A High-order Front-tracking Finite Difference Method for Pricing American Options under Jump-Diffusion Models*

Jari Toivanen†

Abstract

A free boundary formulation is considered for the price of American options under jump-diffusion models with finite jump activity. On the free boundary a Cauchy boundary condition holds due to the smooth pasting principle. An implicit finite difference discretization is performed on time-dependent nonuniform grids. During time stepping solutions are interpolated from one grid to another using Lagrange interpolations. Finite difference stencils are also constructed using Lagrange interpolation polynomials based on either three or five grid points. With these choices second-order and fourth-order convergence with respect to the number of time and space steps can be expected. In numerical examples these convergence rates are observed under the Black-Scholes model and Kou's jump-diffusion model.

Keywords: American option, jump-diffusion model, free boundary problem, front-tracking method, finite difference method, moving grid, Lagrange interpolation

1 Introduction

In this paper, we consider numerical method for pricing American options based on partial differential equations (PDEs) and partial integro-differential equations (PIDEs) arising from diffusion and jump-diffusion models, respectively, for the underlying asset. Already a few years after the seminal papers by Black and Scholes [4] and Merton [26], a jump-diffusion model with a log-normal jump distribution was introduced by Merton in [27]. Particularly, jumps can have large impact on the price of an option near expiry. More recently a model with a log-double-exponential jump distribution was described by Kou in [22]. Both Merton's model and Kou's model

*This research was supported by the Academy of Finland, grant #207089

†Department of Mathematical Information Technology, University of Jyväskylä, PO Box 35 (Agora), FI-40014 University of Jyväskylä, Finland, Jari.Toivanen@mit.jyu.fi

have finite jump activity which is the class of models considered in this paper. A more generic CGMY model [7] can have infinite jump activity. Another approach is to consider stochastic volatility models like Heston's model [16]. Recent studies suggest that better match with the market behavior is obtained by adding jumps to a stochastic volatility model; see [13], for example. We will not consider stochastic volatility models in here.

When pricing an American option we also need to determine the optimal exercise moment as a function of the value of the underlying asset. This leads to a nonlinear problem for the price of the American option. The most traditional way to price these options in the P(I)DE context is to formulate a linear complementarity problem (LCP) or a variational inequality for the price. Already in 1977, Brennan and Schwartz in [5] considered a finite difference method for this type of formulation. More extensively these approaches have been considered in [18] and [21]. During the last decade several approximations for LCPs have been proposed. These include a penalty method [14], an operator splitting method [19], and a Lagrange multiplier method [20]. Usually second-order accurate finite differences are use to discretize the partial differential operators when pricing American options. In [25], [29], [33], fourth-order accurate finite differences were considered for American options. The early exercise boundary requires special treatment when more than second-order accuracy is sought. In [25] correction terms are added near the exercise boundary to increase the accuracy. Time-dependent change of variable which concentrates grid points near the exercise boundary was used in [29], [33].

In this paper, we consider an alternative approach to price American options using a free boundary formulation. The earlier papers have considered this approach only for the Black-Scholes model. The smooth pasting principle says that the first space derivative of the option price is continuous across the early exercise boundary which is the free boundary. This gives us an additional condition which allows us to find the location of the free boundary. As this boundary moves in time the computational domain changes also over time. Methods keeping track of the boundary and discretizing the problem in changing domain are called front-tracking methods. This approach has been considered in [15], [30], [32]. An alternative approach is to employ a time-dependent change of variable to map the changing domain into a fixed domain. Such methods are called front-fixing methods and they have been considered in [17], [28], [32], [36], [38]. Most of these methods use spatially second-order accurate finite differences. In [17], finite element method was used, and in [32], fourth-order accurate finite differences were employed. The implicit discretization lead to a nonlinear system of equations at each time step. Usually these systems have been solved using Newton's method. In [36], an explicit update for the location of free boundary at each time step was used, and in [38], the secant method was employed to solve the nonlinear problems. In [32], [38], the difference of the prices of American option and European option was computed. This is called singularity-separation as the computed quantity is more smooth than the price of the American option.

Jump-diffusion models lead to partial integro-differential operators which are nonlocal due to the integral part. Their discretization yields full matrices which makes many methods computationally too expensive. Zhang [37] considered pricing American options under Merton's model using a variational inequality formulation and finite differences. Since then several papers and books including [2], [3], [8], [9], [10], [24], [34], [35] have considered numerical method for jump-diffusion models based on LCP and variational inequality formulations together with finite difference/element discretizations. One of the the main aims of these studies has been to improve computational efficiency by using second-order accurate discretizations and faster ways to handle the integral operator.

In this paper, we derive a numerical method based on the free boundary formulation for pricing American options under under jump-diffusion models with finite jump activity. For simplicity, we consider only American put options in here. Similar methods can be easily derived also for American call options when the underlying asset is paying continuously dividends. Our front-tracking method performs an implicit finite difference discretization on time-dependent nonuniform grids which are refined near the expiry and free boundary. For interpolations between grids and the construction of finite difference stencils Lagrange interpolation polynomials are used. This gives easy way to implement also fourth-order accurate discretizations. At each time steps a nonlinear system of equations is solved by Brent's root finding method [6] which is easy to use and efficient. Numerical experiments demonstrates the behavior of the proposed method.

The outline of the paper is as follows. We begin by describing the partial integro-differential model and LCP for an American put option in Section 2. Our free boundary formulation is given in Section 3. The nonuniform space-time grids are constructed in Section 4. We describe Lagrange interpolation polynomials and their use to construct discretizations for partial derivatives in Sections 5 and 6, respectively. Section 7 explains how the nonlinear systems are solved using Brent's method. We present numerical results in Section 8 and our conclusions are given in the last section.

2 Partial integro-differential model for American put option

Under a jump-diffusion model the price v of a European option satisfies a partial integro-differential equation (PIDE)

$$v_t = -\frac{1}{2}(\sigma x)^2 v_{xx} - (r - \lambda\zeta)xv_x + (r+\lambda)v - \lambda \int_{\mathbb{R}_+} v(t, xy)f(y)\,dy \tag{1}$$

for all $(t, x) \in [0, T) \times \mathbb{R}_+$, where t is the time, x is the value of the underlying asset, T is the expiry date; see [8], for example. The volatility is σ and the risk free interest rate is r. The volatility can be a function t and x as in [3], [12], [35], for example. This gives more flexibility to calibrate the model to be consistent with the market prices.

We assume the jump process to have a finite activity and that its size distribution is given by the function $f(y)$. The jump intensity is a compound Poisson process with the rate λ. The constant ζ in (1) is given by $\zeta = \int_{\mathbb{R}_+}(y-1)f(y)\,dy$. Examples of this kind of jump-diffusion models are Merton's model [27] and Kou's model [22]. When the jump rate λ is zero the PIDE reduces to the standard Black-Scholes PDE [4], [26].

As usual we reverse the direction of the time integration by making the change of variable $t = T - \tau$. Then for the price v, we have

$$v_\tau = Lv \tag{2}$$

for all $(\tau, x) \in (0, T] \times \mathbb{R}_+$, where the partial integro-differential operator L is defined as

$$Lv = \frac{1}{2}(\sigma x)^2 v_{xx} + (r - \lambda\zeta)x v_x - (r + \lambda)v + \lambda \int_{\mathbb{R}_+} v(t, xy) f(y)\,dy. \tag{3}$$

The payoff function of a put option with the strike price K is

$$g(x) = \max\{K - x, 0\}. \tag{4}$$

The payoff function gives the initial value of v, that is,

$$v(0, x) = g(x), \qquad x \in \mathbb{R}_+. \tag{5}$$

An American option can be exercise any time before its expiry in which case its holder gets payoff $g(x)$. To avoid arbitrage opportunities, the American option prices has to satisfy an early exercise constraint $v(\tau, x) \geq g(x)$ for all $(\tau, x) \in (0, T] \times \mathbb{R}_+$. The value v of an American option satisfies a linear complementarity problem (LCP)

$$\begin{cases} (v_\tau - Lv) \geq 0, \quad v \geq g, \\ (v_\tau - Lv)\big(v - g\big) = 0 \end{cases} \tag{6}$$

with the initial value (5). The behavior of the value of the American put option on the boundaries is given by

$$v(\tau, 0) = K \qquad \text{and} \qquad \lim_{x \to \infty} v(\tau, x) = 0, \qquad \tau \in [0, T]. \tag{7}$$

We truncate the infinite domain at $x = X$ with X being sufficiently large and pose a boundary condition

$$v(\tau, X) = 0. \tag{8}$$

The error in the price v caused by this truncation is considered in [9], [24], for example.

3 Free boundary formulation

In the case of an American put option, at any time $\tau \in [0, T]$ the domain $[0, X]$ can be divided into two regions: An early exercise region $[0, X_f(\tau)]$ in where it holds $v(\tau, x) = g(x)$ and it is optimal to exercise the option. A hold region $[X_f(\tau), X]$ in where it holds $v_\tau(\tau, x) = Lv(\tau, x)$ and it is optimal not to exercise the option. The time-dependent free (early exercise) boundary $x = X_f(\tau)$ is not known a-priori and it has to be solved together with the option price v. For a put option, it is known that $X_f(0) = K$ and $X_f(\tau) \in (0, K)$ for $\tau \in (0, T]$. The price v is continuous and satisfies the equations

$$\begin{aligned} v(T, x) &= g(x) & & x \in [0, X], \\ v(\tau, x) &= g(x) & & (\tau, x) \in (0, T] \times [0, X_f(\tau)], \\ v_\tau(\tau, x) &= Lv(\tau, x) & & (\tau, x) \in (0, T] \times [X_f(\tau), X], \\ v(\tau, X) &= 0. \end{aligned} \tag{9}$$

These equations are not sufficient to determine $v(\tau, x)$ and $X_f(x)$ uniquely.

An additional condition is given by the smooth pasting principle which says that the derivative v_x is continuous across the free boundary $x = X_f(\tau)$. This principle holds for finite activity jump processes [31] considered in this paper. For a put option this principle gives a condition

$$v_x(\tau, X_f(\tau)) = g_x(X_f(\tau)) = -1. \tag{10}$$

Thus, at the free boundary $x = X_f(\tau)$ the price v satisfies the Cauchy boundary condition

$$\begin{aligned} v(\tau, X_f(\tau)) &= \max\{K - X_f(\tau), 0\}, \\ v_x(\tau, X_f(\tau)) &= -1. \end{aligned} \tag{11}$$

The price v and the free boundary location X_f satisfy a nonlinear free boundary problem given by the equations

$$\begin{aligned} v(0, x) &= 0 & & x \in [K, X], \\ v_\tau(\tau, x) &= Lv(\tau, x) & & x \in [X_f(\tau), X], \\ v(\tau, X_f(\tau)) &= \max\{K - X_f(\tau), 0\}, \\ v_x(\tau, X_f(\tau)) &= -1, \\ v(\tau, X) &= 0 \end{aligned} \tag{12}$$

for $\tau \in (0, T]$. In the following, we consider the numerical solution of this problem.

4 Discretization grids

The free boundary problem (12) is posed in the domain

$$\{(\tau, x) \mid \tau \in (0, T], \;\; x \in [X_f(\tau), X]\}. \tag{13}$$

The space interval $[X_f(\tau), X]$ changes over time and, thus, a space grid needs to be time-dependent. An alternative approach would be to perform a time-dependent change of variables in such a way that the domain would be fixed after it. We will not consider this front-fixing approach in here.

Our aim is to construct a fourth-order accurate discretization with respect to the length of the largest space and time steps. To accomplish this it is necessary to reduce the step size in a part of the domain where the solution is less regular. The free boundary moves faster and the solution changes more rapidly immediately after the initial time $\tau = 0$. Thus, we make the first time steps shorter and we choose the time grid with m time steps to be

$$\tau_k = \left(\frac{k}{m}\right)^3 T, \quad i = 1, \ldots, m. \tag{14}$$

The length of the first time step is $\mathcal{O}(\Delta\tau^3)$, where $\Delta\tau = \tau_m - \tau_{m-1} = \mathcal{O}\left(\frac{1}{m}\right)$ is the length of the longest time step. This means that we can perform the first time step with the implicit Euler method and we still can reach $\mathcal{O}(\Delta\tau^4)$ accuracy.

The solution v is less regular near the free boundary $x = X_f(\tau)$. Also reducing the space step near this boundary makes the space discretization easier to construct. At the time τ_k we choose the space grid with n grid points to be

$$x_{k,i} = X_f(\tau_k) + (X - X_f(\tau_k))\left(\frac{i-1}{n-1}\right)^3, \quad i = 1, \ldots, n. \tag{15}$$

Similarly to time steps, we have that the length of the first grid step is $\mathcal{O}(\Delta x^3)$, where $\Delta x = \left(\frac{1}{n}\right)$ is the length of the longest grid step. Thus, we can use a second-order accurate differences near the free boundary and still attain fourth-order accuracy with respect to Δx. Figure 1 shows a space-time grid when $K = 100$, $X = 300$, and $T = 0.25$. In the following, we denote the grid point values of v as

$$v_{k,i} = v(\tau_k, x_{k,i}). \tag{16}$$

5 Lagrange interpolation

We use finite difference discretizations for the time derivative v_t. At the time τ_k they require the values $v(\tau_k, x_{k,i})$ and $v(\tau_{k-1}, x_{k,i})$, $i = 1, \ldots, n$. As the space grids move in time the value $v(\tau_{k-1}, x_{k,i})$ for a given i is not generally directly available, that is, $x_{k,i}$ is not usually one of the grid points $x_{k-1,j}$, $j = 1, \ldots, n$. Hence, we need to interpolate the value $v(\tau_{k-1}, x_{k,i})$ using the available grid point values.

Let us consider interpolating the value $v(\tau_{k-1}, x_{k,i})$ using the Lagrange interpolation. We choose $N = j_{\max} - j_{\min} + 1$ grid points $x_{k-1,j_{\min}} < \cdots < x_{k-1,j_{\max}}$ on the grid at the time τ_{k-1} such that $x_{k-1,j_{\min}} \leq x_{k,i} \leq x_{k-1,j_{\max}}$. The Lagrange interpolation polynomial is defined as

$$p(x) = \sum_{j=j_{\min}}^{j_{\max}} p_j(x) v(\tau_{k-1}, x_{k-1,j}), \tag{17}$$

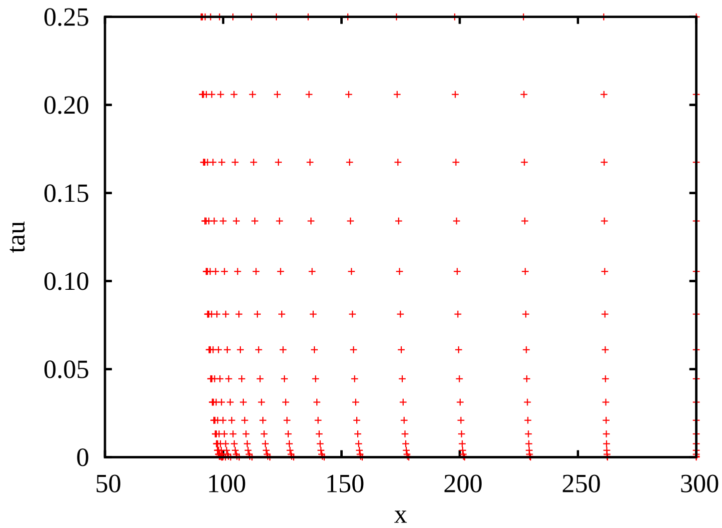

Figure 1: A 16×16 moving space-time grid.

where

$$p_j(x) = \prod_{\substack{l=j_{\min} \\ l \neq j}}^{j_{\max}} \frac{x - x_{k-1,l}}{x_{k-1,j} - x_{k-1,l}}. \tag{18}$$

Under the assumption that $v(\tau_{k-1}, \cdot)$ is N times continuously differentiable in the interval $[x_{k-1,j_{min}}, x_{k-1,j_{max}}]$, the Lagrange interpolation gives an approximation

$$v(\tau_{k-1}, x_{k,i}) = p(x_{k,i}) + \mathcal{O}\left((x_{k-1,j_{\max}} - x_{k-1,j_{\min}})^N\right). \tag{19}$$

Let us consider the interpolation in the leftmost interval $[x_{k-1,1}, x_{k-1,2}]$. We choose $j_{\min} = 1$ and $j_{\max} = 2$, that is, we use only two grid point values. Then with the grid (15), we have $x_{k-1,j_{\max}} - x_{k-1,j_{\min}} = \mathcal{O}\left(\Delta x^3\right)$ and

$$v(\tau_{k-1}, x) = p(x) + \mathcal{O}\left(\Delta x^6\right) \tag{20}$$

for any $x \in [x_{k-1,1}, x_{k-1,2}]$. As another example, we consider some interior interval $[x_{k-1,j_{\min}}, x_{k-1,j_{\max}}]$ such that $j_{\max} = j_{\min} + 4$, that is, $N = 5$. We have $x_{k-1,j_{\max}} - x_{k-1,j_{\min}} = \mathcal{O}\left(\Delta x\right)$ and

$$v(\tau_{k-1}, x) = p(x) + \mathcal{O}\left(\Delta x^5\right) \tag{21}$$

for any $x \in [x_{k-1,j_{\min}}, x_{k-1,j_{\max}}]$. Figure 2 shows an example which uses three grid point values for the interpolation.

6 Discretization of derivatives and integrals

The partial derivatives $v_x(\tau_k, x_{k,i})$, $v_{xx}(\tau_k, x_{k,i})$, and $v_\tau(\tau_k, x_{k,i})$ as well as the integral $\int_{\mathbb{R}_+} v(\tau_k, x_{k,i}y) f(y)\, dy$ needs to be discretized in order to solve the equations (12)

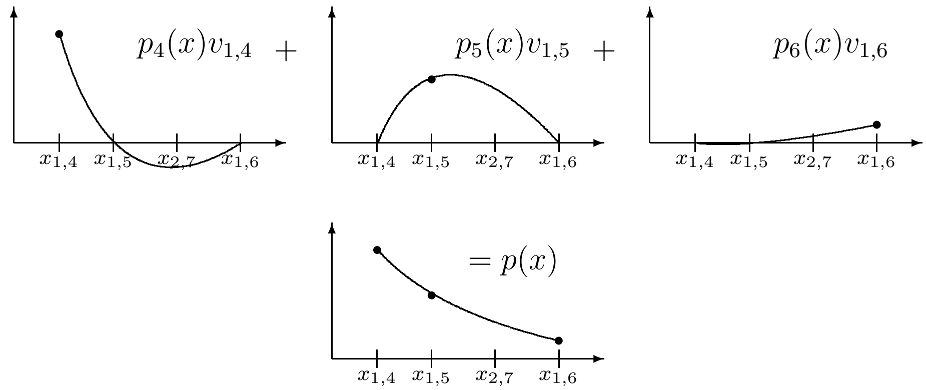

Figure 2: A three point interpolation for approximating $v(\tau_1, x_{2,7})$.

numerically using an implicit time stepping scheme. We could derive explicit formulas for finite difference stencils for the derivatives starting from the Taylor series presentations, but the nonuniform grids make the formulas fairly cumbersome and especially so when more than second-order accurate stencils are sought. Instead we will employ the Lagrange interpolation polynomials to construct the same finite difference stencils.

For the derivative $v_x(\tau_k, x_{k,i})$, we use the approximation

$$v_x(\tau_k, x_{k,i}) = p'(x_{k,i}) + \mathcal{O}\left(\Delta x^{N-1}\right) \tag{22}$$

based on N grid point values at $x_{k,j_{\min}}, \ldots, x_{k,j_{\max}}$. The indices $j_{\min}$ and $j_{\max}$ are chosen to be either $i \pm 1$ or $i \pm 2$ which lead to three and five point stencils, respectively. The derivative of the interpolating polynomial is

$$\begin{aligned} p'(x_{k,i}) &= \sum_{j=j_{\min}}^{j_{\max}} p'_j(x_{k,i}) v(\tau_k, x_{k,j}) \\ &= \begin{pmatrix} p'_{j_{\min}}(x_{k,i}) & \cdots & p'_{j_{\max}}(x_{k,i}) \end{pmatrix} \begin{pmatrix} v_{k,j_{\min}} \\ \vdots \\ v_{k,j_{\max}} \end{pmatrix}. \end{aligned} \tag{23}$$

The above row vector gives the finite difference stencil for approximating the derivative v_x. The accuracy of the above approximation of v_x can be analyzed using the equation (22) and the definition of the grid in (15). Near the free boundary $x_{k,1}$ the accuracy of the three point stencil is $\mathcal{O}\left(\Delta x^6\right)$ and away from the free boundary the accuracy of the five point stencil is $\mathcal{O}\left(\Delta x^4\right)$ assuming that the solution v is smooth enough.

Similarly, for v_{xx} we use an approximation

$$v_{xx}(\tau_k, x_{k,i}) \approx p''(x_{k,i}) = \sum_{j=j_{\min}}^{j_{\max}} p''_j(x_{k,i}) v_{k,j}. \tag{24}$$

A straightforward analysis of the truncation error using Taylor series gives us the accuracy $\mathcal{O}\left(\Delta x^{N-2}\right)$ for this approximation. This can be improved by employing the properties of the grid. Let a twice differentiable function s in the interval $[0,1]$ give the grid points by the formula

$$x_{k,i} = s\left(\frac{i-1}{n-1}\right), \quad i = 1, \ldots, n. \tag{25}$$

The definition of the grid in (15) is of this form. As s is twice differentiable, by Taylor's theorem it has a presentation

$$s(t) = a + bt + c(t)t^2, \tag{26}$$

where a and b are constants, and $c(t)$ is a bounded function. By combining this with the Taylor series for the truncation errors of the stencils we obtain for the three point stencil for v_{xx} the accuracy $\mathcal{O}\left(\Delta x^2\right)$; see [23]. Furthermore, near the free boundary this improves to $\mathcal{O}\left(\Delta x^6\right)$ as grid steps are shorter. For the five point stencil for v_{xx} we obtain the accuracy $\mathcal{O}\left(\Delta x^4\right)$.

Similarly for the time derivative $v_\tau(\tau_k, x_{k,i})$, we employ one-sided finite differences given by the Lagrange interpolation in time. An approximation using M previous time levels reads

$$v_\tau(\tau_k, x_{k,i}) \approx q'(x_{k,i}) = \sum_{j=k-M}^{k} q_j'(\tau_k) v(\tau_j, x_{k,i}), \tag{27}$$

where

$$q_j(\tau) = \prod_{\substack{l=k-M \\ l \neq j}}^{k} \frac{\tau - \tau_l}{\tau_j - \tau_l}. \tag{28}$$

The values $v(\tau_j, x_{k,i})$, $j < k$, need to be interpolated as almost always at the time level τ_j there is no grid point at $x_{k,i}$.

Using only one previous time level ($M = 1$) we obtain

$$v_\tau(\tau_k, x_{k,i}) \approx \frac{1}{\Delta\tau}\left[v(\tau_k, x_{k,i}) - v(\tau_{k-1}, x_{k,i})\right], \tag{29}$$

which is this implicit Euler method and its accuracy is $\mathcal{O}(\Delta t)$. If we would use two previous time levels ($M = 2$) and uniform time steps $\Delta\tau$, we would obtain

$$v_\tau(\tau_k, x_{k,i}) \approx \frac{1}{\Delta\tau}\left[\frac{3}{2}v(\tau_k, x_{k,i}) - 2v(\tau_{k-1}, x_{k,i}) + \frac{1}{2}v(\tau_{k-2}, x_{k,i})\right]. \tag{30}$$

This is the second-order backward difference formula (BDF2) and its accuracy is $\mathcal{O}(\Delta t^2)$. In general, the Lagrange polynomial approximation (27) employing M previous time levels gives us the Mth-order backward difference formula (BDFM) for nonuniform time steps.

At the first time level τ_1 only the initial value $v(0, x) = g(x)$ is available for the time stepping. For this reason we perform the first time step using the implicit Euler (BDF1) method. At the second time steps we can use the BDF2 method as two previous time levels exist. In general, we can use at the Mth time level the BDFM method. In practice, we limit M to be at most 2 or 4 depending on are we aiming for second-order or fourth-order convergence. Furthermore, to improve the stability of time stepping we take four first time steps with the implicit Euler method. This leads to the sequence BDF1, BDF1, BDF1, BDF1, BDF2, BDF3, BDF4, BDF4, $\ldots$, of employed time stepping methods when fourth-order accuracy is sought. It can easily shown using the definition of time steps (14) and the approximation (27) that this sequence leads to fourth-order accuracy with respect to the maximum time step size $\Delta\tau$ if the solution is sufficiently regular.

7 Solution of resulting nonlinear systems

We consider a generic time step from τ_{k-1} to τ_k, $k = 1, \ldots, m$, for the nonlinear free boundary problem (12). To simplify the notations, we denote the location of the free boundary $X_f(\tau_k)$ by ξ^k. The Cauchy boundary condition (11), that is, both Dirichlet and Neumann boundary conditions, is posed at ξ^k. Discretizing the P(I)DE with the Neumann boundary condition $v_x = -1$ at the free boundary $x = \xi^k$ leads to a system

$$\boldsymbol{A}^k(\xi^k)\boldsymbol{v}^k(\xi^k) = \boldsymbol{f}^k(\xi^k), \tag{31}$$

where $\boldsymbol{A}^k(\xi^k)$ is an $n \times n$ matrix which depends on ξ^k, and $\boldsymbol{v}^k(\xi^k)$ is a vector containing the grid point values $v(\tau_k, x_{k,i})$, $i = 1, \ldots, n$. The discrete solution $\boldsymbol{v}^k(\xi^k)$ also has to satisfy the Dirichlet boundary condition

$$d^k(\xi^k) = v^k(\tau_k, \xi^k) - g(\xi^k) = \boldsymbol{v}_1^k(\xi^k) - (K - \xi^k) = 0. \tag{32}$$

Alternatively we could have formed a similar system to (31) using the Dirichlet boundary condition and then give an equation resembling (32) enforcing the Neumann boundary condition.

We solve the nonlinear system for $\boldsymbol{v}(\xi^k)$ and ξ^k given by (31) and (32) iteratively. Substituting the solution of (31) to the equation (32) gives us a nonlinear equation

$$d^k(\xi^k) = \left[\left(\boldsymbol{A}^k(\xi^k)\right)^{-1}\boldsymbol{f}^k(\xi^k)\right]_1 - (K - \xi^k) = 0 \tag{33}$$

with only one variable ξ^k. A host of methods can used to find the root of the function d^k. This needs to be performed at each time step and it is fairly expensive to compute the value of the function d^k as it involves forming and solving a system of linear equations. For these reason the root finding method should be efficient. One possibility would be to use Newton's method which has been used in [17], [28], [30], [33], for example. Newton's method requires the derivative of d^k. It can be computed in a fairly straightforward manner, but it still requires some computational and implementation effort. Furthermore, sometimes Newton's method can have difficulties to converge.

We propose to use Brent's method [6] for finding the root of d^k defined by (33). It is a derivative free method which uses selectively the bisection method, the secant method, and an inverse quadratic interpolation to locate the root. We use a C version of an implementation called zeroin.f which is available from www.netlib.org mathematical software repository. In order to use Brent's method, we first need to find an interval where the function d^k changes its sign. We choose the right end of this interval to be ξ^{k-1}, that is, the location of the free boundary at the previous time step. We extrapolate our initial guess for the left end using the previous free boundary locations. Then we keep moving the left end to the left until d^k has opposite sign at it. Brent's method together with finding an initial interval with sign change leads to very robust algorithm. Furthermore, numerical experiments in Section 8 demonstrate this approach to be also efficient.

Each evaluation of the function $d^k(\xi^k)$ requires the solution of the system of linear equations (31). In absence of jumps in the model for the underlying asset, the matrix $\boldsymbol{A}^k$ is banded. Furthermore, if we use a three point interpolation for discretizing the partial derivatives v_x and v_{xx} the matrix $\boldsymbol{A}^k$ is tridiagonal while employing a five point interpolation leads $\boldsymbol{A}^k$ to be pentadiagonal. These systems can be solved efficiently using $\boldsymbol{LU}$ decomposition with $\mathcal{O}(n)$ floating point operations.

When the model includes jumps the matrix $\boldsymbol{A}^k$ is full and the use of $\boldsymbol{LU}$ decomposition requires $\mathcal{O}(n^3)$ floating point operations. If sufficient accuracy is already obtained with a fairly small number of grid points n the computational cost $\mathcal{O}(n^3)$ operations can be reasonable. This computational cost can be reduced by employing a stationary iterative method to solve (31); see [1], [11], [34], [35]. In this approach, we need to solve systems of linear equations with banded matrices and perform multiplications by full matrices. A straightforward implementation of this approach requires $\mathcal{O}(n^2)$ operations. This can further reduced to $\mathcal{O}(n \log n)$ operations by employing FFT in the multiplication; see [1], [11], [34]. With Kou's jump-diffusion model [22] the multiplication and also the solution of (31) can be performed in $\mathcal{O}(n)$ operations [35].

8 Numerical examples

8.1 Black-Scholes model

As a first numerical example we consider a model without jumps namely the Black-Scholes model. We price an American put option with parameters:

$$\sigma = 0.15, \quad r = 0.05, \quad T = 0.25, \quad \text{and} \quad K = 100. \tag{34}$$

We choose the truncation boundary to be at $X = 300$. Figure 1 shows a 16×16 moving space-time grid for this pricing problem.

Table 1 reports the numerical results when up to three grid points are used in the interpolation and approximating the derivatives. This leads to a three point finite difference stencil for the space derivatives at interior grid points and, thus, to a

m	n	$X_f(T)$	price at K	change at K	ratio	eval.
8	8	91.823438	2.46429445			4.5
16	16	90.615083	2.56012241	9.6×10^{-2}		3.1
32	32	90.775686	2.52365658	3.6×10^{-2}	2.6	3.6
64	64	90.810374	2.50969853	1.4×10^{-2}	2.6	3.8
128	128	90.819280	2.50582415	3.9×10^{-3}	3.6	3.8
256	256	90.821631	2.50490066	9.2×10^{-4}	4.2	4.1
512	512	90.822171	2.50468109	2.2×10^{-4}	4.2	4.2

Table 1: Numerical results based on up to three point interpolations for the Black-Scholes model.

m	n	$X_f(T)$	price at K	change at K	ratio	eval.
8	8	91.846875	2.11026452			4.2
16	16	90.763886	2.45112014	3.4×10^{-1}		2.9
32	32	90.834422	2.49470891	4.4×10^{-2}	7.8	3.8
64	64	90.820855	2.50514310	1.0×10^{-2}	4.2	3.8
128	128	90.822367	2.50459012	5.5×10^{-4}	18.9	3.9
256	256	90.822317	2.50460859	1.8×10^{-5}	29.9	4.2
512	512	90.822341	2.50460903	4.3×10^{-7}	42.7	4.2

Table 2: Numerical results based on up to five point interpolations for the Black-Scholes model.

tridiagonal matrices $\boldsymbol{A}^k$ in (31). Furthermore, the four first time steps are performed using the implicit Euler method and then the BDF2 method is used. With this discretization at most second-order convergence can be expected with respect to the number of time and space steps m and n. Table 2 reports the results when up to five grid points are used for interpolations and finite differences. Thus, we obtain five point finite difference stencils for the space derivatives and pentadiagonal matrices $\boldsymbol{A}^k$. This discretization can lead up to fourth-order convergence with respect to m and n.

In Tables 1 and 2, the number of time steps m and the number of space grid points n varies from 8 to 512. The tables give the location of the free boundary X_f at the expiry T, the option price at K. The column "change at K" gives the change in the price between consecutive grids and the column "ratio" is the ratio of consecutive changes. With second-order and fourth-order methods the ratios should asymptotically approach value $4 = 2^2$ and $16 = 2^4$, respectively, as m and n are doubled from a row to the next one. The column "eval." gives the average number of evaluations of the function d^k in (33) per time step. This measures how efficiently the location of the free boundary can be found using Brent's method at each time step.

The ratios in Table 1 approach value 4 which suggest that the discretizations based on three point interpolations lead to second-order convergence. By comparing the prices in Tables 1 and 2 we observe that with small number of time and space steps m and n five point interpolation does not increase accuracy. But when m and n increase the five point interpolation start to converge much faster. The ratios in Table 2 for larger number of steps are higher than the expected 16. A possible reason is that the discretization of some part of the PDE is superconvergent. Based on these results we conclude that it is beneficial to use five point interpolations when highly accurate prices are sought. In both tables, the average number of function evaluations for the root finding is roughly four. Thus, Brent's method can find very efficiently the location of the free boundary.

8.2 Kou's jump-diffusion model

In our second example, we use Kou's model [22] which has a log-double-exponential jump distribution given by

$$f(y) = \begin{cases} q\alpha_2 y^{\alpha_2 - 1}, & y < 1 \\ p\alpha_1 y^{-\alpha_1 - 1}, & y \geq 1, \end{cases} \tag{35}$$

where p, q, $\alpha_1 > 1$, and α_2 are positive constants such that $p + q = 1$. We use the same model parameter values as in [11], [35] which are given by (34) and for the jump part by

$$\lambda = 0.1, \quad \alpha_1 = 3.0465, \quad \alpha_2 = 3.0775, \quad \text{and} \quad p = 0.3445. \tag{36}$$

As with the Black-Scholes model we choose the truncation boundary to be at $X = 300$ which is three times the strike price K.

Tables 3 and 4 contain the same information under Kou's model as Tables 1 and 2 under the Black-Scholes model. The price of the option is slightly higher under Kou's model. The results are similar to the ones under the Black-Scholes model and, thus, the same conclusions also hold. Particularly, it is possible to obtain second-order convergence with three point interpolations and fourth-order convergence with five point interpolations also under a jump-diffusion model.

We used the stationary iterative method mentioned in Section 7 to solve a linear system with the full matrices $\boldsymbol{A}^k$. For each evaluation of the function d^k we need to solve such a system. Usually only two stationary iterations were enough to obtain the solution with sufficiently high accuracy so that the overall accuracy of option price is essentially the same. With an efficient implementation of the stationary iteration the computational cost increases only a few times when a jump is added to the model.

m	n	$X_f(T)$	price at K	change at K	ratio	eval.
8	8	89.153125	3.08121419			4.9
16	16	89.339929	2.89587743	1.9×10^{-1}		3.2
32	32	89.453671	2.83490569	6.1×10^{-2}	3.0	3.9
64	64	89.496218	2.81490852	2.0×10^{-2}	3.0	3.6
128	128	89.506910	2.80955934	5.3×10^{-3}	3.7	3.9
256	256	89.509905	2.80828299	1.3×10^{-3}	4.2	4.1
512	512	89.510601	2.80797806	3.0×10^{-4}	4.2	4.2

Table 3: Numerical results based on up to three point interpolations for Kou's model.

m	n	$X_f(T)$	price at K	change at K	ratio	eval.
8	8	89.473125	2.66506107			4.8
16	16	89.609046	2.73163569	6.7×10^{-2}		3.0
32	32	89.533852	2.79564411	6.4×10^{-2}	1.0	3.8
64	64	89.507469	2.80840720	1.3×10^{-2}	5.0	3.8
128	128	89.510756	2.80786703	5.4×10^{-4}	23.6	3.9
256	256	89.510799	2.80787736	1.0×10^{-5}	52.3	4.1
512	512	89.510831	2.80787779	4.4×10^{-7}	23.6	4.1

Table 4: Numerical results based on up to five point interpolations for Kou's model.

9 Conclusion

We considered a free boundary formulation for pricing American put options under jump-diffusion models with finite jump activity. This approach can be also easily employed for call options when underlying asset pays continuously dividends. An advantage of this formulation is that can easily develop higher-order methods by tracking the location of the free boundary and then by refining grids sufficiently near the free boundary where the solution is less regular. We proposed second-order and fourth-order accurate discretizations with respect to the number of time and space steps. The numerical experiments demonstrated that these convergence rates are attainable under the Black-Scholes model and Kou's jump-diffusion model. We expect that the considered approach would be well-suited also for developing higher than fourth-order methods.

As our space grids move in time we need to interpolate grid functions between grids for which we employed Lagrange interpolations. We also used the Lagrange interpolation polynomials to construct finite difference discretization on nonuniform space-time grids. This gives an easy way to derive high-order stencils on nonuniform grids. The implicit discretization leads to a system of nonlinear equations for the price and the location of the free boundary at each time step. We solve this as a root finding problem for the the location of the free boundary using derivative free Brent's method. The numerical experiments showed that this method can find the root in a few iterations. Thus, its computation efficiency should be comparable to Newton's method, but it is easier to use and it is more robust. The accuracy and the number of root finding iterations with the Black-Scholes model and Kou's jump-diffusion model were equally good. The main difference was that with jumps in the model the computational cost is a few times higher. This is very reasonable as at each time step it is necessary to solve systems of linear equations with full matrices instead of tridiagonal or pentadiagonal matrices.

References

[1] A. ALMENDRAL AND C. W. OOSTERLEE, *Numerical valuation of options with jumps in the underlying*, Appl. Numer. Math., 53 (2005), pp. 1–18.

[2] ——, *Accurate evaluation of European and American options under the CGMY process*, SIAM J. Sci. Comput., 29 (2007), pp. 93–117.

[3] L. ANDERSEN AND J. ANDREASEN, *Jump-diffusion processes: Volatility smile fitting and numerical methods for option pricing*, Rev. Deriv. Res., 4 (2000), pp. 231–262.

[4] F. BLACK AND M. SCHOLES, *The pricing of options and corporate liabilities*, J. Political Economy, 81 (1973), pp. 637–654.

[5] M. J. BRENNAN AND E. S. SCHWARTZ, *The valuation of American put options*, J. Finance, 32 (1977), pp. 449–462.

[6] R. P. BRENT, *Algorithms for minimization without derivatives*, Prentice-Hall Inc., Englewood Cliffs, N.J., 1973.

[7] P. CARR, H. GEMAN, D. B. MADAN, AND M. YOR, *The fine structure of asset returns: an empirical investigation*, J. Business, 75 (2002), pp. 305–332.

[8] R. CONT AND P. TANKOV, *Financial modelling with jump processes*, Chapman & Hall/CRC, Boca Raton, FL, 2004.

[9] R. CONT AND E. VOLTCHKOVA, *A finite difference scheme for option pricing in jump diffusion and exponential Lévy models*, SIAM Numer. Anal., 43 (2005), pp. 1596–1626.

[10] Y. D'HALLUIN, P. A. FORSYTH, AND G. LABAHN, *A penalty method for American options with jump diffusion processes*, Numer. Math., 97 (2004), pp. 321–352.

[11] Y. D'HALLUIN, P. A. FORSYTH, AND K. R. VETZAL, *Robust numerical methods for contingent claims under jump diffusion processes*, IMA J. Numer. Anal., 25 (2005), pp. 87–112.

[12] B. DUPIRE, *Pricing with a smile*, Risk, 7 (1994), pp. 18–20.

[13] B. ERAKER, M. JOHANNES, AND N. POLSON, *The impact of jumps in volatility and returns*, J. Finance, 58 (2003), pp. 1269–1300.

[14] P. A. FORSYTH AND K. R. VETZAL, *Quadratic convergence for valuing American options using a penalty method*, SIAM J. Sci. Comput., 23 (2002), pp. 2095–2122.

[15] H. HAN AND X. WU, *A fast numerical method for the Black-Scholes equation of American options*, SIAM J. Numer. Anal., 41 (2003), pp. 2081–2095.

[16] S. HESTON, *A closed-form solution for options with stochastic volatility with applications to bond and currency options*, Rev. Financial Stud., 6 (1993), pp. 327–343.

[17] A. D. HOLMES AND H. YANG, *A front-fixing finite element method for the valuation of American options*, SIAM J. Sci. Comput., 30 (2008), pp. 2158–2180.

[18] J. HUANG AND J.-S. PANG, *Option pricing and linear complementarity*, J. Comput. Finance, 2 (1998), pp. 31–60.

[19] S. IKONEN AND J. TOIVANEN, *Operator splitting methods for American option pricing*, Appl. Math. Lett., 17 (2004), pp. 809–814.

[20] K. ITO AND K. KUNISCH, *Parabolic variational inequalities: The Lagrange multiplier approach*, J. Math. Pures Appl., 85 (2006), pp. 415–449.

[21] P. JAILLET, D. LAMBERTON, AND B. LAPEYRE, *Variational inequalities and the pricing of American options*, Acta Appl. Math., 21 (1990), pp. 263–289.

[22] S. G. KOU, *A jump-diffusion model for option pricing*, Management Sci., 48 (2002), pp. 1086–1101.

[23] T. A. MANTEUFFEL AND A. B. WHITE, JR., *The numerical solution of second-order boundary value problems on nonuniform meshes*, Math. Comp., 47 (1986), pp. 511–535.

[24] A.-M. MATACHE, C. SCHWAB, AND T. P. WIHLER, *Fast numerical solution of parabolic integrodifferential equations with applications in finance*, SIAM J. Sci. Comput., 27 (2005), pp. 369–393.

[25] A. MAYO, *High-order accurate implicit finite difference method for evaluating American options*, European J. Finance, 10 (2004), pp. 212–237.

[26] R. C. MERTON, *Theory of rational option pricing*, Bell J. Econom. Management Sci., 4 (1973), pp. 141–183.

[27] ——, *Option pricing when underlying stock returns are discontinuous*, J. Financial Econ., 3 (1976), pp. 125–144.

[28] B. F. NIELSEN, O. SKAVHAUG, AND A. TVEITO, *Penalty and front-fixing methods for the numerical solution of American option problems*, J. Comput. Finance, 5 (2002), pp. 69–97.

[29] C. W. Oosterlee, C. C. W. Leentvaar, and X. Huang, *Accurate American option pricing by grid stretching and high order finite differences*, tech. rep., Delft Institute of Applied Mathematics, Delft University of Technology, Delft, the Netherlands, 2005.

[30] K. N. Pantazopoulos, E. N. Houstis, and S. Kortesis, *Front-tracking finite difference methods for the valuation of American options*, Comput. Econ., 12 (1998), pp. 255–273.

[31] H. Pham, *Optimal stopping, free boundary, and American option in a jump-diffusion model*, Appl. Math. Optim., 35 (1997), pp. 145–164.

[32] D. Y. Tangman, A. Gopaul, and M. Bhuruth, *A fast high-order finite difference algorithm for pricing American options*, J. Comput. Appl. Math., (2008). to appear.

[33] ——, *Numerical pricing of options using high-order compact finite difference schemes*, J. Comput. Appl. Math., 218 (2008), pp. 270–280.

[34] D. Tavella and C. Randall, *Pricing financial instruments: The finite difference method*, John Wiley & Sons, Chichester, 2000.

[35] J. Toivanen, *Numerical valuation of European and American options under Kou's jump-diffusion model*, SIAM J. Sci. Comput., 30 (2008), pp. 1949–1970.

[36] L. Wu and Y. K. Kwok, *A front-fixing finite difference method for the valuation of American options*, J. Financial Eng., 6 (1997), pp. 83–97.

[37] X. L. Zhang, *Numerical analysis of American option pricing in a jump-diffusion model*, Math. Oper. Res., 22 (1997), pp. 668–690.

[38] Y.-l. Zhu, B.-M. Chen, H. Ren, and H. Xu, *Application of the singularity-separating method to American exotic option pricing*, Adv. Comput. Math., 19 (2003), pp. 147–158.