Adaptive wavelet method for pricing two-asset Asian options with floating strike

Dana Černá

ARTICLES YOU MAY BE INTERESTED IN

Fourth order scheme for wavelet based solution of Black-Scholes equation
AIP Conference Proceedings 1910, 030004 (2017); https://doi.org/10.1063/1.5013963

Numerical solution of the right boundary condition inverse problem for the Black-Scholes equation
AIP Conference Proceedings 1910, 030008 (2017); https://doi.org/10.1063/1.5013967

A DG approach to the numerical solution of the Stein-Stein stochastic volatility option pricing model
AIP Conference Proceedings 1910, 030006 (2017); https://doi.org/10.1063/1.5013965
Adaptive Wavelet Method for Pricing Two-Asset Asian Options with Floating Strike

Dana Černá1,a)

1Department of Mathematics and Didactics of Mathematics, Technical University of Liberec, Studentská 2, 461 17 Liberec, Czech Republic.

a)Corresponding author: dana.cerna@tul.cz

Abstract. Asian options are path-dependent option contracts which payoff depends on the average value of the asset price over some period of time. We focus on pricing of Asian options on two assets. The model for pricing these options is represented by a parabolic equation with time variable and three state variables, but using substitution it can be reduced to the equation with only two state variables. For time discretization we use the $\theta$-scheme. We propose a wavelet basis that is adapted to boundary conditions and use an adaptive scheme with this basis for discretization on the given time level. The main advantage of this scheme is small number of degrees of freedom. We present numerical experiments for the Asian put option with floating strike and compare the results for the proposed adaptive method and the Galerkin method.

INTRODUCTION

Our aim is to develop and implement robust and efficient numerical methods for solving option pricing problem. We have already proposed and implemented a wavelet method for option pricing in our previous papers [1, 2, 3, 4]. We used the Black-Scholes model and tested the performance of the wavelet method with respect to the choice of a wavelet basis, compared an isotropic and an anisotropic approach, studied the convergence rate of the method and proposed a construction of a wavelet basis such that the Black-Scholes operator represented in this basis is sparse. Then we applied adaptive wavelet method on more general Heston stochastic volatility model for calculating the price of options.

In this paper, we focus on Asian options. Recall that a put option gives its holder the right, but not the obligation, to sell a group of underlying assets at a specific price on a certain date. Similarly, a call option gives its holder the right, but not the obligation, to buy a group of underlying assets at a specific price on a certain date. In the case of Asian options the payoff is determined by the average value of the asset prices over the prescribed period of time. We focus on two-asset Asian options with floating strike, i.e. the strike is not fixed but is determined by continuous arithmetic average of the price of the basket of two underlying assets over the prescribed period of time.

Wavelet methods have been already successfully used for solving option pricing problems, see e.g. [5, 6, 7, 8, 9], but up to our knowledge the method based on wavelets has not been implemented yet for pricing of multi-asset Asian options. In our previous papers we used an adaptive wavelet method that is a modification of the method from [10]. In this paper, we use a different approach that is simpler and was more efficient in our numerical experiments. The quantitative properties of any wavelet method crucially depend on the choice of a wavelet basis. Therefore, a construction of an appropriate wavelet basis is still an issue [3, 11, 12, 13]. In this paper, we adapt a linear spline wavelet basis with two vanishing moments from [14, 15] to the rectangle and boundary conditions representing the model and use it in the scheme. The main advantage of the scheme is the small number of parameters representing the solution. We present numerical experiments for the Asian put option with real market data, compare the results for the proposed adaptive method and the Galerkin method with the same wavelet basis, and study the convergence rate of the method.
ASIAN OPTIONS WITH FLOATING STRIKE

Let \( S_1 > 0 \) and \( S_2 > 0 \) represent prices of the assets, \( \alpha_1 \) and \( \alpha_2 \) are weights that sum up to 1, \( t \) represents time to maturity, and \([0, T] \) is the prescribed period. Then the continuous arithmetic average

\[
A = \frac{1}{T} \int_0^T \alpha_1 S_1 (t) + \alpha_2 S_2 (t) \, dt
\]

represents the floating strike of the option. We assume that the asset prices \( S_1 \) and \( S_2 \) follow the geometric Brownian motion and that the risk-free interest rate \( r \) and the asset volatilities \( \sigma_1 \) and \( \sigma_2 \) are constants. Let \( \rho \) be the correlation factor between asset prices and let \( V(S_1, S_2, A, t) \) denote the value of the Asian option with the floating-strike \( A \) at time to maturity \( t \), if the prices of the underlying assets are \( S_1 \) and \( S_2 \). Based on the ideas from [16, 17] it was derived in [18] that \( V(S_1, S_2, A, t) \) can be computed as the solution of the equation:

\[
\frac{\partial V}{\partial t} - \mathcal{L}(V) = 0, \quad t \in (0, T),
\]

where the operator \( \mathcal{L} \) is given by

\[
\mathcal{L}(V) = \frac{\sigma_1^2 S_1^2}{2} \frac{\partial^2 V}{\partial S_1^2} + \rho \sigma_1 \sigma_2 S_1 S_2 \frac{\partial^2 V}{\partial S_1 \partial S_2} + \frac{\sigma_2^2 S_2^2}{2} \frac{\partial^2 V}{\partial S_2^2} + r S_1 \frac{\partial V}{\partial S_1} + r S_2 \frac{\partial V}{\partial S_2} + \frac{\alpha_1 S_1 + \alpha_2 S_2 - A}{T-t} \frac{\partial V}{\partial A} - r V.
\]

The initial conditions are given by

\[
V(S_1, S_2, A, 0) = \begin{cases} \max(\alpha_1 S_1 + \alpha_2 S_2 - A, 0) & \text{for call option,} \\ \max(\alpha_1 S_1 - \alpha_2 S_2, 0) & \text{for put option.} \end{cases}
\]

We define \( x_1 = S_1/A, x_2 = S_2/A \) and \( U(x_1, x_2, t) = V(S_1, S_2, A, t)/A \) and substitute it into (2). We obtain [18]:

\[
\frac{\partial U}{\partial t} - \mathcal{L}_r(U) = 0, \quad t \in (0, T),
\]

with

\[
\mathcal{L}_r(U) = \frac{\sigma_1^2 x_1^2}{2} \frac{\partial^2 U}{\partial x_1^2} + \rho \sigma_1 \sigma_2 x_1 x_2 \frac{\partial^2 U}{\partial x_1 \partial x_2} + \frac{\sigma_2^2 x_2^2}{2} \frac{\partial^2 U}{\partial x_2^2} + \left( r - \frac{\alpha_1 x_1 + \alpha_2 x_2 - 1}{T-t} \right) x_2 \frac{\partial U}{\partial x_1} + \left( \frac{\alpha_1 x_1 + \alpha_2 x_2 - 1}{T-t} - r \right) U.
\]

The initial conditions (4) are transformed into

\[
U(S_1, S_2, 0) = \begin{cases} \max(\alpha_1 x_1 + \alpha_2 x_2 - 1, 0) & \text{for call option,} \\ \max(1 - \alpha_1 x_1 - \alpha_2 x_2, 0) & \text{for put option.} \end{cases}
\]

We choose maximal values \( x_1^{\text{max}} \) and \( x_2^{\text{max}} \) large enough and approximate the unbounded domain \([0, \infty) \times [0, \infty)\) by a domain \( \Omega = \left(0, x_1^{\text{max}}\right) \times \left(0, x_2^{\text{max}}\right) \). We denote the parts of the boundary of the domain \( \Omega \) by

\[
\begin{align*}
\Gamma_1 &= \{(x_1,0), x_1 \in (0,x_1^{\text{max}})\} \cup \{(0,x_2), x_2 \in (0,x_2^{\text{max}})\}, \\
\Gamma_2 &= \{x_1^{\text{max}}, x_2\}, x_2 \in (0,x_2^{\text{max}})\} \cup \{x_1, x_1^{\text{max}}\}, x_1 \in (0,x_1^{\text{max}})\}
\end{align*}
\]

and set boundary conditions as

\[
U(x_1, x_2, t) = \begin{cases} 0 & \text{on } \Gamma_1 \text{ for call option,} \\ 0 & \text{on } \Gamma_2 \text{ for put option.} \end{cases}
\]

In the case of a put option we have do-nothing boundary condition on \( \Gamma_1 \), for details see [18]. Due to (10) the solution \( U \) belongs to the space \( V = \{v \in H^1(\Omega) : v = 0 \text{ on } \Gamma_2\} \), where \( H^1(\Omega) \) denotes the Sobolev space. In the following, we present the discretization for a put option, the approach for a call option is similar.
For discretization in time we use the $\theta$-scheme. Let $M \in \mathbb{N}$, $\tau = T/M$, $t_l = l \tau$, $l = 0, \ldots, M$, and denote $U_l(x_1, x_2) = U(x_1, x_2, t_l)$. The $\theta$-scheme has the form:

$$
\frac{U_{l+1} - U_l}{\tau} - \theta \mathcal{L}_{l+1} (U_{l+1}) - (1 - \theta) \mathcal{L}_l (U_l) = 0,
$$

where $\theta \in [0, 1]$ and $l = 0, \ldots, M - 1$. The choice $\theta = 1$ corresponds to the backward Euler scheme and the choice $\theta = 0.5$ corresponds to the Crank-Nicolson scheme.

We briefly review a concept of a wavelet basis, for more details one can see e.g. [19]. Let us consider a separable Hilbert space $H$ with the norm $\|\cdot\|_H$. Let $\mathcal{J}$ be an index set such that each index $\lambda \in \mathcal{J}$ takes the form $\lambda = (j, k)$ and $|\lambda| := j$ denotes a scale or a level and $k$ represents a spatial location. 

**Definition 1.** A family $\Psi = \{\psi_{\lambda}, \lambda \in \mathcal{J}\}$ is called a wavelet basis of the space $H$ if it satisfies the following conditions:

1. $\Psi$ is a Riesz basis for $H$, i.e. the closure of the span of $\Psi$ is $H$ and there exist constants $c, C \in (0, \infty)$ such that

$$
c \|b\|_2 \leq \left\| \sum_{\lambda \in \mathcal{J}} b_{\lambda} \phi_{\lambda} \right\|_H \leq C \|b\|_2,
$$

for all $b = (b_{\lambda})_{\lambda \in \mathcal{J}}$ such that $\|b\|_2^2 = \sum_{\lambda \in \mathcal{J}} b_{\lambda}^2 < \infty$.

2. The functions are local in the sense that $\text{diam supp } \psi_{\lambda} \leq C 2^{-|\lambda|}$ for all $\lambda \in \mathcal{J}$, and at a given level $j$ the supports of only finitely many wavelets overlap at any point $x$.

3. There exist $J_2 \subset \mathcal{J}$ and $L \geq 1$ such that wavelets have $L$ vanishing moments, i.e.

$$
\int_{\text{supp } \psi_{j,l}} p(x) \psi_{j,l}(x) = 0, \quad (j, l) \in J_2,
$$

for all polynomials $p$ up to a degree $L - 1$.

Wavelet basis $\Psi$ has typically a hierarchical structure

$$
\Psi = \Phi_{J_0} \cup \bigcup_{j=J_0}^{\infty} \Psi_j,
$$

The functions $\phi_{j,k}$ from $\Phi_j$ are called scaling functions and the functions $\psi_{j,k}$ from $\Psi_j$ are called wavelets on the level $j$. Wavelets $\psi_{j,k}$ in the inner part of the interval are typically translations and dilations of a function $\psi$ also called wavelet, i.e.

$$
\psi_{j,k}(x) = 2^{-j/2} \psi \left( 2^{j} x - k \right),
$$

and functions near the boundary are derived from functions called boundary wavelets constructed such that they satisfy the boundary conditions.

First, we construct a wavelet basis for the spaces $V_l = \left\{ v \in L^2(0, 1) : v(1) = 0 \right\}$ using the similar approach as in [14, 15]. We define scaling functions as linear B-splines. Let $\phi$ and $\phi_0$ be defined by:

$$
\phi(x) = \begin{cases} x, & x \in [0, 1], \\ 2 - x, & x \in [1, 2], \\ 0, & \text{otherwise}, \end{cases}, \quad \phi_0(x) = \begin{cases} 1 - x, & x \in [0, 1], \\ 0, & \text{otherwise}. \end{cases}
$$
For \( j \geq 2 \) and \( x \in [0, 1] \) we set
\[
\phi_{j,k}^1(x) = 2^{j/2} \phi_k(2^j x), \quad \phi_{j,k}^j(x) = 2^{j/2} \phi_k(2^j x - k + 2), \quad k = 2, \ldots, 2^j, \tag{18}
\]
and
\[
\Phi_j = \{ \phi_{j,k}^1, k = 1, \ldots, 2^j \}. \tag{19}
\]
We define a wavelet \( \psi \) and a boundary wavelets \( \psi_{b1} \) and \( \psi_{b2} \) as
\[
\psi(x) = -\frac{1}{4} \phi(2x) - \frac{1}{2} \phi(2x - 1) + \frac{3}{2} \phi(2x - 2) - \frac{1}{2} \phi(2x - 3) - \frac{1}{4} \phi(2x - 4), \tag{20}
\]
\[
\psi_{b1}(x) = \frac{3}{2} \phi_k(2x) - \frac{9}{8} \phi(2x) + \frac{1}{2} \phi(2x - 1) - \frac{1}{8} \phi(2x - 2),
\]
\[
\psi_{b2}(x) = \frac{3}{2} \phi(2x) - \phi(2x - 1) - \frac{1}{2} \phi(2x - 2).
\]
Then \( \text{supp } \psi = [0, 3] \), \( \text{supp } \psi_{b1} = \text{supp } \psi_{b2} = [0, 2] \), and these wavelets have two vanishing moments. For \( j \geq 2 \) and \( x \in [0, 1] \) we define
\[
\psi_{j,k}^1(x) = 2^{j/2} \psi_{b1}(2^j x), \tag{21}
\]
\[
\psi_{j,k}^j(x) = 2^{j/2} \psi(2^j x - k + 2), k = 2, \ldots, 2^j - 1,
\]
and
\[
\Psi_j = \{ \psi_{j,k}^1, k = 1, \ldots, 2^j \}. \tag{22}
\]
The graphs of scaling functions and wavelets on the level \( j = 2 \) are displayed in Figure 1. Then the set
\[
\Psi^1 = \Phi_2^1 \cup \bigcup_{j=2}^\infty \Psi_j
\]
(23)
is a wavelet basis of the space \( V_1 \) equipped with the \( L^2 \)-norm, see Proof of Theorem 1.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{scaling_functions}
\caption{Scaling functions \( \phi_{j,k}^1 \) (left) and wavelets \( \psi_{j,k}^1 \) (right) on the coarsest level \( j = 2 \).}
\end{figure}

Now, using a simple linear transformation and the tensor product, we obtain a wavelet basis of the space \( V \). For \( x_1 \in \left[0, x_1^{\max}\right] \) let us define
\[
\phi_{j,k}^{1}(x_1) = \phi_{j,k}^1 \left( \frac{x_1}{x_1^{\max}} \right), \quad \psi_{j,k}^{1}(x_1) = \psi_{j,k}^1 \left( \frac{x_1}{x_1^{\max}} \right), \tag{24}
\]
and similarly for \( x_2 \in \left[0, x_2^{\max}\right] \) let
\[
\phi_{j,k}^{2}(x_2) = \phi_{j,k}^2 \left( \frac{x_2}{x_2^{\max}} \right), \quad \psi_{j,k}^{2}(x_2) = \psi_{j,k}^2 \left( \frac{x_2}{x_2^{\max}} \right). \tag{25}
\]
Finally, we define

\[ \Psi = \left\{ \phi_{2k}^{i_1}(x_1) \cdot \phi_{2j}^{i_2}(x_2), k, l \in J_2 \right\} \cup \left\{ \psi_{2k}^{i_1}(x_1) \cdot \psi_{2j}^{i_2}(x_2), k \in J_2, j \geq 2, l \in J_1 \right\} \cup \left\{ \psi_{2k}^{i_1}(x_1) \cdot \psi_{2j}^{i_2}(x_2), j \geq 2, k \in J_J, i, j \geq 2, k \in J_i, l \in J_l \right\}, \]

\( J_j = \{1, \ldots, 2^j\} \).

**Theorem 2.** The set \( \Psi \) defined by (26) is a wavelet basis of the space

\[ \mathcal{V} = \left\{ v \in L^2(\Omega) : v = 0 \text{ on } \Gamma_2 \right\} \]

equipped with the \( L^2 \)-norm. The set \( \Psi \), when normalized with respect to the \( H^1 \)-norm, is a wavelet basis for the space \( \mathcal{V} \) equipped with the \( H^1 \)-norm.

**Proof.** The proof that \( \Psi^i \) is a wavelet basis of the space \( \mathcal{V}^i \) equipped with the \( L^2 \)-norm follows the lines of the Proof of Theorem 14 and Corollary 1 in [14]. Since for \( i = 1, 2 \) we have

\[ \left\| \sum_{k \in J_2} b_{k} \phi_{2k}^{i_1} + \sum_{j=2}^{\infty} \sum_{k \in J_2} b_{j,k} \phi_{j,k}^{i_2} \right\|_{L^2} = x_{i}^{\max} \left\| \sum_{k \in J_2} b_{k} \phi_{2k}^{i_1} + \sum_{j=2}^{\infty} \sum_{k \in J_2} b_{j,k} \phi_{j,k}^{i_2} \right\|_{L^2}, \]

(28)

the set

\[ \Psi^i = \left\{ \phi_{j,k}^{i_1}, k \in J_2 \right\} \cup \left\{ \psi_{j,k}^{i_2}, k \in J_J, j \geq 2 \right\} \]

(29)

is a wavelet basis of the space

\[ \mathcal{V}_{i_1} = \left\{ v \in L^2(0, x_{i}^{\max}) : v(x_{i}^{\max}) = 0 \right\} \]

(30)

equipped with the \( L^2 \)-norm. It is known [20] that the tensor product of Riesz bases of the spaces \( \mathcal{V} \) and \( \mathcal{W} \) is a Riesz basis of the space \( \mathcal{V} \otimes \mathcal{W} \) and therefore \( \Psi = \Psi^1 \otimes \Psi^2 \) is a Riesz basis of the space \( \mathcal{V} \). The fact that \( \Psi \), when normalized with respect to the \( H^1 \)-norm, is a wavelet basis for the space \( \mathcal{V} \) equipped with the \( H^1 \)-norm is a consequence of the smoothness of basis functions and vanishing moments, see e.g. [19]. □

The graphs of three wavelets on the level \( j = 2 \) are displayed in Figure 2.

**ADAPTIVE WAVELET SCHEME**

We use an adaptive wavelet method for discretization with respect to the variables \( x_1 \) and \( x_2 \). In our previous papers we used an adaptive method that was a modification of the method from [10]. In this paper, we use a simpler approach, because it was more efficient in our numerical experiments.

First, we derive a variational formulation but instead of turning to a finite-dimensional approximation, using the suitable wavelet basis the continuous problem is transformed into an infinite-dimensional \( \ell^2 \)-problem. Let \( \Psi = \)

\[ \begin{cases}
\phi_{2k}^{i_1}(x_1) \cdot \phi_{2j}^{i_2}(x_2), k, l \in J_2, i, j \geq 2, k \in J_i, l \in J_l, \\
\psi_{2k}^{i_1}(x_1) \cdot \psi_{2j}^{i_2}(x_2), j \geq 2, k \in J_J, i, j \geq 2, k \in J_i, l \in J_l.
\end{cases} \]
\( \{ \psi_\lambda, \lambda \in \mathcal{J} \} \) be a wavelet basis for \( V \) from the previous section and let \( \mathbf{u}^i = \left[ u^i_\lambda \right]_{\lambda \in \mathcal{J}} \) be the coefficients of the function \( U_i \) from (12) in a basis \( \Psi \), i.e.

\[
U_i = \sum_{\lambda \in \mathcal{J}} u^i_\lambda \psi_\lambda.
\]

Setting

\[
A^i_{\mu, \lambda} = \left( \frac{\psi_\lambda, \psi_\mu}{\tau} \right) - \theta a_{i+1} \left( \psi_\lambda, \psi_\mu \right), \quad \mu, \lambda \in \mathcal{J},
\]

\[
f^i_\mu = (1 - \theta) a_i \left( U_i, \psi_\mu \right) + \frac{(U_i, \psi_\mu)}{\tau}, \quad \mu \in \mathcal{J},
\]

we obtain the \( l^2 \)-problem

\[
A^i \mathbf{u}^{i+1} = \mathbf{f}^i.
\]

The matrix \( A^i \) has so-called finger structure. It is generally not sparse but quasi sparse. It means that it can be approximated by a sparse matrix. We use the standard Jacobi diagonal preconditioner and solve the problem (33) by the method of generalized residuals (GMRES). Let \( G^i \) be defined by

\[
G^i_{\mu, \lambda} = \left( \frac{\psi_\lambda, \psi_\mu}{\tau} \right) + (1 - \theta) a_i \left( \psi_\lambda, \psi_\mu \right), \quad \mu, \lambda \in \mathcal{J}.
\]

Then \( \mathbf{f}^i = G^i \mathbf{u}^i \).

The algorithm comprises the following steps:

1. Choose \( \theta \), time step \( \tau \), and the number of basis functions \( N \).
2. Compute the vector of coefficients \( \mathbf{u}^0 \) for the function \( U_0 \) and do \( \mathbf{u}^0 = \text{COARSE}(\mathbf{u}^0, N) \).
3. For \( l = 0, 1, 2, \ldots, M - 1 \) compute

   \[
   \mathbf{f}^i = G^i \mathbf{u}^i,
   \]

   \[
   \mathbf{u}^{i+1} = \text{GMRES}(A^i, \mathbf{f}^i, \mathbf{u}^i),
   \]

   \[
   \mathbf{u}^{i+1} = \text{COARSE}(\mathbf{u}^{i+1}, N).
   \]

4. Compute the approximate solution \( U_M \) using (31).

In the algorithm, \( \mathbf{u}^{i+1} = \text{GMRES}(A^i, \mathbf{f}^i, \mathbf{u}^i) \), means that \( \mathbf{u}^{i+1} \) is the solution of the system of linear algebraic equations with the matrix \( A^i \) and the right-hand side \( \mathbf{f}^i \) using GMRES method with initial vector \( \mathbf{u}^i \). The routine \( \mathbf{u}^{i+1} = \text{COARSE}(\mathbf{u}^{i+1}, N) \) insists in thresholding, i.e. we take \( N \) entries from the vector \( \mathbf{u}^{i+1} \) that are largest in the absolute value and set to zero the others. Thus the output parameter \( \mathbf{u}^{i+1} \) contains \( N \) nonzero entries. Each iteration of GMRES requires multiplication of the infinite-dimensional matrix with a finitely supported vector. It is computed approximately by the method from [21].

Since we work with the sparse representation of the right-hand side and the sparse representation of the vector representing the solution, the method is adaptive. It is known that the coefficients in the wavelet basis are small in regions where the function is smooth and large in regions where the function has some singularity.

**NUMERICAL EXAMPLE**

We denote the exact solution of (2) at time to maturity \( t_i \) by \( V^i_1 \), an approximate solution computed at time to maturity \( t_i \) by \( V^i_1 \) and we compute the relative errors

\[
e_{\infty} = \frac{\| V_1 - V_1^i \|_{\infty}}{\| V_1^i \|_{\infty}}, \quad e_2 = \frac{\| V_1 - V_1^i \|_2}{\| V_1^i \|_2},
\]

where \( \| \cdot \|_{\infty} \) denotes the \( L^\infty(\Omega) \)–norm. In the case of the Galerkin method the errors \( e_{\infty} \) and \( e_2 \) typically depend on the step size \( h \): \( e_{\infty} \approx C_1 h^p \) with the constant \( C_1 \) independent on \( h \). Since \( h \approx C_2 N^{-1/d} \), where \( d \) denotes the dimension and \( N \) is the number of basis functions, we can rewrite the error estimate as

\[
e_{\infty} \approx C_3 N^{-\frac{2}{d}}, \quad i = 2, \infty.
\]
We compute the estimate of \( r_j \) numerically. If \( e_j(N_j) \) is a computed error for \( N_j \) functions, \( j = 1, 2 \), then the approximate order of convergence is
\[
r_j \approx \frac{d \log (e_j(N_j)/e_j(N_2))}{\log(N_2/N_1)}.
\]
It is well known that \( e_j \approx C_1 N_j^{-3} \approx C_2 N_j^{-1} \) for linear approximation under the assumption that the approximated function is smooth enough. Thus the optimal rate of convergence is \( r_j = 2 \) for \( i = 2, \infty \).

**Example 3.** We solve the same option pricing problem as in [18]. We consider an Asian put option written on exchange rates of EUR and USD, both with respect to GBP. We assume that basket contains 60% EUR and 40% USD, i.e. \( a_1 = 0.6 \) and \( a_2 = 0.4 \). The parameters are computed from real market data: time to maturity is one month, i.e. \( T = 1/12 \), risk-free interest rate \( r = 0 \), volatilities \( \sigma_1 = 0.1 \) and \( \sigma_2 = 0.15 \), the correlation parameter is \( \rho = 0.45 \). We set \( \theta = 0.5 \), i.e. we use the Crank-Nicolson scheme, \( x_1^{\max} = x_2^{\max} = 2 \) and \( A = 0.8 \). The initial condition \( V(S_1, S_2, A, 0) \) and the resulting solution \( V(S_1, S_2, A, t) \) for \( t = 10/365 \) and \( t = 20/365 \), i.e. 10 and 20 days to maturity, respectively, are displayed in Figure 3. Note that the initial function \( V(S_1, S_2, A, 0) \) is not smooth.

![Figure 3](image-url)

**FIGURE 3.** The initial condition \( V(S_1, S_2, A, 0) \) (left) and the resulting solution \( V(S_1, S_2, A, t) \) for \( t = 10/365 \) (middle) and \( t = 20/365 \) (right).

**Example 4.** We study the errors and convergence rate of the method on the model example. We compute the value \( V \) of the Asian put option on two assets with parameters: \( \sigma_1 = \sigma_2 = 0.15 \), \( \rho = 1 \), \( A = 0.8 \), \( r = 0.016 \), \( T = 1/12 \), \( a_1 = a_2 = 0.5 \), \( T = 1/365 \), \( \theta = 2 \). Since the correlation parameter is \( \rho = 1 \) and \( \sigma_1 = \sigma_2 \) it is possible to reduce the problem to a simpler problem with one state variable. Let \( W(S, t) \) be the value of the option on one asset with parameters \( \sigma_1, A, r \) and \( T \) as above. Then \( V(S_1, S_2, t) = W(a_1S_1 + a_2S_2, t) \).

We compute \( W \) with high accuracy as the solution of the parabolic equation with one spatial variable similar as (5). We use the Galerkin method with the large number of basis functions \( N = 8194 \), small time step \( \tau = 1/7300 \), and use quadruple precision. Since we computed \( W \) with a high accuracy, we set \( V^{\text{ex}}(S_1, S_2, t) = W(a_1S_1 + a_2S_2, t) \). Convergence history for the Galerkin method and the proposed adaptive wavelet method is shown in Table 1.

<table>
<thead>
<tr>
<th>( N )</th>
<th>( e_\text{oo} )</th>
<th>( r_\text{oo} )</th>
<th>( e_2 )</th>
<th>( r_2 )</th>
<th>( N )</th>
<th>( e_\text{oo} )</th>
<th>( r_\text{oo} )</th>
<th>( e_2 )</th>
<th>( r_2 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>0.0946</td>
<td>0.0981</td>
<td>100</td>
<td>0.0161</td>
<td>100</td>
<td>0.0138</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>64</td>
<td>0.0428</td>
<td>1.14</td>
<td>0.0343</td>
<td>1.52</td>
<td>200</td>
<td>0.0113</td>
<td>1.02</td>
<td>0.0072</td>
<td>1.85</td>
</tr>
<tr>
<td>256</td>
<td>0.0170</td>
<td>1.33</td>
<td>0.0130</td>
<td>1.40</td>
<td>300</td>
<td>0.0078</td>
<td>1.84</td>
<td>0.0047</td>
<td>2.10</td>
</tr>
<tr>
<td>1024</td>
<td>0.0079</td>
<td>1.11</td>
<td>0.0059</td>
<td>1.14</td>
<td>400</td>
<td>0.0042</td>
<td>4.26</td>
<td>0.0030</td>
<td>3.22</td>
</tr>
<tr>
<td>4096</td>
<td>0.0023</td>
<td>1.78</td>
<td>0.0019</td>
<td>1.63</td>
<td>500</td>
<td>0.0041</td>
<td>0.35</td>
<td>0.0024</td>
<td>2.02</td>
</tr>
<tr>
<td>16384</td>
<td>0.0010</td>
<td>1.20</td>
<td>0.0008</td>
<td>1.24</td>
<td>600</td>
<td>0.0039</td>
<td>0.38</td>
<td>0.0019</td>
<td>2.29</td>
</tr>
</tbody>
</table>

We can see that the adaptive wavelet method required significantly smaller number of basis functions. Moreover the optimal convergence rate was not achieved for the standard Galerkin method, but in the case of the adaptive wavelet method the optimal convergence rate with respect to the \( L^2 \)-norm was achieved.
CONCLUSION

We extended our research to pricing Asian options with floating strike on a basket of two assets. We constructed a piecewise linear wavelet basis with two vanishing moments on the rectangle that is adapted to the prescribed boundary conditions. We proposed and implemented an adaptive wavelet method with this basis for a numerical solution of the partial differential equation representing the model. We presented numerical example for the Asian put option and compared the results with the wavelet Galerkin method. The main advantage of the proposed method is a small number of parameters representing the solution with desired accuracy. Moreover, in numerical experiments the optimal convergence rate with respect to the $L^2$–norm was achieved for the proposed adaptive wavelet method.

ACKNOWLEDGMENTS

This work was supported by grant No. GA16-09541S of the Czech Science Foundation.

REFERENCES