Bergische Universität Wuppertal

Fachbereich Mathematik und Naturwissenschaften

Institute of Mathematical Modelling, Analysis and Computational Mathematics (IMACM)

Preprint BUW-IMACM 16/20

Zuzana Bučková, Pedro Pólvora, Matthias Ehrhardt, Michael Günther

Implementation of Alternating Direction Explicit Methods for Higher Dimensional Black-Scholes Equations

October 19, 2016

http://www.math.uni-wuppertal.de

Implementation of alternating direction explicit methods for higher dimensional Black-Scholes equations

Z. Bučková, P. Pólvora, M. Ehrhardt, and M. Günther

Citation: AIP Conference Proceedings **1773**, 030001 (2016); doi: 10.1063/1.4964961
View online: http://dx.doi.org/10.1063/1.4964961
View Table of Contents: http://scitation.aip.org/content/aip/proceeding/aipcp/1773?ver=pdfcov
Published by the AIP Publishing

Implementation of Alternating Direction Explicit Methods for Higher Dimensional Black-Scholes Equations

Z. Bučková[1,2,a)], P. Pólvora[2,3,b)], M. Ehrhardt[1,c)] and M. Günther[1,d)]

[1]*Lehrstuhl für Angewandte Mathematik und Numerische Analysis, Fachbereich C Mathematik und Naturwissenschaften, Bergische Universität Wuppertal, Gaußstr. 20, 42119 Wuppertal, Germany*
[2]*Department of Applied Mathematics and Statistics, Faculty of Mathematics, Physics and Informatics, Comenius University, 842 48 Bratislava, Slovakia*
[3]*Department of Mathematics, ISEG, University of Lisbon, Rua do Quelhas 6, 1200-781 Lisbon, Portugal*

[a)]Corresponding author: zuzana.zikova.buckova@gmail.com
[b)]ppolvora@iseg.ulisboa.pt
[c)]ehrhardt@math.uni-wuppertal.de
[d)]guenther@math.uni-wuppertal.de

Abstract. In this work we propose Alternating Direction Explicit (ADE) schemes for the two and three dimensional linear Black-Scholes pricing model. Our implemented methodology can be easily extended to higher dimensions. The main advantage of ADE schemes is that they are explicit and exhibit good stability properties. Results concerning the experimental order of convergence are included.

INTRODUCTION TO ALTERNATING DIRECTION EXPLICIT SCHEMES AND MULTIDIMENSIONAL BLACK-SCHOLES MODEL

In this section we introduce alternating direction explicit methods used for one-dimensional models and the description of the higher dimensional Black-Scholes model.

Alternating Direction Explicit Schemes

Alternating Direction Explicit Schemes (ADE) are efficient finite-difference schemes to solve partial differential equations where the discretization of the spatial derivatives is made using available information of both the current and the previous time-steps such that the solution can be determined without solving a linear system of equations.

ADE schemes were proposed by Saul'ev [1] in 1957, later developed by Larkin [2], Bakarat and Clark [3] in 1964-66. More recently, these schemes have received some attention by Duffy [4], [5] 2013 and Leung and Osher [6] 2005 who have studied and applied these schemes in both financial modelling and other applications.

The ADE methods have some similarities with the Alternating Direction Implicit (ADI) methods which, as the name implies, are implicit in time. It should be noted that although the ADI methods are very well studied for the case of high-dimensional equations in finance [7] the same cannot be said about ADE methods in finance and this was partly our motivation as we consider that ADE methods are an interesting alternative over ADI. There are also other methods which are possible to use for solving multidimensional Black-Scholes equations, *e.g.*, spectral methods for linear models and their modifications for nonlinear cases. Again, while these methods are well studied, the same cannot be said about ADE methods. In spite of some drawbacks of the ADE schemes (such as conditional stability and consistency) it is worthy to consider them as an alternative approach to the aforementioned methods as they possess interesting positive aspects. Although ADE schemes belong to the group of explicit FDM, they have better stability and consistency properties than classical explicit Euler schemes. The uniqueness of the scheme comes from the way of combining two particular solutions,the so-called sweeps. This way we can get the desired properties from the numerical analysis point of view. For special cases of equations in one-dimension it was shown unconditional

Application of Mathematics in Technical and Natural Sciences
AIP Conf. Proc. 1773, 030001-1–030001-13; doi: 10.1063/1.4964961
Published by AIP Publishing. 978-0-7354-1431-0/$30.00

stability, *e.g.*, in [6, 8, 5]. For higher dimensional models, especially including the mixed derivatives, it leads to a conditional stable scheme. Although there are no stability proofs for higher-dimensional cases, the scheme perform reasonably well from the experimental point of view.

Some advantages of the ADE methods are that they can be implemented in a parallel framework and are very fast due to their explicitness; for a complete survey on the advantages and the motivation to use them in a wide range of problems we refer the reader to [4], [5].

Numerical analysis results focusing on stability and consistency considerations are described in [6] and [8]. In [8] a numerical analysis of convection-diffusion-reaction equation with constant coefficients and smooth initial data is provided. The authors proved that the ADE method applied to the one-dimensional reaction-diffusion equation on a uniform mesh with the discretization of the diffusion according to Saul'ev [1] and the discretization of the convection term following Towler and Yang [9] is unconditionally stable. If a convection term is added to the equation and upwind discretization for this term is used, the ADE scheme is also unconditionally stable *cf.* [8].

In the ADE schemes one computes for each time level two different solutions which are referred to as sweeps. Hereby the number of sweeps does not depend on the dimension. It has been shown [8, 4, 6] that for the upward and downward sweep the consistency is of order $O((d\tau)^2 + h^2 + \frac{d\tau}{h})$, where $d\tau$ is the time step and h denotes the space step. An exceptionality of the ADE method is that the average of upward and downward solutions has consistency of order $O((d\tau)^2 + h^2)$. For linear models, unconditional stability results and the $O((d\tau)^2 + h^2)$ order of consistency leads to the $O((d\tau)^2 + h^2)$ convergence order.

Stability, consistency and convergence analysis can be extended to higher dimensional models.

The straightforward implementation also to nonlinear cases with preserving good stability and consistency properties of the scheme is a strong advantage. In this paper we show how one can implement this scheme for higher dimensional models by focusing on a linear model. However, one could use this procedure for non-linear models as well. One way how to do it is to solve nonlinear equation in each time level, instead of system of nonlinear equations in case of implicit schemes. Another way is to keep nonlinearity in the explicit form and solve it directly. Powerful tool for nonlinear equations represents also Alternating segment explicit-implicit and implicit-explicit parallel difference method [10].

Multi-dimensional Black-Scholes models

One of the simplest financial derivative pricing models is the Black-Scholes model which has the form of a one dimensional parabolic partial differential equation (PDE) with one space dimension and one time dimension.

Considering more complex models, that include a variety of market effects such as stochastic volatility or correlation among financial assets can increase the dimensionality of the pricing PDE. Also, pricing financial derivatives with more than one underlying asset yields PDEs that have at least as many spatial dimensions as the number of underlying assets.

Since a closed form formula can be only found in very special cases, determining solutions for these models has to be done in general using numerical methods, but the higher the dimension of the PDE models the bigger the overall complexity of the implementation of these methods.

Since the ADE scheme is explicit, stable and thus efficient, it represents a good candidate to compute the numerical solution of these multi-dimensional models in finance.

Here we present the implementation of the ADE schemes to two and three dimensional models appearing in finance, esp. the multi-dimensional linear Black-Scholes model. One of the advantages of this approach is that its fundamental implementation set-up can be transferred to higher dimensions.

We study a financial derivative that can be exercised only at a pre-fixed maturity time T (commonly referred as 'European' option) and whose payoff depends on the value of N financial assets with prices $S_1, \dots, S_N$. We assume a financial market with the standard Black-Scholes assumptions, explained in details, *e.g.*, in [11]. Although this is very restrictive from the modeling point of view, it is enough to illustrate the implementation of the ADE schemes in a high-dimensional setting. Under this model the price of a derivative $V(S_1, \dots, S_N, \tau)$ is given by the following $N-$dimensional linear parabolic PDE:

$$\frac{\partial V}{\partial \tau} = \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{\Gamma_{ij} S_i S_j}{2} \frac{\partial^2 V}{\partial S_i \partial S_j} + \sum_{i=1}^{N} r S_i \frac{\partial V}{\partial S_i} - rV, \qquad \tau \geq 0,\ \forall S \in \mathbb{R}_0^+, \tag{1}$$

where r denotes the risk-free interest rate, $\tau = T - t$ is the remaining time to the maturity time T and we have the covariance matrix,

$$[\Gamma]_{ij} \equiv \rho_{ij}\sigma_i\sigma_j \qquad i, j = 1, \ldots, N, \tag{2}$$

with ρ_{ij} being the correlation between asset i and j and σ_i the standard deviation of the asset i. Additionally we have an initial condition which is defined by the payoff of the option,

$$V(S_1, \ldots, S_N, 0) = \Phi(S_1, \ldots, S_N). \tag{3}$$

We obtain different models by choosing different numbers of underlying assets (*i.e.*, the number of spatial variables) and defining different payoff functions with corresponding initial conditions. Here we consider both spread options and call options which have payoffs given by:

$$2-D \text{ Spread option: } V(S_1, S_2, 0) = \max(S_1 - S_2 - K, 0),$$
$$N-D \text{ Call option: } V(S_1, \ldots, S_N, 0) = \max(\max(S_1, \ldots, S_N) - K, 0)).$$

This paper is structured as follows. After the introduction of the ADE schemes and the multi-dimensional Black-Scholes models, we focus on the details of the ADE scheme in the second section. In the third section we introduce the numerical scheme with difference quotients for the ADE. The fourth section consists of the numerical results focusing on the experimental study of convergence for two examples using different payoff structures: two dimensional spread option model and three dimensional call option model. The last section sums up results and presents the outlook.

ADE SCHEMES FOR MULTI-DIMENSIONAL MODELS

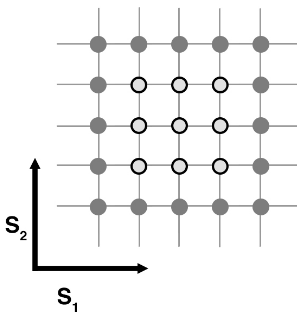
In this section we introduce the ADE scheme for multi-dimensional PDE models. We first consider the 2D case and then we proceed to higher dimensional cases.

ADE Schemes for Two-dimensional Models

We now explain in detail how to construct the ADE scheme for two-dimensional PDE models, *i.e.*, $N = 2$ in (1).

The first key aspect of this scheme is choosing the difference quotients approximating the partial derivatives of our equation in a way that we use the information from both time levels without the need to solve a linear system of equations. In particular, in a two dimensional setting, we would use the points as exemplified in Figure 1: we wish to compute the value in black, at time level $n + 1$, and we use the information from the neighbor points with an empty filling, from both time level n and $n + 1$.

The second key aspect is that in order to improve the accuracy of this scheme, for each time-level two different calculations of the grid points are done using different difference quotients, these are referred to as the *downward sweep* and the *upward sweep*. Then, the solution at that time level is taken as the average of both sweeps. From Figure 1 right and Figure 2 right the difference between the two sweeps is apparent.



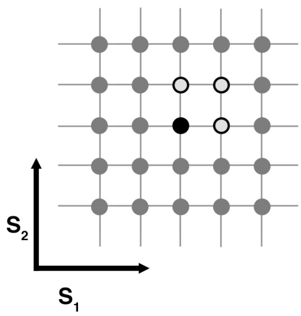


FIGURE 1. Downward sweep. Left figure: time level n. Right figure: time level $n + 1$. We depict the spatial grid for two different time-steps. The empty circles represent the points used in the computation of the value at the location of the black circle. S_1 and S_2 denote the spatial dimensions

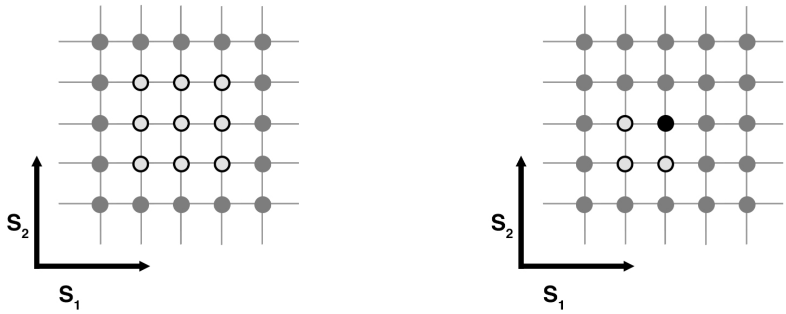


FIGURE 2. Upwards sweep. As in Figure 1, the empty circles represent the points used in the computation of the value at the location of the black circle. S_1 and S_2 denote the spatial dimensions. Left figure: time level n. Right figure: time level $n + 1$

The final key aspect is that the structure imposed by the stencil illustrated in Figures 1 and 2 is not by itself enough to guarantee that the scheme is explicit, we must make sure that the empty filling points in the time level $n + 1$ have been computed before we compute the black point. This imposes a structure on the algorithm to compute the points as illustrated in Figure 3.

For a fixed time level and starting from the boundary we see that in the first step we can only compute the points numbered as 1, since, our stencil is as described in the Figures 1 right, 2 right. After computing these points we have a total of four points that can be now computed, these are numbered as 2. Hence, we chose any of those points which in turn allows new points to be computed, and so forth. As long as we respect this order, our algorithm is fully explicit.

As we can see in the second step, we have more than one possibility per step as to what point to compute, hence, there are different sequences of points. A natural choice is to choose the sequence of points as shown in Figure 4. We called this approach of numbering as a *jumping approach* or *house numbering approach*. We are moving from one corner of the square to another where diagonal points are computed and the others. We could do the same strategy in higher dimensions, but it is not straightforward and yields no advantage in comparison with the next approach. The approach we have implemented is a *row-wise ordering* and it is displayed in Figure 5. It is just more straightforward way of ordering grid points. It is also more convenient to use this approach in hypercubes.

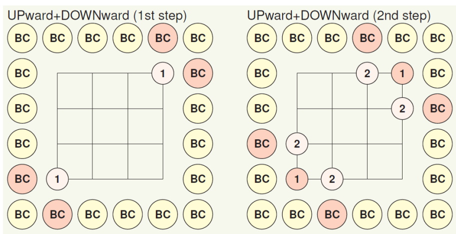


FIGURE 3. First steps of the algorithm in the 2D case. Elements numbered 1 correspond to the step 1 from both sweeps UP and DOWN. Elements numbered 2 correspond to the elements that can be computed as the second step also for both sweeps

ADE Schemes for Three and Higher Dimensional Models

In this section we describe how to extend the two-dimensional ADE scheme introduced before to three and higher dimensional models. We suggest an algorithm which can be extended to higher dimensional models quite easily.

As for the two dimensional case, a key part of the ADE in higher dimensions is to choose the proper difference quotients such that we keep good stability and consistency properties and explicitness of the scheme. Solely for simplicity we will use a uniform grid.

Consider the three dimensional case where we are solving the PDE of the price of an option under the linear Black-Scholes model introduced before, with three underlying assets. The PDE's solution will be a four-dimensional function where one dimension represents time and the other three are spatial dimensions representing the values of

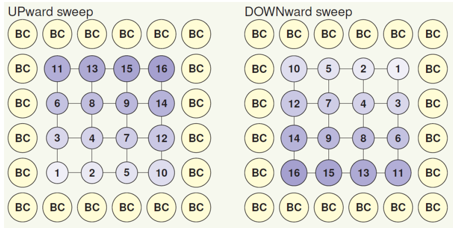


FIGURE 4. The complete algorithm in the 2D case. The points are computed in the order of the numbering. The left part of the figure refers to the UP sweep and the right to the DOWN sweep. Approach of numbering is called jumping approach or house numbering

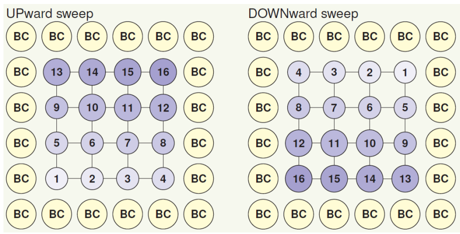


FIGURE 5. The complete algorithm in the 2D case. The points are computed in the order of the numbering. The left part of the figure refers to the UP sweep and the right to the DOWN sweep. Approach of numbering is called line approach or sequence approach

the underlying assets. For each time level, we have a three dimensional solution which can be illustrated as a three-dimensional grid. Recall that the initial condition is given for $V(S_1, S_2, S_3, 0)$ and step by step we calculate the values for the new time layer.

As before, we retain the explicitness of the scheme by using only values that have already been computed at the current time level. Specifically, this explicit (as in the lower dimensional case) is obtained by computing the value of points in a particular sequence that only uses points that either arise from the previous time level or that have been already computed for the current time level.

For illustration purposes we depict a two-dimensional slice of the domain in Figure 6. We see that we move in a straight line in one dimension until we hit the boundary and then we proceed to the next point in the second dimension and so forth. By using this approach, the extension to higher dimensional models is straightforward.

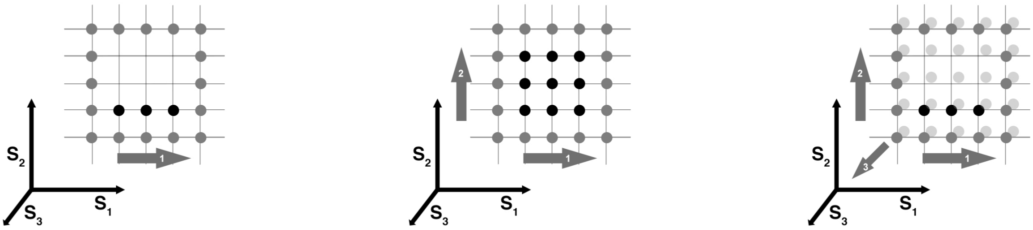


FIGURE 6. Algorithm for computing all the points in the upward sweep solution of the three dimensional implementation of the ADE, the grey dots represent the boundary conditions and the black dots represent the computed values. The arrows represent the direction and sequential order of the computation. First, second and third direction in the pictures, respectively

Boundary Conditions

In higher dimensional models we also have to deal with the issue of boundary conditions. Just as in the three dimensional model 8 boundary conditions are required (each edge of the cube), for a N-dimensional model 2^N boundary conditions have to be prescribed. In an ideal case we prescribe values for the maximum values of the assets prices (truncated values) as Dirichlet boundary conditions. Alternatively one could also consider Neumann boundary conditions or Robin type boundary conditions.

NUMERICAL SCHEME

The discretization of the PDE (1) is done on a uniform grid. In the time domain we have N_τ subintervals of the interval $[0, T]$, thus the time step size is defined as $d\tau = T/N_t$. As we have N different underlying assets our spatial space is N-dimensional. In our numerical studies we consider both $N = 2$ and $N = 3$.

For the 3-dimensional model we have 3 spatial intervals $[x_{\min}, x_{\max}]$, $[y_{\min}, y_{\max}]$, $[z_{\min}, z_{\max}]$, specifically $[0, S_{1\max}]$, $[0, S_{2\max}]$, $[0, S_{3\max}]$ as all stocks have non-negative values.

The space steps on the uniform grid are defined by the following $h_\alpha = S_{\alpha\max}/N_\alpha$ for $\alpha = 1, \ldots, 3$, where $S_{\alpha\max}$ denotes the maximal value for the asset α and N_α denotes the number of points for the direction of the α asset.

A point on the spatial grid is then given by $[x_i, y_j, z_k]$ with $x_i = (i-1)h_1$, $y_j = (j-1)h_2$,$z_k = (k-1)h_3$; where $i = 1, \ldots, N_1 + 1$, $j = 1, \ldots, N_2 + 1$, $k = 1, \ldots, N_3 + 1$.

The discrete numerical solution of the 3-dimensional Black-Scholes equation at $[x_i, y_j, z_k]$ and time level n for the upward sweep is denoted by $u^n_{ijk} = u(x_i, y_j, z_k, n)$ and for the downward sweep is denoted by $u^n_{ijk} = d(x_i, y_j, z_k, n)$,

Since this notation would easily become very cumbersome we will introduce some abbreviations: $u(x_i, y_j, z_k, n)$ and $d(x_i, y_j, z_k, n)$ will be shortened to u^n and d^n. When we consider u at a point shifted from the point indexed by (i, j, k) we will introduce a subscript $u^n_{\beta^+}$ where β denotes the direction where we are performing the shift. For example,

$$u(x_i, y_{j+1}, z_k, n) =: u^n_{2^+} \qquad u(x_i, y_j, z_{k-1}, n) =: u^n_{3^-}. \tag{4}$$

In the case that we have shifts in multiple directions we simply introduce another subscript, for example,

$$u(x_{i-1}, y_{j+1}, z_k, n) =: u^n_{1^-2^+} \qquad u(x_i, y_{j+1}, z_{k-1}, n) =: u^n_{3^-2^+}. \tag{5}$$

Note that this notation would not be suitable if we denote a point such as $u(x_i, y_{j+3}, z_k, n)$ but since we are considering only a first-order scheme we will not have shifts of more than 1 unit, therefore this notation is appropriate.

Algorithm of the Scheme

We can construct the upward sweep and the downward sweep separately for each time step and then combine them, this bring opportunities for the parallelization of the scheme. The upward sweep is calculated in a way that we are moving from one corner of the hypercube to the opposite. The downward sweep is constructed in the opposite way. This procedure can be done in different ways, but it is important to keep the explicitness of the scheme in each of the sweeps. In the following we outline the algorithms. As an illustration the upward sweep of this algorithm is represented in Figure 6.
According to the described procedure we construct upward and downward sweep of the solution and after each time level we calculated its average. This way we get final numerical solution c^n.
For $n = 0, 1, \ldots, N_t - 1$ we repeat

1. Initialization: $u^n = c^n$; $\quad d^n = c^n$;
2. Upward: u^{n+1}_{ijk}; $\qquad i = 1, \ldots, N_1 - 1$; $j = 1, \ldots, N_2 - 1$; $k = 1, \ldots, N_3 - 1$;
3. Downward: d^{n+1}_{ijk}; $\quad i = N_1 - 1, \ldots, 1$; $j = N_2 - 1, \ldots, 1$; $k = N_3 - 1, \ldots, 1$;
4. $c^n = (u^{n+1} + d^{n+1})/2$.

Upward Finite Difference Quotients and Its Numerical Scheme

Finite difference quotients using the upward sweep in the ADE scheme are introduced. Exact continuous solution of the PDE (1) in the point $x_i, y_j, z_k, \tau_{n+\frac{1}{2}}$ is denoted as: $V := V(x, y, z, \tau)|_{(x_i, y_j, z_k, \tau_{n+\frac{1}{2}})}$ and, *e.g.*, in the time level n it is denoted as: $V^n := V(x,y,z,\tau)|_{(x_i,y_j,z_k,\tau_n)}$, For derivatives it holds as follow: $\frac{\partial V}{\partial \tau} := \frac{\partial V(x,y,z,\tau)}{\partial \tau}|_{(x_i,y_j,z_k,\tau_{n+\frac{1}{2}})}$. Approximation of the V^n is denoted as u^n for an upward sweep

$$V \simeq \frac{V^n + V^{n+1}}{2}. \tag{6}$$

For the time derivative the explicit Euler discretization is used:

$$\frac{\partial V}{\partial \tau} = \frac{V^{n+1} - V^n}{d\tau} + O(\tau_n^2). \tag{7}$$

In the convection term we choose the Roberts and Weiss approximation, Ref. [12]

$$\frac{\partial V}{\partial S_\alpha} = \frac{V^n_{\alpha^+} - V^n + V^{n+1} - V^{n+1}_{\alpha^-}}{2h_\alpha} + O(h_\alpha^2), \qquad \forall \alpha = 1,2,3 \tag{8}$$

and the diffusion term is approximated by a special kind of central difference,

$$\frac{\partial^2 V}{\partial S_\alpha^2} = \frac{V^n_{\alpha^+} - V^n - V^{n+1} + V^{n+1}_{\alpha^-}}{h_\alpha^2} + O(h_\alpha^2), \qquad \forall \alpha = 1,2,3. \tag{9}$$

Note that in all the above mentioned difference quotients we use values from two time layers in the fashion that we can use all the values from the previous time layer, but due to the algorithm explained in Figure 5 only known values from the current time layer are used to keep the explicitness of the algorithm.
We approximate mixed term derivatives in an explicit way, as well:

$$\frac{\partial^2 V}{\partial S_\alpha S_\beta} = \frac{V^n_{\alpha^+\beta^+} - V^n_{\alpha^+\beta^-} - V^n_{\alpha^-\beta^+} + V^n_{\alpha^-\beta^-}}{4h_\alpha h_\beta} + O(h_\alpha^2 + h_\beta^2) \qquad \forall \alpha,\beta = 1,2,3. \tag{10}$$

We now use the difference quotients introduced above to discretize the 3-dimensional Black-Scholes PDE (1). Let us define,

$$\gamma_1^{ij}(x_1,x_2) \equiv \frac{dt}{2h_ih_j}\Gamma_{ij}S_i(x_1)S_j(x_2), \qquad \gamma_2^i(x_1) \equiv \frac{dt}{2h_i} rS_i(x_1),$$

with $S_i(p) = (p-1)h_i$. The discretized equation for the 3D model becomes,

$$u^{n+1} - u^n = \sum_{i=1}^{3} \gamma_1^{ii}\left[u^n_{i^+} - u^n - u^{n+1} + u^{n+1}_{i^-}\right] + \sum_{i=1}^{3}\sum_{j=1,i\neq j}^{3} \frac{\gamma_1^{ij}}{4}\left[u^n_{i^+j^+} - u^n_{i^+j^-} - u^n_{i^-j^+} + u^n_{i^-j^-}\right]$$
$$+ \sum_{i=1}^{3} \gamma_2^i\left[u^n_{i^+} - u^n + u^{n+1} - u^{n+1}_{i^-}\right] - r\frac{u^n + u^{n+1}}{2}. \tag{11}$$

The resulting algorithm is fully explicit, if we follow the procedure illustrated in Figure 3. From Eq. (11) we express u^{n+1} and we realize an explicit formula for the scheme.

Difference Quotients and Numerical Scheme for the Downward Sweep

Let V be the exact continuous solution in point $x_i, y_j, z_k, \tau_{n+\frac{1}{2}}$. Approximation of the V^n obtained by downward sweep is d^n, where the following difference quotients are used:

$$V \simeq \frac{V^n + V^{n+1}}{2}, \qquad \frac{\partial V}{\partial \tau} = \frac{V^{n+1} - V^n}{dt} + O(\tau_n^2), \tag{12}$$

$$\frac{\partial V}{\partial S_\alpha} = \frac{V^{n+1}_{\alpha^+} - V^n + V^{n+1} - V^n_{\alpha^-}}{2h_\alpha} + O(h_\alpha^2), \qquad \forall \alpha = 1,2,3, \tag{13}$$

$$\frac{\partial^2 V}{\partial S_\alpha^2} = \frac{V_{\alpha^+}^{n+1} - V^n - V^{n+1} + V_{\alpha^-}^n}{h_\alpha^2} + O(h_\alpha^2), \qquad \forall\alpha = 1,2,3, \tag{14}$$

$$\frac{\partial^2 V}{\partial S_\alpha S_\beta} = \frac{V_{\alpha^+\beta^+}^n - V_{\alpha^+\beta^-}^n - V_{\alpha^-\beta^+}^n + V_{\alpha^-\beta^-}^n}{4h_\alpha h_\beta} + O(h_\alpha^2 + h_\beta^2), \qquad \forall\alpha,\beta = 1,2,3. \tag{15}$$

In the same manner we get the discretized equation for the downward sweep,

$$d^{n+1} - d^n = \sum_{i=1}^{3}\gamma_1^{ii}\left[d_{i^+}^{n+1} - d^n - d^{n+1} + d_{i^-}^n\right] + \sum_{i=1}^{3}\sum_{j=1,i\neq j}^{3}\frac{\gamma_1^{ij}}{4}\left[d_{i^+j^+}^n - d_{i^+j^-}^n - d_{i^-j^+}^n + d_{i^-j^-}^n\right]$$
$$+\sum_{i=1}^{3}\gamma_2^i\left[d_{i^+}^{n+1} - d^n + d^{n+1} - d_{i^-}^n\right] - r\frac{d^n + d^{n+1}}{2}. \tag{16}$$

NUMERICAL RESULTS AND EXPERIMENTAL STUDY OF CONVERGENCE

We now present numerical results for two particular cases of the implementation of the ADE scheme to Black-Scholes pricing models. In particular, we show the results for the price of a Spread option depending on two underlying assets S_1 and S_2 and a three-dimensional European Call Option on three underlying assets S_1, S_2 and S_3. For both cases we show illustrations of the obtained price surfaces and experimental convergence rates.

Two Dimensional Black-Scholes Model

We denote the Black-Scholes price for a spread option by $V(S_1, S_2, \tau)$ where $\tau = T - t$ is the time to maturity T. Recall that the payoff of a spread option is

$$V(S_1, S_2, 0) = \max(S_1 - S_2 - K, 0)$$

where $K \in \mathbb{R}^+$ denotes the strike price. The boundary conditions are given by:

$$V(S_1, 0, \tau) = BS_{1d}(S_1, \tau), \quad S_1, \tau \in \mathbb{R}^+,$$

$$V(0, S_2, \tau) = 0, \quad S_2, \tau \in \mathbb{R}^+,$$

$$V(S_1^{\max}, S_2, \tau) = e^{-q_1\tau}S_1 - e^{-r\tau}(S_2 + K), \quad S_1^{\max} := S_1 \gg S_2 + K,$$

$$V(S_1, S_2^{\max}, \tau) = V_{kirk}(S_1, S_2^{\max}, \tau),$$

where $BS_{1d}(S_1, \tau)$ denotes the Black-Scholes price formula for a call option on a stock with price S and time to maturity τ and $V_{kirk}(S_1, S_2^{\max}, \tau)$ denotes the approximation in [13].

TABLE 1. Parameters in two dimensional BS model

Parameter	Value
volatility of S_2 σ_2	0.3
volatility of S_1, σ_1	0.4
correlation of S_1 and S_2, ρ	0.5
maturity time T	1 (in years)
strike price K	3
maximal stock price for S_1 $S_{1\max}$	12
maximal stock price for S_2 $S_{2\max}$	45

We choose the parameters given by Table 1 and the different grid configurations displayed in Table 2.

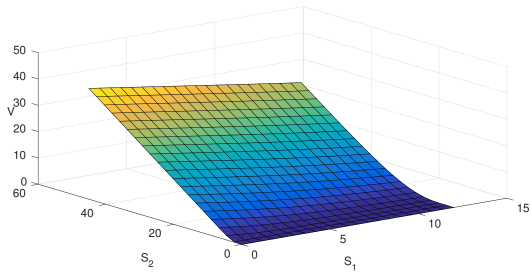


FIGURE 7. Numerical solution at the final time for two dimensional spread option on the grid with $N_1 = N_2 = 20$ space steps and $N_t = 50$ time steps

TABLE 2. Specifications of different grids.

	N_1	N_2	N_t	$d\tau/h_1^2$	$d\tau/h_2^2$
solution 1	5	5	3	0.0578	0.004
solution 2	10	10	12	0.0578	0.004
solution 3	20	20	50	0.0578	0.004
solution 4	40	40	200	0.0578	0.004
solution 5	80	80	800	0.0578	0.004
solution 6	160	160	3200	0.0578	0.004

As an example we display the numerical solution for the option price at $\tau = T$ (or equivalently $t = 0$) with a grid of $N_1 = N_2 = 20$ spatial points and $N_t = 50$ temporal points in Figure 7.

In Figure 8 we display a log-log plot of the errors in the L_2 norm (solid line) and the theoretical second order of convergence (dashed line).

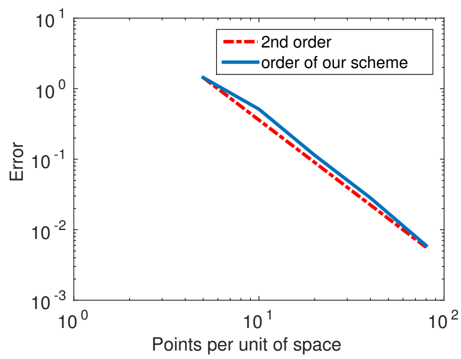


FIGURE 8. Experimental convergence analysis

Three Dimensional Black-Scholes Model

We now show the results of the implementation of the ADE to the three dimensional Black-Scholes model for the price $V(S_1, S_2, S_3, \tau)$ of a call option, where $\tau = T - t$ denotes the time to maturity T and S_i denotes the value of the underlying asset i. Recall the payoff for a call option:

$$V(S_1, S_2, S_3, 0) = \max\left((\max(S_1, S_2, S_3) - K, 0)\right).$$

with $K \in \mathbb{R}^+$ denoting the strike price. The boundary conditions are taken from the numerical solution of the 2D Black-Scholes model, BS_{2d}, implemented as outlined in d) but for a call-option payoff,

$$V(S_i = 0, t) = BS_{2d}(S_j, S_k, t), \quad i, j, k = 1, 2, 3, \quad i \neq j \neq k$$

$$V(S_i = S_i^{\max}, t) = \max(S_i^{\max} - K, 0), \quad i, j, k = 1, 2, 3.$$

In Figure 9 we show the price of the call option for a fixed value of S_3. The model parameters are in Table 3 and the grid parameters are as follows: $N_1 = N_2 = N_3 = 20$; $N_t = 50$.

TABLE 3. Parameters in three dimensional BS model

Parameter	Value
volatility of S_1 σ_1	0.4
volatility of S_2 σ_2	0.3
volatility of S_3, σ_3	0.2
correlation of S_1 and S_2, ρ	0.0
correlation of S_2 and S_3, ρ	0.0
correlation of S_1 and S_3, ρ	0.0
maturity time T	1 (in years)
strike price K	3
maximal stock price for S_1 $S_{1\max}$	12
maximal stock price for S_2 $S_{2\max}$	12
maximal stock price for S_2 $S_{2\max}$	12

Note that in this case we have a symmetric solution with respect to the underlying assets and hence fixing S_3 or any other asset would be identical.

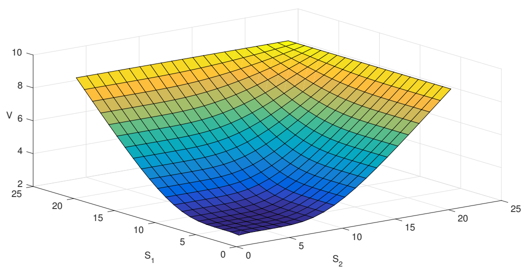


FIGURE 9. Solution of the three dimensional call option model for the fixed $S_3 = N_3/2$

TABLE 4. Usage of different grids.

N_1	N_2	N_3	N_t	$d\tau/h_1^2$	$d\tau/h_2^2$	$d\tau/h_3^2$
5	5	5	3	0.004	0.004	0.004
10	10	10	12	0.004	0.004	0.004
20	20	20	50	0.004	0.004	0.004
40	40	40	200	0.004	0.004	0.004
80	80	80	800	0.004	0.004	0.004
160	160	160	3200	0.004	0.004	0.004

Analogously to the two dimensional case, for the three dimensional case we've computed the experimental order of convergence using different grid settings *cf.* Table 4. Experimental results (Figure 10) confirm that we keep second order of convergence also in the three dimensional model.

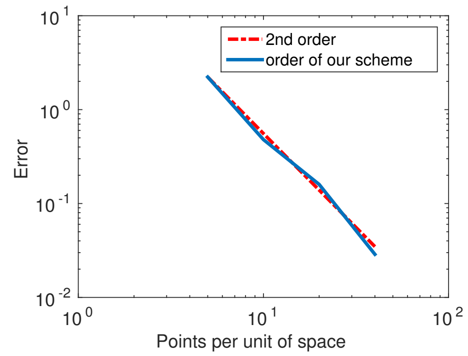


FIGURE 10. Experimental order of convergence in three dimensional call option model

INFLUENCE OF DIMENSIONALITY ON COMPUTATIONAL COMPLEXITY OF THE SCHEME

In this section we highlight the fact, where the ADE scheme has a good potential to be an effective scheme in higher dimensions. We compare it with the behavior of the classical Crank-Nicolson scheme.

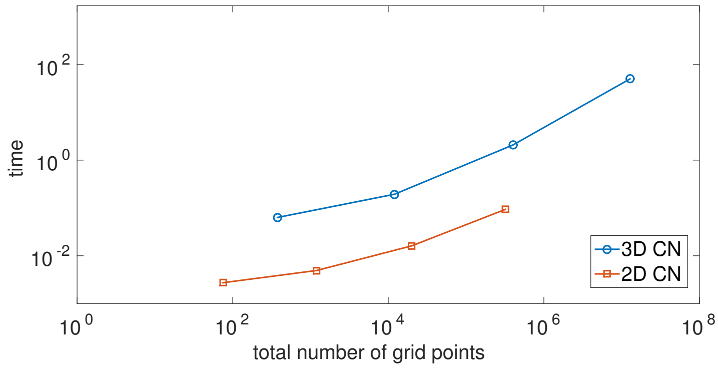


FIGURE 11. Computational complexity with respect to the total number of points in the grid for Crank-Nicolson scheme

Solution of the option price for Crank-Nicolson (CN) scheme is implemented with a lot of optimization steps, so we do not compare real time for the calculation. We focus on the fact observed in Figure 11 for CN scheme is growing with dimension. It means for the same number of total points in a grid we need more time in the 3D model as in the 2D model. The explanation is coming from the construction of the scheme. Although for the same number of total points in a grid the size of the matrix is the same, but its structure is different. For 3D more non-diagonal terms are present and to compute solution in the implicit scheme is becoming costly for higher-dimensional models.
Costs for the ADE schemes in Figure 12 for higher dimensions are not growing, even opposite, since the calculation of the explicit scheme depends only on the total number of grid points and size of the stencil.

CONCLUSION

We suggest the usage of Alternating Direction Explicit methods to numerically solve higher-dimensional partial differential equations. We implemented it for the linear 2D and 3D Black-Scholes pricing equation. The order of consistency of the implemented the ADE method is $O(k^2 + h^2)$ and this was verified experimentally.

Further studies will be made on the implementation of this scheme to higher-dimensional non-linear models

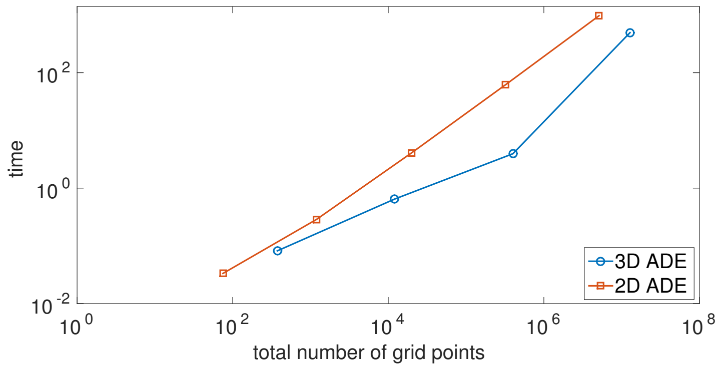


FIGURE 12. Computational complexity with respect to the total number of points in the grid for ADE scheme

Black-Scholes models, *e.g.*, of Zakamouline [14]. Also, since the ADE approach is quite suitable to parallelization, an implementation using a parallel computing environment will be envisaged.

ACKNOWLEDGEMENTS

The authors were partially supported by the European Union in the FP7-PEOPLE-2012-ITN Program under Grant Agreement Number 304617 (FP7 Marie Curie Action, Project Multi-ITN *STRIKE – Novel Methods in Computational Finance*) and by VEGA 1/0251/16 grant.

REFERENCES

[1] V. K. Saul'ev (1957) *Dokl. Akad. Nauk SSSR* **115**, 1077–1079. [in Russian]
[2] B. K. Larkin (1964) *Mathematics of Computation* **18**, 196–202.
[3] H. Barakat and J. Clark (1966) *Journal of Heat transfer* **88**, 421–427.
[4] D. J. Duffy and G. Andrea, *C# for Financial Markets* (John Wiley & Sons, 2013).
[5] D. J. Duffy, *Finite Difference Methods in Financial Engineering: a Partial Differential Equation Approach* (John Wiley & Sons, 2013).
[6] S. Leung and S. Osher, Preprint UCLA, **9**, p.2005, June 2005.
[7] K. in t Hout and C. Mishra (2013) *Applied Numerical Mathematics* **74**, 83–94.
[8] Z. Bučková, M. Ehrhardt, and M. Günther (2015) *Acta Math. Univ. Comenianae* **LXXXIV**(2), 309–325.
[9] B. F. Towler and R. Y. Yang (1978) *Computers & Chemical Engineering* **2**, 45–51.
[10] W. Zhao, X. Yang, and L. Wu (2016) *Advances in Difference Equations* **2016**, 1–18.
[11] P. Wilmott, J. Dewynne, and S. Howison, *Option Pricing: Mathematical Models and Computation* (Oxford Financial Press, 1993).
[12] K.V. Roberts and N.O. Weiss (1966) *Mathematics of Computation* **20**, 272–299.
[13] C. Alexander and A. Venkatramanan (2007), Available at SSRN 1012521, `http://ssrn.com/abstract=1012521`.
[14] V. I. Zakamouline, "Option pricing and hedging in the presence of transaction costs and nonlinear partial differential equations," in [22].
[15] N. N. Yanenko, *The Method of Fractional Steps* (Springer, 1971).
[16] D. Tavella and C. Randall, *Pricing Financial Istruments: The Finite Difference Method*, Vol. 13 (John Wiley & Sons, 2000).
[17] S. A. Piacsek and G. P. Williams (1970) *Journal of Computational Physics* **6**, 392–405.
[18] G. Pealat and D. J. Duffy (2011) *Wilmott Magazine Ltd.* **2011**(54), 54–60.
[19] Y. K. Kwok, *Mathematical Models of Financial Derivatives* (Springer-Verlag, Singapore, 1998).
[20] M. Jandačka and D. Ševčovič (2005) *Journal of Applied Mathematics* **2005**, 235–258.

[21] R. Frey and P. Patie, "Risk management for derivatives in illiquid markets: a simulation study," in *Advances in Finance and Stochastics*, (Springer, 2002), pp. 137–159.

[22] M. Ehrhardt, *Nonlinear Models in Mathematical Finance: New Research Trends in Option Pricing* (Nova Science Publishers, 2008).

[23] L. Campbell and B. Yin (2007) *Numerical Methods for Partial Differential Equations* **23**, 1429–1444.

[24] R. Company, L. Jódar, E. Ponsoda, and C. Ballester (2010) *Computers & Mathematics with Applications* **59**, 2964–2975.

[25] L. A. Bordag and A. Y. Chmakova (2007) *International Journal of Theoretical and Applied Finance* **10**, 1–21.

[26] G. Barles and H. M. Soner (1998) *Finance and Stochastics* **2**, 369–397.