

Reports of the Department of Mathematical Information Technology
Series B. Scientific Computing
No. B 11/2004

Operator Splitting Methods for Pricing American Options with Stochastic Volatility

Samuli Ikonen Jari Toivanen

University of Jyväskylä
Department of Mathematical Information Technology
P.O. Box 35 (Agora)
FIN-40014 University of Jyväskylä
FINLAND
fax +358 14 260 2731
<http://www.mit.jyu.fi/>

Copyright © 2004
Samuli Ikonen and Jari Toivanen and
University of Jyväskylä
ISBN 951-39-1980-3
ISSN 1456-436X

Operator Splitting Methods for Pricing American Options with Stochastic Volatility

Samuli Ikonen*

Jari Toivanen *

Abstract

Stochastic volatility models lead to more realistic option prices than the Black-Scholes model which uses a constant volatility. Based on such models a two-dimensional parabolic partial differential equation can be derived for option prices. Due to the early exercise possibility of American option contracts the arising pricing problems are free boundary problems. In this paper we consider the numerical pricing of American options when the volatility follows a stochastic process.

We propose operator splitting methods for performing time stepping after a finite difference space discretization is done. The idea is to decouple the treatment of the early exercise constraint and the solution of the system of linear equations into separate fractional time steps. With this approach we can use any efficient numerical method to solve the system of linear equations in the first fractional step before making a simple update to satisfy the early exercise constraint. Our analysis suggests that the Crank-Nicolson method and the operator splitting method based on it have the same asymptotic order of accuracy. The numerical experiments show that the operator splitting does not increase essentially the error. They also demonstrate the efficiency of the operator splitting methods when a multigrid method is used for solving the systems of linear equations.

Keywords: American option pricing, stochastic volatility model, operator splitting method, time discretization, multigrid method, linear complementarity problem

1 Introduction

The option prices obtained using the Black-Scholes model [2] with a constant volatility are not consistent with observed option prices. One possible remedy is to make the volatility to be a function of time and strike price. By calibrating this function consistent option prices can be obtained. Another approach is to assume that the volatility of the price process is also stochastic. Such models have been considered for example

*Department of Mathematical Information Technology, PO BOX 35 (Agora), FIN-40014 University of Jyväskylä, Finland, Samuli.Ikonen@mit.jyu.fi, Jari.Toivanen@mit.jyu.fi

in [1], [13], [17], [18]. In this paper, we will use the stochastic volatility model introduced by Heston for pricing American options [17]. This model seems to be consistent with market prices. For example, Drăgulescu and Yakovenko showed that the time-dependent probability distribution of stock price changes generated by Heston's model is in good agreement with the Dow-Jones index after the calibration of the parameters appearing in the model [10].

The price of European options can be calculated using analytical expressions for Heston's model [17]. The early exercise possibility of American options leads to nonlinear free boundary problems described by partial differential inequalities which are also called in different disciplines as variational inequalities, linear complementarity problems and obstacle problems. No useful analytical methods are available for pricing American options and, thus, numerical methods are used to approximate the prices. Already Brennan and Schwartz proposed a finite difference discretization for the one-dimensional Black-Scholes equation and then a direct solution method for pricing American options [4]. Heston's model has an additional space variable related to the volatility. In numerical approximations it is usually chosen to be the variance of the price process, which is the square of the volatility. This additional variable makes the discretization and solution procedures more complicated as well as computationally more expensive. In the following we review three papers and one report which have considered such option pricing problems.

Clarke and Parrott considered the discretization of a partial differential inequality for the price of an American option using finite differences for space derivatives and a slightly stabilized Crank-Nicolson method for the time derivative [7], [8]. They performed a coordinate transformation to increase the accuracy of the space discretization with uniform grid step sizes. They proposed a special version of a projected full approximation scheme (PFAS) multigrid [3] for the solving arising linear complementarity problems. Their method uses a special projected line Gauss-Seidel smoother which in part made the considered method rather complicated and problem specific. The advantage of such a multilevel method is that the number of iterations required to solve a linear complementarity problem is essentially independent of the grid step size and, thus, the computational cost of this iterative solution is of optimal order.

Zvan, Forsyth, and Vetzal discretized the partial differential inequality using finite element/volume method together with a nonlinear flux limiter for convection terms [28]. They formulated linear complementarity problems using nonlinear penalty terms. The solutions of the arising nonlinear problems were obtained with an inexact Newton method and the linear problems with approximate Jacobians were solved with an incomplete LU preconditioned CGSTAB method. The method proposed by Zvan, Forsyth and Vetzal is simpler than the one by Clarke and Parrott, but the number of iterations required by the CGSTAB method increases when the grid step size decreases.

Oosterlee performed the space discretization using central differences for the diffusion and second-order upwind differences for the convection [23]. The time discretization was based on the second-order accurate backward finite difference scheme. Oosterlee studied the PFAS multigrid for linear complementarity problems. The quality of various smoothers for the multigrid was analyzed. This led to the conclusion

that only an alternating line smoother is robust. Also a recombination technique for iterates was proposed and tested. It improved the convergence of multigrid methods, but it could not regain a grid step size independent convergence if the multigrid method did not have this property without the recombination. Based on [23], it seems that a robust and efficient PFAS multigrid has to use a rather involved smoother for linear complementarity problems arising from the pricing American options using Heston's model.

Due to the previous observations we will pursue an alternative way to treat the early exercise constraint. Our approach is based on operator splitting methods which are commonly used to handle the incompressibility constraint in computational fluid dynamics [14]. Previously operator splitting methods have been applied for obstacle problems by Lions and Mercier [21], for example. In our earlier paper, [20], we proposed operator splitting methods for pricing American options using the Black-Scholes model. The idea is divide each time step into two fractional time steps. The first step integrates a partial differential equation with an auxiliary variable over the time step. The second step makes a simple update so that the solution and the auxiliary variable satisfies the linear complementarity conditions due to the early exercise constraint. Such a splitting introduces additional error and, thus, it is necessary to show that this error is sufficiently small so that this approach can be used to price options accurately. Our numerical experiments in [20] showed that for the Black-Scholes model the accuracy with a splitting is essentially the same as without it. The computational cost of the operator splitting method is about the same as the cost of the explicit payoff method which has been considered for one-dimensional American option pricing problems in [9], [11], [19], for example. The difference between these methods is that the operator splitting method uses the auxiliary variable to improve the accuracy.

The purpose of this paper is to introduce operator splitting methods for pricing American options using Heston's model and also study their accuracy and efficiency. We analyse the difference between the solutions obtained using the Crank-Nicolson method and the operator splitting method based on it. We also compare numerically the accuracy of solutions when the operator splitting method and the projected SOR (PSOR) method are used. Our analysis and numerical experiments suggest that the additional error due to the splitting does not reduce the order of accuracy when compared to the corresponding unsplit scheme. The operator splitting method enables the use of efficient solution procedures for the system of linear equations since the early exercise constraint is treated in a separate fractional time step. In the numerical experiment we apply a multigrid method at the first fractional step of the splitting.

The outline of this paper is the following. In the second section we describe Heston's model and a partial differential inequality for pricing American options. The third section introduces finite difference discretizations applied for the parabolic partial differential equation. In the fourth section we propose our operator splitting method in the case of four different time discretization schemes. The next section analyses the operator splitting method based on the Crank-Nicolson method. The multigrid method used in our experiments is described in the following section. The seventh section presents numerical experiments and the last section contains conclusions.

2 Option pricing model

In the following we describe stock price and variance processes, a partial differential inequality, an initial value and boundary conditions for the American option pricing model. This section defines a problem whose numerical solution is studied in the consecutive sections. Our formulation and notations are based on [17], [23], [28].

In Heston's model, stochastic differential equations

$$dx_t = \mu x_t dt + \sqrt{y_t} x_t dw_1, \quad (1)$$

$$dy_t = \alpha(\beta - y_t)dt + \gamma\sqrt{y_t}dw_2, \quad (2)$$

define the stock price process x_t and the variance process y_t . Equation (1) models the stock price process x_t . The parameter μ is the deterministic growth rate of the stock price and $\sqrt{y_t}$ is the standard deviation (the volatility) of the stock returns dx/x . The model for the variance process y_t is given by (2). The volatility of the variance process y_t is denoted by γ and the variance will drift back to a mean value $\beta > 0$ at a rate $\alpha > 0$. These two processes contain randomness, that is, w_1 and w_2 are Brownian motions with a correlation factor $\rho \in [-1, 1]$ [8], [28].

A two-dimensional parabolic partial differential inequality can be derived for the price of the American option using the previous stochastic volatility model; see for example [28] and references therein. The generalised Black-Scholes operator is

$$Lu := \frac{\partial u}{\partial t} - \frac{1}{2}yx^2\frac{\partial^2 u}{\partial x^2} - \rho\gamma yx\frac{\partial^2 u}{\partial x\partial y} - \frac{1}{2}\gamma^2 y\frac{\partial^2 u}{\partial y^2} - rx\frac{\partial u}{\partial x} - \{\alpha(\beta - y) - \vartheta\gamma\sqrt{y}\}\frac{\partial u}{\partial y} + ru, \quad (3)$$

where the parameter ϑ is a so-called market price of the risk. In the following we assume ϑ to be zero as has been done in many previous studies like in [23]. The original option pricing problem is a final value problem, since the value of the option is known at the expiry. Similarly to [8], [17], [23], we have transformed this problem to be an initial value problem with the operator L in (3) which is a more common form.

The option pricing problem is defined in an unbounded domain $\{(x, y, t) \mid x \geq 0, y \geq 0, t \in [0, T]\}$. In order to use finite difference approximations for space variables, we truncate it into a finite size computational domain

$$(x, y, t) \in [0, X] \times [0, Y] \times [0, T] =: \Omega \times [0, T], \quad (4)$$

where X and Y are sufficiently large.

For the American put option an initial value and boundary conditions are described in [23], for example. The initial value is

$$u(x, y, 0) = \max(E - x, 0), \quad (5)$$

where E is the exercise price and the boundary conditions are

$$u(0, y, t) = E, \quad (y, t) \in [0, Y] \times [0, T], \quad (6)$$

$$u(x, 0, t) = \max(E - x, 0), \quad (x, t) \in [0, X] \times [0, T], \quad (7)$$

$$\frac{\partial u(X, y, t)}{\partial x} = 0, \quad (y, t) \in [0, Y] \times [0, T], \quad (8)$$

$$\frac{\partial u(x, Y, t)}{\partial y} = 0, \quad (x, t) \in [0, X] \times [0, T]. \quad (9)$$

Due to the early exercise possibility of the American option, we have to include the following constraint for the option price:

$$u(x, y, t) \geq \max(E - x, 0) =: g(x), \quad (x, y, t) \in \Omega \times [0, T]. \quad (10)$$

The price of the American option based on the stochastic volatility model can be obtained by solving a time dependent complementarity problem

$$\begin{cases} Lu \geq 0, \\ u \geq g, \\ (u - g)Lu = 0, \end{cases} \quad (11)$$

for $(x, y, t) \in \Omega \times [0, T]$ with the initial and boundary conditions (5)–(9).

In this paper, we propose operator splitting methods to solve the linear complementarity problem (11). These splittings are based on the formulation with an auxiliary variable λ and hence, the complementarity problem is reformulated in the form

$$\begin{cases} Lu = \lambda, \\ \lambda \geq 0, \quad u \geq g, \\ (u - g)\lambda = 0, \end{cases} \quad (12)$$

for $(x, y, t) \in \Omega \times [0, T]$ with the initial and boundary conditions (5)–(9).

3 Discretization of the PDE

A numerical solution of the American option pricing problem (11) requires the discretization of the generalised Black-Scholes operator (3). In the following, we perform the discretization of the spatial derivatives using a seven point finite difference stencil and moreover, we consider four time discretization schemes.

The discretization is performed using a uniform space-time finite difference grid for the computational domain (4). Let the number of grid steps to be m , n and l in the x -direction, in the y -direction, and in the t -direction, respectively. The grid steps to these directions are denoted by $\Delta x := X/m$, $\Delta y := Y/n$, and $\Delta t := T/l$. Furthermore, the grid point values of a finite difference approximation are denoted by

$$u_{i,j}^{(k)} \approx u(x_i, y_j, t_k) = u(i\Delta x, j\Delta y, k\Delta t), \quad (13)$$

where $i = 0, \dots, m$, $j = 0, \dots, n$, and $k = 0, \dots, l$.

3.1 Space discretization

The two-dimensional Black-Scholes operator (3) has variable coefficients for all partial derivatives. In part of the domain the first-order derivatives dominate the second-order ones. The operator also contains a second-order cross-derivative term. Due to these reasons it is not easy to construct a discretization with good properties. The discretization of these spatial derivatives is considered for example in [8], [23], [28].

The stability of discretization schemes ensures that the numerical solution does not have undesired oscillations. The matrix which is strictly diagonal dominant with positive diagonal elements and non-positive off-diagonal elements is known to have good stability properties. This M -matrix property guarantees that the numerical solution does not have oscillations due to the space discretization. Usual finite difference approximations have positive off-diagonals and that is why the use of a special discretization is beneficial.

We use the following second-order accurate finite difference scheme which reduces the number of positive off-diagonal elements when compared to more standard discretizations. Generally it does not lead to an M -matrix. We study the properties of the discretization matrix in the end of this section. The space discretization is based on the central finite difference schemes and on a special approximation of the second-order cross-derivative term. In order to simplify notations, the derivation of the space discretization scheme is described for a partial differential equation with general coefficients

$$\frac{\partial u}{\partial t} + a \frac{\partial^2 u}{\partial x^2} + b \frac{\partial^2 u}{\partial x \partial y} + c \frac{\partial^2 u}{\partial y^2} + d \frac{\partial u}{\partial x} + e \frac{\partial u}{\partial y} + fu = 0. \quad (14)$$

The first-order and second-order spatial derivatives, except the cross-derivative term, are approximated with the standard second-order accurate central finite differences. For this purpose, we denote the finite difference operators for the first-order derivatives by

$$\delta_x u_{i,j} = \frac{u_{i+1,j} - u_{i-1,j}}{2\Delta x} \quad \text{and} \quad \delta_y u_{i,j} = \frac{u_{i,j+1} - u_{i,j-1}}{2\Delta y}, \quad (15)$$

and for the second-order derivatives by

$$\delta_x^2 u_{i,j} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2} \quad \text{and} \quad \delta_y^2 u_{i,j} = \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2}. \quad (16)$$

Next, we consider the discretization of the second-order cross-derivative term; a similar discretization for this term is described in [22]. We assume here that the coefficient b for the cross-derivative in (14) is non-positive. From the Taylor series we obtain approximations

$$\begin{cases} u(x_{i+1}, y_{j+1}) \approx u + \Delta x \frac{\partial u}{\partial x} + \Delta y \frac{\partial u}{\partial y} + \frac{1}{2} \left(\Delta x^2 \frac{\partial^2 u}{\partial x^2} + 2\Delta x \Delta y \frac{\partial^2 u}{\partial x \partial y} + \Delta y^2 \frac{\partial^2 u}{\partial y^2} \right), \\ u(x_{i-1}, y_{j-1}) \approx u - \Delta x \frac{\partial u}{\partial x} - \Delta y \frac{\partial u}{\partial y} + \frac{1}{2} \left(\Delta x^2 \frac{\partial^2 u}{\partial x^2} + 2\Delta x \Delta y \frac{\partial^2 u}{\partial x \partial y} + \Delta y^2 \frac{\partial^2 u}{\partial y^2} \right), \end{cases} \quad (17)$$

where the value for u and its derivatives on the right side are evaluated at the grid point (x_i, y_j) . The accuracy of these approximations is $\mathcal{O}(\max(\Delta x, \Delta y)^3)$. If b would be positive then we would form similar approximations for the grid point values $u(x_{i+1}, y_{j-1})$ and $u(x_{i-1}, y_{j+1})$, and use them in the derivation. By summing the equations in (17), we obtain

$$2\Delta x \Delta y \frac{\partial u}{\partial x \partial y} \approx u(x_{i+1}, y_{j+1}) - 2u(x_i, y_j) + u(x_{i-1}, y_{j-1}) - \Delta x^2 \frac{\partial^2 u}{\partial x^2} - \Delta y^2 \frac{\partial^2 u}{\partial y^2}, \quad (18)$$

and dividing this by $2\Delta x \Delta y$, we get

$$\frac{\partial u}{\partial x \partial y} \approx \frac{1}{2\Delta x \Delta y} [u(x_{i+1}, y_{j+1}) - 2u(x_i, y_j) + u(x_{i-1}, y_{j-1})] - \frac{\Delta x}{2\Delta y} \frac{\partial^2 u}{\partial x^2} - \frac{\Delta y}{2\Delta x} \frac{\partial^2 u}{\partial y^2}. \quad (19)$$

By performing the discretization of the cross-derivative using (19), we obtain an approximation for the second-order derivatives in (14) having the form

$$\begin{aligned} a \frac{\partial^2 u}{\partial x^2} + b \frac{\partial^2 u}{\partial x \partial y} + c \frac{\partial^2 u}{\partial y^2} \approx & \left[a - \frac{b\Delta x}{2\Delta y} \right] \frac{\partial^2 u}{\partial x^2} + \left[c - \frac{b\Delta y}{2\Delta x} \right] \frac{\partial^2 u}{\partial y^2} \\ & + \frac{b}{2\Delta x \Delta y} [u(x_{i+1}, y_{j+1}) - 2u(x_i, y_j) + u(x_{i-1}, y_{j-1})]. \end{aligned} \quad (20)$$

Using the central finite differences (15) and (16) together with (20), we can approximate the partial differential equation (14) by the semi-discrete equation

$$\begin{aligned} \frac{\partial u}{\partial t} + \left[a - \frac{b\Delta x}{2\Delta y} \right] \delta_x^2 u_{i,j} + \left[c - \frac{b\Delta y}{2\Delta x} \right] \delta_y^2 u_{i,j} + d\delta_x u_{i,j} + e\delta_y u_{i,j} + f u_{i,j} \\ + \frac{b}{2\Delta x \Delta y} [u_{i+1,j+1} - 2u_{i,j} + u_{i-1,j-1}] = 0. \end{aligned} \quad (21)$$

After using the definitions (15) and (16), and rearranging terms, this equation has the form

$$\begin{aligned} \frac{\partial u}{\partial t} + \left(\frac{b}{2\Delta x \Delta y} \right) u_{i-1,j-1} + \left(\frac{1}{\Delta y^2} \left[c - \frac{b\Delta y}{2\Delta x} \right] - \frac{e}{2\Delta y} \right) u_{i,j-1} \\ + \left(\frac{1}{\Delta x^2} \left[a - \frac{b\Delta x}{2\Delta y} \right] - \frac{d}{2\Delta x} \right) u_{i-1,j} \\ + \left(-\frac{2}{\Delta x^2} \left[a - \frac{b\Delta x}{2\Delta y} \right] - \frac{2}{\Delta y^2} \left[c - \frac{b\Delta y}{2\Delta x} \right] - \frac{b}{\Delta x \Delta y} + f \right) u_{i,j} \\ + \left(\frac{1}{\Delta x^2} \left[a - \frac{b\Delta x}{2\Delta y} \right] + \frac{d}{2\Delta x} \right) u_{i+1,j} \\ + \left(\frac{1}{\Delta y^2} \left[c - \frac{b\Delta y}{2\Delta x} \right] + \frac{e}{2\Delta y} \right) u_{i,j+1} + \left(\frac{b}{2\Delta x \Delta y} \right) u_{i+1,j+1} = 0. \end{aligned} \quad (22)$$

Finally, we apply this finite difference scheme for the generalised Black-Scholes partial differential operator (3). After replacing the coefficients of (14) by the coefficients

of the generalised Black-Scholes equation, the space discretization scheme reads

$$\begin{aligned}
\frac{\partial u}{\partial t} &- \left(\frac{\rho\gamma}{2\Delta x\Delta y} x_i y_j \right) u_{i-1,j-1} - \left(\frac{1}{\Delta y^2} \left[\frac{1}{2} \gamma^2 y_j - \frac{\rho\gamma\Delta y}{2\Delta x} x_i y_j \right] - \frac{\alpha(\beta - y_j)}{2\Delta y} \right) u_{i,j-1} \\
&- \left(\frac{1}{\Delta x^2} \left[\frac{1}{2} x_i^2 y_j - \frac{\rho\gamma\Delta x}{2\Delta y} x_i y_j \right] - \frac{r}{2\Delta x} x_i \right) u_{i-1,j} \\
&+ \left(\frac{2}{\Delta x^2} \left[\frac{1}{2} x_i^2 y_j - \frac{\rho\gamma\Delta x}{2\Delta y} x_i y_j \right] + \frac{2}{\Delta y^2} \left[\frac{1}{2} \gamma^2 y_j - \frac{\rho\gamma\Delta y}{2\Delta x} x_i y_j \right] + \frac{\rho\gamma}{\Delta x\Delta y} x_i y_j + r \right) u_{i,j} \\
&- \left(\frac{1}{\Delta x^2} \left[\frac{1}{2} x_i^2 y_j - \frac{\rho\gamma\Delta x}{2\Delta y} x_i y_j \right] + \frac{r}{2\Delta x} x_i \right) u_{i+1,j} \\
&- \left(\frac{1}{\Delta y^2} \left[\frac{1}{2} \gamma^2 y_j - \frac{\rho\gamma\Delta y}{2\Delta x} x_i y_j \right] + \frac{\alpha(\beta - y_j)}{2\Delta y} \right) u_{i,j+1} - \left(\frac{\rho\gamma}{2\Delta x\Delta y} x_i y_j \right) u_{i+1,j+1} = 0,
\end{aligned} \tag{23}$$

for $i = 1, \dots, m-1$ and $j = 1, \dots, n-1$. This defines a seven point discretization stencil for the generalised Black-Scholes equation.

The Dirichlet boundary conditions (6) and (7) are posed on the boundaries $x = 0$ and $y = 0$, respectively. These can be implemented in a straightforward manner with the finite difference stencil (23). Next, we describe the treatment of the Neumann boundary conditions (8) and (9). Let us consider the boundary $x = X$. At a grid point (m, j) , $j = 1, \dots, n$, the boundary condition is $\partial u(X, y_j, t)/\partial x = 0$. We approximate it using the central finite difference operator in (15) and we obtain

$$\delta_x u_{m,j} = \frac{u_{m+1,j} - u_{m-1,j}}{2\Delta x} = 0. \tag{24}$$

From this it follows that the fictitious grid point value $u_{m+1,j}$ outside the computational domain has to be the same as the grid point value $u_{m-1,j}$. We can use this knowledge to eliminate all fictitious grid point values $u_{m+1,j}$, $j = 1, \dots, n$, appearing in the stencil (23) when it is used on the boundary $x = X$. The other Neumann boundary condition on $y = Y$ can be handled in the same way.

The space discretization leads to the semi-discrete equation which has the matrix representation

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{A} \mathbf{u} = 0, \tag{25}$$

where \mathbf{A} is an $mn \times mn$ matrix and \mathbf{u} is a vector of length mn .

For usual values of the parameters in the option pricing problems the proposed finite difference scheme does not lead \mathbf{A} to be an M -matrix, since some off-diagonal elements are positive. Also, some rows of \mathbf{A} might not be diagonally dominant. By adding a sufficient amount of artificial diffusion it would be possible to obtain an M -matrix but such considerations are outside the scope of this paper. We remark that the time discretization of the semi-discrete equation (25) increases diagonal dominance and with small enough time step the arising discretization matrix is diagonally dominant.

In the numerical experiments in Section 7 the matrices have only a few rows with positive off-diagonal elements and even fewer rows are not diagonal dominant. Still the

numerical experiments show that the discretization scheme leads to accurate solutions without oscillations. Moreover, the lack of these matrix properties does not seem to deteriorate the convergence of the iterative methods.

3.2 Time discretization

In the following we consider the discretization of the first-order time derivative in (3). The stability properties of time discretization schemes are essential in the option pricing problems, since the initial value (5) has discontinuous first derivative. We consider the implicit Euler method and three second-order accurate methods, which are the Crank-Nicolson method, the backward difference formula, and a Runge-Kutta scheme.

The first-order accurate implicit Euler scheme reads

$$(\mathbf{I} + \Delta t \mathbf{A}) \mathbf{u}^{(k+1)} = \mathbf{u}^{(k)}, \quad \text{for } k = 0, \dots, l-1. \quad (26)$$

A well known second-order accurate time discretization is the Crank-Nicolson method given by

$$\left(\mathbf{I} + \frac{1}{2}\Delta t \mathbf{A}\right) \mathbf{u}^{(k+1)} = \left(\mathbf{I} - \frac{1}{2}\Delta t \mathbf{A}\right) \mathbf{u}^{(k)}, \quad \text{for } k = 0, \dots, l-1. \quad (27)$$

This scheme is unconditionally stable which means that no restriction for the step size Δt is needed [27]. Despite of this stability property, this method can cause oscillations in numerical solutions; see [24], for example.

The second-order accurate backward difference formula (BDF2) is

$$\left(\mathbf{I} + \frac{2}{3}\Delta t \mathbf{A}\right) \mathbf{u}^{(k+1)} = \frac{4}{3}\mathbf{u}^{(k)} - \frac{1}{3}\mathbf{u}^{(k-1)}, \quad \text{for } k = 1, \dots, l-1, \quad (28)$$

where $\mathbf{u}^{(k+1)}$ is computed using the solutions at two previous time steps k and $k-1$. Hence, the solution $\mathbf{u}^{(1)}$ of the first time step has to be obtained using another method. The most typical choice for this is the implicit Euler method which we will also use in our numerical experiments. This scheme is L -stable which is a stronger property than the unconditional stability [16]. Due to this the BDF2 formula does not cause oscillations in numerical solutions like the Crank-Nicolson method. This time discretization is applied to option pricing problems in [23].

The Runge-Kutta scheme considered here consists of the following two steps:

$$\begin{cases} (\mathbf{I} + \theta \Delta t \mathbf{A}) \bar{\mathbf{u}}^{(k+1)} = (\mathbf{I} - (1 - \theta) \Delta t \mathbf{A}) \mathbf{u}^{(k)}, \\ (\mathbf{I} + \theta \Delta t \mathbf{A}) \mathbf{u}^{(k+1)} = \left(\mathbf{I} - \frac{1}{2} \Delta t \mathbf{A}\right) \mathbf{u}^{(k)} - \left(\frac{1}{2} - \theta\right) \Delta t \mathbf{A} \bar{\mathbf{u}}^{(k+1)}, \end{cases} \quad (29)$$

for $k = 0, \dots, l-1$. This scheme was proposed and shown to be L -stable in [6]. In order to obtain a second-order accuracy, the parameter θ has to be $\theta = 1 - 1/\sqrt{2}$.

4 Operator splitting methods

Once the space and the time discretizations are performed, the solution of (11) is obtained by solving the sequence of discrete linear complementarity problems

$$\begin{cases} \mathbf{B}\mathbf{u}^{(k+1)} \geq \mathbf{C}\mathbf{u}^{(k)}, \\ \mathbf{u}^{(k+1)} \geq \mathbf{g}, \\ (\mathbf{B}\mathbf{u}^{(k+1)} - \mathbf{C}\mathbf{u}^{(k)})^T (\mathbf{u}^{(k+1)} - \mathbf{g}) = 0, \end{cases} \quad (30)$$

for $k = 0, \dots, l - 1$. For the alternative formulation (12) the discrete problems read

$$\begin{cases} \mathbf{B}\mathbf{u}^{(k+1)} = \mathbf{C}\mathbf{u}^{(k)} + \Delta t \boldsymbol{\lambda}^{(k+1)}, \\ \boldsymbol{\lambda}^{(k+1)} \geq 0, \quad \mathbf{u}^{(k+1)} \geq \mathbf{g}, \\ \left(\boldsymbol{\lambda}^{(k+1)} \right)^T (\mathbf{u}^{(k+1)} - \mathbf{g}) = 0, \end{cases} \quad (31)$$

for $k = 0, \dots, l - 1$. Next, we present the operator splitting methods for solving time dependent linear complementarity problems. The following operator splitting methods are based on the formulation (31).

Our operator splitting methods have two fractional time steps. In the first fractional step a system of linear equations is solved and in the second step, an intermediate solution and the variable $\boldsymbol{\lambda}$ are updated in such a way that they satisfy their constraints. According to (31), the solution is required to satisfy the early exercise constraint of the option pricing problem while the variable $\boldsymbol{\lambda}$ is required to be positive. Since the form of the operator splitting methods depends on the underlying time discretization, we describe the operator splitting methods for all previously considered time discretizations.

The operator splitting method in the case of the implicit Euler scheme reads

$$(\mathbf{I} + \Delta t \mathbf{A}) \tilde{\mathbf{u}}^{(k+1)} = \mathbf{u}^{(k)} + \Delta t \boldsymbol{\lambda}^{(k)}, \quad (32)$$

$$\begin{cases} \mathbf{u}^{(k+1)} - \tilde{\mathbf{u}}^{(k+1)} - \Delta t \left(\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^{(k)} \right) = 0, \\ \left(\boldsymbol{\lambda}^{(k+1)} \right)^T (\mathbf{u}^{(k+1)} - \mathbf{g}) = 0, \quad \mathbf{u}^{(k+1)} \geq \mathbf{g} \quad \text{and} \quad \boldsymbol{\lambda}^{(k+1)} \geq 0, \end{cases} \quad (33)$$

for $k = 0, \dots, l - 1$. At each time step k the operator splitting method requires the solution of the system of linear equations (32) and the update for the intermediate solution $\tilde{\mathbf{u}}^{(k+1)}$ and the variable $\boldsymbol{\lambda}^{(k)}$. The update to satisfying (33) can be performed componentwise using a simple formula. The computational cost of this step is negligible compared to the cost of the solution of the system of linear equations (32) and, thus, the efficiency of the operator splitting method (32)–(33) is determined by the solution method of the system of linear equations.

Similarly the operator splitting method based on the Crank-Nicolson method is

$$\left(\mathbf{I} + \frac{1}{2}\Delta t\mathbf{A}\right)\tilde{\mathbf{u}}^{(k+1)} = \left(\mathbf{I} - \frac{1}{2}\Delta t\mathbf{A}\right)\mathbf{u}^{(k)} + \Delta t\boldsymbol{\lambda}^{(k)}, \quad (34)$$

$$\begin{cases} \mathbf{u}^{(k+1)} - \tilde{\mathbf{u}}^{(k+1)} - \Delta t\left(\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^{(k)}\right) = 0, \\ \left(\boldsymbol{\lambda}^{(k+1)}\right)^T\left(\mathbf{u}^{(k+1)} - \mathbf{g}\right) = 0, \quad \mathbf{u}^{(k+1)} \geq \mathbf{g} \quad \text{and} \quad \boldsymbol{\lambda}^{(k+1)} \geq 0, \end{cases} \quad (35)$$

for $k = 0, \dots, l-1$. The proposed operator splitting based on the BDF2 time discretization reads

$$\left(\mathbf{I} + \frac{2}{3}\Delta t\mathbf{A}\right)\tilde{\mathbf{u}}^{(k+1)} = \frac{4}{3}\mathbf{u}^{(k)} - \frac{1}{3}\mathbf{u}^{(k-1)} + \frac{2}{3}\Delta t\boldsymbol{\lambda}^{(k)}, \quad (36)$$

$$\begin{cases} \mathbf{u}^{(k+1)} - \tilde{\mathbf{u}}^{(k+1)} - \frac{2}{3}\Delta t\left(\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^{(k)}\right) = 0, \\ \left(\boldsymbol{\lambda}^{(k+1)}\right)^T\left(\mathbf{u}^{(k+1)} - \mathbf{g}\right) = 0, \quad \mathbf{u}^{(k+1)} \geq \mathbf{g} \quad \text{and} \quad \boldsymbol{\lambda}^{(k+1)} \geq 0, \end{cases} \quad (37)$$

for $k = 1, \dots, l-1$. Again, the intermediate solution $\tilde{\mathbf{u}}^{(k+1)}$ is obtained by solving (36) and the update for this solution and the $\boldsymbol{\lambda}^{(k)}$ variable are performed using equations (37).

The Runge-Kutta scheme is a two step time discretization method and, thus, the operator splitting method based on this Runge-Kutta scheme requires two solutions of the linear systems of equations. After these, the intermediate solution $\tilde{\mathbf{u}}^{(k+1)}$ and the variable $\boldsymbol{\lambda}^{(k)}$ are updated to satisfy their constraints. The splitting method is

$$\left(\mathbf{I} + \theta\Delta t\mathbf{A}\right)\bar{\mathbf{u}}^{(k+1)} = \left(\mathbf{I} - (1-\theta)\Delta t\mathbf{A}\right)\mathbf{u}^{(k)} + (1-\theta)\Delta t\boldsymbol{\lambda}^{(k)}, \quad (38)$$

$$\left(\mathbf{I} + \theta\Delta t\mathbf{A}\right)\tilde{\mathbf{u}}^{(k+1)} = \left(\mathbf{I} - \frac{1}{2}\Delta t\mathbf{A}\right)\mathbf{u}^{(k)} - \left(\frac{1}{2} - \theta\right)\Delta t\mathbf{A}\bar{\mathbf{u}}^{(k+1)} + (1-\theta)\Delta t\boldsymbol{\lambda}^{(k)}, \quad (39)$$

$$\begin{cases} \mathbf{u}^{(k+1)} - \tilde{\mathbf{u}}^{(k+1)} - \Delta t\left(\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^{(k)}\right) = 0, \\ \left(\boldsymbol{\lambda}^{(k+1)}\right)^T\left(\mathbf{u}^{(k+1)} - \mathbf{g}\right) = 0, \quad \mathbf{u}^{(k+1)} \geq \mathbf{g} \quad \text{and} \quad \boldsymbol{\lambda}^{(k+1)} \geq 0, \end{cases} \quad (40)$$

for $k = 0, \dots, l-1$.

5 Accuracy considerations for operator splitting methods

In the following we study the accuracy of the operator splitting method based on the Crank-Nicolson method. These considerations can be easily extended also for the

operator splitting method based on the implicit Euler method. We compare the solutions of discrete linear complementarity problems obtained using the Crank-Nicolson method and the operator splitting method based on it. Under some assumptions we give an estimate for this difference in Theorem 1.

For the analysis we need a version of the Crank-Nicolson method which uses the auxiliary variable λ . It is given by

$$\left(\mathbf{I} + \frac{1}{2}\Delta t\mathbf{A}\right)\mathbf{u}^{(k+1)} = \left(\mathbf{I} - \frac{1}{2}\Delta t\mathbf{A}\right)\mathbf{u}^{(k)} + \Delta t\boldsymbol{\lambda}^{(k+1)}, \quad (41)$$

$$\boldsymbol{\lambda}^{(k+1)} \geq 0, \quad \mathbf{u}^{(k+1)} \geq \mathbf{g}, \quad \left(\boldsymbol{\lambda}^{(k+1)}\right)^T (\mathbf{u}^{(k+1)} - \mathbf{g}) = 0. \quad (42)$$

The operator splitting method is the same as in (34)–(35). In order to compare the solutions, we denote the approximations for \mathbf{u} and $\boldsymbol{\lambda}$ in the operator splitting method by $\hat{\mathbf{u}}$ and $\hat{\boldsymbol{\lambda}}$, respectively. Thus, the method reads

$$\left(\mathbf{I} + \frac{1}{2}\Delta t\mathbf{A}\right)\tilde{\mathbf{u}}^{(k+1)} = \left(\mathbf{I} - \frac{1}{2}\Delta t\mathbf{A}\right)\hat{\mathbf{u}}^{(k)} + \Delta t\hat{\boldsymbol{\lambda}}^{(k)}, \quad (43)$$

$$\hat{\mathbf{u}}^{(k+1)} - \tilde{\mathbf{u}}^{(k+1)} - \Delta t\left(\hat{\boldsymbol{\lambda}}^{(k+1)} - \hat{\boldsymbol{\lambda}}^{(k)}\right) = 0, \quad (44)$$

$$\hat{\boldsymbol{\lambda}}^{(k+1)} \geq 0, \quad \hat{\mathbf{u}}^{(k+1)} \geq \mathbf{g}, \quad \left(\hat{\boldsymbol{\lambda}}^{(k+1)}\right)^T (\hat{\mathbf{u}}^{(k+1)} - \mathbf{g}) = 0. \quad (45)$$

The following lemma gives two algebraic relations for the differences of the solution vectors generated by the methods.

Lemma 1 *The vectors in the Crank-Nicolson method (41)–(42) and the operator splitting method (43)–(45) satisfy*

$$\left(\mathbf{I} + \frac{1}{2}\Delta t\mathbf{A}\right)\left(\mathbf{d}^{(1)} - \Delta t\boldsymbol{\delta}^{(1)}\right) = \left(\mathbf{I} - \frac{1}{2}\Delta t\mathbf{A}\right)\mathbf{d}^{(0)} - \frac{1}{2}(\Delta t)^2\mathbf{A}\left(\boldsymbol{\lambda}^{(1)} - \hat{\boldsymbol{\lambda}}^{(0)}\right) \quad (46)$$

and

$$\left(\mathbf{I} + \frac{1}{2}\Delta t\mathbf{A}\right)\left(\mathbf{d}^{(l)} - \Delta t\boldsymbol{\delta}^{(l)}\right) = \left(\mathbf{I} - \frac{1}{2}\Delta t\mathbf{A}\right)\mathbf{d}^{(l-1)} - \frac{1}{2}(\Delta t)^2\mathbf{A}\left(\boldsymbol{\delta}^{(l-1)} + \boldsymbol{\lambda}^{(l)} - \boldsymbol{\lambda}^{(l-1)}\right) \quad (47)$$

for $l \geq 2$, where

$$\mathbf{d}^{(k)} = \mathbf{u}^{(k)} - \hat{\mathbf{u}}^{(k)} \quad \text{and} \quad \boldsymbol{\delta}^{(k)} = \boldsymbol{\lambda}^{(k)} - \hat{\boldsymbol{\lambda}}^{(k)}. \quad (48)$$

Proof. Let the index k be an arbitrary integer such that $0 \leq k \leq l - 1$. We start by eliminating the intermediate solution $\tilde{\mathbf{u}}^{(k+1)}$ in the splitting method (43)–(45). The equation (44) gives us $\tilde{\mathbf{u}}^{(k+1)} = \hat{\mathbf{u}}^{(k+1)} - \Delta t\left(\hat{\boldsymbol{\lambda}}^{(k+1)} - \hat{\boldsymbol{\lambda}}^{(k)}\right)$ and by substituting it to (43) we get

$$\left(\mathbf{I} + \frac{1}{2}\Delta t\mathbf{A}\right)\hat{\mathbf{u}}^{(k+1)} = \left(\mathbf{I} - \frac{1}{2}\Delta t\mathbf{A}\right)\hat{\mathbf{u}}^{(k)} + \Delta t\hat{\boldsymbol{\lambda}}^{(k+1)} + \frac{1}{2}(\Delta t)^2\mathbf{A}\left(\hat{\boldsymbol{\lambda}}^{(k+1)} - \hat{\boldsymbol{\lambda}}^{(k)}\right). \quad (49)$$

By subtracting this from (41) we obtain

$$\left(\mathbf{I} + \frac{1}{2}\Delta t\mathbf{A}\right)\mathbf{d}^{(k+1)} = \left(\mathbf{I} - \frac{1}{2}\Delta t\mathbf{A}\right)\mathbf{d}^{(k)} + \Delta t\boldsymbol{\delta}^{(k+1)} - \frac{1}{2}(\Delta t)^2\mathbf{A}\left(\hat{\boldsymbol{\lambda}}^{(k+1)} - \hat{\boldsymbol{\lambda}}^{(k)}\right). \quad (50)$$

Subtracting $\left(\mathbf{I} + \frac{1}{2}\Delta t\mathbf{A}\right)\Delta t\boldsymbol{\delta}^{(k+1)}$ from both sides and using $\boldsymbol{\delta}^{(k+1)} + \hat{\boldsymbol{\lambda}}^{(k+1)} - \hat{\boldsymbol{\lambda}}^{(k)} = \boldsymbol{\lambda}^{(k+1)} - \hat{\boldsymbol{\lambda}}^{(k)}$ gives us

$$\left(\mathbf{I} + \frac{1}{2}\Delta t\mathbf{A}\right)\left(\mathbf{d}^{(k+1)} - \Delta t\boldsymbol{\delta}^{(k+1)}\right) = \left(\mathbf{I} - \frac{1}{2}\Delta t\mathbf{A}\right)\mathbf{d}^{(k)} - \frac{1}{2}(\Delta t)^2\mathbf{A}\left(\boldsymbol{\lambda}^{(k+1)} - \hat{\boldsymbol{\lambda}}^{(k)}\right). \quad (51)$$

From this we get the equation (46) by choosing $k = 0$.

By subtracting from (51) the same equation with one smaller indices ($k \rightarrow k - 1$) we get

$$\begin{aligned} & \left(\mathbf{I} + \frac{1}{2}\Delta t\mathbf{A}\right)\left[\mathbf{d}^{(k+1)} - \Delta t\boldsymbol{\delta}^{(k+1)} - \left(\mathbf{d}^{(k)} - \Delta t\boldsymbol{\delta}^{(k)}\right)\right] \\ &= \left(\mathbf{I} - \frac{1}{2}\Delta t\mathbf{A}\right)\left(\mathbf{d}^{(k)} - \mathbf{d}^{(k-1)}\right) - \frac{1}{2}(\Delta t)^2\mathbf{A}\left(\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^{(k)} - \hat{\boldsymbol{\lambda}}^{(k)} + \hat{\boldsymbol{\lambda}}^{(k-1)}\right) \end{aligned} \quad (52)$$

for $1 \leq k \leq l - 1$. Calculating the sum of these equations from $k = 1$ to $k = l - 1$ gives us

$$\begin{aligned} & \left(\mathbf{I} + \frac{1}{2}\Delta t\mathbf{A}\right)\left[\mathbf{d}^{(l)} - \Delta t\boldsymbol{\delta}^{(l)} - \left(\mathbf{d}^{(1)} - \Delta t\boldsymbol{\delta}^{(1)}\right)\right] \\ &= \left(\mathbf{I} - \frac{1}{2}\Delta t\mathbf{A}\right)\left(\mathbf{d}^{(l-1)} - \mathbf{d}^{(0)}\right) - \frac{1}{2}(\Delta t)^2\mathbf{A}\left(\boldsymbol{\lambda}^{(l)} - \boldsymbol{\lambda}^{(1)} - \hat{\boldsymbol{\lambda}}^{(l-1)} + \hat{\boldsymbol{\lambda}}^{(0)}\right). \end{aligned} \quad (53)$$

By adding the equation (46) to this and using $\boldsymbol{\lambda}^{(l)} - \hat{\boldsymbol{\lambda}}^{(l-1)} = \boldsymbol{\delta}^{(l-1)} + \boldsymbol{\lambda}^{(l)} - \boldsymbol{\lambda}^{(l-1)}$ we get the equation (47). \square

In order to make the relations in Lemma 1 to be useful, we need to be able to estimate the norms of associated vectors. Using the constraints (42) and (45) we obtain the following result.

Lemma 2 *Let $\|\cdot\|_\infty$ be the l_∞ -norm and $\mathbf{d}^{(l)}$ and $\boldsymbol{\delta}^{(l)}$ are defined as in Lemma 1. Then the equation*

$$\|\mathbf{d}^{(l)}\|_\infty + \Delta t\|\boldsymbol{\delta}^{(l)}\|_\infty = \|\mathbf{d}^{(l)} - \Delta t\boldsymbol{\delta}^{(l)}\|_\infty \quad (54)$$

holds for all $\Delta t > 0$.

Proof. We consider the i th component of the vectors. The index i belongs to one of the following four sets:

Set 1: i is such that $\lambda_i^{(l)} = \hat{\lambda}_i^{(l)} = 0$. We have $\delta_i^{(l)} = \lambda_i^{(l)} - \hat{\lambda}_i^{(l)} = 0$ and, thus, $|d_i^{(l)}| + \Delta t|\delta_i^{(l)}| = |d_i^{(l)}| = |d_i^{(l)} - \Delta t\delta_i^{(l)}|$.

Set 2: i is such that $\lambda_i^{(l)} > 0$ and $\hat{\lambda}_i^{(l)} = 0$. We have $\delta_i^{(l)} = \lambda_i^{(l)} - \hat{\lambda}_i^{(l)} = \lambda_i^{(l)} > 0$. Due to (42) and (45) we have $u_i^{(l)} = g_i$ and $\hat{u}_i^{(l)} \geq g_i$. Thus, we get $|d_i^{(l)}| + \Delta t |\delta_i^{(l)}| = -d_i^{(l)} + \Delta t \delta_i^{(l)} = |-d_i^{(l)} + \Delta t \delta_i^{(l)}| = |d_i^{(l)} - \Delta t \delta_i^{(l)}|$.

Set 3: i is such that $\lambda_i^{(l)} = 0$ and $\hat{\lambda}_i^{(l)} > 0$. We have $\delta_i^{(l)} = \lambda_i^{(l)} - \hat{\lambda}_i^{(l)} = -\hat{\lambda}_i^{(l)} < 0$. Due to (42) and (45) we have $u_i^{(l)} \geq g_i$ and $\hat{u}_i^{(l)} = g_i$. Thus, we get $|d_i^{(l)}| + \Delta t |\delta_i^{(l)}| = d_i^{(l)} - \Delta t \delta_i^{(l)} = |d_i^{(l)} - \Delta t \delta_i^{(l)}|$.

Set 4: i is such that $\lambda_i^{(l)} > 0$ and $\hat{\lambda}_i^{(l)} > 0$. Due to (42) and (45) we have $u_i^{(l)} = \hat{u}_i^{(l)} = g_i$. Thus, we get $|d_i^{(l)}| + \Delta t |\delta_i^{(l)}| = \Delta t |\delta_i^{(l)}| = |d_i^{(l)} - \Delta t \delta_i^{(l)}|$.

Hence the equation $|d_i^{(l)}| + \Delta t |\delta_i^{(l)}| = |d_i^{(l)} - \Delta t \delta_i^{(l)}|$ holds for all i . The equation (54) follows from this. \square

Using the previous lemmas we obtain the following main result which states that under given assumptions the difference between the solutions is $\mathcal{O}((\Delta t)^2)$.

Theorem 1 *Let the initial value for the operator splitting method (43)–(45) be $\hat{\mathbf{u}}^{(0)} = \mathbf{u}^{(0)}$ and $\hat{\boldsymbol{\lambda}}^{(0)} = \mathbf{0}$. Assume that the sequence $\boldsymbol{\lambda}^{(1)}, \boldsymbol{\lambda}^{(2)}, \dots, \boldsymbol{\lambda}^{(l)}$ in the Crank-Nicolson method (41)–(42) satisfies the condition*

$$\|\boldsymbol{\lambda}^{(1)}\|_\infty + \sum_{k=2}^l \|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^{(k-1)}\|_\infty \leq C_1, \quad (55)$$

where C_1 is a positive constant independent of $l = T/\Delta t$. Furthermore assume that \mathbf{A} is diagonally dominant matrix with positive diagonal entries, that is, for the entries a_{ij} of \mathbf{A} it is assumed that

$$\sum_{j \neq i} |a_{ij}| \leq a_{ii} \quad \text{for all } i. \quad (56)$$

Then the inequality

$$\|\mathbf{u}^{(l)} - \hat{\mathbf{u}}^{(l)}\|_\infty + \Delta t \|\boldsymbol{\lambda}^{(l)} - \hat{\boldsymbol{\lambda}}^{(l)}\|_\infty \leq C_2 (\Delta t)^2 \quad (57)$$

holds for $\Delta t \leq 1/(\max_i a_{ii})$ and a positive constant C_2 independent of Δt .

Proof. We begin by showing that for any vector \mathbf{x} the inequality

$$\|\mathbf{x}\|_\infty \leq \left\| \left(\mathbf{I} + \frac{1}{2} \Delta t \mathbf{A} \right) \mathbf{x} \right\|_\infty \quad (58)$$

holds. Let i be such that $|x_i| \geq |x_j|$ for all j . Due to the inequalities $a_{ii} - \sum_{j \neq i} |a_{ij}| \geq 0$, $-\sum_{j \neq i} |a_{ij}| |x_i| \leq -\sum_{j \neq i} |a_{ij} x_j|$, and $|b| - |c| \leq |b + c|$ for any real numbers b, c , we obtain

$$\begin{aligned} |x_i| &\leq |x_i| + \frac{1}{2} \Delta t \left(a_{ii} - \sum_{j \neq i} |a_{ij}| \right) |x_i| \leq |x_i| + \frac{1}{2} \Delta t \left(|a_{ii} x_i| - \sum_{j \neq i} |a_{ij} x_j| \right) \\ &\leq \left| x_i + \frac{1}{2} \Delta t a_{ii} x_i \right| - \frac{1}{2} \Delta t \left| \sum_{j \neq i} a_{ij} x_j \right| \leq \left| x_i + \frac{1}{2} \Delta t \sum_j a_{ij} x_j \right|. \end{aligned} \quad (59)$$

From this the inequality (58) follows.

Since $\hat{\mathbf{u}}^{(0)} = \mathbf{u}^{(0)}$, we have $\mathbf{d}^{(0)} = 0$. Due to this and $\hat{\boldsymbol{\lambda}}^{(0)} = 0$ Lemma 1 gives us

$$\left(\mathbf{I} + \frac{1}{2}\Delta t\mathbf{A}\right) \left(\mathbf{d}^{(1)} - \Delta t\boldsymbol{\delta}^{(1)}\right) = -\frac{1}{2}(\Delta t)^2\mathbf{A}\boldsymbol{\lambda}^{(1)}. \quad (60)$$

From this we get the inequality

$$\|\mathbf{d}^{(1)} - \Delta t\boldsymbol{\delta}^{(1)}\|_\infty \leq \left\| \left(\mathbf{I} + \frac{1}{2}\Delta t\mathbf{A}\right) \left(\mathbf{d}^{(1)} - \Delta t\boldsymbol{\delta}^{(1)}\right) \right\|_\infty \leq \frac{1}{2}(\Delta t)^2\|\mathbf{A}\|_\infty\|\boldsymbol{\lambda}^{(1)}\|_\infty, \quad (61)$$

where $\|\mathbf{A}\|_\infty = \max_i \sum_j |a_{ij}|$.

Due to (56) and $\Delta t \leq 1/(\max_i a_{ii})$ we obtain the inequality

$$\frac{1}{2}\Delta t\|\mathbf{A}\|_\infty = \frac{1}{2}\Delta t \max_i \sum_j |a_{ij}| \leq \frac{1}{2}\Delta t \max_i 2a_{ii} = \Delta t \max_i a_{ii} \leq \frac{1}{\max_i a_{ii}} \max_i a_{ii} = 1. \quad (62)$$

The inequality $1 - \frac{1}{2}\Delta ta_{ii} \geq 0$ follows from $\Delta t \leq 1/(\max_i a_{ii})$ and $a_{ii} - \sum_{j \neq i} |a_{ij}| \geq 0$ follows from (56). Using these we get the inequality

$$\begin{aligned} \left\| \mathbf{I} - \frac{1}{2}\Delta t\mathbf{A} \right\|_\infty &= \max_i \left(\left| 1 - \frac{1}{2}\Delta ta_{ii} \right| + \frac{1}{2}\Delta t \sum_{j \neq i} |a_{ij}| \right) \\ &= \max_i \left(1 - \frac{1}{2}\Delta ta_{ii} + \frac{1}{2}\Delta t \sum_{j \neq i} |a_{ij}| \right) \\ &= \max_i \left[1 - \frac{1}{2}\Delta t \left(a_{ii} - \sum_{j \neq i} |a_{ij}| \right) \right] \leq 1. \end{aligned} \quad (63)$$

Let k be an integer such that $k \geq 2$. Using Lemma 1, the previous inequalities (62),(63), and Lemma 2 we get

$$\begin{aligned} \|\mathbf{d}^{(k)} - \Delta t\boldsymbol{\delta}^{(k)}\|_\infty &\leq \left\| \left(\mathbf{I} + \frac{1}{2}\Delta t\mathbf{A}\right) \left(\mathbf{d}^{(k)} - \Delta t\boldsymbol{\delta}^{(k)}\right) \right\|_\infty \\ &\leq \left\| \left(\mathbf{I} - \frac{1}{2}\Delta t\mathbf{A}\right) \mathbf{d}^{(k-1)} - \frac{1}{2}(\Delta t)^2\mathbf{A} \left(\boldsymbol{\delta}^{(k-1)} + \boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^{(k-1)}\right) \right\|_\infty \\ &\leq \left\| \mathbf{I} - \frac{1}{2}\Delta t\mathbf{A} \right\|_\infty \|\mathbf{d}^{(k-1)}\|_\infty + \frac{1}{2}\Delta t\|\mathbf{A}\|_\infty\Delta t\|\boldsymbol{\delta}^{(k-1)}\|_\infty \\ &\quad + \frac{1}{2}(\Delta t)^2\|\mathbf{A}\|_\infty\|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^{(k-1)}\|_\infty \\ &\leq \|\mathbf{d}^{(k-1)}\|_\infty + \Delta t\|\boldsymbol{\delta}^{(k-1)}\|_\infty + \frac{1}{2}(\Delta t)^2\|\mathbf{A}\|_\infty\|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^{(k-1)}\|_\infty \\ &= \|\mathbf{d}^{(k-1)} - \Delta t\boldsymbol{\delta}^{(k-1)}\|_\infty + \frac{1}{2}(\Delta t)^2\|\mathbf{A}\|_\infty\|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^{(k-1)}\|_\infty. \end{aligned} \quad (64)$$

Choosing $k = l$ we have

$$\|\mathbf{d}^{(l)} - \Delta t \boldsymbol{\delta}^{(l)}\|_\infty \leq \|\mathbf{d}^{(l-1)} - \Delta t \boldsymbol{\delta}^{(l-1)}\|_\infty + \frac{1}{2}(\Delta t)^2 \|\mathbf{A}\|_\infty \|\boldsymbol{\lambda}^{(l)} - \boldsymbol{\lambda}^{(l-1)}\|_\infty. \quad (65)$$

Using (64) from $k = l - 1$ to $k = 2$ and then (61) we get

$$\|\mathbf{d}^{(l)} - \Delta t \boldsymbol{\delta}^{(l)}\|_\infty \leq \frac{1}{2} \|\mathbf{A}\|_\infty (\Delta t)^2 \left(\|\boldsymbol{\lambda}^{(1)}\|_\infty + \sum_{k=2}^l \|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^{(k-1)}\|_\infty \right). \quad (66)$$

The assumption (55) gives us

$$\|\mathbf{d}^{(l)} - \Delta t \boldsymbol{\delta}^{(l)}\|_\infty \leq \frac{1}{2} \|\mathbf{A}\|_\infty C_1 (\Delta t)^2. \quad (67)$$

By choosing $C_2 = \frac{1}{2} \|\mathbf{A}\|_\infty C_1$ and using Lemma 2 we get the inequality (57). \square

Remark 1 It is easy to show that the assumption (55) in Theorem 1 is fulfilled in the case that $\|\boldsymbol{\lambda}^{(k)}\|_\infty$ is bounded for all k and the sequence is $\boldsymbol{\lambda}^{(1)}, \boldsymbol{\lambda}^{(2)}, \dots$, is increasing or decreasing componentwise. Based on our numerical investigations $\boldsymbol{\lambda}^{(k)}$ s seem to be bounded and they are decreasing componentwise in the problems considered in Section 7.

Remark 2 Our space discretization does not lead to a diagonal dominant matrix \mathbf{A} and, thus, the assumption (56) is not satisfied. Nevertheless in practice the convergence might be as good as in Theorem 1. In the numerical experiments in Section 7 the lack of diagonal dominance does not seem to deteriorate the convergence.

Under the assumptions in Theorem 1 the accuracy of the operator splitting method based on the Crank-Nicolson method has the same order of accuracy as the Crank-Nicolson method which is at most $\mathcal{O}((\Delta t)^2)$. The possible irregularity of the solution with respect to time might reduce these both orders of accuracy. For one-dimensional American option pricing problems this reduction has been studied in [12], for example.

6 Multigrid

Multigrid methods are scalable iterative methods in the sense that their convergence does not deteriorate when the grid step size decreases. For example the convergence of the (projected) SOR method slows down as the grid step size is made smaller. Here, we describe only briefly the multigrid method which we have used in the numerical experiments. For details we refer to the books [5], [15], [25], [26].

The basic idea of multigrid methods is to reduce low frequency error using coarser grids while high frequency error can be typically reduced using a standard iterative method on the fine grid. A multigrid method uses a restriction operation to transfer the solution or residual to a coarser grid. We have chosen to use the full weighting restriction. For coarser grids we construct matrices by discretizing the generalized

Black-Scholes operator on them. The solution or correction to it is transferred to a finer grid using a prolongation (interpolation) operation. We use the bilinear prolongation operator.

In a multigrid the iterative method which reduces the high frequency error is called a smoother. For the Laplace operator, the (point) Gauss-Seidel or SOR method can be used as a smoother. The generalized Black-Scholes operator requires a more involved smoother. Clarke and Parrott used a x -line Gauss-Seidel type smoother in their multigrid [7]. Oosterlee performed Fourier analysis for smoothers for the generalized Black-Scholes operator [23]. According to the analysis the point Gauss-Seidel method is not good smoother while the x -line Gauss-Seidel smoother is efficient when the grid is fine in the y -direction. An alternating direction line Gauss-Seidel method was found out to have good smoothing properties for all grids. Thus, we chose to use the alternating direction line Gauss-Seidel smoother.

One cycle of the multigrid method is much more expensive than one (P)SOR iteration. In the alternating direction line Gauss-Seidel smoother, the x -line and y -line Gauss-Seidel sweeps can be both considered to be comparable in computational cost to a matrix vector multiplication. We use one presmooth iteration before moving to a coarser grid and one postsmooth iteration after adding the correction from the coarser grid. Thus, the computational cost of the smoothing is similar to four matrix vector multiplications. The multigrid also performs two matrix vector multiplication in order to compute the residual twice during one cycle. Thus, one cycle requires six matrix vector multiplication type operations on the finest grid. By summing operations over all grids we get that one multigrid cycle corresponds approximately eight matrix vector multiplications. Since one (P)SOR iteration is comparable to a matrix vector multiplication, one multigrid cycle is roughly eight times more expensive than one (P)SOR iteration. On the other hand a large number of SOR iterations are required to reduce the residual by the same amount as one multigrid cycle does.

7 Numerical experiments

In the following experiments, we study the properties of the operator splitting methods numerically. The prices of American put options are computed using the model (11) with the parameter values:

$$\alpha = 5.0, \quad \beta = 0.16, \quad \gamma = 0.9, \quad \rho = 0.1, \quad r = 0.1, \quad \text{and} \quad E = 10.0 \quad (68)$$

in the computational domain

$$[0, X] \times [0, Y] \times [0, T] = [0, 20] \times [0, 1] \times [0, 0.25]. \quad (69)$$

In order to compare computed option prices to the reference values, we chose the parameter values (68) to be the same as in [7], [23], [28]. Furthermore, the computational domain is the same as in [23].

In the first numerical experiment, we compare the option prices obtained by an operator splitting method with the prices computed using the PSOR method and the

method	(m, n, l)	asset value				
		8.0	9.0	10.0	11.0	12.0
OS	(80,32,16)	2.00000	1.10442	0.50736	0.20446	0.07900
	(160,64,32)	2.00000	1.10695	0.51651	0.21106	0.08121
	(320,128,64)	2.00000	1.10751	0.51904	0.21294	0.08181
PSOR	(80,32,16)	2.00000	1.10435	0.50755	0.20462	0.07909
	(160,64,32)	2.00000	1.10675	0.51653	0.21112	0.08125
	(320,128,64)	2.00000	1.10738	0.51902	0.21296	0.08183
ref. [7]		2.0000	1.1080	0.5316	0.2261	0.0907
ref. [23]		2.00	1.107	0.517	0.212	0.0815
ref. [28]		2.0000	1.1076	0.5202	0.2138	0.0821

Table 1: Option prices for five different asset values at $y = 0.0625$.

prices presented in the references. The time convergence rates of four operator splitting methods are studied in the second experiment, and in the last numerical experiment, we compare the efficiency of the operator splitting algorithm with the multigrid method to the efficiency of the PSOR method.

7.1 Option prices computed using the operator splitting method

We computed the prices of the American put options using the operator splitting (OS) method (38)–(40). These prices are presented in Tables 1 and 2 for the asset values $x = 8.0, 9.0, 10.0, 11.0, 12.0$, and for the variance values $y = 0.0625$ and $y = 0.25$. We have used three different discretization grids in order to study the accuracy of numerical solutions. In the tables the grids are defined by the triplets (m, n, l) where m, n and l are the numbers of steps in the x -direction, in the y -direction, and in the t -direction, respectively. The arising systems of linear equations in the operator splitting method were solved by the SOR method.

For the comparison, the option prices computed using the PSOR method are also given in these tables as well as the prices reported in [7], [23], [28]. As one can see from Tables 1 and 2, the operator splitting method produces prices which are in good agreement with the prices obtained by the PSOR method. Only one price differs at the third decimal while other prices have three, four or even five same decimals. Furthermore, the prices obtained with the finest grid are fairly similar to the ones given in [23], [28].

7.2 Time convergence

Next, we study the time convergence rates of the operator splitting methods based on the implicit Euler scheme, the Crank-Nicolson method, the backward difference formula, and the Runge-Kutta scheme. The same American put option pricing problem

method	(m, n, l)	asset value				
		8.0	9.0	10.0	11.0	12.0
OS	(80,32,16)	2.07798	1.33205	0.79378	0.44634	0.24157
	(160,64,32)	2.07845	1.33333	0.79543	0.44775	0.24246
	(320,128,64)	2.07846	1.33360	0.79585	0.44813	0.24271
PSOR	(80,32,16)	2.07744	1.33192	0.79388	0.44650	0.24170
	(160,64,32)	2.07811	1.33317	0.79541	0.44779	0.24250
	(320,128,64)	2.07829	1.33350	0.79581	0.44813	0.24272
ref. [7]		2.0733	1.3290	0.7992	0.4536	0.2502
ref. [23]		2.079	1.334	0.796	0.449	0.243
ref. [28]		2.0784	1.3337	0.7961	0.4483	0.2428

Table 2: Option prices for five different asset values at $y = 0.25$.

is considered as before. In this experiment we compare the errors and the convergence rates obtained with the operator splitting methods and with the PSOR method. Moreover, we have computed the errors and the convergence rates also for the explicit payoff method.

The spatial grid was chosen to be $(m, n) = (80, 32)$ while the number of time steps varies from 4 to 256. The errors reported in the following are the absolute maximum differences between the computed numerical solutions and the reference solution. Our reference solution was computed with the PSOR method using the Runge-Kutta time discretization and with the grid defined by $(m, n, l) = (80, 32, 2048)$. In Tables 3 and 4, we report the maximum errors over the whole computational domain. The associated convergence rates in the tables were calculated by dividing the error of the previous coarser time discretization by the error of the current discretization. In these convergence tables the PSOR method with the implicit Euler scheme is referred as PSOR-IE and the operator splitting method based on the implicit Euler scheme is referred as OS-IE, etc.

Similar errors can be observed for the operator splitting method and the PSOR method for all time discretizations. Actually, for many discretizations the errors for the operator splitting method are smaller. This suggest that the splitting error has the opposite sign to the error of the time discretization method which leads to the cancellation of a part of the error. The errors for all time step sizes are smaller for the PSOR method only in the case of the Runge-Kutta time discretization.

A first-order convergence rate can be observed for the implicit Euler scheme and the operator splitting method based on it. The second-order accurate time discretization schemes and the operator splitting methods based on them cannot maintain a second-order convergence rate when the number of time steps is increased. This suggest that the solution might not be not regular enough with respect to time to obtain a second-order convergence. For a small number of time steps, the operator splitting methods based on the backward difference formula and the Runge-Kutta scheme lead the most

l	PSOR-IE		PSOR-CN		PSOR-BDF2		PSOR-RK	
	error	ratio	error	ratio	error	ratio	error	ratio
4	6.15884E-2		1.11281E-1		1.25347E-2		1.71188E-3	
8	3.19686E-2	1.93	4.47692E-2	2.49	3.16622E-3	3.95	1.03731E-3	1.65
16	1.64541E-2	1.94	8.68625E-3	5.15	1.00550E-3	3.15	3.16830E-4	3.27
32	8.41289E-3	1.96	1.47103E-4	59.05	4.09668E-4	2.45	7.18950E-5	4.41
64	4.28124E-3	1.97	3.16917E-5	4.64	1.66863E-4	2.46	3.27586E-5	2.19
128	2.16936E-3	1.97	1.60005E-5	1.98	6.69046E-5	2.49	1.48919E-5	2.20
256	1.09595E-3	1.98	7.79188E-6	2.05	2.67999E-5	2.50	7.12583E-6	2.09

Table 3: Maximum errors and time convergence rates for the PSOR method.

l	OS-IE		OS-CN		OS-BDF2		OS-RK	
	error	ratio	error	ratio	error	ratio	error	ratio
4	6.20757E-2		1.11083E-1		1.76833E-2		2.58122E-2	
8	3.14090E-2	1.98	4.42568E-2	2.51	4.56060E-3	3.88	8.25943E-3	3.13
16	1.59583E-2	1.97	8.32375E-3	5.32	1.45107E-3	3.14	2.32439E-3	3.55
32	8.11941E-3	1.97	5.03790E-4	16.52	4.66273E-4	3.11	5.52405E-4	4.21
64	4.13625E-3	1.96	1.13952E-4	4.42	1.37406E-4	3.39	2.16980E-4	2.55
128	2.11098E-3	1.96	3.18526E-5	3.58	3.81111E-5	3.61	4.56701E-5	4.75
256	1.07588E-3	1.96	7.83804E-6	4.06	1.57298E-5	2.42	1.07787E-5	4.24

Table 4: Maximum errors and time convergence rates for the operator splitting methods.

accurate numerical solutions. The Crank-Nicolson time discretization method is only unconditionally stable while other schemes are L -stable. This is one reason why the convergences of the Crank-Nicolson method and the operator splitting method based on it are quite erratic.

In order to study the efficiency of the operator splitting methods, we computed also the convergence rates for the explicit payoff (EP) method. This method is considered in the case of one-dimensional American option pricing problem in [9], [11], [19], for example. In this method at each time step the system of linear equations arising from the discretization of the PDE is solved and then the maximum between the solution and the payoff function is taken componentwise. In the operator splitting method and in the explicit payoff method a system of linear equations is solved and a simple update is performed at each time step and, hence, the computational cost of these methods are essentially the same. According to Tables 4 and 5, the proposed operator splitting methods are more accurate than the explicit payoff methods. Only first-order convergence rates can be observed with the explicit payoff method even when the second-order accurate time discretization schemes are used. According to Table 4, the operator splitting methods lead to higher convergence rates with the second-order accurate time discretization schemes.

l	EP-IE		EP-CN		EP-BDF2		EP-RK	
	error	ratio	error	ratio	error	ratio	error	ratio
4	7.16377E-2		1.17374E-1		2.82341E-2		1.27054E-2	
8	3.74640E-2	1.91	4.79576E-2	2.45	1.11258E-2	2.54	5.94883E-3	2.14
16	1.94058E-2	1.93	1.03447E-2	4.64	5.35422E-3	2.08	2.50590E-3	2.37
32	9.97900E-3	1.94	2.27315E-3	4.55	2.73577E-3	1.96	1.44624E-3	1.73
64	5.10040E-3	1.96	1.17174E-3	1.94	1.52171E-3	1.80	7.03254E-4	2.06
128	2.59306E-3	1.97	5.98495E-4	1.96	7.99009E-4	1.90	3.50692E-4	2.01
256	1.31282E-3	1.97	2.98776E-4	2.00	3.99815E-4	2.00	1.74749E-4	2.01

Table 5: Maximum errors and time convergence rates for the explicit payoff method.

In Figure 1, we have plotted some contour lines of the absolute errors in the computational domain $[0, 20] \times [0, 1]$ for the operator splitting method and the PSOR method. These error plots are for the numerical solutions computed using the grid defined by $(m, n, l) = (80, 32, 16)$ and the errors are computed with respect to the previously mentioned reference solution. The contour plots are given for the operator splitting methods based on the Crank-Nicolson and Runge-Kutta discretizations. It can be observed that the contour lines are quite similar for the operator splitting method and the PSOR method. The methods based on the Runge-Kutta discretization lead to quite smooth errors while the error for the methods based on the Crank-Nicolson discretization are much more oscillatory. It can be also seen that the splitting error cancels a part of the time discretization error.

7.3 CPU-time comparison

In our last numerical experiment, we compare the efficiency of the PSOR method and the operator splitting approach. We report the CPU-times required to solve the previously described American put option pricing problem on a J5600 HP workstation when different discretization grids are used. The time discretization is based on the Runge-Kutta scheme.

For the PSOR method the stopping criterion and the relaxation parameter have to be chosen. The iteration was stopped when the norm of an approximate residual was reduced by the factor of 10^{-5} . After some numerical tests, we decided that this criterion is sufficient for all discretization grids. The error due to the termination of iterations is much smaller than the discretization error. On the other hand, the criterion is not unnecessary tight which would increase the CPU-time without any benefit. The relaxation parameter ω was optimised for the different grids because the choice of ω has significant effect to the efficiency of the PSOR method. For example the value $\omega = 1.5$ for the grid $(m, n, l) = (320, 128, 64)$ leads the CPU-time to be 58.6 seconds while the choice $\omega = 1.8$ halves this computational time. These parameters and also the average numbers of iterations are presented in Table 6.

With the operator splitting method any efficient solution method for the system of linear equations can be used. In this experiment we employ the multigrid (MG)

(m, n, l)	PSOR				OS with MG		
	error	iter ave	ω	CPU	error	iter ave	CPU
(40,16,16)	8.076E-3	8.3	1.3	0.03	8.183E-3	1.1	0.03
(40,32,16)	6.645E-3	9.2	1.4	0.07	6.720E-3	2.0	0.09
(80,16,16)	3.562E-3	16.5	1.6	0.11	3.689E-3	2.0	0.09
(80,32,16)	2.098E-3	16.0	1.6	0.22	2.192E-3	2.0	0.19
(80,32,32)	2.028E-3	9.8	1.5	0.27	1.999E-3	1.7	0.28
(80,64,32)	1.716E-3	12.5	1.5	0.70	1.686E-3	2.0	0.73
(160,32,32)	8.757E-4	19.6	1.7	1.06	8.553E-4	2.0	0.74
(160,64,32)	5.680E-4	19.8	1.7	2.22	5.416E-4	2.0	1.70
(160,64,64)	5.198E-4	12.4	1.6	2.86	4.841E-4	1.7	2.73
(160,128,64)	4.535E-4	16.3	1.7	7.91	4.132E-4	2.0	6.57
(320,64,64)	2.292E-4	25.1	1.8	11.75	1.916E-4	2.0	6.68
(320,128,64)	1.623E-4	25.7	1.8	25.63	1.226E-4	2.0	13.73
(320,128,128)	1.383E-4	15.9	1.7	32.38	1.165E-4	1.4	18.42
(320,256,128)	1.218E-4	22.7	1.8	93.50	1.158E-4	1.3	35.94
(640,128,128)	6.480E-5	37.4	1.8	147.10	3.989E-5	1.7	45.32
(640,256,128)	4.859E-5	41.2	1.8	323.32	2.521E-5	1.8	102.32

Table 6: CPU-times for the PSOR method and the operator splitting method.

grid is (640, 256, 128). For coarser grids the CPU-times are quite similar for these two solution methods. The good performance of the PSOR method is due to optimised relaxation parameter ω and strongly diagonally dominant matrices leading to faster convergence. Based on the presented CPU-times, we see that the operator splitting method with the multigrid method requires much less time when finer discretization grids are used.

8 Conclusion

We have considered operator splitting methods for the numerical pricing of the American options when Heston's stochastic volatility model is used. In these splitting methods each time step is divided into two fractional steps. In the first step a system of linear equations is solved while in the second step the constraint due to the early exercise possibility of the American option is enforced by performing a simple update. With this approach any efficient solution method for a system of linear equations can be applied without modifications.

Our analysis suggest that the Crank-Nicolson method with the implicit treatment of complementarity conditions (for example, the PSOR method) and the operator splitting method based on it have the same order of accuracy. In the numerical experiments the accuracy of solutions obtained using any of the tested four implicit time discretizations with the PSOR method and the corresponding operator splitting method were similar. Thus, the splitting procedure did not essentially increase the error. Also, the

computed prices were in good agreement with the prices published in the literature. The time convergences of the Crank-Nicolson method and the operator splitting based on it were somewhat erratic while the time convergences of the BDF2 formula, the Runge-Kutta method and the splitting methods based on them were more steady. In our experiments none of the second-order time discretizations or the operator splitting methods based on them could maintain a second-order time convergence with uniform time steps. The explicit treatment of complementarity conditions in the explicit payoff methods leads to reduced accuracy and first-order time convergence. The operator splitting method with a multigrid method for solving the systems of linear equations was faster than the PSOR method. For finer grids the operator splitting approach was several times faster while the difference was smaller for coarser grids.

The analysis of regularity of the solutions and its impact on the accuracy of discretizations is one possible topic of future research. For example, it could be useful to employ with Heston's stochastic volatility model a time step selector like the one used in [12]. Also, one could consider applying similar operator splitting methods for pricing more complicated American options.

Acknowledgements

This research has been supported by the University of Jyväskylä and by the Academy of Finland, grant #53588. We thank Dr. Roland Glowinski for introduction and helpful discussions on operator splitting methods. We also thank Dr. Peter Forsyth for fruitful discussions on option pricing and suggestions to improve this paper.

References

- [1] C. A. Ball and A. Roma. Stochastic volatility option pricing. *Journal of Financial and Quantitative Analysis*, 29:589–607, 1994.
- [2] F. Black and M. Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy*, 81:637–654, 1973.
- [3] A. Brandt and C. W. Cryer. Multigrid algorithms for the solution of linear complementarity problems arising from free boundary problems. *SIAM Journal on Scientific and Statistical Computing*, 4:655–684, 1983.
- [4] M. J. Brennan and E. S. Schwartz. The valuation of American put options. *Journal of Finance*, 32:449–462, 1977.
- [5] W. L. Briggs, V. E. Henson, and S. F. McCormick. *A multigrid tutorial*. SIAM, Philadelphia, PA, second edition, 2000.
- [6] J. R. Cash. Two new finite difference schemes for parabolic equations. *SIAM Journal on Numerical Analysis*, 21:433–446, 1984.

- [7] N. Clarke and K. Parrott. The multigrid solution of two-factor American put options. Technical Report Res. Report 96-16, Oxford Comp. Lab, Oxford, 1996.
- [8] N. Clarke and K. Parrott. Multigrid for American option pricing with stochastic volatility. *Applied Mathematical Finance*, 6:177–195, 1999.
- [9] T. F. Coleman, Li. Yuying, and V. Arun. A Newton method for American option pricing. *Journal of Computational Finance*, 5(3), 2002.
- [10] A. D. Drăgulescu and V. M. Yakovenko. Probability distribution of returns in the Heston model with stochastic volatility. *Quantitative Finance*, 2:443–453, 2002.
- [11] D. Duffie. *Dynamic Asset Pricing Theory*. Princeton University Press, Princeton, NJ, second edition, 1996.
- [12] P. A. Forsyth and K. R. Vetzal. Quadratic convergence for valuing American options using a penalty method. *SIAM Journal on Scientific Computing*, 23:2095–2122, 2002.
- [13] J.-P. Fouque, G. Papanicolaou, and K. R. Sircar. *Derivatives in financial markets with stochastic volatility*. Cambridge University Press, Cambridge, 2000.
- [14] Roland Glowinski. *Finite element methods for incompressible viscous flow*, volume IX of *Handbook of Numerical Analysis*. North-Holland, Amsterdam, 2003.
- [15] W. Hackbusch. *Multigrid methods and applications*, volume 4 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1985.
- [16] E. Hairer and G. Wanner. *Solving ordinary differential equations. II: Stiff and differential-algebraic problems*, volume 14 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 1996.
- [17] S. Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Review of Financial Studies*, 6:327–343, 1993.
- [18] J. Hull and A. White. The pricing of options on assets with stochastic volatilities. *Journal of Finance*, 42:281–300, 1987.
- [19] J. C. Hull. *Options, Futures, and Other Derivatives*. Prentice Hall, Upper Saddle River, NJ, third edition, 1997.
- [20] S. Ikonen and J. Toivanen. Operator splitting methods for American option pricing. *Applied Mathematical Letters*, 17:809–814, 2004.
- [21] P.-L. Lions and B. Mercier. Splitting algorithms for the sum of two nonlinear operators. *SIAM Journal on Numerical Analysis*, 16:964–979, 1979.
- [22] A. R. Mitchell and D. F. Griffiths. *The finite difference method in partial differential equations*. John Wiley & Sons Ltd., Chichester, 1980.

- [23] C. W. Oosterlee. On multigrid for linear complementarity problems with application to American-style options. *Electronic Transactions on Numerical Analysis*, 15:165–185, 2003.
- [24] D. M. Pooley, K. R. Vetzal, and P. A. Forsyth. Convergence remedies for non-smooth payoffs in option pricing. *Journal of Computational Finance*, 6:25–40, 2003.
- [25] U. Trottenberg, C. W. Oosterlee, and A. Schüller. *Multigrid*. Academic Press Inc., San Diego, CA, 2001.
- [26] P. Wesseling. *An introduction to multigrid methods*. John Wiley & Sons Ltd., Chichester, 1992.
- [27] P. Wilmott, S. Howison, and J. Dewynne. *The mathematics of financial derivatives*. Cambridge University Press, Cambridge, 1995.
- [28] R. Zvan, P. A. Forsyth, and K. R. Vetzal. Penalty methods for American options with stochastic volatility. *Journal of Computational and Applied Mathematics*, 91:199–218, 1998.