

# A Spectral Element Approach to Option Pricing

Wuming Zhu  
Advisor: Prof. Kopriva

July 18, 2006

## Abstract

We present a new approach to price options when the underlying asset follows a jump diffusion process. To solve the resulting Partial Integro-Differential Equation (PIDE), we start from its weak form and use the Gauss-Lobatto-Legendre points and quadrature to derive a system of ODEs. The system is solved by a direct method integrated with the trapezoidal rule as the time differencing scheme. The numerical results demonstrate that the method converges exponentially in the solution (spectrally accurate) with *a factor of 7 to 30* fewer spatial points and a factor of 5 fewer time steps than the finite difference method widely used in the financial industry.

<i>CONTENTS</i>	2
-----------------	---

## **Contents**

<b>1 Introduction</b>	<b>3</b>
<b>2 Jump Diffusion Model and PIDE</b>	<b>4</b>
2.1 Origin of the Model . . . . .	4
2.2 Derivation of the PIDE . . . . .	5
<b>3 Numerical Approximations</b>	<b>7</b>
3.1 Finite Difference Approach . . . . .	8
3.2 Spectral Element Approach . . . . .	9
3.3 Approximation of the convolution integral . . . . .	12
<b>4 Numerical results</b>	<b>15</b>
4.1 Convergence . . . . .	15
4.2 Speed . . . . .	21
<b>5 Conclusion</b>	<b>23</b>

## 1 Introduction

The pricing of American, path-dependent, and multi-asset options in the Black-Scholes-Merton (BSM) framework generally requires the use of numerical methods. Lattice and finite difference (and Monte Carlo simulation) methods are typically popular in financial derivative pricing.

Lattice methods start by discretizing the stochastic differential equation (SDE) and match the local mean and variance by choosing the appropriate up and down percentage changes,  $u$  and  $d$  and the associated probabilities. The method is easy to explain and to implement. Therefore, it is attractive for pedagogical purposes and for computing derivative prices in simple models. The four widely used lattices are CRR [10], Jarrow and Rudd [11], Amin [12], and Boyle [13]. Lattice methods can be viewed as explicit finite difference schemes that suffer slow convergence  $E = O(\Delta t)$ , and instability, i.e.  $\Delta t$  has to be made sufficiently small to keep the scheme absolutely stable.

The finite difference method (FDM) for option pricing was first proposed by Brennan and Schwartz [14], [15]. The advantages of this approach are as follows [9]: (1) There is a wealth of existing theory and algorithms; (2) Issues of numerical consistency, convergence, and stability have been well established; (3) Second or higher order convergence rate can be achieved; (For example, the Crank-Nicholson scheme with a central difference spatial approximation for derivatives in BSM framework.) (4) Derivatives in more complicated models can be handled, for example, when the underlying asset follows a jump-diffusion (JD) process or stochastic volatility model, see [3], [4].

The disadvantage of the FDM is that it needs hundreds of spatial points and time steps to achieve a reasonable accuracy. For example, in [3],[4], one basis point accuracy in the option price requires more than 500 spatial points and 250 time steps. This means it is hardly possible to extend the method to multi-asset options both for storage and for computational reasons.

In this work, we present a new approach, a Spectral Element Method (SEM), to price a European option in the JD model. It turns out that one basis point accuracy can be achieved by only about 60 spatial points and 40 time steps.

In the following sections, we first carefully introduce the JD model and then derive the PIDE for its contingent claim. Then, we discuss several numerical issues from other papers and describe our approach in detail. Finally, we analyze the numerical results and make a conclusion.

## 2 Jump Diffusion Model and PIDE

### 2.1 Origin of the Model

In the classical BSM framework, the stock price dynamics follows a geometric Brownian motion, which has a continuous sample path

$$dS/S = \mu dt + \sigma dz. \quad (1)$$

Here,  $S$  is the stock price at time  $t$ ,  $\mu$  is the instantaneous expected return on the stock,  $\sigma$  is the instantaneous volatility of the return,  $dz$  is the standard Brownian motion or Weiner process.

The continuity of the sample path indicates that the stock price can only change by a small amount in a short interval, whereas empirical studies tend to show too many outliers for a constant-variance log-normal distribution. To account for this, Merton [1] posited the total change in the stock price to be the composition of two types of changes:

1. The “normal” vibrations in price due, for example, to a temporary imbalance between supply and demand, changes in capitalization rates, changes in the economic outlook, or other new information that causes marginal change in the price. The component of this uncertainty is modeled by a standard geometric Brownian motion with constant variance and has a continuous sample path.
2. The “abnormal” vibrations in price due to the arrival of important new information about the stock that has more than a marginal effect on price. Usually, such information will be specific to the firm or possibly its industry and arrives only at the discrete points in time. This component is modeled by a jump process that reflects the non-marginal impact of the information.

The posited stock price returns can be formally written as the following SDE,

$$dS/S = (\mu - \lambda\kappa)dt + \sigma dz + dq. \quad (2)$$

Again,  $\mu$  is the instantaneous expected return on the stock;  $\lambda$  is the jump arrival rate;  $q(t)$  is the independent Poisson process;  $\kappa = E(\eta - 1)$  where  $E$  means “Expectation” and  $\eta - 1$  is the random percentage jump in the stock price if the Poisson event occurs. So  $(q(t) - \lambda\kappa t) = E(q(s) - \lambda\kappa s | F_t)$ ,  $s \geq t$  is a martingale, which is consistent with the general Efficient Market Hypothesis (EMH) [16]. The process can be viewed as superposition of a Brownian motion and a Poisson jump process.

## 2.2 Derivation of the PIDE

In [7], Kushner and Dupris give a version of the Itô formula that accommodates Poisson jumps. Let  $F$  be a twice-differentiable function of  $S$ , then

$$dF = (F_t + (\mu - \lambda\kappa)SF_S + \frac{1}{2}\sigma^2 S^2 F_{SS})dt + \sigma SF_S dz + dQ_F \quad (3)$$

where  $dQ_F = F(S(t)) - F(S(t^-))$ . Here, as an example, we use (3) to derive the formula for  $S(t)$ . Let  $F = \ln S$ , then

$$d\ln S = (\mu - \lambda\kappa - \frac{1}{2}\sigma^2)dt + \sigma dz + \ln S(t) - \ln S(t^-).$$

Assume all the parameters are constant so that when we integrate both sides, we get

$$S(t) = S \cdot \exp[(\mu - \lambda\kappa - \frac{1}{2}\sigma^2)t + \sigma \cdot z] \prod_{i=1}^{n(t)} \eta_i$$

where  $n(t)$  is a Poisson random variable with mean  $\lambda t$  and  $\eta_i - 1$  is a random percentage jump from a specified distribution.

Now, according to (2), we can write the option-return dynamics in a similar form as

$$dF/F = (\mu_F - \lambda\kappa_F)dt + \sigma_F dz + dq_F \quad (4)$$

where  $\kappa_F = E(\frac{F(S\eta,t) - F(S,t)}{F(S,t)}) = \frac{E(F(S\eta,t))}{F(S,t)} - 1$ ;  $q_F(t)$  is the jump process of option-return induced by  $q(t)$ . Namely, the Poisson event for the option price occurs if and only if the Poisson event for the stock price occurs. Matching the first two components of (3) and (4), we get

$$\mu_F = \frac{F_t + (\mu - \lambda\kappa)SF_S + \frac{1}{2}\sigma^2 S^2 F_{SS} + \lambda\kappa_F F}{F} \quad (5)$$

$$\sigma_F = \frac{\sigma SF_S}{F} \quad (6)$$

To get the pricing equation, we consider a portfolio strategy that holds the stock, the option and the riskless asset, whose weights are  $W_S$ ,  $W_F$  and  $1 - W_S - W_F$ , respectively. In the Black-Scholes analysis, the portfolio can be made riskless, and earn risk-free rate, i.e.,

$$\begin{cases} W_S \sigma + W_F \sigma_F = 0 \\ W_S(\mu - r) + W_F(\mu_F - r) + r = r \end{cases} \quad (7)$$

This gives us the following important result:

$$\frac{\mu - r}{\sigma} = \frac{\mu_F - r}{\sigma_F} \quad (8)$$

which is known as the market price of risk. By substituting (5) with  $\lambda = 0$  and (6) into (8), we will get the standard Black-Scholes equation,

$$F_t + rSF_S + \frac{1}{2}\sigma^2S^2F_{SS} - rF = 0 \quad (9)$$

Unfortunately, the portfolio will not be riskless in the presence of a jump process. To see this, we know from equation (4) that the Poisson event for the option price occurs if and only if the Poisson event for the stock price occurs. Since the option price is a nonlinear function of the stock price, while the portfolio mixing is linear and the arrival of the Poisson jump is random, the risk of portfolio cannot be completely hedged away.

However, if we assume the Weiner process represents “systematic” risk and the jump component of the stock’s return represents “nonsystematic” risk, then we can form a zero-beta portfolio by choosing  $W_S$  and  $W_F$ , so that there’s no “systematic” risk. That is,  $W_S$  and  $W_F$  satisfy the first equation in (7). Furthermore, if we assume that Capital Asset Pricing Model (CAPM)<sup>1</sup> is a valid description of equilibrium security returns, then the expected return on all zero-beta portfolios must equal the riskless rate. That means the second equation in (7) will be valid, which implies (8) must be satisfied.<sup>2</sup>

Therefore, by substituting (5) and (6) into (8), we get the following PDE for the option price:

$$F_t + (r - \lambda\kappa)SF_S + \frac{1}{2}\sigma^2S^2F_{SS} - rF + \lambda\kappa_FF = 0.$$

Substituting  $\kappa_F$  into it gives

$$F_t + (r - \lambda\kappa)SF_S + \frac{1}{2}\sigma^2S^2F_{SS} - rF + \lambda[E(F(S\eta, t)) - F(S, t)] = 0$$

---

<sup>1</sup>The main result of the CAPM is:  $E(r) - r_f = \beta[E(r_M) - r_f]$ , where E means “Expectation”,  $r$  is the return of a risky asset,  $r_f$  is the risk-free rate,  $r_M$  is the return of the market portfolio and  $\beta = \frac{Cov(r, r_M)}{\sigma_M^2}$ .

<sup>2</sup>This assumption is crucial. It means the resulting equation cannot be applied to the derivatives on the market portfolio. Please see [5] for options on the market portfolio with discontinuous returns.

If we assume the percentage jump size is log-normally distributed, then the above equation becomes

$$F_t + (r - \lambda\kappa)SF_S + \frac{1}{2}\sigma^2 S^2 F_{SS} - (r + \lambda)F + \lambda \int_0^\infty F(S\eta, t)G(\eta)d\eta = 0 \quad (10)$$

where

$$G(\eta) = \frac{\exp(-\frac{1}{2}(\frac{\ln(\eta)-m}{\delta})^2)}{\sqrt{2\pi}\delta\eta} \quad (11)$$

and  $m$  is the mean of  $\ln(\eta)$  and  $\delta^2$  is the variance of  $\ln(\eta)$ . Since (10) contains an integral, we call it a ‘‘Partial Integro-Differential Equation’’ (PIDE).

The closed-form solution for a European option from (10) can be written as

$$c = \sum_{n=0}^{\infty} \frac{e^{-\lambda'T} (\lambda'T)^n}{n!} f_n \quad (12)$$

where  $\lambda' = \lambda(1 + \kappa)$ ,  $\kappa = E(\eta - 1) = \exp(m + \frac{1}{2}\sigma^2) - 1$ , and  $T$  is the maturity. The variable  $f_n$  is the Black-Scholes option price where the instantaneous variance rate is  $\sigma^2 + \frac{n\delta^2}{T}$ , and the risk-free rate is  $r - \lambda\kappa + \frac{n\gamma}{T}$ , where  $\gamma = \ln(1 + \kappa)$ . The formula, (12) is actually the sum of a series of Black-Scholes option prices weighted by the Poisson probabilities,  $\frac{e^{-\lambda'T} (\lambda'T)^n}{n!}$ .

### 3 Numerical Approximations

Though the closed-form solution for European option is available, a numerical approach is necessary to calculate American option prices. Here, we only give an algorithm for a European put option. This serves as a starting point for the future, more interesting work.

A European put option price is the solution of the following initial boundary value problem (IBVP),

$$\begin{cases} F_\tau = (r - \lambda\kappa)SF_S + \frac{1}{2}\sigma^2 S^2 F_{SS} - (\lambda + r)F + \lambda \int_0^\infty F(S\eta, t)G(\eta)d\eta \\ F(S, 0) = \max(K - S, 0) \\ F(0, \tau) = Ke^{-r\tau} \\ F(S_m, \tau) \approx 0 \end{cases} \quad (13)$$

where  $\tau = T - t$ ,  $K$  is the strike price, and  $S_m$  is the maximum stock price on the grid, which is chosen to be large enough to safely set the asymptotic value at the boundary.

### 3.1 Finite Difference Approach

To date, several papers and a book have used the finite difference method to solve (13). The challenge when jump diffusion is present is to compute the convolution integral efficiently.

Tavella and Randall [2] evaluate the truncated integral on a different spatial grid (in log) by interpolating and extrapolating the values on the original spatial grid and then using a trapezoidal rule for each partitioned integral. This results in a dense matrix for a vector of option prices. In order to get second order accuracy in time and to avoid directly solving the dense matrix system, they then use so-called “lagging” (iteration) along with a tri-diagonal matrix solver to solve the system at each time step, i.e.

$$M_1 \vec{F}^{i,j+1} - A \vec{F}^{i,j} = M_2 \vec{F}^i + A \vec{F}^i$$

where  $A \vec{F}^i = \int_{m-5\delta}^{m+5\delta} \vec{F}^i(\vec{S}e^x)G(x)dx$ ,  $M_1, M_2$  are both tridiagonal matrices, and  $\vec{F}^{i,j}$  is the vector of the intermediate values at the  $j$ th iteration,  $i$ th time step. This is essentially a “splitting” iteration method. Because of the excellent initial guess (discounted value), it only requires two or three iterations at each time step. However, as the number of spatial points increases, the matrix vector multiplication gets expensive.

Andersen and Andreasen [3] use an  $FFT$  to evaluate the convolution integral and solve the system explicitly in time (1st order accurate). Note:

$$FT\left(\int_{-\infty}^{\infty} F(e^{\ln S+x})G(x)dx\right) = FT(F)FT(G)$$

where  $FT$  means Fourier Transform. So two forward and one backward  $FFT$  gives a set of integral values. This is a big improvement on the speed due to the complexity of  $FFT$ ,  $O(n \log_2 n)$ .

D’Halluin, Forsyth, and Labahn [4] use both iteration and an  $FFT$  to price the corresponding American option and provide a detailed proof on the convergence of the iteration. Their method is fast and second order accurate.

However, by using  $FFT$ , the method introduces aliasing errors, i.e. low frequencies couple with high frequencies for every  $n$  mode, where  $n$  is the number of spatial points. Also, to be accurate to a basis point, it needs more than 500 spatial points, 216 time steps, and total 456 iterations.<sup>3</sup> To further decrease the error by a factor of 4, it needs to double both the number of spatial points and time steps. So, it’s hard to extend the method to multi-asset options.

---

<sup>3</sup>These numbers are from [4], page 345, Table 3.

### 3.2 Spectral Element Approach

Unlike the finite difference method, a spectral method uses an orthogonal polynomial expansion to approximate the solution globally. If the initial condition is smooth enough on the whole domain, then the solution will converge faster than any polynomial order in  $1/N$ , where  $N$  is the number of the spatial points.<sup>4</sup> This is known as “infinite order” or “spectral” accuracy. In practice, the convergence rate of a spectral method usually appears as exponential for smooth enough solutions.

Unfortunately, the option payoff function,  $F(S, 0) = \max(K - S, 0)$ , is not smooth, i.e. there is a jump discontinuity in its first derivative. If we directly apply the spectral method, we will only get slow convergence. This is where the spectral element method (SEM) comes into play. Since the derivative of  $F(S, 0)$  is piecewise smooth, we can start with the weak form of (13), break up the domain into several pieces where the derivative is smooth, and then use the spectral method on each sub-domain.

The weak form of (13) is

$$\begin{aligned} \int_0^{S_m} F_\tau \phi dS &= \int_0^{S_m} ((\mu - \lambda\kappa)SF_S + \frac{1}{2}\sigma^2 S^2 F_{SS} - (\lambda + r)F)\phi dS \\ &\quad + \lambda \int_0^{S_m} \int_0^\infty F(S\eta, t)G(\eta)d\eta\phi dS, \end{aligned} \quad (14)$$

for all  $\phi \in H_B^1(0, S_m)$ , where

$$H_B^1(0, S_m) = \left\{ \psi \in L^2(0, S_m) \left| \frac{d\psi}{dx} \in L^2(0, S_m), \psi(0) = Ke^{-r\tau}, \psi(S_m) = 0 \right. \right\}$$

The function  $\phi$ , which satisfies the boundary conditions, is called a test function.

To illustrate the method and to avoid messy algebra, we consider the following simplified equation,

$$\int_0^{S_m} F_\tau \phi dS = \int_0^{S_m} S^2 F_{SS} \phi dS$$

with  $\phi(0) = \phi(S_m) = 0$ .<sup>5</sup>

---

<sup>4</sup>In this problem, it is to be the number of the Gauss-Lobatto-Legendre points.

<sup>5</sup>It turns out that setting  $\phi(0) = \phi(S_m) = 0$  does not change the results in this problem. We do so to simplify the algebra.

Integrating by parts and applying the boundary conditions of  $\phi$ , we get

$$\int_0^{S_m} F_\tau \phi dS = - \int_0^{S_m} S^2 F_S \phi_S dS - \int_0^{S_m} 2SF_S \phi dS \quad (15)$$

Let's consider two elements  $\Omega^L = [0, K]$  and  $\Omega^R = [K, S_m]$  that split the domain into two intervals of length  $d^L = K$ ,  $d^R = S_m - K$ . Now we break up each integral in (15) into two parts, namely

$$\begin{aligned} \int_0^K F_\tau \phi dS + \int_K^{S_m} F_\tau \phi dS &= - \int_0^K S^2 F_S \phi_S dS - \int_K^{S_m} S^2 F_S \phi_S dS \\ &\quad - \int_0^K 2SF_S \phi dS - \int_K^{S_m} 2SF_S \phi dS \end{aligned} \quad (16)$$

so that we can approximate the solution on each side by an orthogonal expansion of Legendre polynomials and map each element onto the "standard interval"  $[-1, 1]$  by an affine transformation:

$$\xi = \begin{cases} \frac{2}{d^L} S - 1, S \in \Omega^L \\ \frac{2}{d^R} (S - K) - 1, S \in \Omega^R \end{cases}, \quad S = \begin{cases} \frac{d^L}{2} (\xi + 1), S \in \Omega^L \\ \frac{d^R}{2} (\xi + 1) + K, S \in \Omega^R \end{cases}$$

Then the derivatives and the integrals are transformed by

$$dS = \begin{cases} \frac{d^L}{2} d\xi, S \in \Omega^L \\ \frac{d^R}{2} d\xi, S \in \Omega^R \end{cases}, \quad \frac{\partial}{\partial S} = \frac{\partial}{\partial \xi} \frac{\partial \xi}{\partial S} = \begin{cases} \frac{2}{d^L} \frac{\partial}{\partial \xi}, S \in \Omega^L \\ \frac{2}{d^R} \frac{\partial}{\partial \xi}, S \in \Omega^R \end{cases}$$

to give

$$\begin{aligned} \frac{d^L}{2} \int_{-1}^1 F_\tau^L \phi^L d\xi + \frac{d^R}{2} \int_{-1}^1 F_\tau^R \phi^R d\xi &= \\ - \frac{d^L}{2} \int_{-1}^1 (\xi + 1)^2 F_\xi^L \phi_\xi^L d\xi - \frac{2}{d^R} \int_{-1}^1 \left(\frac{d^R}{2}(\xi + 1) + K\right)^2 F_\xi^R \phi_\xi^R d\xi & \\ - 2 \frac{d^L}{2} \int_{-1}^1 (\xi + 1) F_\xi^L \phi^L d\xi - 2 \int_{-1}^1 \left(\frac{d^R}{2}(\xi + 1) + K\right) F_\xi^R \phi^R d\xi & \end{aligned} \quad (17)$$

The polynomial interpolants of these functions and their derivatives in Lagrange form are

$$\begin{aligned} F(\xi, t) &= \sum_{j=0}^N F_j(t) l_j(\xi), & \phi(\xi, t) &= \sum_{j=0}^N \phi_j(t) l_j(\xi) \\ F_\xi(\xi, t) &= \sum_{j=0}^N F_j(t) l'_j(\xi), & \phi_\xi(\xi, t) &= \sum_{j=0}^N \phi_j(t) l'_j(\xi) \end{aligned}$$

where  $l_j = \prod_{i=0, i \neq j}^N \frac{(\xi - \xi_i)}{(\xi_j - \xi_i)}$ .

By choosing  $\xi_i (i = 0, 1, \dots, N)$  to be the Gauss-Lobatto-Legendre points, the integrals in (17) approximated by the Gauss-Lobatto-Legendre quadrature will be exact for polynomials of degree,  $2N - 1$  or less.<sup>6</sup>

For example,

$$\int_{-1}^1 F_\xi^L \phi_\xi^L d\xi = \sum_{j=0}^N F_\xi^L |_j \phi_\xi^L |_j w_j$$

where  $w_j$  is the quadrature weight, is exact because the integrand is a polynomial of degree  $2N - 2$ , while

$$\int_{-1}^1 F_\tau^L \phi^L d\xi \approx \sum_{j=0}^N \dot{F}_j^L \phi_j^L w_j$$

is not exact because the integrand is a polynomial of degree  $2N$ . Nevertheless, the error is spectrally small.

Replacing each integral in (17) with the Gauss-lobatto-Legendre quadrature and rearranging the terms, we end up with the following equation,

$$\begin{aligned} & \frac{d^L}{2} \sum_{j=1}^N \dot{F}_j^L \phi_j^L w_j + \frac{d^R}{2} \sum_{j=0}^{N-1} \dot{F}_j^R \phi_j^R w_j = \\ & - \frac{d^L}{2} \sum_{m=1}^N \phi_m^L \sum_{j=0}^N F_\xi^L |_j l'_m(\xi_j) (\xi_j + 1)^2 w_j \\ & - \frac{2}{d^R} \sum_{m=0}^{N-1} \phi_m^R \sum_{j=0}^N F_\xi^R |_j l'_m(\xi_j) [(\xi_j + 1) \frac{d^R}{2} + K]^2 w_j \\ & - d^L \sum_{j=1}^N F_\xi^L |_j \phi_j^L (\xi_j + 1) w_j - 2 \sum_{j=1}^{N-1} F_\xi^R |_j \phi_j^R [(\xi_j + 1) \frac{d^R}{2} + K] w_j \end{aligned}$$

This has to be true for all  $\phi \in H_0^1$  and  $\phi_j (j = 1, \dots, N - 1)$  are interior, so they are linear independent, which means each factor of  $\phi_j$  must match. (Note:  $\phi_N^L = \phi_0^R$ .) Applying the similar method to each integral in (14) except the last integral containing the jump component, we get a system of ODEs in matrix form,

$$W\dot{F} = BF + \vec{\beta}, \tag{18}$$

where  $\vec{\beta}$  includes the boundary conditions,

---

<sup>6</sup>The Gauss-Lobatto-Legendre points are  $-1, +1$ , plus zeros of  $L'_N(x)$  on  $(-1, 1)$ , where  $L_N(x)$  is the Legendre polynomial of degree  $N$ .





which is shown below. For  $0 \leq x \leq \frac{K}{S_m}$ , we approximate the second integral in  $I_1(x)$  the same way as before, but the first integral as follows,

$$\begin{aligned} & \frac{1}{x} \int_0^{\frac{K}{S_m}} F(zS_m)G\left(\frac{z}{x}\right)dz = \frac{1}{x} \int_0^{\frac{K}{S_m}} \sum_{j=0}^N F_j^L l_j^L(z)G\left(\frac{z}{x}\right)dz \\ &= \frac{1}{x} \sum_{j=0}^N F_j^L \int_0^{\frac{K}{S_m}} l_j^L(z)G\left(\frac{z}{x}\right)dz = \frac{1}{x} \sum_{j=0}^N F_j^L \int_{-1}^1 l_j^L\left(\frac{K(\xi+1)}{2S_m}\right)G\left(\frac{K(\xi+1)}{2S_mx}\right)\frac{K}{2S_m}d\xi \\ &= \frac{1}{x} \sum_{j=0}^N F_j^L \sum_{k=1}^M l_j^L\left(\frac{(\xi_k+1)d^L}{2(d^R+d^L)}\right)G\left(\frac{(\xi_k+1)d^L}{2(d^R+d^L)x}\right)\frac{d^L}{2(d^R+d^L)}w_k \end{aligned}$$

where  $M$  is chosen large enough relative to  $N$ , for example,  $M = 4N$ . Similarly, it can be written in matrix form,  $G \cdot w \cdot L \cdot \vec{F}^L$ , where  $G \cdot w \cdot L$  only needs to be computed once and stored.

To compute  $I_2(x) = \frac{1}{x} \int_1^\infty F(z, t)G\left(\frac{z}{x}\right)dz$ , we have considered two approaches. First, since the stock price is outside the spatial grid, i.e.  $S > S_m$ , we use the asymptotic linearity of the option value at the right boundary to calculate  $I_2(x)$  analytically. Let  $F(z, t) = az + b$ , then

$$I_2(x) = \frac{1}{x} \int_1^\infty (az + b)G\left(\frac{z}{x}\right)dz$$

where  $a, b$  can be estimated from the computed solution at the right boundary.<sup>8</sup> This method tends to underestimate the integral due to the convexity of the option price. The error will grow substantially as  $m, \delta, \lambda$  get large.

The second approach is to add a third element that makes  $S_m$  very large so that  $I_2(x)$  for every  $x$  in the region of interest,  $x \in [0, \frac{2K}{S_m}]$  or  $S \in [0, 2K]$ , can be accurately represented by the integral on the third element. Furthermore, by choosing  $S_m$  sufficiently large, we can now safely set the right boundary condition to its asymptotic value that will have little effect on the solution in the region of interest.

Now, grouping all the terms, we get a final ODE system in matrix form below,

$$W \cdot \dot{\vec{F}} = BI \cdot \vec{F} + \vec{\beta} \quad (20)$$

where  $BI$  is now a dense matrix and  $\vec{\beta}$  is the vector of boundary conditions. Crank-Nicholson time differencing scheme (trapezoidal rule in time) can be

---

<sup>8</sup>Of course, we cannot integrate it all the way to  $\infty$ , since it will make the integrand negative, eventually. So, by restricting the integral within the domain  $[1, -b/a]$ , we can improve the linear approximation. However, the method highly depends on the accuracy of the boundary value on the right.

applied to (20), which becomes

$$(I - \frac{\Delta t}{2}W^{-1}BI)\vec{F}^{n+1} = (I + \frac{\Delta t}{2}W^{-1}BI)\vec{F}^n + \frac{\Delta t}{2}W^{-1}(\vec{\beta}^{n+1} + \vec{\beta}^n)$$

Since the size of the matrix is small, around  $60 \times 60$ , we can compute an  $LU$  factorization once at the beginning and solve the system quickly by a forward and backward substitution method.

## 4 Numerical results

Before we present our numerical results, let us first look at the behavior of the European put option price in the JD model. Table 1 lists the parameters used in the put example.

Table 1: Parameters used in the put example

$T$	$r$	$K$	$\sigma$	$\lambda$	$m$	$\delta$
0.25	0.05	100	0.15	0.10	-0.9	0.45

Figure 1 shows the analytical solutions of the European put options in the standard BS model and in the JD model and their differences. We can see that the option is more valuable in the JD model than in the BS model. It is important to note that in the presence of the jump component, the value of the out-of-the-money put option<sup>9</sup> increases substantially.

It is also interesting to compare their deltas,  $\Delta = \frac{\partial F}{\partial S}$ , and their gammas,  $\Gamma = \frac{\partial^2 F}{\partial S^2}$ . Figure 2 shows the differences. Note that the option price is less sensitive to the stock price near  $S = K$  and has more curvature when  $S < K$  in the JD model.

### 4.1 Convergence

Figure 3 compares the numerical solution with the analytical solution of a European put option in the standard BS model. Clearly, the numerical solution is graphically accurate with only 32 total spatial points and 40 time steps.

Figure 4 shows the maximum error of a European put option in the standard BS model. We can see for small  $NX$ , where  $NX$  is the number of total spatial points, it decreases exponentially for all  $NT = 40, 80, 160, 320$ , where  $NT$  is the number of time steps. It then stops and stays at a constant

---

<sup>9</sup> $S \gg K$

Figure 1: *Analytical prices of the European put options in both the BS model and the JD model and their differences. The parameters as in Table 1.*

Figure 2: *Analytical values of the European put Delta and Gamma in both the BS model and the JD model. The parameters as in Table 1.*

level due to the time differencing error. Also, we can see as  $NT$  doubles, the error decreases by a factor of 4, which is consistent with the order (second) of the time differencing scheme.

Figure 5 compares the convergence rate of the SEM with that of the FDM. By fixing  $NX = 2NT$ , the FDM achieves second order convergence (Slope = -2), which means by doubling both  $NX$  and  $NT$  simultaneously, the error decreases by a factor of 4. As we can see from the right graph, an error of  $0.2554 \cdot 10^{-3}$  requires  $NX = 1024$ ,  $NT = 512$  for the FDM, but

Figure 3: *Numerical and analytical solution for European put option in the standard BS model.  $NX = 16 \times 2$ ;  $NT = 40$ ; SEM(2): Elements on  $[0, K]$  and  $[K, 2K]$ ;  $S_m = 2K$ ;  $F(S_m, t) = 0$ . Parameters as in Table 1.*

Figure 4: *Convergence of SEM(2) for the European put option in the Standard BS model. SEM(2): Elements on  $[0, K]$  and  $[K, 2K]$ ;  $S_m = 2K$ ;  $F(S_m, t) = 0$ . Parameters as in Table 1.*

only requires  $NX = 30$ ,<sup>10</sup>  $NT = 40$  for the SEM with 2 elements. The left graph is the semi-log plot that emphasizes the difference between the two convergence rates.

Although the complexity is  $O(NX^2 \cdot NT)$  for the SEM and  $O(NX \cdot NT)$  for the FDM, the huge differences in  $NX$ ,  $NT$  make the SEM more attractive even for 1-D problem.

As we mentioned in Section 3.3, the oscillations at the left boundary for

---

<sup>10</sup>Note:  $NX$  is the total number of spatial points.

Figure 5: *Convergence Comparison of SEM and FDM. For SEM,  $NT = 40$ ; For FDM,  $NT = NX/2$ .  $S_m = 2K$ ;  $F(S_m, t) = 0$ . Parameters as Table 1.*

Figure 6: *Oscillations at the left boundary of JD put.  $NT = 60$ ;  $S_m = 0$ . SEM(3): Elements on  $[0, K]$ ,  $[K, 2K]$  and  $[2K, 20K]$ . Parameters as in Table 1.*

the European option in the JD model can be removed by over-integration on the left. Figure 6 demonstrates the oscillation error decreases roughly spectrally. Note, the over-integration only adds setup work, which means the corresponding matrix only needs to be computed once and stored.

Figure 7 compares the solutions using SEM(3)<sup>11</sup> with those using SEM(2) with a slightly large jump arrival rate,  $\lambda = 0.5$ . SEM(2) uses the linear approximation for  $I_2$  along with the analytical right boundary condition. As we can see, the trough near the right boundary shows that the linear

---

<sup>11</sup>3 elements.

approximation underestimates  $I_2$ . However, SEM(3) gives an excellent fit by extending the 3rd element all the way to  $S_m = 20K$  so that  $I_2(S)$  for  $S \in [0, 2K]$  can be represented accurately enough by the integral over  $[2K, 20K]$ .<sup>12</sup>

Figure 7: *Comparison between SEM(2) and SEM(3). For SEM(2): Elements on  $[0, K]$ ,  $[K, 2K]$  with  $NX = 24 \times 2$  and  $S_m = \text{Analytical value}$ ; For SEM(3): Elements on  $[0, K]$ ,  $[K, 2K]$  and  $[2K, 20K]$  with  $NX = 24 \times 3$  and  $S_m = 0$ . Parameters as in Table 1.*

Another nice thing about SEM(3) is the way it handles the right boundary condition. Unlike the BS case, in which the put value decreases to zero fast as  $S$  gets large, the put value in the JD model goes to zero fairly slowly due to the jump integral term, as shown in “Figure 1”. But the SEM allows us the flexibility to add extra elements to reach the asymptotic value so that it has little effect on the region of interest, for example,  $S \in [0, 2K]$ .

Figure 8 shows the numerical and analytical solutions of a European put option in the JD model. Again, the numerical solution is graphically accurate with only 48 total spatial points and 40 time steps. Figure 9 shows a roughly exponential convergence of the JD put using SEM(3).

Usually, the derivatives of the numerical solution are subject to larger errors even though the solution looks correct. Therefore, it is also important to look at the behavior of its derivatives, delta and gamma. Figure 10 shows the computed numerical delta function and gamma function along with the

---

<sup>12</sup>How far the third element needs to extend on the right will depend on the distribution of the jump size,  $G(\eta)$ , and the arrival rate of the jump,  $\lambda$ . Also, the number of the quadrature points required by the third element will depend on the overall curvature on the extended domain. These issues are still heuristic at this point and they will be examined in the future research.

Figure 8: *Numerical and analytical solutions of the European put options in the JD model.  $NX = 16 \times 3$ ;  $NT = 40$ ;  $SEM(3)$ : Elements on  $[0, K]$ ,  $[K, 2K]$  and  $[2K, 20K]$  with  $S_m = 0$ . The parameters as in Table 1.*

Figure 9: *Convergence of  $SEM(3)$ . Maximum errors on  $[0, 2K]$ .  $SEM(3)$ : Elements on  $[0, K]$ ,  $[K, 2K]$  and  $[2K, 20K]$  with  $S_m = 0$ . Parameters as in Table 1.*

analytical ones. Clearly, the numerical solutions are graphically accurate. Figure 11 compares the convergence of the original put option, its delta function and its gamma function. As we can see, the convergence of delta and gamma is not smooth with a period of about 12 spatial points. However, they still show the overall exponential convergence with slightly less steep slopes.

Figure 10: *Numerical and Analytical Delta and Gamma using SEM(3).  $NX = 20 \times 3$ ;  $NT = 40$ . SEM(3): Elements on  $[0, K]$ ,  $[K, 2K]$  and  $[2K, 20K]$  with  $S_m = 0$ . Parameters as in Table 1.*

Figure 11: *Convergence of Put option, its Delta and its Gamma using SEM(3).  $NT = 320$ . SEM(3): Elements on  $[0, K]$ ,  $[K, 2K]$  and  $[2K, 20K]$  with  $S_m = 0$ . Parameters as in Table 1.*

## 4.2 Speed

The complexity of the algorithm is a necessarily crude approach to the measuring of program efficiency. To really compare the speeds of the SEM and the FD, we do the following experiment. Table 2 and Table 3 summarize the results.

Table 2 is basically from the paper by D'Halluin, Forsyth and Labahn

Table 2: Finite difference with Crank-Nicolson

Nodes	Timesteps	Itns	Value	Error @ $S = K$	FFT Time(Sec)
508	216	456	3.14889738	0.00012836	0.005
1016	448	896	3.14899401	0.00003173	0.02
2032	913	1826	3.14901783	0.00000791	0.09

Table 3: SEM(3) with Crank-Nicolson

Nodes	Timesteps	Max Error	For...Loop Time(Sec)
$24 \times 3$	40	0.00006253	0.005
$24 \times 3$	80	0.00001554	0.01
$24 \times 3$	160	0.00000631	0.02

[4]<sup>13</sup> with the extra column, “FFT Time”. The first 5 columns show their numerical results on an European put option<sup>14</sup> with the parameters in Table 1. The extra column, “FFT Time”, shows the required computational time to run an FFT alone with the corresponding number of nodes and the number of the total iterations. This computational time does not include the time required to solve the tridiagonal system at each time step. Table 3 shows the numerical results using SEM(3) with the same set of parameters. The last column shows the required computational time to solve the whole system forward in time. We can see that with the same level of accuracy,<sup>15</sup> SEM(3) becomes more and more efficient than FDM as more accuracy is needed.

Solving the PIDE results in a set of option prices for a given strike price,  $K$ . In reality, we are given a stock price at a particular time and would like to know a set of option price for different strike prices. With the free scaling property of this PIDE, all we need to determine is the ratio,  $S/K$ , and then we can easily read its relative value off the grid by interpolation, and finally calculate the desired price by multiplying the relative value with the strike price. For example, if we calculate the option prices for  $S \in [0, 2K]$ , then it actually gives all the option prices for  $K \in [\frac{S}{2}, \infty)$ . This uniform spectral accuracy from SEM guarantees that the interpolation anywhere on the domain,  $S \in [0, 2K]$ , will be accurate enough. Therefore, the solution

---

<sup>13</sup>Page 346, Table 3

<sup>14</sup>Its analytical value is 3.14902574.

<sup>15</sup>Here, we already lose the comparison conditions. Note that the errors in Table 2 are point-wise at  $S = K$ , while the errors in Table 3 are the maximum errors in the whole region of interest,  $S \in [0, 2K]$ .

only need to be computed once for every fixed set of parameters and the number of option prices needs to be stored is less than 100. This is another big advantage over the FDM when pricing multi-asset options.

## 5 Conclusion

We have presented a spectral element approach for pricing European options in Jump diffusion model. It turns out the method is both more efficient and more accurate than the popular finite difference method used in the financial industry. The basic methodology of the method is straightforward and it's easy to code up given a standard subroutine generating the Gauss-Lobatto-Legendre points and weights.

We expect the practitioners would start using this approach in the near future.

*Acknowledgements.* I greatly appreciate Dr. Kopriva's ideas and tremendous help on this work.

## References

- [1] Robert C. Merton : *Continuous Time Finance*, 1990
- [2] Domingo Tavella, Curt Randall : *Pricing Financial Instruments: The Finite Difference Method*, 2000
- [3] Andersen, Andreasen : *Jump-Diffusion Processes: Volatility Smile Fitting and Numerical Methods for Option Pricing*, Review of Derivatives Research, 4, 231-262, 2000.
- [4] d'Halluin, Y, Forsyth, P.A., Labahn, G. : *A penalty method for American options with jump diffusion processes*, Numerische Mathematik 97: 321-352, 2004.
- [5] Naik, Lee : *General equilibrium pricing of options on the market portfolio with discontinuous returns*, Review of Financial Studies, 3, 493-521, 1990.
- [6] John C. Hull: *Options, Futures and other derivatives*, 4th Edition, 2000.
- [7] Kushner, Dupuis: *Numerical Methods for Stochastic Control Problems in Continuous Time*, 2nd Edition, 1992.

- [8] Kopriva, Hussaini: *Spectral methods for scientists and engineers*, “Upcoming”.
- [9] Broadie, Detemple: *Option Pricing: Valuation models and applications*, Management Science, Vol. 50, No. 9, 1145-1177, 2004.
- [10] Cox, Ross, Rubinstein: *Option pricing: A simplified approach*, J. Financial Econom. 7 229-263, 1979.
- [11] Jarrow, Rudd: *Option Pricing*, Dow Jones-Irwin, Homewood, IL. 1983.
- [12] Amin: *On the computation of continuous time option prices using discrete approximations*, J. Financial Quant. Anal. 26(4) 477-496. 1991.
- [13] Boyle: *Option valuation using a three-jump process*, Internat. Options J. 3 7-12. 1986.
- [14] Brennan, Schwartz: *The valuation of American put options*, J. Finance 32 449-462. 1977.
- [15] Brennan, Schwartz: *Finite difference methods and jump processes arising in the pricing of contingent claims: A synthesis*, J. Financial Quant. Anal. 13 461-474.
- [16] Bodie, Kane, Marcus: *Investments*, 5th Edition, 2002.