right], \tag{6.2.4}$$

where $s \wedge \tau_{t,x}$ denotes the minimum of $\tau_{t,x}$ and s, $\tau_{t,x}$ is given in Eq. (6.2.2), and $X_{s\wedge\tau_{t,x}}^{t,x}$ is defined based on Eq. (6.2.1).

We then discretize the probabilistic representation of u in Eq. (6.2.4). The SDE in Eq. (6.1.2) can be discretized in the interval $[t_n, t_{n+1}]$ using the forward Euler scheme:

$$X_{n+1}^{t_n,x} = x + b(x)\Delta t + \sigma(x)\Delta B, \tag{6.2.5}$$

where $\Delta B := B_{t_{n+1}} - B_{t_n}$ and such that the Eq. (6.2.4) can be discretized as

$$u^n(x) = \mathbb{E}\left[u^{n+1}\left(X_{n+1}^{t_n,x}\right)\mathbf{1}_{\{\tau_{t_n,x}>t_{n+1}\}}\right] + \mathbb{P}\left(\tau_{t_n,x}^1 \leq t_{n+1}\right), \tag{6.2.6}$$

where $u^n(x) \approx u(t_n, x)$, $\tau_{t_n,x}$ is defined in Eq. (6.2.2). The escape time $\tau_{t_n,x}$ in Eq. (6.2.6) should be defined by replacing $X_s^{t,x}$ with the Euler discretization, i.e., $X_s^{t_n,x} = x + b(x)(s - t_n) + \sigma(x)(B_s - B_{t_n})$ for $s \geq t_n$ in Eq. (6.2.2). We use the same notation without creating confusion. And $\mathbf{1}_{\{\tau_{t_n,x}>t_{n+1}\}}$ is the characteristic function of the event that $X_s^{t_n,x}$ does not escape the domain $\mathcal{D}$ via $\partial\mathcal{D}_1 \cup \partial\mathcal{D}_2$ before t_{n+1}.

6.2.2 Sparse-grid interpolation for spatial discretization

To extend the time-stepping scheme in Eq. (6.2.6) to a fully-discrete scheme, we need to a spatial discretization scheme to approximate u^n as well as a quadrature rule to estimate the conditional expectation $\mathbb{E}[\cdot]$. In this work, we intend to use piecewise sparse grid interpolation to approximate $u^n(x)$ in $\mathcal{D}$. Specifically, since the terminal condition of the adjoint equation in Eq. (6.1.4) is discontinuous, we used hierarchical sparse grids with piecewise polynomials [7], which is easy to incorporate adaptivity to handle the discontinuity.

Hierarchical sparse grid interpolation

The one-dimensional hat function having support $[-1, 1]$ is defined by $\psi(x) = \max\{0\,,\, 1-|x|\}$ from which an arbitrary hat function with support $(x_{L,i} - \Delta x_L, x_{L,i} + \Delta x_L)$ can be generated by dilation and translation, i.e.,

$$\psi_{L,i}(x) := \psi\Big(\frac{x + 1 - i\Delta x_L}{\Delta x_L}\Big),$$

where L denotes the resolution level, $\Delta x_L = 2^{-L+1}$ for $L = 0, 1, \ldots,$ denotes the grid size of the level L grid for the interval $[-1, 1]$, and $x_{L,i} = i\,\Delta x_L - 1$ for $i = 0, 1, \ldots, 2^L$ denotes the grid points of that grid. The basis function $\psi_{L,i}(x)$ has local support and is centered at the grid point $x_{L,i}$ and the number of grid points in the level L grid is $2^L + 1$.

One can generalize the piecewise linear hierarchical polynomials to high-order hierarchical polynomials [7]. As shown in Fig 6.1, for $L \geq 0$, a piecewise linear polynomial $\psi_{L,i}(x)$ is

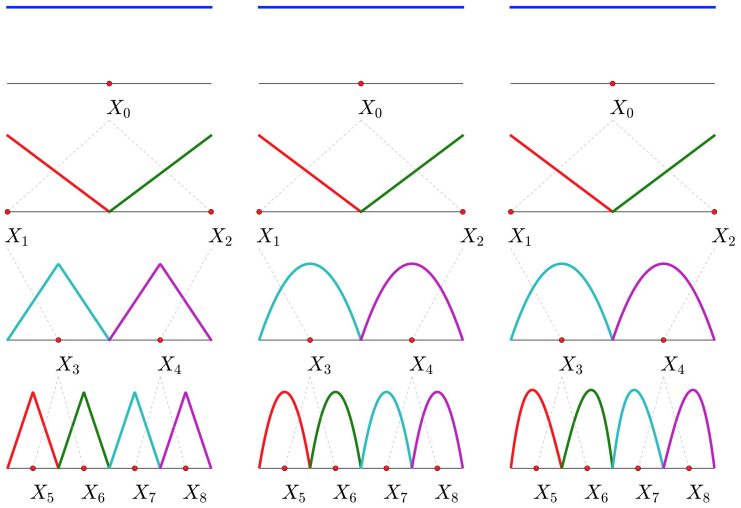

Figure 6.1: Left: linear hierarchical basis; Middle: quartic hierarchical basis where the quadratic polynomials appear since level 2; Right: cubic hierarchical basis where the cubic polynomials appear since level 3.

defined based on 3 supporting points, i.e., $x_{L,i}$ and its two ancestors that are also the endpoints of the support $[x_{L,i} - \Delta x_L, x_{L,i} + \Delta x_L]$. For q-th order polynomials, $q+1$ supporting points are needed to define a Lagrange interpolating polynomial. To do this, at each grid point $x_{L,i}$, additional ancestors outside of $[x_{L,i} - \Delta x_L, x_{L,i} + \Delta x_L]$ are borrowed to help build a higher-order Lagrange polynomial. Then, the desired high-order polynomial is defined by restricting the resulting polynomial to the support $[x_{L,i} - \Delta x_L, x_{L,i} + \Delta x_L]$. Fig 6.1 illustrates the linear, quadratic and cubic hierarchical bases, respectively.

With $Z = L^2(\mathcal{D})$, a sequence of subspaces $\{Z_L\}_{L=0}^{\infty}$ of Z of increasing dimension $2^L + 1$ can be defined as

$$Z_L = \text{span}\big\{\psi_{L,i}(x) \mid i = 0, 1, \ldots, 2^L\big\} \quad \text{for } L = 0, 1, \ldots.$$

Due to the nesting property of $\{Z_L\}_{l=0}^{\infty}$, we can define a sequence of hierarchical subspaces as $W_L = \text{span}\big\{\psi_{L,i}(x) \mid i \in B_L\big\}$, where $B_L = \big\{i \in \mathbb{N} \;\big|\; i = 1, 3, 5, \ldots, 2^L - 1\big\}$ for $L = 1, 2, \ldots$, such that $Z_L = Z_{L-1} \oplus W_L$ and $W_L = Z_L / \oplus_{L'=0}^{L-1} Z_{L'}$ for $L = 1, 2, \ldots$. Then,

the hierarchical subspace splitting of Z_L is given by

$$Z_L = Z_0 \oplus W_1 \oplus \cdots \oplus W_L \quad \text{for } L = 1, 2, \ldots.$$

The one-dimensional hierarchical polynomial basis can be extended to the N-dimensional domains using sparse tensorization. Specifically, the N-variate basis function $\psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{x})$ associated with the point $\boldsymbol{x}_{\mathbf{l},\mathbf{i}} = (x_{L_1,i_1}, \ldots, x_{L_N,i_N})$ is defined using tensor products, i.e., $\psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{x}) := \prod_{n=1}^{N} \psi_{L_n,i_n}(x_n)$, where $\{\psi_{L_n,i_n}(x_n)\}_{n=1}^{N}$ are the one-dimensional hierarchical polynomials associated with the point $x_{L_n,i_n} = i_n \Delta x_{L_n} - 1$ with $\Delta x_{L_n} = 2^{-L_n+1}$ and $\mathbf{l} = (L_1, \ldots, L_N)$ is a multi-index indicating the resolution level of the basis function. The N-dimensional hierarchical incremental subspace $W_{\mathbf{l}}$ is defined by

$$W_{\mathbf{l}} = \bigotimes_{n=1}^{N} W_{L_n} = \text{span}\left\{ \psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{x}) \mid \mathbf{i} \in B_{\mathbf{l}} \right\},$$

where the multi-index set $B_{\mathbf{l}}$ is given by

$$B_{\mathbf{l}} := \left\{ \mathbf{i} \in \mathbb{N}^N \;\middle|\; \begin{array}{lll} i_n \in \{1, 3, 5, \ldots, 2^{L_n} - 1\} & \text{for } n = 1, \ldots, N & \text{if } L_n > 0 \\ i_n \in \{0, 1\} & \text{for } n = 1, \ldots, N & \text{if } L_n = 0 \end{array} \right\}.$$

Similar to the one-dimensional case, a sequence of subspaces, again denoted by $\{Z_L\}_{L=0}^{\infty}$, of the space $Z := L^2(\mathcal{D})$ can be constructed as

$$Z_L = \bigoplus_{L'=0}^{L} W_{L'} = \bigoplus_{L'=0}^{L} \bigoplus_{\alpha(\mathbf{l}')=L'} W_{\mathbf{l}'},$$

where the key is how the mapping $\alpha(\mathbf{l})$ is defined because it defines the incremental subspaces $W_{L'} = \oplus_{\alpha(\mathbf{l}')=L'} W_{\mathbf{l}'}$. For example, $\alpha(\mathbf{l}) = |\mathbf{l}| = L_1 + \ldots + L_N$ leads to a standard isotropic sparse polynomial space.

The level L hierarchal sparse grid interpolant of the approximation $u^n(x)$ in Eq. (6.2.6) is defined by

$$\begin{aligned} u_L^n(\boldsymbol{x}) &:= \sum_{L'=0}^{L} \sum_{|\mathbf{l}'|=L'} (\Delta_{L'_1} \otimes \cdots \otimes \Delta_{L'_N}) u^n(\boldsymbol{x}) \\ &= u_{L-1}^n(\boldsymbol{x}) + \sum_{|\mathbf{l}'|=L} (\Delta_{L'_1} \otimes \cdots \otimes \Delta_{L'_N}) u^n(\boldsymbol{x}) \\ &= u_{L-1}^n(\boldsymbol{x}) + \sum_{|\mathbf{l}'|=L} \sum_{\mathbf{i} \in B_{\mathbf{l}'}} \left[u^n(\boldsymbol{x}_{\mathbf{l}',\mathbf{i}}) - u_{L'-1}^n(\boldsymbol{x}_{\mathbf{l}',\mathbf{i}}) \right] \psi_{\mathbf{l}',\mathbf{i}}(\boldsymbol{x}) \\ &= u_{L-1}^n(\boldsymbol{x}) + \sum_{|\mathbf{l}'|=L} \sum_{\mathbf{i} \in B_{\mathbf{l}'}} c_{\mathbf{l}',\mathbf{i}}\, \psi_{\mathbf{l}',\mathbf{i}}(\boldsymbol{x}), \end{aligned} \tag{6.2.7}$$

where $c_{\mathbf{l}',\mathbf{i}} = u^n(\boldsymbol{x}_{\mathbf{l}',\mathbf{i}}) - u_{L'-1}^n(\boldsymbol{x}_{\mathbf{l}',\mathbf{i}})$ is the multi-dimensional hierarchical surplus. This interpolant is a direct extension, via the Smolyak algorithm [42], of the one-dimensional hierarchical interpolant.

Numerical strategy for handling the boundary condition

After the sparse grid, denoted by $\mathcal{S}$, is constructed, the task becomes to estimate the right-hand side of Eq. (6.2.6) at all the interior sparse grid points $x_{\mathbf{i}} \in \mathcal{S} \cap \mathcal{D}$. The accuracy of such estimation also depends on how to deal with $\mathbb{P}(\tau^1_{t_n,x} \leq t_{n+1})$. It is known that $\mathbb{P}(\tau^1_{t_n,x} \leq t_{n+1}) \to 1$ as $x \to \partial\mathcal{D}_1$. In Lemma 5.1, we stated, in the one-dimensional case, that if b and σ are bounded functions, i.e.,

$$|b(t,x)| \leq \overline{b} \quad \text{and} \quad |\sigma(t,x)| \leq \overline{\sigma} \quad \text{for} \quad (t,x) \in [0,T] \times \mathcal{D},$$

with $0 \leq \overline{b}, \overline{\sigma} \leq +\infty$, and the starting point x in Eq. (6.2.5) is sufficiently far from the boundary $\partial\mathcal{D}$ satisfying $dist(x, \partial\mathcal{D}) \sim \mathcal{O}((\Delta t)^{1/2-\varepsilon})$ for any given constant $\varepsilon > 0$, then for sufficiently small Δt, it holds that

$$\mathbb{P}(\tau^1_{t_n,x} \leq t_{n+1}) \leq C(\Delta t)^{\varepsilon} \exp\left(-\frac{1}{(\Delta t)^{2\varepsilon}} \right), \tag{6.2.8}$$

where the constant $C > 0$ is independent of Δt.

Even though the estimate in Eq. (6.2.8) was proved for the one-dimensional case, it still provides a good insight to design an accurate numerical scheme. The key idea is to eliminate the destructive effect of $\mathbb{P}(\tau^1_{t_n,x} \leq t_{n+1})$ in the construction of the temporal-spatial discretization scheme by exploiting the estimate in Eq. (6.2.8). Specifically, we define the spatial mesh size Δx of the sparse grid is on the order of

$$\Delta x \sim \mathcal{O}\left((\Delta t)^{\frac{1}{2}-\varepsilon}\right),$$

such that, for each interior grid point $x_{\mathbf{i}}$, $u^n(x_{\mathbf{i}})$ in Eq. (6.2.6) can be approximated by

$$u^n(x_{\mathbf{i}}) \approx \mathbb{E}\left[u_L^{n+1}\left(X_{n+1}^{t_n,x_{\mathbf{i}}}\right)\right], \tag{6.2.9}$$

with the error on the order of $\mathcal{O}((\Delta t)^{\varepsilon}\exp(-1/(\Delta t)^{2\varepsilon}))$. The specific choice of Δx will be given in Section 6.2.3. Such strategy can avoid the approximation of the escape probability $\mathbb{P}(\tau^1_{t_n,x} \leq t_{n+1})$, but the trade-off is that we need to use higher order sparse grid interpolation to balance the total error.

6.2.3 Quadrature for the conditional expectation

The last piece of the puzzle is a quadrature rule for estimating the conditional expectations $\mathbb{E}\left[u_L^{n+1}\left(X_{n+1}^{t_n,x_{\mathbf{i}}}\right)\right]$ for $x_{\mathbf{i}} \in \mathcal{S} \cap \mathcal{D}$. Such expectation can be written as

$$\mathbb{E}\left[u_L^{n+1}(X_{n+1}^{t_n,x_{\mathbf{i}}})\right] = \int_{\mathbb{R}^d} u_L^{n+1}\left(x_{\mathbf{i}} + b(x_{\mathbf{i}})\Delta t + \sigma(x_{\mathbf{i}})\sqrt{\Delta t}\,\xi\right)\varrho(\xi)d\xi, \tag{6.2.10}$$

where $\xi := (\xi_1, \ldots, \xi_d)$ follows the d-dimensional standard normal distribution, i.e., $\rho(\eta) := \frac{1}{\pi^{d/2}}\exp(-(\sum_{\ell=1}^d \eta_\ell^2)/2)$. Thus, we utilized tensor-product Gauss-Hermite quadrature rule to approximate the expectation. Specifically, we denote by $\{w_j\}_{j=1}^J$ and $\{a_j\}_{j=1}^J$ the weights and

abscissae of the J-point tensor-product Gauss-Hermite rule, respectively. Then the approximation, denoted by $\widehat{\mathbb{E}}[u_L^{n+1}(X_{n+1}^{t_n,x_\mathbf{i}})]$ is defined by

$$u_\mathbf{i}^n = \widehat{\mathbb{E}}\left[u_L^{n+1}(X_{n+1}^{t_n,x_\mathbf{i}})\right] = \sum_{j=1}^{J} w_j \; u_L^{n+1}(q_{\mathbf{i}j}), \tag{6.2.11}$$

with

$$q_{\mathbf{i}j} := x_\mathbf{i} + b(x_\mathbf{i})\Delta t + \sigma(x_\mathbf{i})\sqrt{\Delta t}\, a_j \tag{6.2.12}$$

where ω_j is a product of the weights of the one-dimensional rule and a_j is a d-dimensional vector consisting of one-dimensional abscissae, respectively. When $u_L^{n+1}(\cdot)$ is sufficiently smooth, i.e., $\partial^{2J^*} u^{n+1}/\partial\eta_\ell^{2J^*}$ is bounded for $\ell = 1,\ldots,d$ with $J^* = J^{1/d}$, then the quadrature error can be bounded by [39]

$$\left|\widehat{\mathbb{E}}[u_L^{n+1}(X_{n+1}^{t_n,x_\mathbf{i}})] - \mathbb{E}[u_L^{n+1}(X_{n+1}^{t_n,x_\mathbf{i}})]\right| \le C\frac{J^*!}{2^{J^*}(2J^*)!}(\Delta t)^{J^*},$$

where the constant C is independent of J^* and Δt. Note that the factor $(\Delta t)^{J^*}$ comes from the $2J^*$-th order differentiation of the function u^{n+1} with respect to η_ℓ for $\ell = 1,\ldots,d$. Thus, to achieve first order global convergence rate $\mathcal{O}(\Delta t)$, we only need to use a total of $J^* = 27$ quadrature points. Sparse-grid Gauss-Hermite rule could be used to replace the tensor product rule when the dimension d is higher than 3. For the 3D runaway electron problem under consideration, we found that a level 1 sparse Gauss-Hermite rule with 7 quadrature points cannot provide sufficient accuracy, and a level 2 rule with 37 points is more expensive than the tensor product rule. Thus, we chose to use the tensor-product rule in this work.

By putting together all the components introduced in Section 6.2, we summarize our probabilistic scheme as follows:

Scheme 1 (The fully-discrete probabilistic scheme)**.** *Given the temporal spatial partition $\mathcal{T}\times\mathcal{S}$, the terminal condition $u^N(x_\mathbf{i})$ for $x_\mathbf{i}\in\mathcal{S}$, and the boundary condition $u^n(x_\mathbf{i})$ for $x_\mathbf{i}\in\mathcal{S}\cap\partial\mathcal{D}$. For $n = N-1,\ldots,0$, the approximation of $u(t_n,x)$ is constructed via the following steps:*

- *Step 1: generate quadrature abscissae* $\{q_{\mathbf{i}j}\}_{j=1}^{J}$*, in Eq.* (6.2.12)*, for* $x_{\mathbf{i}} \in \mathcal{S} \cap \mathcal{D}$*;*
- *Step 2: interpolate* $u_L^{n+1}(x)$ *at the quadrature abscissae to obtain* $\{u_L^{n+1}(\mathrm{q_{ij}})\}_{\mathrm{j=1}}^{\mathrm{J}}$*;*
- *Step 3: compute the coefficients* $u_{\mathbf{i}}^n$ *using the quadrature rule in Eq.* (6.2.11)*;*
- *Step 4: construct the interpolant* $u_L^n(\mathrm{x})$ *by substituting* $u_n^{\mathbf{i}}$ *into Eq.* (6.2.7)*.*

There are several advantages of our approach. First, the time-stepping scheme is totally explicit but absolutely stable, which has been rigorously proved in work [49, 53]. Second, the Feynman-Kac formula makes it natural to incorporate any sparse grid interpolation strategies to approximate the solution u without worrying about the discretization of the differential operator on the sparse grid. Third, it is easy to incorporate legacy codes for Monte Carlo based RE simulation into our scheme to compute runaway probability. It is a valuable feature because real-world RE models usually involve complex multiscale dynamics that are challenging to solve using PDE approaches.

6.3 Numerical examples

We tested our probabilistic scheme with two examples. The first example is to compute the escape probability of the standard Brownian motion. Since we know the analytical expression of the escape probability, this example is used to demonstrate the accuracy of our approach. The second example is to compute the runaway probability of the three-dimensional RE model given in Section 6.1. The sparse grid interpolation and adaptive refinement are implemented using the TASMANIAN toolbox [43].

6.3.1 Example 1: Escape probability of a Brownian motion

We consider the escape probability of a two-dimensional Brownian motion. The spatial domain $\mathcal{D}$ is set to $[0, 5] \times [0, 5]$ and the temporal domain is set to $t \in [0, 2]$ with $T_{\max} = 2$. The escape

probability $P(t,x)$ can be obtained by solving the standard heat equation

$$\begin{aligned} \frac{\partial u}{\partial t} + \frac{1}{2}\Delta u = 0, \quad & (t,x) \in [0, T_{\max}] \times \mathcal{D}, \\ u(t,x) = 1, \quad & (t,x) \in [0, T_{\max}] \times \partial\mathcal{D}, \\ u(T_{\max}, x) = 0, \quad & x \in \mathcal{D}. \end{aligned} \tag{6.3.1}$$

The exact solution is given by

$$u(t,x) = 1 + \sum_{m=1}^{\infty}\sum_{n=1}^{\infty} A_{mn} \sin(\mu_m x_1) \sin(\nu_n x_2) e^{-\lambda_{mn}^2 t},$$

where $\mu_m = \frac{m\pi}{5}$, $\nu_n = \frac{n\pi}{5}$, $\lambda = \sqrt{\frac{1}{2(\mu_m^2+\nu_n^2)}}$. The escape probability $P(t,x)$ can be obtained by substituting u into Eq. (6.1.6), i.e., $P(t,x) = u(T_{\max} - t, x)$.

We intend to demonstrate that our scheme can achieve first-order convergence $\mathcal{O}(\Delta t)$ when probably choosing the sparse grid resolution, i.e., the level L. To this end, we use compare three cases, i.e.,

(a) Hierarchical cubic basis with $\Delta x \sim \mathcal{O}(\sqrt{\Delta t})$,

(b) Hierarchical linear basis with $\Delta x \sim \mathcal{O}(\sqrt{\Delta t})$,

(c) Hierarchical cubic basis with $\Delta x \sim \mathcal{O}(\Delta t)$,

where Δx denotes the mesh size of the one-dimensional rule for building the sparse grids. The error of the three cases are shown in Fig 6.2 and Fig 6.3. As expected, when setting $\Delta x \sim \mathcal{O}(\sqrt{\Delta t})$, the escape probability $\mathbb{P}(\tau_{t_n,x} \leq t_{n+1})$ for any interior grid point is on the order of $\mathcal{O}((\Delta t)^{\varepsilon} \exp(-1/(\Delta t)^{2\varepsilon}))$, such that neglecting $\mathbb{P}(\tau_{t_n,x} \leq t_{n+1})$ will asymptotically not affect the first-order convergence w.r.t. Δt. On the other hand, we need to use high-order hierarchical basis to achieve comparable accuracy in spatial approximation. It is shown in Fig 6.2 that the use of the hierarchical cubic polynomials, introduced in [7], provides sufficient accuracy to achieve a global convergence rate $\mathcal{O}(\Delta t)$. In comparison, when using linear basis with $\Delta x \sim \mathcal{O}(\sqrt{\Delta t})$, the linear sparse-grid interpolation only provides $\mathcal{O}((\Delta x)^2) = \mathcal{O}(\Delta t)$ local convergence, such that our scheme dose not converge globally. From the second row of

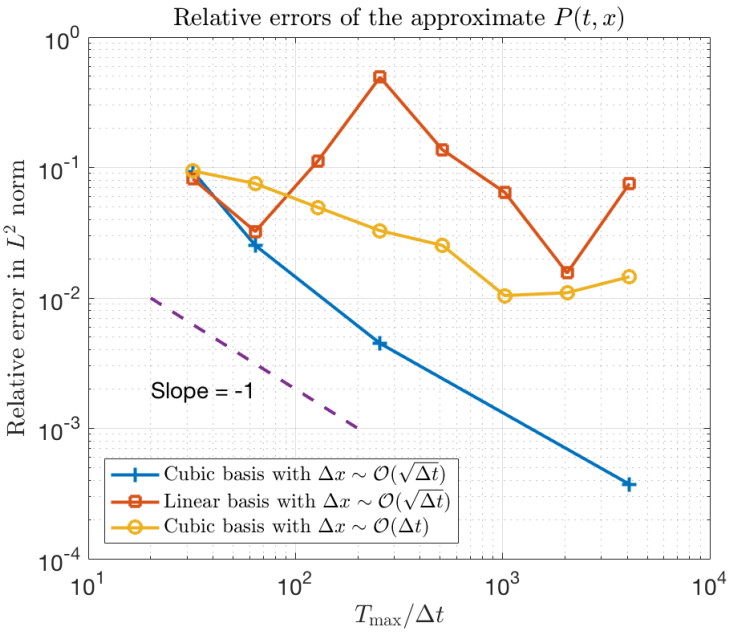

Figure 6.2: The relative error of the approximate escape probability of the standard Brownian motion.

Fig 6.3, we can see that big errors are generated around the boundary of the spatial domain and gradually propagate to the middle region of the domain. Similar phenomenon appears when setting $\Delta x \sim \mathcal{O}(\Delta t)$. In this case, since the interior grid points near the boundary are so close to the boundary that neglecting the escape probability $\mathbb{P}\left(\tau_{t_n,x} \leq t_{n+1}\right)$ leads to significant additional error. This is the reason why big errors are firstly generated near the boundary (i.e., $t = 0.5$), and then propagate to the center.

6.3.2 The runaway probability of the three-dimensional RE model

Here we test our method using the 3D runaway electron model given in Eq. (6.1.1) with the following parameters:

$$T_{\max} = 120, \quad p_{\min} = 2, \quad p_{\max} = 50, \quad Z = 1, \quad \tau = 10^5, \quad \delta = 0.042,$$
$$E = 0.3, \quad \bar{v}_{ee} = 1, \quad \bar{v}_T = 1, \quad D_0 = 0.003, \quad \Delta p = 20.$$

Unlike the example about Brownian motion, where the discontinuous terminal condition is smoothed out very fast, the evolution of the runaway probability P_{RE} is more convection-dominant. As such, we utilized adaptive sparse grids to capture the movement of the sharp

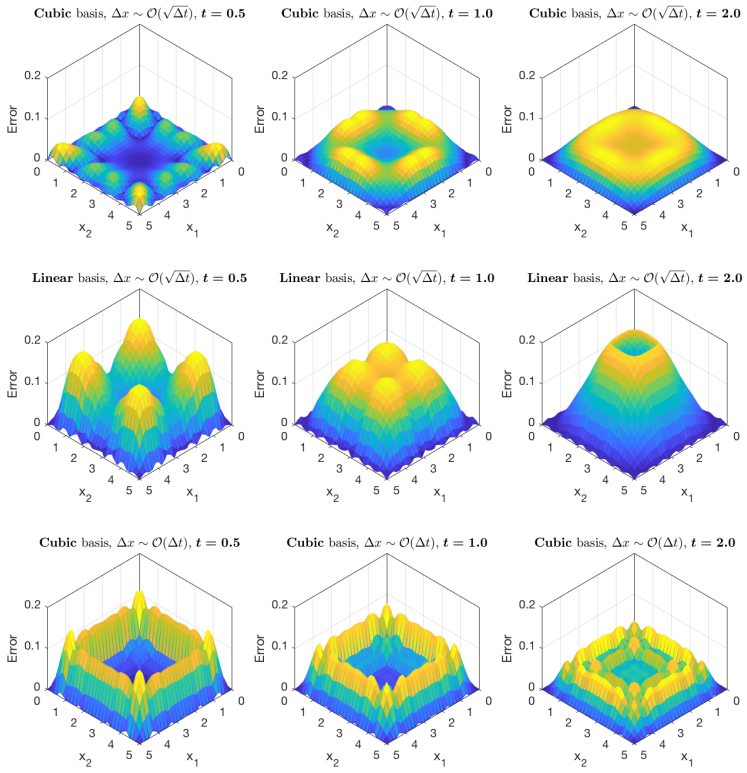

Figure 6.3: The error distribution in the spatial domain $[0, 5] \times [0, 5]$ for the three cases considered in Fig 6.2 for $t = 0.5, 1.0$ and 2.0.

transition layer. The standard refinement approach is to construct an initial grid using all point up to some coarse level, then consider the hierarchical surplus coefficients, e.g., the coefficients of the basis functions, which are estimates of the local approximation error or how much correction is introduced by the associated node. The coarse grid is enriched by adding the children of nodes with large coefficients ignoring all other points, the coefficients of the new nodes are computed and the children of the children are added. The refinement process is repeated until all coefficients fall below some desired tolerance. However, the standard refinement process may stagnate when dealing with functions with localized sharp behavior which results in non-monotonic decay of the coefficients (in the pre-asymptotic regime). In such a scenario, a node located in the sharp region could have parents that all belong to the smooth region, then the node is excluded from the grid. Even if descendants of the node converge on the sharp region (following paths through other parents), the children have restricted support and cannot compensate for the missing parent. A common remedy for this problem is to recursively add all parents of all nodes, but this not desirable as it includes many nodes with small coefficients

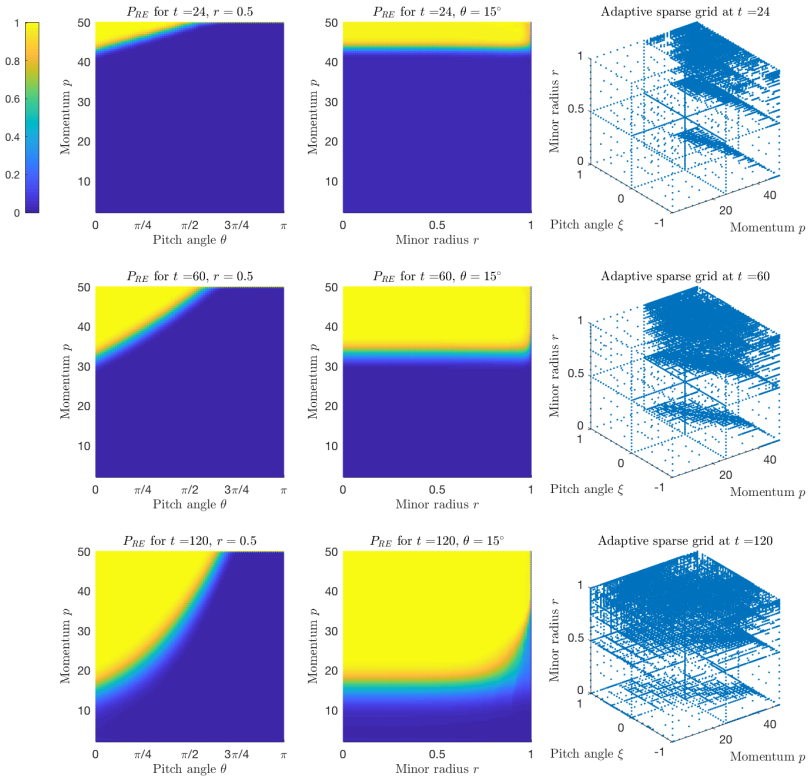

Figure 6.4: Cross sections of the runaway probability P_{RE} as well as the corresponding adaptive sparse grids at three time instances $t = 24, 60$ and 120.

which would have been ignored in the classic refinement. Therefore, we utilized a more flexible refinement procedure that considers both children and parents but is still restricted attention to the immediate relatives to avoid oversampling. The parents' selective refinement procedure is described in details in [44] and it is implemented in the TASMANIAN open-source library [43, 45].

The evolution of the runaway probability P_{RE} as well as the corresponding adaptive sparse grids are shown in Fig 6.4. The runaway boundary is at $p = p_{\max} = 50$ and the main reason of the an electron running away is the electric field acceleration, i.e., the term $E\xi$ in the drift of the momentum dynamics. The factor $\xi = \cos(\theta)$ in $E\xi$ determines that the electrons with small pitch angles will runaway sooner than the electrons with large pitch angles, which is consistent with the simulation results in Fig 6.4. There are two sharp transition layers in this simulation, i.e., the transition between the runaway and the non-runaway regions, and the boundary layer around $r = 1$ due to small diffusion effect in the minor radius direction. In our simulation, we used the 6-level sparse grid as the initial grid and gradually refine it with the tolerance being

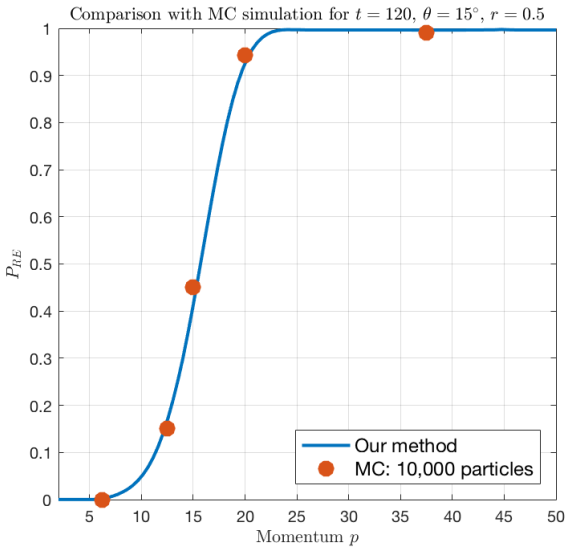

Figure 6.5: Comparison between the our approach and the direct MC for pitch angle $\theta = 15°$ and minor radius $r = 0.5$.

0.001. As expected, the adaptive refinement accurately captured the irregular behaviors. In addition, since the analytical expression of P_{RE} is unknown, we tested the accuracy of our approach by comparing with the direct Monte Carlo method for computing P_{RE} at a few locations in the phase space, and the result is shown in Fig 6.5.

6.4 Concluding remarks

We proposed a sparse-grid probabilistic scheme for the accurate and efficient computation of the time-dependent probability of runaway. The method is based on the direct numerical solution of the Feynman-Kac formula. At each time step, the algorithm reduces to the computation of an integral involving the previously computed probability of runaway and the Gaussian propagator. Sparse-grid interpolation is utilized to recover the runaway probability function as well as evaluate the quadrature points for estimating the conditional expectation in the Feynman-Kac formulation. Even though the advantages of sparse grids have already been revealed in solving the three-dimensional RE problem, we will extend our approach to higher dimensional RE problems involving more complicated dynamics. For example, an important RE model to be resolved is to incorporate the relativistic guiding center equations of electron motion into the RE scenario. In this case, the deterministic dynamic of the guiding center motion is six order

of magnitudes smaller than the collisional dynamics, which presents a significant challenge to the design of numerical schemes.

Chapter 7

Summary and Future Work

7.1 Summary

The main contribution of this dissertation is to develop and analyze the numerical solutions of partial integro-differential equations with the second-order integral-differential operator and the fractional Laplacian operator. For the PIDEs in unbounded domains, we constructed the probabilistic numerical schemes of the solution and carried out relative error analysis, the first-order temporal convergence rate and the high order spatial convergence rate. In high dimensional space, domains are meshed by Delaunay triangulation and sparse grid quadrature rule is used to approximate high order spatial integrals.

For the partial differential equations with fractional Laplacian operator, we first approximate the SαS process by Lévy process that consists of Brownian motion and Poisson process with finite jump amplitude based on Gaussian approximation studied by [4, 8]. Then we construct the corresponding PIDE, which aims to approximate the goal fractional Laplacian equation. Based on the probabilistic numerical schemes introduced in Chapter 3, we can numerically approach the solution of fractional Laplacian equation.

For the PIDEs with volume constraints, we first derived the probabilistic representation of the solution. To get the numerical schemes, we divided all possible paths of the underlying stochastic process $\overleftarrow{X}_t$ into two subsets base on the exit time and proved that the possibility of the event that $\overleftarrow{X}_t$ exits the bounded domain can be controlled by $\mathcal{O}(\Delta t)^2$ as the starting point of $\overleftarrow{X}_t$ is sufficiently far from the boundary of the interior domain. Based on that fact, a full discrete algorithm is constructed.

We proposed a sparse-grid probabilistic scheme for the accurate and efficient computation of the time-dependent probability of runaway. The method is based on the direct numerical solution of the Feynman-Kac formula. At each time step, the algorithm reduces to the computation of an integral involving the previously computed probability of runaway and the Gaussian propagator. Sparse-grid interpolation is utilized to recover the runaway probability function as well as evaluate the quadrature points for estimating the conditional expectation in the Feynman-Kac formulation.

7.2 Future work

Even the sparse grid integration method was applied in our probabilistic numerical schemes for solving the partial integro-differential equations. The computation cost is still huge when the dimensionality is higher than three. We plan to incorporate the sparse gird interpolation rule with our numerical schemes to alleviate the curse of dimensionality. There is one more improvement we can make is to estimate the relative error of the temporal-spatial discrete scheme under the norm L_∞. The technical issue is how to deal with the Lebesgue constant of high order interpolation.

Another possible future project is the construct the probabilistic scheme for the nonlocal diffusion equation with fractional Laplacian operator using the Fourier-cosine series. We already know the relationship between the fraction Laplacian operator and the α stable process. The key idea is to construct an approximation of the expectation in the Fourier space.

Even though the advantages of sparse grids have already been revealed in solving the three-dimensional RE problem, we will extend our approach to higher dimensional RE problems involving more complicated dynamics. For example, an important RE model to be resolved is to incorporate the relativistic guiding center equations of electron motion into the RE scenario. In this case, the deterministic dynamic of the guiding center motion is six orders of magnitudes smaller than the collisional dynamics, which presents a significant challenge to the design of numerical schemes.

References

[1] M. Abramowitz and I. Stegun. Handbook of mathematical functions with formulas, graphs and mathematical tables, reprint of the 1972 edition. *Reprint of the 1972 edition. Dover Publications, Inc., New York*, 1992.

[2] D. Applebaum. *Lèvy processes and stochastic calculus*. Cambridge university press, 2009.

[3] M. Askari and H. Adibi. Meshless method for the numerical soluton of the Fokker-Planck equation. *Ain Shams Engineering Journal*, 6:1211–1216, 2015.

[4] S. Asmussen and J. Rosin$\acute{n}$ski. Approximation of Small Jumps of L$\acute{e}$vy Processes with a View Towards Simulation. *J. Appl. Probab*, 38:482–493, 2001.

[5] G. Barles, R. Buckdahn, and E. Pardoux. Backward stochastic differential equations and integral-partial differential equations. *Stochastic and Stochastic Reports*, 60:57–83, 1997.

[6] B. Bouchard and S. Menozzi. Strong approximation of BSDEs in a domain. *Bernoulli*, 15(4):1117–1147, 2009.

[7] H.-J. Bungartz and M. Griebel. Sparse grids. *Acta Numerica*, 13:1–123, June 2004.

[8] S. Cohen and J. Rosi$\acute{n}$ski. Gaussian approximation of multivariate L$\acute{e}$vy processes with applications to simulation of tempered stable processes. *Bernoulli Society for Mathematical Statistics and Probability*, 13:195–210, 2007.

[9] R. Darling and R. Pardoux. Backward SDE with random terminal time and applications to semilinear elliptic PDE. *The annals of Probability*, 25:3:1135–1159, 1997.

[10] Q. Du, Z. Huang, and R. Lehoucq. Nonlocal convection-diffusion volume-constrained problems and jump processes. *Discrete Contin. Dyn. Syst. Ser. B*, 19:373–389, 2014.

[11] Q. Du, L. Ju, L. Tian, and K. Zhou. A posteriori error analysis of finite element method for linear nonlocal diffusion and peridynamic models. *Math. Comp*, 82:1889–1922, 2013.

[12] E. B. Dynkin. *Markov Processes*. Springer-Verlag Berlin Heidelberg, 1965.

[13] N. W. Eidietis, N. Commaux, E. M. Hollmann, D. A. Humphreys, T. C. Jernigan, R. A. Moyer, E. J. Strait, M. A. VanZeeland, J. C. Wesley, and J. H. Yu. Control of post-disruption runaway electron beams in diii-d. *Physics of Plasmas*, 19(5):056109, 2012.

[14] G. FISHMAN. Monte carlo: Concepts, algorithms, applications. *Springer-Verlag*, 1996.

[15] T. Gerstner and M. Griebel. Dimension–Adaptive Tensor–Product Quadrature. *Computing*, 71(1):65–87, Aug. 2003.

[16] J. Goodman and J. O' Rourke. *Handbook of Discrete and Computational Geometry.* CRC Press, 1997.

[17] M. Griebel. Adaptive sparse grid multilevel methods for elliptic PDEs based on finite differences. *Computing*, 61(2):151–179, June 1998.

[18] M. Gunzburger, G. Zhang, and W. Zhao. A sparse grid method for multi-dimensional backward stochastic differential equations. *Journal of Computational Mathematics*, 31:3:221–248, 2013.

[19] M. D. Gunzburger, C. G. Webster, and G. Zhang. Stochastic finite element methods for partial differential equations with inout data. *Acta Numerica*, 23:521–625, 2014.

[20] F. B. Hanson. Applied Stochastic Processes and Control for Jump-Diffusions: Modeling, Analysis and Computation. *Society for Industrial and Applied Mathematics*, 2007.

[21] E. M. Hollmann, P. B. Aleynikov, T. Flp, D. A. Humphreys, V. A. Izzo, M. Lehnen, V. E. Lukash, G. Papp, G. Pautasso, F. Saint-Laurent, and J. A. Snipes. Status of research toward the iter disruption mitigation system. *Physics of Plasmas*, 22(2):021802, 2015.

[22] A. Janicki and A. Weron. Can one see α-stable variables and processes. *Statistical Science*, 9:109–126, Feb, 1994.

[23] A. Klimke and B. Wohlmuth. Algorithm 847: Spinterp: piecewise multilinear hierarchical sparse grid interpolation in matlab. *ACM Transactions on Mathematical Software (TOMS)*, 31(4):561–579, 2005.

[24] P. Kloeden and E. Platen. *Numerical Solution of Stochastic Differential Equations*. Stochastic Modeling and Applied Probability, 1992.

[25] C. Liu, D. P. Brennan, A. Bhattacharjee, and A. H. Boozer. Adjoint fokker-planck equation and runaway electron dynamics. *Physics of Plasmas*, 23(1):010702, 2016.

[26] Q. Marie-Claire and S. Agnès. BSDEs with jumps, optimization and applications to dynamic risk measures. *Stochastic Processes and their Applications*, 123:8:3328–3357, 2013.

[27] E. Marsch. Kinetic physics of the solar corona and solar wind. *Living Reviews in Solar Physics*, 3(1):1, Jul 2006.

[28] J. R. Martn-Sols, A. Loarte, and M. Lehnen. Runaway electron dynamics in tokamak plasmas with high impurity content. *Physics of Plasmas*, 22(9):092512, 2015.

[29] R. Metzler and L. Klafter. The random walk's guide to anomalous diffusion: a fractional dynamics approach. *Phys. Rep*, 339:1–77, 2000.

[30] G. Milstein and M. Tretyakov. Numerical solution of the Dirichlet problem for nonlinear parabolic equations by a probabilistic approach. *IMA Journal of Numerical Analysis*, 21(4):887–917, 2001.

[31] G. Milstein and M. Tretyakov. The simplest random walks for the Dirichlet problem. *Theory of Probability and its applications*, 47(1):53–68, 2003.

[32] G. Milstein and M. Tretyakov. Numerical algorithms for forward-backward stochastic differential equations. *SIAM journal on Scientific Computing*, 28(2):561–582, 2006.

[33] G. P. Nikishkov. *Introduction to the finite element method*. Lecture Notes. University of Aizu, 2004.

[34] E. Pardoux. Backward stochastic differential equations and viscosity solution of systems of semilinear parabolic and elliptic PDEs of second order. *Stochastic Analysis and Related Topics*, VI:79–127, 1998.

[35] E. Pardoux and S. Peng. Adapted solution of a backward stochastic differential equation. *System and Control Letters*, 14(1):55–61, 1990.

[36] M. Pavel. *Delaunay Triangulation in 3D*. PhD thesis, University of West Bohemia in Pilsen, 2002.

[37] D. Pflüger, B. Peherstorfer, and H.-J. Bungartz. Spatially adaptive sparse grids for high-dimensional data-driven problems. *Journal of Complexity*, 26(5):508–522, 2010.

[38] E. Platen and N. Bruti-Liberati. *Numerical Solution of Stochastic Differential Equations with Jumps in Finance*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[39] A. Quarteroni, R. Sacco, and F. Saleri. *Numerical Mathematics*, volume 332. Springer Science Business Media &, 2007.

[40] M. Rosenbluth and S. Putvinski. Theory for avalanche of runaway electrons in tokamaks. *Nuclear Fusion*, 37(10):1355–1362, oct 1997.

[41] M. Royer. Backward stochastic differential equations with jumps and related non-linear expectations. *Stochastic processes and their Applications*, 116:1358–1376, 2006.

[42] S. Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. *Dokl. Akad. Nauk SSSR*, 4:240–243, 1963.

[43] M. Stoyanov. User manual: Tasmanian sparse grids. Technical Report ORNL/TM-2015/596, Oak Ridge National Laboratory, One Bethel Valley Road, Oak Ridge, TN, 2015.

[44] M. Stoyanov. Adaptive sparse grid construction in a context of local anisotropy and multiple hierarchical parents. In *Sparse Grids and Applications-Miami 2016*, pages 175–199. Springer, 2018.

[45] M. Stoyanov, D. Lebrun-Grandie, J. Burkardt, and D. Munster. Tasmanian, November 2013.

[46] E. Suli and D. F. Mayers. An Introduction to Numerical Analysis. *Cambridge University Press, Cambridge*, 2003.

[47] G. Thomas and G. Michael. Numerical integration using sparse grids. *Numerical Algorithms*, 18:209–232, 1998.

[48] J. Yang, G. Zhang, and W. Zhao. A first-order numerical scheme for forward-backward stochastic differential equations in bounded domains. *Journal of Computational Mathematics*, pages 1–12, 2016.

[49] G. Zhang and D. Del-Castillo-Negrete. A backward Monte-Carlo method for time-dependent runaway electron simulations. *Physics of Plasmas*, 24(9):092511, Sept. 2017.

[50] G. Zhang, W. Zhao, C. Webster, and M. Gunzburger. Numerical methods for a class of nonlocal diffusion problems with the use of backward SDEs. *Computers and Mathematics with Applications*, 71:2479–2496, 2016.

[51] W. Zhao, L. Chen, and S. Peng. A new kind of accurate numerical method for backward stochastic differential equations. *SIAM J. Sci. Comput*, 28:1563–1581, 2006.

[52] W. Zhao, Y. Li, and G. Zhang. A generalized θ-scheme for solving backward stochastic differential equations. *Discrete Contin. Dyn. Syst*, Ser. B 117:1585–1603, 2012.

[53] W. Zhao, G. Zhang, and L. Ju. A stable multistep scheme for solving backward stochastic differential equations. *SIAM Journal on Numerical Analysis*, 48(4):1369–1394, 2010.

[54] W. Zhao, W. Zhang, and G. Zhang. Second-order numerical schemes for decoupled forward-backward stochastic differential equations with jumps. *Journal of Computational Mathematics*, 35(2):213–344, 2017.

[55] O. Zienkiewicz and R. Taylor. *The FiniteElement Method, 4th ED*. New York: McGrawHill, 1989.

Appendix

7.3 Construction of backward filtration

Let $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t_n \le t \le t_{n+1}}, \mathbb{P})$ be a *backward* stochastic basis satisfying the hypotheses of completeness, i.e., $\mathcal{F}_{t_{n+1}}$ contains all the sets of $\mathbb{P}$-measure zero and possesses left continuity, i.e., $\mathcal{F}_t = \mathcal{F}_{t-}$. The filtration $\{\mathcal{F}_t\}_{t_n \le t \le t_{n+1}}$ is assumed to be generated by two mutually independent processes, i.e., one d-dimensional *backward* Brownian motion $\overleftarrow{W}_t$ and one Poisson random measure $\mu(A, t)$ on $E \times [t_n, t_{n+1}]$. Under the probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t_n \le t \le t_{n+1}}, \mathbb{P})$, with start point (t_{n+1}, x), stochastic process $\{\overleftarrow{W}_t - \overleftarrow{W}_{t_{n+1}}\}_{t_n \le t \le t_{n+1}}$ is a martingale and

$$\begin{aligned}
\mathbb{E}\left[\left(\int_{t_{n+1}}^{t_n} d\overleftarrow{W}_t\right)^2 \middle| X_{t_{n+1}} = x\right] =& \mathbb{E}\left[\left(\overleftarrow{W}_{t_n} - \overleftarrow{W}_{t_{n+1}}\right)^2 \mid X_{t_{n+1}} = x\right] \\
=& \mathbb{E}\left[\overleftarrow{W}^2_{t_n} + \overleftarrow{W}^2_{t_{n+1}} - 2\overleftarrow{W}_{t_{n+1}}\overleftarrow{W}_{t_n} \mid X_{t_{n+1}} = x\right] \\
=& (t_n - t_{n+1}) + (t_{n+1} - t_{n+1}) - 2(t_{n+1} - t_{n+1}) \\
=& -\Delta t.
\end{aligned} \tag{7.3.1}$$

The *backward* Poisson process N_t, i.e., the number of jump of $\overleftarrow{X}_t$ with intensity λ satisfies $N_{t_n} \ge N_{t_{n+1}}$ and $\mathbb{E}[N_t] = \lambda(t_{n+1} - t)$ where $t \in [t_n, t_{n+1}]$. The compensator of μ and the resulting compensated Poisson random measure are denoted by $\nu(de, dt) = \lambda(de)(-dt)$ and $\tilde{\mu}(de, dt) = \mu(de, dt) - \nu(de, dt)$, respectively, such that $\tilde{\mu}(A, t)$ is a martingale with stationary independent increments.

7.4 Ito formula for backward stochastic differential equation

We consider the *backward* SDE in one dimensional space,

$$\overleftarrow{X}_s^{t_{n+1},x} = x - \int_{t_{n+1}}^{s} \beta(t, \overleftarrow{X}_t^{t_{n+1},x})dt + \int_{t_{n+1}}^{s} \sigma(t, \overleftarrow{X}_t^{t_{n+1},x})d\overleftarrow{W}_t + \int_{t_{n+1}}^{s}\int_E c(t,e)\tilde{\mu}(de,dt). \tag{7.4.1}$$

Taking Itô formula to $u(s, \overleftarrow{X}_s^{t_{n+1},x})$, we have

$$\begin{aligned}
u(s, \overleftarrow{X}_s^{t_{n+1},x}) =& u(t_{n+1},x) + \int_{t_{n+1}}^{s} \frac{\partial u}{\partial t}dt + \int_{t_{n+1}}^{s} (-\beta)\frac{\partial u}{\partial x}(t, \overleftarrow{X}_t^{t_{n+1},x})dt \\
&+ \int_{t_{n+1}}^{s}\int_E \left[u\left(t, \overleftarrow{X}_t^{t_{n+1},x} + c(t,e)\right) - u(t, \overleftarrow{X}_t^{t_{n+1},x}) - c(t,e)\frac{\partial u}{\partial x}\right]\lambda(de)(-dt) \\
&+ \int_{t_{n+1}}^{s} \frac{\partial u}{\partial x}\sigma(t, \overleftarrow{X}_t^{t_{n+1},x})d\overleftarrow{W}_t + \int_{t_{n+1}}^{s} \frac{1}{2}\sigma^2\frac{\partial^2 u}{\partial x^2}(t, \overleftarrow{X}_t^{t_{n+1},x})(d\overleftarrow{W}_t)^2 \\
&+ \int_{t_{n+1}}^{s}\int_E \left[u\left(t, \overleftarrow{X}_t^{t_{n+1},x} + c(t,e)\right) - u(t, \overleftarrow{X}_t^{t_{n+1},x})\right]\tilde{\mu}(de,dt) \\
=& u(t_{n+1},x) + \int_{t_{n+1}}^{s} \left(\frac{\partial u}{\partial t} - \mathcal{L}^*[u]\left(t, \overleftarrow{X}_t^{t_{n+1},x}\right)\right)dt \\
&+ \int_{t_{n+1}}^{s} \frac{\partial u}{\partial x}\sigma(t, \overleftarrow{X}_t^{t_{n+1},x})d\overleftarrow{W}_t \\
&+ \int_{t_{n+1}}^{s} [u(t, \overleftarrow{X}_t^{t_{n+1},x} + c(t,e)) - u(t, \overleftarrow{X}_t^{t_{n+1},x})]\tilde{\mu}(de,dt).
\end{aligned} \tag{7.4.2}$$

Taking mathematical expectation $\mathbb{E}^x_{t_{n+1}}[\cdot]$ on both sides of (7.4.2), due to the martingale property,

$$\int_{t_{n+1}}^{s} \frac{\partial u}{\partial x}\sigma(t, \overleftarrow{X}_t^{t_{n+1},x})d\overleftarrow{W}_t = 0,$$

$$\int_{t_{n+1}}^{s} [u(t, X_t^{t_{n+1},x} + c(t,e)) - u(t, X_t^{t_{n+1},x})]\tilde{\mu}(de,dt) = 0.$$

Thus the probabilistic representation of $u(t_{n+1},x)$ is given as

$$u(t_{n+1},x) = \mathbb{E}^x_{t_{n+1}}[u(t_n, \overleftarrow{X}_{t_n}^{t_{n+1},x})] - \int_{t_{n+1}}^{t_n} \mathbb{E}^x_{t_{n+1}}\left[g(t, \overleftarrow{X}_t^{t_{n+1},x}, u(t, \overleftarrow{X}_t^{t_{n+1},x})\right]dt, \tag{7.4.3}$$

7.5 River models in numerical examples

• River Model in Example 4 for testing the convergence rate with respect to Δt.

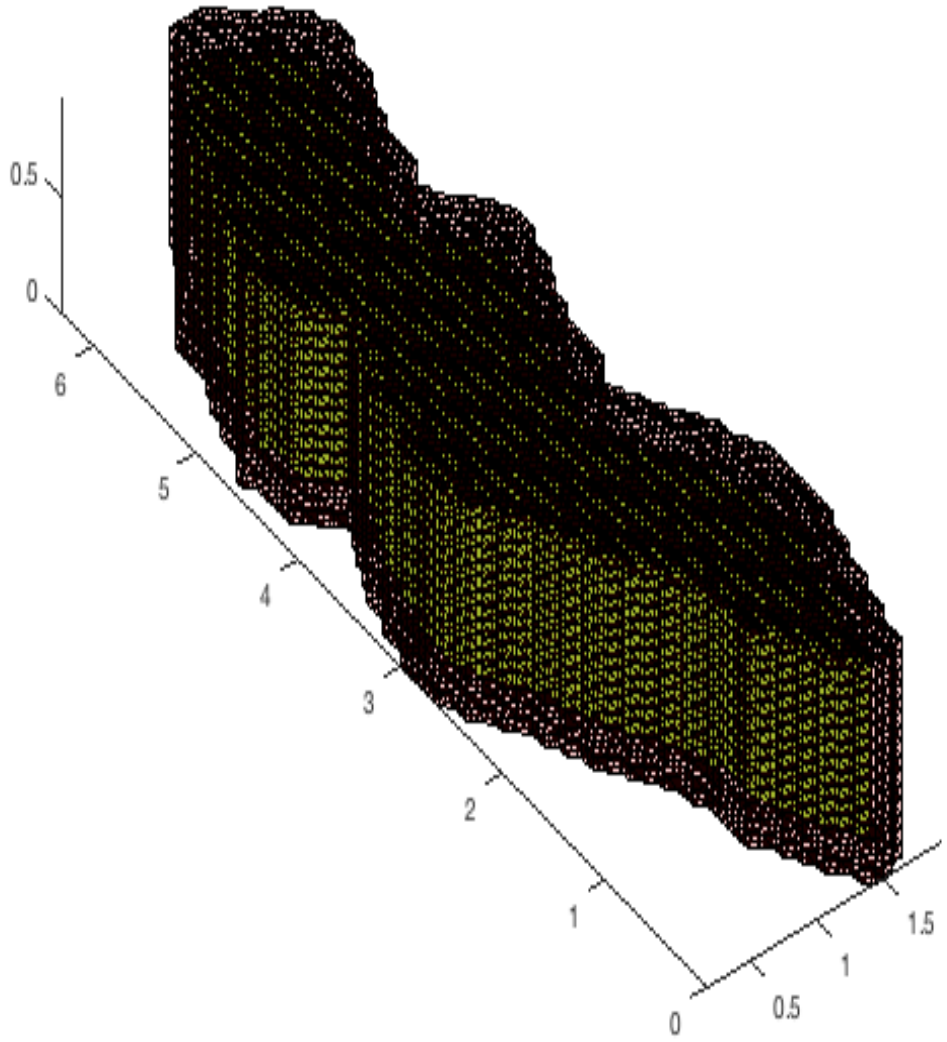

Figure 7.1: River Model 1

• River Model for testing the convergence rate with respect to Δx.

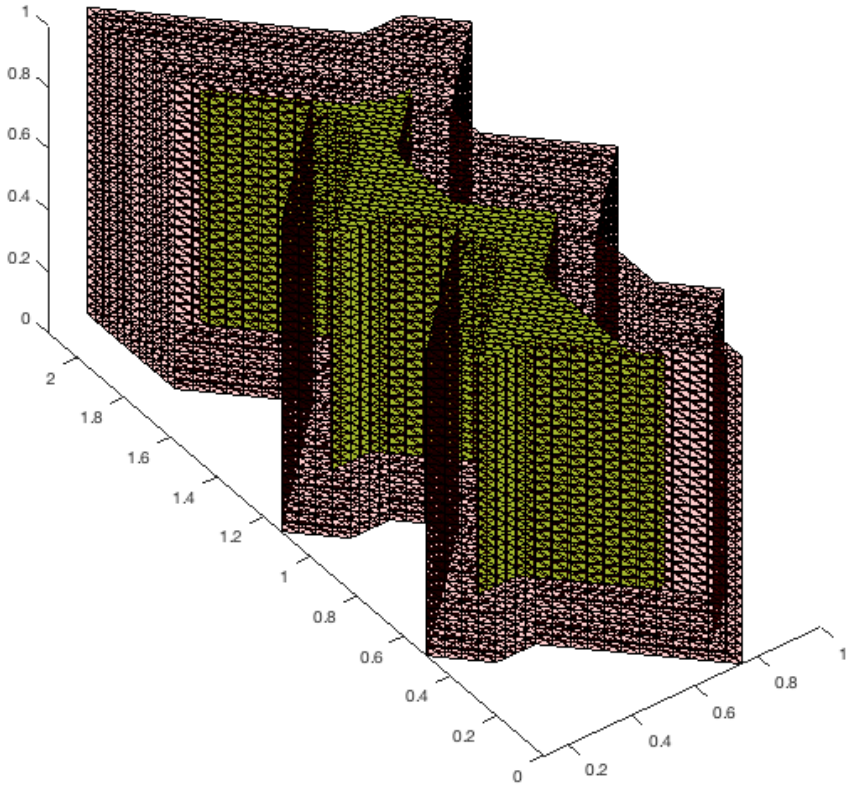

Figure 7.2: River Model 2

• River Model in Example 2.

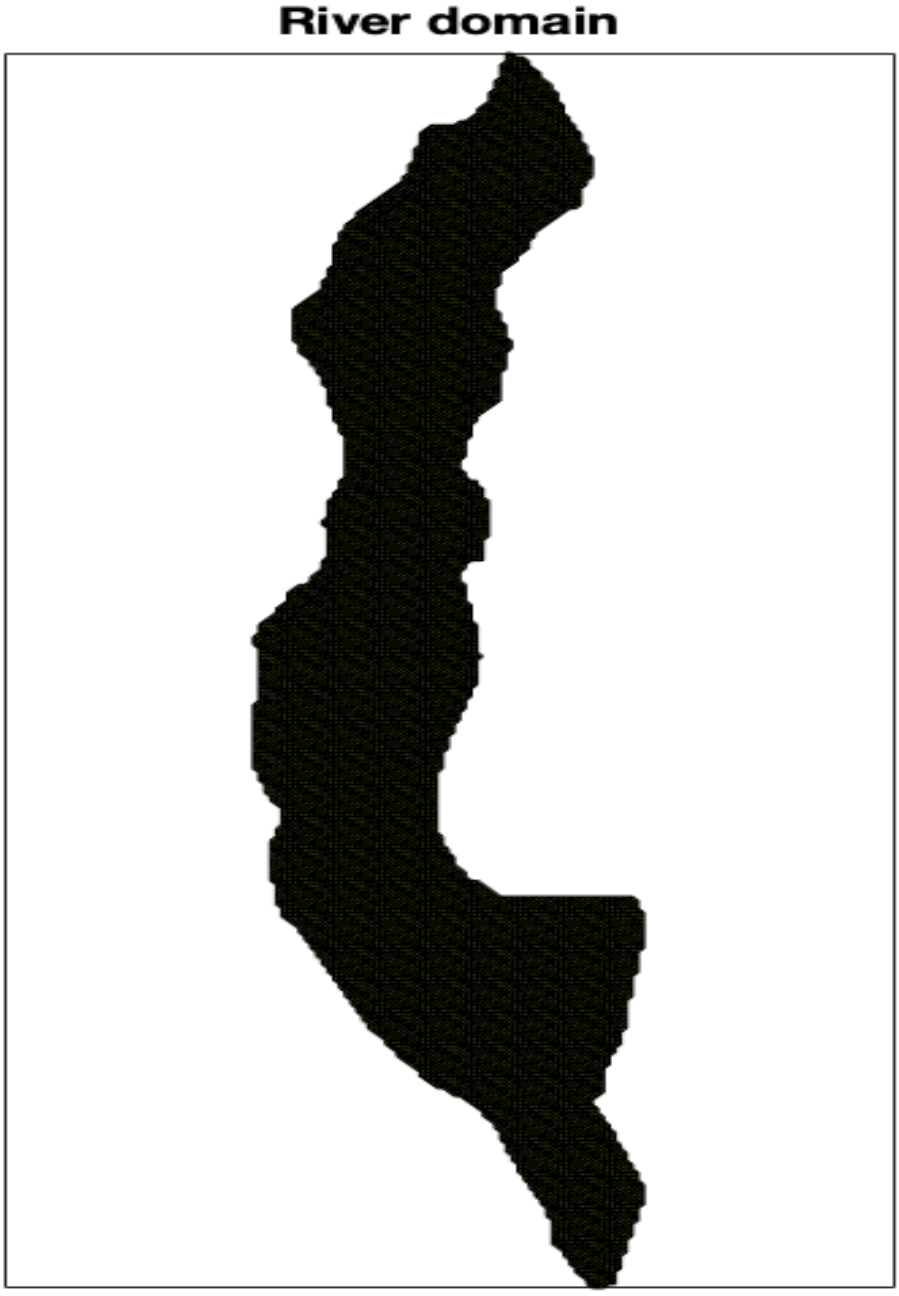

Figure 7.3: River Model 3