A tree-based method to price American Options in the Heston Model

Michel Vellekoop* and Hans Nieuwenhuis†

March 1, 2012

Abstract

We develop an algorithm to price American options on assets that follow the stochastic volatility model defined by Heston. We use an approach which is based on a modification of an explicitly defined stock price tree where the number of nodes grows quadratically in the number of time steps. We show in a number of numerical tests that we get accurate results in a fast manner. Features which are important for the practical use of option pricing algorithms, such as the incorporation of cash dividends and a term structure of interest rates, can easily be incorporated.

The version implemented in PREMIA is an improved version of the algorithm presented here. Note that the number of timesteps and number of grid points for stock and volatility can be freely specified in the PREMIA implementation but that the theorem in the paper states explicit conditions that need to be satisfied to obtain convergence when increasing these steps.

Premia 14

1 Introduction

The most popular model for equity option pricing under stochastic volatility seems to be the model defined by Heston [9]:

\[
\begin{align*}
    dS_t & = \mu S_t dt + \sqrt{V_t} S_t dW_t^1 \\
    dV_t & = \kappa (\theta - V_t) dt + \omega \sqrt{V_t} dW_t^2
\end{align*}
\]

In this model for the stock price process \(S\) and squared volatility process \(V\) we denote by \(W^1\) and \(W^2\) standard Brownian Motion processes that may have nonzero correlation, and \(\mu, \kappa, \theta\) and \(\omega\) are fixed and known strictly positive parameters.

The popularity of this model can be explained to a large extent by the possibility to derive option price formulas for European options in closed form using Fourier transforms [3]. This closed form actually requires the numerical approximation of a certain integral, and some care has to be taken when dealing with the complex logarithm in this integral (see the excellent recent papers by Lord, Kahl and Jäckel [16, 12] and Albracher et al [1] on this issue). But since this method is still a lot faster than methods in which the corresponding partial differential equation is solved

*Corresponding Author: Financial Engineering Laboratory, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands. Tel +31 53 489 2087, Fax +31 53 489 3800, m.h.vellekoop@math.utwente.nl
†University of Groningen, Faculty of Economics, PO Box 800, 9700 AV Groningen.
numerically, calibration of the Heston model to European options is a lot easier than calibration of other stochastic volatility models which do not admit closed form pricing functions.

For American options, or asset dynamics which involve the payment of cash dividends at fixed dividend dates instead of continuous dividend payments, such closed form pricing functions do not exist. Methods to determine the prices of such options in the Heston model therefore typically focus on solving the Heston partial differential equation under early exercise constraints.

Clarke and Parrott [4] formulate the American put pricing problem as a linear complementarity problem and use an implicit finite difference scheme combined with a multigrid procedure based on earlier work by Brandt and Cryer [2] to find price approximations. Their method was further improved by Oosterlee [18] who used Fourier analysis methods to optimize the smoothing procedure in the multigrid procedure. Forsyth, Vetzal and Zvan have used a penalty method to deal with the early exercise constraint, and showed that in the limit this is in fact equivalent to a linear complementarity formulation [21]. Ikonen and Toivanen [11] used operator splitting methods to price American options in the Heston model. They find that the error induced by the splitting does not reduce the error of accuracy when compared to Crank-Nicholson methods without splitting.

In this paper we will develop an alternative method which is based on building a discrete time process that approximates the dynamics of the Heston model. Not only will we show that American options and options including cash dividends can be priced very fast on such trees, but we will also argue that the structure is very transparent, and therefore particularly easier to implement.

Approximating trees for the Heston model have been considered before, for example in the paper by Leisen [15]. In that paper a multinomial tree is constructed with 4 successor nodes per timestep for the asset price process, and 2 successor nodes for the volatility process. The goal behind the construction of this tree is weak convergence to the correct joint distribution of asset price and volatility when the number of timesteps grows to infinity. However, Florescu and Viens [6] mention that since a transformation to eliminate the volatility is used, weak convergence cannot be proven along the lines suggested in the paper. They therefore propose a different algorithm [7], which combines a tree with Monte Carlo simulations based on a finite number of particles between time steps, but this is computationally expensive.

Hilliard and Schwartz [10] use a transformation of the \( S \) and \( V \) variables to end up with stochastic differential equations for the transformed processes which lead to recombining discretizations in a two-dimensional tree. The transformation itself is based on the one proposed by Nelson and Ramaswamy [17]. However, their method has some numerical problems since negative probabilities may arise in the discretized model if certain bounds on the correlation coefficient are not satisfied. Moreover, since a lot of computational time is used for calculations in parts of the state space which have a low probability, the method is not always very efficient.

Almost all tree methods proposed in the literature use a grid in which both the distance between (transformed) asset price grid points and (transformed) volatility grid points are of the order \( \sqrt{\Delta t} \), where \( \Delta t \) denotes the discrete time step. To make sure that the stochastic process defined on the tree converges (weakly) to the correct continuous time process, second order moment matching conditions are used and this generates equations that need to be satisfies by the single time step transition probabilities under the riskneutral measure. In the detailed implementation described by Leisen [15] this leads to a tree with 8 successor nodes per node on a fixed grid.

An exception is the paper by Guan and Xiaoliang [8], which is based on a scheme introduced in earlier work by Finucane and Thomas [5]. In the Guan and Xiaoliang paper, a tree is constructed in which the problem that nodes do not recombine is solved by an interpolation/extrapolation scheme. A grid is define, and whenever option prices are needed that or not on the grid, interpolation or extrapolation based on neighbouring points is applied. Negative option prices may occur in the tree, but these are set to zero. The method is shown to work well for short maturity options, where the extrapolation errors are not yet too large.

In this paper, we propose an extension to this method in which we use different interpolation schemes in order to exploit the regularity of the option payoff function. We define a flexible grid for the logarithm of the stock price \( Z_t = \ln S_t \) and the squared volatility process \( V_t \). We take a meshsize which is of order \( o(\Delta t) \) at every timestep. The discrete time stochastic process
that we define takes its values on this grid, and has 16 or 64 successor nodes in every point. This obviously means that we require quite some computation time per timestep. But the extra degrees of freedom that we create by our setup allow us to exploit the fact that the American Option price function is once continuously differentiable in both $S$ and $V$. This smoothness is used to improve the speed of convergence, which means that we require more computations per timestep but far less timesteps than in alternative methods. This means that we retain the flexibility of the Guan and Xiaoqiang method when introducing for example a term structure of interest rates and cash dividends, but at the same time we can get quicker convergence.

2 The Algorithm

The Heston model for squared volatility process $V$ and log stock price process $Z$ is

\[
dV_t = \kappa(\theta - V_t)dt + \omega \sqrt{V_t}dW_t^1 \\
dZ_t = (r - \frac{1}{2}V_t)dt + \sqrt{V_t}dW_t^2
\]

for given $(V_0, Z_0) = (\sigma^2, \ln S_0)$, where $\{(W_t^1, W_t^2), \ t \in [0, T]\}$ are correlated standard Brownian Motion processes with correlation coefficient $\rho$ under the riskneutral measure $Q$, and $\kappa$, $\omega$, $\theta$ and $r$ are strictly positive given constants. The value of the European option with maturity $T$ and payoff function $\Phi : \mathbb{R}^+ \times \mathbb{R} \rightarrow \mathbb{R}$ equals (see for example [14])

\[
H_t(V_t, Z_t) = e^{-r(T-t)}\mathbb{E}^Q[\Phi(e^{Z_T}, \sqrt{V_T}) \mid \mathcal{F}_t]
\]

with $(\mathcal{F}_t)_{t \in [0, T]}$ the filtration generated by the process $\{(V_t, Z_t), \ t \in [0, T]\}$. The righthand side can be shown to be indeed a function of $t$, $V_t$ and $Z_t$ only, because of the Markov property of $(V_t, Z_t)$.

We now define the following discrete time stochastic processes $(k = 0..n-1)$

\[
V_{k+1}^n = V_k^n + \kappa(\theta - V_k^n)\Delta t^n + Y_{k+1}^{n,1}\omega \sqrt{V_k^n}\Delta t^n \\
Z_{k+1}^n = Z_k^n + (r - \frac{1}{2}V_k^n)\Delta t^n + Y_{k+1}^{n,2}\sqrt{V_k^n}\Delta t^n
\]

with $(V_0^n, Z_0^n) = (V_0, Z_0)$ and where the variables $(Y_{k+1}^{n,1}, Y_{k+1}^{n,2})$ are i.i.d. distributed in $k$, with

\[
Q^n(Y_{k+1}^{n,1} = +1, \ Y_{k+1}^{n,2} = +1) = \frac{1}{4}(1 + \rho) \\
Q^n(Y_{k+1}^{n,1} = -1, \ Y_{k+1}^{n,2} = +1) = \frac{1}{4}(1 - \rho) \\
Q^n(Y_{k+1}^{n,1} = +1, \ Y_{k+1}^{n,2} = -1) = \frac{1}{4}(1 - \rho) \\
Q^n(Y_{k+1}^{n,1} = -1, \ Y_{k+1}^{n,2} = -1) = \frac{1}{4}(1 + \rho).
\]

under a new pricing measure $Q^n$. Note that this essentially means that we define a simple Euler simulation scheme for the correlated Brownian Motion processes that drive the riskneutral dynamics. This seems the most natural discrete time stochastic process to approximate the Heston model in continuous time, but it is of course non-recombining. The number of possible values for the process $(V^n, Z^n)$ grows exponentially in the number of timesteps and it is therefore not very well suited to do actual computations.

Let

\[
z_{k,\text{max}}^n = \max\{z : Q^n(Z_k^n = z) > 0\}
\]

and define $z_{k,\text{min}}^n$, $v_{k,\text{max}}^n$ and $v_{k,\text{min}}^n$ analogously. We take

\[
\Delta z_k^n = (z_{k,\text{max}}^n - z_{k,\text{min}}^n)/m_z, \quad \Delta v_k^n = (v_{k,\text{max}}^n - v_{k,\text{min}}^n)/m_v
\]
for certain \( m_v, m_z \in \mathbb{N}^+ \) which describe how fine the mesh is that we will take, and define the set of gridpoints in \( \mathbb{R}^2 \):
\[
\mathcal{S}_k^n = \{(v_k^{n,\min} + i\Delta v_k^n, z_k^{n,\min} + j\Delta z_k^n) \mid i = 0...m_v, j = 0...m_z \}.
\]

For any set \( S = S_x \times S_y \subset \mathbb{R}^2 \) of gridpoints and functions \( f : S \rightarrow \mathbb{R} \) we define the piecewise bilinear interpolating function corresponding to this function on the grid, where \( \mathcal{C}(S) \) denotes the convex hull of \( S \). This means that
\[
\mathcal{L}[f](x, y) = \begin{cases} c_{00}^S(x, y) f(x_0^S, y_0^S) + c_{01}^S(x, y) f(x_1^S, y_0^S) + c_{10}^S(x, y) f(x_0^S, y_1^S) + c_{11}^S(x, y) f(x_1^S, y_1^S) & (x, y) \in S, \\
\end{cases}
\]
where the functions \( x_0^S, y_0^S : \mathcal{C}(S) \rightarrow \mathbb{R} \) and \( y_1^S : \mathcal{C}(S) \rightarrow \mathbb{R} \) are defined in such a way that
\[
(x_0^S(x), y_0^S(y)) \in S, \quad (x_1^S(x), y_1^S(y)) \in S, \quad x_0^S(x) \leq x \leq x_1^S(x), \quad y_0^S(y) \leq y \leq y_1^S(y),
\]
i.e., \( x \) and \( y \) are always in between the adjacent grid points \( x_0^S(x), x_1^S(x) \) and \( y_0^S(y), y_1^S(y) \) respectively. The functions \( c_{ij}^S \) are defined by
\[
\begin{align*}
c_{00}^S(x, y) &= (1-x)(1-y), & \tilde{x} = \frac{x-x_0^S(x)}{x_1^S(x)-x_0^S(x)}, & \tilde{y} = \frac{y-y_0^S(y)}{y_1^S(y)-y_0^S(y)}, \\
c_{01}^S(x, y) &= \tilde{x}(1-y), \\
c_{10}^S(x, y) &= (1-\tilde{x})y, \\
c_{11}^S(x, y) &= \tilde{x}\tilde{y}.
\end{align*}
\]
We define for \( u \in \{0, 1\} \)
\[
h_u^{k,n}(x) = x - x_u^S(x), \\
g_u^{k,n}(y) = y - y_u^S(y).
\]
Obviously, \( \left| h_u^{k,n}(x) \right| < \Delta v_k^n \) and \( \left| g_u^{k,n}(x) \right| < \Delta z_k^n \), and we will need these expressions in the proof of Theorem 2 in the next section.

We take a new process \((\hat{V}_n^k, \hat{Z}_n^k)\) on the space \( \hat{S}_k^n \), as follows. First we define
\[
v^n(y, v, z) = v + \kappa(\theta - v)\Delta t^n + y\omega\sqrt{v\Delta t^n}, \\
z^n(y, v, z) = z + (r - \frac{1}{2}v)\Delta t^n + y\sqrt{v\Delta t^n}
\]
where \( \Delta t^n = T/n \). Let
\[
(\hat{V}_n^{k+1}, \hat{Z}_n^{k+1}) = \left(x_{k+1}^n(Y_{k+1}^{n,1}, V_n^{k,1}, Z_n^{k,1}), y_{k+1}^n(z^n(Y_{k+1}^{n,2}, V_n^{n,1}, Z_n^{k,1}))\right)
\]
where under a new measure \( \hat{Q}_n \), the \((Y_{k+1}^{n,1}, Y_{k+1}^{n,2})\) have the same distribution as under \( Q^n \)
\[
\hat{Q}_n(Y_{k+1}^{n,1} = i, Y_{k+1}^{n,2} = j) = \frac{1}{2}(1 + ij\rho), \quad i, j \in \{-1, 1\} \tag{2.1}
\]
and are independent for different values of \( k \), while the \((Y_{k+1}^{n,3}, Y_{k+1}^{n,4})\) have the following conditional distribution:
\[
\hat{Q}_n(Y_{k+1}^{n,3} = i, Y_{k+1}^{n,4} = j \mid Y_{k+1}^{n,1}, Y_{k+1}^{n,2}, V_n^{n,1}, Z_n^{k,1}) = c_{ij}^n(v^n(Y_{k+1}^{n,1}, V_n^{n,1}, Z_n^{k,1}), z^n(Y_{k+1}^{n,2}, V_n^{n,1}, Z_n^{k,1})) \quad i, j \in \{0, 1\}
\]
and are conditionally independent for different values of \( k \). Note that this means that in every time step there are four times four is 16 new possible values for \((\hat{V}_n^{k+1}, \hat{Z}_n^{k+1})\) based on the current value of \((\hat{V}_n^k, \hat{Z}_n^k)\), since every one of the four next time step values is split over four new points.

Having thus defined a discrete-time homogeneous Markov process, we will now investigate the weak convergence of this stochastic process in the next section.
3 Weak Convergence

From now on, we take $T = 1$ without loss of generality. We will use the following theorem by Stroock and Varadhan [20] (see their section 11.2).

**Theorem 1.** Let $Q^n$ be a probability measure and $\Pi_n$ a transition function on $\mathbb{R}^d$ such that $Q^n$-a.s. $X^n_0 = x_0$ and for all $k > 0$

$$X^n_t = (k + 1 - tn)X^n_{k/n} + (tn - k)X^n_{(k+1)/n}, \quad \frac{k}{n} \leq t \leq \frac{k+1}{n}.$$ 

Define for all $k > 0$ and all Borel sets $A$ in $\mathbb{R}^d$

$$\Pi_n(X_{k/n}, A) = Q^n(X_{(k+1)/n} \in A | F_{k/n}),$$

where $F_{k/n} = \sigma(\{X_t, 0 \leq t \leq k/n\})$ and assume that for all $R > 0$, $\epsilon > 0$ and certain continuous functions $a : \mathbb{R}^d \to S_d$ and $b : \mathbb{R}^d \to \mathbb{R}^d$:

$$\lim_{n \to \infty} \sup_{||x|| \leq R} ||a_n(x) - a(x)|| = 0, \quad a_n^j(x) = n \int_{||y-x|| \leq 1} (y_i - x_i)(y_j - x_j) \Pi_n(x, dy)$$

$$\lim_{n \to \infty} \sup_{||x|| \leq R} ||b_n(x) - b(x)|| = 0, \quad b_n^j(x) = n \int_{||y-x|| \leq 1} (y_i - x_i) \Pi_n(x, dy)$$

$$\lim_{n \to \infty} \sup_{||x|| \leq R} \Delta_n^x(x) = 0, \quad \Delta_n^x(x) = n \Pi^n(x, \{y \in \mathbb{R}^d : ||y - x|| > \epsilon\}).$$

If the martingale problem for $a$ and $b$ has exactly one solution $Q_x$ for every initial value $x_0 = x$ then $Q^n$ converges weakly to $Q_{x_0}$, uniformly on compact subsets of $\mathbb{R}^d$.

We take $\hat{V}^n_{k/n} = \hat{V}^n_k$ and $\hat{Z}^n_{k/n} = \hat{Z}^n_k$ and define

$$\hat{V}^n_t = (k + 1 - tn)\hat{V}^n_{k/n} + (tn - k)\hat{V}^n_{(k+1)/n}, \quad \frac{k}{n} \leq t \leq \frac{k+1}{n}$$

$$\hat{Z}^n_t = (k + 1 - tn)\hat{Z}^n_{k/n} + (tn - k)\hat{Z}^n_{(k+1)/n}.$$ 

Our main result is the following.

**Theorem 2.** Assume that

$$\lim_{n \to \infty} \Delta t^n = 0, \quad \lim_{n \to \infty} \max_{k=1..n-1} \frac{\Delta z^n_k}{\Delta t^n} = 0, \quad \lim_{n \to \infty} \max_{k=1..n-1} \frac{\Delta z^n_k}{\Delta t^n} = 0.$$

Then the processes $(\hat{V}^n, \hat{Z}^n)$ converge weakly to the process $(V, Z)$, i.e.

$$((\hat{V}^n, \hat{Z}^n) | Q^n) \overset{W}{\to} ((V, Z) | Q)$$

where $W$ stands for weak convergence in the space of continuous processes on $[0, 1]$.

**Proof.**

Define $X^n_{k/n} = (\hat{V}^n_k, \hat{Z}^n_k)$ and

$$\Delta X^n_k = X^n_{(k+1)/n} - X^n_{k/n}.$$ 

Take $x = (v, z)$. Since the drift and diffusion functions of the Heston-model

$$b(x) = \begin{bmatrix} \kappa(\theta - v) \\ r - \frac{1}{2} v^2 \end{bmatrix}, \quad a(x) = v \begin{bmatrix} \omega^2 & \rho \omega \\ \rho \omega & 1 \end{bmatrix},$$

Where $\kappa, \theta, r, \omega, \rho$ are parameters.
define a unique solution for the associated martingale problem (see for example [13]), we only need to check the conditions stated in the previous theorem. This boils down to proving that for all \( R > 0 \) and all \( \varepsilon > 0 \) we have

\[
0 = \lim_{n \to \infty} \sup_{\|x\| \leq R} \left\| b(x) - n \mathbb{E}^{\tilde{Q}^n} \left[ (\Delta X^n_k)^1_{\{\|\Delta X^n_k\| \leq 1\}} \right| X^n_{k/n} = x \right\|
\]

\[
0 = \lim_{n \to \infty} \sup_{\|x\| \leq R} \left\| a(x) - n \mathbb{E}^{\tilde{Q}^n} \left[ (\Delta X^n_k)^T (\Delta X^n_k)^T 1_{\{\|\Delta X^n_k\| \leq 1\}} \right| X^n_{k/n} = x \right\|
\]

\[
0 = \lim_{n \to \infty} \sup_{\|x\| \leq R} n \tilde{Q}^n (\|\Delta X^n_k\| > \varepsilon \mid X^n_{k/n} = x)
\]

Let \( X^n_{k/n} = x = (v, z) \). Then

\[
\Delta X^n_k = \left[ \begin{array}{c}
\dot{V}^n_{k+1} - \dot{V}^n_k \\
\dot{Z}^n_{k+1} - \dot{Z}^n_k
\end{array} \right] = \left[ \begin{array}{c}
\frac{\partial_{x,Y}^{n+1}(v^n(Y_{k+1}^n, v, z)) - v}{r - \frac{1}{2}v} \Delta t^n + Y_{k+1}^{n-1} \omega \sqrt{\nu \Delta t^n} \\
\frac{\partial_{z,Y}^{n+1}(z^n(Y_{k+1}^n, v, z)) - z}{r - \frac{1}{2}v} \Delta t^n + Y_{k+1}^{n-1} \omega \sqrt{\nu \Delta t^n}
\end{array} \right]
\]

so

\[
\Delta X^n_k = \left[ \begin{array}{c}
k(\theta - v) \Delta t^n + Y_{k+1}^{n-1} \omega \sqrt{\nu \Delta t^n} \\
(k - \frac{1}{2}v) \Delta t^n + Y_{k+1}^{n-1} \omega \sqrt{\nu \Delta t^n}
\end{array} \right]
\]

with

\[
\tilde{Q}^n(Y_{k+1}^{n-1} = i, Y_{k+1}^{n-2} = j) = \frac{1}{2}(1 + ij) \quad i, j \in \{-1, 1\}
\]

Since \( \Delta t^n = 1/n \) we see from (3.2) that for \( n > n_1 \) with \( n_1 \) large enough we have \( \|\Delta X^n_k\| \leq 1 \) since \( |Y_{k+1}^{n-1}| = 1 \) and at the same time

\[
|h_k^{k+1,n}(v)| < \Delta v_{k+1}^n, \quad |g_k^{k+1,n}(z)| < \Delta x_{k+1}^n.
\]

which implies that \( h_k^{k+1,n}(v) = o(\Delta t^n) \) and \( g_k^{k+1,n}(z) = o(\Delta t^n) \) for \( u \in \{0, 1\} \). But this means that for \( n > n_1 \) we find

\[
b(x) - n \mathbb{E}^{\tilde{Q}^n} \left[ (\Delta X^n_k)^1_{\{\|\Delta X^n_k\| \leq 1\}} \right| X^n_{k/n} = x \] = \mathbb{E}^{\tilde{Q}^n} \left[ \begin{array}{c}
h_{k+1,n}^{k+1,n}(v) \\
g_{k+1,n}^{k+1,n}(z)
\end{array} \right]
\]

and the norm of this clearly converges to zero. A similar argument shows that the second condition is satisfied as well, because

\[
a(x) - n \mathbb{E}^{\tilde{Q}^n} \left[ (\Delta X^n_k)^T (\Delta X^n_k)^T 1_{\{\|\Delta X^n_k\| \leq 1\}} \right| X^n_{k/n} = x \] = \mathbb{E}^{\tilde{Q}^n} \left[ \begin{array}{c}
a_1^{k+1,n} \\
a_2^{k+1,n}
\end{array} \right]
\]

for \( n > n_1 \), with

\[
\begin{align*}
a_1^{k+1,n} &= v\omega^2 - \mathbb{E}^{\tilde{Q}^n} \frac{1}{\Delta t^n} \left( \kappa(\theta - v) \Delta t^n + Y_{k+1}^{n-1} \omega \sqrt{\nu \Delta t^n} + o(\Delta t^n) \right)^2 \\
a_2^{k+1,n} &= v\rho - \mathbb{E}^{\tilde{Q}^n} \frac{1}{\Delta t^n} \left( (r - \frac{1}{2}v) \Delta t^n + Y_{k+1}^{n-2} \omega \sqrt{\nu \Delta t^n} + o(\Delta t^n) \right) \left( \kappa(\theta - v) \Delta t^n + Y_{k+1}^{n-1} \omega \sqrt{\nu \Delta t^n} + o(\Delta t^n) \right)
\end{align*}
\]

which shows convergence in norm. The third condition is trivial since when \( \|x\| < R \)

\[
\tilde{Q}^n(\|\Delta X^n_k\| > \varepsilon \mid X^n_{k/n} = x)
\]

\[
= \tilde{Q}^n \left( \left\| \left[ \begin{array}{c}
k(\theta - v) \Delta t^n + Y_{k+1}^{n-1} \omega \sqrt{\nu \Delta t^n} \\
(r - \frac{1}{2}v) \Delta t^n + Y_{k+1}^{n-2} \omega \sqrt{\nu \Delta t^n}
\end{array} \right] \right\| > \varepsilon \right)
\]
and this is actually equal to zero if we take $n$ large enough, since the vector of which the norm is taken can be made arbitrarily small since $|Y^{n,n}_k| = 1$ for $u \in \{1, 2\}$ while $h_{k+1,n} = o(\Delta^n)$ and $g^{k+1,n}_u = o(\Delta^n)$ for $u \in \{0, 1\}$.

The proof given above relies on the fact that we use a bilinear interpolation for option price values which fall outside grid points, as can clearly be seen from the functions $c_{ij}^S$ that we defined in Section 2. Obviously, we could use other approximations as well. For example, interpolation based on bicubic splines instead of bilinear ones would use

$$\mathcal{L}^S[f](x, y) = \sum_{i=-1}^{2} \sum_{j=-1}^{2} c_{ij}^S(x, y) f(x_i^S(x), y_j^S(y))$$

where the functions $x_i^S$ and $y_j^S$ are defined as before, $x_{-1}^S(x) = x_0^S(x) - \Delta x$, $x_1^S(x) = x_1^S(x) + \Delta x$, and similar definitions hold for $y_{-1}^S$ and $y_2^S$, while

$$c_{00}^S(x, y) = x_0^S(x) - 1 - 1(3\bar{x}^2 - 2\bar{x} - 2)(3\bar{y}^2 - 2\bar{y} - 2)
\quad c_{-1,1}^S(x, y) = 1\bar{x}(x - 1)^2(\bar{y} - 1)^2
\quad c_{0,-1}^S(x, y) = -1\bar{x}(x - 1)^2(3\bar{y}^2 - 2\bar{y} - 2)$$

while the rest follows from symmetries: $c_{i,j} \to c_{-i,j}$ means $(\bar{x}, \bar{y}) \to (\bar{x}, \bar{y})$, $c_{i,j} \to c_{i,-j}$ means $(\bar{x}, \bar{y}) \to (1 - \bar{x}, \bar{y})$ and finally $c_{i,j} \to c_{1-i,j}$ means $(\bar{x}, \bar{y}) \to (\bar{x}, 1 - \bar{y})$.

For this bicubic interpolation we cannot give a similar convergence proof as for the bilinear case. In the bilinear case we use the fact that the functions $c_{ij}$ all map into $[0, 1]$ while the sum over all $i$ and $j$ of $c_{ij}(x, y)$ equals 1 for all $x$ and $y$. This allows us to use the bilinear interpolation to redistribute probability mass over neighbouring points, to create a stochastic process and use weak convergence results for such processes. However, when we use bicubic interpolation some of the weights $c_{ij}(x, y)$ can be negative so we cannot use the probabilistic result we used earlier.

Numerical error analysis helps to illustrate what may go wrong. If some weights $c_{ij}(x, y)$ are negative, numerical errors that we introduce in our scheme may increase over time. Indeed, we would like to have that $\mathbb{E}\sup_L \mathcal{L}[\Delta f] \leq \sup_L \Delta f$ for an error function $\Delta f$ on the grid. This is satisfied for bilinear interpolation, but not for bicubic interpolation. It would therefore be better to use a local scheme which gives positive values for all positive functions and is $C^1$ at the same time. Such grids exist [19] but are very computationally intensive. Therefore we prefer to use the bicubic interpolation. Notice that if the errors $\Delta f$ do not change signs in neighbouring points all the time, the errors can be expected to decrease on average and we will see in numerical results that this indeed what is observed in practice.

### 4 Numerical Results

In Tables 1 to 8 we report the results of some experiments that we performed with our new numerical method. The test case we took is the same as in earlier papers that deal with American options in the Heston model [2, 4, 11, 18, 21]. The relevant parameter values are

$$\kappa = 5, \quad \theta = 0.16, \quad \omega = 0.9, \quad \rho = 0.1, \quad r = 0.10.$$ 

The strike was chosen to be $K = 10.0$ and different values of the starting volatility $V_0$ and initial stock price $S_0$ were considered.

Tables 1 and 2 give the option prices, and relative errors for the method involving 16 successors for every state, while Tables 3 and 4 provide the same for the model with 64 successors per state. To get error estimates we used exact Heston option prices for the European case, and the results by Zvë, Forsyth and Vetzal in the American case; the results reported by other authors are close to these. Results are provided for two initial volatility values, $\sqrt{V_0} = 25\%$ and $\sqrt{V_0} = 50\%$, and for initial stock prices $S_0 \in \{8, 9, 10, 11, 12\}$, i.e. the values that were used earlier in the papers.
mentioned above. Per option that we consider, we present the results for our algorithm with four different number of timesteps and grid sizes to show the speed of convergence. The results by other authors are indicated by ZFV (Zven, Forsyth, Vetzal), IT-PSOR (Ikonen and Toivanen), OO (Oosterlee) and CP (Clarke and Parott). Errors are based on the Heston closed-form formula for European options, and on the results by Zven, Forsyth, and Vetzal for American options; these values are close to the other values reported.

We see from these tables that the bicubic method indeed performs considerably better than the bilinear method. Indeed, for the bicubic method we find option prices with errors smaller than one cent (or maximally 1%) even for the smallest number of timesteps. The errors converge to zero in all cases but it is clear that the bicubic method converges a lot faster, while the computational time is of the same order of magnitude as for the bilinear calculations. The CPU times for the smallest grid are approximately 0.19 seconds for the method with 16 successors and 0.35 seconds for the method with 64 successors, while on the largest grids they are 17 and 27 seconds respectively.

5 Conclusions

In this paper we have shown that tree-based methods can be used to find the price of American Options in the Heston model. We believe that the speed and flexibility of this method may make it a suitable alternative for the PDE-based methods that have been mentioned in this paper. Moreover, the basic idea behind our method, to approximate a diffusion process on a grid which has a finer mesh for $\Delta S$ and $\Delta V$ for the stock and volatility than the usual square root of the time spacing $\Delta t$, may be applied to other methods as well. As indicated in the paper, the precise properties of the interpolation schemes, such as whether strictly positive functions are mapped to strictly positive interpolating functions, will be essential for the further analysis of such schemes. Interpolation schemes exist which preserve positivity, monotonicity, differentiability and local convexity in the space variables and once these become computationally feasible, possibly even better methods to approximate American options in the Heston model may become available.

References


\*Algorithms were implemented in Matlab on a Pentium 4, 2GHz machine


<table>
<thead>
<tr>
<th>steps</th>
<th>steps</th>
<th>steps</th>
<th>$S_0$</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>125</td>
<td>6</td>
<td>25</td>
<td>1.8427</td>
<td>1.0572</td>
<td>0.5153</td>
<td>0.2267</td>
<td>0.0975</td>
<td></td>
</tr>
<tr>
<td>250</td>
<td>12</td>
<td>35</td>
<td>1.8417</td>
<td>1.0543</td>
<td>0.5103</td>
<td>0.2206</td>
<td>0.0920</td>
<td></td>
</tr>
<tr>
<td>500</td>
<td>24</td>
<td>50</td>
<td>1.8408</td>
<td>1.0527</td>
<td>0.5077</td>
<td>0.2169</td>
<td>0.0884</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>48</td>
<td>71</td>
<td>1.8402</td>
<td>1.0515</td>
<td>0.5060</td>
<td>0.2142</td>
<td>0.0857</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Exact)</td>
<td>1.8389</td>
<td>1.0483</td>
<td>0.5015</td>
<td>0.2082</td>
<td>0.0804</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Error)</td>
<td>0.2%</td>
<td>0.8%</td>
<td>2.8%</td>
<td>8.9%</td>
<td>21.3%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.2%</td>
<td>0.6%</td>
<td>1.8%</td>
<td>5.9%</td>
<td>14.3%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.1%</td>
<td>0.4%</td>
<td>1.2%</td>
<td>4.2%</td>
<td>9.9%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.1%</td>
<td>0.3%</td>
<td>0.9%</td>
<td>2.9%</td>
<td>6.6%</td>
</tr>
</tbody>
</table>

Table 1: European Option, 16 Successors, $\sqrt{V_0} = 0.25$

<table>
<thead>
<tr>
<th>steps</th>
<th>steps</th>
<th>steps</th>
<th>$S_0$</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>125</td>
<td>6</td>
<td>25</td>
<td>1.9933</td>
<td>1.1189</td>
<td>0.5366</td>
<td>0.2333</td>
<td>0.0995</td>
<td></td>
</tr>
<tr>
<td>250</td>
<td>12</td>
<td>35</td>
<td>1.9948</td>
<td>1.1149</td>
<td>0.5307</td>
<td>0.2268</td>
<td>0.0938</td>
<td></td>
</tr>
<tr>
<td>500</td>
<td>24</td>
<td>50</td>
<td>1.9960</td>
<td>1.1128</td>
<td>0.5276</td>
<td>0.2230</td>
<td>0.0902</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>48</td>
<td>71</td>
<td>1.9970</td>
<td>1.1112</td>
<td>0.5254</td>
<td>0.2201</td>
<td>0.0875</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2.0000</td>
<td>1.1076</td>
<td>0.5202</td>
<td>0.2138</td>
<td>0.0821</td>
<td>ZFV</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2.0000</td>
<td>1.1074</td>
<td>0.5190</td>
<td>0.2130</td>
<td>0.0818</td>
<td>IT-PSOR</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2.0000</td>
<td>1.1070</td>
<td>0.5170</td>
<td>0.2120</td>
<td>0.0815</td>
<td>Oo</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2.0000</td>
<td>1.1080</td>
<td>0.5316</td>
<td>0.2261</td>
<td>0.0907</td>
<td>CP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Error)</td>
<td>-0.3%</td>
<td>1.0%</td>
<td>3.2%</td>
<td>9.1%</td>
<td>21.2%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-0.3%</td>
<td>0.7%</td>
<td>2.0%</td>
<td>6.1%</td>
<td>14.3%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-0.2%</td>
<td>0.5%</td>
<td>1.4%</td>
<td>4.3%</td>
<td>9.9%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-0.1%</td>
<td>0.3%</td>
<td>1.0%</td>
<td>3.0%</td>
<td>6.6%</td>
</tr>
</tbody>
</table>

Table 2: American Option, 16 Successors, $\sqrt{V_0} = 0.25$

<table>
<thead>
<tr>
<th>steps</th>
<th>steps</th>
<th>steps</th>
<th>$S_0$</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>125</td>
<td>6</td>
<td>25</td>
<td>1.8378</td>
<td>1.0474</td>
<td>0.5001</td>
<td>0.2080</td>
<td>0.0802</td>
<td></td>
</tr>
<tr>
<td>250</td>
<td>12</td>
<td>35</td>
<td>1.8378</td>
<td>1.0479</td>
<td>0.5012</td>
<td>0.2079</td>
<td>0.0797</td>
<td></td>
</tr>
<tr>
<td>500</td>
<td>24</td>
<td>50</td>
<td>1.8380</td>
<td>1.0481</td>
<td>0.5015</td>
<td>0.2079</td>
<td>0.0798</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>48</td>
<td>71</td>
<td>1.8382</td>
<td>1.0481</td>
<td>0.5016</td>
<td>0.2080</td>
<td>0.0799</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Exact)</td>
<td>1.8389</td>
<td>1.0483</td>
<td>0.5015</td>
<td>0.2082</td>
<td>0.0804</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Error)</td>
<td>-0.1%</td>
<td>-0.1%</td>
<td>-0.3%</td>
<td>-0.1%</td>
<td>-0.3%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-0.1%</td>
<td>0.0%</td>
<td>0.0%</td>
<td>-0.1%</td>
<td>-0.9%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.0%</td>
<td>0.0%</td>
<td>0.0%</td>
<td>-0.1%</td>
<td>-0.8%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.0%</td>
<td>0.0%</td>
<td>0.0%</td>
<td>-0.1%</td>
<td>-0.6%</td>
</tr>
</tbody>
</table>

Table 3: European Option, 64 Successors, $\sqrt{V_0} = 0.25$

<table>
<thead>
<tr>
<th>steps</th>
<th>steps</th>
<th>steps</th>
<th>$S_0$</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>125</td>
<td>6</td>
<td>25</td>
<td>1.9922</td>
<td>1.1107</td>
<td>0.5192</td>
<td>0.2133</td>
<td>0.0815</td>
<td></td>
</tr>
<tr>
<td>250</td>
<td>12</td>
<td>35</td>
<td>1.9942</td>
<td>1.1075</td>
<td>0.5199</td>
<td>0.2133</td>
<td>0.0812</td>
<td></td>
</tr>
<tr>
<td>500</td>
<td>24</td>
<td>50</td>
<td>1.9957</td>
<td>1.1076</td>
<td>0.5201</td>
<td>0.2133</td>
<td>0.0813</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>48</td>
<td>71</td>
<td>1.9968</td>
<td>1.1076</td>
<td>0.5202</td>
<td>0.2134</td>
<td>0.0815</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2.0000</td>
<td>1.1076</td>
<td>0.5202</td>
<td>0.2138</td>
<td>0.0821</td>
<td>ZFV</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2.0000</td>
<td>1.1074</td>
<td>0.5190</td>
<td>0.2130</td>
<td>0.0818</td>
<td>IT-PSOR</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2.0000</td>
<td>1.1070</td>
<td>0.5170</td>
<td>0.2120</td>
<td>0.0815</td>
<td>Oo</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2.0000</td>
<td>1.1080</td>
<td>0.5316</td>
<td>0.2261</td>
<td>0.0907</td>
<td>CP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Error)</td>
<td>-0.4%</td>
<td>0.0%</td>
<td>-0.2%</td>
<td>-0.2%</td>
<td>-0.7%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-0.3%</td>
<td>0.0%</td>
<td>-0.1%</td>
<td>-0.2%</td>
<td>-1.1%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-0.2%</td>
<td>0.0%</td>
<td>0.0%</td>
<td>-0.2%</td>
<td>-1.0%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-0.2%</td>
<td>0.0%</td>
<td>0.0%</td>
<td>-0.2%</td>
<td>-0.7%</td>
</tr>
</tbody>
</table>

Table 4: American Option, 64 Successors, $\sqrt{V_0} = 0.25$
<table>
<thead>
<tr>
<th>$S_0$</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>steps s</td>
<td>6</td>
<td>25</td>
<td>1.9837</td>
<td>1.2933</td>
<td>0.7887</td>
</tr>
<tr>
<td>steps v</td>
<td>12</td>
<td>35</td>
<td>1.9816</td>
<td>1.2886</td>
<td>0.7821</td>
</tr>
<tr>
<td>steps t</td>
<td>250</td>
<td>12</td>
<td>1.9803</td>
<td>1.2864</td>
<td>0.7786</td>
</tr>
<tr>
<td>500</td>
<td>24</td>
<td>50</td>
<td>1.9794</td>
<td>1.2846</td>
<td>0.7760</td>
</tr>
<tr>
<td>1000</td>
<td>48</td>
<td>71</td>
<td>1.9773</td>
<td>1.2800</td>
<td>0.7697</td>
</tr>
</tbody>
</table>

(Exact) 1.9773 1.2800 0.7697 0.4361 0.2373

(Error) 0.3% 1.0% 2.5% 5.1% 9.5%
0.2% 0.7% 1.6% 3.3% 6.2%
0.2% 0.5% 1.2% 2.3% 4.2%
0.1% 0.4% 0.8% 1.6% 2.8%

Table 5: European Option, 16 Successors, $\sqrt{V_0} = 0.50$

<table>
<thead>
<tr>
<th>$S_0$</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>steps s</td>
<td>6</td>
<td>25</td>
<td>2.0873</td>
<td>1.3514</td>
<td>0.8190</td>
</tr>
<tr>
<td>steps v</td>
<td>12</td>
<td>35</td>
<td>2.0839</td>
<td>1.3450</td>
<td>0.8109</td>
</tr>
<tr>
<td>steps t</td>
<td>250</td>
<td>12</td>
<td>2.0820</td>
<td>1.3416</td>
<td>0.8064</td>
</tr>
<tr>
<td>500</td>
<td>24</td>
<td>50</td>
<td>2.0809</td>
<td>1.3392</td>
<td>0.8033</td>
</tr>
<tr>
<td>1000</td>
<td>48</td>
<td>71</td>
<td>2.0784</td>
<td>1.3337</td>
<td>0.7961</td>
</tr>
</tbody>
</table>

(ZFV) 2.0784 1.3337 0.7961 0.4483 0.2428

(IT-PSOR) 2.0783 1.3335 0.7958 0.4481 0.2427

(OO) 2.0790 1.3340 0.7960 0.4490 0.2430

(CP) 2.0733 1.3290 0.7992 0.4536 0.2502

(Error) 0.4% 1.3% 2.9% 5.5% 9.9%
0.3% 0.8% 1.9% 3.6% 6.4%
0.2% 0.6% 1.3% 2.5% 4.3%
0.1% 0.4% 0.9% 1.7% 2.9%

Table 6: American Option, 16 Successors, $\sqrt{V_0} = 0.50$

<table>
<thead>
<tr>
<th>$S_0$</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>steps s</td>
<td>6</td>
<td>25</td>
<td>1.9781</td>
<td>1.2832</td>
<td>0.7740</td>
</tr>
<tr>
<td>steps v</td>
<td>12</td>
<td>35</td>
<td>1.9777</td>
<td>1.2823</td>
<td>0.7730</td>
</tr>
<tr>
<td>steps t</td>
<td>250</td>
<td>12</td>
<td>1.9776</td>
<td>1.2816</td>
<td>0.7720</td>
</tr>
<tr>
<td>500</td>
<td>24</td>
<td>50</td>
<td>1.9775</td>
<td>1.2811</td>
<td>0.7713</td>
</tr>
<tr>
<td>1000</td>
<td>48</td>
<td>71</td>
<td>1.9773</td>
<td>1.2800</td>
<td>0.7697</td>
</tr>
</tbody>
</table>

(Exact) 1.9773 1.2800 0.7697 0.4361 0.2373

(Error) 0.0% 0.2% 0.6% 0.8% 1.0%
0.0% 0.2% 0.4% 0.6% 0.6%
0.0% 0.1% 0.3% 0.4% 0.4%
0.0% 0.1% 0.2% 0.3% 0.3%

Table 7: European Option, 64 Successors, $\sqrt{V_0} = 0.50$

<table>
<thead>
<tr>
<th>$S_0$</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>steps s</td>
<td>6</td>
<td>25</td>
<td>2.0797</td>
<td>1.3377</td>
<td>0.8011</td>
</tr>
<tr>
<td>steps v</td>
<td>12</td>
<td>35</td>
<td>2.0794</td>
<td>1.3365</td>
<td>0.7996</td>
</tr>
<tr>
<td>steps t</td>
<td>250</td>
<td>12</td>
<td>2.0790</td>
<td>1.3357</td>
<td>0.7986</td>
</tr>
<tr>
<td>500</td>
<td>24</td>
<td>50</td>
<td>2.0789</td>
<td>1.3351</td>
<td>0.7978</td>
</tr>
<tr>
<td>1000</td>
<td>48</td>
<td>71</td>
<td>2.0784</td>
<td>1.3337</td>
<td>0.7961</td>
</tr>
</tbody>
</table>

(ZFV) 2.0784 1.3337 0.7961 0.4483 0.2428

(IT-PSOR) 2.0783 1.3335 0.7958 0.4481 0.2427

(OO) 2.0790 1.3340 0.7960 0.4490 0.2430

(CP) 2.0733 1.3290 0.7992 0.4536 0.2502

(Error) 0.1% 0.3% 0.6% 0.9% 1.0%
0.0% 0.2% 0.4% 0.6% 0.5%
0.0% 0.1% 0.3% 0.4% 0.4%
0.0% 0.1% 0.2% 0.3% 0.3%

Table 8: American Option, 64 Successors, $\sqrt{V_0} = 0.50$