Management Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Option Pricing for a Jump-Diffusion Model with General Discrete Jump-Size Distributions

Michael C. Fu, Bingqing Li, Guozhen Li, Rongwen Wu

MANAGEMENT SCIENCE
Articles in Advance, pp. 1–18
ISSN 0025-1909 (print) | ISSN 1526-5501 (online)

http://dx.doi.org/10.1287/mnsc.2016.2522

Option Pricing for a Jump-Diffusion Model with General Discrete Jump-Size Distributions

Michael C. Fu
Robert H. Smith School of Business and Institute for Systems Research, University of Maryland, College Park, Maryland 20742, mfu@rhsmith.umd.edu

Bingqing Li, Guozhen Li
School of Finance, Nankai University, 300350 Tianjin, China {libq@nankai.edu.cn, jennylea152@gmail.com}

Rongwen Wu
Capital One Financial Corporation, McLean, Virginia 22102, rongwen_wu@hotmail.com

We obtain a closed-form solution for pricing European options under a general jump-diffusion model that can incorporate arbitrary discrete jump-size distributions, including nonparametric distributions such as an empirical distribution. The flexibility in the jump-size distribution allows the model to better capture leptokurtic features found in real-world data. The model uses a discrete-time framework and leads to a pricing formula that is provably convergent to the continuous-time price as the discretization is increased. The solution is easy to implement with fast convergence properties. Numerical results illustrate the efficiency and accuracy of the proposed model and highlight its robustness and flexibility.

Keywords: jump-diffusion process; option pricing; European option; generating function; lattice path
History: Received November 11, 2015; accepted April 4, 2016, by Noah Gans, stochastic models and simulation. Published online in *Articles in Advance* September 29, 2016.

1. Introduction

Jump-diffusion processes are commonly used to model the price dynamics for the underlying asset in option pricing, because they are able to capture leptokurtic features and volatility smiles observed in real-world data. The jump-diffusion process was first proposed for option pricing by Merton (1976) to address important characteristics observed in actual market data that cannot be captured by the Black–Scholes model (Black and Scholes 1973), such as negative skewness and excess kurtosis. Since the seminal paper of Merton (1976), jump-diffusion processes have become one of the main models in option pricing for capturing jumps in an underlying asset. Surveys of critical developments in the jump-diffusion literature, as well as research challenges, can be found in Eraker et al. (2003), Bates (2003), Broadie and Detemple (2004), and Kou (2008a); see also Schoutens (2003) for more general Lévy processes. Jump-diffusion models complement stochastic volatility models, which are better able to capture volatility clustering effects; combining the two features (jumps and stochastic volatility) leads to more generals models (see Garcia et al. 2010) but at the cost of tractability, as concluded in Kou (2008a, p. 88):

> jump-diffusion models attempt to strike a balance between reality and tractability, especially for short maturity options and short term behavior of asset pricing.

Further motivation for our work is summarized in Cai and Kou (2011, p. 2067):

> a key question for jump diffusion models is what jump size distribution will be used.

For general jump-size distributions, Merton (1976) provided a European option pricing formula written in the form of an infinite sum of Black–Scholes-type terms, which is independent of the specific jump-size distribution; however, each term involves complex computationally intensive multidimensional integration, which generally depends heavily on the actual jump-size distribution, requiring a case-by-case analysis for each distribution. Merton (1976) considered two special jump-size distributions—a discrete distribution described by Samuelson (1973), where there is a positive probability of immediate ruin, and the log-normal distribution—and provided corresponding pricing formulas. The log-normal assumption makes estimation and hypothesis testing tractable, and it has become the most important representation of the jump-diffusion models (e.g., see Ball and Torous 1983, Jarrow and Rosenfeld 1984, Bates 1991).

However, a large body of empirical studies shows that the log-normal jump-size distribution fits actual returns data rather poorly, primarily because empirical log-return distributions exhibit excess kurtosis and skewness relative to the normal distribution (e.g., see

Bookstaber and McDonald 1987, Madan and Seneta 1990, Tucker 1992, and references therein). As a result, there is a vast literature on analyzing and developing more accurate models for option pricing. We briefly survey some representative streams.

Ramezani and Zeng (1998) conducted a maximum likelihood estimation on security prices and showed that allowing for a mixture of distributions for the upward jumps and downward jumps proves to be a better fit to the data than having a common distribution spawning both. They proposed an asymmetric jump-size distribution with the upward-jump and downward-jump magnitude following Pareto and Beta distributions, respectively.

Kou (2002) proposed a separate model with jump sizes following a (asymmetric) log-double-exponential distribution, for which he was able to derive closed-form pricing formulas for European options, later extended to path-dependent options such as American options, barrier options, and look-back options (Kou and Wang 2004, Kou 2008b). More recently, the log-double-exponential distribution of jump size has been generalized to log-phase-type-exponential (Asmussen et al. 2004), log-hyperexponential (Cai 2009) and log-mixed-exponential (Cai and Kou 2011) distributions.

As pointed out by Bollerslev and Todorov (2011), the jump-size distributions are distinct for different underlying assets and different sampling frequencies. Kaeck (2013) demonstrated that improved option pricing accuracy comes mainly from a better fit to the jumps, which highlights the importance of the choice of jump-size distribution for option pricing. After investigating seven kinds of alternative jump-size distributions, he concluded that the log-double-Gamma distribution outperforms the other jump-size distributions.

The literature review makes clear that the vast majority of jump-diffusion models have focused on continuous jump-size distributions. The main advantage of continuous distributions is their ability to be described by a small set of parameters. However, one could argue that the effect of jumps is more critical when they are large in magnitude, and these often occur with a relative infrequency that is not well captured by continuous distributions and might be best represented, for example, by an empirical distribution. Discrete jump-size distributions provide flexibility and robustness that could serve as alternative to continuous distributions in some settings.

Thus, our proposed model incorporates discrete jump-size distributions in a discrete-time framework that naturally gives rise to a multinomial lattice framework to model the jump-diffusion process. The resulting multinomial lattice is a generalization of the multinomial lattice developed in Amin (1993), who priced options using backwards induction. Under our framework, the option price, which is an expectation under the martingale measure, requires the calculation of a summation. For most options, the summation is an implicit enumeration problem of the probability-weighted lattice paths for the underlying asset price. The use of generating functions from enumerative combinatorics provides an efficient method to carry out such calculations, and Li and Zhao (2009) used the generating function method to price Parisian options under the binomial lattice framework. Here, we apply the method to the multinomial lattice framework, which is significantly more computationally burdensome than a recombining binomial tree. By calculating the cardinal number and probability weight of all the possible price paths, we derive the price distribution of the underlying asset on the expiration date. Then we derive a unified pricing formula that leads to a closed-form solution for European options on the proposed discrete jump-diffusion model. We prove that the resulting formula converges to Merton's continuous-time pricing formula as the number of time steps N goes to infinity, where the computation complexity is of order $O(N^{1.5})$.

Our proposed solution is very flexible and robust, since the same valuation formula applies to European options with any jump-size distribution. For example, discrete jump-size distributions (see Rachev et al. 2010), such as binomial and negative binomial distributions (Benninga and Wiener 1998) and hypergeometric distributions (Humpage 1997, Santos and Guerra 2015), have been widely used in financial modeling but not so much in option pricing, as a result of the challenge of deriving a separate analytical option pricing formula for each different jump-size distribution, but they could be easily incorporated in our framework. Numerical experiments on valuing options under different jump-size distributions using our pricing formula indicate that the shape of the jump-size distribution could have a significant impact on the price of an option. With the flexibility of being able to incorporate any jump-size distribution, our model allows one to easily compare the pricing performance resulting from different distributions, which can help improve price accuracy by virtue of better distribution choices, e.g., to model rare events in a financial crisis.

In sum, our work contributes to the option pricing research literature as follows:

- We provide a *closed-form solution* for European options on jump-diffusion processes with *general discrete jump-size distributions*, which includes nonparametric distributions such as empirical distributions that can be used to model actual data directly and better capture leptokurtic features.

- We prove that the analytical solution under the discrete-time framework converges to Merton's continuous-time formula as the number of time steps N goes to infinity, with a computation complexity of order $O(N^{1.5})$.
- We develop a general combinatorial approach for derivatives pricing on a multinomial lattice that is significantly more computationally efficient than previous dynamic programming-based backward recursion methods.

The rest of the paper is organized as follows. In Section 2, the multinomial lattice model and lattice path option pricing method are presented. In Section 3, we derive the European option pricing formula for discrete-time jump-diffusion processes and prove convergence of the formula to the price under the continuous-time process. Section 4 presents numerical experiments both to demonstrate the computational properties of the pricing formula and to empirically investigate the impact of the shape of jump-size distributions on the option prices. Section 5 concludes this paper and provides some directions for future research.

2. Model Setting

The theoretical model for option pricing in discrete time is constructed in this section.

2.1. Lattice Construction

In a discrete-time framework, Cox et al. (1979) provided an option pricing formula using the binomial lattice model. Here, we generalize the binomial lattice model to the multinomial lattice model, where the constraint of asset price dynamics being discrete and moving among lattice points only is retained. In the multinomial lattice model, the asset price lattice has more than two—possibly infinite—lattice points at each time step.

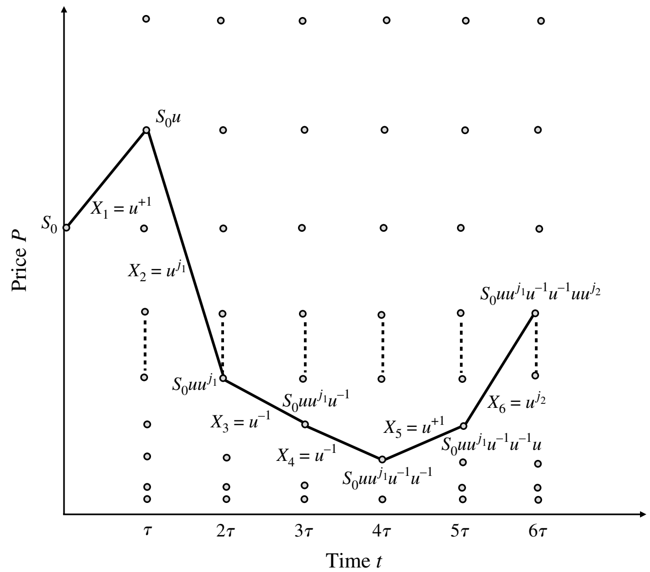
We consider the price dynamics of an asset $\{S_t\}$ over the interval $t \in [0, T]$, which is divided into N equal intervals of length $\tau \doteq T/N$, where S_0 denotes the initial price state of the asset. Then we have the asset price path of an asset $\{S_0, S_\tau, \ldots, S_{k\tau}, \ldots, S_{N\tau}\}$ over the N time steps. Defining the (proportional) price change at time step k by X_k, then the price relationship between time steps $k-1$ and k is given by

$$X_k = \frac{S_{k\tau}}{S_{(k-1)\tau}}, \tag{1}$$

where X_k is a random variable that takes value in the state space $\{u^i, i \in \mathbb{Z}\}$, $u > 1$.

Therefore, for a fixed initial price S_0, each sample asset price path can be defined by a sequence of (one-step) moves on the lattice that we will call a lattice path. The concept of lattice path was first introduced by Li and Zhao (2009) in the binomial lattice framework.

Figure 1 **A Multinomial Lattice and an Example Lattice Path $\alpha_0 = (u^{+1}, u^{j_1}, u^{-1}, u^{-1}, u^{+1}, u^{j_2})$, with Initial State S_0, Price Change $X_k = u^{i_k}$, $i_k \in \mathbb{Z}$, and Price State $S_k = S_0 \prod_{i=1}^{k} X_i$**



Definition 1. A *lattice path* α of *length* $n = l(\alpha)$ is a sequence of moves:

$$\alpha = (x_1, x_2, \ldots, x_n), \quad x_k \in \{u^i, i \in \mathbb{Z}\}, \; k = 1, \ldots, n,$$

where the difference between the lattice points of the terminal and starting points of the path is called the *height* of the lattice path, denoted by

$$|\alpha| = \log_u \prod_{k=1}^{n} x_k = \sum_{k=1}^{n} \log_u x_k.$$

An example of lattice path $\alpha_0 = (u^{+1}, u^{j_1}, u^{-1}, u^{-1}, u^{+1}, u^{j_2})$ with initial price S_0 is shown in Figure 1. The length of this lattice path is $l(\alpha_0) = 6$, and the height is $|\alpha_0| = 1 + j_1 + (-1) + (-1) + 1 + j_2 = j_1 + j_2$.

2.2. Probability Measure

We express the probability of a lattice path in terms of one-step move probabilities, which are assumed to be mutually independent and given by

$$\mathbb{P}(X_k = u^i) = p_i, \quad i \in \mathbb{Z}. \tag{2}$$

As a result, we can define the lattice path probability as below.

Definition 2. For lattice path $\alpha = (x_1, x_2, \ldots, x_n)$, the *lattice path probability* is defined by

$$\mathbb{P}(\alpha) \doteq \prod_{k=1}^{n} \mathbb{P}(X_k = x_k), \tag{3}$$

where $\mathbb{P}(X_k)$ is defined as above.

Using the lattice path α_0 in Figure 1 as an example, we have

$$\mathbb{P}(\alpha_0) = p_{+1} p_{j_1} p_{-1} p_{-1} p_{+1} p_{j_2} = p_{+1}^2 p_{-1}^2 p_{j_1} p_{j_2}.$$

2.3. Pricing Technique

Here, we provide an overview on how we will use the lattice path technique defined for any given probability measure to price an option. From arbitrage-free option pricing theory, under a specific martingale probability measure $\tilde{\mathscr{P}}$, the value of a European option is equal to the expectation of its discounted future payoffs (see Harrison and Kreps 1979, Harrison and Pliska 1981); i.e.,

$$V = \mathrm{e}^{-rT}\,\mathbb{E}_{\tilde{\mathscr{P}}}[F_T],$$

where F_T denotes the European option payoff at the expiration date T. Common examples are $F_T = (S_T - K)^+$ for a European call option and $F_T = (K - S_T)^+$ for a European put option, where K is the strike price. In Section 3.1, we will specify $\tilde{\mathscr{P}}$ for the jump-diffusion model.

In the lattice framework, the underlying asset price S_T at expiration date T can be written as

$$S_T = S_{N\tau} = S_0 u^{|\alpha|},$$

where $\alpha = (x_1, x_2, \ldots, x_N)$ is the lattice path (see Definition 1). Hence the option value can be calculated by the weighted-sum of the payoff along all feasible lattice paths. For payoffs that only depend on the lattice height, we can count all possible paths ending with height h and then compute the probability weight of each h.

Taking a European call option as an example,

$$\begin{aligned} V^{\text{call}} &= \mathrm{e}^{-rT}\,\mathbb{E}_{\tilde{\mathscr{P}}}[(S_T - K)^+] = \mathrm{e}^{-rT}\,\mathbb{E}_{\tilde{\mathscr{P}}}[(S_{N\tau} - K)^+] \\ &= \mathrm{e}^{-rT} \sum_{\text{lattice paths } \alpha=(x_1,x_2,\ldots,x_N)} \mathbb{P}(\alpha)(S_0 u^{|\alpha|} - K)^+ \\ &= \mathrm{e}^{-rT} \sum_h \sum_{\alpha:\, l(\alpha)=N,\, |\alpha|=h} \mathbb{P}(\alpha)(S_0 u^h - K)^+ \\ &= \mathrm{e}^{-rT} \sum_{h:\, S_0 u^h \ge K} (S_0 u^h - K)\,\mathbb{P}_N(h), \end{aligned}$$

where

$$\mathbb{P}_N(h) \doteq \sum_{\alpha:\, l(\alpha)=N,\, |\alpha|=h} \mathbb{P}(\alpha) = \sum_{i_1+\cdots+i_N=h} p_{i_1}\cdots p_{i_N}. \tag{4}$$

Therefore, we have the following pricing formula for an arbitrary discrete stochastic process on the general lattice model.

Proposition 1. *For a European option with strike price K, expiration date T, and initial underlying asset value S_0, the values of the call and put options under a multinomial lattice with grid size u can be written as*

$$\begin{aligned} V^{\text{call}} &= \mathrm{e}^{-rT} \sum_{h \ge h^{\text{call}}} \mathbb{P}_N(h)(S_0 u^h - K) \quad \textit{and} \\ V^{\text{put}} &= \mathrm{e}^{-rT} \sum_{h \le h^{\text{put}}} \mathbb{P}_N(h)(K - S_0 u^h), \end{aligned} \tag{5}$$

respectively, where $\mathbb{P}_N(h)$ is given by Equation (4), r is the riskless rate, and

$$h^{\text{call}} = \lceil \log_u(K/S_0) \rceil, \quad h^{\text{put}} = \lfloor \log_u(K/S_0) \rfloor. \tag{6}$$

Note that $\lceil\cdot\rceil$ and $\lfloor\cdot\rfloor$ denote the (greatest/least integer) ceiling and floor functions, respectively, and h^{call} and h^{put} are the least and greatest integers guaranteeing $(S_0 u^h - K)^+ = S_0 u^h - K$ and $(K - S_0 u^h)^+ = K - S_0 u^h$, respectively.

Thus, we just need to calculate the payoff and probability weight for each possible lattice path to obtain the option value or, more precisely, the payoff and probability weight $\mathbb{P}_N(h)$ of each height h on the N-length lattice. We have converted a European option pricing problem into a lattice path enumeration problem in combinatorics mathematics.

It is relatively straightforward to calculate the number of lattice paths leading to the same lattice height if the underlying asset price follows a binomial lattice model, but for a multinomial lattice model, the task becomes much more challenging, especially in the setting where one-step moves can take the price to any point in the lattice, which itself could in principle be unbounded.

3. Pricing European Options for the Jump-Diffusion Model

In this section, we derive the closed-form pricing formula for a European option where the underlying asset price dynamics follows a discrete-time jump-diffusion process using the lattice model introduced in Section 2. Then we prove that the formula converges to the corresponding continuous-time formula when the number of time steps N goes to infinity (for a fixed T), and we demonstrate that the formula has an efficient computational complexity of $O(N^{1.5})$. We also generalize the formula to the setting where the asset price volatility is time dependent but deterministic.

3.1. Pricing the Discrete Jump-Diffusion Model

Section 2 introduced the general lattice model for an arbitrary discrete stochastic process. Here, we specify the general lattice model for jump-diffusion processes.

A jump-diffusion process is made up of a diffusion process and a jump process. Following the lattice model with one-step moves X_k defined by (1), we define the discrete-time jump-diffusion model by a one-step move that is a random walk $W = u^{\pm 1}$ (up or down) or a jump Y of size $\{u^j, j \in \mathbb{Z}\backslash\{\pm 1\}\}$, as illustrated in Figure 2.

As in Merton (1976) and Amin (1993), we assume that the jump risks are diversified over time, so that the jump component can be described as a Poisson

Figure 2 **Possible Moves in a Time Step: From State Price S in Time $k\tau$, the Next Move Could Be a Random Walk Move W Up (u) or Down ($d = u^{-1}$), or a Jump of Size Y**

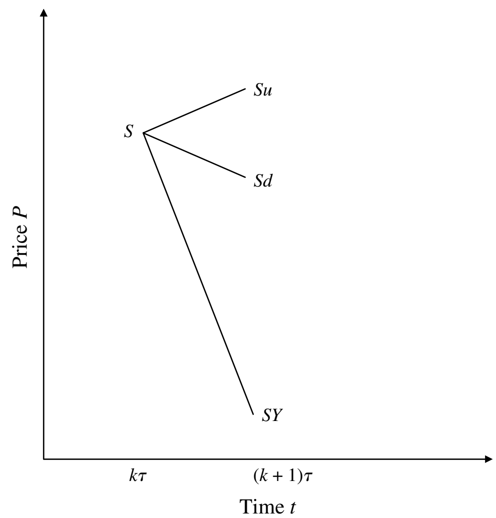
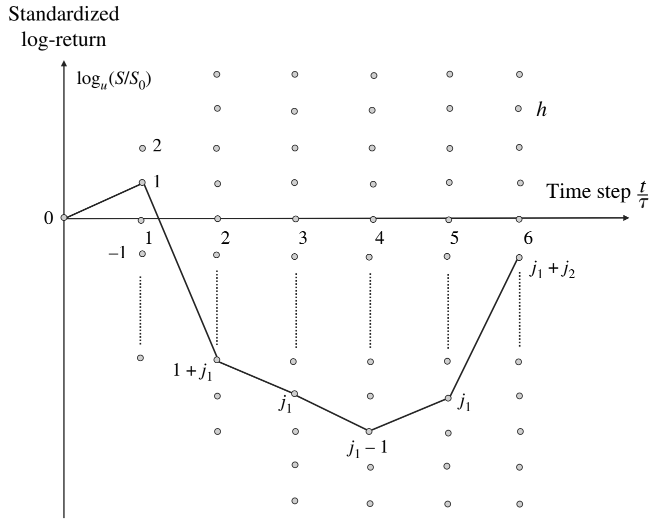


Figure 3 **Multinomial Lattice for Standardized Log-Returns with Example Lattice Path $\alpha_0 = (+1, j_1, -1, -1, +1, j_2)$**



process with arrival rate λ. If the jump risks are systematic, we can obtain a corresponding martingale process $\tilde{\mathcal{P}}$ by means of the discrete Radon–Nikodym derivative, from which the option can be priced, as in Amin (1993). If the volatility of the diffusion process under the physical probability measure is σ, and the jump-size distribution is $\mathcal{Y}$, then the one-step move probabilities under the martingale probability measure $\tilde{\mathcal{P}}$ are given by

$$\mathbb{P}(X_k) = \begin{cases} 1-q & X_k = W, \\ q & X_k = Y, \end{cases}$$
$$= \begin{cases} (1-q)p & X_k = W = u, \\ (1-q)(1-p) & X_k = W = d, \\ q\rho_j & X_k = Y = u^j, \ j \in \mathbb{Z}\backslash\{\pm 1\}, \end{cases}$$

where

$$q = \lambda\tau, \quad p = \frac{(e^{r\tau} - \lambda\tau\mathbb{E}_{\mathcal{Y}}[Y])/(1-\lambda\tau) - d}{u-d}, \quad u = e^{\sigma\sqrt{\tau}}, \quad d = e^{-\sigma\sqrt{\tau}}, \tag{7}$$

and the jump-size distribution is written as

$$Y \sim \mathcal{Y}\colon \ \mathbb{P}(Y = u^j) = \rho_j, \quad j \in \mathbb{Z}\backslash\{\pm 1\}, \ \sum_j \rho_j = 1. \tag{8}$$

Also, to ensure that $p > 0$, the length of time subinterval τ must be taken sufficiently small. Under the assumption of diversifiability, the jump-size distribution under the martingale probability measure is the same as under physical probability measure (Amin 1993).

For ease of enumeration, we standardize the multinomial lattice by considering the logarithm of the asset price $\log_u S_t$. The previously defined concepts on the lattice model can be naturally translated to the standard lattice[1]: lattice path $\alpha = (x_1, x_2, \ldots, x_n)$, $x_k \in \{i, i \in \mathbb{Z}\}$, lattice height $|\alpha| = \sum_{k=1}^n x_k$, and the log jump-size distribution

$$J \sim \mathcal{J}\colon \ \mathbb{P}(J = j) = \mathbb{P}(Y = u^j) = \rho_j, \quad j \in \mathbb{Z}\backslash\{\pm 1\}. \tag{9}$$

Figure 3 shows the lattice path α_0 from Figure 1 on the standard lattice. Note that this standardization procedure does not change the one-step move probabilities nor the resulting lattice path probabilities.

Now we calculate the probability weight $\mathbb{P}_N(h)$ of lattice height for the discrete jump-diffusion model to obtain a closed-form pricing formula for European options.

Theorem 1. *For the discrete jump-diffusion process defined on a multinomial lattice with one-step move probabilities,*

$$\mathbb{P}(X_k) = \begin{cases} 1-q & X_k = W, \\ q & X_k = J, \end{cases}$$
$$= \begin{cases} (1-q)p & X_k = W = 1, \\ (1-q)(1-p) & X_k = W = -1, \\ q\rho_j & X_k = J = j, \ j \in \mathbb{Z}\backslash\{\pm 1\}; \end{cases} \tag{10}$$

with p and q defined by (7), the probability weight is given by

$$\mathbb{P}_N(h) = \sum_{I=0}^{N}\sum_{U=0}^{N-I} \binom{N}{I, U} q^I (1-q)^{N-I} p^U \cdot (1-p)^{N-I-U} \rho^{(I)}_{N+h-I-2U}, \tag{11}$$

[1] For simplification, we retain almost all the same notation after standardization, such as X_k, W, α, with the meaning understood by context.

where $\binom{N}{I,U} = N!/(I!\,U!\,(N-I-U)!)$ *is the multinomial coefficient, and* $\rho_L^{(I)} = \sum_{j_1+\cdots+j_I=L} \rho_{j_1}\cdots\rho_{j_I}$ *is the conditional probability of a cumulative jump size of L lattice points over I jumps.*

Proof. For an h-height lattice path $\alpha = (x_1, x_2, \ldots, x_N)$, letting U, D, and I denote the number of random walk up, random walk down, and jump moves in the first N time steps, respectively, and letting j_i denote the jump size as a result of the ith jump, we have

$$h = |\alpha| = \sum_{k=1}^{N} x_k = U(+1) + D(-1) + \sum_{i=1}^{I} j_i$$
$$= U - D + L = 2U + I + L - N,$$

where $L = \sum_{i=1}^{I} j_i$.

From Lemma 3 of Appendix B.1, we know that under the discrete jump-diffusion model, the generating function of $\mathbb{P}_N(h)$ is

$$G_N(z;\tilde{\mathcal{P}}) = \sum_{I=0}^{N}\sum_{U=0}^{N-I}\sum_{L=-\infty}^{+\infty} \binom{N}{I,U} q^I(1-q)^{N-I} \cdot p^U(1-p)^{N-I-U}\rho_L^{(I)} \cdot z^{2U+I+L-N}.$$

Since $\mathbb{P}_N(h)$ is the coefficient of the term z^h, (11) follows. □

Combining Proposition 1 and Theorem 1, we obtain the closed-form pricing formula for European options.

Corollary 1. *For a European option with strike price K, expiration date T, and initial underlying asset value* S_0*, where the underlying asset price process* $\{S_t\}$ *follows a discrete-time jump-diffusion process with one-step move probabilities given by* (10)*, the values of the call and put options under a multinomial lattice with grid size u are given by* (5), (6), *and* (11).

3.2. Continuous-Time Limit

Now we prove that as the number of time steps N goes to infinity, the discrete-time call option pricing formula given by Corollary 1 converges to the general continuous-time jump-diffusion call option formula[2] (Merton 1976) given by

$$V_c = \sum_{I=0}^{\infty} e^{-\lambda T}\frac{(\lambda T)^I}{I!}\mathbb{E}_{\mathcal{Y}_c}\left[\mathrm{BS}\left(e^{-\lambda(\mu-1)T}S_0\prod_{i=1}^{I} y_i, T; K, \sigma^2, r\right)\right],$$

where r is the riskless rate, $\mathcal{Y}_c$ is the continuous jump-size distribution (subscript c is added to differentiate from the discrete jump-size distribution $\mathcal{Y}$ defined earlier), $\mu = \mathbb{E}_{\mathcal{Y}_c}[Y]$ is the conditional expectation of the return from one jump, and BS denotes the Black–Scholes formula given by

$$\mathrm{BS}(S_0, T; K, \sigma^2, r) = S_0\mathcal{N}(d_1) - e^{-rT}K\mathcal{N}(d_2),$$
$$\text{where } \mathcal{N}(x) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{x} e^{-s^2/2}\,ds,$$
$$d_1 = \frac{\ln(S_0/K) + (r+\sigma^2/2)T}{\sigma\sqrt{T}},$$
$$d_2 = \frac{\ln(S_0/K) + (r-\sigma^2/2)T}{\sigma\sqrt{T}}.$$

For notational convenience, we denote

$$q(I) \doteq \frac{e^{-\lambda T}(\lambda T)^I}{I} \quad \text{and}$$
$$V_c(I) \doteq \mathbb{E}_{\mathcal{Y}_c}\left[\mathrm{BS}\left(\mathbf{e}^{-\lambda(\mu-1)T}S_0\prod_{i=1}^{I} y_i, T; K, \sigma^2, r\right)\right];$$

then the continuous-time pricing formula is written as

$$V_c = \sum_{I=0}^{\infty} q(I)V_c(I) = \mathbb{E}_{\mathbf{\Pi}}[V_c(\xi)], \quad \xi \sim \mathbf{\Pi}(\lambda T), \tag{12}$$

where $\mathbf{\Pi}$ denotes the Poisson distribution.

Before presenting the convergence result, we introduce the following notation to simplify the expressions of the discrete-time formula:

$$q_N(I) \doteq \binom{N}{I}q^I(1-q)^{N-I}, \quad \text{where } \binom{N}{I} = \frac{N!}{I!(N-I)!},$$
$$V_N(I) \doteq e^{-rT}\sum_{h\ge h^{\text{call}}}(S_0u^h - K)\sum_{U=0}^{N-I}\binom{N-I}{U} \cdot p^U(1-p)^{N-I-U}\rho_{N+h-I-2U}^{(I)}. \tag{13}$$

It follows that we can rewrite the value of European call option V^{call} as

$$V_N \doteq V^{\text{call}} = \sum_{I=0}^{N} q_N(I)V_N(I) = \mathbb{E}_{\mathcal{B}}[V_N(\xi_N)], \quad \xi_N \sim \mathcal{B}(N, q), \tag{14}$$

where $\mathcal{B}$ denotes the binomial distribution.

Notice that in Equation (8), we assume the jump size follows a discrete distribution. If the jump-size distribution is continuous, we approximate the possible values of Y by integer powers of u (see Appendix D.1 for details):

$$Y \sim \hat{\mathcal{Y}}: \ \mathbb{P}(Y = u^j) = \hat{\rho}_j.$$

Convergence of the proposed discrete-time analytical pricing solution given by Equations (13) and (14) to the continuous-time formula (12) is established by the following theorem, whose proof is provided in Appendix C.2.

[2] This formula has no requirement on the form of the jump-size distribution, differing from the usual version found in the literature.

Figure 4 Algorithm for Pricing European Call and Put Options on the Discrete Jump-Diffusion Model

JUMP-DIFFUSION GENERATING FUNCTION (JDGF) EUROPEAN OPTION PRICING ALGORITHM
Input: starting price S_0, strike price K, maturity T, risk-free rate r, volatility σ, Poisson rate λ, discrete jump-size distribution $\tilde{\mathcal{Y}}$: $\{\rho_j\}_{j=M^-}^{M^+}$; parameters N, I_0.
Initialization:
Set $\tau = \dfrac{T}{N}$, $q = \lambda\tau$, $u = \mathrm{e}^{\sigma\sqrt{\tau}}$, $h^{\text{call}} = \left\lfloor \log_u \dfrac{K}{S_0} \right\rfloor$, $h^{\text{put}} = \left\lceil \log_u \frac{K}{S_0} \right\rceil$; $p = \dfrac{(\mathrm{e}^{r\tau} - \lambda\tau\tilde{\mu})/(1-\lambda\tau) - u^{-1}}{u - u^{-1}}$, where $\tilde{\mu} = \sum_{j=M^-}^{M^+} \rho_j u^j$.
For $I = 0$ **to** I_0,

- Calculate $q_N(I) = \dbinom{N}{I} q^I (1-q)^{N-I}$,
- **For** $U = 0$ **to** $N - I$, calculate $p_I(U) = \dbinom{N-I}{U} p^U (1-p)^{N-I-U}$;
- Calculate generating function $G_N(z; I) = \left(\sum_{j=M^-}^{M^+} \rho_j z^j\right)^I \left(\sum_{U=0}^{N-I} p_I(U) z^{2U+I-N}\right)$;

For $h = \min\{I_0(M^- + 1) - N, -N\}$ **to** $\max\{I_0(M^+ - 1) + N, N\}$,
Compute $\mathbb{P}_N(h) = [z^h] \sum_{I=0}^{I_0} q_N(I) G_N(z; I)$, where $[z^h]g(z)$ denotes the coefficient of z^h in polynomial $g(z)$.
Output: $V^{\text{call}} = \mathrm{e}^{-rT} \sum_{h \ge h^{\text{call}}} \mathbb{P}_N(h)(S_0 u^h - K)$, $V^{\text{put}} = \mathrm{e}^{-rT} \sum_{h \le h^{\text{put}}} \mathbb{P}_N(h)(K - S_0 u^h)$.

Theorem 2. *Assume the underlying asset follows a continuous jump-diffusion process, and the relations in (7) hold. For the same European call option on an underlying asset described by the discrete-time jump-diffusion process, $V_N \to V_c$ as $N \to \infty$, where V_N and V_c are defined by Equations (13)/(14) and (12), respectively; i.e.,*

$$\sum_{I=0}^{N} q_N(I) V_N(I) = \mathbb{E}_{\mathcal{B}}[V_N(\xi_N)] \xrightarrow{N\to\infty} \sum_{I=0}^{\infty} q(I) V_c(I) = \mathbb{E}_{\mathbf{\Pi}}[V_c(\xi)].$$

3.3. Algorithm Implementation and Complexity

The implemented algorithm includes two sets of truncation parameters: the maximum number of jumps realized in a lattice path, denoted by I_0, and two parameters determining the maximum and minimum values of the lattice, which are related to the maximum and minimum size of a single jump (M^- and M^+), denoted by M_0 and R. Details on choosing these parameters are given in Appendix D. The effect of parameter I_0 is simply to replace the first summation's upper limit of N in (11) with it, giving the following truncated probability weight:

$$\mathbb{P}_N(h) = \sum_{I=0}^{I_0} \sum_{U=0}^{N-I} \binom{N}{I, U} q^I (1-q)^{N-I} \cdot p^U (1-p)^{N-I-U} \rho_{N+h-I-2U}^{(I)}, \tag{15}$$

with I_0 is chosen as follows (see Appendix D.2 for justification):

$$I_0 = \min\left\{ j \,\middle|\, \sum_{i=j+1}^{\infty} \mathrm{e}^{-\lambda T} \frac{(\lambda T)^i}{i} < \epsilon \right\}, \tag{16}$$

where $\epsilon > 0$ is a small number (10^{-5} in the numerical examples later). Our proposed jump-diffusion generating function (JDGF) algorithm for implementing the European call and put option pricing formulas is given in Figure 4. Explicit pseudocode in MATLAB and Mathematica can be found in Appendix E. A method for obtaining the discrete jump-size distribution $\tilde{\mathcal{Y}}$: $\{\rho_j\}_{j=M^-}^{M^+}$ from $\mathcal{Y}$ or $\mathcal{Y}_c$, which is used in all of the numerical examples, is given in Appendix D.

We now show that in terms of computational complexity, the pricing formulas given by (5) and (15)—as implemented with the additional truncations parameters M_0 and R—represent a dramatic improvement over DP-based recursion. Specifically, we prove that the computational complexity of our proposed JDGF algorithm is $O(N^{1.5})$, whereas the complexity of the DP-based recursive algorithm on the same lattice is $O(N^3)$.[3]

Theorem 3. *The computational complexity of the JDGF option pricing algorithm implemented using (5), (15), and (16) is $O(N^{1.5})$.*

Proof. During the numerical estimation of the price of a European call option, as demonstrated in Appendix D, we only need to calculate

$$V_N = \sum_{I=0}^{I_0} q_N(I) V_N(I),$$

where $V_N(I)$ is given by Equation (13) for a call option (the put option expression is similar). Note that the value of I_0 defined by (16) is independent of N, so the computational complexity of V_N is the same as that of $V_N(I)$.

To compute the complexity of $V_N(I)$, we note that, as shown in Section 3.1 and Appendix B.1, the coefficient of $V_N(I)$ is exactly the coefficient of the term z^h

[3] The special truncation procedure for the multinomial lattice described in Amin (1993, p. 1852) results in an improved $O(N^{2.5})$ complexity, but this truncation procedure is less accurate and may exhibit numerical instabilities.

in the generating function $G_N(z;\tilde{\mathcal{P}};I)$ given by Equation (B2) in Appendix B.1. Since $V_N(I)$ only includes terms with $h \geq h^{\text{call}}$, the computational complexity of $V_N(I)$ is not more than $G_N(z;\tilde{\mathcal{P}};I)$, which we now show has complexity $O(N^{1.5})$.

Consider the form of $G_N(z;\tilde{\mathcal{P}};I)$ in the line prior to Equation (B2) in Appendix B.1:

$$G_N(z;\tilde{\mathcal{P}};I) = \sum_{U=0}^{N-I}\binom{N-I}{U}p^U(1-p)^{N-I-U} \cdot z^{2U+I-N}\left[\sum_{j=-\infty}^{+\infty}\rho_j z^j\right]^I;$$

$\sum_{U=0}^{N-I}\binom{N-I}{U}p^U(1-p)^{N-I-U}z^{2U+I-N}$ is a polynomial of $N-I+1$ monomials, and $\sum_{j=-\infty}^{+\infty}\rho_j z^j$ is approximated by $\sum_{j=M^-}^{M^+}\rho_j z^j$, as discussed in Appendix D.1. Note that M^- and M^+ are both $O(N^{0.5})$—the same as M_0. Therefore, for a fixed I, the function $G_N(z;\tilde{\mathcal{P}};I)$ is a product of two polynomials having $O(IN^{0.5})$ and $O(N)$ monomials, so the complexity of $G_N(z;\tilde{\mathcal{P}};I)$ is $O(N^{1.5})$; thus, the complexity of the JDGF option pricing algorithm is at most $O(N^{1.5})$. □

The next result establishes the computational complexity of DP-based recursion on the same multinomial lattice constructed in Section 2.

Proposition 2. *The computational complexity of the DP-based recursive algorithm on the multinomial lattice with one-step move probabilities given by* (10) *is* $O(N^3)$.

Proof. Since N is the number of time steps, at time step k, there are $kO(N^{0.5})$ points in the multinomial lattice. For each point, the price could reach $O(N^{0.5})$ possible points at the next time step; thus the holding value for each point is calculated on $O(N^{0.5})$ possible points using backward induction. Therefore, the complexity for estimating the expected option holding value is $\sum_{k=1}^{N-1}kO(N^{0.5})O(N^{0.5}) = O(N^3)$. □

3.4. Time-Varying Volatility Case

We now extend the analysis to the setting of time-dependent volatility, defined by

$$\sigma(t) = \sigma_s, \quad t \in [t_{s-1}, t_s),\ s = 1, \ldots, \varsigma, \tag{17}$$

where $0 = t_0 < t_1 < \cdots < t_\varsigma = T$; i.e., the volatility is piecewise constant over each time interval $[t_{s-1}, t_s)$.

We divide each time interval $[t_{s-1}, t_s)$ into N_s equal subintervals:

$$\tau_s \doteq \frac{t_s - t_{s-1}}{N_s}, \quad s = 1, \ldots, \varsigma.$$

Note that τ_s is chosen in each time interval $[t_{s-1}, t_s)$ to guarantee that $u_s = e^{\sigma_s\sqrt{\tau_s}}$, $s = 1, \ldots, \varsigma$ remains a constant (which is denoted as u) throughout the entire time horizon.

As a result, the one-step move probability in each subinterval $[t_{s-1}, t_s)$ retains the same form as Equation (2):

$$\mathbb{P}(X_k = u^i) = p_i^{(s)}, \quad i \in \mathbb{Z}, \tag{18}$$

where $k \in (N_1 + \cdots + N_{s-1}, N_1 + \cdots + N_{s-1} + N_s]$ is an integer representing the time step in the time interval $[t_{s-1}, t_s)$. Thus, for the general lattice model of an arbitrary discrete stochastic process, we have the following pricing formula.

Theorem 4. *On the multinomial lattice where the underlying asset price has time-varying volatility* (17) *and one-step move probabilities* (18), *the probability weight is given by*

$$\mathbb{P}_{\vec{N}}(h) = \sum_{h=h_1+\cdots+h_\varsigma}\prod_{s=1}^{\varsigma}\mathbb{P}_{N_s}(h_s),$$

where $\vec{N} \doteq (N_1, N_2, \ldots, N_\varsigma)$, *and* $\mathbb{P}_{N_s}(h_s)$ *represents the probability weight in time interval* $[t_{s-1}, t_s)$ *as defined in* (4).

The proof is provided in Appendix B.2.

Proposition 3. *For a European option with strike price* K, *expiration date* T, *and underlying asset initial value* S_0, *where the underlying asset price has time-varying volatility* (17) *and one-step move probabilities* (18), *the values of the call and put options under a multinomial lattice with grid size* u *can be written as*

$$V_{\vec{N}}^{\text{call}} = e^{-rT}\sum_{h \geq h^{\text{call}}}\mathbb{P}_{\vec{N}}(h)(S_0u^h - K) \quad \text{and}$$

$$V_{\vec{N}}^{\text{put}} = e^{-rT}\sum_{h \leq h^{\text{put}}}\mathbb{P}_{\vec{N}}(h)(K - S_0u^h),$$

respectively, where r *is the riskless rate, and* h^{call} *and* h^{put} *are defined in* (6).

Next we focus on the setting where the underlying asset dynamics follows a jump-diffusion process. The one-step move probabilities are given by

$$\mathbb{P}(X_k) = \begin{cases} (1-q_s)p_s & X_k = W = 1, \\ (1-q_s)(1-p_s) & X_k = W = -1, \\ q_s\rho_j & X_k = J = j, \end{cases} \tag{19}$$

where $k \in (N_1 + \cdots + N_{s-1}, N_1 + \cdots + N_{s-1} + N_s]$ is an integer representing the time step in the time interval $[t_{s-1}, t_s)$, and q_s, p_s are defined as

$$q_s = \lambda\tau_s, \quad p_s = \frac{(e^{r\tau_s} - \lambda\tau_s\mathbb{E}_{\mathcal{Y}}[Y])/(1-\lambda\tau_s) - d}{u - d}. \tag{20}$$

Combining Theorem 1, Theorem 4, and Proposition 3, we have the following corollary.

Corollary 2. *For a European option with strike price K, expiration date T, and underlying asset initial value S_0, where the underlying asset price follows a discrete-time jump-diffusion process with time-varying volatility (17) and one-step move probabilities (19), the values of the call and put options under a multinomial lattice with grid size u can be written as*

$$V_{\vec{N}}^{\text{call}} = \mathrm{e}^{-rT} \sum_{h \geq h^{\text{call}}} \mathbb{P}_{\vec{N}}(h)(S_0 u^h - K) \quad \textit{and}$$

$$V_{\vec{N}}^{\text{put}} = \mathrm{e}^{-rT} \sum_{h \leq h^{\text{put}}} \mathbb{P}_{\vec{N}}(h)(K - S_0 u^h),$$

respectively, where r is the riskless rate, h^{call} and h^{put} are defined in (6), and

$$\mathbb{P}_{\vec{N}}(h) = \sum_{h=h_1+\cdots+h_s} \prod_{s=1}^{s} \sum_{I=0}^{N_s} \sum_{U=0}^{N_s-I} \binom{N_s}{I, U} q_s^I (1-q_s)^{N_s-I} \cdot p_s^U (1-p_s)^{N_s-I-U} \rho_{N_s+h_s-2U-I}^{(I)},$$

where q_s and p_s are defined in (20).

4. Numerical Results

In this section, we price European options in our discrete-time framework for the underlying asset price following a jump-diffusion process. We first consider log-normal and log-double-exponential jump-size distributions, where analytical results exist for the continuous-time setting. The results demonstrate that the proposed JDGF option pricing algorithm converges to the continuous-time model for large enough N and is computationally efficient, especially when compared with dynamic programming. In particular, the numerical experiments validate the theoretical computational complexity. We then investigate the impact of different jump-size distributions on option prices using parameters estimated from daily data on the stock price of IBM. All numerical results were obtained using a MATLAB implementation of the JDGF European option pricing algorithm in Figure 4 (see Appendix E for pseudocode in both MATLAB and Mathematica), with CPU times reported using a 2.90 GHz 8 GB RAM desktop computer.

4.1. Merton's Log-Normal Distribution

We price a European put option with diffusion parameters $r = 0.08$ and $\sigma^2 = 0.05$, Poisson rate $\lambda = 5$, jump-size distribution $\ln \mathcal{N}(-0.025, 0.05)$, initial asset price $S_0 = 40$, strike price $K = 45$, and expiration $T = 0.5$. The jump truncation parameter is $I_0 = 12$, calculated using Equation (16) for $\epsilon = 10^{-5}$. Under different time steps N, we calculate the truncation parameters M_0 and R using the procedure in Appendix D.

European put option prices under different time steps N are shown in Table 1, where the option price under Merton's pricing formula is given in the last column.

Table 1 **Pricing European Put Options Under Log-Normal Jump-Size Distribution** $\ln \mathcal{N}(-0.025, 0.05)$; $\sigma^2 = 0.05$, $S_0 = 40$, $r = 0.08$, $T = 0.5$, $\lambda = 5$, $K = 45$; $I_0 = 12$

N	500	1,000	1,500	2,000	2,500	3,000	Merton
M_0	95	135	165	190	213	233	
R	10	14	17	19	21	23	
Price	7.9028	7.9037	7.9044	7.9039	7.9040	7.9044	7.9045
CPU (secs.)	0.2	0.3	0.6	0.9	1.3	1.8	0.05

Table 2 **Pricing European Call Options Under Log-Double-Exponential Jump-Size Distribution** $\ln \mathscr{E}(0.6, 0.4, 20, 20)$; $\sigma = 0.3$, $S_0 = 100$, $r = 0.05$, $T = 1$, $\lambda = 3$; $I_0 = 12$

	N					
K	500	1,000	2,000	3,000	5,000	Kou
M_0	61	85	121	148	190	
R	955	194	1	1	1	
90	20.4244	20.4341	20.4485	20.4497	20.4535	20.4563
100	15.0987	15.1173	15.1261	15.1290	15.1312	15.1343
110	10.9424	10.9630	10.9719	10.9747	10.9772	10.9813
CPU (secs.)	0.13	0.32	0.95	1.85	4.59	0.13

4.2. Kou's Log-Double-Exponential Distribution

We price European call options with diffusion parameters $r = 0.05$ and $\sigma = 0.3$; Poisson rate $\lambda = 3$; jump size distribution $\ln \mathscr{E}(0.6, 0.4, 20, 20)$ with probability density function

$$f_{\ln Y}(y) = 0.6 \cdot 20\mathrm{e}^{-20y} \cdot \mathbb{1}_{\{y \geq 0\}} + 0.4 \cdot 20\mathrm{e}^{20y} \cdot \mathbb{1}_{\{y<0\}};$$

initial asset price $S_0 = 100$; strike prices $K = 90, 100, 110$; and expiration $T = 1$.

We choose M_0, R, and I_0 based on the rules described in Appendix D. The prices using our pricing formula for different time steps N are shown in Table 2, where the corresponding values under Kou's pricing formula are displayed in the last column.

4.3. Computational Complexity

The theoretical computational complexity results in Section 3.3 indicate a substantial improvement over the DP-based recursive approach. Here, we test empirical performance on several numerical examples, comparing the computation time of our JDGF algorithm with DP. We fix $I_0 = 12$ and price the European put option with riskless rate $r = 0.08$, strike price $K = 45$, and expiration time $T = 0.5$. The underlying asset has initial price $S_0 = 40$, volatility $\sigma^2 = 0.05$, and jump frequency parameter $\lambda = 5$, with the same log-normal jump-size distribution used in Section 4.1. Table 3 shows the option price and CPU time with different time steps N under both algorithms. The results indicate that the JDGF algorithm is both more accurate and computationally more efficient. CPU time as a function of the number of time steps N for both algorithms is plotted in Figures 5

Table 3 **Pricing European Put Options Under Log-Normal Jump-Size Distribution** $\ln\mathcal{N}(-0.025, 0.05)$; $\sigma^2 = 0.05$, $S_0 = 40$, $r = 0.08$, $T = 0.5$, $\lambda = 5$, $K = 45$; $l_0 = 12$

N	100	200	500	800	1,000	1,500	2,000	5,000
DP recursive								
Option price	7.8961	7.9006	7.9036	7.9036	7.9043	7.9045		
CPU time (secs.)	1.90	11.73	127	468	953	3,072		
JDGF								
Option price	7.8890	7.8982	7.9028	7.9021	7.9037	7.9044	7.9039	7.9046
CPU time (secs.)	0.02	0.03	0.13	0.24	0.33	0.59	0.94	4.61
Monte Carlo								
Price range					7.8884 to 7.9605			
CPU time (secs.)					196 (3.4)			

Note. The Monte Carlo results are based on 10 macroreplications of 100,000 paths with $N = 1{,}000$, where the entries show the option price range and CPU time mean (standard errors in parentheses).

and 6, and the results support the theoretical computational complexity results.

4.4. Sensitivity of Option Prices to Jump-Size Distributions

To investigate the sensitivity of option prices to the jump-size distribution, we compare the option prices with various jump-size distributions by applying the JDGF algorithm to an example where the diffusion component is identical and the first two moments of the jump-size distribution are matched.

The underlying asset price parameters are fitted using daily data on the stock price of IBM from January 1962 to December 2015. To fit the discrete jump-diffusion model, we use the box-plot method to determine jumps. Specifically, we assume values of the log returns $x_i = \ln(S_{t_i+1}/S_{t_i})$ outside the range $(Q_1 - kR_f, Q_3 + kR_f)$ constitute jumps, where Q_1 is the lower quartile, Q_3 is the upper quartile, the interquartile range R_f is defined as $Q_3 - Q_1$, and k is a constant. Here, we set $k = 1.5$, giving a jump range of higher than 3.28% or lower than −3.23%. Then we have the empirical distribution

$$\mathscr{H}\colon \mathbb{P}(x_i) = \frac{1}{618}, \quad i = 1, \ldots, 618.$$

Then the parameters for the discrete jump-diffusion model estimated from the data are volatility $\sigma = 0.19$ and Poisson arrival rate $\lambda = 11.5$ for the jump-diffusion process and mean $\alpha = 0.0011$ and standard deviation $\delta = 0.05$ for the log-jump distribution. In addition to the discrete empirical distribution, three other (continuous) distributions are considered.

- Normal $\mathcal{N}$:

$$f^{\mathcal{N}}_{\ln Y}(y) = \frac{1}{\sqrt{2\pi}0.05} e^{-(y-0.0011)^2/(2\times 0.05^2)}$$

- Double-exponential $\mathscr{E}$:

$$f^{\mathscr{E}}_{\ln Y}(y) = 0.335 \times 16.5 e^{-16.5y} \times \mathbb{1}_{\{y\geq 0\}} + 0.665 \times 34.5 e^{34.5y} \times \mathbb{1}_{\{y<0\}}$$

- Double-Gamma $\mathscr{G}$:

$$f^{\mathscr{G}}_{\ln Y}(y) = 0.48 \times 35.3 e^{-35.3y} \times \mathbb{1}_{\{y\geq 0\}} + 0.52 \frac{(-y)^{0.21-1}}{\Gamma(0.21)0.1145^{0.21}} e^{y/0.1145} \times \mathbb{1}_{\{y<0\}}$$

The density functions of the three continuous distributions and the discrete empirical distribution are shown in Figure 7.

Figure 5 **(Color online) CPU Time as a Function of the Number of Time Steps N for the DP Recursive and JDGF Algorithms**

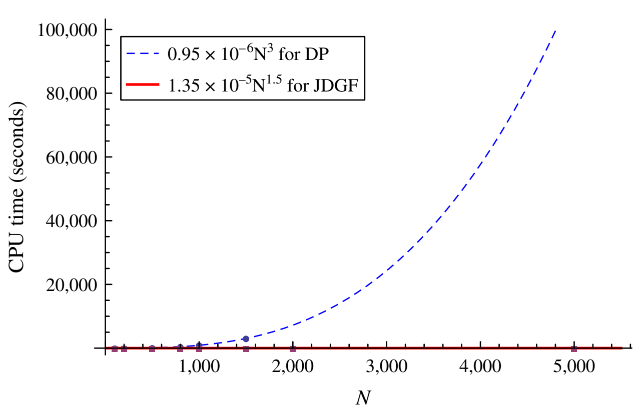


Figure 6 **(Color online) CPU Time on Log Scale as a Function of the Number of Time Steps N for the Two Algorithms**

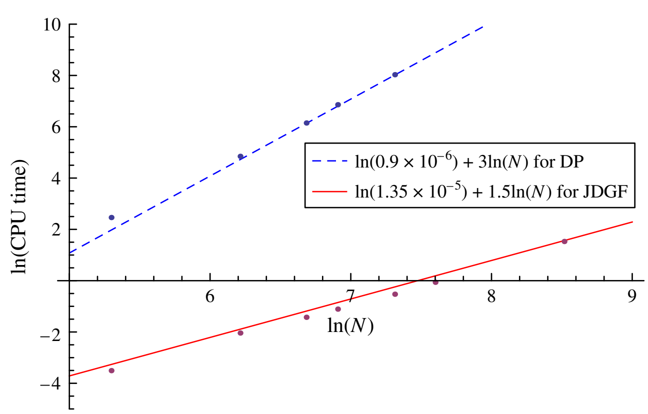
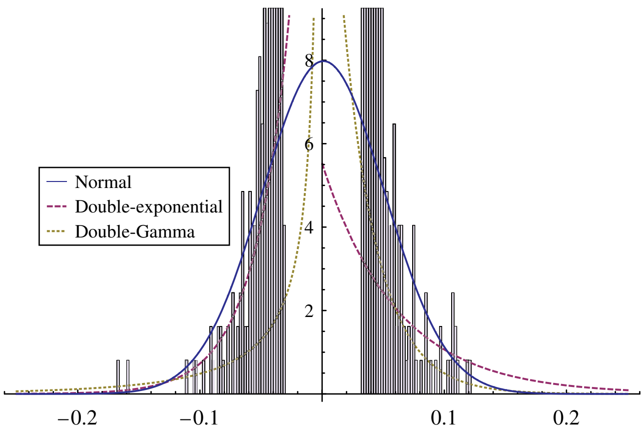


Figure 7 (Color online) Plots of Four Probability Density Functions



Note. To see more clearly what the tails of the distributions look like, the parts of the Double-exponential and Double-Gamma density functions with log jumps close to zero are omitted in the graph.

We price a European put option on the four different jump-diffusion processes, with riskless rate $r = 0.05$, expiration date $T = 0.25$, and initial price $S_0 = 100$. The option prices under different jump-size distributions and different strike prices are given in Table 4, where we observe significant differences in option prices (as well as different skewness and kurtosis, not shown here). If the empirical distribution is taken as truth, the option prices using other jump-size distributions fluctuate within several cents of the "true" prices across the range of strike prices, whether the option is in-the-money, at-the-money, or out-of-the-money.

These numerical results illustrate that the proposed model is considerably robust in handling any form of jump-size distribution. In practice, one can adjust the jump-size distribution accordingly to see its impact on the option prices and to assess the model risk for the pricing model as a result of the selection of the jump-size distributions.

5. Conclusions and Future Research

Using the generating function technique from enumerative combinatorics, we derive an analytic pricing formula for European options under a discrete-time jump-diffusion framework. In contrast to the continuous setting, where the pricing methodology is usually tailored to a specific jump-size distribution, our pricing formula can be applied to general jump-size distributions. As the number of time steps N goes to infinity, the formula converges to the corresponding continuous-time formula of Merton (1976), with a computational complexity of $O(N^{1.5})$. Numerical experiments demonstrate the flexibility and efficiency of the JDGF algorithm, as well as the sensitivity of option prices to different jump-size distributions. Computationally, the method is orders of magnitude faster than Monte Carlo simulation for European options, although it would worthwhile to investigate how it performs against various transform methods (e.g., Carr and Madan 1999, Cai et al. 2014, Feng and Linetsky 2009, Feng and Lin 2013).

The empirical results in Section 4.4 used asset prices to estimate the parameters of the jump-diffusion model are for the purpose of demonstrating sensitivity of option prices to different types of jump-size distributions. In practice, market data on actual option prices are used to calibrate any option pricing model. Although not a focus of this work, a calibration procedure based on actual option prices is required to make the JDGF algorithm implementable for a practitioner. We briefly suggest one possible approach here, which adopts the regularization method of Cont and Tankov (2004); however, determining a good procedure is definitely a critical need for further research.

The calibration approach in Cont and Tankov (2004) is a nonparametric method for fitting a (finite activity) jump-diffusion process to a finite set of observed option prices. Their approach appears to be well suited to our model, since it can be applied to an arbitrary discrete jump distribution, such as an empirical distribution, and is especially effective for compound

Table 4 Prices and Implied Volatilities of European Put Option Under Different Jump-Size Distributions Based on IBM Daily Asset Price Data from January 1962 to December 2015

	Empirical		Normal		Double-Gamma		Double-exponential	
K	Price	Imp. vol.	Price	Imp. vol.	Price	Imp. vol.	Price	Imp. vol.
110	10.6935	0.2548	10.5909 (−0.96%)	0.2487	10.4231 (−2.53%)	0.2385	11.2193 (+4.92%)	0.2854
100	4.4394	0.2543	4.3489 (−2.04%)	0.2497	4.2149 (−5.06%)	0.2429	4.7885 (+7.86%)	0.2720
90	1.1658	0.2561	1.1119 (−4.62%)	0.2571	1.1814 (+1.34%)	0.2574	1.2751 (+9.38%)	0.2649

Notes. Here, $S_0 = 100$, $r = 0.05$, $T = 0.25$, $\sigma = 0.19$, and $\lambda = 11.5$. The percent difference relative to the empirical distribution price is reported in parentheses.

Poisson processes such as the jump process. Moreover, in numerical experiments they reported that the calibration procedure did a good job of recovering option prices generated (artificially) from Kou's model.

Because jump-diffusion models have nonunique martingale measures for option pricing, additional criteria are needed to determine a unique measure. The main idea of the approach in Cont and Tankov (2004) is to add to the commonly used least-squares criterion a convex penalization term that yields a unique and stable solution to the inverse problem and for which a gradient-based optimization algorithm can be applied to find the optimal measure. Basically, the additional term minimizes the relative entropy (or Kullback–Leibler divergence) from a prior distribution, which must be specified, in addition to some weights and regularization parameters, for which they provide some guidelines. In terms of the prior distribution, in our setting one possible candidate could be obtained using the box-plot method in Section 4.4.

Although we derived the explicit pricing formula for the jump-diffusion setting, the proposed method can be applied to more general independent increments processes that can be modeled using a multinomial lattice. Furthermore, the methodology developed here should also be applicable in other contexts beyond option pricing, e.g., interest rate derivatives. Path-dependent options present another challenge, and future research could tackle barrier options extending ideas from Li and Zhao (2009) to multinomial lattices and American-style options along the lines of Laprise et al. (2006). Finally, using the model for hedging is clearly an important practical topic worthy of further investigation.

Acknowledgments
The authors thank the associate editor and three referees for their constructive comments and suggestions, which have led to an improved exposition. This work was supported in part by the U.S. National Science Foundation [Grants CMMI-1362303 and CMMI-1434419], the U.S. Air Force Office of Scientific Research [Grant FA9550-15-10050], the Natural Science Foundation of China [Grants 71171119 and 71671094], and the Fundamental Research Funds for the Central Universities [Grant NKZXTD1403]. The views and opinions expressed in this article are solely the authors' own and do not reflect the business and positions of R. Wu's affiliation.

Appendix A. Review of Generating Functions
The generating function for sequence $\{f_n\}_{n\geq 0}$ is given by

$$F(x)=\sum_{n\geq 0} f_n x^n.$$

We illustrate two examples relevant to lattice path enumeration.

- Let $\mathscr{L}$ be the set of all lattice paths. Define $\mathscr{L}_n=\{\alpha\in\mathscr{L}\colon \ell(\alpha)=n\}$. Let f_n be the number of elements in $\mathscr{L}_n$, denoted by $\#[\mathscr{L}_n]$. The generating function for the sequence $\{f_n\}$ is

$$\mathscr{L}(x)=\sum_{n\geq 0}\#[\mathscr{L}_n]x^n.$$

Since $\mathscr{L}=\{\alpha\in\mathscr{L}_n\colon n\geq 0\}$, we also call the above equation the generating function for the set $\mathscr{L}$.

- Let $\mathscr{U}=\{(1)\}$ be the set consisting of a single lattice path $\alpha=(1)$. Then $\mathscr{L}_n$ is the empty set except for $\mathscr{U}_1=\mathscr{U}$. It follows that $\mathscr{U}(x)=x$. Similarly, if we let $\mathscr{D}=\{(-1)\}$, then $\mathscr{D}(x)=x$. Thus in our lattice framework, x can be understood as one step of a lattice path, which is either a (1) step or a (-1) step.

The following lemma is the basic tool, and it explains the combinatorial meaning of the product of generating functions.

Lemma 1. *Let $\mathscr{A}$, $\mathscr{B}$, and $\mathscr{L}$ be three sets of lattice paths. If any path γ in $\mathscr{L}$ can be uniquely factored as $\alpha\cdot\beta\doteq(a_1,a_2,\dots,a_{l_\alpha},b_1,b_2,\dots,b_{l_\beta})$, $\alpha=(a_1,a_2,\dots,a_{l_\alpha})\in\mathscr{A}$, and $\beta=(b_1,b_2,\dots,b_{l_\beta})\in\mathscr{B}$, and for any $\alpha\in\mathscr{A}$ and $\beta\in\mathscr{B}$, we have $\alpha\cdot\beta\in\mathscr{L}$, then $\mathscr{L}(x)=\mathscr{A}(x)\mathscr{B}(x)$.*

Proof. The generating functions for the set $\mathscr{A}$ and $\mathscr{B}$ are, respectively,

$$\mathscr{A}(x)=\sum_{n\geq 0}\#[\mathscr{A}_n]x^n \quad \mathscr{B}(x)=\sum_{n\geq 0}\#[\mathscr{B}_n]x^n, \quad \text{and}$$

where $\mathscr{A}_n=\{\alpha\in\mathscr{A}\colon \ell(\alpha)=n\}$, and $\mathscr{B}_n=\{\beta\in\mathscr{B}\colon \ell(\beta)=n\}$. Since any path γ in $\mathscr{L}$ is uniquely factored as $\alpha\cdot\beta$, $\alpha\in\mathscr{A}$, and $\beta\in\mathscr{B}$, and for any $\alpha\in\mathscr{A}$ and $\beta\in\mathscr{B}$, we have $\alpha\cdot\beta\in\mathscr{L}$, then we can obtain that

$$\#[\mathscr{L}_n]=\sum_{k=0}^{n}\#[\mathscr{A}_k]\#[\mathscr{B}_{n-k}],$$

where $\mathscr{L}_n=\{\gamma\in\mathscr{L}\colon \ell(\gamma)=n\}$. By the usual Cauchy product rule of power series, we have

$$\begin{aligned}\mathscr{L}(x)&=\sum_{n\geq 0}\#[\mathscr{L}_n]x^n=\Big(\sum_{n\geq 0}\#[\mathscr{A}_n]x^n\Big)\Big(\sum_{n\geq 0}\#[\mathscr{B}_n]x^n\Big)\\&=\mathscr{A}(x)\mathscr{B}(x).\qquad\square\end{aligned}$$

Given a set $\mathscr{L}$ of lattice paths, if there exists a subset $\mathscr{A}$ of $\mathscr{L}$ such that every path $\alpha\in\mathscr{L}$ can be uniquely factored as $\alpha_1\alpha_2\cdots\alpha_m$ with $\alpha_i\in\mathscr{A}$, then we say that $\mathscr{A}$ is the *prime* of $\mathscr{L}$.

Lemma 2. *Let $\mathscr{L}$ be a set of lattice paths. If $\mathscr{A}$ is the prime of $\mathscr{L}$, then*

$$\mathscr{L}(x)=\frac{1}{1-\mathscr{A}(x)}.$$

Proof. Let $\tilde{\mathscr{L}}_m$ be the set of lattice paths that are uniquely factored as $\alpha_1\alpha_2\cdots\alpha_m$, $\alpha_i\in\mathscr{A}$. By Lemma 1, it is easy to see $\tilde{\mathscr{L}}_m(x)=(\mathscr{A}(x))^m$, where $\tilde{\mathscr{L}}_0(x)=1$ corresponds to the void path. So the generating function for the set $\mathscr{L}$ is

$$\mathscr{L}(x)=\sum_{m\geq 0}\tilde{\mathscr{L}}_m(x)=\sum_{m\geq 0}(\mathscr{A}(x))^m=\frac{1}{1-\mathscr{A}(x)}.\qquad\square$$

Appendix B. Support for Derivation of the Discrete Pricing Formula

B.1. Generating Function for Lattice Height Probability

Lemma 3. *On a multinomial lattice with one-step move probabilities given by (10), the generating function of N-length lattice path probability weight is*

$$G_N(z;\tilde{\mathcal{P}})=\sum_{I=0}^{N}\binom{N}{I}q^I(1-q)^{N-I}G_N(z;\tilde{\mathcal{P}};I), \tag{B1}$$

where

$$G_N(z;\tilde{\mathcal{P}};I)=\sum_{U=0}^{N-I}\sum_{L=-\infty}^{+\infty}\binom{N-I}{U}p^U(1-p)^{N-I-U}\rho_L^{(I)}z^{2U+I+L-N} \tag{B2}$$

is the generating function of N-length lattice path probability weight with I jumps.

Proof. Using the generating function techniques of Appendix A, the generating function of the jump-diffusion multinomial lattice is

$$G(z,x;\tilde{\mathcal{P}})=\frac{1}{1-[(1-q)pz^1+(1-q)(1-p)z^{-1}+q\sum_{-\infty}^{\infty}\rho_j z^j]x},$$

where x is a variable whose power indicates the length of paths, z's power indicates the height of paths, and p, q, and ρ_j's represent related probability of $\tilde{\mathcal{P}}$ in (10).

For paths of length N, the generating function is the coefficient of term x^N in the function above, written as

$$G_N(z;\tilde{\mathcal{P}})=\left[(1-q)pz^1+(1-q)(1-p)z^{-1}+q\sum_{-\infty}^{\infty}\rho_j z^j\right]^N.$$

We expand this N-steps generating function as follows:

$$\begin{aligned}
&G_N(z;\tilde{\mathcal{P}})\\
&=\sum_{I=0}^{N}\binom{N}{I}[(1-q)pz^1+(1-q)(1-p)z^{-1}]^{N-I}\left(q\sum_{-\infty}^{\infty}\rho_j z^j\right)^I\\
&=\sum_{I=0}^{N}\binom{N}{I}q^I(1-q)^{N-I}[pz^1+(1-p)z^{-1}]^{N-I}\left(\sum_{-\infty}^{\infty}\rho_j z^j\right)^I\\
&\doteq\sum_{I=0}^{N}\binom{N}{I}q^I(1-q)^{N-I}G_N(z;\tilde{\mathcal{P}};I),
\end{aligned}$$

which is Equation (B1), and

$$\begin{aligned}
&G_N(z;\tilde{\mathcal{P}};I)\\
&\quad=[pz^1+(1-p)z^{-1}]^{N-I}\left(\sum_{j=-\infty}^{+\infty}\rho_j z^j\right)^I\\
&\quad=\sum_{U=0}^{N-I}\binom{N-I}{U}p^U(1-p)^{N-I-U}z^{U-(N-I-U)}\sum_{L=-\infty}^{+\infty}\rho_L^{(I)}z^L\\
&\quad=\sum_{U=0}^{N-I}\sum_{L=-\infty}^{+\infty}\binom{N-I}{U}p^U(1-p)^{N-I-U}\rho_L^{(I)}z^{2U+I+L-N},
\end{aligned}$$

which is Equation (B2). □

B.2. Proof of Time-Varying Lattice Height Probability

From the proof of Theorem 1 and Lemma 3, we know that for paths with length N_s and probability measure $\tilde{\mathcal{P}}_s\doteq\tilde{\mathcal{P}}|_{[t_{s-1},\,t_s]}$, the lattice probability weight $\mathbb{P}_{N_s}(h_s)$ is the coefficient of the term z^{h_s} of its generating function $G_{N_s}(z;\tilde{\mathcal{P}}_s)$. Therefore, from Lemma 1, during the time interval $[0,\ T]$, the generating function under the probability measure $\tilde{\mathcal{P}}$ is

$$G_{\vec{N}}(z;\tilde{\mathcal{P}})=\prod_{s=1}^{s}G_{N_s}(z;\tilde{\mathcal{P}}_s),\quad \vec{N}\doteq(N_1,N_2,\ldots,N_s).$$

As a result, the probability weight $\mathbb{P}_{\vec{N}}(h)$, which is the coefficient of the z^h term, is

$$\begin{aligned}
\mathbb{P}_{\vec{N}}(h)&=[z^h]G_{\vec{N}}(z;\tilde{\mathcal{P}})=\sum_{h=h_1+\cdots+h_s}\prod_{s=1}^{s}[z^{h_s}]G_{N_s}(z;\tilde{\mathcal{P}}_s)\\
&=\sum_{h=h_1+\cdots+h_s}\prod_{s=1}^{s}\mathbb{P}_{N_s}(h_s).\qquad\square
\end{aligned}$$

Appendix C. Proof of the Convergence of Discrete Pricing Formula

C.1. Support for the Proof

In Section 3.2, we want to prove Theorem 2:

$$\begin{aligned}
V_N&=\sum_{I=0}^{N}V_N(I)q_N(I)=\mathbb{E}_{\mathcal{B}}[V_N(\xi_N)]\xrightarrow{N\to\infty}V_c\\
&\qquad\qquad=\sum_{I=0}^{\infty}V_c(I)q(I)=\mathbb{E}_{\mathbf{\Pi}}[V_c(\xi)].
\end{aligned}$$

Here, we lay the foundation for the proof.

First we take some transforms to make the relationship clearer, and $h_0\doteq h^{\text{call}}$. The coefficient of $\mathrm{e}^{-rT}K$ in $V_N(I)$ is given by

$$\begin{aligned}
&\sum_{h\ge h_0}\sum_{U=0}^{N-I}\binom{N-I}{U}p^U(1-p)^{N-I-U}\hat{\rho}_{N+h-I-2U}^{(I)}\\
&\quad=\sum_{U=0}^{N-I}\sum_{L+I+2U-N\ge h_0}\binom{N-I}{U}p^U(1-p)^{N-I-U}\hat{\rho}_L^{(I)}\\
&\quad=\sum_{U=0}^{N-I}\sum_{L\ge h_0+N-I-2U}\hat{\rho}_L^{(I)}\binom{N-I}{U}p^U(1-p)^{N-I-U}\\
&\quad=\sum_{L\ge h_0-(N-I)}\hat{\rho}_L^{(I)}\sum_{U=\lceil(h_0+N-I-L)/2\rceil}^{N-I}\binom{N-I}{U}p^U(1-p)^{N-I-U}\\
&\quad=\mathbb{E}^L\left[\sum_{U=\lceil(h_0+N-I-L)/2\rceil}^{N-I}\binom{N-I}{U}p^U(1-p)^{N-I-U}\right]\\
&\quad\doteq\mathbb{E}^L[\psi(N,I,p)],
\end{aligned}$$

where the superscript L indicates that the expectation $\mathbb{E}$ is under the distribution of random variable L.

Similarly, the coefficient of S_0 of $V_N(I)$ is given by

$$\begin{aligned}
&\mathrm{e}^{-rT}\sum_{h\ge h_0}u^h\sum_{U=0}^{N-I}\binom{N-I}{U}p^U(1-p)^{N-I-U}\hat{\rho}_{N+h-I-2U}^{(I)}\\
&=\mathrm{e}^{-rT}\mathbb{E}^L\left[\sum_{U=\lceil(h_0+N-I-L)/2\rceil}^{N-I}u^{L+I+2U-N}\binom{N-I}{U}p^U(1-p)^{N-I-U}\right]
\end{aligned}$$

$$
\begin{aligned}
&= e^{-rT}\mathbb{E}^L\left[\sum_{U=\lceil(h_0+N-I-L)/2\rceil}^{N-I} u^L u^U d^{N-I-U}\binom{N-I}{U}p^U(1-p)^{N-I-U}\right]\\
&= e^{-rT}\mathbb{E}^L\left[\sum_{U=\lceil(h_0+N-I-L)/2\rceil}^{N-I} u^L \tilde{r}^{N-I}\binom{N-I}{U}\right.\\
&\qquad\left.\cdot\left(\frac{u}{\tilde{r}}p\right)^U\left(\frac{d}{\tilde{r}}(1-p)\right)^{N-I-U}\right]\\
&= e^{-rT}\tilde{r}^{N-I}\mathbb{E}^L\left[u^L\sum_{U=\lceil(h_0+N-I-L)/2\rceil}^{N-I}\binom{N-I}{U}\tilde{p}^U(1-\tilde{p})^{N-I-U}\right]\\
&\doteq e^{-rT}\tilde{r}^{N-I}\mathbb{E}^L[u^L\psi(N,I,\tilde{p})],\quad \text{where } \tilde{r}=\frac{e^{rt}-\lambda t\mu}{1-\lambda t},\ \tilde{p}=\frac{u}{\tilde{r}}.
\end{aligned}
$$

After separating into the two terms along the lines of the Black–Scholes formula, we now have

$$
\begin{aligned}
V_N(I) &= S_0 e^{-rT}\tilde{r}^{N-I}\,\mathbb{E}^L[u^L\psi(N,I,\tilde{p})]\\
&\quad - e^{-rT}K\,\mathbb{E}^L[\psi(N,I,p)], \qquad \text{(C1)}\\
V_c(I) &= S_0 e^{-\lambda(\mu-1)T}\,\mathbb{E}^Y\left[\prod_{i=1}^{I} y_i\Psi(I,\sigma^2/2)\right]\\
&\quad - e^{-rT}K\,\mathbb{E}^Y[\Psi(I,-\sigma^2/2)], \qquad \text{(C2)}
\end{aligned}
$$

where the functions

$$\psi(N,I,p)\doteq\sum_{U=\lceil(h_0+N-I-L)/2\rceil}^{N-I}\binom{N-I}{U}p^U(1-p)^{N-I-U}$$

and $\Psi(I,\sigma^2/2)\doteq\mathcal{N}\left(\dfrac{\ln(e^{-\lambda\mu T}\prod_{i=1}^I y_i S_0/K)+(r+\sigma^2/2)T}{\sigma\sqrt{T}}\right).$

Lemma 4. *For fixed* I, $V_N(I)\to_{N\to\infty}V_c(I)$.

Proof. We consider the coefficients of S_0 and $e^{-rT}K$ separately.

(i) *Coefficient of* $e^{-rT}K$: We want to show that the relation $\mathbb{E}^L[\psi(N,I,p)]\xrightarrow{N\to\infty}\mathbb{E}^Y[\Psi(I,-\sigma^2/2)]$. First we have

$$
\begin{aligned}
&1-\psi(N,I,p)\\
&\quad=1-\sum_{U=\lceil(h_0+N-I-L)/2\rceil}^{N-I}\binom{N-I}{U}p^U(1-p)^{N-I-U}\\
&\quad=\mathbb{P}\left(U\le\left\lceil\frac{h_0+N-I-L}{2}\right\rceil-1\right)\\
&\quad=\mathbb{P}\left[\frac{U-(N-I)p}{\sqrt{(N-I)p(1-p)}}\right.\\
&\qquad\qquad\left.\le\frac{\lceil(h_0+N-I-L)/2\rceil-1-(N-I)p}{\sqrt{(N-I)p(1-p)}}\right].
\end{aligned}
$$

We substitute

$$h_0=\left\lceil\frac{\ln(K/S_0)}{\ln u}\right\rceil,\quad L=\sum_{i=1}^{I}\left\lfloor\frac{\ln y_i}{\ln u}\right\rfloor,\quad u=e^{\sigma\sqrt{\tau}}$$

in the right-hand side of the above and then transform it identically to get

$$
\begin{aligned}
&\frac{\lceil(h_0+(N-I)-L)/2\rceil-1-(N-I)p}{\sqrt{(N-I)p(1-p)}}\\
&=\frac{\ln(K/S_0)-\sum_{i=1}^I\ln y_i-\sigma\sqrt{\tau}(N-I)(2p-1)+2\Theta\sigma\sqrt{\tau}}{2\sigma\sqrt{\tau}\sqrt{(N-I)p(1-p)}}
\end{aligned}
$$

(here, Θ is a real number in the interval $(0,I)$)

$$
\begin{aligned}
&\to\left(\ln(K/S_0)-\sum_{i=1}^I\ln y_i-\tau(N-I)(r-\lambda(\mu-1)-\sigma^2/2)\right.\\
&\qquad\left.+2\Theta\sigma\sqrt{\tau}\right)\left(2\sigma\sqrt{\tau}\sqrt{(N-I)p(1-p)}\right)^{-1}
\end{aligned}
$$

$\Big($as the Taylor's series expansion on $2p-1$ in series of $\sqrt{\tau}$ is $2p-1=\sqrt{\tau}(r-\lambda(\mu-1)-\sigma^2/2)/\sigma+o(\tau)\Big)$

$$
\begin{aligned}
&=\left(\ln(K/S_0)-\sum_{i=1}^I\ln y_i-T((N-I)/N)(r-\lambda(\mu-1)-\sigma^2/2)\right.\\
&\qquad\left.+2\Theta\sigma\sqrt{\tau}\right)\left(2\sigma\sqrt{T}\sqrt{((N-I)/N)p(1-p)}\right)^{-1}\\
&\to\frac{\ln(K/S_0)-\sum_{i=1}^I\ln y_i-T(r-\lambda(\mu-1)-\sigma^2/2)}{\sigma\sqrt{T}}.
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
&1-\psi(N,I,p)\\
&\quad\to\Phi\left(\frac{\ln(K/S_0)-\sum_{i=1}^I\ln y_i-T(r-\lambda(\mu-1)-\sigma^2/2)}{\sigma\sqrt{T}}\right).
\end{aligned}
$$

Consequently,

$$
\begin{aligned}
&\psi(N,I,p)\\
&\quad\to\Phi\left(-\frac{\ln(K/S_0)-\sum_{i=1}^I\ln y_i-T(r-\lambda(\mu-1)-\sigma^2/2)}{\sigma\sqrt{T}}\right)\\
&\quad=\Psi(I,-\sigma^2/2).
\end{aligned}
$$

By noticing that the series

$$\psi(N,I,p)=\sum_{U=\lceil(h_0+N-I-L)/2\rceil}^{N-I}\binom{N-I}{U}p^U(1-p)^{N-I-U}$$

is bounded by 1, we know the dominated convergence theorem applies. So we have

$$\lim_{N\to\infty}\mathbb{E}[\psi(N,I,p)]=\mathbb{E}\left[\lim_{N\to\infty}\psi(N,I,p)\right]=\mathbb{E}[\Psi(I,-\sigma^2/2)].$$

(ii) *Coefficient of* S_0: We want to show that

$$e^{-rT}\tilde{r}^{N-I}\mathbb{E}^L[u^L\psi(N,I,\tilde{p})]\xrightarrow{N\to\infty}e^{-\lambda(\mu-1)T}\mathbb{E}^Y\left[\prod_{i=1}^I y_i\Psi(I,\sigma^2/2)\right].$$

One the one hand,

$$u^L=\left(\prod_{i=1}^I y_i\right)\frac{1}{u^\Theta}\longrightarrow\prod_{i=1}^I y_i,$$

because

$$L=\sum_{i=1}^I\left\lfloor\frac{\ln y_i}{\ln u}\right\rfloor=\sum_{i=1}^I\frac{\ln y_i}{\ln u}-\Theta,\quad\text{and}\quad u\to 1.$$

And similar to the result for p, we have the limit

$$\psi(N,I,\tilde{p})\longrightarrow\Psi(I,\sigma^2/2),$$

since

$$2\tilde{p}-1=\sqrt{\tau}\frac{r-\lambda(\mu-1)+\sigma^2/2}{\sigma}+o(\tau).$$

Analogous to (i), by applying the dominated convergence theorem,

$$\mathbb{E}^L[u^L\psi(N,I,\tilde{p})]\xrightarrow{N\to\infty}\mathbb{E}^Y\left[\prod_{i=1}^{I}y_i\Psi(I,\sigma^2/2)\right].$$

On the other hand, by taking a Taylor series expansion in τ, it is easy to see that

$$\mathrm{e}^{-rT}\tilde{r}^{N-I}=\mathrm{e}^{-rT}\left(\frac{\mathrm{e}^{r\tau}-\lambda\tau\mu}{1-\lambda\tau}\right)^{N-I}\longrightarrow\mathrm{e}^{-\lambda(\mu-1)T},$$

establishing the convergence of (ii).

Combining (i) and (ii), we conclude that $V_N(I)\to_{N\to\infty}V_c(I)$. $\square$

Lemma 5. *$V_N(I)$ and $V_c(I)$ are bounded functions for fixed I. More specifically, there exists $n\in\mathbb{Z}^+$ such that for all $N>n$, there exists $A>0$ such that*

$$|V_N(I)|<A\mu^I,\quad |V_c(I)|<A\mu^I.$$

Proof. (i) *Boundedness of $V_c(I)$*: Because of inequalities $0\le\Psi(I,\sigma^2/2)\le 1$ and $0\le\Psi(I,-\sigma^2/2)\le1$, we have

$$0\le\mathbb{E}^Y[\Psi(I,-\sigma^2/2)]\le1,$$

$$0\le\mathbb{E}^Y\left[\prod_{i=1}^{I}y_i\Psi(I,\sigma^2/2)\right]\le\mathbb{E}^Y\left[\prod_{i=1}^{I}y_i\right]$$

(noticing that y_i is positive).

Therefore,

$$\begin{aligned}
&|V_c(I)|\\
&=\left|S_0\mathrm{e}^{-\lambda(\mu-1)T}\mathbb{E}^Y\left[\prod_{i=1}^{I}y_i\Psi(I,\sigma^2/2)\right]-\mathrm{e}^{-rT}K\mathbb{E}^Y[\Psi(I,-\sigma^2/2)]\right|\\
&\le\left|S_0\mathrm{e}^{-\lambda(\mu-1)T}\mathbb{E}^Y\left[\prod_{i=1}^{I}y_i\Psi(I,\sigma^2/2)\right]\right|\le\left|S_0\mathrm{e}^{-\lambda(\mu-1)T}\mathbb{E}^Y\left[\prod_{i=1}^{I}y_i\right]\right|\\
&=\left|S_0\mathrm{e}^{-\lambda(\mu-1)T}\prod_{i=1}^{I}\mathbb{E}^Y[y_i]\right|\quad(\text{as } Y_i'\text{s are independent})\\
&=S_0\mathrm{e}^{-\lambda(\mu-1)T}\mu^I.
\end{aligned}$$

(ii) *Boundedness of $V_N(I)$*: As a result of $0\le\psi(N,I,p)\le1$, $0\le\psi(N,I,\tilde{p})\le1$, we have

$$0\le\mathbb{E}^L[\psi(N,I,p)]\le1,$$

$$0\le\mathbb{E}^L\Big[u^L\psi(N,I,\tilde{p})]\le\mathbb{E}^L[u^L].$$

Therefore,

$$\begin{aligned}
|V_N(I)|&=|S_0\mathrm{e}^{-rT}\tilde{r}^{N-I}\mathbb{E}^L[u^L\psi(N,I,\tilde{p})]-\mathrm{e}^{-rT}K\mathbb{E}^L[\psi(N,I,p)]|\\
&\le|S_0\mathrm{e}^{-rT}\tilde{r}^{N-I}\mathbb{E}^L[u^L\psi(N,I,\tilde{p})]|\\
&\le|S_0\mathrm{e}^{-rT}\tilde{r}^{N-I}\mathbb{E}^L[u^L]|\\
&\le|S_0(\mathrm{e}^{-\lambda(\mu-1)T}+\theta)(\mu^I+\theta')|.
\end{aligned}$$

(from the proof of Lemma 4, we know that $u^L\to\prod_{i=1}^{I}y_i$, $\mathrm{e}^{-rT}\tilde{r}^{N-I}\to\mathrm{e}^{-\lambda(\mu-1)T}$)

Hence there exists $n\in\mathbb{Z}^+$ such that when $N>n$, there exists $A>0$ such that

$$|V_N(I)|<A\mu^I,\quad |V_c(I)|<A\mu^I.\quad\square$$

Lemma 6. *For any $\varepsilon>0$, there exists $N_0\in\mathbb{Z}^+$ such that for all $N>N_0$,*

$$\sum_{I=N_0}^{N}V_N(I)q_N(I)<\varepsilon.$$

Proof. Pick an $\varepsilon>0$ arbitrary and fixed.

First, from Lemma 5, there exists $N_1\in\mathbb{Z}^+$ and $A\in\mathbb{R}^+$ such that for any $N>N_1$,

$$V_N(I)<A\mu^I.$$

Second, as $qN=\lambda T$ is constant, we have $q_N(I)\longrightarrow q(I)$, so there exists $N_2\in\mathbb{Z}^+$ such that for any $N>N_2$,

$$q_N(I)<2q(I)=2\mathrm{e}^{-\lambda T}\frac{(\lambda T)^I}{I!}.$$

Next, from the Taylor's series expansion of e^x, we know the series $\sum_{I=0}^{\infty}((\lambda T\mu)^I/I!)$ is convergent. Hence for $\varepsilon>0$, there exist $N_3\in\mathbb{Z}^+$ such that for any $N'>N_3$,

$$\sum_{I=N'}^{N}\frac{(\lambda T\mu)^I}{I!}<\sum_{I=N'}^{\infty}\frac{(\lambda T\mu)^I}{I!}<\frac{\varepsilon}{2A\mathrm{e}^{-\lambda T}}.$$

Therefore, for $\varepsilon>0$, there exists $N_0=\max\{N_1,N_2,N_3\}$ such that for $N>N_0$,

$$\begin{aligned}
\sum_{I=N_0+1}^{N}V_N(I)q_N(I)&<\sum_{I=N_0+1}^{N}A\mu^I\cdot2\mathrm{e}^{-\lambda T}\frac{(\lambda T)^I}{I!}\\
&=2A\mathrm{e}^{-\lambda T}\sum_{I=N_0+1}^{N}\frac{(\lambda T\mu)^I}{I!}<\varepsilon.\quad\square
\end{aligned}$$

Lemma 7. $\mathbb{E}_{\mathscr{B}}[V_c(\xi_N)]\to_{N\to\infty}\mathbb{E}_{\mathbf{\Pi}}[V_c(\xi)]$.

Proof. First, we know that $q_N(I)\to q(I)$, as $qN=\lambda T$ is constant; thus we have $\xi_N\to_d\xi$.

And $V_c(I)$ is a function whose domain is $\mathbb{N}$. We can expand its domain to $\mathbb{R}^*$ just by naturally adjoining the points $(I,V_c(I))$ and $(I+1,V_c(I+1))$ on the graph of $V_c(I)$. After this intuitive operation, we obtain a continuous function $V_c(x)$. Hence for a continuous function $V_c(x)$,

$$\mathbb{E}_{\mathscr{B}}[V_c(\xi_N)]\xrightarrow{N\to\infty}\mathbb{E}_{\mathbf{\Pi}}[V_c(\xi)].\quad\square$$

C.2. Proof of Theorem 2

Proof. Pick $\varepsilon>0$ arbitrary but fixed.

From Lemma 7, $\mathbb{E}[V_c(\xi_N)]\to_{N\to\infty}\mathbb{E}[V_c(\xi)]$, so there exists $n_1\in\mathbb{Z}^+$ such that for any $N>n_1$,

$$|\mathbb{E}[V_c(\xi_N)-V_c(\xi)]|<\frac{\varepsilon}{3}.$$

From Lemma 6, there exists $n_2\in\mathbb{Z}^+$ such that for any $N'>n_2$,

$$\left|\sum_{I=N'}^{N}V_N(I)q_N(I)\right|<\frac{\varepsilon}{3}.$$

And from Lemma 4, $V_N(I) \to_{N\to\infty} V_c(I)$ for fixed $I \in \mathbb{N}$, so there exists $N(I) \in \mathbb{N}$ such that for all $N \ge N(I)$,

$$|V_N(I) - V_c(I)| < \frac{\varepsilon}{3}.$$

Set $n_3 = \max_{0\le I\le n_2}\{N(I)\}$; then for $N > n_3$,

$$\sum_{I=0}^{n_2} |V_N(I) - V_c(I)| q_N(I) < \frac{\varepsilon}{3}\sum_{I=0}^{n_2} q_N(I) < \frac{\varepsilon}{3}.$$

Let $n = \max\{n_1, n_2, n_3\}$; then for $N > n$,

$$\begin{aligned}
&|\mathbb{E}[V_N(\xi_N) - V_c(\xi)]| \\
&\quad \le |\mathbb{E}[V_N(\xi_N) - V_c(\xi_N)]| + |\mathbb{E}[V_c(\xi_N) - V_c(\xi)]| \\
&\quad = \left|\sum_{I=0}^{N}[V_N(I) - V_c(I)]q_N(I)\right| + |\mathbb{E}[V_c(\xi_N) - V_c(\xi)]| \\
&\quad \le \left|\sum_{I=0}^{n_2}[V_N(I) - V_c(I)]q_N(I)\right| + \left|\sum_{I=n_2+1}^{N}[V_N(I) - V_c(I)]q_N(I)\right| \\
&\qquad + |\mathbb{E}[V_c(\xi_N) - V_c(\xi)]| \\
&\quad \le \sum_{I=0}^{n_2}|V_N(I) - V_c(I)|q_N(I) + \left|\sum_{I=n_2+1}^{N} V_N(I)q_N(I)\right| \\
&\qquad + |\mathbb{E}[V_c(\xi_N) - V_c(\xi)]| < \frac{\varepsilon}{3} + \frac{\varepsilon}{3} + \frac{\varepsilon}{3} = \varepsilon.
\end{aligned}$$

Therefore, $\mathbb{E}_{\mathscr{B}}[V_N(\xi_N)] \to_{N\to\infty} \mathbb{E}_{\Pi}[V_c(\xi)]$, $V_N \to_{N\to\infty} V_c$. $\square$

Appendix D. Computational Implementation

Implementation of the pricing formulas requires specifying the truncation parameter I_0 on the maximum number of jumps in a lattice path. In the latter case, if the original jump-size distribution is continuous or discrete with infinite support, then an approximation procedure is required. Furthermore, the size of the lattice must also be bounded, which is controlled by two additional parameters, denoted by M_0 and R. This appendix discusses these two issues and offers various implementation suggestions that were used in the numerical experiments reported in the paper.

D.1. Approximation of Jump-Size Distribution

The algorithm requires a finite discrete jump-size distribution. Here, we describe methods for discretizing a continuous distribution and for truncating an infinite discrete distribution.

(i) *Discretizing a continuous distribution*

Partition the range for the jump size into a set of small intervals to obtain the discrete distribution

$$\mathbb{P}(Y = u^j) = \mathbb{P}(J = j) = F\Big(\big(j+\tfrac{1}{2}\big)\ln u\Big) - F\Big(\big(j-\tfrac{1}{2}\big)\ln u\Big).$$

Without loss of generality, we write the discrete distribution as $\{\varrho_j\}_{j=-\infty}^{+\infty}$, where the value of ϱ_j could be 0 if the distribution is finite. To be consistent with the model setting, we require the probability of a small jump (± 1) to be 0 and move the original mass at $\varrho_{\pm 1}$ to ϱ_0 to get the distribution $\mathscr{Y}$: $\{\rho_j\}_{j=-\infty}^{\infty}$, where $\rho_{\pm 1} = 0$, $\rho_0 = \varrho_0 + \varrho_1 + \varrho_{-1}$, $\rho_j = \varrho_j$, $\forall\, j \ne 0, \pm 1$.

(ii) *Truncating an infinite distribution*

Approximate the left tail and/or right tail of the distribution $\mathscr{Y}$: $\{\rho_j\}_{j=-\infty}^{+\infty}$ using a single bucket as follows:

$$\tilde{\mathscr{Y}}: \tilde{\rho}_j = \begin{cases} \sum_{j=-\infty}^{\lfloor \alpha/\ln u\rfloor - M_0 - 1}\rho_j & j = M^- \doteq \lfloor \alpha/\ln u\rfloor - M_0 - R, \\ \rho_j & \lfloor \alpha/\ln u\rfloor - M_0 \le j \\ & \quad \le \lceil \alpha/\ln u\rceil + M_0, \\ \sum_{j=\lceil \alpha/\ln u\rceil + M_0 + 1}^{\infty}\rho_j & j = M^+ \doteq \lceil \alpha/\ln u\rceil + M_0 + R, \\ 0 & \text{otherwise}; \end{cases}$$

where α is the expectation of log jump for the original jump-size distribution $\mathscr{Y}$ or $\mathscr{Y}_c$. The truncation parameter M_0 is chosen to guarantee that the shape of $\tilde{\mathscr{Y}}$ is close enough to the original jump-size distribution $\mathscr{Y}$ or $\mathscr{Y}_c$, and the integer $R > 0$ minimizes $|\tilde{\delta}^2 - \delta^2|$ such that the variance $\tilde{\delta}^2$ of $\tilde{\mathscr{Y}}$ is close enough to δ^2, the variance of $\mathscr{Y}$ or $\mathscr{Y}_c$.

As an example, for a normal distribution $\mathscr{N}(\alpha, \delta^2)$, we use the 3δ-principle to derive

$$M_0 = \left\lceil \frac{3\delta}{\ln u} \right\rceil,$$

where the chance of jump size greater than 3δ under a normal distribution case is less than 10^{-3}. For the double-exponential distribution, we can also calculate the corresponding M_0 for each N.

D.2. Determination of the Maximum Number of Jumps I_0

From Lemma 6, for an appropriate I_0, we have

$$\sum_{I=I_0}^{N} V_N(I) q_N(I) \longrightarrow 0;$$

i.e., lattice paths that have more than I_0 jumps have negligible impact on the option value. Thus we turn to the determination of I_0.

For the Poisson process, we have that $qN = \lambda T$ is a constant, and $\mathscr{B}_\tau(N, q) \to \mathbf{\Pi}(\lambda T)$. Therefore, we choose the maximum number of jumps I_0 by bounding the tail probability of the Poisson distribution $\mathbf{\Pi}(\lambda T)$; i.e., I_0 is the smallest integer satisfying

$$\sum_{i=I_0+1}^{\infty} e^{-\lambda T}\frac{(\lambda T)^i}{i} < \epsilon,$$

which leads to (16). Note that by (16), the value of the maximum number of jumps I_0 is determined by λ and T only; i.e., it is independent of N.

We show an example under the jump-diffusion process described in Section 4.1 for a particular time step N to see the computational implementation.

1. Taking $\epsilon = 10^{-5}$ in (16) gives $I_0 = 12$.

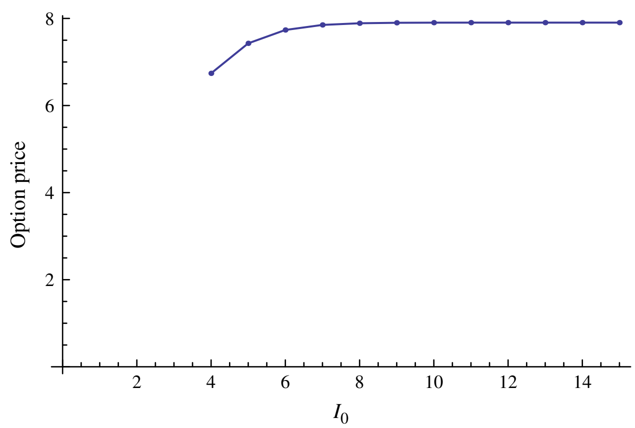
2. For each time step N, do the following operations. Here, we illustrate for $N = 2{,}000$.

3. By definition of τ and relationship (7), calculate $u = e^{\sigma\sqrt{\tau}} = 1.00354$.

Table D.1 **European Put Option Prices with Different I_0 Under Log-Normal Jump-Size Distribution** $\ln \mathcal{N}(-0.025, 0.05)$ $\sigma^2 = 0.05$, $S_0 = 40$, $r = 0.08$, $T = 0.5$, $\lambda = 5$, **and** $K = 45$ **(Time Steps** $N = 2{,}000$**)**

I_0	4	5	6	7	8	9
Option price	6.7392	7.4283	7.7351	7.8511	7.8892	7.9002
I_0	10	11	12	13	14	15
Option price	7.9031	7.9038	7.9039	7.9039	7.9040	7.9040

Figure D.1 **(Color online) European Put Option Prices with a Different Maximum Number of Jumps, Under Jump-Size Distribution** $\ln \mathcal{N}(-0.025, 0.05)$**, and Parameters** $\sigma^2 = 0.05$, $S_0 = 40$, $r = 0.08$, $T = 0.5$, $\lambda = 5$, **and** $K = 45$ **(Time Steps** $N = 2{,}000$**)**



4. Following Appendix D.1 gives $M_0 = 190$, $R = 19$, and $-8 < \alpha/\ln u < -7$; then the corresponding discrete jump-size distribution is written as follows:

$$\tilde{\mathcal{Y}}\colon\ \tilde{\rho}_j = \begin{cases} F\big((-198-\frac{1}{2})\ln u\big) & j = -217,\\ F\big((j+\frac{1}{2})\ln u\big) - F\big((j-\frac{1}{2})\ln u\big) & -198 \le j < -1,\\ F\big((1+\frac{1}{2})\ln u\big) - F\big((-1-\frac{1}{2})\ln u\big) & j = 0,\\ F\big((j+\frac{1}{2})\ln u\big) - F\big((j-\frac{1}{2})\ln u\big) & 1 < j \le 183,\\ 1 - F\big((183+\frac{1}{2})\ln u\big) & j = 202,\\ 0 & \text{otherwise.} \end{cases}$$

5. Compute using the main pricing formulas after calculation of $q = 0.00125$ and $p = 0.501948$ by Equation (7).

To see the effect of a different maximum number of jumps on the option price, we show the resulting option price with different I_0 under this set of parameters in Table D.1 and Figure D.1.

It can be seen from Table D.1 and Figure D.1 that the option price converges within 10^{-4} precision for $I_0 = 12$.

Appendix E. Pseudocode

In MATLAB

```
h1 = min(I0 * (-M + 1) - N, -N);
  h2 = max(I0 * (M - 1) + N, N);  tm = (h2 - h1 + 2)/2;
I = [I: I0];  G = zeros(I0 + 1, h2 - h1 + 1);
for i = 1: length(I)
    U{I} = {0: N - I(i)};
end
Q = ComBinN(N, I, q);
for i = 1: length(I)
    temp = ComBinN(N - I(i), cell2mat(U{i}), p);
    temp0 = zeros(1, length(temp));
    P = [temp; temp0];
    P = reshape(P, [1, numel(P)]);
    P(end) = [];
    if I(i) == 0
        G(1, tm - N: tm + N) = P;
        y = Y;
    else
        G(I(i) + 1, tm - (M - 1) * I(i) - N:
            tm + (M - 1) * I(i) + N) = conv(y, P);
        y = conv(y, Y);
    end
end
PX = Q * G;
su = exp((h^call : 1 : h2).
        * (log(u) * ones(1, h2 - h^call + 1)));
V^call = exp(-r * T).
        * (PX(1, h^call - h1 + 1: h2 - h1 + 1) * (S0. * su - K)');
```

In Mathematica

```
J[z_] := Sum[ρ[l]z^l, {l, -M0 - R, M0 + R}];
qni = Table[Binomial[n, i]q^i (1 - q)^(n-i), {i, 0, I0}];
pnu[i_] := Table[Binomial[n - i, u]p^u (1 - p)^(n-i-u),
                {u, 0, n - i}];
dg = Table[Total[pnu[i] * Table[z^(2u+i-n), {u, 0, n - i}]],
           {i, 0, I0}];
jdg = Table[Expand[(J[z])^i dg[[i + 1]]], {i, 0, I0}];
pnh = Table[CoefficientList[Expand[jdg[[i + 1]]
            · z^(i(M0+R-1)+n)], z], {i, 0, I0}];
vni = Table[Exp[-rT]Sum[
        pnh[[i + 1]][[k + 1]](K - S0Exp[Δ(k - i(M0 + R - 1)
        -n)]), {k, 0, h^put + i(M0 + R - 1) + n}], {i, 0, I0}];
vput = Sum[vni[[i + 1]]qni[[i + 1]], {i, 0, I0}]
```

References

Amin KI (1993) Jump diffusion option valuation in discrete time. *J. Finance* 48(5):1833–1863.

Asmussen S, Avram F, Pistorius MR (2004) Russian and American put options under exponential phase-type Lévy models. *Stochastic Processes Appl.* 109(1):79–111.

Ball CA, Torous WN (1983) A simplified jump process for common stock returns. *J. Financial Quant. Anal.* 18(1):53–65.

Bates DS (1991) The crash of '87: Was it expected? The evidence from options markets. *J. Finance* 46(3):1009–1044.

Bates DS (2003) Empirical option pricing: A retrospection. *Econometrics* 116(1):387–404.

Benninga S, Wiener Z (1998) Binomial term structure models. *Mathematica Ed. Res.* 7(3):1–10.

Black F, Scholes M (1973) The pricing of options and corporate liabilities. *J. Political Econom.* 81(3):637–654.

Bollerslev T, Todorov V (2011) Estimation of jump tails. *Econometrica* 79(6):1727–1783.

Bookstaber RM, McDonald JB (1987) A general distribution for describing security price returns. *J. Bus.* 60(3):401–424.

Broadie M, Detemple J (2004) Option pricing: Valuation models and applications. *Management Sci.* 50(9):1145–1177.

Cai N (2009) On first passage times of a hyper-exponential jump-diffusion process. *Oper. Res. Lett.* 37(2):127–134.

Cai N, Kou S (2011) Option pricing under a mixed-exponential jump diffusion model. *Management Sci.* 57(11):2067–2081.

Cai N, Kou SG, Liu Z (2014) A two-sided Laplace inversion algorithm with computable error bounds and its applications in financial engineering. *Adv. Appl. Probab.* 46:766–789.

Carr P, Madan D (1999) Option pricing using the fast Fourier transform. *J. Comput. Finance* 2(4):61–73.

Cont R, Tankov P (2004) Nonparametric calibration of jump-diffusion option pricing models. *J. Computational Finance* 7(3):1–49.

Cox J, Ross S, Rubinstein M (1979) Option pricing: A simplified approach. *J. Financial Econom.* 7(3):229–263.

Eraker B, Johannes M, Polson N (2003) The impact of jumps in volatility and returns. *J. Finance* 58(3):1269–1300.

Feng L, Lin X (2013) Pricing Bermudan options in Lévy process models. *SIAM J. Finan. Math.* 4(1):474–493.

Feng L, Linetsky V (2009) Computing exponential moments of the discrete maximum of a Lévy process and lookback options. *Finance Stochastics* 13(4):501–529.

Garcia R, Ghysels E, Renault E (2010) The econometrics of option pricing. Aït-Sahalia Y, Hansen LP, eds. *Handbook of Financial Econometrics*, Vol. 1 (Elsevier Science, Amsterdam), 479–552.

Harrison JM, Kreps DM (1979) Martingales and arbitrage in multiperiod securities markets. *J. Econom. Theory* 20(3):381–408.

Harrison JM, Pliska SR (1981) Martingales and stochastic integrals in the theory of continuous trading. *Stochastic Processes Their Appl.* 11(3):215–260.

Humpage OF (1997) Recent U.S. intervention: Is less more? *Econom. Rev. (Cleveland)* 33(3):2–10.

Jarrow RA, Rosenfeld ER (1984) Jump risks and the intertemporal capital asset pricing model. *J. Bus.* 57(3):337–351.

Kaeck A (2013) Asymmetry in the jump-size distribution of the S&P 500: Evidence from equity and option markets. *J. Econom. Dynam. Control* 37(9):1872–1888.

Kou SG (2002) A jump-diffusion model for option pricing. *Management Sci.* 48(8):1086–1101.

Kou SG (2008a) Jump-diffusion models for asset pricing in financial engineering. Birge JR, Linetsky V, eds. *Financial Engineering*, Handbooks in Operations Research and Management Science, Vol. 15, Chap. 2 (Elsevier, Amsterdam), 73–116.

Kou SG (2008b) Discrete barrier and lookback options. Birge JR, Linetsky V, eds. *Financial Engineering*, Handbooks in Operations Research and Management Science, Vol. 15, Chap. 8 (Elsevier, Amsterdam), 343–373.

Kou SG, Wang H (2004) Option pricing under a double exponential jump diffusion model. *Management Sci.* 50(9):1178–1192.

Laprise SB, Fu MC, Marcus SI, Lim AE, Zhang H (2006) Pricing American-style derivatives with European call options. *Management Sci.* 52(1):95–110.

Li BQ, Zhao HJ (2009) Pricing Parisian options by generating functions. *J. Derivatives* 16(4):72–81.

Madan D, Seneta E (1990) The variance gamma model of share market returns. *J. Bus.* 63(4):511–525.

Merton RC (1976) Option pricing when underlying stock returns are discontinuous. *J. Financial Econom.* 3(1–2):125–144.

Rachev ST, Höchstötter M, Fabozzi FJ, Focardi SM (2010) Discrete probabilities distributions. *Probability and Statistics for Finance*, Chap. 9 (John Wiley & Sons, Hoboken, NJ).

Ramezani CA, Zeng Y (1998) Maximum likelihood estimation of asymmetric jump-diffusion processes: Application to security prices. Available at SSRN: http://ssrn.com/abstract=606361.

Samuelson PA (1973) Mathematics of speculative price. *SIAM Rev.* 15(1):1–42.

Santos A, Guerra J (2015) Implied risk neutral densities from option prices: Hypergeometric, spline, lognormal, and Edgeworth functions. *J. Futures Markets* 35(7):655–678.

Schoutens W (2003) *Lévy Processes in Finance: Pricing Financial Derivatives* (John Wiley & Sons, Chichester, UK).

Tucker AL (1992) A reexamination of finite-variance and infinite-variance distributions as models of daily stock returns. *J. Bus. Econom. Statist.* 10(1):73–81.