Universiteit Antwerpen

This item is the archived preprint of:

Operator splitting schemes for the two-asset Merton jump-diffusion model

Operator splitting schemes for the two-asset Merton jump-diffusion model

Lynn Boen* and Karel J. in 't Hout*

Dedicated to the memory of Willem Hundsdorfer

January 15, 2019

Abstract

This paper deals with the numerical solution of the two-dimensional time-dependent Merton partial integro-differential equation (PIDE) for the values of rainbow options under the two-asset Merton jump-diffusion model. Key features of this well-known equation are a two-dimensional nonlocal integral part and a mixed spatial derivative term. For its efficient and stable numerical solution, we study seven recent and novel operator splitting schemes of the implicit-explicit (IMEX) and the alternating direction implicit (ADI) kind. Here the integral part is always conveniently treated in an explicit fashion. The convergence behaviour and the relative performance of the seven schemes are investigated in ample numerical experiments for both European put-on-the-min and put-on-the-average options.

1 Introduction

The inherent risk diversification of multi-name or rainbow options, for which the payoff depends on multiple underlying asset prices, has made them increasingly popular over the recent years. Margrabe [31] proposed such a rainbow option, which allows to exchange one asset for another, and derived a valuation formula under the Black–Scholes framework. The term rainbow option was later introduced by Rubinstein [35], who referred to the number of underlying assets as the number of colors of the option. The development of fast, accurate and stable methods for obtaining the fair values of options forms a central topic in the field of computational finance. The fair value of an option is equal to its expected discounted payoff under a risk-neutral measure. Acquiring a semi-closed analytical valuation formula, as for example Black & Scholes [5] and Margrabe [31] did, would be most favorable, but this is often not possible. Accordingly, there is a strong demand for reliable numerical methods that can efficiently approximate fair option values.

For the numerical approximation of the fair value of an option, three main approaches are considered in the literature up to now. Firstly, Monte Carlo methods estimate the expected discounted payoff value by computing sample means. The second way is by numerical integration, which is employed in for example the Carr–Madan method [9] and the COS method by Fang & Oosterlee [13]. The third main approach is to numerically solve a time-dependent partial differential equation (PDE) that holds for the option value. This approach was introduced in computational finance

---

*Department of Mathematics and Computer Science, University of Antwerp, Middelheimlaan 1, B-2020 Antwerp, Belgium. Email: {lynn.boen,karel.inthout}@uantwerpen.be.

1

by Schwartz [38], who considered a finite difference discretization of the celebrated Black–Scholes PDE. The development, analysis and application of numerical methods for time-dependent PDEs has since then been widely studied in the literature for valuing many types of options under a broad range of asset pricing models, see for example [21, 39, 41].

It is well-known in financial markets that sudden, large changes in asset prices can occur. In view of this common phenomenon, Merton [32] added a jump term to the original stochastic process for the asset price assumed by Black & Scholes [5], yielding a so-called jump-diffusion model. Under such an advanced asset price process, the option value satisfies a one-dimensional time-dependent *partial integro-differential equation* (PIDE). The integral part, which stems from the jump term, runs over the whole asset price domain, and hence, is nonlocal.

In the present paper we consider the extension of the prominent Merton jump-diffusion model to two assets where the jumps in the asset prices occur simultaneously. This yields a two-dimensional time-dependent PIDE for the values of two-asset options. The efficient and stable numerical solution of this equation poses two principal challenges: (a) an effective treatment of the two-dimensional PDE part, which involves a mixed derivative term, and (b) an effective treatment of the two-dimensional integral part, which is nonlocal.

We follow the well-known and versatile method of lines (MOL) approach [19], whereby the PIDE is first discretized in the spatial variables and subsequently in the temporal variable. Spatial discretization of the integral part gives rise to a very large, dense matrix that is computationally expensive to work with. However, by a change of variables, the double integral can be turned into a two-dimensional cross-correlation. Spatial discretization of this new integral version leads to a matrix having a block-Toeplitz structure for which matrix-vector products can be computed efficiently using the fast Fourier transform (FFT). For one-dimensional PIDEs, the application of FFT in approximating the integral part has been advocated in [1, 2, 12]. In particular, Almendral & Oosterlee [1] observed that the matrix corresponding to the jump term is Toeplitz and can be embedded in a circulant matrix, so that matrix-vector products can be computed exactly by means of FFT. In [10, 37] the FFT technique has next been applied in the case of two-dimensional PIDEs with a two-dimensional integral part. Salmi, Toivanen & von Sydow [37] considered a useful embedding of a block-Toeplitz matrix into a block-circulant one. Here we shall employ a similar idea, which was introduced in a different context in Barrowes, Teixeira & Kong [3].

After spatial discretization of the two-dimensional Merton PIDE one arrives at a very large, stiff system of ordinary differential equations (ODEs). The application of common implicit temporal discretization methods, such as the Crank–Nicolson scheme, requires the solution of very large, dense linear systems in every time step. The main aim of the present paper is to study a variety of recent and novel operator splitting schemes that effectively overcome this.

For one-dimensional option valuation PIDEs, Cont & Voltchkova [11] proposed an *implicit-explicit* (IMEX) Euler time-stepping scheme, where the PDE part is treated implicitly and the integral part explicitly. This IMEX Euler scheme is only first-order consistent. Various authors subsequently considered higher-order IMEX schemes for PIDEs in finance, see for example [8, 14, 30]. Salmi & Toivanen [36] recommended the IMEX CNAB scheme, where the PDE part is handled in a Crank–Nicolson manner and the integral part is treated in a two-step Adams–Bashforth fashion. This second-order two-step IMEX scheme has next been applied to two-dimensional option valuation PIDEs in for example [23, 37].

An alternative approach was proposed by Tavella & Randall [39] and d'Halluin, Forsyth & Vetzal [12]. These authors considered the Crank–Nicolson scheme together with a fixed-point iteration on the integral part for the solution of the linear system in each time step. This approach has been applied to two-dimensional option valuation PIDEs in Clift & Forsyth [10], including the two-dimensional Merton PIDE. It is found that in general two to three iterations are sufficient to reach

a satisfactory error level. When the number of fixed-point iterations is frozen, a particular one-step IMEX scheme is obtained, see Section 4 below.

For multidimensional option valuation PDEs (without integral part), operator splitting schemes are widely considered in the literature. Notably, *alternating direction implicit* (ADI) methods, which combine implicit unidirectional correction stages with explicit stages involving the mixed derivative terms, are highly effective, see for example [16, 21, 23, 25, 40]. Hundsdorfer & in 't Hout [18] recently studied a novel class of multistep stabilizing correction methods for PDEs with mixed derivative terms and showed that such methods can be competitive with the well-established one-step modified Craig–Sneyd (MCS) scheme [25] in the application to the familiar two-dimensional Heston PDE [17].

In the past years, ADI methods have been extended and applied to two-dimensional PIDEs in finance. In [28] Strang splitting was employed between the PDE and integral parts and an ADI scheme was used for the PDE subproblems, see also [27]. The direct extension of ADI methods to PIDEs has recently been considered in [24, 29]. Here the implicit unidirectional stages are combined with explicit stages involving both the mixed derivative term and the integral part. In 't Hout & Toivanen [24] defined three adaptations of the MCS scheme to two-dimensional PIDEs that are all second-order consistent. They found that the adaptation where the integral part is treated in a two-step Adams–Bashforth fashion is preferred in the application to the well-known two-dimensional Bates PIDE [4], which merges the Heston PDE and the one-dimensional Merton PIDE.

An outline of the rest of this paper is as follows. Section 2 describes the asset price dynamics under the two-asset Merton jump-diffusion model and introduces the two-dimensional PIDE that holds for the value of a rainbow option under this model. Section 3 deals with the first step in the MOL approach, the spatial discretization of the two-dimensional Merton PIDE. Here particular attention is given to a suitable discretization of the double integral part, which enables application of the FFT algorithm from [3]. Section 4 concerns the second step in the MOL approach, the temporal discretization of the semidiscretized PIDE. We formulate seven recent and novel operator splitting schemes of both the IMEX and the ADI kind, which all treat the integral part in an explicit fashion. In Section 5 extensive numerical experiments are presented. Here two types of rainbow options are considered: a European put-on-the-min option and a European put-on-the-average option. For all seven splitting schemes, the behaviour of the temporal discretization error is investigated in detail. Moreover, employing the recent semi-closed analytical formula derived in Boen [6], also the total discretization error is studied in the case of the put-on-the-min option. The final Section 6 gives conclusions.

2 The two-asset Merton jump-diffusion model

Let $(\Omega, \mathcal{F}, \{\mathcal{F}_\tau\}, \mathbb{Q})$ be a probability measure space, with $\mathbb{Q}$ a risk-neutral measure. Under a two-asset jump-diffusion model, the asset prices under $\mathbb{Q}$ are given by

$$S_\tau^{(1)} = S_0^{(1)} \exp\left((r - \tfrac{1}{2}\sigma_1^2 - \lambda\kappa_1)\tau + \sigma_1 W_\tau^{(1)} + \sum_{k=1}^{N_\tau} Y_k^{(1)} \right),$$

$$S_\tau^{(2)} = S_0^{(2)} \exp\left((r - \tfrac{1}{2}\sigma_2^2 - \lambda\kappa_2)\tau + \sigma_2 W_\tau^{(2)} + \sum_{k=1}^{N_\tau} Y_k^{(2)} \right),$$

where

- r is the risk-free interest rate,

3

- σ_i $(i = 1, 2)$ is the instantaneous volatility of asset i, conditional on the event that no jumps occur,

- $W_\tau = (W_\tau^{(1)}, W_\tau^{(2)})^\top$ consists of two correlated standard Brownian motions, with correlation coefficient ρ,

- λ is the jump intensity of the Poisson arrival process $N = \{N_\tau, \tau \geqslant 0\}$,

- $Y_k^{(i)}$ $(k = 1, 2, 3, \ldots)$ are independent jump sizes that have an identical distribution to the random jump size $Y^{(i)}$ and, for any given k, the jumps $Y_k^{(i)}$ $(i = 1, 2)$ occur simultaneously, driven by the same Poisson arrival process N, and are correlated with correlation coefficient $\widehat{\rho}$,

- κ_i is the expected relative jump size

$$\kappa_i = \mathbb{E}_\mathbb{Q}[e^{Y^{(i)}} - 1] \quad (i = 1, 2).$$

The W_τ, N_τ and $Y = (Y^{(1)}, Y^{(2)})^\top$ are assumed to be independent of each other whenever $\tau \geqslant 0$.

In the two-dimensional version of the Merton model [32], the random vector Y is bivariate normally distributed with mean $(\gamma_1, \gamma_2)^\top$ and covariance matrix

$$\Sigma_Y = \begin{pmatrix} \delta_1^2 & \widehat{\rho}\delta_1\delta_2 \\ \widehat{\rho}\delta_1\delta_2 & \delta_2^2 \end{pmatrix},$$

where δ_i^2 is the variance of $Y^{(i)}$ $(i = 1, 2)$. The moment generating function of Y is then given by

$$M_{\mathbb{Q},Y}(u) = \mathbb{E}_\mathbb{Q}\left[e^{u^\top Y}\right] = \exp\left(\gamma_1 u_1 + \gamma_2 u_2 + \tfrac{1}{2}\left(\delta_1^2 u_1^2 + 2\widehat{\rho}\delta_1\delta_2 u_1 u_2 + \delta_2^2 u_2^2\right)\right), \quad u = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix},$$

and the expected relative jump size is

$$\kappa_i = e^{\gamma_i + \frac{1}{2}\delta_i^2} - 1 \quad (i = 1, 2).$$

Under the two-asset Merton jump-diffusion model, the value $v = v(s_1, s_2, t)$ of a European-style[1] option with maturity date $T > 0$ and s_i $(i = 1, 2)$ representing the price of asset i at time $\tau = T - t$, satisfies the following PIDE [10]:

$$\frac{\partial v}{\partial t} = \tfrac{1}{2}\sigma_1^2 s_1^2 \frac{\partial^2 v}{\partial s_1^2} + \rho\sigma_1\sigma_2 s_1 s_2 \frac{\partial^2 v}{\partial s_1 \partial s_2} + \tfrac{1}{2}\sigma_2^2 s_2^2 \frac{\partial^2 v}{\partial s_2^2} + (r - \lambda\kappa_1)s_1 \frac{\partial v}{\partial s_1} + (r - \lambda\kappa_2)s_2 \frac{\partial v}{\partial s_2}$$

$$- (r + \lambda)v + \lambda \int_0^\infty \int_0^\infty v(s_1 y_1, s_2 y_2, t) f(y_1, y_2)\mathrm{d}y_1\mathrm{d}y_2 \tag{2.1}$$

whenever $s_1 > 0$, $s_2 > 0$, $0 < t \leqslant T$. Here

$$f(y_1, y_2) = \frac{1}{2\pi\delta_1\delta_2\sqrt{1 - \widehat{\rho}^2}\, y_1 y_2} \exp\left(-\frac{\left(\frac{\ln(y_1) - \gamma_1}{\delta_1}\right)^2 + \left(\frac{\ln(y_2) - \gamma_2}{\delta_2}\right)^2 - 2\widehat{\rho}\left(\frac{\ln(y_1) - \gamma_1}{\delta_1}\right)\left(\frac{\ln(y_2) - \gamma_2}{\delta_2}\right)}{2(1 - \widehat{\rho}^2)}\right)$$

is the probability density function of a bivariate lognormal distribution. The initial condition for (2.1) is given by

$$v(s_1, s_2, 0) = \phi(s_1, s_2),$$

where ϕ denotes the payoff function of the option, that is, the option value at expiry. Further, as the boundary condition, it holds that the PIDE (2.1) is itself fulfilled on the two sides $s_1 = 0$ and $s_2 = 0$, respectively.

---

[1] A European-style option can only be exercised at the maturity date T.

3 Spatial discretization

For the numerical solution of the initial-boundary value problem for (2.1) we consider the well-known and versatile method of lines (MOL) approach, which consists of two consecutive steps [19]: first the PIDE (2.1) is discretized in space and subsequently in time. This section discusses the spatial discretization. In the next section we turn to the temporal discretization.

3.1 Convection-diffusion-reaction part

To render the numerical solution of (2.1) feasible, the spatial domain is truncated to a bounded set $[0, S_{\max}] \times [0, S_{\max}]$ with fixed value $S_{\max}$ taken sufficiently large. On the two far sides $s_1 = S_{\max}$ and $s_2 = S_{\max}$ a linear boundary condition is imposed, which is common in finance,

$$\frac{\partial^2 v}{\partial s_1^2} = 0 \ \text{ (if } s_1 = S_{\max}) \quad \text{and} \quad \frac{\partial^2 v}{\partial s_2^2} = 0 \ \text{ (if } s_2 = S_{\max}). \tag{3.1}$$

In this subsection we describe the finite difference discretization of the convection-diffusion-reaction part of (2.1), given by

$$\mathcal{D}v = \tfrac{1}{2}\sigma_1^2 s_1^2 \frac{\partial^2 v}{\partial s_1^2} + \rho\sigma_1\sigma_2 s_1 s_2 \frac{\partial^2 v}{\partial s_1 \partial s_2} + \tfrac{1}{2}\sigma_2^2 s_2^2 \frac{\partial^2 v}{\partial s_2^2} + (r - \lambda\kappa_1)s_1\frac{\partial v}{\partial s_1} + (r - \lambda\kappa_2)s_2\frac{\partial v}{\partial s_2} - (r + \lambda)v.$$

The option value function v will be approximated at a nonuniform, Cartesian set of spatial grid points,

$$(s_{1,i}, s_{2,j}) \in [0, S_{\max}] \times [0, S_{\max}] \quad (0 \leqslant i \leqslant m_1, \ 0 \leqslant j \leqslant m_2), \tag{3.2}$$

with $s_{1,0} = s_{2,0} = 0$ and $s_{1,m_1} = s_{2,m_2} = S_{\max}$. The nonuniform grid in each spatial direction is chosen such that relatively many grid points lie in a region of financial and/or numerical interest. It is generated by applying a smooth transformation to an artificial uniform grid. In Section 5 more details shall be given of the specific type of spatial grids that is used.

Let $0 = s_0 < s_1 < \cdots < s_m = S_{\max}$ be any given smooth, nonuniform, unidirectional spatial grid with mesh widths $h_i = s_i - s_{i-1}$ $(1 \leqslant i \leqslant m)$ and let $u : [0, S_{\max}] \to \mathbb{R}$ be any given smooth function. The following second-order central finite difference formulas are employed for the approximation of the first and second derivatives of u:

$$u'(s_i) \approx \omega_{i,-1} u(s_{i-1}) + \omega_{i,0} u(s_i) + \omega_{i,1} u(s_{i+1}),$$

with

$$\omega_{i,-1} = \frac{-h_{i+1}}{h_i(h_i + h_{i+1})}, \quad \omega_{i,0} = \frac{h_{i+1} - h_i}{h_i h_{i+1}}, \quad \omega_{i,1} = \frac{h_i}{h_{i+1}(h_i + h_{i+1})}$$

and

$$u''(s_i) \approx \omega_{i,-1} u(s_{i-1}) + \omega_{i,0} u(s_i) + \omega_{i,1} u(s_{i+1}),$$

with

$$\omega_{i,-1} = \frac{2}{h_i(h_i + h_{i+1})}, \quad \omega_{i,0} = \frac{-2}{h_i h_{i+1}}, \quad \omega_{i,1} = \frac{2}{h_{i+1}(h_i + h_{i+1})}.$$

for $1 \leqslant i \leqslant m-1$. If $i = 0$, no finite difference formulas are required in view of the degeneracy of (2.1) at the zero boundaries. If $i = m$, the first derivative is approximated by the first-order backward finite difference formula and the second derivative vanishes by the linear boundary condition (3.1).

Concerning the mixed derivative term in (2.1), this is discretized by successively applying the relevant finite difference formulas for the first derivative in the two spatial directions.

Let the vector $V(t) = (V_{0,0}(t), V_{1,0}(t), \ldots, V_{m_1-1,m_2}(t), V_{m_1,m_2}(t))^\top$ where entry $V_{i,j}(t)$ denotes the semidiscrete approximation to $v(s_{1,i}, s_{2,j}, t)$ whenever $0 \leqslant i \leqslant m_1$, $0 \leqslant j \leqslant m_2$. The semidiscrete version of the convection-diffusion-reaction part $\mathcal{D}v$ can then be written as

$$A^{(D)}V(t)$$

with matrix

$$A^{(D)} = A^{(M)} + A_1 + A_2,$$

where

$$A^{(M)} = \rho\sigma_1\sigma_2 \left(X_2 D_2^{(1)}\right) \otimes \left(X_1 D_1^{(1)}\right),$$

$$A_1 = I_2 \otimes \left(\tfrac{1}{2}\sigma_1^2 X_1^2 D_1^{(2)} + (r - \lambda\kappa_1)X_1 D_1^{(1)} - \tfrac{1}{2}(r + \lambda)I_1\right),$$

$$A_2 = \left(\tfrac{1}{2}\sigma_2^2 X_2^2 D_2^{(2)} + (r - \lambda\kappa_2)X_2 D_2^{(1)} - \tfrac{1}{2}(r + \lambda)I_2\right) \otimes I_1.$$

Here I_k, X_k, $D_k^{(l)}$ are given $(m_k + 1) \times (m_k + 1)$ matrices for $k, l \in \{1, 2\}$, where I_k is the identity matrix, X_k is the diagonal matrix

$$X_k = \text{diag}(s_{k,0}, s_{k,1}, \ldots, s_{k,m_k})$$

and $D_k^{(l)}$ is the matrix representing numerical differentiation of order l in the k-th spatial direction by the relevant finite difference formula above. The matrix $A^{(M)}$ corresponds to the mixed derivative term in (2.1) and A_k corresponds to all derivative terms in the k-th spatial direction ($k = 1, 2$). Further, the reaction term has been distributed equally across A_1 and A_2.

3.2   Integral part

In order to discretize the integral part in the PIDE (2.1), we apply a well-known transformation to the log-price variable. Let $x_i = \ln(s_i)$ be the log-price of the i-th asset and $\eta_i = \ln(y_i)$ ($i = 1, 2$), then

$$\mathcal{J}(s_1, s_2) := \lambda \int_0^\infty \int_0^\infty v(s_1 y_1, s_2 y_2, t)f(y_1, y_2)\mathrm{d}y_1\mathrm{d}y_2$$

$$= \lambda \int_{-\infty}^\infty \int_{-\infty}^\infty \overline{v}(x_1 + \eta_1, x_2 + \eta_2, t)\overline{f}(\eta_1, \eta_2)\mathrm{d}\eta_1\mathrm{d}\eta_2$$

$$= \lambda \int_{-\infty}^\infty \int_{-\infty}^\infty \overline{v}(\xi_1, \xi_2, t)\overline{f}(\xi_1 - x_1, \xi_2 - x_2)\mathrm{d}\xi_1\mathrm{d}\xi_2,$$

where $\overline{v}(\xi_1, \xi_2, t) = v(e^{\xi_1}, e^{\xi_2}, t)$ and $\overline{f}(\eta_1, \eta_2) = f(e^{\eta_1}, e^{\eta_2})e^{\eta_1}e^{\eta_2}$ is the probability density function of a bivariate normal distribution. Since $\overline{f}$ is a real-valued function, it is clear that the double integral in the log-price variable can be viewed as a two-dimensional cross-correlation (see e.g. [7]):

$$\overline{\mathcal{J}}(x_1, x_2) := \lambda \int_{-\infty}^\infty \int_{-\infty}^\infty \overline{v}(\xi_1, \xi_2, t)\overline{f}(\xi_1 - x_1, \xi_2 - x_2)\mathrm{d}\xi_1\mathrm{d}\xi_2 = \lambda(\overline{v} \star \overline{f})(x_1, x_2),$$

where $\star\star$ denotes the two-dimensional cross-correlation operator and $\overline{\mathcal{J}}(x_1, x_2) = \mathcal{J}(e^{x_1}, e^{x_2})$. For the semidiscretization, we restrict the log-price domain to $[-X_{\max}, X_{\max}] \times [-X_{\max}, X_{\max}]$

with $X_{\max} = \ln(S_{\max})$ and consider a $M_1 \times M_2$ uniform grid $(x_{1,k}, x_{2,l}) = (k\Delta x_1, l\Delta x_2)$ for $k = -\widetilde{M}_1 + 1, \ldots, \widetilde{M}_1$, $l = -\widetilde{M}_2 + 1, \ldots, \widetilde{M}_2$, where $\widetilde{M}_i = M_i/2$, $\widetilde{M}_i \Delta x_i = X_{\max}$ and M_i is chosen to be a power of 2 such that the mesh width Δx_i is smaller than the smallest mesh width in the nonuniform $\ln(s_i)$-grid ($i = 1, 2$). The double integral $\overline{\mathcal{J}}(x_{1,k}, x_{2,l})$ is approximated by

$$\overline{J}_{k,l} = \lambda \sum_{j=-\widetilde{M}_2+1}^{\widetilde{M}_2} \sum_{i=-\widetilde{M}_1+1}^{\widetilde{M}_1} \overline{V}_{i,j} \overline{F}_{i-k,j-l},$$

with approximation $\overline{V}_{i,j} \approx \overline{v}(x_{1,i}, x_{2,j}, t)$ defined below and $\overline{F}_{i-k,j-l} = \overline{f}_{i-k,j-l} \Delta x_1 \Delta x_2$ where

$$\overline{f}_{i-k,j-l} = \overline{f}(x_{1,i} - x_{1,k}, x_{2,j} - x_{2,l}).$$

Let $E(M, d)$ denote the $M \times M$ matrix with ones on the d-th off-diagonal and zeros elsewhere. The matrix

$$\overline{A}^{(J)} = \lambda \overline{F}$$

with

$$\overline{F} = \sum_{d=-M_2+1}^{M_2-1} E(M_2, d) \otimes \overline{F}_d^1$$

and

$$\overline{F}_d^1 = \sum_{c=-M_1+1}^{M_1-1} E(M_1, c) \overline{F}_{c,d}$$

is then an asymmetrical two-level block-Toeplitz matrix of size $(M_1 M_2) \times (M_1 M_2)$. Its structure is illustrated in Appendix A. Let the vectors

$$\overline{J} = (\overline{J}_{-\widetilde{M}_1+1, -\widetilde{M}_2+1}, \overline{J}_{-\widetilde{M}_1+2, -\widetilde{M}_2+1}, \ldots, \overline{J}_{\widetilde{M}_1-1, \widetilde{M}_2}, \overline{J}_{\widetilde{M}_1, \widetilde{M}_2})^\top,$$

$$\overline{V} = (\overline{V}_{-\widetilde{M}_1+1, -\widetilde{M}_2+1}, \overline{V}_{-\widetilde{M}_1+2, -\widetilde{M}_2+1}, \ldots, \overline{V}_{\widetilde{M}_1-1, \widetilde{M}_2}, \overline{V}_{\widetilde{M}_1, \widetilde{M}_2})^\top,$$

then

$$\overline{J} = \overline{A}^{(J)} \overline{V}.$$

This matrix-vector product can be computed very efficiently using the FFT algorithm for matrix-vector multiplications with asymmetric multilevel block-Toeplitz matrices, introduced in Barrowes, Teixeira & Kong [3]. This fast algorithm computes the matrix-vector product using two FFTs and one inverse FFT (IFFT), reducing the computational cost to $\mathcal{O}(M_1 M_2 \log(M_1 M_2))$. The algorithm from [3] essentially embeds the block-Toeplitz matrix in a circulant matrix before applying the FFT, conveniently avoiding any wrap-around effect, which would have been present when applying the FFT directly to the block-Toeplitz matrix. Further, we mention that the algorithm from [3] is more efficient than that employed in [37], as the former uses only one-dimensional FFTs, whereas the latter requires two-dimensional FFTs.

Note that the spatial grid on which we discretize the PIDE (2.1) is nonuniform, and in general does not lead to a uniform log-price grid. We therefore (bi)linearly interpolate the option value approximations between the two grids directly before and after application of the algorithm of [3]. Linearly interpolating V to $\overline{V}$ (using Lagrange basis functions on the s-grid, as in e.g. [12]), yields

$$\begin{aligned}
\overline{V}_{i,j} &= \phi_{\Pi(i)} \psi_{\Omega(j)} V_{\Pi(i), \Omega(j)} + (1 - \phi_{\Pi(i)}) \psi_{\Omega(j)} V_{\Pi(i)+1, \Omega(j)} \\
&\quad + \phi_{\Pi(i)} (1 - \psi_{\Omega(j)}) V_{\Pi(i), \Omega(j)+1} + (1 - \phi_{\Pi(i)})(1 - \psi_{\Omega(j)}) V_{\Pi(i)+1, \Omega(j)+1},
\end{aligned} \tag{3.3}$$

whenever $s_{1,\Pi(i)} \leqslant e^{x_{1,i}} \leqslant s_{1,\Pi(i)+1}$ and $s_{2,\Omega(j)} \leqslant e^{x_{2,j}} \leqslant s_{1,\Omega(j)+1}$, where $\phi_{\Pi(i)}, \psi_{\Omega(j)} \in [0,1]$ are interpolation weights . This can be written as

$$\overline{V} = \mathcal{X}V,$$

where the matrix $\mathcal{X}$ contains the interpolation weights such that (3.3) holds. Similarly, linearly interpolating $\overline{J}$ back to the original s-grid, leads to approximations $J_{k,l}$ of $\mathcal{J}(s_{1,k}, s_{2,l})$ given by

$$\begin{aligned}
J_{k,l} = &\overline{\phi}_{\Lambda(k)}\overline{\psi}_{\Upsilon(l)}\overline{J}_{\Lambda(k),\Upsilon(l)} + (1-\overline{\phi}_{\Lambda(k)})\overline{\psi}_{\Upsilon(l)}\overline{J}_{\Lambda(k)+1,\Upsilon(l)} \\
&+ \overline{\phi}_{\Lambda(k)}(1-\overline{\psi}_{\Upsilon(l)})\overline{J}_{\Lambda(k),\Upsilon(l)+1} + (1-\overline{\phi}_{\Lambda(k)})(1-\overline{\psi}_{\Upsilon(l)})\overline{J}_{\Lambda(k)+1,\Upsilon(l)+1},
\end{aligned} \tag{3.4}$$

whenever $e^{x_{1,\Lambda(k)}} \leqslant s_{1,k} \leqslant e^{x_{1,\Lambda(k)+1}}$ and $e^{x_{2,\Upsilon(l)}} \leqslant s_{2,l} \leqslant e^{x_{2,\Upsilon(l)+1}}$, where $\overline{\phi}_{\Lambda(k)}, \overline{\psi}_{\Upsilon(l)} \in [0,1]$ are interpolation weights. This can be written as

$$J = \overline{\mathcal{X}}\,\overline{J},$$

where the matrix $\overline{\mathcal{X}}$ contains the interpolation weights such that (3.4) holds. Consequently, we have

$$J = \overline{\mathcal{X}}\,\overline{J} = \overline{\mathcal{X}}\,\overline{A}^{(J)}\overline{V} = \overline{\mathcal{X}}\,\overline{A}^{(J)}\mathcal{X}V.$$

The entire procedure is summarized in Algorithm 1.

Algorithm 1 Computing the approximation to the integral part in (2.1)

---

1. Linearly interpolate $V(t) = (V_{0,0}(t), V_{1,0}(t), \ldots, V_{m_1-1,m_2}(t), V_{m_1,m_2}(t))^\top$ onto the uniform log-price grid, leading to the vector $\overline{V}$.

2. Extend the log-price grid $(x_{1,k}, x_{2,l})$ for $k = -\widetilde{M}_1 + 1, \ldots, \widetilde{M}_1$, $l = -\widetilde{M}_2 + 1, \ldots, \widetilde{M}_2$ on both sides, keeping it uniform:

$$x_{1,-M_1+1}, \ldots, x_{1,-\widetilde{M}_1}, \ldots, x_{1,0}, \ldots, x_{1,\widetilde{M}_1}, \ldots, x_{1,M_1-1},$$

$$x_{2,-M_2+1}, \ldots, x_{2,-\widetilde{M}_2}, \ldots, x_{2,0}, \ldots, x_{2,\widetilde{M}_2}, \ldots, x_{2,M_2-1},$$

and compute $\overline{F}_{i-k,j-l}$ for $i, k = -\widetilde{M}_1 + 1, \ldots, \widetilde{M}_1$, $j, l = -\widetilde{M}_2 + 1, \ldots, \widetilde{M}_2$.

3. Compute the matrix-vector product $\overline{A}^{(J)}\overline{V}$ using the fast algorithm of [3], resulting in the vector $\overline{J}$.

4. Linearly interpolate $\overline{J}$ back to the original nonuniform grid $(s_{1,i}, s_{2,j})$ for $i = 1, \ldots m_1$, $j = 1, \ldots m_2$, leading to the vector J.

Note that in the interpolation step from the uniform log-price grid to the nonuniform price grid, we obtain values $J_{i,j}$ for $i = 1, \ldots, m_1$, $j = 1, \ldots, m_2$. On the boundaries $s_1 = 0$ and $s_2 = 0$, the two-dimensional Merton PIDE (2.1) reduces to the one-dimensional Merton PIDE and the jump part can be computed using a similar FFT technique (see f.i. [12]).

Denoting the approximation of $\mathcal{J}$ on the full grid (3.2) by $A^{(J)}V(t)$, we arrive at the following semidiscrete system of ODEs:

$$V'(t) = AV(t), \tag{3.5}$$

for $0 < t \leqslant T$, where $A = A^{(D)} + A^{(J)} = A^{(M)} + A_1 + A_2 + A^{(J)}$.

The maximum norm of the matrix $A^{(J)}$ can be bounded independently of the spatial grid by a moderate constant whenever λ is moderate. Accordingly, the dense matrix $A^{(J)}$ constitutes a nonstiff part of the semidiscrete system, whereas the sparse matrix $A^{(D)}$ constitutes a stiff part.

4 Temporal discretization

For the temporal discretization of the semidiscrete system (3.5) we shall investigate seven operator splitting schemes. Each of these schemes conveniently treats the nonstiff integral part, given by the matrix $A^{(J)}$, in an explicit fashion.

Let integer $N \geqslant 1$ be given and step size $\Delta t = T/N$. Each of the following schemes defines an approximation V^n to $V(t_n)$ at the temporal grid point $t_n = n\Delta t$ successively for $n = 1, 2, \ldots, N$. The initial vector $V(0) = V^0$ is determined by the payoff function and will be specified in Section 5.

1. *Crank–Nicolson Forward Euler (CNFE) scheme*:

   In this basic method the convection-diffusion-reaction part, including the mixed derivative, is handled implicitly using the Crank–Nicolson scheme and the integral part is treated explicitly in a forward Euler manner:

   $$\left(I - \tfrac{1}{2}\Delta t A^{(D)}\right) V^n = \left(I + \tfrac{1}{2}\Delta t A^{(D)}\right) V^{n-1} + \Delta t A^{(J)} V^{n-1}. \tag{4.1}$$

   Method (4.1) is a one-step IMEX scheme. Due to the application of forward Euler, its order[2] is just equal to one.

2. *Crank–Nicolson scheme with fixed-point iteration (CNFI)*:

   In Tavella & Randall [39] and d'Halluin, Forsyth & Vetzal [12] the Crank–Nicolson scheme has been proposed with a fixed-point iteration on the integral part:

   $$\left(I - \tfrac{1}{2}\Delta t A^{(D)}\right) Y_k = \left(I + \tfrac{1}{2}\Delta t A^{(D)}\right) V^{n-1} + \tfrac{1}{2}\Delta t A^{(J)}(Y_{k-1} + V^{n-1}) \tag{4.2}$$

   for $k = 1, 2, \ldots, l$ and $V^n = Y_l$ using the starting value $Y_0 = V^{n-1}$. Clearly, with just one iteration, method (4.1) is obtained. Method (4.2) can be applied with a dynamic convergence criterion. Numerical experiments in [10, 12] reveal that this generally leads to $l = 2$ or $l = 3$ iterations. For comparable computational work to the other schemes, formulated below, we choose here a fixed number of $l = 2$ iterations. Then method (4.2) can be regarded as a one-step IMEX scheme, where the integral part is treated by the explicit trapezoidal rule, also called the modified Euler method. It can be verified, by Taylor expansion, that its order is equal to two.

3. *Implicit-Explicit Trapezoidal Rule (IETR)*:

   A different combination of the implicit trapezoidal rule (Crank–Nicolson) for the convection-diffusion-reaction part and its explicit variant for the integral part is considered by in 't Hout [21]:

   $$\begin{cases} Y_0 = V^{n-1} + \Delta t \left(A^{(D)} + A^{(J)}\right) V^{n-1}, \\ \widehat{Y}_0 = Y_0 + \tfrac{1}{2}\Delta t A^{(J)}\left(Y_0 - V^{n-1}\right), \\ Y_1 = \widehat{Y}_0 + \tfrac{1}{2}\Delta t A^{(D)}\left(Y_1 - V^{n-1}\right), \\ V^n = Y_1. \end{cases} \tag{4.3}$$

   Here the first two internal stages Y_0, $\widehat{Y}_0$ are explicit, whereas the third internal stage Y_1 is implicit. Method (4.3) is also a one-step IMEX scheme with order equal to two. This method is already well-known in the literature on the numerical solution of PDEs, without integral part, see e.g. Hundsdorfer & Verwer [19].

---

[2]This refers to the classical order of consistency, that is, for fixed nonstiff ODEs.

4. *Crank–Nicolson Adams–Bashforth (CNAB) scheme*:

   In the CNAB method, the convection-diffusion-reaction part is again treated by the Crank–Nicolson scheme, but the integral part is now handled in a two-step Adams–Bashforth fashion:

   $$\left(I - \tfrac{1}{2}\Delta t A^{(D)}\right) V^n = \left(I + \tfrac{1}{2}\Delta t A^{(D)}\right) V^{n-1} + \tfrac{1}{2}\Delta t A^{(J)}(3V^{n-1} - V^{n-2}). \qquad (4.4)$$

   Method (4.4) forms a two-step IMEX scheme and is of order equal to two. This method has first been studied for the application to PIDEs in Salmi & Toivanen [36] and Salmi, Toivanen & von Sydow [37]. Previously, it has been investigated for the numerical solution of PDEs in e.g. [15, 19, 20].

It is well-known that the nonsmoothness of the initial (payoff) function can have an adverse effect on the convergence behaviour of the Crank–Nicolson scheme, which can be resolved by first applying, at initial time $t = 0$, two time steps with step size $\Delta t/2$ using the implicit Euler method to define the approximation V^1 to $V(t_1)$, see e.g. [21, 33, 34]. In the present application to PIDEs, the implicit Euler method can be computationally demanding. We therefore apply, for all four schemes above, the IMEX Euler scheme as the starting method:

$$\left(I - \tfrac{1}{2}\Delta t A^{(D)}\right) V^{\frac{1}{2}} = V^0 + \tfrac{1}{2}\Delta t A^{(J)} V^0,$$

$$\left(I - \tfrac{1}{2}\Delta t A^{(D)}\right) V^1 = V^{\frac{1}{2}} + \tfrac{1}{2}\Delta t A^{(J)} V^{\frac{1}{2}}.$$

In the following, three recent operator splitting schemes for PIDEs are formulated that employ a subsequent, useful splitting of the two-dimensional convection-diffusion-reaction part, given by the matrix $A^{(D)}$.

5. *One-step adaptation of the modified Craig–Sneyd (MCS) scheme*:

   This method is a direct adaptation of a well-established one-step ADI scheme:

   $$\begin{cases} Y_0 = V^{n-1} + \Delta t\,(A^{(D)} + A^{(J)})V^{n-1}, \\ Y_j = Y_{j-1} + \theta\Delta t A_j(Y_j - V^{n-1}) \quad (j = 1, 2), \\ \widehat{Y}_0 = Y_0 + \theta\Delta t\,(A^{(M)} + A^{(J)})(Y_2 - V^{n-1}), \\ \widetilde{Y}_0 = \widehat{Y}_0 + (\tfrac{1}{2} - \theta)\Delta t\,(A^{(D)} + A^{(J)})(Y_2 - V^{n-1}), \\ \widetilde{Y}_j = \widetilde{Y}_{j-1} + \theta\Delta t A_j(\widetilde{Y}_j - V^{n-1}) \quad (j = 1, 2), \\ V^n = \widetilde{Y}_2, \end{cases} \qquad (4.5)$$

   where $\theta > 0$ is a given parameter. The MCS scheme was introduced by in 't Hout & Welfert [25] for the numerical solution of PDEs with mixed derivative terms. The above, direct adaptation to PIDEs has recently been studied in in 't Hout & Toivanen [24]. Method (4.5) is of order two for any value θ. Here we make the common choice $\theta = \tfrac{1}{3}$, which is motivated by stability and accuracy results in the literature for two-dimensional problems, see e.g. [22, 24, 25, 26]. It is readily seen that in (4.5) the integral part and mixed derivative term are both treated by the explicit trapezoidal rule. Notice that the explicit stages $\widehat{Y}_0$, $\widetilde{Y}_0$ can be combined, so that the integral part is evaluated twice per time step. The implicit stages Y_j, $\widetilde{Y}_j$ (for $j = 1, 2$) are often called stabilizing corrections and are unidirectional. The pertinent linear systems for these stages are tridiagonal, so that they can be solved very efficiently using an a priori LU factorization.

10

6. *Two-step adaptation of the MCS (MCS2) scheme*:

   In [24] an alternative adaptation of the MCS scheme to PIDEs has been proposed, where the integral part is dealt with in a two-step Adams–Bashforth fashion:

   $$\begin{cases} X_0 = V^{n-1} + \Delta t A^{(D)} V^{n-1}, \\ Y_0 = X_0 + \frac{1}{2}\Delta t A^{(J)}(3V^{n-1} - V^{n-2}), \\ Y_j = Y_{j-1} + \theta\Delta t A_j(Y_j - V^{n-1}) \quad (j = 1, 2), \\ \widehat{Y}_0 = Y_0 + \theta\Delta t A^{(M)}(Y_2 - V^{n-1}), \\ \widetilde{Y}_0 = \widehat{Y}_0 + (\frac{1}{2} - \theta)\Delta t A^{(D)}(Y_2 - V^{n-1}), \\ \widetilde{Y}_j = \widetilde{Y}_{j-1} + \theta\Delta t A_j(\widetilde{Y}_j - V^{n-1}) \quad (j = 1, 2), \\ V^n = \widetilde{Y}_2. \end{cases} \tag{4.6}$$

   Method (4.6) is also of order two for any value θ. We take again $\theta = \frac{1}{3}$ and, for starting (4.6), define V^1 by the one-step method (4.5).

7. *Stabilizing correction two-step Adams-type (SC2A) scheme*:

   In Hundsdorfer & in 't Hout [18] a novel class of stabilizing correction multistep methods has recently been investigated for the numerical solution of PDEs. We select here a prominent member of this class, the two-step Adams-type scheme called SC2A, and consider its direct adaptation to PIDEs:

   $$\begin{cases} Y_0 = V^{n-1} + \Delta t\,(A^{(M)} + A^{(J)})\sum_{i=1}^2 \widehat{b}_i V^{n-i} + \Delta t\,(A_1 + A_2)\sum_{i=1}^2 \widecheck{b}_i V^{n-i}, \\ Y_j = Y_{j-1} + \theta\Delta t A_j(Y_j - V^{n-1}) \quad (j = 1, 2), \\ V^n = Y_2, \end{cases} \tag{4.7}$$

   with coefficients $(\widehat{b}_1, \widehat{b}_2) = \left(\frac{3}{2}, -\frac{1}{2}\right)$ and $(\widecheck{b}_1, \widecheck{b}_2) = \left(\frac{3}{2} - \theta, -\frac{1}{2} + \theta\right)$. The integral part and mixed derivative term are now both handled by the two-step Adams–Bashforth scheme. Method (4.7) is also of order two for any value θ. Following [18] we take $\theta = \frac{3}{4}$, which is motivated by stability and accuracy results. For starting (4.7), the one-step method (4.5) is used with $\theta = \frac{1}{3}$ to define V^1.

In each of the seven time-stepping schemes described above, any matrix-vector product with the matrix $A^{(J)}$ means applying Algorithm 1, hence requiring two FFTs and one IFFT. In the schemes (4.1), (4.4), (4.6), (4.7) only one such matrix-vector product arises per time step, whereas two such products appear in the schemes (4.2), (4.3), (4.5).

5 Numerical study

In this section we examine through extensive numerical experiments the convergence behaviour of the seven operator splitting schemes formulated in Section 4 in the numerical solution of the semidiscrete two-dimensional Merton PIDE. For the numerical study we consider the three parameter sets for the two-asset Merton jump-diffusion model and European-style option given in Table 1. Set 1 is taken from Clift & Forsyth [10]. Set 2 has the same diffusion parameters as in Zvan, Forsyth & Vetzal [42], and is complemented with jump parameters where the intensity λ is taken larger than in Set 1. Set 3 is fully new and yields more intensive jumps. Here λT is rather large, which poses a particular numerical challenge. Notice that for all three sets the correlation coefficients ρ and $\widehat{\rho}$ are both nonzero.

	σ_1	σ_2	ρ	λ	γ_1	γ_2	$\widehat{\rho}$	δ_1	δ_2	r	K	T
Set 1	0.12	0.15	0.30	0.60	-0.10	0.10	-0.20	0.17	0.13	0.05	100	1
Set 2	0.30	0.30	0.50	2	-0.50	0.30	-0.60	0.40	0.10	0.05	40	0.5
Set 3	0.20	0.30	0.70	8	-0.05	-0.20	0.50	0.45	0.06	0.05	40	1

Table 1: Parameter sets for the two-asset Merton jump-diffusion model and European option.

Two types of rainbow options are considered: a European put-on-the-min option and a European put-on-the-average option. Their payoff functions are, respectively,

$$\phi_{\text{put-on-min}}(s_1, s_2) = \max(0, K - \min(s_1, s_2))$$

and

$$\phi_{\text{put-on-average}}(s_1, s_2) = \max\left(0, K - \frac{s_1 + s_2}{2}\right).$$

Clearly, the two payoff functions ϕ are continuous, but not everywhere continuously differentiable. For each option a specific nonuniform spatial grid $\{(s_{1,i}, s_{2,j}) : 0 \leqslant i \leqslant m_1, 0 \leqslant j \leqslant m_2\}$ is chosen, where the nonsmoothness of ϕ is taken into account. The number of grid points is taken to be the same in both directions, $m_1 = m_2 = m$. For the put-on-the-min option, relatively many points $s_{1,i}$ and $s_{2,j}$ are placed around the locations $s_1 = K$ and $s_2 = K$, respectively. For the put-on-the-average option, the payoff is nonsmooth on the line segment given by $s_1 + s_2 = 2K$. Here the grid in each direction is taken to be uniform on the interval $[0, 2K]$ and nonuniform outside, with relatively many points inside this interval. For both options, the nonuniform grids are generated by applying a smooth transformation to an artificial uniform grid, see e.g. [21]. For Sets 1, 2, 3 we heuristically choose $S_{\max} = 5K, 30K, 50K$, respectively, in the case of the put-on-the-min and $S_{\max} = 5K, 15K, 25K$, respectively, in the case of the put-on-the-average option. As an illustration, Figure 1 displays the two spatial grids if $m = 50$, $K = 100$, $S_{\max} = 5K$. In Figure 2 the graphs of the two payoff functions are shown.

The initial vector $V(0) = V^0$ is defined by pointwise evaluation of the payoff function ϕ at the spatial grid points, except at those points that lie closest to the set of nonsmoothness of ϕ, where cell averaging is applied. For $k = 1, 2$ let

$$s_{k,l+1/2} = \tfrac{1}{2}(s_{k,l} + s_{k,l+1}) \quad \text{if } 0 \leqslant l < m,$$
$$h_{k,l+1/2} = s_{k,l+1/2} - s_{k,l-1/2} \quad \text{if } 0 \leqslant l \leqslant m,$$

with $s_{k,-1/2} = 0$ and $s_{k,m+1/2} = S_{\max}$. Then we define [21]

$$V_{i,j}(0) = \frac{1}{h_{1,i+1/2} h_{2,j+1/2}} \int_{s_{1,i-1/2}}^{s_{1,i+1/2}} \int_{s_{2,j-1/2}}^{s_{2,j+1/2}} \phi(s_1, s_2) \mathrm{d}s_2 \mathrm{d}s_1$$

whenever the cell

$$[s_{1,i-1/2}, s_{1,i+1/2}) \times [s_{2,j-1/2}, s_{2,j+1/2})$$

has a nonempty intersection with the set of points where ϕ is nonsmooth. For both payoffs under consideration, the above integral is readily calculated.

In the following we investigate, for all seven operator splitting methods formulated in Section 4 for the semidiscrete system (3.5), the temporal discretization error at $t = T$ on a natural region of

financial interest (ROI),

$$\widehat{E}^{ROI}(m,N) = \max\left\{ |V_{i,j}^{N'} - V_{i,j}(T)| : \Delta t = T/N' \text{ and } \tfrac{1}{2}K < s_{1,i}, s_{2,j} < \tfrac{3}{2}K \right\}. \qquad (5.1)$$

Clearly, the temporal discretization error is measured in the maximum norm, which is the most relevant norm in finance. Numerical experiments reveal that the evaluation of the two-dimensional integral part is computationally the dominant part in each time step, compare also [37]. In view of this, for a fair comparison of the different time-stepping methods, we consider the three schemes (4.2), (4.3), (4.5) with $N' = N$ time steps and the four schemes (4.1), (4.4), (4.6), (4.7) with $N' = 2N$ time steps, as the latter four schemes require only one evaluation of the integral part per time step, whereas the former three schemes employ two such evaluations. A reference value for $V(T)$ has been computed by applying the MCS2 scheme with 3000 time steps.

Figure 3 displays the temporal errors $\widehat{E}^{ROI}(m,N)$ for $m = 150$ and a range of values $10 \leqslant N \leqslant 1000$ for the put-on-the-min option (left column) and put-on-the-average option (right column) and the three parameter sets given in Table 1. For all seven splitting methods, the positive result holds that the temporal errors always lie below a moderate value and show a regular, monotonically decreasing behaviour as N increases. The CNFE scheme (4.1) shows an order of convergence just equal to one, as expected. Each of the other six splitting schemes (4.2)–(4.7) attains a favourable order of convergence equal to two. Comparing the different second-order methods, the MCS2 scheme yields the smallest temporal error constant in all experiments. It is further interesting to notice that for Set 3, with more intensive jumps, each of the two-step schemes CNAB, MCS2, SC2A outperforms all of the one-step schemes under consideration. The range of the temporal error constants for the different second-order methods is also the largest for this set. We performed additional numerical experiments, choosing a coarser ($m = 75$) as well as a finer ($m = 250$) spatial grid. The obtained results were almost identical to those displayed in Figure 3, indicating that the observed temporal convergence orders are valid in a favourable, stiff sense. Recall that the CNFI scheme has been applied with a fixed number of $l = 2$ iterations. We also carried out experiments for this scheme with $l = 3$ iterations, taking now $N' = \lceil 2N/3 \rceil$ time steps for a fair comparison. This did not lead to a significant improvement in accuracy, however, and did not alter the above conclusions.

A semi-closed analytical formula has recently been derived in Boen [6] for the value of a European put-on-the-min option under the two-asset Merton model. This formula is stated in Appendix B. Hence, in this case, we can also study the total discretization error at $t = T$ on the region of interest,

$$E^{ROI}(m,N) = \max\left\{ |V_{i,j}^{N'} - v(s_{1,i}, s_{2,j}, T)| : \Delta t = T/N' \text{ and } \tfrac{1}{2}K < s_{1,i}, s_{2,j} < \tfrac{3}{2}K \right\}. \qquad (5.2)$$

Figure 4 displays the total errors $E^{ROI}(m,N)$ for a range of values $10 \leqslant m \leqslant 500$ and the three parameter sets given in Table 1, where we have chosen $N = \lceil m/3 \rceil$ and N' as above. Clearly, with all splitting methods except CNFE, a second-order convergence behaviour is found, which is as desired. The total errors obtained with these six time-stepping methods are visually almost identical, indicating that the spatial discretization error dominates in these experiments.

6   Conclusion

In this paper we have studied seven different operator splitting schemes when applied to the two-dimensional Merton PIDE with a focus on IMEX and ADI methods. Each of the considered schemes conveniently treats the nonstiff, nonlocal integral part in an explicit fashion. Any matrix-vector product arising from the semidiscretization of the integral part is computed efficiently by means of FFT using Algorithm 1. Through ample numerical experiments, we have investigated the

convergence behaviour of the seven schemes and examined their relative performance. Except for the first-order CNFE scheme, all of the considered splitting schemes showed a stiff order of temporal convergence equal to two, which is as desired. The two-step MCS2 scheme with $\theta = \frac{1}{3}$ stood out as yielding the smallest temporal error constant in all the conducted experiments. Together with the positive results recently derived in [24], we therefore recommend this splitting scheme for the efficient and stable temporal discretization of two-dimensional PIDEs in financial option valuation.

Acknowledgements
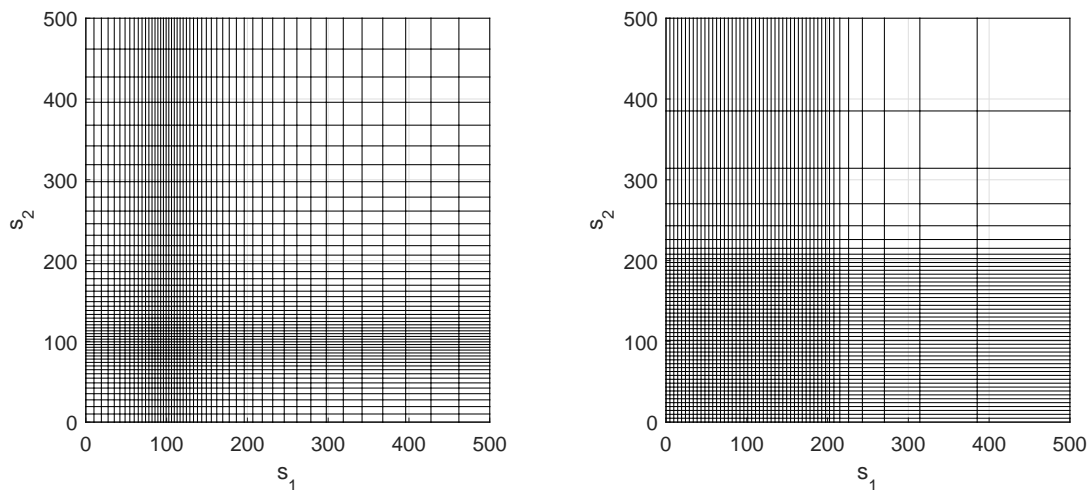
Figures

Figure 1: Spatial grid for the European put-on-the-min (left) and the European put-on-the-average (right) option if $m = 50$, $K = 100$, $S_{\max} = 5K$.
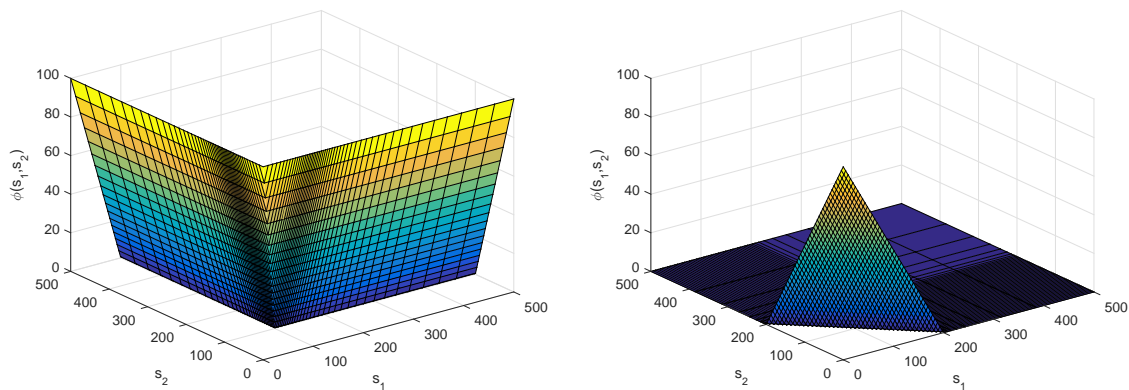
Figure 2: Payoff of the European put-on-the-min (left) and the European put-on-the-average (right) option if $K = 100$, $S_{\max} = 5K$.
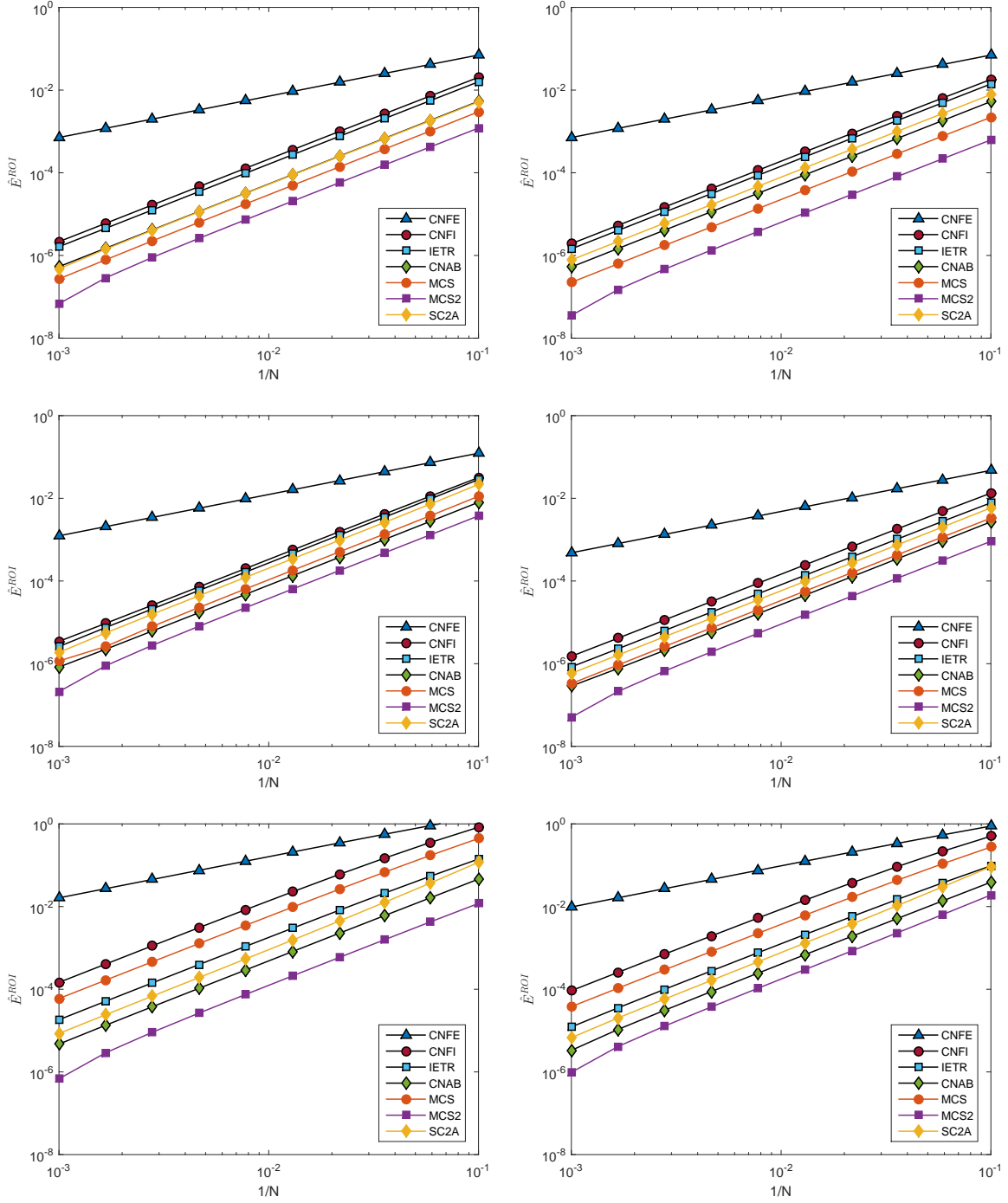
Figure 3: Temporal errors $\widehat{E}^{ROI}(150, N)$ of the seven operator splitting schemes in the case of the European put-on-the-min (left) and the put-on-the-average (right) option under the two-asset Merton model with parameter Set 1 (top), Set 2 (mid) and Set 3 (bottom) from Table 1. The CNFI, IETR, MCS schemes are applied with $\Delta t = T/N$ and the CNFE, CNAB, MCS2, SC2A schemes with $\Delta t = T/(2N)$.
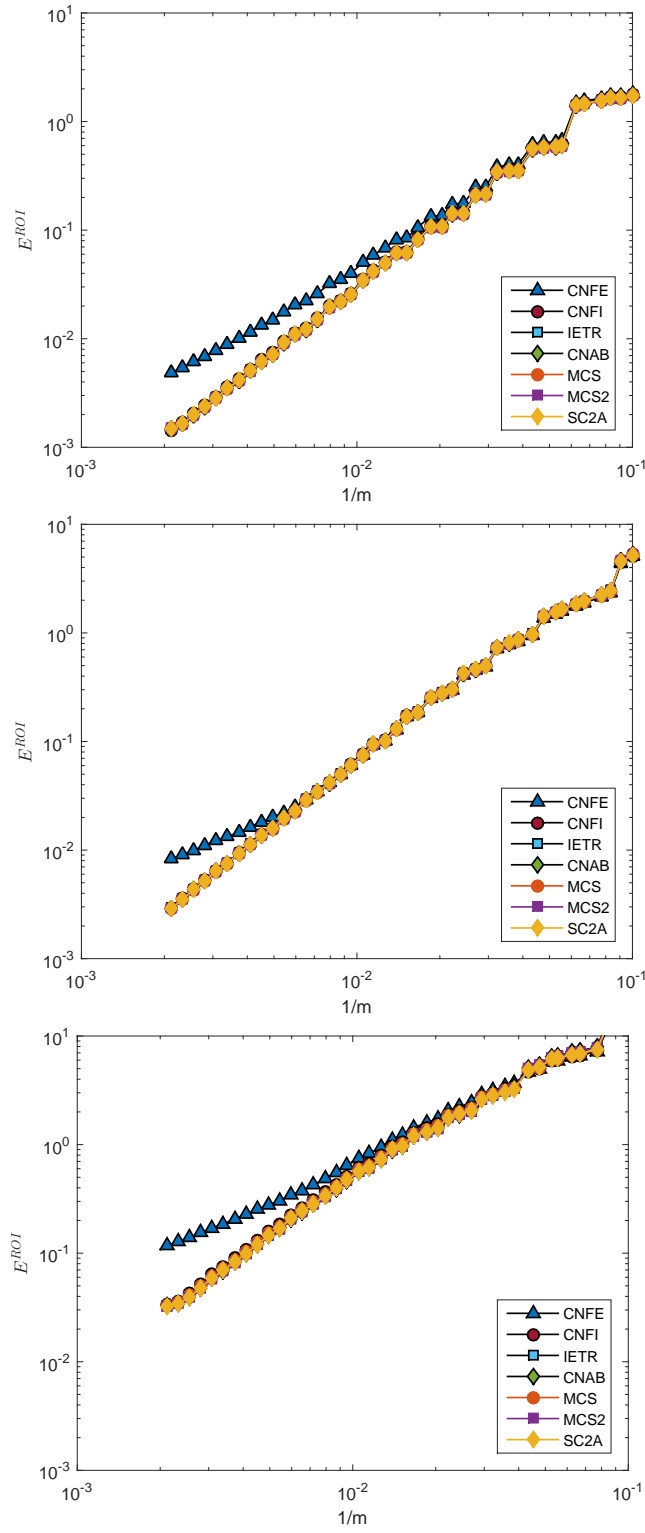
Figure 4: Total errors $E^{ROI}(m, N)$ in the case of the European put-on-the-min option under the two-asset Merton model with parameter Set 1 (top), Set 2 (mid) and Set 3 (bottom) from Table 1. The CNFI, IETR, MCS schemes are applied with $\Delta t = T/N$ and the CNFE, CNAB, MCS2, SC2A schemes with $\Delta t = T/(2N)$ and $N = \lceil m/3 \rceil$.

A The block-Toeplitz structure of $\overline{A}^{(J)}$

As an illustration, if $M_1 = M_2 = 4$, then $\overline{F}$ in the matrix $\overline{A}^{(J)} = \lambda\overline{F}$ is of the following form:

$$
\begin{pmatrix}
\overline{F}_{0,0} & \overline{F}_{1,0} & \overline{F}_{2,0} & \overline{F}_{3,0} & \overline{F}_{0,1} & \overline{F}_{1,1} & \overline{F}_{2,1} & \overline{F}_{3,1} & \overline{F}_{0,2} & \overline{F}_{1,2} & \overline{F}_{2,2} & \overline{F}_{3,2} & \overline{F}_{0,3} & \overline{F}_{1,3} & \overline{F}_{2,3} & \overline{F}_{3,3} \\
\overline{F}_{-1,0} & \overline{F}_{0,0} & \overline{F}_{1,0} & \overline{F}_{2,0} & \overline{F}_{-1,1} & \overline{F}_{0,1} & \overline{F}_{1,1} & \overline{F}_{2,1} & \overline{F}_{-1,2} & \overline{F}_{0,2} & \overline{F}_{1,2} & \overline{F}_{2,2} & \overline{F}_{-1,3} & \overline{F}_{0,3} & \overline{F}_{1,3} & \overline{F}_{2,3} \\
\overline{F}_{-2,0} & \overline{F}_{-1,0} & \overline{F}_{0,0} & \overline{F}_{1,0} & \overline{F}_{-2,1} & \overline{F}_{-1,1} & \overline{F}_{0,1} & \overline{F}_{1,1} & \overline{F}_{-2,2} & \overline{F}_{-1,2} & \overline{F}_{0,2} & \overline{F}_{1,2} & \overline{F}_{-2,3} & \overline{F}_{-1,3} & \overline{F}_{0,3} & \overline{F}_{1,3} \\
\overline{F}_{-3,0} & \overline{F}_{-2,0} & \overline{F}_{-1,0} & \overline{F}_{0,0} & \overline{F}_{-3,1} & \overline{F}_{-2,1} & \overline{F}_{-1,1} & \overline{F}_{0,1} & \overline{F}_{-3,2} & \overline{F}_{-2,2} & \overline{F}_{-1,2} & \overline{F}_{0,2} & \overline{F}_{-3,3} & \overline{F}_{-2,3} & \overline{F}_{-1,3} & \overline{F}_{0,3} \\
\overline{F}_{0,-1} & \overline{F}_{1,-1} & \overline{F}_{2,-1} & \overline{F}_{3,-1} & \overline{F}_{0,0} & \overline{F}_{1,0} & \overline{F}_{2,0} & \overline{F}_{3,0} & \overline{F}_{0,1} & \overline{F}_{1,1} & \overline{F}_{2,1} & \overline{F}_{3,1} & \overline{F}_{0,2} & \overline{F}_{1,2} & \overline{F}_{2,2} & \overline{F}_{3,2} \\
\overline{F}_{-1,-1} & \overline{F}_{0,-1} & \overline{F}_{1,-1} & \overline{F}_{2,-1} & \overline{F}_{-1,0} & \overline{F}_{0,0} & \overline{F}_{1,0} & \overline{F}_{2,0} & \overline{F}_{-1,1} & \overline{F}_{0,1} & \overline{F}_{1,1} & \overline{F}_{2,1} & \overline{F}_{-1,2} & \overline{F}_{0,2} & \overline{F}_{1,2} & \overline{F}_{2,2} \\
\overline{F}_{-2,-1} & \overline{F}_{-1,-1} & \overline{F}_{0,-1} & \overline{F}_{1,-1} & \overline{F}_{-2,0} & \overline{F}_{-1,0} & \overline{F}_{0,0} & \overline{F}_{1,0} & \overline{F}_{-2,1} & \overline{F}_{-1,1} & \overline{F}_{0,1} & \overline{F}_{1,1} & \overline{F}_{-2,2} & \overline{F}_{-1,2} & \overline{F}_{0,2} & \overline{F}_{1,2} \\
\overline{F}_{-3,-1} & \overline{F}_{-2,-1} & \overline{F}_{-1,-1} & \overline{F}_{0,-1} & \overline{F}_{-3,0} & \overline{F}_{-2,0} & \overline{F}_{-1,0} & \overline{F}_{0,0} & \overline{F}_{-3,1} & \overline{F}_{-2,1} & \overline{F}_{-1,1} & \overline{F}_{0,1} & \overline{F}_{-3,2} & \overline{F}_{-2,2} & \overline{F}_{-1,2} & \overline{F}_{0,2} \\
\overline{F}_{0,-2} & \overline{F}_{1,-2} & \overline{F}_{2,-2} & \overline{F}_{3,-2} & \overline{F}_{0,-1} & \overline{F}_{1,-1} & \overline{F}_{2,-1} & \overline{F}_{3,-1} & \overline{F}_{0,0} & \overline{F}_{1,0} & \overline{F}_{2,0} & \overline{F}_{3,0} & \overline{F}_{0,1} & \overline{F}_{1,1} & \overline{F}_{2,1} & \overline{F}_{3,1} \\
\overline{F}_{-1,-2} & \overline{F}_{0,-2} & \overline{F}_{1,-2} & \overline{F}_{2,-2} & \overline{F}_{-1,-1} & \overline{F}_{0,-1} & \overline{F}_{1,-1} & \overline{F}_{2,-1} & \overline{F}_{-1,0} & \overline{F}_{0,0} & \overline{F}_{1,0} & \overline{F}_{2,0} & \overline{F}_{-1,1} & \overline{F}_{0,1} & \overline{F}_{1,1} & \overline{F}_{2,1} \\
\overline{F}_{-2,-2} & \overline{F}_{-1,-2} & \overline{F}_{0,-2} & \overline{F}_{1,-2} & \overline{F}_{-2,-1} & \overline{F}_{-1,-1} & \overline{F}_{0,-1} & \overline{F}_{1,-1} & \overline{F}_{-2,0} & \overline{F}_{-1,0} & \overline{F}_{0,0} & \overline{F}_{1,0} & \overline{F}_{-2,1} & \overline{F}_{-1,1} & \overline{F}_{0,1} & \overline{F}_{1,1} \\
\overline{F}_{-3,-2} & \overline{F}_{-2,-2} & \overline{F}_{-1,-2} & \overline{F}_{0,-2} & \overline{F}_{-3,-1} & \overline{F}_{-2,-1} & \overline{F}_{-1,-1} & \overline{F}_{0,-1} & \overline{F}_{-3,0} & \overline{F}_{-2,0} & \overline{F}_{-1,0} & \overline{F}_{0,0} & \overline{F}_{-3,1} & \overline{F}_{-2,1} & \overline{F}_{-1,1} & \overline{F}_{0,1} \\
\overline{F}_{0,-3} & \overline{F}_{1,-3} & \overline{F}_{2,-3} & \overline{F}_{3,-3} & \overline{F}_{0,-2} & \overline{F}_{1,-2} & \overline{F}_{2,-2} & \overline{F}_{3,-2} & \overline{F}_{0,-1} & \overline{F}_{1,-1} & \overline{F}_{2,-1} & \overline{F}_{3,-1} & \overline{F}_{0,0} & \overline{F}_{1,0} & \overline{F}_{2,0} & \overline{F}_{3,0} \\
\overline{F}_{-1,-3} & \overline{F}_{0,-3} & \overline{F}_{1,-3} & \overline{F}_{2,-3} & \overline{F}_{-1,-2} & \overline{F}_{0,-2} & \overline{F}_{1,-2} & \overline{F}_{2,-2} & \overline{F}_{-1,-1} & \overline{F}_{0,-1} & \overline{F}_{1,-1} & \overline{F}_{2,-1} & \overline{F}_{-1,0} & \overline{F}_{0,0} & \overline{F}_{1,0} & \overline{F}_{2,0} \\
\overline{F}_{-2,-3} & \overline{F}_{-1,-3} & \overline{F}_{0,-3} & \overline{F}_{1,-3} & \overline{F}_{-2,-2} & \overline{F}_{-1,-2} & \overline{F}_{0,-2} & \overline{F}_{1,-2} & \overline{F}_{-2,-1} & \overline{F}_{-1,-1} & \overline{F}_{0,-1} & \overline{F}_{1,-1} & \overline{F}_{-2,0} & \overline{F}_{-1,0} & \overline{F}_{0,0} & \overline{F}_{1,0} \\
\overline{F}_{-3,-3} & \overline{F}_{-2,-3} & \overline{F}_{-1,-3} & \overline{F}_{0,-3} & \overline{F}_{-3,-2} & \overline{F}_{-2,-2} & \overline{F}_{-1,-2} & \overline{F}_{0,-2} & \overline{F}_{-3,-1} & \overline{F}_{-2,-1} & \overline{F}_{-1,-1} & \overline{F}_{0,-1} & \overline{F}_{-3,0} & \overline{F}_{-2,0} & \overline{F}_{-1,0} & \overline{F}_{0,0}
\end{pmatrix}
$$

This is a block-Toeplitz matrix with Toeplitz blocks $\overline{F}_d^1$, $d = -3, \ldots, 3$:

$$
\overline{F} = \begin{pmatrix}
\overline{F}_0^1 & \overline{F}_1^1 & \overline{F}_2^1 & \overline{F}_3^1 \\
\overline{F}_{-1}^1 & \overline{F}_0^1 & \overline{F}_1^1 & \overline{F}_2^1 \\
\overline{F}_{-2}^1 & \overline{F}_{-1}^1 & \overline{F}_0^1 & \overline{F}_1^1 \\
\overline{F}_{-3}^1 & \overline{F}_{-2}^1 & \overline{F}_{-1}^1 & \overline{F}_0^1
\end{pmatrix}.
$$

B Semi-closed analytical formula for put-on-the-min option

Let $M(x_1, x_2, \Sigma)$ denote the bivariate normal cumulative distribution function, evaluated at (x_1, x_2), with mean $(0, 0)^\top$ and covariance matrix Σ. Under the two-asset Merton jump-diffusion model, the value of a European put-on-the-min option at inception is given in Boen [6]:

$$
v(S_0^{(1)}, S_0^{(2)}, T) = \sum_{n=0}^{\infty} e^{-\lambda T} \frac{(\lambda T)^n}{n!} \left[e^{-rT} K - e^{-rT} K M(b_1, b_2, \Sigma_3) - \right.
$$
$$
\left. S_0^{(1)} e^{-\lambda \kappa_1 T + n\gamma_1 + \frac{1}{2}n\delta_1^2} M(d_{11}, d_1, \Sigma_1) - S_0^{(2)} e^{-\lambda \kappa_2 T + n\gamma_2 + \frac{1}{2}n\delta_2^2} M(d_{22}, d_2, \Sigma_2) \right],
$$

where

$$
b_i = \frac{\ln\left(\frac{S_0^{(i)}}{K}\right) + (r - \frac{1}{2}\sigma_i^2 - \lambda\kappa_i)T + n\gamma_i}{\sqrt{T}\sqrt{\sigma_i^2 + \frac{n\delta_i^2}{T}}}, \quad d_i = -b_i - \sqrt{T}\sqrt{\sigma_i^2 + \frac{n\delta_i^2}{T}} \quad (i = 1, 2),
$$

$$
d_{11} = \frac{\ln\left(\frac{S_0^{(2)}}{S_0^{(1)}}\right) + \left(-\frac{1}{2}\sigma^2 + \lambda(\kappa_1 - \kappa_2)\right)T - n(\gamma_1 - \gamma_2 + \delta_1^2 - \widehat{\rho}\delta_1\delta_2)}{\sqrt{T}\sqrt{\sigma^2 + \frac{n\delta^2}{T}}}, \quad d_{22} = -d_{11} - \sqrt{T}\sqrt{\sigma^2 + \frac{n\delta^2}{T}},
$$

$$
\Sigma_i = \begin{pmatrix} 1 & \rho_i \\ \rho_i & 1 \end{pmatrix} \quad (i = 1, 2, 3),
$$

with

$$
\sigma^2 = \sigma_1^2 - 2\rho\sigma_1\sigma_2 + \sigma_2^2, \quad \delta^2 = \delta_1^2 - 2\widehat{\rho}\delta_1\delta_2 + \delta_2^2,
$$

$$
\rho_i = \frac{\sqrt{\sigma_i^2 + \frac{n\delta_i^2}{T}}}{\sqrt{\sigma^2 + \frac{n\delta^2}{T}}} - \frac{\rho\sigma_1\sigma_2 + \frac{n\widehat{\rho}\delta_1\delta_2}{T}}{\sqrt{\left(\sigma^2 + \frac{n\delta^2}{T}\right)\left(\sigma_i^2 + \frac{n\delta_i^2}{T}\right)}} \quad (i = 1, 2),
$$

and

$$
\rho_3 = \frac{\rho\sigma_1\sigma_2 + \frac{n\widehat{\rho}\delta_1\delta_2}{T}}{\sqrt{\left(\sigma_1^2 + \frac{n\delta_1^2}{T}\right)\left(\sigma_2^2 + \frac{n\delta_2^2}{T}\right)}}.
$$

References

[1] A. Almendral and C. W. Oosterlee. Numerical valuation of options with jumps in the underlying. *Appl. Numer. Math.*, 53:1–18, 2005.

[2] L. Andersen and J. Andreasen. Jump-diffusion processes: volatility smile fitting and numerical methods for option pricing. *Rev. Deriv. Res.*, 4:231–262, 2000.

[3] B. E. Barrowes, F. L. Teixeira, and J. A. Kong. Fast algorithm for matrix-vector multiply of asymmetric multilevel block-Toeplitz matrices in 3D scattering. *Microw. Opt. Tech. Lett.*, 31:28–32, 2001.

[4] D. S. Bates. Jumps and stochastic volatility: exchange rate processes implicit in Deutsche Mark options. *Rev. Finan. Stud.*, 9:69–107, 1996.

[5] F. Black and M. Scholes. The pricing of options and corporate liabilities. *J. Polit. Econ.*, 81:637–654, 1973.

[6] L. Boen. European rainbow option values under the two-asset Merton jump-diffusion model. 2018. Submitted for publication.

[7] R. N. Bracewell. *The Fourier Transform and Its Applications.* McGraw Hill, 2000.

[8] M. Briani, R. Natalini, and G. Russo. Implicit-explicit numerical schemes for jump-diffusion processes. *Calcolo*, 44:33–57, 2007.

[9] P. Carr and D. B. Madan. Option valuation using the fast Fourier transform. *J. Comp. Finan.*, 2:61–73, 1999.

[10] S. S. Clift and P. A. Forsyth. Numerical solution of two asset jump diffusion models for option valuation. *Appl. Numer. Math.*, 58:743–782, 2008.

[11] R. Cont and E. Voltchkova. A finite difference scheme for option pricing in jump diffusion and exponential Lévy models. *SIAM J. Numer. Anal.*, 43:1596–1626, 2005.

[12] Y. d'Halluin, P. A. Forsyth, and K. R. Vetzal. Robust numerical methods for contingent claims under jump diffusion processes. *IMA J. Numer. Anal.*, 25:87–112, 2005.

[13] F. Fang and C. W. Oosterlee. A novel pricing method for European options based on Fourier-cosine series expansions. *SIAM J. Sci. Comp.*, 31:826–848, 2008.

[14] L. Feng and V. Linetsky. Pricing options in jump-diffusion models: an extrapolation approach. *Oper. Res.*, 56:304–325, 2008.

[15] J. Frank, W. Hundsdorfer, and J. G. Verwer. On the stability of implicit-explicit linear multi-step methods. *Appl. Numer. Math.*, 25:193–205, 1997.

[16] S. González-Pinto, E. Hairer, D. Hernández-Abreu, and S. Pérez-Rodríguez. AMF-type W-methods for parabolic problems with mixed derivatives. *SIAM J. Sci. Comp.*, 40:A2905–A2929, 2018.

[17] S. L. Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Rev. Finan. Stud.*, 6:327–343, 1993.

[18] W. Hundsdorfer and K. J. in 't Hout. On multistep stabilizing correction splitting methods with applications to the Heston model. *SIAM J. Sci. Comp.*, 40:A1408–A1429, 2018.

[19] W. Hundsdorfer and J. G. Verwer. *Numerical Solution of Time-Dependent Advection-Diffusion-Reaction Equations.* Springer, 2003.

[20] K. J. in 't Hout. On the contractivity of implicit-explicit linear multistep methods. *Appl. Numer. Math.*, 42:201–212, 2002.

[21] K. J. in 't Hout. *Numerical Partial Differential Equations in Finance Explained.* Palgrave Macmillan, 2017.

[22] K. J. in 't Hout and C. Mishra. Stability of the modified Craig–Sneyd scheme for two-dimensional convection-diffusion equations with mixed derivative term. *Math. Comput. Simul.*, 81:2540–2548, 2011.

[23] K. J. in 't Hout and J. Toivanen. Application of operator splitting methods in finance. In *Splitting Methods in Communication, Imaging, Science, and Engineering*, pages 541–575. Springer, 2016.

[24] K. J. in 't Hout and J. Toivanen. ADI schemes for valuing European options under the Bates model. *Appl. Numer. Math.*, 130:143–156, 2018.

[25] K. J. in 't Hout and B. D. Welfert. Unconditional stability of second-order ADI schemes applied to multi-dimensional diffusion equations with mixed derivative terms. *Appl. Numer. Math.*, 59:677–692, 2009.

[26] K. J. in 't Hout and M. Wyns. Convergence of the Modified Craig–Sneyd scheme for two-dimensional convection-diffusion equations with mixed derivative term. *J. Comp. Appl. Math.*, 296:170–180, 2016.

[27] A. Itkin. *Pricing Derivatives under Lévy Models.* Birkhäuser, 2017.

[28] A. Itkin and P. Carr. Jumps without tears: a new splitting technology for barrier options. *Int. J. Numer. Anal. Model.*, 8:667–704, 2011.

[29] V. Kaushansky, A. Lipton, and C. Reisinger. Numerical analysis of an extended structural default model with mutual liabilities and jump risk. *J. Comp. Sci.*, 24:218–231, 2018.

[30] Y. Kwon and Y. Lee. A second-order finite difference method for option pricing under jump-diffusion models. *SIAM J. Numer. Anal.*, 49:2598–2617, 2011.

[31] W. Margrabe. The value of an option to exchange one asset for another. *J. Finan.*, 33:177–186, 1978.

[32] R. C. Merton. Option pricing when underlying stock returns are discontinuous. *J. Finan. Econ.*, 3:125–144, 1976.

[33] D. M. Pooley, K. R. Vetzal, and P. A. Forsyth. Convergence remedies for non-smooth payoffs in option pricing. *J. Comp. Finan.*, 6:25–40, 2003.

[34] R. Rannacher. Finite element solution of diffusion problems with irregular data. *Numer. Math.*, 43:309–327, 1984.

[35] M. Rubinstein. Somewhere over the rainbow. *Risk*, 4:63–66, 1991.

[36] S. Salmi and J. Toivanen. IMEX schemes for pricing options under jump-diffusion models. *Appl. Numer. Math.*, 84:33–45, 2014.

[37] S. Salmi, J. Toivanen, and L. von Sydow. An IMEX-scheme for pricing options under stochastic volatility models with jumps. *SIAM J. Sci. Comp.*, 36:B817–B834, 2014.

[38] E. S. Schwartz. The valuation of warrants: implementing a new approach. *J. Finan. Econ.*, 4:79–93, 1977.

[39] D. Tavella and C. Randall. *Pricing Financial Instruments.* Wiley, 2000.

[40] L. von Sydow, S. Milovanović, E. Larsson, K. J. in 't Hout, M. Wiktorsson, C.W. Oosterlee, V. Shcherbakov, M. Wyns, A. Leitao, S. Jain, T. Haentjens, and J. Waldén. BENCHOP - SLV: the BENCHmarking project in Option Pricing - Stochastic and Local Volatility problems. *Int. J. Comp. Math.* DOI: 10.1080/00207160.2018.1544368.

[41] P. Wilmott. *Paul Wilmott on Quantitative Finance*. Wiley, 2006.

[42] R. Zvan, P. A. Forsyth, and K. R. Vetzal. A finite volume approach for contingent claims valuation. *IMA J. Numer. Anal.*, 21:703–731, 2001.