Fitted finite volume method for pricing options under jump diffusion process

Dumisani Zawokane (dumisani@aims.ac.za)
African Institute for Mathematical Sciences (AIMS)

Supervised by: Professor Phillip Mashele
North-West University & African Institute for Mathematical Sciences (AIMS), South Africa

22 September 2020

Submitted in partial fulfillment of a structured masters degree at AIMS South Africa

Abstract

In this essay, we derive the jump-diffusion process to obtain the pricing model. Firstly, we evaluate the underlying stock that has been affected by the jump-diffusion process. Then we develop the numerical scheme to work out the partial integro-differential equation (PIDE). We use the fitted volume method to discretize the spatial domain and Crank-Nicolson time discretization to obtain the matrix then with the use of regular splitting we would have iterative method. Using the matrix we truncated the integral term using Fast Fourier Transform (FFT) for efficiency. Finally, we compare the Black-Scholes exact solution with the numerical solution.

Keywords: PIDE, FFT, SDE, Crank-Nicolson, and jump-diffusion process

Declaration

I, the undersigned, hereby declare that the work contained in this research project is my original work, and that any work done by others or by myself previously has been acknowledged and referenced accordingly.

D. Zawokane.

Dumisani Zawokane, 22 September 2020

Contents

1. Introduction

1.1 Background

Option trading forms part of the world's financial markets. A traded option gives the owner the right, not the obligation, the option to buy which is the call option or the option to sell which is the put option, both cases are in the fixed quantity of assets for the specified stock at a fixed point. In the European option, the options can only be exercised on the expiry date. In this essay we will use (Wang and Yang, 2008) approach. There have been several mathematical models that determine the value of derivatives that have been used for European option pricing and one of the standard models is the Black-Scholes model which is widely recognized and became the quality standard in the industry (Zhang and Wang, 2008). The model was developed by Fischer Black and Myron Scholes in 1973 (Black and Scholes, 1973), later that year, the model was modified by Robert Merton. The pricing model is based on the assumption that the underlying stock price follows a geometric Brownian motion with a log-normal diffusion and the volatility is constant, based on these assumptions the Black-Scholes was not consistent with the market price this is called the volatility smile (Lesmana and Wang, 2016).

In 1976 (Merton, 1976), Merton improved the model and the main goal was to balance option pricing when the stock price dynamics are restricted from the possibility of changes and to show the importance of the non-continuous stock price dynamics. Other assumptions were maintained throughout the analysis and this model was referred as jump-diffusion model. The stock price in jump-diffusion has an impact in markets due to rare jump events and also the volatility curves are the same as the volatility smiles (Zhang and Wang, 2008). Pricing under jump-diffusion process requires solving partial integro-differential equation (PIDE) which can be hard to solve numerically since there is a non-local integration (Zhang and Wang, 2008). In (Cont and Voltchkova, 2005), the authors treated the integral term implicitly in their numerical scheme, however the method was conditionally stable. Other approaches used the operator splitting method which was connected with Fast Fourier Transform (FFT) to evaluate the integral as presented in (Andersen and Andreasen, 2000). It is known that when the volatility or underlying stock price goes to zero the jump-diffusion process becomes conventionally dominant so to overcome this difficulty a finite fitted volume method was designed to price the European option (Zhang and Wang, 2009).

1.2 The objective of the study

In this essay, we numerically solve the partial integro-differential equation (PIDE) which has no analytical solution. We use the fitted volume method to discretize spatial derivatives of the linear part of the PIDE where we truncate the specific region of the underlying stock, S we then apply the midpoint quadrature rule for a fitted local approximation, For time computation we discretized by Crank-Nicolson time stepping scheme. Furthermore, we present the system of discretized variables in a form of M-matrix. We used FFT to evaluate the integral term. In this method, integral term is treated implicitly resulting matrix that has elements that are mostly zero to avoid the inversion of dense matrix we proposed the iterative method based on a regular splitting to solve the resulting system (Almendral and Oosterlee, 2005). The price is performed for vanilla put option that we numerically evaluated, and that was implemented using python. We compared it with the exact Black-Scholes analytical solution.

1.3 Essay structure

The essay is structured as follows; Chapter 2 has preliminaries which give brief overview of all the notations needed for derivation of the jump-diffusion method. In Chapter 3, we discretize the partial integro-differential equation (PIDE) using the spatial domain by applying the midpoint quadrature theorem for the non-integral term, then the integral term is discretized lastly, using linear interpolation. We apply Crank-Nicolson for time stepping scheme. In Chapter 4, we implement numerical evaluation on the discretized jump-diffusion process and the iteration method. We implement the FFT on the discretized integral term. We implement the algorithm to get numerical results so that we can verify the robustness of the numerical scheme. Finally in Chapter 5, we present the conclusion.

2. Preliminaries

In this chapter, we present basic definitions, derivation and important notation used in finance such as probability theory, stochastic process, option theory, derivation of jump diffusion method and Fast Fourier Transform.

2.1 Probability theory

2.1.1 Definition. Sample space (Durrett, 2019) A sample space is the set of all possible outcomes and it is denoted by Ω. The elements of a sample space are denoted by ω.

2.1.2 Definition. σ-field (Øksendal, 2003) If Ω is given set, then a σ-algebra $\mathcal{F}$ on Ω is a family $\mathcal{F}$ of subsets of Ω with the following

- The empty set $\emptyset$ belongs to $\mathcal{F}$ ($\emptyset \in \mathcal{F}$).
- F $\in \mathcal{F} \Rightarrow \mathcal{F}^c \in \mathcal{F}$, where $\mathcal{F}^c = F/\Omega$ is the complement of F in Ω.
- $A_1, A_2, \ldots \in \mathcal{F} \Rightarrow A := \bigcup_{i=1}^{\infty} A_i \in \mathcal{F}$.

2.1.3 Definition. Probability measure (Øksendal, 2003) The pair $(\Omega, \mathcal{F})$ is called a measurable space. A probability measure P on a measurable space $(\Omega, \mathcal{F})$ is a function $P : F \to [0, 1]$ such that

- $P(\emptyset) = 0,\ P(\Omega) = 1.$
- If $A_1, A_2, ... \in \mathcal{F}$ and $\{A_i\}_{i=1}^{\infty}$ is disjoint (in other words $A_i \cap A_j = \emptyset$ if $i \neq j$) then,

$$P\Big(\bigcup_{i=1}^{\infty} A_i\Big) = \sum_{i=1}^{\infty} P(A_i).$$

2.1.4 Definition. Probability space (Øksendal, 2003) The triple $(\Omega, \mathcal{F}, P)$ is called a probability space. It is called a complete probability space if $\mathcal{F}$ contains all subsets G of Ω with P outer measure zero, in other words with

$$P^*(G) := \inf\{P(F); F \in \mathcal{F}, G \subset F\} = 0.$$

Any probability space can be made complete simply by adding to $\mathcal{F}$ all sets of outer measure 0 and by extending P accordingly.

2.1.5 Definition. Random variable (Focardi and Fabozzi, 2004) For a given probability space $(\Omega, \mathcal{F}, P)$, a random variable X is defined as a measurable function $X(\omega)$ described over the sample space Ω that takes values in R:

$$(\omega : X(\omega) \leq x) \in \mathcal{F}.$$

2.1.6 Definition. Probability Density Function (Probability Density Fuction, 2020) The probability that a random variable X takes a value in the (open or closed) interval $[a, b]$ is given by the integral of a function

$$P(a \leq X \leq b) = \int_a^b f_X(x)dx,$$

where f_X is the probability density function of the random variable X. This shows f_X must be integrable in the interval $[a, b]$ this means probability density function satisfies the normalisation condition

$$\int_{-\infty}^{\infty} f_X(x)dx = \int_{0}^{\infty} f_X(x)dx = 1. \tag{2.1.1}$$

2.2 Stochastic process

2.2.1 Definition. Stochastic process (Kloeden and Platen, 2013) $(X_t)_{t\in[0,T]}$ is a collection of random variables on common probability space $(\Omega, \mathcal{F}, P)$.

2.2.2 Definition. Wiener process (Wilmott, Dewynne, and Howison, 1993) To define a standard Wiener process $Z = \{Z(t)\,,\, t \geq 0\}$ to be a Gaussian process with independent increment such that

- $Z(0) = 0$,
- $E[Z(t)] = 0$,
- $Var(Z(t) - Z(0)) = t - s$,

for all $0 \leq s \leq t$, this is a mathematical description of the Brownian motion.

2.2.3 Definition. Geometric Brownian motion (Ross, 2014) Is a continuous-time stochastic process in which logarithm of the randomly varying quantity follows a Wiener process.

2.3 Itô formula for jump process

2.3.1 Definition. Itô jump-diffusion process (Mukam, 2015) The Itô jump-diffusion process is any process with the form

$$X_t = X_0 + \int_0^t a(X_s)ds + \int_0^t b(X_s)dW_s + \int_0^t c(X_s)dN_s, \tag{2.3.1}$$

where $a(X_s)$ is the drift coefficient, $b(X_s)$ is the diffusion coefficient, $c(X_s)$ is the jump coefficient, W is a M-dimensional Brownian motion and N is one dimensional.

2.3.2 Proposition. ***Itô formula for jump process*** *(Mukam, 2015) If X_t is a jump-diffusion process of the form (2.3.1) and $f : [0, \infty) \to \mathbb{R}^n$ any function twice derivable, then $Y_t = f(t, X_t)$ is a jump-diffusion process and satisfies the following equation*

$$f(t, X_t) - f(0, X_0) = \int_0^t \left[\frac{\partial f}{\partial s}(s, X_s) + \frac{\partial f}{\partial x}(s, X_s)a_s\right]ds + \frac{1}{2}\int_0^t \frac{\partial^2 f}{\partial x^2}(s, X_s)b_s^2 ds + \int_0^t \frac{\partial f}{\partial x}(s, X_s)b_s dW_s$$
$$+ \int_0^t (f(X_{s^-} + c(X_{s^-})) - f(X_{s^-}))dN_s. \tag{2.3.2}$$

In differential notation:

$$dY_t = \frac{\partial f}{\partial t}(t, X_t)dt + \frac{\partial f}{\partial x}(t, X_t)a_t dt + \frac{1}{2}\frac{\partial^2 f}{\partial x^2}(t, X_t)b_t^2 dt + \frac{\partial f}{\partial x}(t, X_t)b_t dW_t$$
$$+ (f(X_{t^-} + c(X_{t^-})) - f(X_{t^-}))dN_t. \tag{2.3.3}$$

2.3.3 Lemma. Itô's lemma for product (Mukam, 2015) If X_t and Y_t are two Itô's jump-diffusion process, then X_tY_t is an Itô jump-diffusion process

$$d(X_tY_t) = Y_t dX_t + X_t dY_t + dX_t dY_t, \tag{2.3.4}$$

dX_tdY_t is called the Itô's corrective term and it is computed according to the relations

$$dt \cdot dt = dN_t \cdot dt = dt \cdot dN_t = dW_t \cdot dN_t = dN_t \cdot dW_t = 0, \qquad dW_t \cdot dW_t = dt, \qquad dN_t \cdot dN_t = dN_t.$$

2.4 Fast Fourier Transform

In this section, we apply deeper understanding of Fast Fourier Transform (FFT). The FFT algorithm computes the Discrete Fourier Transform (DFT), so when we to perform a multiplication of the dense matrices using brute force the computational cost gets bigger, and for FFT the computation is more efficient for big N.

2.4.1 Lemma. Discrete Fourier Transform (Heideman, Johnson, and Burrus, 1985) Let $x_0, \ldots, x_{n-1}$ be complex numbers of the DFT formula

$$X_k = \sum_{n=0}^{N-1} x_n e^{-i2\pi kn/N} \qquad , \qquad k = 0, \ldots, N-1, \tag{2.4.1}$$

where $e^{-i2\pi/N}$ is a primitive Nth root. By exploiting symmetry, Cooley and Turkey (Heideman, Johnson, and Burrus, 1985) showed that it plausible to divide and DFT into smaller parts. Suppose we separated the Fourier transform into even and odd index, so we have

$$\begin{cases} n = 2m & \text{if even,} \\ n = 2m+1 & \text{if odd,} \end{cases}$$

where $m = 1, 2, \ldots, N/2 - 1$. Then the split of the DFT would be

$$X_k = \sum_{m=0}^{N/2-1} x_{2m} e^{-i2\pi km/(N/2)} + e^{-i2\pi k/N} \sum_{m=0}^{N/2-1} x_{2m+1} e^{-i2\pi km/(N/2)}, \tag{2.4.2}$$

each term consists $(N/2) * N$ computation with the total of N^2. Suppose there is no stop, we reapply the divide and conquer approach by having the computational cost each time until the values are small enough this approach would scale $\mathcal{O}(N \ln N)$. The purpose this notation is to reduce the computational cost from the order $\mathcal{O}(N^2)$ to $\mathcal{O}(N \ln N)$. For the integral term we compute using the convolution theorem.

2.4.2 Definition. Convolution (Heckbert, 1995) The Fourier transform of a convolution of two signals is the product of their Fourier transforms: $f \otimes g \leftrightarrow F \times G$ The convolution of two continuous signals f and g is

$$(f \otimes g)(x) = \int_{-\infty}^{\infty} f(t) g(x-t) dt, \tag{2.4.3}$$

where $\otimes$ denotes the convolution operator. Then we have

$$\mathcal{F}[f \otimes g](u) = \hat{f}(u) \times \hat{g}(u), \tag{2.4.4}$$

and

$$\mathcal{F}[f \times g](u) = \hat{f}(u) \otimes \hat{g}(u). \tag{2.4.5}$$

2.4.3 Definition. Discretized Convolution (Fortran, Press, Teukolsky, Vetterling, and Flannery, 1992) Since the signal $g(t)$ is represented at equal time interval g_j , $f(t)$ is a discrete set of numbers corresponding to response function then f_k tells what multiple of the input signal is copied into identical output channel

$$(f \otimes g)_j = \sum_{k=-N/2+1}^{N/2} g_{j-k} f_k, \tag{2.4.6}$$

where N is a large number.

2.4.4 Definition. Circular Convolution (Frigo and Johnson, 1998) Multiplying by circulant matrices. Suppose we have $N \times N$ circulant matrix G where we multiply by vector $f = (f_0, f_1, \dots, f_N)$. We would have this kind of operation

$$y = G \times f = \begin{pmatrix} g_0 & g_1 & g_2 & \cdots & g_{N-1} \\ g_{N-1} & g_0 & g_1 & \cdots & g_{N-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ g_1 & g_2 & g_3 & \cdots & g_0 \end{pmatrix} \begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_{N-1} \end{pmatrix}$$

to write down the entries of each rows

$$y_0 = g_0 f_0 + g_1 f_1 + g_2 f_2 + \dots, \tag{2.4.7}$$

$$y_1 = g_{N-1} f_0 + g_0 f_1 + g_1 f_2 + \dots, \tag{2.4.8}$$

$$y_2 = g_{N-2} f_0 + g_{N-1} f_1 + g_0 f_2 + \dots, \tag{2.4.9}$$

computing (2.4.7)-(2.4.9), we were able to see a pattern to write the summation

$$y_j = \sum_{k=0}^{N} g_{j-k} f_k. \tag{2.4.10}$$

If we multiply a matrix with a vector with a circulant matrix the computational cost would be $(N \ln N)$ operations. This notation is done by inverse of the FFT. This shows circulant matrix is best fitted for solving the iterative method.

2.5 Option theory

Options are contracts we can trade in the market that have derivatives based on the value of underlying stocks securities. An option contract gives the owner the right, not the obligation, an option to buy or to sell depending on the type of contract they hold (Investopedia, 2020, b). We have two categories of options namely call and put options. A call option is a contract that gives the buyer the right to buy at a specific price and a put option is a contract that allows the buyer to sell at a specific price all over a period of time. A put option gives the owner the option of selling the underlying asset at the expiry. According to that we summarise the options with the effects of call and put option prices (Guo, 2017),

- The price of a call option increases and the price of a put option decreases in the underlying stock price.
- The price of a call option decreases and the price of a put option increases in the strike price.

- The variance of the underlying asset both increases for call and put options.
- Call and put option prices increases when the time reaches maturity.
- Interest rates increases for call option prices and decreases for put option prices.
- The dividends decreases for call option prices and increases for put option prices.

If $S_T < K$, one should not exercise the option because it would make a loss of $K - S_T$ or else the value will be 0 (Wilmott, Dewynne, and Howison, 1993). This pay-off can be expressed as

$$P(S,T) = \max(K - S_T, 0), \tag{2.5.1}$$

where S_T is the underlying stock and K is the strike price. Figure (2.1) below illustrate the behaviour of the pay-off function for put option

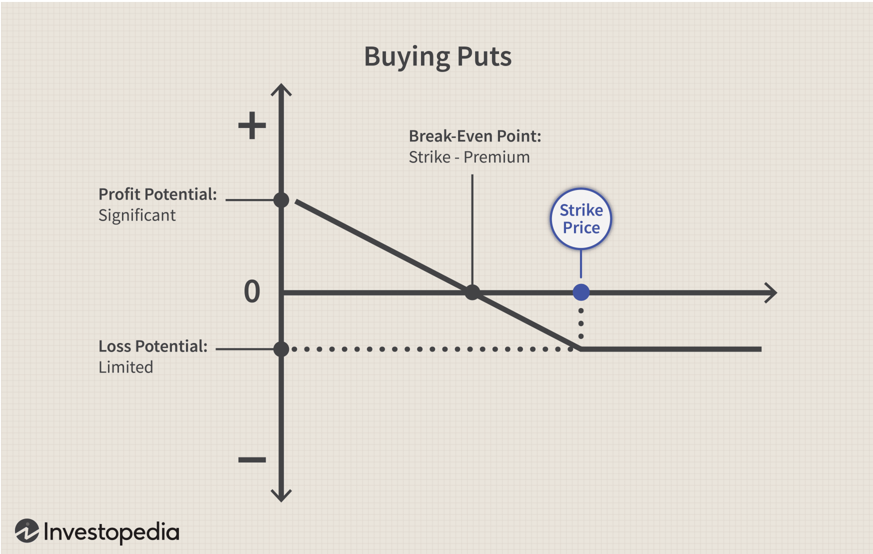

Figure 2.1: Put Option (Investopedia, 2020, a).

A call option gives the owner the option of buying the underlying asset at the expiry. If $S_T > K$, one should exercise the option because it would make a profit of $S_T - K$ or else the value will be 0. This pay-off can be expressed as

$$C(S,T) = \max(S_T - K, 0), \tag{2.5.2}$$

Figure (2.2) below illustrate the behaviour of the pay-off function for call option.

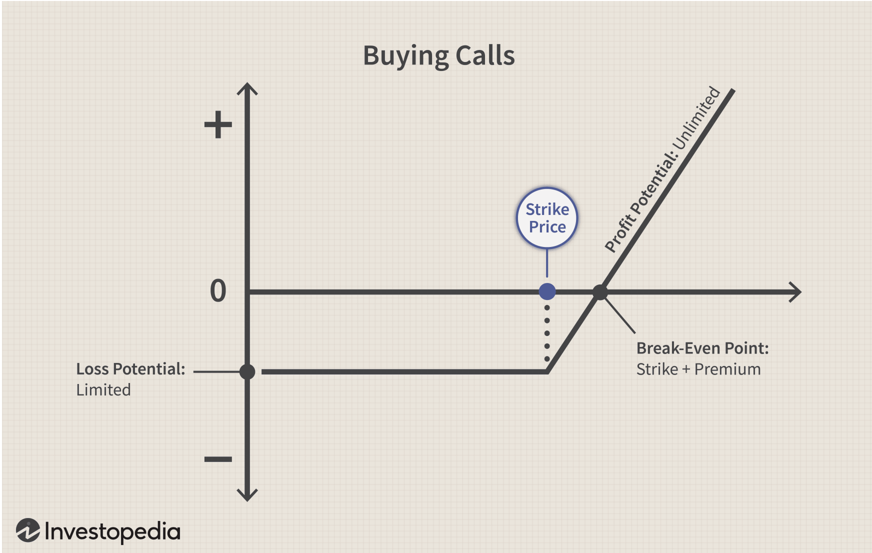

Figure 2.2: Call Option (Investopedia, 2020, a).

We use the term vanilla option to denote payoff $V(S_T)$ where V is some function with exercise time T. There are three ways to describe the behaviour of an option before expiry date, then outcome depends on underlying stock that is in relation with strike price. The option can be in, out or at the money. At the money states that the strike price is equivalent to the underlying stock and time value is at maximum when an option is at the money. In the money option states that, for put option, the underlying stock is less than the strike price of the option and time value decreases as the option gets deeper in the money. Lastly, out of money option states that underlying stock is greater than the strike price and time value decreases as the option get deeper out of the money.

There are many types of style options, we have American option where the holder has the right to buy or sell the underlying stock before the expiry date, Asian option where the pay-off depends on the average price of the underlying stock and Compound option which presents the holder with two separate exercise dates, however in this work we only focus on the option that only exercise at the expiry date which is the European option. Unlike other options, in European option, the owner can exercise when the value is at maturity. European option is often used in stock and foreign currency because it is cheaper than other available options. There are several studies that were conducted to model the European option. Amongst the studies (Klar and Jacobson, 2002) suggested Monte Carlo and Numerical Analysis to price the vanilla price option to determine the numerical solution of the model.

2.6 Derivation of jump-diffusion model

Jump-diffusion model is mathematical model used for option pricing in financial market where the following assumptions apply (Hull, 2003)

- No transaction costs of differential taxes, there is no borrowing and short sales.
- No share of profits.
- Short trading is not prohibited, this means there is possibility of investment or trade if the stock declines.

- No arbitrage possibilities, no one gains without taking any risk.
- The stock price (S_t) is defined as a stochastic differential equation:

$$\frac{dS_t}{S_t} = (\mu - \lambda\kappa)dt + \sigma dZ_t + (\eta - 1)dq_t \tag{2.6.1}$$

  where:

  - μ is the drift rate,
  - σ is the volatility,
  - dZ_t is the Wiener process,
  - η the impulse function,
  - κ expectation operator of an impulse function.
  - λ deterministic jump intensity,
  - dq_t is the Poisson process.

The derivation below was implemented in (Forsyth, 2005). Consider occurrence of a jump in a rare occasion where the jump size does not depend on the interval, then formally we consider the process dq where we have interval from t to the change of $t + dt$. We have

$$\begin{aligned} dq = 1: &\quad \text{with probability } \lambda\text{dt} \\ dq = 0: &\quad \text{with probability 1-}\lambda\text{dt} \end{aligned}$$

the dq outcome does not depend on the change in time dt, then the probability of the jump on the interval shows $dt \to 0$. Note that

$$\begin{aligned} E[dq] &= \lambda dt \cdot 1 + (1 - \lambda dt) \cdot 0 \\ dq_t &= \lambda dt. \end{aligned} \tag{2.6.2}$$

We also assume that the jump size depends on probability density $g(\eta)$, given that jump occurs then the probability of the jump in $[\eta, \eta + d\eta]$ is $g(\eta)d\eta$ and

$$\int_{-\infty}^{\infty} g(\eta)d\eta = \int_{0}^{\infty} g(\eta)d\eta = 1 \tag{2.6.3}$$

assuming $g(\eta) = 0$, if $\eta < 0$ we let $h = h(\eta)$ to be known function then the expected value of h is

$$E[h] = \int_{0}^{\infty} h(\eta)g(\eta)d\eta \tag{2.6.4}$$

the process is geometric Brownian motion with rare discontinuous jumps this process is Figure (2.3) below,

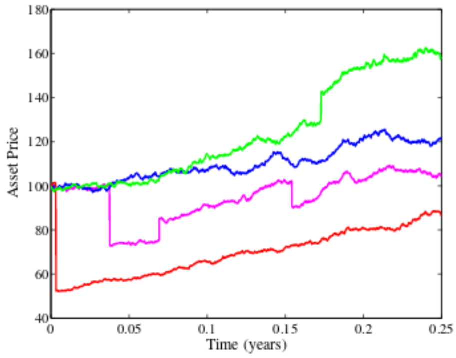

Figure 2.3: The diagram shows the jump-diffusion paths for assets price throughout the years. (Forsyth, 2005)

Since we have combination of geometric Brownian motion and a rare jump event that has been defined as Itô jump-diffusion process in equation (2.3.1)

$$dS_t = (\mu - \lambda\kappa)Sdt + \sigma SdZ_t + (\eta - 1)Sdq_t. \tag{2.6.5}$$

where $\eta - 1$ is an impulse function producing a jump from S to $S\eta$. If we let $V(S,t)$ be the value of a contingent claim that depends on the underlying stock price S and time t, Merton (Merton, 1976) shows that by applying Itô proposition (2.3.2) for the diffusion part of the process in equation (2.6.5) then

$$dV_t = \left((\mu - \lambda\kappa)S\frac{\partial V}{\partial S} + \frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2}\right)dt + \sigma S\frac{\partial V}{\partial S}dZ_t + (V(S\eta) - V(S))dq_t. \tag{2.6.6}$$

Suppose delta-hedge portfolio consisting of one short option and a long position of $\partial V/\partial S$ which are the shares of the underlying stock:

$$\Pi_t = V_t - \Delta S_t, \tag{2.6.7}$$

where $\Delta = \partial V/\partial S$, this shows that the change in portfolio would give

$$d\Pi_t = dV_t - \Delta dS_t, \tag{2.6.8}$$

substituting (2.6.6) and (2.6.5) into (2.6.8) gives

$$\begin{aligned} d\Pi_t = &\left[\left((\mu - \lambda\kappa)S\frac{\partial V}{\partial S} + \frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2}\right)dt + \sigma S\frac{\partial V}{\partial S}dZ_t + (V(S\eta) - V(S))dq_t\right] \\ &+ \frac{\partial V}{\partial S}((\mu - \lambda\kappa)Sdt + \sigma SdZ_t + (\eta - 1)Sdq_t) \end{aligned} \tag{2.6.9}$$

On simplification we have

$$d\Pi_t = \left(\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2}\right)dt + (V(S\eta) - V(S))dq_t - \frac{\partial V}{\partial S}(\eta - 1)Sdq_t. \tag{2.6.10}$$

We still have a random component (dq_t) which is not hedged away to solve this we use $E(\cdot)$ the expectation operator of change in the portfolio

$$E[d\Pi_t] = \left(V_t + \frac{1}{2}\sigma^2 S^2 V_{SS}\right)dt + E[V(S\eta) - V(S)]E[dq_t] - V_S S E[\eta - 1]E[dq_t] \tag{2.6.11}$$

applying condition (2.6.2) and we defining $\kappa = E[\eta - 1]$, (2.6.11) becomes

$$E[d\Pi_t] = \left(V_t + \frac{1}{2}\sigma^2 S^2 V_{SS}\right)dt + E[V(S\eta) - V(S)]\lambda dt - V_S S\kappa\lambda dt. \tag{2.6.12}$$

Merton (Merton, 1976) argues that jump component dq_t of an asset are uncorrelated with a market as a hole, so there should be no risk premium and variance of this portfolio of portfolios is small.Therefore, the portfolio is expected to grow at the risk-free interest rate r. Hence, the expected return should be

$$E[d\Pi_t] = r\Pi dt = r(V - V_S)dt \tag{2.6.13}$$

then equating (2.6.13) with (2.6.12) gives

$$r(V - V_S)dt = \left(V_t + \frac{1}{2}\sigma^2 S^2 V_{SS}\right)dt + E[V(S\eta) - V(S)]\lambda dt - V_S S\kappa\lambda dt. \tag{2.6.14}$$

We know that $E[V] = V$ since the jump does not occur, then solving (2.6.14) further gives

$$V_t + \frac{1}{2}\sigma^2 S^2 V_{SS} + (r - \lambda\kappa)SV_S - (r + \lambda)V + \lambda E[V(S\eta)] = 0, \tag{2.6.15}$$

since $E[V(S\eta)]$ jump occurs as defined by (2.6.4) then

$$V_t + \frac{1}{2}\sigma^2 S^2 V_{SS} + (r - \lambda\kappa)SV_S - (r + \lambda)V + \lambda\int_0^\infty V(S\eta)g(\eta)d\eta = 0. \tag{2.6.16}$$

By setting $\tau = T - t$, then we have $d\tau = -dt$ this would give

$$\frac{\partial V}{\partial \tau} = -\frac{\partial V}{\partial t} \tag{2.6.17}$$

then partial integro-differential equation (PIDE) for pay-off in European option, so we have

$$V_\tau = \frac{1}{2}\sigma^2 S^2 V_{SS} + (r - \lambda\kappa)SV_S - (r + \lambda)V + \lambda\int_0^\infty V(S\eta)g(\eta)d\eta, \tag{2.6.18}$$

for $(S, \tau) \in [0, \infty) \times [0, T]$, where $\tau = T - t$ is the time until the maturity at time T, r is continuous risk free interest rate and $g(\eta)$ is the probability density function of the jump amplitude η with obvious properties that $\forall\eta$, $g(\eta) \geq 0$ and $\int_0^\infty g(\eta)d\eta = 1$ (see more on (Merton, 1976; Andersen and Andreasen, 2000; Zhang and Wang, 2008)). In this essay we consider a specific model: Merton's model where $g(\eta)$ is given by the log-normal density

$$g(\eta) = \frac{1}{\sqrt{2\pi}\sigma_J\eta}\exp\left(\frac{-(\ln\eta - \nu)^2}{2\sigma_J^2}\right). \tag{2.6.19}$$

Hence, $\kappa = E(\eta - 1) = \exp(\nu + \sigma_J^2/2) - 1$, where ν is the mean and σ_J is the variance of the jump return. The boundary and the initial condition where V^* is the pay off function for the European vanilla put option

$$\begin{aligned} V(0,\tau) &= Ke^{rt}, \qquad S \to 0, \\ V(\infty,\tau) &= 0, \qquad S \to +\infty, \\ V(S,\tau=0) &= V^*(S) = \max(K-S,0), \end{aligned} \tag{2.6.20}$$

where K is a strike price. When we compress (2.6.18) we have

$$V_\tau = \frac{\partial}{\partial S}\left(aS^2\frac{\partial V}{\partial S} + bSV\right) - cV + \lambda Q(S), \tag{2.6.21}$$

where $a = \sigma^2/2$, $b = r - \lambda\kappa - \sigma^2$ and $c = r + \lambda + b$. The integral part $Q(S)$, of (2.6.21) is defined

$$Q(S) = \int_0^\infty V(S\eta)g(\eta)d\eta. \tag{2.6.22}$$

In the next chapter, we transfer the pricing model (2.6.21) which is the continuous function into discrete counterpart.

3. Discretizing the jump-diffusion model

In this chapter, we discretize the jump-diffusion pricing model using the finite fitted volume method on the non-integral terms, we also discretize the integral term with linear interpolation. Lastly, we fully discretize using the Crank Nicolson time-stepping method to obtain the matrix form. To get iterative method we will use regular splitting definition so that we can be able to implement the algorithm.

3.1 Fitted volume method

In this section, we apply fitted volume method on (2.6.21). We have underlying stock of S which has interval of $(0, \infty)$, during the discretization we introduce S_{max} as the largest value then the asset transforms into $I = (0, S_{max})$ using this set we are able to neglect the truncation error, so that it is efficient to implement fitted volume method. Firstly we define two partitions for I which can be divided into N sub-intervals that gives

$$I_i = (S_i, S_{i+1}), \qquad i = 1, \dots, N,$$

with $0 = S_1 < S_2 < \cdots < S_{N+1} = S_{max}$. For the second partition we divide further,

$$J_i = (S_{i-1/2}, S_{i+1/2}), \qquad i = 1, \dots, N,$$

where

$$S_{i-1/2} = \frac{S_{i-1} + S_i}{2}, \tag{3.1.1}$$

and

$$S_{i+1/2} = \frac{S_{i+1} + S_i}{2}, \tag{3.1.2}$$

the initial spacial points are $S_1 = S_{1/2}$ and $S_{N+3/2} = S_{N+1}$ then integrating (2.6.21) over J_i, we have

$$\int_{J_i} V_\tau dS = S\Big(aS\frac{\partial V}{\partial S} + bV\Big)\Big|_{S_{i-1/2}}^{S_{i+1/2}} - \int_{J_i} cV dS + \lambda \int_{J_i} Q(S) dS, \tag{3.1.3}$$

then we apply the mid-point quadrature rule on (3.1.3) on each term, we are able to compute only three terms which are

$$\int_{J_i} V_\tau dS \approx \frac{\partial V_i}{\partial \tau} l_i, \quad \int_{J_i} cV dS \approx c l_i V_i \quad \text{and} \quad \int_{J_i} Q(S) dS \approx l_i Q(S_i), \tag{3.1.4}$$

where $l_i = S_{i+1/2} - S_{i-1/2}$, that is in J_i interval and V_i is the nodal approximation. For the last term we have

$$S\Big(aS\frac{\partial V}{\partial S} + bV\Big)\Big|_{S_{i-1/2}}^{S_{i+1/2}} \approx S_{i+1/2}\rho(V)|_{S_{i+1/2}} - S_{i-1/2}\rho(V)|_{S_{i-1/2}}, \tag{3.1.5}$$

where the flux $\rho(V) = aSV' + bV$, we apply (3.1.4) and (3.1.5) into (3.1.3) we get

$$\frac{\partial V_i}{\partial \tau} l_i = S_{i+1/2}\rho(V)|_{S_{i+1/2}} - S_{i-1/2}\rho(V)|_{S_{i-1/2}} - c l_i V_i + \lambda l_i Q(S_i). \tag{3.1.6}$$

To get the approximation of the flux, we have two boundary conditions and the derivative of the flux which equals to zero and the approximation of the flux is defined above at the midpoint so we will use $S_{i+1/2}$ as indication

$$\begin{aligned} (aSV' + bV)' = 0, \qquad S \in I_i, \\ V(S_i) = V_i, \qquad V(S_{i+1}) = V_{i+1}, \end{aligned} \tag{3.1.7}$$

then solving the problem analytically we obtain,

$$\rho_i(V) = \rho(V)|_{S_{i+1/2}} = \frac{bS_{i+1}^{\eta}V_{i+1} - S_i^{\eta}V_i}{S_{i+1}^{\eta} - S_i^{\eta}}, \tag{3.1.8}$$

where $\eta = b/a$. For boundary condition at $S_{i-1/2}$ gives

$$\begin{aligned} (aSV' + bV)' = 0, \qquad S \in I_i, \\ V(S_i) = V_i, \qquad V(S_{i-1}) = V_{i-1}, \end{aligned} \tag{3.1.9}$$

then solving the problem analytically we obtain

$$\rho(V)|_{S_{i-1/2}} = \frac{bS_i^{\eta}V_i - S_{i-1}^{\eta}V_{i-1}}{S_i^{\eta} - S_{i-1}^{\eta}}, \tag{3.1.10}$$

where $\eta = b/a$. The flux approximation does not apply on the first two underlying asset so in the interval $I = (0, S_2)$ we consider the flux with extra degree of freedom

$$\begin{aligned} (aSV' + bV)' = C, \qquad S \in I_1, \\ V(0) = V_1, \qquad V(S_2) = V_2, \end{aligned} \tag{3.1.11}$$

then solving the problem analytically we obtain

$$\rho_1(V) = \frac{1}{2}[(a+b)V_2 - (a-b)V_1], \tag{3.1.12}$$

using (3.1.12) and (3.1.8) we define a global piecewise constant approximation of $\rho(V)$ by $\rho_h(V)$ which is

$$\rho_h(V) = \rho_i(V), \qquad S \in I_i, \tag{3.1.13}$$

then substituting (3.1.8) and (3.1.10) into (3.1.6) gives

$$\frac{\partial V_i}{\partial \tau} = \alpha_i V_{i-1} + \gamma_i V_i + \beta_i V_{i+1} + \lambda Q(S_i), \tag{3.1.14}$$

where

$$\begin{aligned} \alpha_2 &= \frac{S_2}{4l_2}(a-b) \\ \beta_2 &= \frac{bS_{2+1/2}S_3^{\eta}}{S_3^{\eta} - S_2^{\eta}} \\ \gamma_2 &= -\frac{S_2}{4l_2}(a+b) - \frac{bS_{2+1/2}S_3^{\eta}}{S_3^{\eta} - S_2^{\eta}} - c, \end{aligned}$$

and

$$\alpha_i = \frac{bS_{i-1/2}S^\eta_{i-1}}{(S^\eta_i - S^\eta_{i-1})l_i}$$
$$\beta_i = \frac{bS_{i+1/2}S^\eta_{i+1}}{(S^\eta_{i+1} - S^\eta_i)l_i}$$
$$\gamma_i = -\frac{bS_{i-1/2}S^\eta_{i-1}}{(S^\eta_i - S^\eta_{i-1})l_i} - \frac{bS_{i+1/2}S^\eta_{i+1}}{(S^\eta_{i+1} - S^\eta_i)l_i} - c,$$

for $i = 3, \ldots, N$.

3.2 Discretization of the integral term

In this section, we introduce logarithmic price on the integral form that has been presented on the partial integro-differential equation (PIDE) in (2.6.18). We define the integral term in the form of

$$Q(S) = \int_0^\infty V(S\eta)g(\eta)d\eta. \tag{3.2.1}$$

If we let $x = \ln(S)$ and $y = ln(\eta)$ this means that in (2.6.22) we would replace $S = e^x$ and $\eta = e^y$ therefore the expression with a logarithmic price for the integral term which is discretized in the form of

$$\overline{Q}(x) = \int_{-\infty}^\infty \overline{V}(x+y)f(y)dy, \tag{3.2.2}$$

where $f(y) = g(e^y)e^y$ and $\overline{V}(x+y) = V(e^{x+y})$. Then to approximate the integral and also applying the midpoint quadrature theorem, we apply the finite sequence given by $x_i = i\Delta x$ for $i \in \mathbb{Z}$ where Δx is a constant step length. Let $y_i = x_i$ for $i \in \mathbb{Z}$, this means that $\Delta x = \Delta y$ to get discrete form of the integral the approximation at the nodal point

$$Q(x_i) = \sum_{j=-N/2+1}^{N/2} \overline{V}_{i+j} f_j \Delta y, \tag{3.2.3}$$

where $\overline{V}_k$ is the nodal approximation for $\overline{V}(k\Delta x)$ and

$$f_j = \frac{1}{\Delta y}\int_{(j-1/2)\Delta y}^{(j+1/2)\Delta y} f(y)dy. \tag{3.2.4}$$

Since $f(y)$ is a probability density function then f_j decays rapidly for $|j| > 0$ this notation would not be difficult to approximate, We can only have difficulties with $\overline{V}_k$ then we apply the interpolation scheme if we assume $\overline{V}_{N/2+j}$, $j > 0$ can be approximated by an asymptotic boundary condition and also the lower bound $\overline{V}_{-N/2+j}, j < 0$ can be interpolated. The discrete form of the correlation integral use equally spaced grid, the approach is not suitable for discretizing the PIDE. Then we implement the unequal spaced grid in S coordinates to determine the appropriate values, we let

$$S_{p(j)} \leq e^{x_j} \leq S_{p(j)+1}, \qquad j \in \mathbb{Z}, \tag{3.2.5}$$

where $p(j)$ is the index, using the linear interpolation of $V(S)$, we define approximation to $V(x_j)$ is

$$V(x_j) = \phi_{p(j)}V(S_{p(j)}) + (1-\phi_{p(j)})V(S_{p(j)+1}), \tag{3.2.6}$$

where the interpolation weight is

$$\phi_{p(j)} = \frac{e^{j\Delta x} - S_{p(j)}}{S_{p(j)+1} - S_{p(j)}}, \tag{3.2.7}$$

then for the integral $\overline{Q}_i$ which is evaluated at $S = e^{x_i}$ that does not approach the grid at point S_i for any $i = 1, 2, \ldots$ therefore we let q_i to be integer

$$e^{x_{q(i)}} \leq S_i \leq e^{x_{q(i)+1}}, \qquad i \in \mathbb{Z}, \tag{3.2.8}$$

then apply the linear interpolation on $Q(S)$ to define $\overline{Q}(x_i)$ which gives

$$\overline{Q}(x_i) = \psi_{q(i)}\overline{Q}(x_{q(i)}) + (1 - \psi_{q(i)})\overline{Q}(x_{q(i)+1}), \tag{3.2.9}$$

where the interpolation weight is

$$\psi_{q(i)} = \frac{\ln S_i - x_{q(i)}}{x_{q(i)+1} - x_{q(i)}}. \tag{3.2.10}$$

Note that it is clear that interpolation weighted vector can be given

$$0 \leq \phi_{p(j)} \leq 1 \tag{3.2.11}$$
$$0 \leq \psi_{q(i)} \leq 1. \tag{3.2.12}$$

By substituting equation (3.2.6),(3.2.10) and (3.2.3), we define the approximation $Q(S)$ by

$$Q(x_i) = \sum_{j=-N/2+1}^{N/2} \Pi_j^i(V) f_j \Delta y, \tag{3.2.13}$$

where

$$\begin{aligned}\Pi_j^i(V) = {} & \psi_{q(i)}[\phi_{p(q(i)+j)} V(S_{p(q(i)+j)}) + (1 - \phi_{p(q(i)+j)}) V(S_{p(q(i)+j)+1})] \\ & + (1 - \psi_{q(i)})[\phi_{p(q(i)+1+j)} V(S_{p(q(i)+1+j)}) + (1 - \phi_{p(q(i)+1+j)}) V(S_{p(q(i)+1+j)+1})],\end{aligned} \tag{3.2.14}$$

and this shows that $\Pi(V, i, j)$ is linear in V therefore, using the weight property

$$\Pi(V, i, j) = 1, \qquad \forall i, j, \tag{3.2.15}$$

then we have discretized the integral term. To complete (3.1.14), we to substitute $Q(x_i)$ to get a full interpretation of the equation. Now we have

$$\frac{\partial V_i}{\partial \tau} = \alpha_i V_{i-1} + \gamma_i V_i + \beta_i V_{i+1} + \lambda \sum_{j=-N/2+1}^{N/2} \Pi_j^i(V) f_j \Delta y. \tag{3.2.16}$$

3.3 Time discretization

For the time discretization we let τ denote a point from $[0, T]$ such that

$$0 = \tau_0 < \tau_1 < \tau_2 < ... < \tau_{nt} = T, \quad \text{and} \quad \Delta\tau = \tau_n - \tau_{n-1} \geq 0, \quad n = 0, 1, 2, ..., nt,$$

where $nt > 1$ is a positive integer. To improve the accuracy we apply Crank-Nicolson scheme for time discretization to (3.2.16) for simplicity which gives

$$\begin{aligned}\frac{V_i^{n+1} - V_i^n}{\Delta\tau} = \alpha_i\left(\frac{V_{i-1}^{n+1} + V_{i-1}^n}{2}\right) + \gamma_i\left(\frac{V_i^{n+1} + V_i^n}{2}\right) + \beta_i\left(\frac{V_{i+1}^{n+1} + V_{i+1}^n}{2}\right) \\ + \lambda\sum_j \Pi_j^i\left(\frac{V^{n+1} + V^n}{2}\right) f_j \Delta y,\end{aligned} \tag{3.3.1}$$

for simplicity we have

$$\begin{aligned}V_i^{n+1} - V_i^n = \frac{\alpha_i\Delta\tau}{2}(V_{i-1}^{n+1} + V_{i-1}^n) + \frac{\gamma_i\Delta\tau}{2}(V_i^{n+1} + V_i^n) + \frac{\beta_i\Delta\tau}{2}(V_{i+1}^{n+1} - V_{i+1}^n) \\ + \frac{\lambda\Delta\tau}{2}\sum_j \Pi_j^i(V^{n+1}) f_j \Delta y + \frac{\lambda\Delta\tau}{2}\sum_j \Pi_j^i(V^n) f_j \Delta y,\end{aligned} \tag{3.3.2}$$

then rearranging (3.3.2),

$$\begin{aligned}-\frac{\alpha_i\Delta\tau}{2}V_{i-1}^{n+1} + \left[1 - \frac{\gamma_i\Delta\tau}{2}\right]V_i^{n+1} - \frac{\beta_i\Delta\tau}{2}V_{i+1}^{n+1} - \frac{\lambda\Delta\tau}{2}\sum_j \Pi_j^i(V^{n+1}) f_j \Delta y = \\ \frac{\alpha_i\Delta\tau}{2}V_{i-1}^n + \left[1 + \frac{\gamma_i\Delta\tau}{2}\right]V_i^n + \frac{\beta_i\Delta\tau}{2}V_{i+1}^n + \frac{\lambda\Delta\tau}{2}\sum_j \Pi_j^i(V^n) f_j \Delta y.\end{aligned} \tag{3.3.3}$$

Let vector $V^n = [V_1^n, V_2^n, ..., V_{N+1}^n]$, M to be on $N \times N$ matrix and $D = d_{ij}$ we have

$$\left[\frac{M}{2}V^n\right]_i = -\frac{\alpha_i\Delta\tau}{2}V_{i-1}^n - \frac{\gamma_i\Delta\tau}{2}V_i^n - \frac{\beta_i\Delta\tau}{2}V_{i+1}^n, \tag{3.3.4}$$

$$\left[\frac{D}{2}V^n\right]_i = \frac{\lambda\Delta\tau}{2}\sum_j \Pi_j^i(V^n) f_j \Delta y, \tag{3.3.5}$$

then (3.3.3) in the form of a matrix would be

$$\left[I + \frac{M}{2} + \frac{D}{2}\right]V^{n+1} = \left[I - \frac{M}{2} - \frac{D}{2}\right]V^n. \tag{3.3.6}$$

We assume that $I + \frac{M}{2} + \frac{D}{2}$ has off diagonal element, positive diagonal element and diagonally dominant when n and $\Delta\tau$ are sufficiently small. Therefore we can say $I + \frac{M}{2} + \frac{D}{2}$ is M-matrix. The dense matrix D that arise from discretization of correlation product term makes (3.3.6) to be computationally expensive. To solve this difficulty, we use the iterative method. Let

$$A = I + \frac{M}{2} + \frac{D}{2} \quad \text{and} \quad b = \left[I - \frac{M}{2} - \frac{D}{2}\right]V^n, \tag{3.3.7}$$

system (3.3.6) would be in this form

$$AV^{n+1} = b, \tag{3.3.8}$$

then we split A into

$$A = \left(I + \frac{M}{2}\right) - \left(-\frac{D}{2}\right) = P - R, \tag{3.3.9}$$

where P is tridiagonal matrix and R is the Toeplitz matrix.

4. Numerical simulation and results

In this chapter, we implement the numerical simulation, evaluate the jump integral term and implement the Fast Fourier Transform (FFT) method, we also give a numerical example to illustrate the performance and the convergence of the method that was presented in Chapter 3.

4.1 Numerical implementation

In this section, we have the correlation product term. We know that matrix D has elements that are mostly zero this makes the evaluation in (3.3.8) to be computationally expensive as cost is order $\mathcal{O}(N^2)$ thus we use the FFT to evaluate the matrix where the computational cost of in this case is of order $\mathcal{O}(N \ln N)$. To apply the FFT to $R\hat{V}^\ell$ would produce the wrap around pollution then we fix firmly a Toeplitz matrix R into circulant matrix C (Van Loan, 1992). In numerical analysis, circulant matrices have play a huge role because they are diagonalised by a discrete Fourier transform this notation is solved faster by using FFT. They can be interpreted analytically as the integral kernel of a convolution operator hence below we have

$$C = \left[\begin{array}{cccccc|cccccc} f_0 & f_1 & f_2 & \cdots & f_{N-1} & f_N & f_{-N} & f_{1-N} & f_{2-N} & \cdots & f_{-2} & f_{-1} \\ f_{-1} & f_0 & f_1 & \cdots & f_{N-2} & f_{N-1} & f_N & f_{-N} & f_{1-N} & \cdots & f_{-3} & f_{-2} \\ f_{-2} & f_{-1} & f_0 & \cdots & f_{N-3} & f_{N-2} & f_{N-1} & f_N & f_{-N} & \cdots & f_{-4} & f_{-3} \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ f_{-N} & f_{1-N} & f_{2-N} & \cdots & f_{-1} & f_0 & f_1 & f_2 & f_3 & \cdots & f_{N-1} & f_N \\ \hline f_N & f_{-N} & f_{1-N} & \cdots & f_{-2} & f_{-1} & f_0 & f_1 & f_2 & \cdots & f_{N-2} & f_{N-1} \\ f_{N-1} & f_N & f_{-N} & \cdots & f_{-3} & f_{-2} & f_{-1} & f_0 & f_1 & \cdots & f_{N-3} & f_{N-2} \\ f_{N-2} & f_{N-1} & f_N & \cdots & f_{-4} & f_{-3} & f_{-2} & f_{-1} & f_0 & \cdots & f_{N-4} & f_{N-3} \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ f_1 & f_2 & f_3 & \cdots & f_N & f_{-N} & f_{1-N} & f_{2-N} & f_{3-N} & \cdots & f_{-1} & f_0 \end{array}\right],$$

this circulant matrix (C) show that each row vector is rotated one element to the right relative to the proceeding row vector. If we define

$$V^{\ell+1} = [\hat{V}_1^{\ell+1}, \ldots, \hat{V}_{N+1}^{\ell+1}, \underbrace{0, \ldots, 0}_{N}]^T \tag{4.1.1}$$

then the matrix vector product $R\hat{V}^\ell$ is then recognised as the first $(N+1)$ entry in $CV^{\ell+1}$. We then use the Dirichlet boundary conditions stated in (2.6.20), this would apply on the first and last notation of the matrix vector product. We compute the product $R\hat{V}^\ell$ in the following three FFT define the vector

$$F = (f_0, f_1, \ldots, f_N, f_{-N}, f_{1-N}, \ldots, f_{-2}, f_{-1}) \tag{4.1.2}$$

since $f(z)$ is probability density function, we compute $FFT(F)$ where the notation presents the FFT function applied F. Since correlation is dense then FFT is the best efficient choice for computation. Firstly, we compute the correlation in the frequency domain, then the product of $R\hat{V}^\ell$ assuming F is real

$$R\hat{V}^\ell = -\frac{D}{2}\hat{V}^\ell = IFFT(FFT(F) \times FFT(\hat{V})^*), \tag{4.1.3}$$

where $(\cdot)^*$ denotes the complex conjugate, $IFFT(\cdot)$ represents the inverse Fourier transform function. Summarizing the numerical implementation for system, we have the numerical algorithm:

4.1.1 Algorithm. Numerical algorithm. (Zhang and Wang, 2008)

1. Choose the parameters, tolerance and let $n = 0$;
2. Compute the $FFT(F)$;
3. Set $\ell = 0$ and $\hat{V}^\ell = V^n$;
4. Compute $FFT(\hat{V}^\ell)$;
5. Compute the inverse FFT
$$-\frac{D}{2}\hat{V}^\ell = IFFT(FFT(F) \times FFT(\hat{V})^*);$$
6. Using the European option iterative scheme, solve
$$\left[I + \frac{M}{2}\right]\hat{V}^{\ell+1} = -\frac{D}{2}\hat{V}^\ell + \left[I - \frac{M}{2} - \frac{D}{2}\right]V^n,$$
for $\hat{V}^{\ell+1}$;
7. If $\max_i \frac{|\hat{V}_i^{\ell+1} - \hat{V}_i^\ell|}{|\max(1, \hat{V}_i^{\ell+1})|} <$ tolerance then stop; $\ell = \ell + 1$, go to step 4
8. Set $V^{n+1} = \hat{V}_i^\ell$ and $n = n + 1$, to go step 3.

Algorithm (4.1.1) summarises the implementation of FFT approach and the iterative scheme used in Python for pricing the European vanilla put option.

The code can be accessed in this link below:

https://colab.research.google.com/drive/1EQJ4dbJsVShY5zB7G8opQ0vkUJwfebv3?usp=sharing

4.2 Results

In the previous chapter, we showed how to discretize the partial integro-differential equation (PIDE), we price the option using FFT approach and compare the numerical experiment with analytical values of the Black-Scholes model. We know that the exact solution for the Black-Scholes for the put option is given by

$$P(S, t) = Ke^{-r(T-t)}\Phi(d_2) - S\Phi(d_1), \tag{4.2.1}$$

$$d_1 = \frac{\log\left(\frac{S}{K}\right) + \left(r + \frac{1}{2}\sigma^2\right)(T - t)}{\sigma(\sqrt{T - t})}, \tag{4.2.2}$$

and

$$d_2 = \frac{\log\left(\frac{S}{K}\right) + \left(r - \frac{1}{2}\sigma^2\right)(T - t)}{\sigma(\sqrt{T - t})}, \tag{4.2.3}$$

where $P(S,t)$ is the pay-off of the put option, T is the time at maturity, K is the strike price, S is the stock price, t is the present time, r is the risk free interest rate and σ is the volatility with $\Phi(\cdot)$ is the cumulative distribution function of the standard normal distribution. We will also illustrate the European vanilla put option value for the FFT and study the convergence of the iterative method. For the numerical experiment we use the following parameters: for Black-Scholes we use the following

Table 4.1: Parameter used to value a vanilla put option under Merton's model.

Parameter Values		
$K = 100$	$\mu = -0.9$	$T = 0.25$
$S = 200$	$\sigma_J = 0.45$	$\sigma = 0.15$
$r = 0.05$	$\lambda = 0.10$	

parameters:

Table 4.2: Parameter used to value a vanilla put option under Black-Scholes model.

Parameter Values		
$K = 100$	$\sigma = 0.15$	$T = 0.25$
$S = 200$	$r = 0.05$	

.

We plot the values to investigate the accuracy of the put option price. Figure (4.1) below shows the exact solution of the European vanilla put option with the parameters given on Table (4.2).

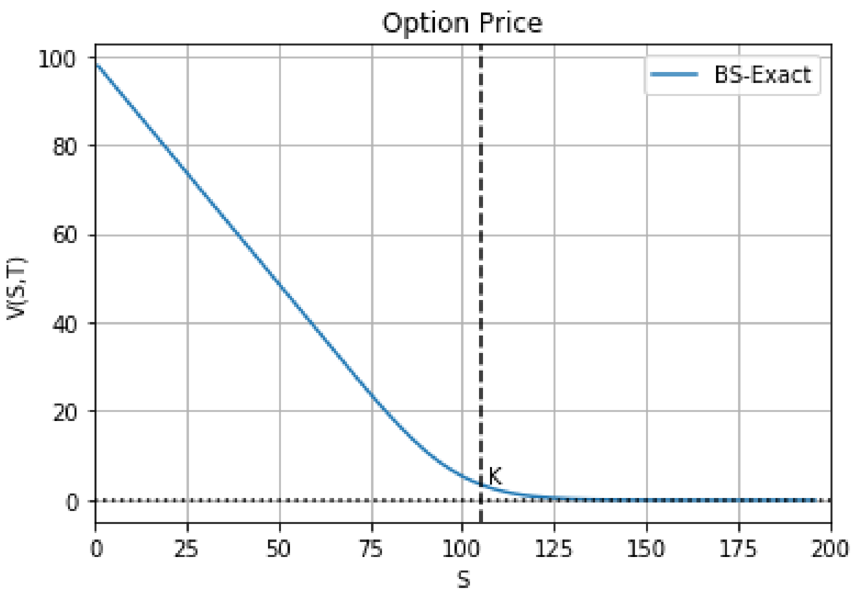

Figure 4.1: Exact solution of the European put vanilla option.

Figure (4.2) below shows European vanilla put option obtained by the numerical simulation using the FFT approach with the parameters given on Table (4.1)

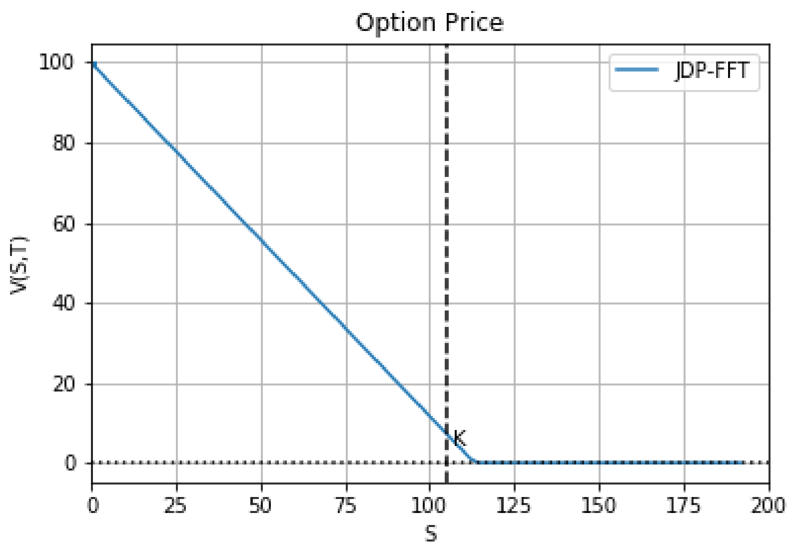

Figure 4.2: European vanilla put option obtained by the numerical simulation with tolerance $= 10^{-8}$ and the grid:s-steps $= 2^8$,τ-steps $= 800$

where s-steps is the space steps and τ-steps is the time steps. To compare the option prices, we analyse the two vanilla put options. Figure (4.3) showed that the numerical simulation converges faster than the exact solution which is shown below.

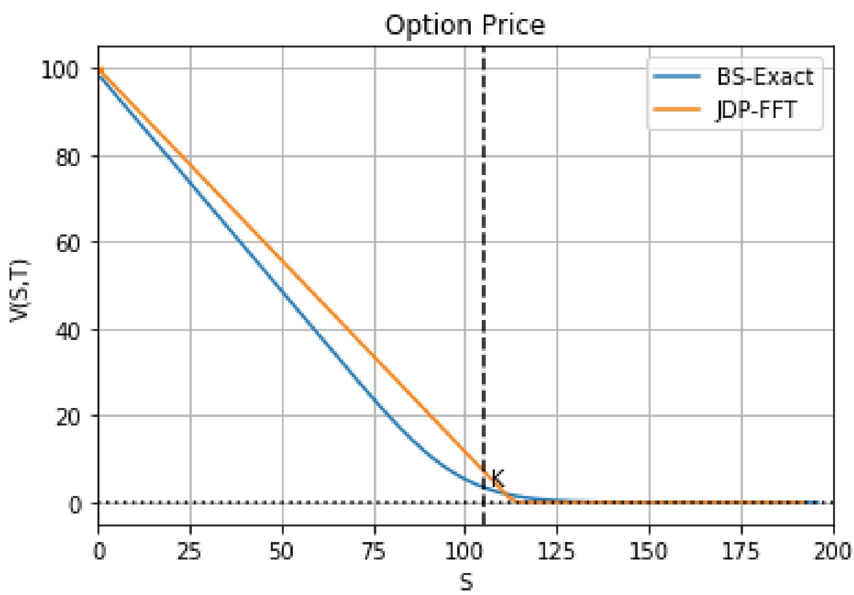

Figure 4.3: The exact and numerical simulation European vanilla put option.

Figure (4.4) below shows relative error of evaluating the European put option using the FFT approach.

This diagram was computed with the formula

$$\text{relative error} = \frac{|BS - FFT|}{|BS|} \tag{4.2.4}$$

where BS is the Black-Scholes exact solution and FFT is the numerical simulation using the FFT approach.

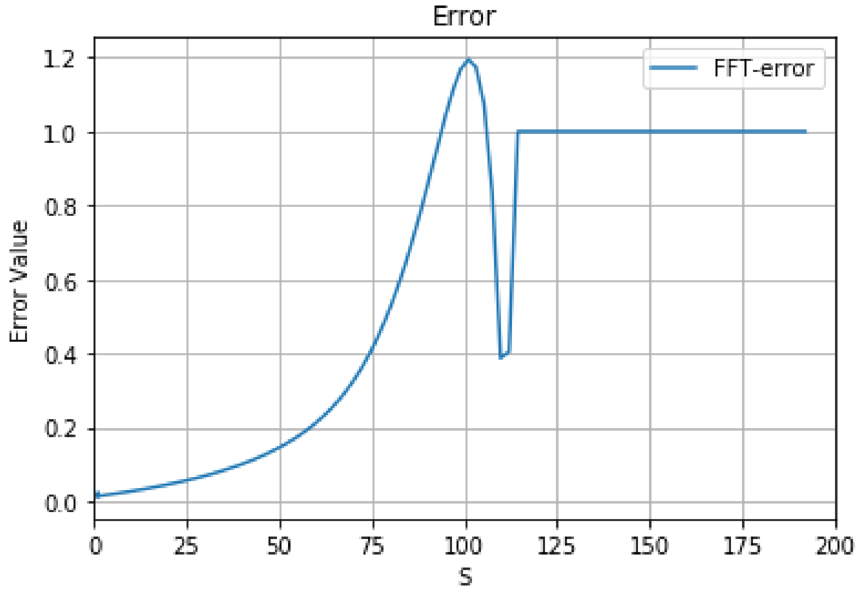

Figure 4.4: Error evaluating numerical simulation European vanilla put option compared to the exact value.

The figures show that the FFT method converges faster than the exact solution when asset price is almost the same as the strike price this signifies that the numerical approximation had an error that may be based on the choice of algorithm, computation of the integral term or the parameters. The relative error that showed in Figure (4.4) signifies that the computed option may be adjacent to the exact solution with less significant amount of error value. We can conclude that numerical simulation was not a success, by this remarks we can improve the simulation by using linear function approach instead of FFT, or even to further study the FFT approach to get better results on pricing an option.

5. Conclusion

In this essay, we consider the jump-diffusion model which is a stochastic differential equation (SDE) including the Poisson process. Thereby, using Itô Lemma we are able to derive a partial integro-differential equation (PIDE) which is solved numerically. The diffusion part and the advection part of the PIDE were discretised using the fitted finite volume method and the other terms including the integral terms were discretised using the mid-quadrature rule. After applying the mid-quadrature rule on the integral term, we used a linear interpolation to finalise the spatial discretization. To discretize fully we applied the explicit method which is the Crank-Nicolson method for better accuracy.

We had a dense matrix that resulted in the Crank-Nicolson scheme to be a non-linear, we developed a fast iterative method that was based on a regular splitting technique then the solution requires the Fast Fourier Transform (FFT) to reduce the computational cost. Then the FFT approach evaluates the correlation integral and also speeds up the process. The vanilla model was under investigation to check the robustness of the fitted finite volume method and also the numerical solution was compared with the exact solution of the Black-Scholes to check the accuracy and the convergence. The described methods were applied to find the price of European vanilla options which it was programmed using Python. We concluded that the numerical simulation converges faster than exact solution. For future work, we can apply the linear function approach to investigate the performance and observe that fitted finite volume method combined with the Crank–Nicolson scheme can be robust, this approach requires more time to compute.

Acknowledgements

I want to thank AIMS and it's funders for giving me this opportunity to write this essay, as well as my supervisor, Prof Phillip Mashele from North West University for guiding me throughout this essay. A special thank you to Rock Stephane KOFFI for taking his time to help me throughout this essay. To my tutor, Alice Nanyanzi, thank you for giving me the best advice. Lastly, a special thank you to the AIMS family without forgetting my family back home this is for you Msizo.

References

Almendral, A. and Oosterlee, C. W. Numerical valuation of options with jumps in the underlying. *Applied Numerical Mathematics*, 53(1):1–18, 2005.

Andersen, L. and Andreasen, J. Jump-diffusion processes: Volatility smile fitting and numerical methods for option pricing. *Review of derivatives research*, 4(3):231–262, 2000.

Black, F. and Scholes, M. The pricing of options and corporate liabilities. *Journal of political economy*, 81(3):637–654, 1973.

Cont, R. and Voltchkova, E. A finite difference scheme for option pricing in jump diffusion and exponential Lévy models. *SIAM Journal on Numerical Analysis*, 43(4):1596–1626, 2005.

Durrett, R. *Probability: theory and examples*, volume 49. Cambridge University Press, 2019.

Focardi, S. M. and Fabozzi, F. J. *The mathematics of financial modeling and investment management*, volume 138. John Wiley & Sons, 2004.

Forsyth, P. An introduction to computational finance without agonizing pain. *School of Computer Science, University of Waterloo*, 2005.

Fortran, I., Press, W., Teukolsky, S., Vetterling, W., and Flannery, B. Numerical recipes. *Cambridge, UK, Cambridge University Press*, 01 1992.

Frigo, M. and Johnson, S. G. FFTW: An adaptive software architecture for the FFT. In *Proc. 1998 IEEE Intl. Conf. Acoustics Speech and Signal Processing*, volume 3, pages 1381–1384. IEEE, 1998.

Guo, H. Review of applying European option pricing models. In *Proceedings of the 3rd Czech-China Scientific Conference 2017*. IntechOpen, 2017.

Heckbert, P. Fourier transforms and the Fast Fourier Transform (FFT) algorithm. *Computer Graphics*, 2:15–463, 1995.

Heideman, M., Johnson, D., and Burrus, C. Gauss and the history of the Fast Fourier Transform. *Archive for History of Exact Sciences*, 34:265–277, 01 1985. doi: 10.1007/BF00348431.

Hull, J. C. *Options futures and other derivatives*. Pearson Education India, 2003.

Investopedia, 2020. Options trading strategies: A guide for beginners. Elvin Mirzayev, Investopedia, https://www.investopedia.com/articles/active-trading/040915/guide-option-trading-strategies-beginners.asp, Accessed September, 2020a.

Investopedia, 2020. Options. James Chen, Investopedia, https://www.investopedia.com/terms/o/option.asp, Accessed September, 2020b.

Klar, L. and Jacobson, J. Pricing of European call options. *Uppsala: Uppsala University*, 2002.

Kloeden, P. E. and Platen, E. *Numerical solution of stochastic differential equations*, volume 23. Springer Science & Business Media, 2013.

Lesmana, D. and Wang, S. A numerical scheme for pricing American options with transaction costs under a jump diffusion process. *Journal of Industrial and Management Optimization*, 13:19–19, 12 2016. doi: 10.3934/jimo.2017019.

Merton, R. C. Option pricing when underlying stock returns are discontinuous. *Journal of financial economics*, 3(1-2):125–144, 1976.

Mukam, J. D. Stochastic calculus with jumps processes: Theory and numerical techniques. *arXiv preprint arXiv:1510.01236*, 2015.

Øksendal, B. Stochastic differential equations. In *Stochastic differential equations*, pages 65–84. Springer, 2003.

Probability Density Fuction, 2020. Continous random variable - probability density fuction(pdf). Brilliant, Wiki, https://brilliant.org/wiki/continuous-random-variables-probability-density/, Accessed September 2020.

Ross, S. M. *Introduction to probability models*. Academic Press, 2014.

Tavella, D. and Randall, C. Pricing financial instruments: the finite difference method, 2000. *John Willey & Sons, USA*, 2000.

Van Loan, C. *Computational frameworks for the Fast Fourier Transform*. SIAM, 1992.

Wang, G. and Yang, X. The regularization method for a degenerate parabolic variational inequality arising from American option valuation. *International Journal of Numerical Analysis and Modeling*, 5 (2):222–238, 2008.

Wilmott, P., Dewynne, J., and Howison, S. Option pricing: Mathematical models and computation 1993, 1993.

Zhang, K. and Wang, S. Pricing options under jump diffusion processes with fitted finite volume method. *Applied Mathematics and Computation*, 201(1-2):398–413, 2008.

Zhang, K. and Wang, S. A computational scheme for options under jump diffusion processes. *International journal of numerical analysis and modeling*, 6(1):110–123, 2009.