Proceedings of the 2018 Winter Simulation Conference
M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, eds.

EUROPEAN OPTION PRICING WITH STOCHASTIC VOLATILITY AND JUMPS: COMPARISON OF MONTE CARLO AND FAST FOURIER TRANSFORM METHODS

Uro Lyi

7030 Preinkert Drive
Prince Fredrick Hall
University of Maryland
College Park, MD 20742, USA

Michael C. Fu

Robert H. Smith School of Business
Institute of Systems Research
University of Maryland
College Park, MD 20742, USA

ABSTRACT

In this paper, European option prices are computed analytically, as well as simulated, for underlying asset price models with stochastic volatility and jump discontinuities. The analytical price is derived using the Fast Fourier Transform method developed in previous literature, which enables prices to be computed quickly. This model is compared with the Black-Scholes model, and the results suggest that this model addresses a known issue with the Black-Scholes model, the under and over valuations of short maturity options. The analytical solution is also used to investigate effective control variates in Monte Carlo simulations. Simulation experiments indicate that the random motion of the asset price serves as an effective control variate.

1 INTRODUCTION

According to the Option Clearing Corporation, over 329 million equity options contracts were traded in August 2017. In such a large market, these contracts can benefit from well-developed option pricing models. Furthermore, well-developed option pricing models can be used to determine the volatility of the underlying assets, providing insight about the risk of certain assets.

Option contracts provide the contract holder the right to buy or sell an underlying asset at a predetermined price and date. For example, if a European call option has a strike price of \$110 and a maturity of half a year, then in half a year the contract holder would have the right to buy the underlying asset at \$110. European means that this right can only be exercised at exactly the maturity date and not before. A call option gives the contract holder the right to buy the asset, whereas a put option gives the right to sell at a certain price. Most importantly, though, the contract holder is not forced to buy it at the strike price. If the asset is worth less than \$110 at the maturity date, then there is no requirement to buy it at \$110; however, if the asset is worth more, then the contract holder can buy it at the strike price of \$110. The payoff of this call option can be described mathematically as $(S_T - K)^+$ where $x^+ = max(x,0)$, K is the strike price and S_T is the asset price at the maturity date. This is because asset bought at the strike price can be sold immediately back at market price, yielding a profit of the difference or not exercised at all, yielding no profit or loss.

In the real world, these contracts are sold for a premium, raising an important question: How much are these options contracts worth? The fair price would be the expected payoff of the contract discounted for time. In Black and Scholes (1973), Fischer Black and Myron Scholes derived a mathematical formula for the price of these contracts based on certain assumptions. The effects were immediate. According to Bernstein (2012), the number of call options traded at the Chicago Board Options Exchange on opening day in 1973 was 911 and grew to more than 20,000 by mid-1974 and to 100,000 in 1977. Bernstein also

978-1-5386-6572-5/18/\$31.00 ©2018 IEEE

comments that every trader in the Chicago Board Options Exchange was using the Black-Scholes model. However, today there are known flaws with their model and assumptions.

The Black-Scholes model has a few key assumptions. The first is that the underlying asset follows Geometric Brownian Motion (GBM), which takes the form $dS_t = \mu S_t dt + \sigma S_t dW_t$, where dW_t is a Wiener Process. This leads to one of the major issues with the Black-Scholes framework. In GBM, the random motion is normally distributed, and thus extreme changes in prices are rare, resulting in severe undervaluations of short maturity options that are deep out-of-the-money. "Deep" refers to a large difference between the strike price and current asset price. For example, if the current asset price is \$50 and the strike price is \$100 on a call option, the option would be called deep out-of-the-money. An example of a deep in-the-money call option would be one where the asset price is \$100 and the strike price is \$50. In the Black-Scholes model, the lack of extreme changes in asset price makes it almost impossible for these deep out-of-the-money options to get in the money in that short period of time. Thus, the Black-Scholes model prices them far cheaper than they are sold for in the real market. For the same reasons the Black-Scholes model has been known to overvalue deep in-the-money options. One common solution is to add jump discontinuities in the asset price. These jumps can represent earnings reports or unexpected real-world events that cause shocks in the market. These jump discontinuities allow the asset to make large movements over short periods of time, allowing the results to more closely match the empirical data.

The second issue comes from the assumption that the volatility of underlying asset remains constant from the beginning of the contract until the end. There is empirical evidence that contradicts this assumption, which is reflected in the volatility smile illustrated in Figure 1. If the underlying asset is assumed to have

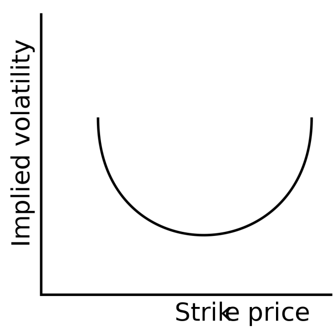

Figure 1: Example graph of the implied volatility for a single stock option across different strike prices, showing the well-known volatility smile.

constant volatility, then across all strike prices the implied volatility should be the same. Note, implied volatility is the calculated volatility of the asset that would yield the observed option price in the Black-Scholes model. However, the curve in Figure 1 has been observed in real data, suggesting that the volatility is not constant. As a result, stochastic volatility has been added to many recent models, where volatility is also a stochastic process.

Adding jumps and stochastic volatility to the original Black-Scholes model, however, creates challenges with tractability. Since Monte Carlo simulations are computationally expensive, analytical solutions are usually desirable. In addition, there has been literature suggesting the use of Fast Fourier Transform (FFT) to increase computational speeds even further. The idea of using FFT was proposed in Carr and Madan (1999). They used FFT to invert a characteristic function and recover a probability density function, which was then used to price the option by calculating the expected payoff using the recovered probability density function.

We propose using the FFT to invert the characteristic function of the Heston (1993) model of stochastic volatility and combining it with a separate jump distribution, leading to a model that contains jumps and stochastic volatility while remaining computationally inexpensive. The Heston model was chosen because it has a known characteristic function for the log asset price. Furthermore, it has been widely studied and applied to real data (Crisóstomo 2015). A similar model and method was proposed in Yan and Hanson (2006), but our approach is able to accommodate general jump distributions rather than being limited to log-uniform jump distributions.

When the characteristic function is not known, Monte Carlo techniques are often used, and variance reduction techniques such as control variates are important for computational efficiency. Effective control variates allow fewer replications while maintaining the same level of precision. Using the model developed in this paper as a benchmark, we investigate several candidate control variates and evaluate them on numerical examples.

In summary, our contribution is both a new analytical model that accounts for stochastic volatility and jumps and also an investigation of control variates for Monte Carlo simulation. The rest of the paper is organized as follows. Section 2 will provide the overview of the the model, discussing the characteristic function inversion and final jump height distribution construction. Section 3 contains the Monte Carlo simulation framework and information on the control variates used. The numerical results from the model and Monte Carlo simulation experiment are in Section 4, and Section 5 contains the conclusions and potential future research directions.

2 ANALYTICAL APPROACH

2.1 Stochastic Volatility Model

Stochastic volatility diffusion processes have been well studied. One of the most famous stochastic volatility models is the following proposed in Heston (1993):

$$\begin{aligned} dS_t &= \mu S_t dt + \sqrt{V_t} S_t dW_{1,t} \\ dV_t &= \kappa(\theta - V_t)dt + \varepsilon\sqrt{V_t} dW_{2,t} \\ dW_{1,t}W_{2,t} &= \rho dt. \end{aligned} \tag{1}$$

where S_t and μ are the asset price at time t and risk-free return rate, respectively, k, ε and θ are the mean reversion speed, volatility of volatility, and mean volatility, respectively, and the two Wiener processes, $W_{1,t}$ and $W_{2,t}$, have correlation ρ. Volatility in this model is a mean-reverting stochastic process; as V_t deviates from θ, the drift moves the process back towards θ. The speed at which the process reverts can be controlled using κ, while ε controls volatility of the volatility process.

2.2 Characteristic Function Inversion

Heston (1993) derived the characteristic function for $\ln(S_T)$, the log asset price at the maturity date. The characteristic function can be used to recover a probability density function (PDF), because the characteristic function is essentially the Fourier transform of the PDF,

$$f_X(x) = \frac{1}{\pi}\int_0^\infty e^{-itx}\phi_X(t)dt, \tag{2}$$

where $f_X(x)$ is the probability density function of the random variable X and $\phi_X(t)$ is the characteristic function of the random variable X. Using the Fast Fourier Transform (FFT), it is possible to efficiently invert the characteristic function and recover a PDF. FFT is an algorithm that quickly computes the Discrete Fourier Transform (DFT) of a given sequence $\{a_m\}$ given by

$$A_k = \sum_{m=0}^{N-1} a_m e^{-2\pi i \frac{mk}{N}}. \tag{3}$$

The FFT algorithm is useful because it reduces the number of computations from $O(N^2)$ to $O(N \log N)$, where N is a power of 2 and the number of elements in the series. The DFT maps the sequence $\{a_m\}$ to a new sequence $\{A_k\}$. In this case, $\{A_k\}$ will represent the PDF values. To approximate (2), the trapezoidal rule is used to write a series approximation:

$$\frac{1}{\pi}\int_0^\infty e^{-itx}\phi_X(t)dt \approx \frac{1}{\pi}\sum_{m=0}^{N-1}\delta_m e^{-it_m x}\phi_X(t_m)\Delta t \tag{4}$$
$$t_m = m\Delta t.$$

where δ_m is equal to $\frac{1}{2}$ for $m = 0$ and equal to 1 for all other values of m. The sum in (4) is very similar to (3), and with a few modifications, it becomes the same form. First, by setting the product $\Delta x \Delta t$ equal to $2\pi/N$, $e^{-imk2\pi/N}$ is equivalent to $e^{-it_m x_k}$, where t_m is the sequence of t values for which the characteristic function was calculated, and x_k is the sequence of x values for which the PDF values are approximated. Furthermore, there needs to be a shift factor e^{ibt_m}, which will allow the PDF to be calculated for a chosen initial x value, so that x_k ranges from $-b$ to $-b+(N-1)\Delta x$. Using all this, the final series a_m can be written in the following way:

$$a_m = \frac{1}{\pi}\delta_m e^{ibt_m}\phi_X(t_m)\Delta t$$
$$\Delta x \Delta t = \frac{2\pi}{N}$$
$$f_X(-b+k\Delta x) = \sum_{m=0}^{N-1} a_m e^{-2\pi i \frac{mk}{N}}.$$

This series will yield approximate PDF values for all x values of $-b+k\Delta x$ where $k = 0,1,2\ldots,N-1$.

2.3 Adding Jumps

To factor in jumps, a Compound Poisson jump process is introduced, meaning that the jumps occur according to a Poisson process with the height of the jump being a random value from a specified single jump height probability distribution. The goal in this section is to derive a jump height probability distribution that takes into account all possible numbers of jumps that could occur over the lifetime of the option. To find this probability distribution, first note that given n jumps, the jump height probability distribution is an n-fold convolution of the single jump height probability distribution. Then, to derive the final total probability distribution, sum each of the distributions for a specific number of jumps after scaling them by the probability of that many jumps occurring, i.e.,

$$F = \sum_{n=0}^{\infty}\frac{\lambda^n e^{-\lambda}}{n!}F_n. \tag{5}$$

where F is the probability distribution of total jump heights, n is the number of jumps, F_n is the probability distribution of jump heights from n jumps, and λ is the Poisson arrival rate of jumps.

2.4 Implementing Jumps

Practical implementation of the jump process requires truncation of the number of possible jumps, so a maximum number of jumps I_0 is chosen such that the tail probability after I_0 in the Poisson distribution is less than some threshold:

$$I_0 = \min_j \left[\sum_{i=j+1}^{\infty} \frac{\lambda^i e^{-\lambda}}{i!} < \varepsilon \right]. \tag{6}$$

The tail probability was then added back to the probability of I_0 jumps occurring.

If the single jump height probability distribution is continuous, then it must be discretized so it can be convolved with the probability distribution of $ln(S_T)$. The discretization process is given by

$$\mathbb{P}(J = j) = C\left(j + \frac{\Delta x}{2}\right) - C\left(j - \frac{\Delta x}{2}\right), \tag{7}$$

which is the probability that the jump height J is equal to j. $C(x)$ is the cumulative distribution function of the distribution being discretized and Δx is the spacing of the discretization. Note that the Δx here must equal the Δx in the characteristic function inversion.

If the distribution is infinite, then the tail probabilities are truncated and then added to the ends of the discretized distribution. Overall, this discretization method is very similar to (Fu et al. 2016). The only difference between these two schemes is the handling of the tail probabilities.

2.5 Putting It Together

We want our final model to account for stochastic volatility and jumps.

$$dS_t = \mu S_t dt + \sqrt{V_t} S_t dW_{1,t} + J S_t dN_t$$
$$dV_t = \kappa(\theta - V_t)dt + \varepsilon\sqrt{V_t} dW_{2,t}$$
$$dW_{1,t} W_{2,t} = \rho dt$$

In addition to the Heston model terms defined in Section 2.1 there are now jump process terms in our model. J is the random jump height and N_t is the Poisson jump counting process. It is clear from the dynamics that the model incorporates both stochastic volatility and jumps, specifically the Heston model plus a compound Poisson process. To find the analytical solution to this model we use the two probability distributions derived earlier: the total jump height probability distribution from Sections 2.3 and 2.4 and the recovered distribution for $\ln(S_T)$ from Sections 2.1 and 2.2. The final price probability distribution can now be derived by simply convolving the two distributions, yielding a price distribution that accounts for both stochastic volatility and jumps. The price of European options can then be priced using this final distribution through a simple sum. For a European call option the price is the discounted expectation of the payoff of the option:

$$\text{Price} = e^{-rT} \sum_{s \in S} \mathbb{P}(\ln(S_T) = s)(e^s - K)^+, \tag{8}$$

where K is the strike price, T is the time to maturity, r is the risk-free interest rate. $(e^s - K)^+$ represents the payoff of the option if the final log asset price is s, and $\mathbb{P}(\ln(S_T) = s)$ is the probability of the log asset price being s, which is taken from the final price probability distribution. The option price is then the sum in (8) calculated for all s in the set of possible log stock prices in the final price probability distribution, S. For the remaining part of this paper, this model will be referred to as the stochastic volatility with jumps (SV-J) model.

3 MONTE CARLO SIMULATION

3.1 Framework

When the option pricing model is complex, it is difficult to compute the analytical solution. Therefore, most researchers and practitioners turn to Monte Carlo simulation methods, due to its ease of accommodating these complicated formulations. However, these simulations can become computationally expensive, so it is crucial to develop methods that reduce computation time while improving precision and accuracy, e.g., variance reduction techniques such as control variates, which we investigate in our work here. In this paper, only the the stochastic volatility diffusion is simulated. The probabilities of the jump heights are known, because they were derived in Sections 2.3 and 2.4, and the jumps are added analytically via the distribution derived in Section 2. This was done because if jumps were also simulated, there would be too many sources of variation, requiring a large number of replications to achieve precise results.

In order to simulate the Heston model, the processes S_t and V_t must be discretized for simulation:

$$\begin{aligned} S_{t+1} &= S_t + \mu S_t \Delta t + S_t \sqrt{V_t^+ \Delta t} W_{1,t} \\ V_{t+1} &= V_t^+ + \kappa(\theta - V_t^+)\Delta t + \varepsilon \sqrt{V_t^+ \Delta t} W_{2,t}. \end{aligned} \tag{9}$$

Another important part of the simulation is the generation of two random variables $W_{1,t}$ and $W_{2,t}$ with ρ correlation, which are generated via

$$\begin{aligned} W_{1,t} &\sim \mathcal{N}(0,1), Z \sim \mathcal{N}(0,1) \\ W_{2,t} &= \rho W_{1,t} + \sqrt{1-\rho^2} Z. \end{aligned} \tag{10}$$

Once $W_{1,t}$ and $W_{2,t}$ are generated for the entire sample path, the asset price path can be generated up to the maturity date, T. The price at the maturity date, S_T, is added to the jump distribution and the expectation is calculated.

$$\begin{aligned} \text{Price} &= e^{-rT} \mathbb{E}[(S_T e^J - K)^+] \\ &= e^{-rT} \sum_{j \in J} \mathbb{P}(j)(S_T e^j - K)^+ \end{aligned} \tag{11}$$

In (11), J is the random variable representing the log jump height and j represents a possible log jump height. T is the maturity date and K is the strike price.

3.2 Control Variates

The control variates method uses a statistic generated from the sample with known expectation to help correct the statistic of interest; details of the method are provided in (Banks et al. 2009). In this paper, multiple control variates were tested. Choosing control variates can be tricky, as the requirement of known expectation is not always easy to satisfy. In this case, the average value of $W_{1,t}$, $W_{2,t}$, and V_t of the option were tested as control variates. The two variables, $W_{1,t}$ and $W_{2,t}$ have a known mean of zero and thus are suitable.

The average level of volatility was also tested, as it had a known expectation. However, it is important to note that the expectation of $\frac{1}{N}\sum V_t$ is not simply θ, the mean volatility level; this is only true if $V_0 = \theta$. However, it is possible to calculate the expectation using the sequence $\{X_i\}$ defined as follows:

$$\begin{aligned} \frac{1}{N}\mathbb{E}\left[\sum_{i=0}^{N-1} V_i\right] &= \frac{1}{N}\sum_{i=0}^{N-1} X_i \\ X_0 &= V_0 \\ X_{i+1} &= X_i + \kappa(\theta - X_i)\Delta t. \end{aligned} \tag{12}$$

The new process X_i captures the expectation of V_i, because the random motion of V_i has expectation zero, and satisfies (12), since

$$\frac{1}{N}\mathbb{E}\left[\sum_{i=0}^{N-1} V_i\right] = \frac{1}{N}\mathbb{E}[V_0 + V_0 + \kappa(\theta - V_0)\Delta t$$
$$+\varepsilon\sqrt{V_0}\Delta W_{2,1}\ldots V_{N-2} + \kappa(\theta - V_{N-1})\Delta t + \varepsilon\sqrt{V_{N-1}}\Delta W_{2,N-1}].$$

Note that $\Delta W_{2,t}$ has expectation zero, so all terms that contain $\Delta W_{2,t}$ can be ignored, which leaves

$$\frac{1}{N}\mathbb{E}[V_0 + V_0 + \kappa(\theta - V_0)\Delta t + \cdots + V_{N-1} + \kappa(\theta - V_{N-1})\Delta t].$$

This new expression is the same as the sum of X_i, as it only contains the mean reverting portion of V_i. In the simulation, the process X_i was generated and averaged to determine the average value of V_t. Now that we have the expectations of the average values of $W_{1,t}$, $W_{2,t}$, and V_t, each will be tested as a control variate.

4 SIMULATION AND ANALYTICAL RESULTS

The Monte Carlo simulations and analytical solution are presented in this section. The results are obtained using the exact model and Monte Carlo framework detailed in the previous sections. The code was written in Python and run on a laptop with 8GB LPDDR3 Onboard SDRAM and an Intel Core i5-7200U 2.5GHz Processor. The parameters for the simulation are $S_0 = 100$, $K = 100$, $T = 1$, $\kappa = 6.21$, $\theta = 0.019$, $V_0 = 0.010201$, $\varepsilon = .61$, $\mu = 0.0319$, $\rho = -.7$. As for the jump distribution, $J \sim ln\mathcal{N}(-.025, .05)$, $\lambda = 5$, and the simulation used 1,000 time steps. The results from the analytical model, basic Monte Carlo estimation and Monte Carlo estimation with $W_{1,t}$ as a control variate are given in Table 1.

Table 1: Comparison of Analytical Solution and Monte Carlo

Replications	95% CI	% Err	SE	Time (Seconds)
SV-J	7.421	0	0	0.359
MC (1,000)	(6.687, 7.548)	4.1	0.22	11
MC (10,000)	(7.229, 7.506)	0.7	0.07	110
MC (100,000)	(7.419, 7.507)	0.6	0.02	1202
CV (1,000)	(6.958, 7.479)	2.7	0.13	11
CV (10,000)	(7.362, 7.529)	0.3	0.04	110
CV (100,000)	(7.419, 7.472)	0.3	0.01	1202

From left to right, the columns are the number of replications and type of simulation estimation, 95% confidence interval (CI), the percent error from the SV-J value, standard error, and time required for the calculation. The first row in Table 1 provides the analytical solution provided by the stochastic volatility with jumps (SV-J) model discussed in Section 2. This is the price that was used to calculate percent errors in the Monte Carlo simulation results. Rows 2 through 4 are the results of the Monte Carlo without any control variates. As the replications increased, the standard error and the percent error decreased. However, it is important to note the computation cost. At 100,000 replications the simulation ran for 1202 seconds. This is a substantial computational cost that must be considered when deciding to implement the Monte Carlo method for pricing. Additionally, while adding control variates does have some computational cost, it is virtually negligible when compared to the computational cost of the Monte Carlo simulation itself, resulting in the identical rounded run times in Table 1 between the naive Monte Carlo estimation and control variate estimation.

The rows titled CV are the results from the control variate Monte Carlo. The specific control variate used in this case was $W_{1,t}$, the random motion of the price path. The control variate increased the precision of the simulation compared to the basic the Monte Carlo, as seen by the standard errors of Table 1 and in

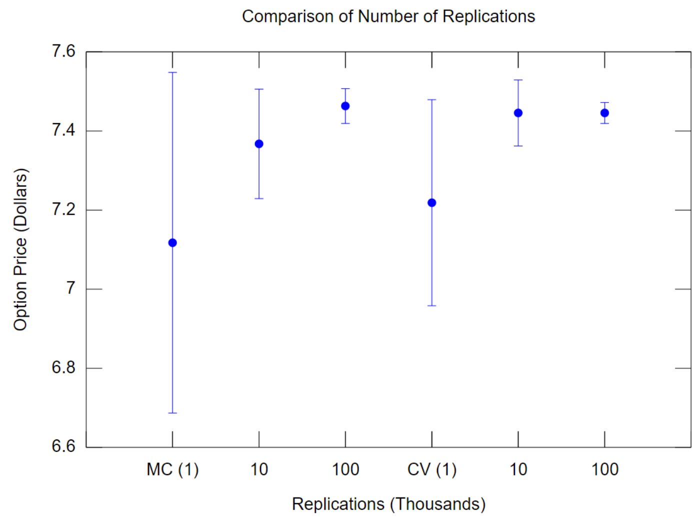

Figure 2: Confidence intervals from Table 1

a graph of the confidence intervals in Figure 2. However, not all the control variates tested increased the precision as significantly as $W_{1,t}$.

Table 2 and Figure 3 presents the results of various other control variates. Each of these Monte Carlo simulations was run with 10,000 replications. The computation times are not shown, as they are very similar to the computation times of 10,000 replications in Table 1. In Table 2, $W_{1,t}$ resulted in the lowest standard error. The other two control variates, $W_{2,t}$ and V_t, also reduced the standard error but not as much as $W_{1,t}$. The confidence intervals are even smaller when the maturity is shorter. Using the exact parameters

Table 2: Comparison of Various Control Variates

Control Variate	**95% CI**	**% Err**	**SE**
SV-J	7.421	0	0
MC	(7.191, 7.465)	1.3	.07
$W_{1,t}$	(7.310, 7.476)	0.4	.04
$W_{2,t}$	(7.213, 7.458)	1.1	.06
V_t	(7.292, 7.535)	0.1	.06

as Table 1 except a maturity of $T = 0.1$ years, confidence intervals were calculated for the basic Monte Carlo estimation and using $W_{1,t}$ as a control variate. The results are show in Table 3.

Table 3: Simulation results for Short Maturity, $T = 0.1$

Replications	**95% CI**	**SE**
SV-J	3.170	0
MC (1,000)	(2.989, 3.186)	0.050
MC (10,000)	(3.141, 3.202)	0.016
MC (100,000)	(3.151, 3.170)	0.005
CV (1,000)	(3.088, 3.175)	0.022
CV (10,000)	(3.167, 3.194)	0.007
CV (100,000)	(3.166, 3.174)	0.002

Table 3 illustrates that for shorter maturities the standard errors are much smaller because the price underlying asset has less time to change, leading to less variance in the final price distribution. Furthermore,

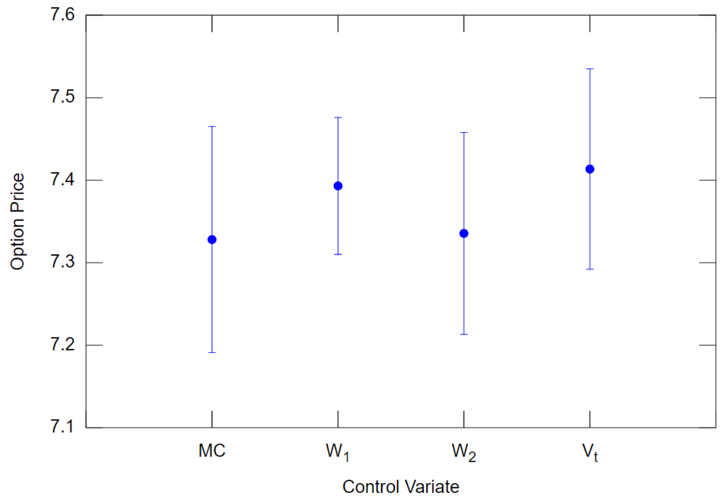

Figure 3: Confidence intervals from Table 2

the control variate $W_{1,t}$ continues to reduce the standard error by around 50% compared to the basic Monte Carlo simulation.

Next, the SV-J model is compared to the standard Black-Scholes model. To illustrate the differences between these two models, the option prices from both models were calculated over a range of strike prices. The prices were calculated for a short maturity option and a longer maturity option. The SV-J model should show characteristics that the Black-Scholes fails to capture when pricing short maturity options. However, for the longer maturity case, when the Black-Scholes model works relatively well, the proposed model should have similar prices to the Black-Scholes model.

The parameters for the SV-J model were the same as the ones used in Table 1 with two exceptions. Instead of a single strike price, Figure 4 calculates the call option price for a range of strike prices. Second, the initial volatility, V_0 is set to equal θ, the mean level of volatility in the Heston model. This is done so the mean level of volatility in the path is θ. As for the Black-Scholes prices, the prices are calculated using the well known Black-Scholes equation,

$$C(S_t,T) = S_t N(d_1) - e^{-\mu(T-t)} K N(d_2)$$
$$d_1 = \frac{1}{\sigma\sqrt{T-t}}\left(ln\left(\frac{S_t}{K}\right) + \left(\mu + \frac{\sigma^2}{2}\right)(T-t)\right)$$
$$d_2 = d_1 - \sigma\sqrt{T-t},$$

where $C(S_t,T)$ is the Black-Scholes call price for an asset with current value S_t at the maturity date T. The asset is assumed to follow geometric Brownian motion with parameters μ and σ, and $N(x)$ is the standard normal cumulative distribution function.

The parameters used in the Black-Scholes equation are shown below.

$$S_0 = 100, K = [85,120], \sigma = \sqrt{\theta}, \mu = 0.0319$$

Most parameters are the same as Table 1 as long as they are applicable to Black-Scholes. The σ was chosen as the $\sqrt{\theta}$ because θ is the expected mean level of volatility, and the V_t process is square rooted in the S_t process. These parameters are used to calculated the option prices for both a short, $T = .1$ year, and long, $T = 1$ year, maturity option. The results are presented in Figures 4 and 5, respectively. Also note that computation time for these curves are negligible, since they are analytical solutions and do not require simulation.

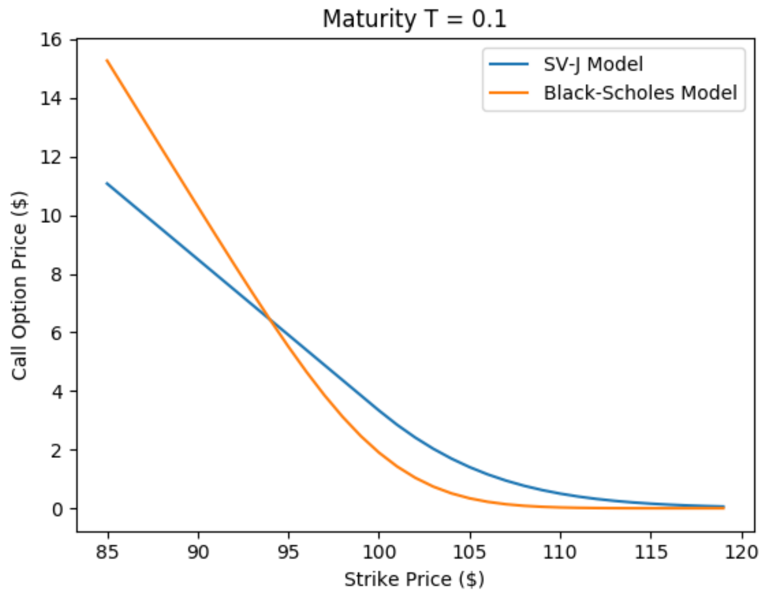

Figure 4: Comparison of call option price from SV-J and Black-Scholes model in short maturity cases ($T = 0.1$).

Figure 4 illustrates the short maturity option prices from both models. For the low strike prices in Figure 4 the Black-Scholes prices are significantly higher than the SV-J model. The Black-Scholes model has been known to overvalue these types of options, especially on short maturities and, in Figure 4, the SV-J model prices these options cheaper than the Black-Scholes. Figure 4 provides some evidence that the SV-J model, with the right calibration of parameters and jumps, could resolve this issue. Next, we can look at the larger strike prices, the deep out-of-the-money options, the Black-Scholes model has historically undervalued these options. So, the SV-J model should have prices higher than the Black-Scholes model, and this is exactly what we see in Figure 4.

For longer maturities, the Black-Scholes model performs well, so we would hope that the SV-J model would produce similar results. Figure 5 graphs the prices from the Black-Scholes model and SV-J model across the same range of strikes as Figure 4. From the graph, it is apparent that the two models yield very similar results for longer maturity options. This is reassuring, and combined with the results from Figure 4, provides strong evidence that the SV-J model can solve many of the shortcomings of Black-Scholes while still performing well for the long maturity options.

5 CONCLUSION AND FUTURE RESEARCH

In this paper, an asset price model with stochastic volatility and jump discontinuities is proposed. The Fast Fourier transform is used to recover a probability density function from the characteristic function of the Heston Model, which is convolved with the jump size distribution, yielding a final price distribution from which the expected payoff of European options can be calculated. The model is shown to resolve the issues Black-Scholes traditionally has with short maturity options. The analytical solution is then also compared to traditional Monte Carlo simulations with control variates implemented.

Three control variates were tested in the simulation experiment, and the results suggest that $W_{1,t}$, the random motion of the stock price, is the best control variate for variance reduction. In the future, it would be interesting to test other control variates or other variance reduction techniques. Another interesting possible area for future research would be to use more than one control variate; identifying the most

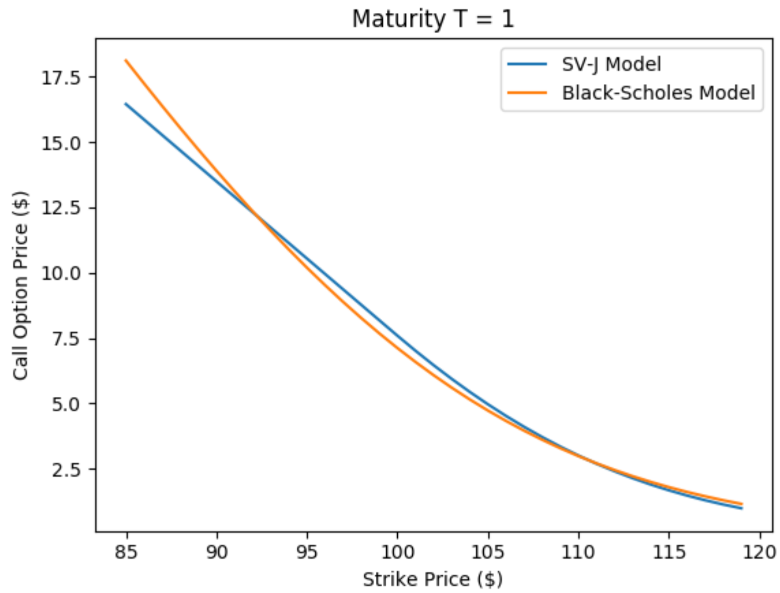

Figure 5: Comparison of call option price from SV-J model and Black-Scholes model in long maturity cases ($T = 1$).

effective combination of control variates would be useful for applying simulations to real data. Finally, the most natural extension of this paper would be to apply this model to real data. To do this, the model parameters would have to be calibrated according to the data.

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation under Grants CMMI-1362303 and CMMI-1434419.

REFERENCES

Banks, J., J. Carson, B. Nelson, and D. Nicol. 2009. *Discrete-Event System Simulation*. Prentice-Hall New Jersey.

Bernstein, P. 2012. *Capital Ideas: The Improbable Origins of Modern Wall Street*. Wiley.

Black, F., and M. Scholes. 1973. "The Pricing of Options and Corporate Liabilities". *Journal of Political Economy* 81(3):637–654.

Carr, P., and D. Madan. 1999. "Option Valuation Using the Fast Fourier Transform". *Journal of Computational Finance* 2(4):61–73.

Crisóstomo, R. 2015. "An Analysis of the Heston Stochastic Volatility Model: Implementation and Calibration Using Matlab". *arXiv preprint arXiv:1502.02963*.

Fu, M. C., B. Li, G. Li, and R. Wu. 2016. "Option Pricing for a Jump-Diffusion Model with General Discrete Jump-Size Distributions". *Management Science* 63(11):3961–3977.

Heston, S. L. 1993. "A Closed-Form Solution for Options with Stochastic Volatility with Applications to Bond and Currency Options". *The Review of Financial Studies* 6(2):327–343.

The Options Clearing Corporation 2017. "Monthly Statistics Report For Equity Options". Accessed August 23rd 2017. https://www.theocc.com/webapps/historical-volume-query.

Wu, L. 2007. "From Characteristic Functions and Fourier Transforms to PDFs/CDFs and Option Prices". *Zicklin School of Business, Baruch College*. http://faculty.baruch.cuny.edu/lwu/890/adp_transform.pdf.

Yan, G., and F. B. Hanson. 2006. "Option Pricing for a Stochastic-Volatility Jump-Diffusion Model with Log-Uniform Jump-Amplitudes". In *Proceedings of the American Control Conference.* Jun 14th-16th, Minneapolis, Minnesota, 6-12.

AUTHOR BIOGRAPHIES

URO LYI graduated from Montgomery Blair High School and was part of the Math, Science, and Computer Science Magnet Program. This reasearch was part of the Magnet Program's Senior Research Project (SRP). He is now an undergraduate student at the University of Maryland, College Park and will be majoring in mathematics. His research interests include financial engineering, mathematical finance, and simulation optimization. His email is urolyi1@gmail.com.

MICHAEL C. FU holds the Smith Chair of Management Science in the Robert H. Smith School of Business, with a joint appointment in the Institute for Systems Research, A. James Clark School of Engineering, University of Maryland, College Park, where he has been since 1989. His research interests include simulation optimization and applied probability, with applications to manufacturing and finance. He attended his first WSC in 1988 and served as Program Chair for the 2011 WSC. He has served as NSF Operations Research Program Director, *Management Science* Stochastic Models and Simulation Department Editor, and *Operations Research* Simulation Area Editor. He is a Fellow of **INFORMS** and **IEEE**. His e-mail address is mfu@umd.edu, and his Web page is https://www.rhsmith.umd.edu/directory/michael-fu.