U.U.D.M. Project Report 2014:19

Calculating the density of a Heston stochastic volatility model

Guanyu Li

Examensarbete i matematik, 15 hp
Handledare och examinator: Erik Ekström
Maj 2014

Department of Mathematics
Uppsala University

Calculating the density of a Heston stochastic volatility model

Guanyu Li[*]
Uppsala University
Degree Project in Financial Mathematics

May 27, 2014

[*]f.guanyu.li@gmail.com

Abstract

In the field of mathematical finance, there is often a need for density calculations in the pricing of securities. The Kolmogorov forward equation gives such a density when the price can be modeled by a stochastic differential equation. We use the Heston model, which gives a volatility dimension to our solution density. In order to avoid tricky boundary conditions involved in numerically solving a forward equation, we use a symmetry relation that transforms the forward equation into a Kolmogorov backward equation. We numerically solve the backward equation and transform it back into the forward solution.

1 Introduction

When modeling financial instruments, the most commonly used stochastic differential equation is the Black-Scholes model. One drawback of this model is that it treats volatility as constant throughout the life of the instrument, when in fact we know that in the real world volatility will change and can itself be modeled by a stochastic differential equation.

The model that we use is the Heston stochastic volatility model. For the general case, we denote a time-homogeneous non-negative diffusion X_t satisfying

$$dX_t = \beta(X_t)dt + \sigma(X_t)dB_t \tag{1}$$

We assume that the drift term is always greater than or equal to zero when $X_t = 0$, and that the diffusion term is equal to zero when $X_t = 0$.

We start with initial conditions at $t = 0$ and are interested in the future behavior. Therefore, the domain is $(x, y, t) \in (0, \infty)^2 \times [0, \infty)$. The solution is expected to stabilize within a finite amount of time.

We are interested in the density distribution of the process,

$$p(x, y, t) = \frac{P_x(X_t \in (y, y + dy))}{dy} \tag{2}$$

which should satisfy the Kolmogorov forward equation

$$\begin{cases} p_t(x, y, t) = (\alpha(y)p(x, y, t))_{yy} - (\beta(y)p(x, y, t))_y \\ p(x, y, 0) = \delta_x(y) \end{cases} \tag{3}$$

as long as it exists and is sufficiently regular. The term $\alpha(y) = \sigma^2(y)/2$, and δ_x is a Dirac delta function at x.

Since we know the exact value of an initial stock price and volatility, we start out at time zero with all the probability density centered at a single point, with the integral over the whole solution area equal to one. Therefore, for the validity of our solution, we should expect the probability density function at any later time to sum up to one.

The outline of this paper is as follows. Section 2 will describe the transformation that makes the numerical implementation of this problem possible. Section 3 will explain how the stochastic volatility part of the Heston model was calculated and implemented, while section 4 explores the problem fully by including the price process as well, thereby making a complete calculation of density in the Heston model. Section 5 will present the results.

2 Transformation between the Kolmogorov forward and backward equations

Although the Kolmogorov forward equation will give us the density of the diffusion process, it is difficult to implement numerically because the boundary conditions for this density are difficult to describe, and the solution may explode at the boundaries. However, the Kolmogorov backward equation is much easier to solve numerically, and hence a conversion between the two solution densities is needed.

The equation that was derived in [1] is

$$m(x)p(x,y,t) = m(y)p(y,x,t) \tag{4}$$

The function m, the density of the speed measure, is given by

$$m(x) = \frac{1}{\alpha(x)} \exp\left(\int_1^x \frac{\beta(z)}{\alpha(z)} dz \right) \tag{5}$$

We use these relations to transform our forward equation into a backward equation, which we will solve numerically. Then we use the same relations to do the opposite transformation on our solution, so that it becomes the solution of the forward equation.

We refer readers to Ekström and Tysk for the proof.

3 Modeling the stochastic volatility

In order to more easily confirm that the Kolmogorov equations and this transformation can be used to find the density of a stochastic volatility model, we start working with only the stochastic volatility equation, and omit the price process for now. Therefore, the solution density is a model in two dimensions, probability versus volatility value, and it changes with time. This is the stochastic volatility component of the Heston model, which is a Cox-Ingersoll-Ross process:

$$dv_t = \kappa(\theta - v_t)dt + \xi\sqrt{v_t}dW_t^v \tag{6}$$

The CIR process is non-negative, because as we can see from the equation, as v_t approaches zero, the drift term will become more and more positive, pushing the process back towards the positive v direction. Additionally, the CIR process is mean-reverting, because as v_t approaches the value of θ, our long run expected variance, the drift term will approach zero, meaning that the process tends to stay around the value of θ.

The parameter κ is the speed at which the process will revert to its mean. The parameter θ is the expected long run volatility, and the parameter ξ is the volatility of volatility term. The time parameter t is in units of one year, and all the parameters used in our simulations will be reasonable in accordance with those observed in the financial markets.

3.1 Calculations

The Kolmogorov forward equation operator as given in [2] is

$$(A * f)(t,x) = -\frac{\partial}{\partial x}\left[\mu(t,x)f(t,x)\right] + \frac{1}{2}\frac{\partial^2}{\partial x^2}\left[\sigma^2(t,x)f(t,x)\right] \tag{7}$$

Applying this to equation 6, and denoting the forward density as f, we get the differential equation giving f as

$$\frac{\partial}{\partial t}f(s,y;t,x) = \frac{1}{2}\frac{\partial^2}{\partial v^2}\left[\xi^2 v f\right] - \frac{\partial}{\partial v}\left[\kappa(\theta - v)f\right] \tag{8}$$

Now we plug in our transformation as given in equation 5, to the following differential equation, where p denotes the density in the backward solution

$$p_t = \alpha \exp\left(-\int_1^x \frac{\beta(z)}{\alpha(z)}dz\right)\left[\left(\exp\left(\int_1^x \frac{\beta(z)}{\alpha(z)}dz\right)f\right)_{vv} - \left(\frac{\beta}{\alpha}\exp\left(\int_1^x \frac{\beta(z)}{\alpha(z)}dz\right)f\right)_v\right] \tag{9}$$

By differentiating and combining terms, we will get

$$p_t = \frac{\xi^2 v}{2}p_{vv} + \kappa(\theta - v)p_v \tag{10}$$

Now we use numerical approximations of first and second order derivatives

$$\frac{p_k^{n+1} - p_k^n}{dt} = \alpha\left(\frac{p_{k+1}^n - 2p_k^n + p_{k-1}^n}{(dv)^2}\right) + \beta\left(\frac{p_{k+1}^n - p_{k-1}^n}{2dv}\right) \tag{11}$$

And rearrange

$$p_k^{n+1} = \left(\alpha\frac{dt}{dv^2} - \beta\frac{dt}{2dv}\right)p_{k-1}^n + \left(1 - 2\alpha\frac{dt}{dv^2}\right)p_k^n + \left(\alpha\frac{dt}{dv^2} + \beta\frac{dt}{2dv}\right)p_{k+1}^n \tag{12}$$

As for our boundary condition at $v = 0$, we plug in for v to get

$$p_t(t,0) = b(0)p_v(t,0) \tag{13}$$

This becomes

$$\frac{1}{dt}\left(p_0^{n+1} - p_0^n\right) = b(0)\left(p_1^n - p_0^n\right)\frac{1}{dv} \tag{14}$$

Rearranging we get the boundary condition as

$$p_0^{n+1} = \kappa\theta\left(p_1^n - p_0^n\right)\frac{dt}{dv} + p_0^n \tag{15}$$

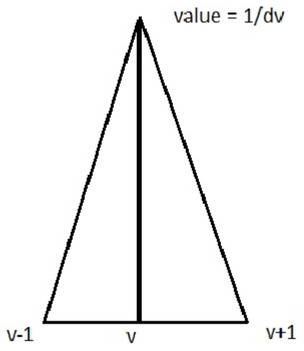
3.2 Implementation

The implementation can be done using a fairly standard finite differences method. We make a matrix out of the coefficients in equation 12.

The algorithm is structured as follows:

1. Set initial conditions
   (a) Heston parameters
   (b) Number of space-steps and time-steps
2. Create the finite difference matrix
3. Create the delta function approximation
4. Iterations of matrix multiplication and resetting bounds for each time-step
5. Transform backward solution to forward solution
6. Sum the total density

To approximate the delta function, we think of a series of 3 gridpoints in the solution vector as forming the base of a triangle. Therefore, if we set the value at the middle gridpoint as equal to $1/dv$, then when we integrate over this segment, our integral will be equal to the area of the triangle, $\frac{1}{2}(2dv)(1/dv) = 1$.



4 Modeling asset price with stochastic volatility

Now we model the full Heston model, which is

$$\begin{cases} dX_t = \mu X_t dt + \sqrt{v_t} X_t dW_t^X \\ dv_t = \kappa(\theta - v_t)dt + \xi\sqrt{v_t} dW_t^v \end{cases} \tag{16}$$

Here, X_t is the price of the stock and v_t is its volatility.

To simplify the calculations, we will drop the drift term in the stock price equation, since this term will not affect the shape of our solution, but will merely shift it. We can later do a simple calculation to find out how the solution is shifted, without the need to incorporate this aspect into the Kolmogorov solution calculations, which we will do shortly. The first equation then becomes

$$dX_t = \sqrt{v_t} X_t dW_t^X \tag{17}$$

4.1 Calculations

We have the Heston stochastic volatility model:

$$\begin{cases} dX = \sqrt{v_t} X_t dW \\ dV = K(\theta - v_t)dt + \xi \sqrt{v_t} dW \end{cases}$$

From the Kolmogorov forward equation, we get

$$p_t = \frac{1}{2}\left(vx^2p\right)_{xx} + \frac{1}{2}\left(\xi^2 vp\right)_{vv} + \rho\left(v\xi xp\right)_{xv} - \left(\kappa(\theta - v)p\right)_v \tag{18}$$

Now we use a transformation into the Kolmogorov backward equation. Let

$$f(t,x,v) = \alpha(v)\exp\left(-\int_0^v \frac{b}{\alpha}du\right)p(t,x,v) \tag{19}$$

Applying this transformation, we get

(20)

$$f_t = \alpha(v)\exp\left(-\int_0^v \frac{b}{\alpha}du\right)p_t$$
$$= \alpha(v)\exp\left(-\int_0^v \frac{b}{\alpha}du\right)\left[\left(\frac{x^2}{\xi^2}\exp\left(\int_0^v \frac{b}{\alpha}du\right)f\right)_{xx} + \left(\exp\left(\int_0^v \frac{b}{\alpha}du\right)f\right)_{vv} + \frac{2\rho}{\xi}\left(x\exp\left(\int_0^v \frac{b}{\alpha}du\right)f\right)_{xv} + \left(\frac{b}{\alpha}\exp\left(\int_0^v \frac{b}{\alpha}du\right)f\right)_v\right]$$

This simplifies to

(21)

$$f_t = vf + 2vxf_x + \frac{1}{2}vx^2f_{xx} + \alpha f_{vv} + \frac{2\rho\alpha}{\xi}xf_{xv} + \beta f_v + \frac{2\rho}{\xi}\alpha f_v + \frac{2\rho}{\xi}\beta xf_x + \frac{2\rho}{\xi}\beta f$$

And rearranging

(22)

$$f_t = (v + \frac{2\rho}{\xi}\beta)f + (2vx + \frac{2\rho}{\xi}\beta x)f_x + (\beta + \frac{2\rho}{\xi}\alpha)f_v + \frac{1}{2}vx^2f_{xx} + \alpha f_{vv} + \frac{2\rho\alpha}{\xi}xf_{xv}$$

We'll denote the coefficient of f as C, the coefficient of f_x as C_x, the coefficient of f_v as C_v, the coefficient of f_{xv} as C_{xv}.

The single and double derivatives with respect to the same variables can be numerically approximated just like in the previous model. There is now also a double derivative with respect to two different variables, f_{xv}. We approximate it as

$$f_{xv} = \frac{f_{j+1,k+1} - f_{j+1,k-1} - f_{j-1,k+1} + f_{j-1,k-1}}{4 * \partial x * \partial v} \tag{23}$$

We therefore end up with our numerical implementation equation as

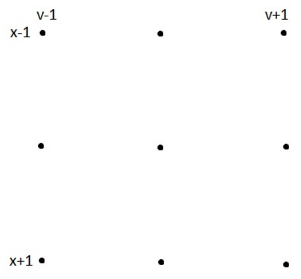
(24)

$$f^{n+1}_{j,k} = f^n_{j,k}\left[1 + Cdt - vx^2\frac{dt}{dx^2} - 2\alpha\frac{dt}{dv^2}\right] + f^n_{j+1,k}\left[C_x\frac{1}{2}\frac{dt}{dx} + \frac{1}{2}vx^2\frac{dt}{dx^2}\right] + f^n_{j-1,k}\left[-C_x\frac{1}{2}\frac{dt}{dx} + \frac{1}{2}vx^2\frac{dt}{dx^2}\right] + f^n_{j,k+1}\left[C_v\frac{1}{2}\frac{dt}{dv} + \alpha\frac{dt}{dv^2}\right] + f^n_{j,k-1}\left[-C_v\frac{1}{2}\frac{dt}{dv} + \alpha\frac{dt}{dv^2}\right] + C_{xv}f^n_{j+1,k+1} - C_{xv}f^n_{j+1,k-1} - C_{xv}f^n_{j-1,k+1} + C_{xv}f^n_{j-1,k-1}$$

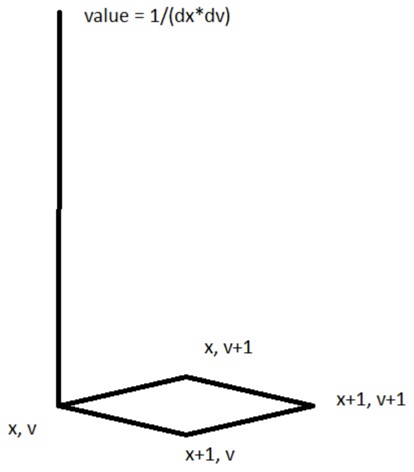
4.2 Implementation

The number of x-steps is denoted x, and the number of v-steps is denoted v. To find the evolution of a solution in two dimensions using a finite differences method, we created a 4-dimensional matrix of size 3*3*x*v. This was necessary since in our solution matrix that is a grid of size x*v, at each time-step each value in this grid is a function of the corresponding value from the previous time-step and all 8 adjacent values.

The 3*3 matrix that is created for each point in the solution matrix is shown below.



The Dirac delta function is approximated by having the starting point within the initial solution matrix equal to the value $\frac{1}{\partial x * \partial v}$. Therefore, integration over the single unit of the grid on which it lies will give a total density of one.

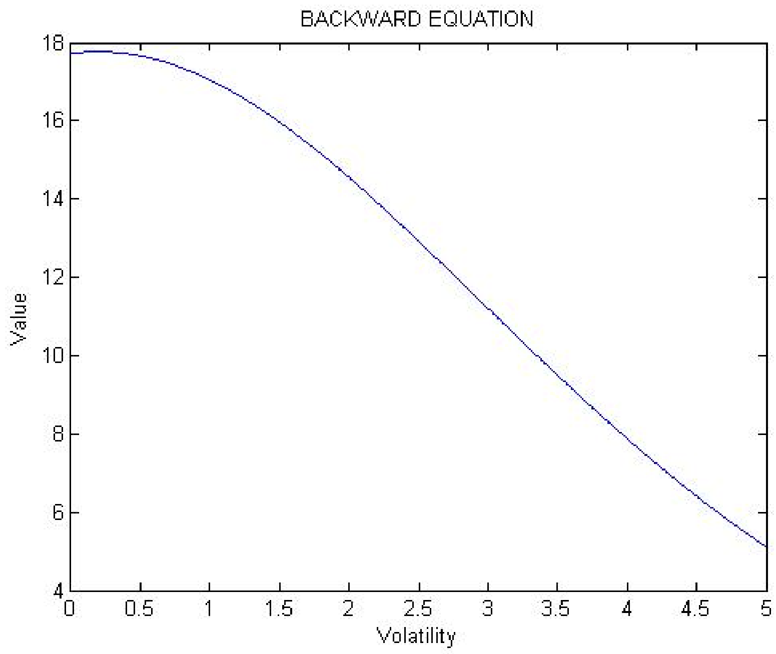


Boundary conditions are calculated as before, except that the method used for boundary $v = 0$ is now used for boundary $x = 0$. Boundary conditions at $v = 0$ in this case are set to equal 0. This is determined again by plugging in for $x = 0$ and $v = 0$.

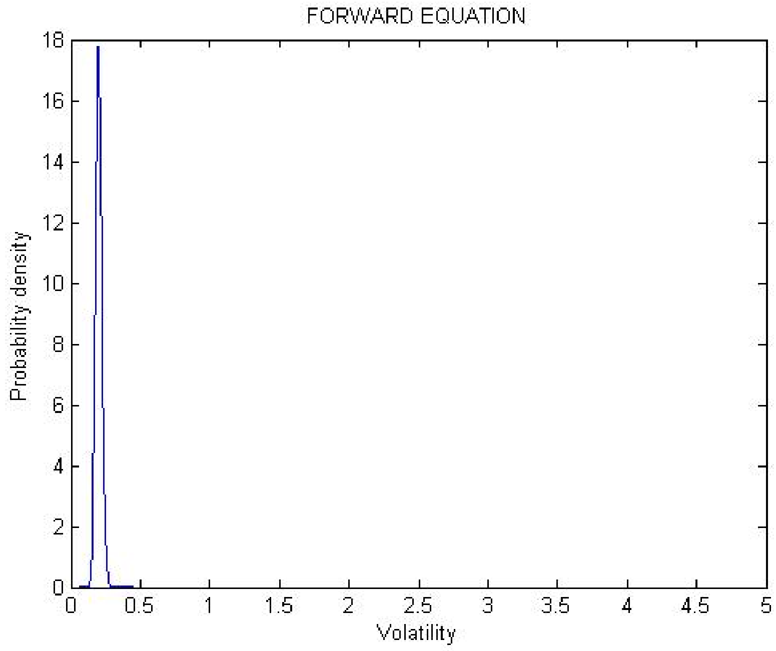
The transformation back to the forward solution is calculated as a function of v, like before, as opposed to a function of x. Therefore, for solution points with differing x-values but the same value of v, the same transformation is applied.

5 Results

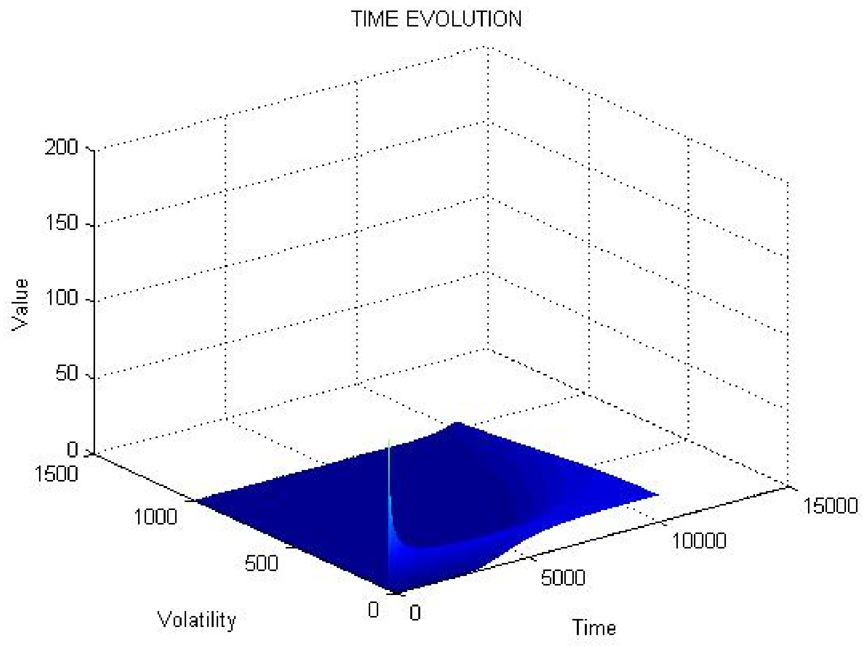
5.1 Stochastic Volatility



The backward solution has obviously spread out from the approximate point mass on which it began onto the whole range of volatility. However, this is an abstract solution with no physical meaning until we convert it back into the forward solution.



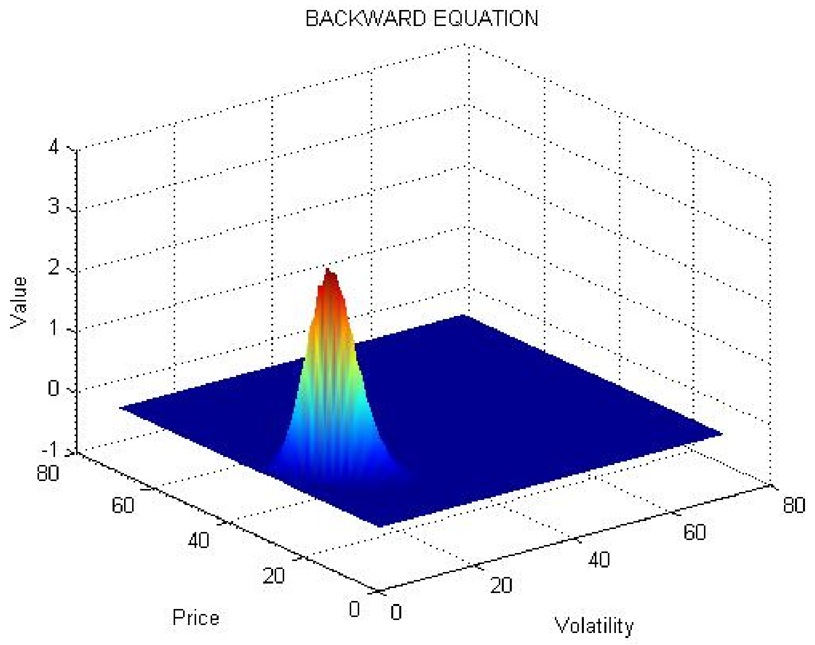
In the forward solution, we see the probability density of volatility at time T. The density is centered around the starting volatility of 0.2.



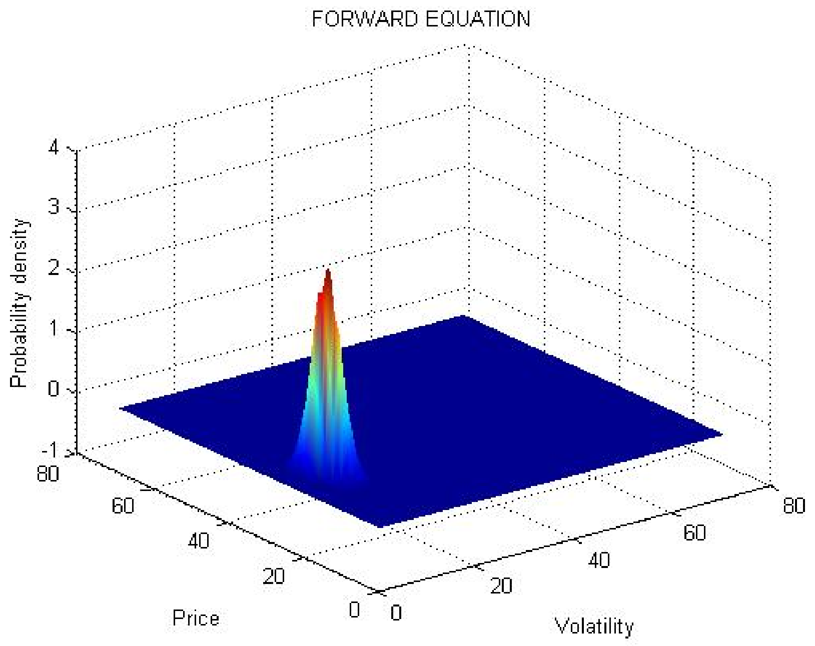
The above graph shows the time evolution of the backward solution. Again, this solution has no physical meaning, but it corresponds with the evolution of the forward solution and therefore shows how the forward solution tends to diffuse.

The trial with the most space-steps and time-steps that we ran had 3000 space-steps and 50000 time-steps. This gave us a total forward solution density of 0.9997, which is convincingly close to one. We expect some loss in density due to discretization error.

5.2 Asset Price and Stochastic Volatility



The backward solution has diffused but remained close to its starting point for a low time-scale.



We can see above the probability density predicted by the forward solution. Volatility becomes higher for the part of the solution that diffuses downward in price.

These figures are for a trial with $x = 70, v = 70, n = 2500$. The total probability density of the backward solution is 2.0024, while the total for the forward solution is 0.9814.

With this implementation, we ran into stability issues resulting in explosions of the backward solution. Therefore, the time horizon is relatively short, limited to $T = 0.07$ years.

6 Conclusion

The numerical solution of the Kolmogorov equation in one dimension (volatility only) that we have attained, with the sum of the density at 0.9997, is strong proof for the validity of the transformation. Taking this method to two dimensions, the solution we get with density summing to 0.9814 strongly hints that the method can model both price and volatility. However, it is not close enough to 1 to be conclusive.

Further study of the topic is needed to ensure the validity of the Heston probability density found. We need the solution to sum closer to 1, and on indefinite time-scales. We employed an explicit finite difference method, but we believe that much more stable and accurate solutions can be attained by using an implicit method and other numerical analysis improvements. Additionally, a comparison with historical data may give evidence supporting the solution.

A Code for Stochastic Volatility

```
clc;
clear all;
close all;

kappa = 8;
theta = 0.2;
xi = 0.2;
T = 0.6;

m = 1000; %space-steps
n = 10000; %time-steps

Vmax = 5;
dv = Vmax/m;
dt = T/n;
V = 0:dv:Vmax;
V = V';

%Matrix (begin)
A = zeros(m+1, m+1);
for i = 1:m
    t1 = xi^2/2*V(i)*dt/dv^2;
    t2 = kappa*(theta-V(i))*dt/(2*dv);
    a = t1-t2;
    b = 1-2*t1;
    c = t1+t2;

    A(i,i) = b;

    if (i > 1)
        A(i,i-1) = a;
    end

    if (i < m+1)
        A(i,i+1) = c;
    end
end
t1 = xi^2/2*V(m+1)*dt/dv^2;
t2 = kappa*(theta-V(m+1))*dt/(2*dv);
A(m+1,m-2) = -t1;
```

```
A(m+1,m-1) = 4*t1 + t2;
A(m+1,m)   = -5*t1-4*t2;
A(m+1,m+1) = 1+2*t1+3*t2;
%Matrix (end)

%The boundary conditions at t=0
delta = 1/dv;
p = zeros(m+1,1);
p(round(m*theta/Vmax + 1)) = delta;

for t = 1:1:n+1
    pmatrix(:,t) = p;
    q = kappa*theta*(p(2) - p(1))*dt/dv + p(1);
    p = A*p;
    p(1) = q;
end

%Transforming back to the forward equation
fun = @(v) kappa*(theta - v)./(xi.^2./2*v);

for i = 2:1:m+1
    q = integral(fun, theta, V(i));
    f(i) = theta/(V(i)) * exp(q) * p(i);
end

%Finding the total density
total = 0;
totalf = 0;
for i = 1:1:m
    total = total + (p(i) + p(i+1))/2 * dv;
    totalf = totalf + (f(i) + f(i+1))/2 * dv;
end
total
totalf

%Plotting the backward and forward solutions
plot(V(1:end), p);
title('BACKWARD EQUATION')
xlabel('Volatility')
ylabel('Value')
figure(2)
plot(V(1:end), f);
title('FORWARD EQUATION')
xlabel('Volatility')
ylabel('Probability density')
```

```
%Plotting the evolution of the backward solution
figure(3)
surf(pmatrix);
shading INTERP;
grid on;
title('TIME EVOLUTION')
xlabel('Time')
ylabel('Volatility')
zlabel('Value')
```

B Code for Asset Price and Stochastic Volatility

```
clc;
clear all;
close all;

kappa = 10;
theta = 0.2;
xi = 0.5;
rho = -0.5;
T = 0.07;

x = 50; %space-steps in x
v = 50; %space-steps in v
n = 1300; %time-steps

dt = T/n;

Xmax = 30;
dx = Xmax/x;
X = 0:dx:Xmax;
X = X';

Vmax = 2;
dv = Vmax/v;
V = 0:dv:Vmax;

%Matrix (begin)
A = zeros(3, 3, x, v);
for j = 1:x
    for k = 1:v
        alpha = xi^2*V(k)/2;
        beta = kappa*(theta-V(k));
        t1 = (V(k) + 2*rho/xi*beta);
```

```
t2 = (2*V(k)*X(j) + 2*rho/xi*beta*X(j));
t3 = (beta + 2*rho/xi*alpha);
t4 = 1/2*rho*alpha/xi*X(j)*dt/dx/dv;

%middle point
A(2,2,j,k) = 1 + t1*dt - V(k)*X(j)^2*dt/dx^2 - 2*alpha*dt/dv^2;

%bottom
if (j < x) %j+1,k
    A(3,2,j,k) = t2 /2*dt/dx + 1/2*V(k)*X(j)^2*dt/dx^2;
end

%top
if (j > 1) %j-1,k
    A(1,2,j,k) = -t2 /2*dt/dx + 1/2*V(k)*X(j)^2*dt/dx^2;
end

%right
if (k < v) %j,k+1
    A(2,3,j,k) = t3 /2*dt/dv + alpha*dt/dv^2;
end

%left
if (k > 1) %j,k-1
    A(2,1,j,k) = -t3 /2*dt/dv + alpha*dt/dv^2;
end

%upper left
if (j > 1 && k > 1) %j-1,k-1
    A(1,1,j,k) = t4;
end

%upper right
if (j > 1 && k < v) %j-1,k+1
    A(1,3,j,k) = -t4;
end

%lower left
if (j < x && k > 1) %j+1,k-1
    A(3,1,j,k) = -t4;
end

%lower right
if (j < x && k < v) %j+1,k+1
    A(3,3,j,k) = t4;
end
```

```
    end
end
%Matrix (end)

%Initializing the solution matrix
p = zeros(x,v+1);
p(round(x*10/Xmax + 1), round(v*theta/Vmax + 1)) = 1/(dx*dv);
pmatrix = zeros(x,v+1,n);

%Calculations
for t = 1:n
    pmatrix(:,:,t) = p;

    pbound = kappa*theta*(p(:,2)-p(:,1))*dt/dx + p(:,1);

    ptemp = zeros(x+2,v+3);
    ptemp(2:x+1,2:v+2) = p;
    ptemp1 = p;
    for j = 1:x
        for k = 1:v
            psum = A(:,:,j,k).*ptemp(j:j+2,k:k+2);
            ptemp1(j,k) = sum(psum(:));
        end
    end
    p = ptemp1;

    p(1,:) = 0;
    p(:,1) = pbound;

    p(x,:) = 0;
    p(:,v) = 0;
end

%Transforming back to forward equation
f = zeros(x,v+1);
fun = @(v) kappa*(theta - v)./(xi.^2./2*v);
for i = 2:1:v+1
    q = integral(fun, theta, V(i));
    f(:,i) = theta/(V(i)) * exp(q) * p(:,i);
    %f(i,:) = theta/(V(i)) * exp(q) * p(i,:);
end

%Finding the total density
total = 0;
totalf = 0;
for i = 1:1:x-1
```

```
    for j = 1:1:v
        total = total + (p(i,j) + p(i+1,j) + p(i,j+1) + p(i+1,j+1)) / 4 * dx * dv;
        totalf = totalf + (f(i,j) + f(i+1,j) + f(i,j+1) + f(i+1,j+1)) / 4 * dx * dv;
    end
end
total
totalf

for i = 1:n
    pgraph = pmatrix(:,:,i);
    surf(pgraph);
    shading INTERP;
    grid on;
    axis([0,v,0,v,0,3/(4*dx*dv)]);
    xlabel('V');
    ylabel('X');
    zlabel('Value')
    drawnow
end
surf(p);
shading INTERP;
grid on;
title('BACKWARD EQUATION')
xlabel('Volatility')
ylabel('Price')
zlabel('Value')

figure(2);
surf(f);
shading INTERP;
grid on;
title('FORWARD EQUATION')
xlabel('Volatility')
ylabel('Price')
zlabel('Probability density')
```

References

[1] Erik Ekström and Johan Tysk, *Boundary behaviour of densities for non-negative diffusions*. Uppsala University, Uppsala, Sweden, 2012.

[2] Tomas Björk, *Arbitrage Theory in Continuous Time*. Oxford University Press, 2009.

[3] Steven Heston, *A closed-form solution for options with stochastic volatility with applications to bond and currency options*. The Review of Financial Studies 6, 327-343, 1993.