Negative Coefficients in Two Factor Option Pricing Models*

R. Zvan†
Financial Analytics and Structured Transactions
Bear Stearns
245 Park Avenue
New York, NY
U.S.A. 10167

P.A. Forsyth‡
Department of Computer Science
University of Waterloo
Waterloo, ON
Canada N2L 3G1

K.R. Vetzal§
Centre for Advanced Studies in Finance
University of Waterloo
Waterloo, ON
Canada N2L 3G1

February 16, 2001

Abstract

The importance of positive coefficients in numerical schemes is frequently emphasized in the finance literature. This topic is explored in detail in this paper, in the particular context of two factor models. First, several two factor lattice type methods are derived using a finite difference/finite element methodology. Some of these methods have negative coefficients, but are nevertheless stable and consistent. Second, we outline the conditions under which finite volume/element methods applied to two factor option pricing partial differential equations give rise to discretizations with positive coefficients. Numerical experiments indicate that constructing a mesh which satisfies positive coefficient conditions may not only be unnecessary, but in some cases even detrimental. As well, it is shown that schemes with negative coefficients due to the discretization of the diffusion term satisfy approximate local maximum and minimum conditions as the mesh spacing approaches zero. This finding is of significance since, for arbitrary diffusion tensors, it may not be possible to construct a positive coefficient discretization for a given set of nodes.

Acknowledgment: This work was supported by the Natural Sciences and Engineering Research Council of Canada, the Social Sciences and Humanities Research Council of Canada, and the Royal Bank of Canada.

*A previous version of this paper was entitled "Diffusion Operators and Meshes in Option Pricing".

†rzvan@bear.com
‡paforsyt@elora.math.uwaterloo.ca
§kvetzal@watarts.uwaterloo.ca

1 Introduction

The connection between the existence of an equivalent martingale measure and the absence of arbitrage is one of the cornerstones of modern financial asset pricing theory. Basically, this means that the value of an asset can be found by calculating its expected future payoff under a particular probability measure (determining, in effect, the certainty equivalent of the payoff) and discounting at the risk free rate of interest. In other words, prices can always be determined, at least in principle, by taking some positively-weighted combination of possible future payoffs and discounting.

When dealing with continuous time models, asset values may also be expressed in terms of partial differential equations (PDEs). It is often necessary to use numerical techniques to solve these PDEs, as analytic solutions are not generally available. This naturally leads to some form of discretization. It is frequently emphasized in the finance literature that the coefficients in the discrete equations should be nonnegative if an explicit type of method is used (Brennan and Schwartz, 1978; Hull and White, 1990; Boyle and Tian, 1998). A vastly longer list of papers could be cited here, if one recognizes that binomial and trinomial trees are simply versions of explicit finite difference schemes.[1] Indeed, the coefficients in these schemes are generally viewed as being equivalent martingale probabilities.

The nonnegativity of coefficients is a sufficient condition for the stability of a consistent explicit scheme (Forsythe and Wasow, 1967). For the one-dimensional diffusion equation (with linear basis functions), nonnegativity is also necessary for stability. Although nonnegative coefficients are only sufficient conditions for the stability of explicit schemes in two dimensions, they are also sufficient conditions for schemes of arbitrary temporal weighting to possess discrete local maximum and minimum principles (see Rafferty et al., 1985; Forsyth, 1991; Putti and Cordes, 1998; Zvan et al., 1998, 2000)). This implies that the numerical solution will not be contaminated by spurious oscillations. It is also worth noting that there exist two-dimensional and higher dimensional explicit schemes that do not satisfy the positive coefficient condition, but which are nonetheless stable (Brandt, 1973) (though possibly oscillatory).

This article analyses negative contributions to discrete coefficients arising from diffusive terms. This issue is important in the context of pricing problems in more than one dimension. Due to the presence of cross-partial derivatives, it may not be possible to construct a finite difference or finite element scheme with nonnegative coefficients. One important case where it is possible is geometric Brownian motion, since transformations can be applied to eliminate the cross-partial terms.[2] However, applying these transformations may result in numerical solutions of poor quality due to grid nodes being placed in sub-optimal locations. Moreover, there are other important cases such as stochastic volatility models (e.g. Heston, 1993) where such transformations do not appear to be possible. Consequently, it is of interest to explore the effects of negative contributions to discrete coefficients arising from diffusive terms. Specifically, this paper investigates whether such contributions deteriorate the quality of solutions for two-dimensional pricing problems, relative to solutions computed when the discretization of diffusion ensures that the coefficients will be positive.

[1] The equivalence of trinomial trees and explicit finite difference methods is generally recognized (Hull, 2000). However, the fact that binomial trees are also forms of standard explicit methods seems to be much less widely known. A demonstration of this is provided in Heston and Zhou (2000) for the Black-Scholes PDE after a transformation to the heat equation.

[2] Much of the finance literature on option pricing in more than one dimension attempts to generalize one-dimensional trees using such transformations to ensure that coefficients are positive (Boyle, 1988; Amin, 1991; Hull and White, 1994; Gao, 1997; Imai, 1997). Note that it can be shown (see Zvan, 2000) that certain positive probability multidimensional lattice schemes such as those in Gao (1997); Imai (1997) are in fact explicit finite element schemes using *skewed* or nonorthogonal quadrilateral meshes.

Option pricing PDEs are typically convection-diffusion type equations.[3] It is important to observe that convection and diffusion have very different effects. Roughly speaking, in a physical context convection is movement or flow in a particular direction, whereas diffusion causes spreading out or scattering. This distinction is obscured by viewing the discrete coefficients as martingale probabilities because the probabilities reflect the contributions of both of these phenomena.

Even if a numerical scheme is stable, it can be prone to spurious oscillations if negative coefficients are present. This can lead to inaccuracy (especially with respect to the derivatives of the numerical solution) and slow convergence. We can isolate three distinct sources of spurious oscillations in numerical computation of option prices:

- Convection dominance (large drift terms compared to the diffusion terms). Various techniques exist to handle problems arising from negative contributions to discrete coefficients due to convective terms. Examples include upstream weighting and flux limiters (see Zvan et al., 1998, 2000, for further details).

- Crank-Nicolson timestepping, if the timesteps are larger than double the maximum stable explicit timestep size, can generate oscillations if the payoff is non-smooth. The methods in Rannacher (1984) can be used to alleviate this source of oscillations.[4]

- Negative coefficients arising from the discretization of the diffusion terms. This source of problem is not easily remedied.

In the following we focus on negative coefficients arising from the discretization of the diffusion terms. We show that:

- A variety of lattice type schemes can be derived using explicit finite difference/finite element methods, all relying on a consistent methodology, without appealing to probabilistic arguments. Several of these lattice methods are equivalent (locally to $O\left[(\Delta t)^2\right]$ where Δt is the timestep size) to known schemes. Some of these lattice schemes have negative coefficients, but are stable and convergent.

- Schemes with negative coefficients due to the discretization of the diffusion term satisfy approximate local maximum and minimum principles as the mesh spacing approaches zero.

- Constructing meshes to ensure that coefficients (of the discretized diffusion operator) are positive does not appear to produce solutions of better quality, compared to solutions obtained without forcing the positive coefficient condition. Tests carried out using an implicit finite volume method (FVM) show that, under certain criteria, the solutions obtained with a positive coefficient discretization may be of poor quality. Placing nodes according to features of the problem appears to be more important than obtaining positive coefficients.

Note that implicit finite difference methods have occasionally been used for multidimensional problems in finance (Brennan and Schwartz, 1980; Wilmott et al., 1993; Dempster and Hutton, 1997), but it has recently been demonstrated in Zvan et al. (2000) that the FVM has several advantages over finite difference schemes in this context.

[3] Readers unfamiliar with this terminology should think of the convective term involving first order derivatives and the diffusive term involving second order derivatives.

[4] Alternatively, a second order BDF type timestepping method (which is strongly A-stable) can also be used (Tavella and Randall, 2000), though this can be a poor choice for some types of American options (Windcliff et al., 1999).

2 Lattice Methods

Two factor lattice methods are typically derived using probabilistic arguments. In this section, we will explore the use of a finite difference/finite element approach to derive two factor lattice methods. We will show that some methods previously described in the literature can be derived (to $O\left[(\Delta t)^2\right]$) using our approach. We will also derive some new methods. All of the methods which we will describe have the same discrete form.

Assume that we have two factors S_1, S_2 which follow processes

$$\begin{aligned} dS_1 &= \mu_{S_1} S_1 dt + \sigma_{S_1} S_1 dW_{S_1} \\ dS_2 &= \mu_{S_2} S_2 dt + \sigma_{S_2} S_2 dW_{S_2}, \end{aligned}$$

where μ_{S_1} and μ_{S_2} are expected returns, σ_{S_1} and σ_{S_2} are volatilities, and W_{S_1} and W_{S_2} are Wiener processes with correlation ρ. Then the underlying PDE for the price of a contingent claim U written on S_1, S_2 has the form

$$-U_t + \mathbf{V} \cdot \nabla U = (\mathbf{D}\nabla) \cdot \nabla U - rU, \tag{2.1}$$

where $\mathbf{V}$ is the velocity tensor, ∇ is the gradient operator, $\mathbf{D}$ is the diffusion tensor, and r is the risk free interest rate. Equation (2.1) is a convection-diffusion equation, where $-\mathbf{V} \cdot \nabla U$ is the convection term and $(\mathbf{D}\nabla) \cdot \nabla U$ is the diffusion term.

Defining the gradient operator as $\nabla = (\partial/\partial S_1, \partial/\partial S_2)'$, the price of an option based on two underlying assets in the original variables (S_1, S_2) has the form of equation (2.1) with

$$\mathbf{V} = -\begin{pmatrix} rS_1 \\ rS_2 \end{pmatrix} \quad \text{and} \quad \mathbf{D} = \frac{1}{2}\begin{pmatrix} \sigma_{S_1}^2 S_1^2 & \rho\sigma_{S_1}\sigma_{S_2} S_1 S_2 \\ \rho\sigma_{S_1}\sigma_{S_2} S_1 S_2 & \sigma_{S_2}^2 S_2^2 \end{pmatrix}. \tag{2.2}$$

Alternatively, let $x = \log(S_1), y = \log(S_2)$, and define the gradient operator as $\nabla = (\partial/\partial x, \partial/\partial y)'$. Then the PDE in terms of the (x, y) variables has the same form as equation (2.1) with

$$\mathbf{V} = -\begin{pmatrix} r - \sigma_{S_1}^2/2 \\ r - \sigma_{S_2}^2/2 \end{pmatrix} \quad \text{and} \quad \mathbf{D} = \frac{1}{2}\begin{pmatrix} \sigma_{S_1}^2 & \rho\sigma_{S_1}\sigma_{S_2} \\ \rho\sigma_{S_1}\sigma_{S_2} & \sigma_{S_2}^2 \end{pmatrix}. \tag{2.3}$$

2.1 Hull and White Method

Hull and White (1990) suggest that a further transformation be carried out to diagonalize the tensor $\mathbf{D}$ in (2.3). With x, y defined as above, let $\psi_1 = \sigma_{S_2} x + \sigma_{S_1} y$ and $\psi_2 = \sigma_{S_2} x - \sigma_{S_1} y$. This corresponds to a stretching and rotation of the coordinate system. After carrying out this transformation, and defining $\nabla = (\partial/\partial\psi_1, \partial/\partial\psi_2)'$, the PDE again has the same form as equation (2.1) with

$$\mathbf{V} = -\begin{pmatrix} (r - \sigma_{S_1}^2/2)\sigma_{S_2} + (r - \sigma_{S_2}^2/2)\sigma_{S_1} \\ (r - \sigma_{S_1}^2/2)\sigma_{S_2} - (r - \sigma_{S_2}^2/2)\sigma_{S_1} \end{pmatrix} \quad \text{and} \quad \mathbf{D} = \begin{pmatrix} (1+\rho)\sigma_{S_1}^2\sigma_{S_2}^2 & 0 \\ 0 & (1-\rho)\sigma_{S_1}^2\sigma_{S_2}^2 \end{pmatrix}. \tag{2.4}$$

Letting $U_{ij}^n = U(i\Delta\psi_1, j\Delta\psi_2, n\Delta t)$, we discretize equation (2.1) using an explicit method

$$-\left(\frac{U_{ij}^{n+1} - U_{ij}^n}{\Delta t}\right) = (-\mathbf{V} \cdot \nabla U + (\mathbf{D}\nabla) \cdot \nabla \mathrm{U})_{ij}^{n+1} - rU_{ij}^n. \tag{2.5}$$

Equation (2.5) with diffusion and convection tensors (2.4) can be easily discretized in the (ψ_1, ψ_2) plane since there is no cross derivative term. For example,

$$(U_{\psi_1,\psi_1})^n_{ij} = \frac{U^n_{i+1,j} - 2U^n_{i,j} + U^n_{i-1,j}}{(\Delta\psi_1)^2} + O\left[(\Delta\psi_1)^2\right]$$
$$(U_{\psi_1})^n_{ij} = \frac{U^n_{i+1,j} - U^n_{i-1,j}}{2\Delta\psi_1} + O\left[(\Delta\psi_1)^2\right], \tag{2.6}$$

with similar expressions for the ψ_2 derivatives. Let

$$\mu_{\psi_1} = (r - \frac{\sigma^2_{S_1}}{2})\sigma_{S_2} + (r - \frac{\sigma^2_{S_2}}{2})\sigma_{S_1} \qquad \sigma_{\psi_1} = \sigma_{S_1}\sigma_{S_2}\sqrt{2(1+\rho)}$$
$$\mu_{\psi_2} = (r - \frac{\sigma^2_{S_1}}{2})\sigma_{S_2} - (r - \frac{\sigma^2_{S_2}}{2})\sigma_{S_1} \qquad \sigma_{\psi_2} = \sigma_{S_1}\sigma_{S_2}\sqrt{2(1-\rho)}. \tag{2.7}$$

Then, choosing $\Delta\psi_1 = \sigma_{\psi_1}\sqrt{2\Delta t}$ and $\Delta\psi_2 = \sigma_{\psi_2}\sqrt{2\Delta t}$, and applying some algebraic manipulation, we obtain the discrete equations (locally correct to $O\left[(\Delta t)^2\right]$)

$$U^n_{i,j} = e^{-r\Delta t}\sum_{\alpha=-1}^{\alpha=+1}\sum_{\beta=-1}^{\beta=+1} W_{\alpha\beta}U^{n+1}_{i+\alpha,j+\beta} \tag{2.8}$$

where $i = -n, \ldots, +n$, $j = -n, \ldots, +n$ and the weights $W_{\alpha\beta}$ are given by

$$W_{+1,0} = \frac{1}{4}\left(1 + \frac{\mu_{\psi_1}2\sqrt{\Delta t}}{\sigma_{\psi_1}\sqrt{2}}\right) \qquad W_{0,+1} = \frac{1}{4}\left(1 + \frac{\mu_{\psi_2}2\sqrt{\Delta t}}{\sigma_{\psi_2}\sqrt{2}}\right)$$
$$W_{-1,0} = \frac{1}{4}\left(1 - \frac{\mu_{\psi_1}2\sqrt{\Delta t}}{\sigma_{\psi_1}\sqrt{2}}\right) \qquad W_{0,-1} = \frac{1}{4}\left(1 - \frac{\mu_{\psi_2}2\sqrt{\Delta t}}{\sigma_{\psi_2}\sqrt{2}}\right), \tag{2.9}$$

with all other weights $W_{\alpha\beta} = 0$.[5] To apply the payoff condition, we must be able to determine the original asset values at each node. These can be determined by noting that

$$(\psi_1)^n_{ij} = \sigma_{S_2}\log\left((S_1)^0_{0,0}\right) + \sigma_{S_1}\log\left((S_2)^0_{0,0}\right) + i\sigma_{\psi_1}\sqrt{2\Delta t}$$
$$(\psi_2)^n_{ij} = \sigma_{S_2}\log\left((S_1)^0_{0,0}\right) - \sigma_{S_1}\log\left((S_2)^0_{0,0}\right) + j\sigma_{\psi_2}\sqrt{2\Delta t}$$
$$(S_1)^n_{ij} = \exp\left(\frac{(\psi_1)^n_{ij} + (\psi_2)^n_{ij}}{2\sigma_{S_2}}\right)$$
$$(S_2)^n_{ij} = \exp\left(\frac{(\psi_1)^n_{ij} - (\psi_2)^n_{ij}}{2\sigma_{S_1}}\right). \tag{2.10}$$

The weights (2.9) are all nonnegative and sum to one if

$$\left|\frac{\mu_{\psi_1}2\sqrt{\Delta t}}{\sigma_{\psi_1}\sqrt{2}}\right| \le 1, \quad \left|\frac{\mu_{\psi_2}2\sqrt{\Delta t}}{\sigma_{\psi_2}\sqrt{2}}\right| \le 1. \tag{2.11}$$

[5]Note that in equation (2.8) we have used the approximation $1/(1+r\Delta t) = e^{-r\Delta t} + O\left[(\Delta t)^2\right]$. Also, we can avoid some unnecessary work by specifying the additional requirement for the indicies that $i+j$ be even if n is even and $i+j$ be odd if n is odd.

For typical parameter values, this is not very restrictive. Consequently, assuming (2.11) holds, it is obvious that this discretization is stable. We can then interpret the weights as (risk neutral) probabilities. In the following, any scheme which (with mild restrictions) has the form (2.8) with nonnegative weights, will be termed a *positive coefficient* discretization.

Restriction (2.11) can be eliminated if the first order terms are discretized appropriately. Note that

$$\begin{aligned} dU &= U_t\ dt + U_x\ dx + U_y\ dy \\ &= dt\left(U_t + U_x\ \frac{dx}{dt} + U_y\ \frac{dy}{dt}\right) \\ &= dt\left(U_t + \mathbf{q}\cdot\nabla U\right), \end{aligned} \tag{2.12}$$

where $(\mathbf{q})_x = \frac{dx}{dt}$, $(\mathbf{q})_y = \frac{dy}{dt}$. The term $U_t + \mathbf{q}\cdot\nabla U$ can then be discretized as

$$\begin{aligned} \left(\frac{dU}{dt}\right)_{ij}^n &= \left(U_t + \mathbf{q}\cdot\nabla U\right)_{ij}^n \\ &= \frac{U_{ij}^{n+1} - U_{ij}^n}{\Delta t}, \end{aligned} \tag{2.13}$$

where $U_{ij}^n = U(x_{ij}^n, y_{ij}^n, t_n)$ and $x_{ij}^{n+1} = x_{ij}^n + (\mathbf{q})_x\Delta t$, $y_{ij}^{n+1} = y_{ij}^n + (\mathbf{q})_y\Delta t$. In our case, we can write the term $-U_t + \mathbf{V}\cdot\nabla U$ as

$$\begin{aligned} -\left(\frac{dU}{dt}\right)_{ij} &= -\left(U_t + \mathbf{q}\cdot\nabla U\right)_{ij}^n \\ &= \left(-U_t - (-\mathbf{V})\cdot\nabla U\right)_{ij}^n \\ &= -\left(\frac{U_{ij}^{n+1} - U_{ij}^n}{\Delta t}\right) \end{aligned} \tag{2.14}$$

where $x_{ij}^{n+1} = x_{ij}^n + (-\mathbf{V})_x\Delta t$, $y_{ij}^{n+1} = y_{ij}^n + (-\mathbf{V})_y\Delta t$.

The discretization of the term $-U_t + \mathbf{V}\cdot\nabla U$ using equation (2.14) is usually called a *characteristic* approach, since this corresponds to following the solution back along the characteristic. Note that if $\mathbf{V}$ is constant, then equation (2.14) is exact.

2.2 A Finite Difference Lattice Method

Given (2.1), with gradient operator $\nabla = (\partial/\partial x, \partial/\partial y)'$ and diffusion and drift tensors as in equation (2.3), we can derive a *lattice type* method which has the same form as equation (2.8), using a characteristic method for the drift term, and a straightforward finite difference approximation of the diffusion term. We will carry out this discretization in the $x = \log(S_1), y = \log(S_2)$ plane, without rotating the coordinates to remove the cross derivative terms.

Let

$$\begin{aligned} U_{ij}^n &= U(x_{ij}^n, y_{ij}^n, n\Delta t) \\ x_{ij}^n &= x_{0,0}^0 + n\mu_x\Delta t + i\sigma_1\sqrt{2\Delta t} \\ y_{ij}^n &= y_{0,0}^0 + n\mu_y\Delta t + j\sigma_2\sqrt{2\Delta t}. \end{aligned} \tag{2.15}$$

where

$$\mu_x = r - \frac{\sigma_1^2}{2}, \quad \mu_y = r - \frac{\sigma_2^2}{2}. \tag{2.16}$$

We can use equation (2.14) to discretize the term $-U_t + \mathbf{V} \cdot \nabla U$ (noting that $\mathbf{V}_x = -\mu_x$ and $\mathbf{V}_y = -\mu_y$) and centered finite differences for the diffusion terms. Using an explicit method in time, we obtain

$$-\left(\frac{U_{ij}^{n+1} - U_{ij}^n}{\Delta t}\right) = ((\mathbf{D}\nabla) \cdot \nabla U)_{ij}^{n+1} - rU_{ij}^n. \tag{2.17}$$

At any given time level n, the finite difference grid (2.15) is equally spaced in the x and y directions. Consequently, the usual centered finite difference approximations can be used for the U_{xx}, U_{yy} terms. The cross derivative term is approximated by

$$(U_{xy})_{ij}^n = \frac{U_{i+1,j+1}^n - U_{i-1,j+1}^n - U_{i+1,j-1}^n + U_{i-1,j-1}^n}{4\Delta x \Delta y} + O(\Delta t), \tag{2.18}$$

where $\Delta x = \sigma_1\sqrt{2\Delta t}$, $\Delta y = \sigma_2\sqrt{2\Delta t}$. We then obtain a discrete equation of the form (2.8) (to $O\left[(\Delta t)^2\right]$) with

$$\begin{array}{ccc} W_{-1,-1} = \frac{-\rho}{8} & W_{-1,0} = \frac{1}{4} & W_{-1,1} = \frac{\rho}{8} \\ W_{0,-1} = \frac{1}{4} & W_{0,0} = 0 & W_{0,1} = \frac{1}{4} \\ W_{1,-1} = \frac{\rho}{8} & W_{1,0} = \frac{1}{4} & W_{1,1} = \frac{-\rho}{8}. \end{array} \tag{2.19}$$

Some of the coefficients in equation (2.19) are negative if $\rho \neq 0$. Consequently, the usual positive coefficient argument cannot be used to demonstrate that the recursion (2.8) is stable. Following the standard von Neumann analysis, we let

$$U_{ij}^n = \alpha^{N-n} \exp(\sqrt{-1}\ i\theta_x) \exp(\sqrt{-1}\ j\theta_y), \tag{2.20}$$

where N is the number of timesteps, in equation (2.8) with weights (2.19) to obtain

$$|\alpha| \leq \left| \frac{\cos(\theta_x) + \cos(\theta_y) - \rho \sin(\theta_x)\sin(\theta_y)}{2} \right|. \tag{2.21}$$

After some tedious algebra, this implies

$$\max_{0 \leq \theta_x \leq \pi, 0 \leq \theta_y \leq \pi} |\alpha| \leq 1; \quad |\rho| \leq 1 \tag{2.22}$$

and hence the recursion is stable.

2.3 A Finite Element Lattice Method

The presence of negative weights in the finite difference approach described above is at first sight somewhat disconcerting, appearing to violate discrete maximum properties. However, as will be shown below in Section 4, a convergent technique can be expected to satisfy an approximate discrete maximum principle as $\Delta t \to 0$.

These negative weights can be avoided if we use a finite element technique with linear basis

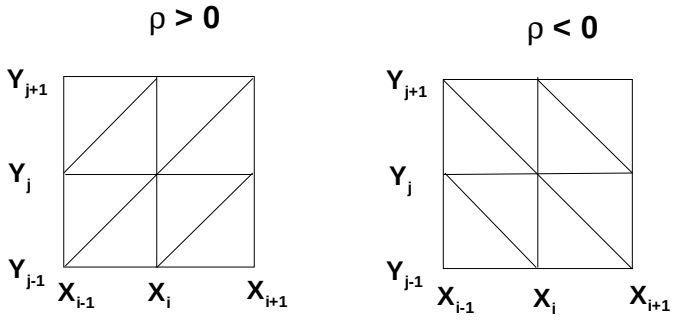


FIGURE 1: *Finite element triangulation for cases $\rho > 0$ and $\rho < 0$.*

functions defined on triangles as shown in Figure 1. Note that the triangulation used changes orientation depending on the sign of ρ. Again, we define U_{ij}^n and x_{ij}^n, y_{ij}^n as in equations (2.15)-(2.16), and use a characteristic method for the $-U_t + \mathbf{V} \cdot \nabla U$ term. Our starting point is the same as for the finite difference method

$$-\left(\frac{U_{ij}^{n+1} - U_{ij}^n}{\Delta t}\right) = ((\mathbf{D}\nabla) \cdot \nabla U)_{ij}^{n+1} - rU_{ij}^n. \tag{2.23}$$

Let N_{ij} be the linear C^0 Lagrange basis functions associated with node (i, j). These have the properties that $N_{ij} = 1$ at node (i, j), $N_{ij} = 0$ at all other nodes, and $\sum_{ij} N_{ij} = 1$ everywhere in the solution domain. The diffusion term is discretized as

$$((\mathbf{D}\nabla) \cdot \nabla U)_{ij}^{n+1} = \sum_{\beta=-1}^{\beta=+1} \sum_{\alpha=-1}^{\alpha=+1} \gamma_{i+\alpha,j+\beta} \left(U_{i+\alpha,j+\beta}^{n+1} - U_{ij}^{n+1}\right) \tag{2.24}$$

where

$$\gamma_{i+\alpha,j+\beta} = \frac{-1}{A_{ij}} \int \nabla N_{ij} \cdot \mathbf{D} \cdot \nabla N_{i+\alpha,j+\beta} \, dx \, dy, \quad A_{ij} = \int N_{ij} \, dx \, dy. \tag{2.25}$$

This gives a discretization of the form (2.8) with

$$\begin{aligned} \rho < 0 \qquad & \begin{Bmatrix} W_{-1,-1} = 0 & W_{-1,0} = \frac{1+\rho}{4} & W_{-1,1} = \frac{-\rho}{4} \\ W_{0,-1} = \frac{1+\rho}{4} & W_{0,0} = \frac{-\rho}{2} & W_{0,1} = \frac{1+\rho}{4} \\ W_{1,-1} = \frac{-\rho}{4} & W_{1,0} = \frac{1+\rho}{4} & W_{1,1} = 0 \end{Bmatrix} \\ \rho > 0 \qquad & \begin{Bmatrix} W_{-1,-1} = \frac{\rho}{4} & W_{-1,0} = \frac{1-\rho}{4} & W_{-1,1} = 0 \\ W_{0,-1} = \frac{1-\rho}{4} & W_{0,0} = \frac{\rho}{2} & W_{0,1} = \frac{1-\rho}{4} \\ W_{1,-1} = 0 & W_{1,0} = \frac{1-\rho}{4} & W_{1,1} = \frac{\rho}{4} \end{Bmatrix}. \end{aligned} \tag{2.26}$$

These weights are always nonnegative (for $|\rho| \leq 1$) and sum to unity, and so this method is stable.

2.4 Gao Lattice Method

Gao (1997) develops a two dimensional lattice scheme. It is shown in Zvan (2000) that this method

can be derived by a characteristic discretization of the $-U_t + \mathbf{V} \cdot \nabla U$ term and discretizing the diffusion term using bilinear basis functions on skewed quadrilaterals. Again, we obtain a method of the form (2.8), with

$$\begin{array}{ccc} W_{-1,-1} = \frac{1}{36} & W_{-1,0} = \frac{1}{9} & W_{-1,1} = \frac{1}{36} \\ W_{0,-1} = \frac{1}{9} & W_{0,0} = \frac{4}{9} & W_{0,1} = \frac{1}{9} \\ W_{1,-1} = \frac{1}{36} & W_{1,0} = \frac{1}{9} & W_{1,1} = \frac{1}{36} \end{array} \tag{2.27}$$

where

$$\begin{aligned} U_{ij}^n &= U(x_{ij}^n, y_{ij}^n, n\Delta t) \\ x_{ij}^n &= x_0^0 + n\mu_x \Delta t + i\sigma_1 \sqrt{3\Delta t} \\ y_{ij}^n &= y_0^0 + n\mu_y \Delta t + (i\rho + j\sqrt{1-\rho^2})\sigma_2 \sqrt{3\Delta t} \end{aligned} \tag{2.28}$$

with μ_x and μ_y being given in equation (2.16).

2.5 Comparison of Lattice Methods

From the above, it is clear that many lattice schemes can be derived using the following steps:

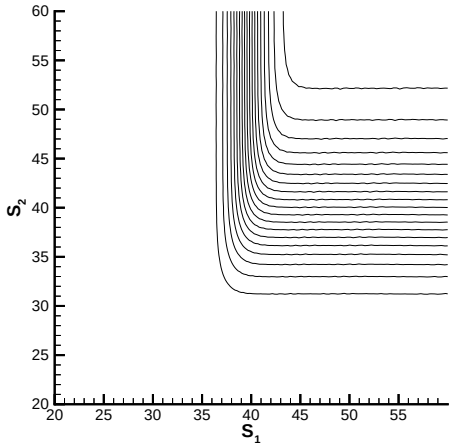
- The coordinates are transformed (usually $x = \log(S_1), y = \log(S_2)$), so that the PDE coefficients become constant in the (x, y) plane. Further rotations can be performed to eliminate cross derivative terms.
- A finite difference or finite element method is used to discretize the diffusion term.
- A finite difference (in the time direction) or characteristic method is used to discretize the convection term.
- Explicit timestepping is used.

There is no need to appeal to probabilistic arguments. Note that this procedure results in discrete schemes which are identical (locally to $O\left[(\Delta t)^2\right]$) to known lattice methods. As well, we can produce several lattice type methods which have not been described previously, at least in the finance literature. In particular, the finite difference lattice method results in some negative weights.

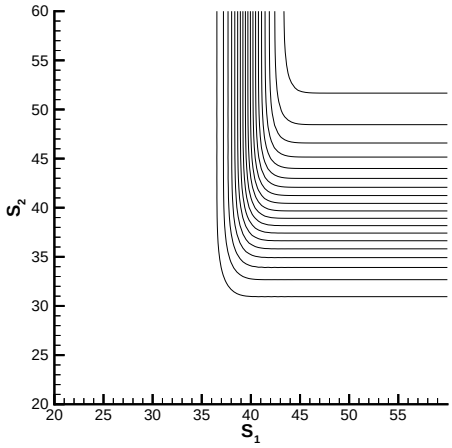
We now proceed to provide some numerical results for the various lattice schemes. We begin with the simple case of a European call on the best of two assets. More precisely, the payoff is $U(S_1, S_2, t = T) = \max(\max(S_1, S_2) - K, 0)$, where K is the strike. Table 1 presents the results. It appears that the Hull and White rotated grid method converges more smoothly than the other lattice methods.

A more difficult pricing problem is a two factor digital call, with payoff $U(S_1, S_2, t = T) = 1$ if $S_1 \geq K$ and $S_2 \geq K$, and 0 otherwise. Convergence results for all four lattice methods are given in Table 2. In this case, all methods show somewhat erratic convergence. This is probably due to the fact that these methods will not, in general, have element faces and nodes which are aligned with the non-smooth payoff. In order to obtain smooth convergence, some projection onto the space of basis functions is generally required (Rannacher, 1984). However, as noted in Wahlbin (1980), it is not easy to carry out the projection operation with sufficient accuracy to restore good convergence if the payoff is not aligned with the mesh nodes.

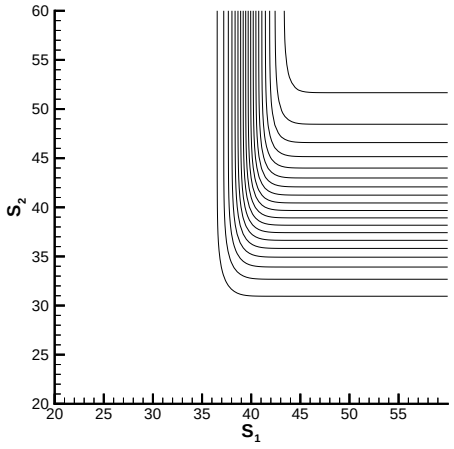
Contours of constant prices are plotted in Figure 2 for all four lattice methods. We obtained the contour values by setting $T = .5$, and then outputting the values at all nodes in the tree at $t = .25$.



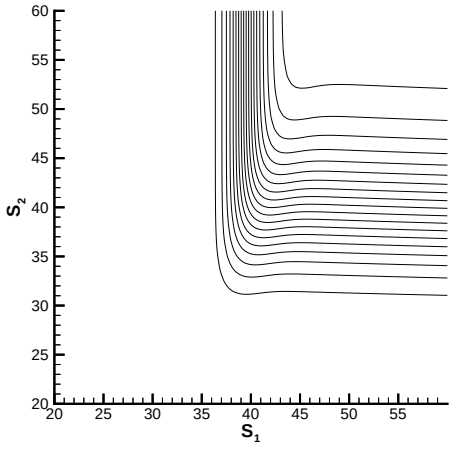
(a) *Hull and White* (2.9).



(b) *Finite Difference* (2.19).



(c) *Finite Element* (2.26).



(d) *Gao* (2.27).

FIGURE 2: *Level curves of values of a European digital call when* $r = 0.05$, $\sigma_{S_1} = 0.10$, $\sigma_{S_2} = 0.30$, $T - t = 0.25$, $K = 40$, $\rho = .7$.

N	H & W	Finite Difference	Finite Element	Gao
40	2.89428	2.89276	2.88776	2.88931
80	2.89286	2.89321	2.89086	2.88946
160	2.89172	2.89092	2.89014	2.89075
320	2.89133	2.89114	2.89063	2.89068

TABLE 1: *Value of European call option on the maximum of two assets when* $r = 0.05$, $\sigma_{S_1} = 0.10$, $\sigma_{S_2} = 0.30$, $T - t = 0.25$, $S_1 = S_2 = 40$, $K = 40$, $\rho = .7$. *Exact solution is* 2.89055 *to six figures.* N *is the number of timesteps. H & W refers to the method* (2.9), *Finite Difference refers to method* (2.19), *Finite Element refers to method* (2.26), *and Gao refers to method* (2.27).

N	H & W	Finite Difference	Finite Element	Gao
40	.407732	.450678	.454974	.405284
80	.410572	.427917	.430310	.429116
160	.411267	.431633	.431561	.411696
320	.411136	.413938	.413938	.416583

TABLE 2: *Value of European digital call when* $r = 0.05$, $\sigma_{S_1} = 0.10$, $\sigma_{S_2} = 0.30$, $T - t = 0.25$, $S_1 = S_2 = 40$, $K = 40$, $\rho = .7$. *Exact solution to six figures* .410929. N *is the number of timesteps. H & W refers to method* (2.9), *Finite Difference refers to method* (2.19), *Finite Element refers to method* (2.26), *and Gao refers to method* (2.27).

160 steps were used. Note that the contours are cut off in some cases due to lack of data for those points. For the finite difference method (2.19), there are no observable oscillations in the contours, even though some of the weights are negative. This would appear to indicate that negative weights are not a serious problem in practice.

Our last lattice test problem is a European call on the maximum of two assets, with discretely observed knockout barriers. The barriers are observed at increments of .025 years. Denoting the upper and lower barriers by H_u and H_l, the value of the option must satisfy the constraint

$$U(S_1, S_2, t_-) = \begin{cases} U(S_1, S_2, t_+) & \text{if } H_l < S_1 < H_u \text{ and } H_l < S_2 < H_u \\ 0 & \text{otherwise,} \end{cases} \tag{2.29}$$

where t_+ and t_- are the times just before and after the application (monitoring date) of a barrier, respectively.

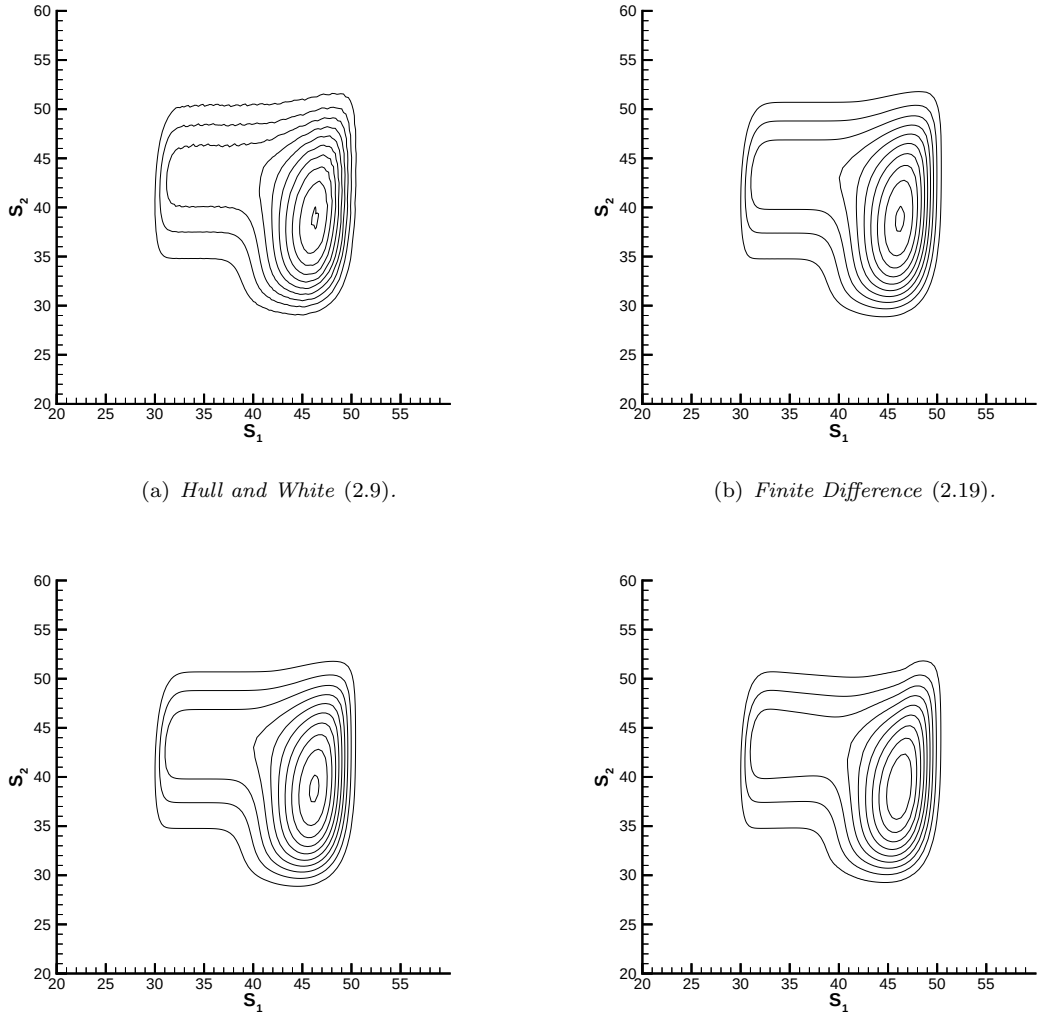
Table 3 gives the prices at $S_1 = S_2 = 40$ for a sequence of timesteps for all methods. The convergence behavior is somewhat erratic. This is quite typical for discretely observed barrier option pricing problems when standard lattice methods are used (Boyle and Lau, 1994).[6]

Figure 3 shows the contours of constant prices for the various lattice methods for this test problem. Compared with previous tests, the contour plots for the different methods are observably dissimilar. We can also see some slight ripples in the contours for the Hull and White method. At first glance, this is puzzling because the Hull and White method is a positive coefficient method. In fact, as we shall see later in Section 6, this is an artifact of the interpolation used in producing the plot since the Hull and White nodes are placed on a skewed, rotated grid.

[6]It is possible to modify lattice type approaches to improve convergence (Ritchken, 1995; Cheuk and Vorst, 1996), but this becomes very difficult when barriers are applied in more than one dimension.

N	H & W	Finite Difference	Finite Element	Gao
40	1.75998	1.71568	1.70926	1.70869
80	1.73728	1.78735	1.78443	1.68015
160	1.73993	1.79313	1.79206	1.77838
320	1.74433	1.71764	1.71698	1.73214

TABLE 3: *Values of European call options on the maximum of two assets when* $r = 0.05$, $\sigma_{S_1} = 0.10$, $\sigma_{S_2} = 0.30$, $T - t = 0.25$, $S_1 = S_2 = 40$, $K = 40$, $\rho = .7$. *Discretely observed knockout barriers at* $H_u = 50$, $H_l = 30$. *Barriers observed at increments of* .025. N *is the number of timesteps. H & W refers to method* (2.9), *Finite Difference refers to method* (2.19), *Finite Element refers to method* (2.26), *and Gao refers to method* (2.27).



(a) *Hull and White* (2.9).

(b) *Finite Difference* (2.19).

(c) *Finite Element* (2.26).

(d) *Gao* (2.27).

FIGURE 3: *Level curves of values of European call options on the maximum of two assets when* $r = 0.05$, $\sigma_{S_1} = 0.10$, $\sigma_{S_2} = 0.30$, $T - t = 0.25$, $K = 40$, $\rho = .7$. *Discretely observed knockout barriers at* $H_u = 50$, $H_l = 30$. *Barriers observed at increments of* .025.

At this point, it is worth summarizing our investigation of lattice methods. Essentially, these are explicit discretizations of the PDE (2.1). The simple form of equation (2.8) can be obtained if the log transformation eliminates the dependence of the equation coefficients on S_1, S_2. This permits a constant grid spacing in the transformed coordinates. In other words, the weights dictate the placement of the nodes. The node spacing is also of size $\sqrt{\Delta t}$ so that stability conditions are automatically satisfied.

The requirement that the weights in equation (2.8) be nonnegative is very restrictive. Nevertheless, nonnegativity is a desirable property since it guarantees that the numerical solution will be oscillation-free. However, our initial tests with a lattice method with negative weights did not result in observable oscillations.

Observe that the finite difference lattice and the finite element lattice use the same node locations and the same timestepping method. Convergence of these two methods is very similar, in spite of the fact that the finite difference lattice has negative coefficients.

In general, it is desirable to use a method where the nodes can be placed according to the characteristics of the payoff, or other features of the pricing problem (e.g. barriers), rather than being dictated by the equation coefficients. We would also like to have the flexibility of using unequally spaced nodes, or even unstructured grids. This flexibility would be very useful if we were using the PDE solver inside an optimization method for calibration of volatility/correlation surfaces. It would obviously be desirable, for example, if the grid is held fixed and the weights changed to reflect updated volatility estimates.

In order to further investigate the effect of negative coefficients on the discretization, an implicit finite volume method will be used in the following. We will use an implicit method because i) the stability condition for an explicit method can only be determinined numerically for general grids with negative coefficients; ii) we avoid any stability problems; and iii) we can easily use this method in cases where the coefficients are not constant.

3 Finite Volume Discretization

We give a brief description of the finite volume method (FVM) in the following. For more details, we refer the reader to Zvan et al. (2000). It is convenient to convert equation (2.1) into a forward equation

$$U_\tau = -\mathbf{V} \cdot \nabla U + (\mathbf{D}\nabla) \cdot \nabla U - rU \tag{3.1}$$

where $\tau = T - t$.

We now proceed to discretize equation (3.1) using a FVM. Consider a discrete two dimensional computational domain Ω which is tiled by triangles. Define a control volume or finite volume around each node, as shown in Figure 4. The finite volume is constructed by joining the midpoints of each triangle edge to the triangle centroid (or alternatively to the intersecion of the perpendicular bisector of each triange edge). Integrating equation (3.1) over the finite/control volume FV_i gives

$$\int_{FV_i} U_\tau d\Omega = -\int_{FV_i} \mathbf{V} \cdot \nabla U d\Omega + \int_{FV_i} (\mathbf{D}\nabla) \cdot \nabla U d\Omega - \int_{FV_i} rU d\Omega. \tag{3.2}$$

As in section 2.3, let N_i be the standard C^0 Lagrange basis functions defined on triangles, and let $U^{n+1} = \sum_j U_j^{n+1} N_j$ where $U_j^{n+1} = U(x_j, y_j, \tau_{n+1})$ is the value of U at (x_j, y_j, t_n).

Using fully implicit time stepping for ease of exposition, the following approximations are used

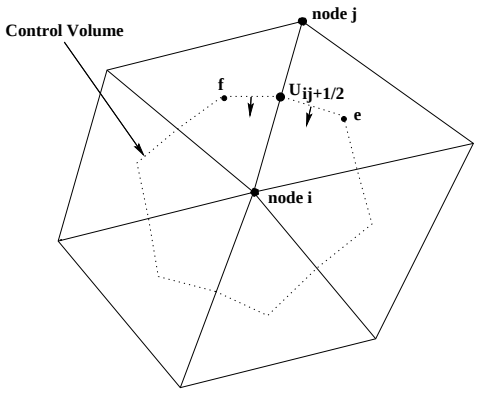


FIGURE 4: *Centroid control volume on a triangulated domain. Points e and f are the centroids of their respective triangles. The face (line segments from e to f) passes through the middle of the edge connecting nodes i and j.*

for the terms in equation (3.2):

$$\int_{FV_i} U_\tau d\Omega \approx A_i \left(\frac{U_i^{n+1} - U_i^n}{\Delta\tau} \right),$$

$$-\left(\int_{FV_i} \mathbf{V} \cdot \nabla U d\Omega \right)^{n+1} \approx -\mathbf{V}_i \cdot \oint_{\partial FV_i} U^{n+1} \vec{n} d\Gamma \approx \mathbf{V}_i \cdot \sum_{j \in \Omega_i} \vec{L}_{ij} U_{ij+\frac{1}{2}}^{n+1}, \tag{3.3}$$

$$\left(\int_{FV_i} (\mathbf{D}\nabla) \cdot \nabla U d\Omega \right)^{n+1} \approx \oint_{\partial FV_i} (\mathbf{D}_i \nabla U^{n+1}) \cdot \vec{n} d\Gamma \tag{3.4}$$

$$\approx \sum_{j \in \Omega_i} -\int_\Omega \nabla' N_i \mathbf{D}_i \nabla N_j d\Omega (U_j^{n+1} - U_i^{n+1}),$$

$$\left(\int_{FV_i} r U d\Omega \right)^{n+1} \approx A_i r_i U_i^{n+1}. \tag{3.5}$$

where the symbol definitions are provided in Table 4.

Note that in equations (3.3)-(3.5) the integrals have been approximated by evaluating terms which depend on the space-like variables at node i.

Putting together equations (3.2)-(3.5) gives the final form for the discretization of equation (3.1)

$$\begin{aligned} A_i \left(\frac{U_i^{n+1} - U_i^n}{\Delta\tau} \right) = \theta & \left(\sum_{j \in \Omega_i} \eta_{ij} (U_j^{n+1} - U_i^{n+1}) + \sum_{j \in \Omega_i} \vec{L}_{ij} \cdot \mathbf{V}_i U_{ij+\frac{1}{2}}^{n+1} - A_i r_i U_i^{n+1} \right) \\ & + (1-\theta) \left(\sum_{j \in \Omega_i} \eta_{ij} (U_j^n - U_i^n) + \sum_{j \in \Omega_i} \vec{L}_{ij} \cdot \mathbf{V}_i U_{ij+\frac{1}{2}}^n - A_i r_i U_i^n \right) \\ & + \theta w_i^{n+1} + (1-\theta) w_i^n, \end{aligned} \tag{3.6}$$

where the η_{ij} term in (3.6) is defined as

$$\eta_{ij} = -\int_\Omega \nabla' N_i \mathbf{D}_i \nabla N_j d\Omega. \tag{3.7}$$

Symbol	Definition
A_i	area of the control volume for node i
U_i^{n+1}	solution at node i at time step $n+1$
Δt	time step size
θ	temporal weighting factor (where $\theta = 1$ is a fully implicit scheme, $\theta = 1/2$ is the Crank-Nicolson method and $\theta = 0$ is a fully explicit scheme)
Ω_i	set of nodes that neighbor node i
$\vec{L}_{ij}$	$\int_e^f \hat{n} ds$ where e and f are the endpoints of the face separating nodes i and j (see Figure 4)
$\hat{n}$	inward pointing unit normal to the face separating nodes i and j
$U_{ij+\frac{1}{2}}^{n+1}$	value at the control volume face separating nodes i and j
r_i	interest rate at node i

TABLE 4: *Symbol definitions for finite volume discretization.*

The w_i^{n+1} term in equation (3.6) is used to handle boundary conditions (see Zvan et al., 2000). Note that when $\mathbf{V}$ and $\mathbf{D}$ are constant, (3.6) can be regarded as a Galerkin finite element method with mass lumping (Selmin and Formaggia, 1996; Zienkiewicz, 1977).

4 Diffusion Operators

This section explores the conditions under which discretizations of the diffusion term $(\mathbf{D}\nabla) \cdot \nabla U$ term in equation (3.1) are guaranteed to produce nonnegative coefficients, when using the FVM with linear shape functions. In other words, we desire $\eta_{ij} \geq 0$ in equation (3.6). For clarity, we will only examine the diffusion term, dropping time dependence and the convective term. In Zvan et al. (2000) it was shown that the convective term can be discretized using arbitrary temporal weighting such that the coefficients are positive. Note that for explicit and Crank-Nicolson schemes, a condition on the timestep size must also be satisfied in order to obtain nonnegative coefficients.

4.1 Constant Coefficients

We begin with the simple case of constant coefficients. Consider applying the FVM used in discretization (3.6) to

$$(\mathbf{D}\nabla) \cdot \nabla U = 0 \tag{4.1}$$

when $\mathbf{D}$ is constant and symmetric positive definite. For each U_i we obtain

$$\sum_{j \in \Omega_i} \eta_{ij}(U_j - U_i) = 0 \tag{4.2}$$

where

$$\eta_{ij} = -\int_\Omega \nabla' N_i \mathbf{D} \nabla N_j d\Omega. \tag{4.3}$$

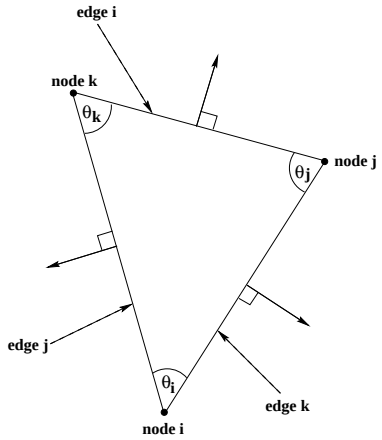


FIGURE 5: *Finite element/volume triangle.*

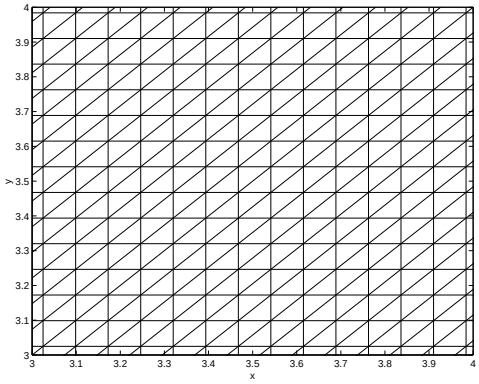


FIGURE 6: *A regular triangular mesh.*

Let Δ_1 and Δ_2 be two triangles which share the common edge k (see Figure 5). Let θ_{k_1} and θ_{k_2} be the angles opposite the shared edge. For the special case where $\mathbf{D}$ is an identity matrix, it is straightforward to show that $\eta_{ij} \geq 0$ if and only if $\theta_{k_1} + \theta_{k_2} \leq \pi$ (Forsyth, 1991).

In other words, the sum of the angles opposite each interior edge must be less than or equal to π. The condition $\theta_{k_1} + \theta_{k_2} \leq \pi$ is satisfied if the mesh is a Delaunay triangulation (Barth, 1994). A regular triangular mesh is a Delaunay triangulation. Note that we define a regular triangular mesh in this work to be a triangulation where the triangles have one edge parallel to one axis and another edge parallel to the other axis (see Figure 6). In general, for triangles with a boundary edge we require that the angle opposite the boundary edge be non-obtuse in order to ensure that the $\eta_{ij} \geq 0$ for boundary nodes. However, due to the nature of the usual boundary conditions in finance, (either a Dirichlet condition is specified, or the normal diffusion is zero), this does not add any additional constraints on the mesh in practice.

Now consider the more general case

$$\mathbf{D} = \begin{pmatrix} k_x & k_{xy} \\ k_{xy} & k_y \end{pmatrix} \tag{4.4}$$

where k_x, k_{xy} and k_y are constant, and $\mathbf{D}$ is positive definite. In this situation, there is a transformation to an (x', y') coordinate system such that $\mathbf{D}'$ is an identity matrix. A Delaunay triangulation

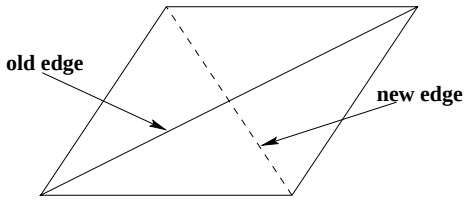


FIGURE 7: *Edge-swapping.*

in this coordinate system will produce $\eta_{ij} \geq 0$. Alternatively, we can bypass this coordinate transformation by performing an edge-swapping procedure in the original coordinates (Forsyth, 1991). This algorithm proceeds by examining each edge in turn. If $\eta_{ij} < 0$, then the mesh is altered by swapping the edge as shown in Figure 7. This affects adjacent edges, which must then be examined, and so on.

In summary, for a constant diffusion tensor (4.4), meshes where the η_{ij} for interior nodes are nonnegative can always be constructed in the original coordinate system, but such meshes will generally not be regular triangulations. Regular triangulations will be guaranteed to produce nonnegative η_{ij} only when $k_{xy} = 0$. Similarly, standard finite difference schemes will not produce positive coefficients for equation (4.1) unless $k_{xy} = 0$. Finite element methods using bilinear quadrilateral elements will in general require nonorthogonal meshes (unless $k_{xy} = 0$) to ensure that the η_{ij} are nonnegative. The use of bilinear elements will also generally require that an aspect ratio condition be satisfied in order to ensure nonnegative η_{ij}.

4.2 Nonconstant Coefficients

In option pricing models the diffusion coefficients are typically nonconstant. In this case, the discretization of equation (4.1) becomes

$$\sum_{j \in \Omega_i} \eta_{ij}(U_j - U_i) = 0,$$

where

$$\eta_{ij} = -\int_{\Omega} \nabla' N_i \mathbf{D}_i \nabla N_j d\Omega.$$

If transformations can be performed that produce constant diffusion coefficients, then, as pointed out in Section 4.1, a mesh where all the η_{ij} in discretization (3.6) are nonnegative for interior nodes can always be constructed for any node placement. However, if such transformations are not available, this is no longer true: one cannot ensure, in general, that $\eta_{ij} \geq 0$ for interior nodes for a given node placement when the diffusion tensor is nonconstant (see Zvan, 2000, for details).

This somewhat negative result is mitigated by the following fact, which is proven in Appendix A. Let h be the mesh size parameter (i.e. radius of largest circumcircle of any triangle in the mesh), and let

$$\begin{aligned} U_i^{\max} &= \max(U_i^n, U_{j \in \Omega_i}^{n+1}) \\ U_i^{\min} &= \min(U_i^n, U_{j \in \Omega_i}^{n+1}). \end{aligned} \tag{4.5}$$

If the exact solution satisfies a Lipschitz condition, then

$$U_i^{\min} - O(h) \leq U_i^{n+1} \leq U_i^{\max} + O(h).$$

In other words, an approximate discrete maximum principle holds. In the case of a discontinuous payoff, the solution does not satisfy a Lipschitz condition at the terminal time. However, due to the parabolic smoothing nature of the pricing PDE, we can expect that a Lipschitz condition will be satisfied at any finite time prior to maturity.

5 Comparison of Finite Volume and Lattice Methods

Based on the results of the comparison of the lattice methods in Section 2.5, it appears that the Hull and White method (Section 2.1) is the best technique, and so we will use it as a benchmark lattice technique. For the FVM, tests were conducted using Crank-Nicolson timestepping with the modification suggested in Rannacher (1984) for cases where the payoff was discontinuous (i.e. digital options) or for discretely observed barriers. A regular Cartesian grid was used, and, since a non-zero correlation was specified, the discretization of the diffusion term resulted in negative coefficients. As well, negative coefficients are caused by the use of Crank-Nicolson timestepping with timesteps larger than the explicit stability limit. We use the version of the PDE in the original (S_1, S_2) variables for these tests. We use the same sample problems as in Section 2.5.

In Table 5, we compare the convergence of the Hull and White lattice method and the FVM for the European call on the maximum of two assets. We also give the relative CPU times for each method. Since the lattice algorithm (2.9) is very simple to code, the lattice times should be regarded as close to optimal. The PDE algorithm was coded for maximum generality, (i.e. nonconstant coefficients) and should not be regarded as the most efficient implementation for these comparatively simple cases. It should also be recalled that the lattice methods return the value at a single point, while the PDE technique produces values for the entire range of (S_1, S_2). Also, as the tests were run on a Sun Ultra Sparc server with many simultaneous users, CPU times should be regarded as only accurate to within $\pm 5\%$. Table 5 shows that the FVM converges asymptotically at a second order rate, while the lattice method converges more slowly. Nevertheless, for low accuracy solutions, with constant coefficients and vanilla payoffs, the lattice method can be superior to the FVM in terms of CPU cost.

Table 6 shows the results for the digital call on two assets. Again, we see that the PDE method converges asymptotically at a second order rate, while the lattice method converges more slowly and somewhat erratically.

A similar picture emerges for the results for the discretely observed barrier option for a call on the maximum of two assets, which are provided in Table 7. The lattice technique converges slowly and erratically, compared to the FVM.

In summary, we can see that in all cases, even for problems with discontinuous payoffs and discretely observed barriers, we obtain quadratic convergence for the FVM. This occurs in spite of the fact that the discretization has negative coefficients.

In contrast, the Hull and White lattice method, which has positive coefficients, converges more slowly and, in the presence of discontinuities, erratically. We would expect that this lattice method would converge at a first order rate. However, since the nodes in the lattice method are not aligned with the payoff or the barrier locations, this reduces the rate of convergence (Wahlbin, 1980; Heston and Zhou, 2000). It would appear that it is beneficial to use a grid which is aligned with important features of the problem, even if this is at the expense of using a discretization which has negative

H & W		
N	Normalized CPU	Value
40	1.0	2.89428
80	7.75	2.89286
160	62	2.89172
320	560	2.89133
640	4462	2.89084
FVM		
Grid/N	Normalized CPU	Value
2025/25	3.4	2.87944
7921/50	31	2.88775
31329/100	282	2.88985
124609/200	2331	2.89037

TABLE 5: *Values of European call options on the maximum of two assets when* $r = 0.05$, $\sigma_{S_1} = 0.10$, $\sigma_{S_2} = 0.30$, $T - t = 0.25$, $S_1 = S_2 = 40$, $K = 40$, $\rho = .7$. *The exact solution is* 2.89055 *to six figures.* N *is the number of timesteps, grid is the number of nodes. H & W refers to the method* (2.9). *CPU times are given relative to the time for the H & W method for* 40 *timesteps.*

H & W		
N	Normalized CPU	Value
40	1.0	0.407732
80	7.8	0.410572
160	60	0.411267
320	555	0.411136
640	4413	0.410925
FVM		
Grid/N	Normalized CPU	Value
2025/25	3.3	0.410701
7921/50	31	0.410862
31329/100	276	0.410912
124609/200	2349	0.410925

TABLE 6: *Values of European digital call when* $r = 0.05$, $\sigma_{S_1} = 0.10$, $\sigma_{S_2} = 0.30$, $T - t = 0.25$, $S_1 = S_2 = 40$, $K = 40$, $\rho = .7$. *The exact solution is* 0.410929 *to six figures.* N *is the number of timesteps, grid is the number of nodes. H & W refers to the method* (2.9). *CPU times are given relative to the time for the H & W method for* 40 *timesteps.*

H & W		
N	Normalized CPU	Value
40	1.0	1.75998
80	7.4	1.73728
160	56.6	1.73993
320	501	1.74433
640	3788	1.74477
FVM		
Grid/N	Normalized CPU	Value
2304/20	10.6	1.73254
9025/40	97.4	1.74106
35721/80	556	1.74318
143641/160	3338	1.74369

TABLE 7: *Values of European call options on the maximum of two assets when* $r = 0.05$, $\sigma_{S_1} = 0.10$, $\sigma_{S_2} = 0.30$, $T - t = 0.25$, $S_1 = S_2 = 40$, $K = 40$, $\rho = .7$. *Discretely observed knockout barriers at* $H_u = 50$, $H_l = 30$. *Barriers observed at increments of* .025. N *is the number of timesteps, grid is the number of nodes. H & W refers to the method* (2.9). *CPU times are given relative to the time for the H & W method for* 40 *timesteps.*

coefficients.

6 Qualitative Results for the Finite Volume Method

In this section, we will explore the qualitative effect of negative coefficients on the behavior of the solution. We have already seen (in Section 5) that we observe good convergence behavior (at single points) with a FVM discretization which has negative coefficients.

In order to ensure that any negative coefficients were only caused by the discretization of the diffusion term, a flux limiting scheme (Zvan et al., 1998, 2000) was used for convection and the discretization was fully implicit. We emphasize that, in practice, one would be ill-advised in general to use a fully implicit method because this reduces convergence to a first order rate. However, our objective in this section is to examine the effect of the negative coefficients due to the discrete diffusion operator. By using a fully implicit method and a flux limiter, we eliminate all other causes of negative coefficients. We use the log transformed version of the PDE (equation (2.3)) so that it is possible to construct a positive coefficient method for a given node placement via edge-swapping (see Figure 8). Note that the nodes are at the same positions for both the regular mesh and the edge-swapped mesh.

As our focus here is on qualitative aspects, we will simply present contour plots of values and deltas ($\partial U/\partial S_1$ and $\partial U/\partial S_2$, calculated using the method described in Zienkiewicz and Wu (1994)). These plots seem to suggest that the positive coefficient edge-swapped mesh produced solutions of poor quality. The level curves of the values and deltas are relatively smooth when the regular mesh is used (Figure 9), but they are *jagged* when the edge-swapped mesh is used (Figure 10). The jaggedness in the contour levels persists even when the edge-swapped mesh is refined (Figure 11). It should be noted that the jaggedness is not caused by convection—the effect occurs even if the convection term is removed entirely from equation (3.1). Nor is the jaggedness in the level curves of the deltas an artifact of the method used to compute the deltas, since it is clear that it is already

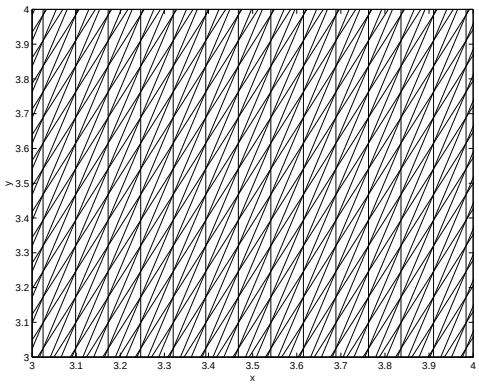
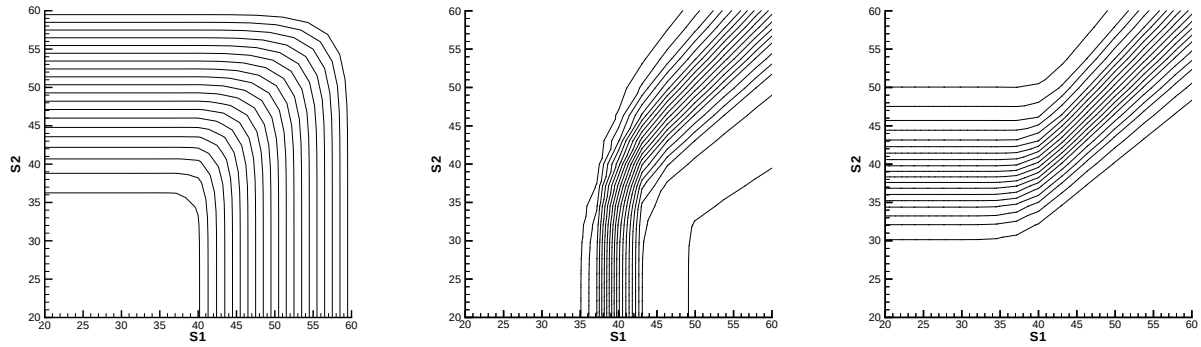
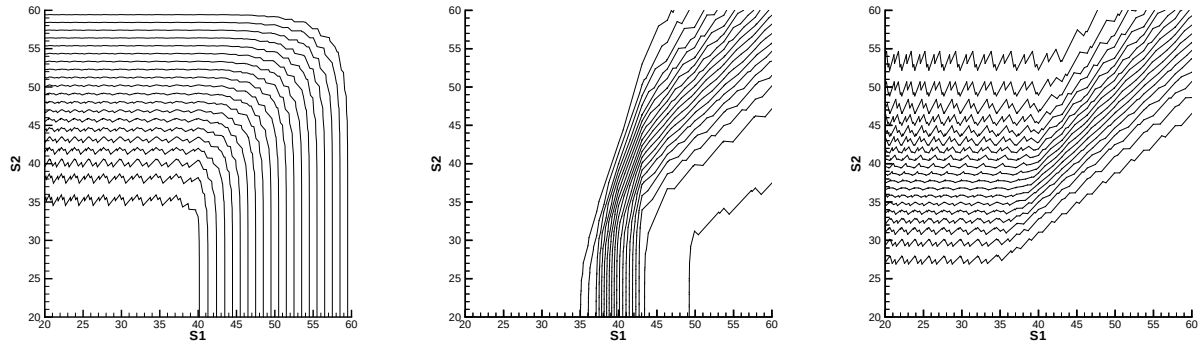


FIGURE 8: *A positive coefficient mesh constructed using edge-swapping when* $\sigma_{S_1} = 0.10$, $\sigma_{S_2} = 0.30$ *and* $\rho = 0.70$.



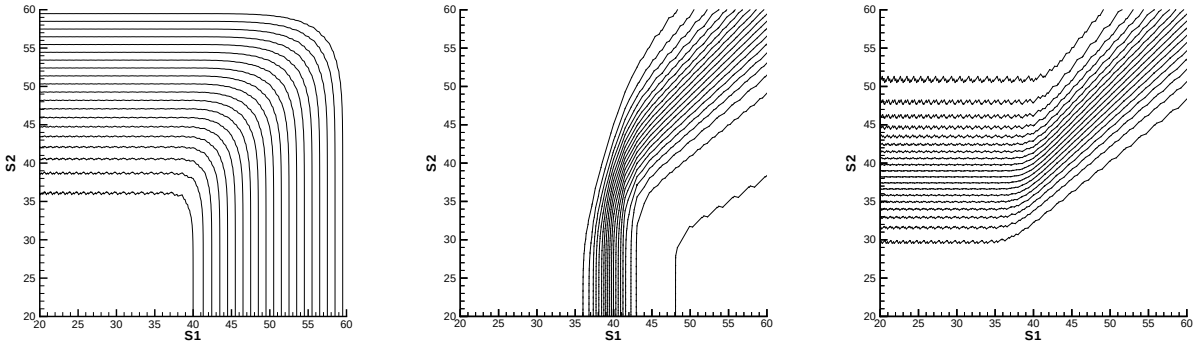
(a) *Value* U. (b) *Delta* $\partial U/\partial S_1$. (c) *Delta* $\partial U/\partial S_2$.

FIGURE 9: *Level curves of values and deltas of a European call option on the maximum of two assets when* $r = 0.05$, $\sigma_{S_1} = 0.10$, $\sigma_{S_2} = 0.30$, $\rho = 0.70$, $T - t = 0.25$ *and* $K = 40$. *The solutions were computed with 50 timesteps and a regular mesh with 6724 nodes.*



(a) *Value* U. (b) *Delta* $\partial U/\partial S_1$. (c) *Delta* $\partial U/\partial S_2$.

FIGURE 10: *Level curves of values and deltas of a European call option on the maximum of two assets when* $r = 0.05$, $\sigma_{S_1} = 0.10$, $\sigma_{S_2} = 0.30$, $\rho = 0.70$, $T - t = 0.25$ *and* $K = 40$. *The solutions were computed with 50 timesteps and a positive coefficient edge-swapped mesh with 6724 nodes.*



(a) *Value* U. (b) *Delta* $\partial U/\partial S_1$. (c) *Delta* $\partial U/\partial S_2$.

FIGURE 11: *Level curves of values and deltas of a European call option on the maximum of two assets when* $r = 0.05$, $\sigma_{S_1} = 0.10$, $\sigma_{S_2} = 0.30$, $\rho = 0.70$, $T - t = 0.25$ *and* $K = 40$. *The solutions were computed with 100 timesteps and a positive coefficient edge-swapped mesh with 26569 nodes.*

present in the level curves of the values.

A possible cause of the jaggedness is that the kinks in the payoff function occur within elements for the edge-swapped meshes and not, as is the case for the regular meshes used in the numerical examples, only at element edges. Accuracy may deteriorate when the second derivatives of the initial (terminal) condition do not exist within elements (Wahlbin, 1980). To test this hypothesis the initial condition was smoothed by projecting it onto the space spanned by the basis functions using

$$\left(\int_x \int_y \mathbf{N}\mathbf{N}' dydx\right) \mathbf{U}^0 = \int_x \int_y g(x, y, T)\mathbf{N} dydx, \tag{6.1}$$

where $\mathbf{N}$ are the basis functions, $\mathbf{U}^0$ are the smoothed initial data, and $g(\cdot)$ is the payoff function. As pointed out by Wahlbin (1980), this should restore optimal convergence rates. However, we found that smoothing the initial condition had little effect on the quality of the contour plots.

The jaggedness would seem to contradict the fact that the discretization with the edge-swapped mesh is a positive coefficient scheme. Although discrete local maximum and minimum principles hold for the discretization when positive coefficient meshes are used, it is interesting to note that nodes are not connected to their nearest spatial neighbors in an edge-swapped mesh. Consequently, the value at a node may not be bounded by the values at its spatially nearest neighboring nodes.

Figure 12 provides a close up view of the level curves of values for a European call option on the maximum of two assets when the solution was computed using an edge-swapped mesh. In this figure, the values at the nodes are indicated. Inspection of these values demonstrates that, although the level curves seem to indicate an oscillatory solution, the computed values at the nodes are actually well-behaved. Hence, the jaggedness is an artifact of calculating the level curves. More specifically, linear interpolation on the edge-swapped meshes produced poor results.[7]

The poor quality of level curves computed on edge-swapped meshes is most likely due to the fact that the elements are stretched (i.e. long and thin). Although stretched elements are optimal for interpolation with respect to minimizing the error bound when there is a direction with dominant

[7]Recall that jagged contours were also observed in one case for the Hull and White lattice method, even though the Hull and White lattice method is a positive coefficient scheme.

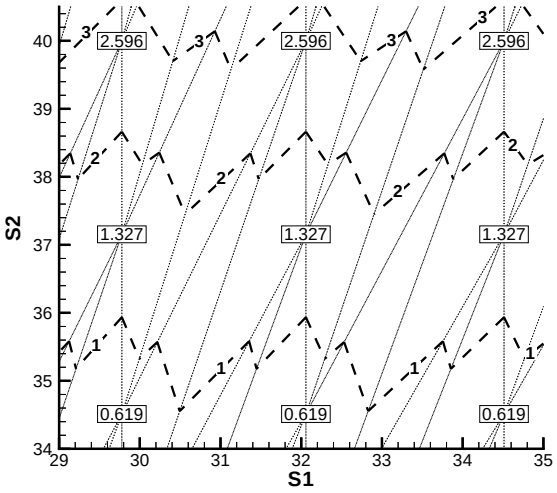


FIGURE 12: *Level curves of values and values at nodes for a European call option on the maximum of two assets when* $r = 0.05$, $\sigma_{S_1} = 0.10$, $\sigma_{S_2} = 0.30$, $\rho = 0.70$, $T - t^* = 0.25$ *and* $K = 40$. *The solutions were computed with 50 timesteps using a positive coefficient edge-swapped mesh with 6724 nodes.*

curvature, the placement of the elements must be data dependent (see D'Azevedo and Simpson, 1989; Rippa, 1992). In particular, the triangles should be short in directions where the curvature of U is high, and long in directions where it is low. Edge swapping to ensure positive coefficients ignores the curvature of U, so that elements may be placed with their long side in the direction of high curvature. We note that poor interpolation results may be of concern when solving a model if jump conditions are present. For example, in the presence of discrete dividends, interpolation will generally be performed at ex-dividend dates.

An alternative to edge-swapping is to place nodes in such a manner as to ensure positive coefficients. Recall that a regular mesh will ensure positive coefficients when the coordinate system has been rotated to eliminate the cross-partial terms. Rotating such a mesh back into the original coordinate system will ensure that the coefficients are positive without generating stretched elements (see Figure 13). To remove the cross-partial terms from (2.3), the coordinate system must be rotated by an angle of $(1/2)\tan^{-1}\left[(2\rho\sigma_{S_1}\sigma_{S_2})/(\sigma_{S_1}^2 - \sigma_{S_2}^2)\right]$.

Figure 14 contains contour plots of values and deltas of a quarter year European call on the maximum of two assets computed using a positive coefficient rotated mesh. The node positions in the rotated mesh differ from those in the regular mesh, but the spacings are identical. The contours calculated using the rotated mesh are better than the contours computed using the edge-swapped mesh, but they are not as smooth as the contours calculated using regular meshes.

The contour plots computed using the rotated mesh suggest an oscillatory solution. Using projection (6.1) to smooth the initial condition again had little effect on the quality of the contour plots. As with the edge-swapped meshes, the computed solutions at the nodes were well behaved (see Figure 15). Hence, the jagged contours were once again a result of poor interpolation.

The rotated mesh may have produced poor interpolation results because the element edges in it were not aligned with the curvature of U. Like edge-swapping, ensuring positive coefficients as a criterion for node placement ignores other aspects of the problem.

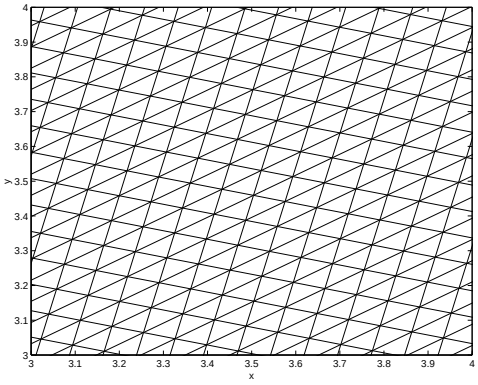
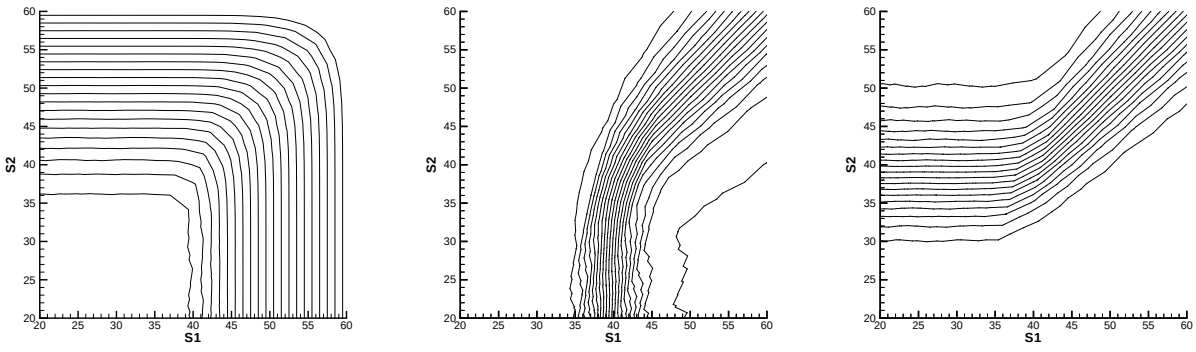


FIGURE 13: *A positive coefficient mesh which corresponds to a regular mesh when the coordinate system is rotated by an angle of* $(1/2)\tan^{-1}\left[(2\rho\sigma_{S_1}\sigma_{S_2})/(\sigma_{S_1}^2-\sigma_{S_2}^2)\right]$ *when* $\sigma_{S_1}=0.10$, $\sigma_{S_2}=0.30$, *and* $\rho=0.70$.



(a) *Value* U. (b) *Delta* $\partial U/\partial S_1$. (c) *Delta* $\partial U/\partial S_2$.

FIGURE 14: *Level curves of values and deltas of a European call option on the maximum of two assets when* $r=0.05$, $\sigma_{S_1}=0.10$, $\sigma_{S_2}=0.30$, $\rho=0.70$, $T-t=0.25$ *and* $K=40$. *The solutions were computed with 50 timesteps using a positive coefficient rotated mesh with 6614 nodes.*

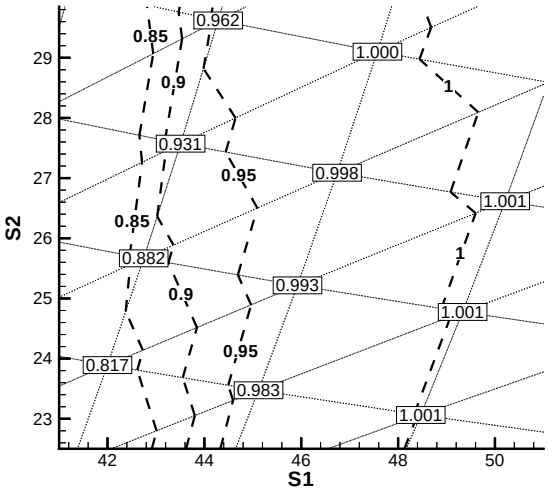


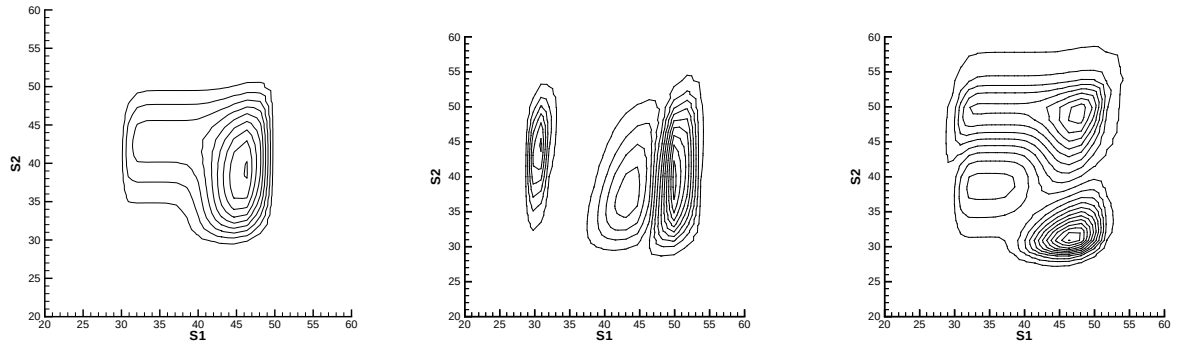
FIGURE 15: *Level curves of deltas with respect to S_1 and deltas at nodes for a European call option on the maximum of two assets when $r = 0.05$, $\sigma_{S_1} = 0.10$, $\sigma_{S_2} = 0.30$, $\rho = 0.70$, $T - t = 0.25$ and $K = 40$. The solutions were computed with 50 timesteps using a positive coefficient rotated mesh with 6614 nodes.*

As a final example, we consider our earlier test problem of a quarter year European discrete double barrier call option on the maximum of two assets. Contour plots of solutions computed using regular, edge-swapped and rotated meshes are contained in Figures 16, 17 and 18, respectively. Again we see that the regular mesh produced the smoothest contours, while the edge-swapped mesh produced the worst contours. Note that when using a rotated mesh, one cannot generally place nodes such that they line up with barriers. This again highlights the fact that if ensuring positive coefficients is used as a criterion for constructing the mesh, other aspects of the problem such as the curvature of U or constraints on the solution may have to be ignored.

6.1 Significance of the Results

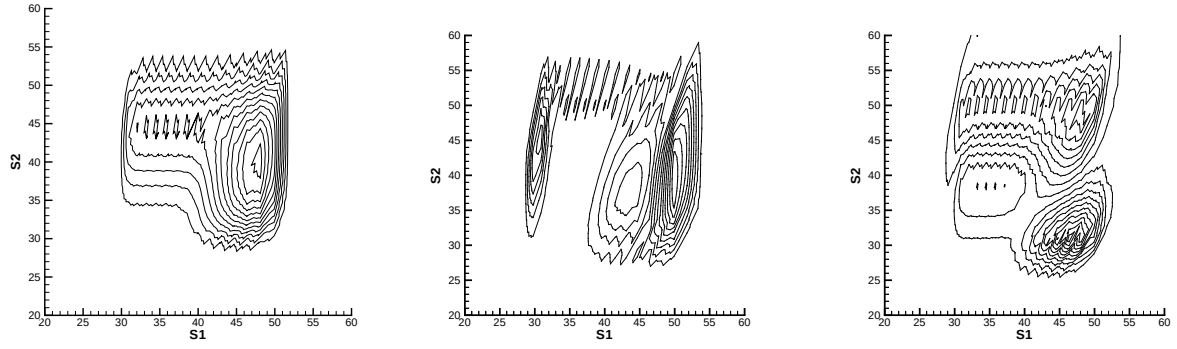
Standard lattice methods essentially require that the coefficients in the discrete equations take on positive values *a priori*. From a finite element/volume context, this amounts to forcing certain node locations. Such an approach is exactly the opposite of that typically used in finite element discretizations, where the mesh is constructed first and the discrete coefficients are a consequence of the node locations. In fact, for a constant diffusion tensor, a positive coefficient discretization can be obtained (via edge-swapping) for any placement of nodes.

Using nonnegative coefficients as a criterion for node placement ignores other aspects of the pricing problem. In the numerical examples, we have observed that forcing the positive coefficient condition does not appear to result in solutions of better quality. For example, the finite difference and finite element lattice schemes in Section 2.5 give comparable results, in spite of the fact that the finite difference lattice has negative coefficients, whereas the finite element lattice is a positive coefficient scheme. In fact, if we are concerned with interpolation of the price and hedging parameters, the positive mesh discretizations seem to be of poor quality. The results suggest that it is better to place nodes in such a way as to capture rapid changes in the solution (i.e. based on the shape of the payoff function) rather than to force the positive coefficient condition, which ignores



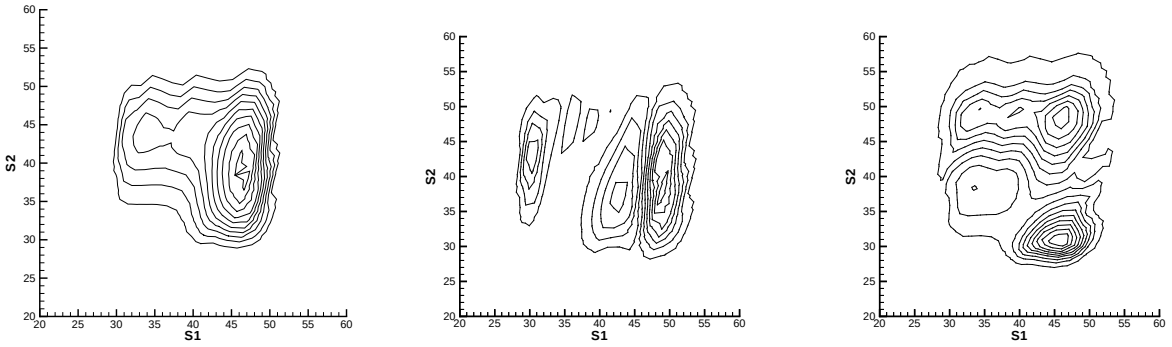
(a) *Value* U. (b) *Delta* $\partial U/\partial S_1$. (c) *Delta* $\partial U/\partial S_2$.

FIGURE 16: *Level curves of values and deltas of a European discrete double barrier call option on the maximum of two assets when* $r = 0.05$, $\sigma_{S_1} = 0.10$, $\sigma_{S_2} = 0.30$, $\rho = 0.70$, $T - t = 0.25$ *and* $K = 40$. *The barrier is applied in time increments of 0.025, and* $H_l = 30$ *and* $H_u = 50$. *The solutions were computed with 100 timesteps using a regular mesh with 26569 nodes.*



(a) *Value* U. (b) *Delta* $\partial U/\partial S_1$. (c) *Delta* $\partial U/\partial S_2$.

FIGURE 17: *Level curves of values and deltas of a European discrete double barrier call option on the maximum of two assets when* $r = 0.05$, $\sigma_{S_1} = 0.10$, $\sigma_{S_2} = 0.30$, $\rho = 0.70$, $T - t = 0.25$ *and* $K = 40$. *The barrier is applied in time increments of 0.025, and* $H_l = 30$ *and* $H_u = 50$. *The solutions were computed with 100 timesteps using a positive coefficient edge-swapped mesh with 26569 nodes.*



(a) *Value* U. (b) *Delta* $\partial U/\partial S_1$. (c) *Delta* $\partial U/\partial S_2$.

FIGURE 18: *Level curves of values and deltas of a European discrete double barrier call option on the maximum of two assets when* $r = 0.05$, $\sigma_{S_1} = 0.10$, $\sigma_{S_2} = 0.30$, $\rho = 0.70$, $T - t = 0.25$ *and* $K = 40$. *The barrier is applied in time increments of 0.025, and* $H_l = 30$ *and* $H_u = 50$. *The solutions were computed with 100 timesteps using a positive using a positive coefficient rotated mesh with 26455 nodes.*

the payoff condition.

Fortunately, it does not appear necessary in practice to ensure that discretizations of the diffusion term result in positive coefficients. This is because for a given set of nodes it is not generally possible to ensure that finite volume discretizations will produce nonnegative coefficients when a nonconstant diffusion tensor cannot be transformed into a constant tensor. The results suggest that finite volume methods can be used to obtain high quality solutions, without imposing additional constraints on the mesh.

7 Conclusions

For any given set of nodes, a mesh can be constructed through edge-swapping that will ensure that all coefficients in a finite element/volume discretization are nonnegative in the presence of correlation when the diffusion tensor is constant. However, an edge-swapped mesh will often contain stretched or skewed elements. In order to avoid such elements, or when using standard finite differences, one needs to rotate the coordinate system, which may not be possible in some contexts.

For the pricing problems considered in this work, it was demonstrated that meshes which ensured nonnegative coefficients produced poor interpolation results compared to regular meshes which did not ensure nonnegative coefficients. This may be of concern when certain jump conditions are present, for example, when there are discrete dividends. The poor interpolation results appear to be due to the fact that ensuring positive coefficients ignores the curvature of the solution, as well as other aspects of the pricing problem.

It was also shown that when meshes that allow negative coefficients are used, discretizations can approximately satisfy discrete maximum and minimum principles as the mesh size parameter approaches zero. Consequently, it would appear that meshes where the edges are aligned with features of the payoff function (which may not result in positive coefficients) will produce solutions of high quality. Fortunately, it does not appear necessary to enforce the positive coefficient condition for option pricing problems, because it is generally not possible to produce a positive coefficient discretization for a given set of nodes when the diffusion tensor is nonconstant.

This paper has also shown that a variety of lattice schemes can be constructed using explicit

finite difference/element schemes. In some cases, these methods are equivalent (to $O\left[(\Delta t)^2\right]$) to known lattice methods. Explicit schemes can be constructed which have negative coefficients, but are nonetheless stable. Generally, positive coefficient methods are constructed assuming that a transformation exists which makes the diffusion coefficients constant.

It is worth remarking that these results highlight the fact that the pervasive focus on equivalent martingale probabilities in much of the finance literature may not be very useful when constructing numerical schemes for option pricing problems in more than one dimension. A better approach is to consider the separate effects of convection and diffusion and to use discretizations which handle each of these effects appropriately.

A Appendix - Approximate Discrete Local Maximum and Minimum Principles

It cannot be shown that discrete local maximum and minimum principles hold for a scheme when any of the η_{ij} are negative. That is, it cannot be proved that $\min_{j\in\Omega_i} U_j \le U_i \le \max_{j\in\Omega_i} U_j$. However, we will now show that if U satisfies a Lipschitz condition, then $\min_{j\in\Omega_i} U_j + O(h) \le U_i \le \max_{j\in\Omega_i} U_j + O(h)$, (where h denotes the mesh spacing) for a finite volume discretization of equation (4.1) when

$$\mathbf{D} = \begin{pmatrix} k_x(x,y) & k_{xy}(x,y) \\ k_{xy}(x,y) & k_y(x,y) \end{pmatrix}.$$

Assume that the computational domain is bounded. After discretizing (4.1), we have

$$\left(\sum_{\Delta_m\in\Delta_{i_+}} -\eta_{ii}^m + \sum_{\Delta_k\in\Delta_{i_-}} -\eta_{ii}^k\right) U_i(x_i,y_i) = \sum_{\Delta_m\in\Delta_{i_+}}\sum_{j\in\Omega_i} \eta_{ij}^m U_j(x_j,y_j) + \sum_{\Delta_k\in\Delta_{i_-}}\sum_{j\in\Omega_i} \eta_{ij}^k U_j(x_j,y_j), \tag{A.1}$$

where (x_i,y_i) are the coordinates at node i and η_{ii}^m is the value of η_{ii} over triangle Δ_m. In (A.1), $\Delta_{i_+}\cup\Delta_{i_-}$ is the set of triangles that have node i as a vertex, where Δ_{i_+} is the set of triangles where each η_{ij} is nonnegative and Δ_{i_-} is the set of triangles where an η_{ij} is negative. Let $j_{+_k}\in\Omega_i$ be the set of nodes where $\eta_{ij}\ge 0$, and $j_{-_k}\in\Omega_i$ be the set of nodes where $\eta_{ij}<0$. Also, let $x_{j_{-_k}} = x_{j_{+_k}} + \alpha_{j_{-_k}} h$ and $y_{j_{-_k}} = y_{j_{+_k}} + \beta_{j_{-_k}} h$, where $\alpha_{j_{-_k}}$ and $\beta_{j_{-_k}}$ are arbitrary constants. Then equation (A.1) can be rewritten as

$$\begin{aligned}\left(\sum_{\Delta_m\in\Delta_{i_+}} -\eta_{ii}^m + \sum_{\Delta_k\in\Delta_{i_-}} -\eta_{ii}^k\right) U_i(x_i,y_i) = {} & \sum_{\Delta_m\in\Delta_{i_+}}\sum_{j\in\Omega_i} \eta_{ij}^m U_j(x_j,y_j) \\ & + \sum_{\Delta_k\in\Delta_{i_-}} [\eta_{ij_{+_k}}^k U_{j_{+_k}}(x_{j_{+_k}},y_{j_{+_k}}) \\ & + \eta_{ij_{-_k}}^k U_{j_{-_k}}(x_{j_{+_k}}+\alpha_{j_{-_k}}h, y_{j_{+_k}}+\beta_{j_{-_k}}h)].\end{aligned} \tag{A.2}$$

If U is Lipschitz continuous, equation (A.2) becomes

$$\left(\sum_{\Delta_m\in\Delta_{i_+}} -\eta_{ii}^m + \sum_{\Delta_k\in\Delta_{i_-}} -\eta_{ii}^k\right) U_i(x_i,y_i) = \sum_{\Delta_m\in\Delta_{i_+}}\sum_{j\in\Omega_i} \eta_{ij}^m U_j(x_j,y_j) + \sum_{\Delta_k\in\Delta_{i_-}} [\eta_{ij_{+_k}}^k U_{j_{+_k}}(x_{j_{+_k}},y_{j_{+_k}}) + \eta_{ij_{-_k}}^k (U_{j_{+_k}}(x_{j_{+_k}},y_{j_{+_k}}) + O(h))].$$

After noting that $-\eta_{ii}^k = \eta_{ij_{+_k}}^k + \eta_{ij_{-_k}}^k$, this can be simplified to

$$\left(\sum_{\Delta_m\in\Delta_{i_+}} -\eta_{ii}^m + \sum_{\Delta_k\in\Delta_{i_-}} -\eta_{ii}^k\right) U_i = \sum_{\Delta_m\in\Delta_{i_+}}\sum_{j\in\Omega_i} \eta_{ij}^m U_j + \sum_{\Delta_k\in\Delta_{i_-}} -\eta_{ii}^k U_{j_{+_k}} + \sum_{\Delta_k\in\Delta_{i_-}} \eta_{ij_{-_k}}^k O(h). \tag{A.3}$$

Note that $\eta_{ij_{-_k}}^k = \vec{n}_i^k \cdot \mathbf{D}_i \vec{n}_{j_{-_k}}^k / (2\sin\theta_{j_{+_k}}^k)$, where $\vec{n}_i^k$ is the outward pointing unit normal to edge i for Δ_k (see Figure 5) and $\theta_{j_{+_k}}^k$ is $\theta_{j_{+_k}}$ for Δ_k. Since the computational domain is bounded, $\sum_{\Delta_k\in\Delta_{i_-}} \eta_{ij_{-_k}}^k$ is a finite quantity as long as the triangles are nondegenerate. Equation (A.3) can be written as

$$\left(\sum_{\Delta_m\in\Delta_{i_+}} -\eta_{ii}^m + \sum_{\Delta_k\in\Delta_{i_-}} -\eta_{ii}^k\right) U_i = \sum_{\Delta_m\in\Delta_{i_+}}\sum_{j\in\Omega_i} \eta_{ij}^m U_j + \sum_{\Delta_k\in\Delta_{i_-}} -\eta_{ii}^k U_{j_{+_k}} + O(h). \tag{A.4}$$

Since it can be shown that $-\eta_{ii} > 0$, and recalling that $\eta_{ij}^m > 0$ ($\Delta_m \in \Delta_{i_+}$), we can deduce an approximate maximum principle. By defining $U_i^{\max} = \max_{j\in\Omega_i} U_j$, we can write equation (A.4) as

$$\left(\sum_{\Delta_m\in\Delta_{i_+}} -\eta_{ii}^m + \sum_{\Delta_k\in\Delta_{i_-}} -\eta_{ii}^k\right) U_i \le \left(\sum_{\Delta_m\in\Delta_{i_+}} -\eta_{ii}^m + \sum_{\Delta_k\in\Delta_{i_-}} -\eta_{ii}^k\right) U_i^{\max} + O(h). \tag{A.5}$$

This implies $U_i \le U_i^{\max} + O(h)$ (recall $-\eta_{ii} > 0$). Similarly, if we define $U_i^{\min} = \min_{j\in\Omega_i} U_j$, then $U_i \ge U_i^{\min} + O(h)$. Hence, $\min_{j\in\Omega_i} U_j + O(h) \le U_i \le \max_{j\in\Omega_i} U_j + O(h)$, so that discrete local maximum and minimum principles are approximately satisfied as $h \to 0$. Similar bounds were derived in Brandt (1973) for finite difference operators.

The above result can be extended to incorporate time dependence with arbitrary temporal weighting. For example, if we discretize the equation

$$U_\tau = (\mathbf{D}\nabla)\cdot\nabla U \tag{A.6}$$

using a fully implicit scheme, then equation (A.5) becomes

$$\left(1+\frac{\Delta t}{A_i}\left(\sum_{\Delta_m\in\Delta_{i_+}}-\eta_{ii}^m+\sum_{\Delta_k\in\Delta_{i_-}}-\eta_{ii}^k\right)\right)U_i^{n+1} \leq \left(1+\frac{\Delta t}{A_i}\left(\sum_{\Delta_m\in\Delta_{i_+}}-\eta_{ii}^m+\sum_{\Delta_k\in\Delta_{i_-}}-\eta_{ii}^k\right)\right)U_i^{\max}+O(h)\frac{\Delta\tau}{A_i}, \quad \text{(A.7)}$$

where A_i is the area of the control volume and $U_i^{\max}=\max(U_i^n, U_{j\in\Omega_i}^{n+1})$. After noting that $A_i = O(h^2)$, equation (A.7) can be written as

$$U_i^{n+1} \leq U_i^{\max}+\frac{O(h)}{(h^2/\Delta\tau)+C}, \quad \text{(A.8)}$$

where C is some positive constant. Since $1/\left[(h^2/\Delta\tau)+C\right] \leq 1/C$, equation (A.8) becomes

$$U_i^{n+1} \leq U_i^{\max}+O(h).$$

References

Amin, K. I. (1991). On the computation of continuous time option prices using discrete approximations. *Journal of Financial and Quantitative Analysis* 26, 477–495.

Barth, T. J. (1994). Aspects of unstructured grids and finite-volume solvers for the Euler and Navier-Stokes equations. In *Lecture Series 1994-05: Computational Fluid Dynamics*. von Karman Institute for Fluid Dynamics.

Boyle, P. P. (1988). A lattice framework for option pricing with two state variables. *Journal of Financial and Quantitative Analysis* 13, 1–12.

Boyle, P. P. and S. H. Lau (1994). Bumping up against the barrier with the binomial method. *Journal of Derivatives 1*(Summer), 6–14.

Boyle, P. P. and Y. Tian (1998). An explicit finite difference approach to the pricing of barrier options. *Applied Mathematical Finance* 5, 17–43.

Brandt, A. (1973). Generalized local maximum principles for finite-difference operators. *Mathematics of Computation* 27, 685–718.

Brennan, M. J. and E. S. Schwartz (1978). Finite difference methods and jump processes arising in the pricing of contingent claims: A synthesis. *Journal of Financial and Quantitative Analysis* 13, 461–474.

Brennan, M. J. and E. S. Schwartz (1980). Analyzing convertible bonds. *Journal of Financial and Quantitative Analysis* 15, 907–929.

Cheuk, T. H. F. and T. C. F. Vorst (1996). Complex barrier options. *Journal of Derivatives 4*(Fall), 8–22.

D'Azevedo, E. F. and R. B. Simpson (1989). On optimal interpolation triangle incidences. *SIAM Journal on Scientific and Statistical Computing* 10, 1063–1075.

Dempster, M. A. H. and J. P. Hutton (1997). Fast numerical valuation of American, exotic and complex options. *Applied Mathematical Finance* 4, 1–20.

Forsyth, P. A. (1991). A control volume finite element approach to NAPL groundwater contamination. *SIAM Journal on Scientific and Statistical Computing* 12, 1029–1057.

Forsythe, G. E. and W. R. Wasow (1967). *Finite-Difference Methods for Partial Differential Equations*. John Wiley & Sons, Inc., New York.

Gao, B. (1997). Convergence rate of option prices from discrete- to continuous-time. Working paper, University of North Carolina at Chapel Hill.

Heston, S. (1993). A closed-form solution for options with stochastic volatility with applications to bond and currency options. *The Review of Financial Studies* 6, 327–343.

Heston, S. and G. Zhou (2000). On the rate of convergence of discrete-time contingent claims. *Mathematical Finance* 10, 53–75.

Hull, J. (2000). *Options, Futures, and Other Derivatives* (Fourth ed.). Prentice Hall, Upper Saddle River, New Jersey.

Hull, J. and A. White (1990). Valuing derivative securities using the explicit finite difference method. *Journal of Financial and Quantitative Analysis* 25, 87–100.

Hull, J. and A. White (1994). Numerical procedures for implementing term structure models II: Two-factor models. *Journal of Derivatives 2*(Winter), 37–48.

Imai, J. (1997). An alternative lattice approach for multidimensional geometric Brownian motions. Working paper, Tokyo Institute of Technology.

Putti, M. and C. Cordes (1998). Finite element approximation of the diffusion operator on tetrahedra. *SIAM Journal on Scientific Computing* 19, 1154–1168.

Rafferty, C. S., M. R. Pinto, and R. W. Dutton (1985). Iterative methods in semiconductor device simulation. *IEEE Transactions on Computer Aided Design* 4, 462–471.

Rannacher, R. (1984). Finite element solution of diffusion problems with irregular data. *Numerische Mathematik* 43, 309–327.

Rippa, S. (1992). Long and thin triangles can be good for linear interpolation. *SIAM Journal on Numerical Analysis* 29, 257–270.

Ritchken, P. (1995). On pricing barrier options. *Journal of Derivatives 3*(Winter), 19–28.

Selmin, V. and L. Formaggia (1996). Unified construction of finite element and finite volume discretizations for compressible flows. *International Journal for Numerical Methods in Engineering* 39, 1–32.

Tavella, D. and C. Randall (2000). *Pricing Financial Instruments: The Finite Difference Approach*. Wiley, New York.

Wahlbin, L. B. (1980). A remark on parabolic smoothing and the finite element method. *Siam Journal on Numerical Analysis* 17, 33–38.

Wilmott, P., J. Dewynne, and S. Howison (1993). *Option Pricing: Mathematical Models and Computation*. Oxford Financial Press, Oxford.

Windcliff, H., P. A. Forsyth, and K. R. Vetzal (1999). Shout options: A framework for pricing contracts which can be modified by the investor. *Journal of Computational and Applied Mathematics*, forthcoming.

Zienkiewicz, O. C. (1977). *The Finite Element Method* (Third ed.). McGraw-Hill Limited, Toronto.

Zienkiewicz, O. C. and J. Wu (1994). Automatic directional refinement in adaptive analysis of compressible flows. *International Journal of Numerical Methods in Engineering* 37, 2189–2210.

Zvan, R. (2000). *The Numerical Solution of Two-Factor PDE Option Pricing Models*. Ph. D. thesis, Department of Computer Science, University of Waterloo.

Zvan, R., P. A. Forsyth, and K. R. Vetzal (1998). Robust numerical methods for PDE models of Asian options. *Journal of Computational Finance 1*(2), 39–78.

Zvan, R., P. A. Forsyth, and K. R. Vetzal (2000). A finite volume approach for contingent claims valuation. *IMA Journal of Numerical Analysis*, forthcoming.