Computing Two-Factor Deltas Using Unstructured Meshes

A C Bélanger and R B Simpson *

November 16, 2006

Abstract

The Black-Scholes value and risk free portfolio for options that depend on two underlying assets are defined by the solution of the B-S partial differential equation, and its gradient . A finite volume, or finite element, method using an unstructured mesh to discretize the underlying asset price region can be used to compute a piecewise linear (pwlinear) approximation for the valuation. A gradient recovery method applied to this computed valuation can be used to compute a pwlinear approximation to the B-S delta hedging parameters.

In this paper, we discuss efficient unstructured mesh design for computing both of these contributions to the B-S risk free portfolio function. Mesh designs are viewed as determining sequences of meshes with increasingly accurate computed valuations and deltas. We argue that meshes that provide quadratic convergence to zero of the errors in the deltas are more efficient that meshes with lower convergence rates for these errors. However, requiring quadratic convergence of the recovered gradient values of a function places strong restrictions on the mesh design. We present computations that demonstrate how meshes with suitably chosen uniform submeshes can produce quadratic convergence of the deltas in a weighted error measure.

Keywords *gradient recovery, finite volume method, finite element method, Black-Scholes, efficiency*

1 Introduction

The context of this paper is the computation of the Black-Scholes valuation and dynamic hedging parameters for an option that depends on two underlying asserts (Chapter 11, Wilmott, [24]). The prices for these assets will be denoted by S_1 and S_2 and the pair will often be written as column vector S. For a wide range of options, the valuation, $V(S,t)$, can be specified as the solution of the Black-Scholes partial differential equation, (pde), in a domain Σ of the S plane for a time interval between the issuing date, t_{start}, and the expiry date, t_{exp}. As per the Black-Scholes theory, the dynamic hedging parameters, $\Delta(S,t)$, are

*School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, N2L 3G1

the spatial gradient of V, i.e. $\Delta(S_1, S_2, t) = grad_S V(S_1, S_2, t)$, considered to be a column vector and the theoretically risk free portfolio function is $P(S,t) = -V(S,t) + S^T \Delta(S,t)$, as introduced at (6) below.

The computation of the valuation and hedging parameters on an unstructured mesh for Σ can be carried out in two stages:

1.a) the computation of a piecewise linear (pwlinear) approximation, V, of the valuation using the finite element method (FEM) or finite volume method (FVM)

1.b) the computation of approximate gradient values from a pwlinear function by a gradient recovery method. This produces $\Delta \equiv grad^R V$ as a post-processing step following 1.a). (1)

The use of the FEM/FVM methods[1] for computing V, approximately, on unstructured meshes for general parabolic pdes is well established. The application of the FVM to option valuation is presented in Zvan, Forsyth and Vetzal, [26] and [27], and in Pooley et al. [14]. The application of the FEM is presented in Forsyth, Vetzal and Zvan in [3]. A FEM computation of the valuation function for the demonstration two factor European call of §4 is presented by Zhu and Stokes in [25]. An extensive review of gradient recovery techniques and theory is presented by Lakhany, Marek and Whiteman, [10] ; see also Bank and Xu [1].

Two of the issues influenced by the mesh are:

a) the accuracy of the approximate values, and

b) the cost of their computation.

These two issues are in conflict in the sense that making the errors smaller invariably involves larger meshes and hence larger computation costs. But how much bigger need the cost be to accomplish a specified increase in accuracy? This is partly a question of mesh design efficiency, which is the major focus of this paper.

1.1 Efficiency: unstructured and structured meshes

Efficiency is pursued differently with structured and unstructured discretizations, at least in principle. Structured meshes have limited control over the location and hence the number of vertices in the discretization. However, the computational cost per vertex can often be kept low by exploiting the structure of the grid. Unstructured meshes have substantial flexibility in locating vertices where they provide needed error control, but the corresponding methods generally incur higher computational costs per vertex than well implemented structured mesh based counterparts. So, to be efficient, unstructured meshes should be designed to use fewer

[1] While implementations differ between these two methodologies, the results computed by them are essentially equivalent, in our view. See [8].

vertices by placing them only where they are effective in reducing errors[2]. In this paper, we assume that the size of the mesh, e.g. the number of vertices, adequately reflects the contribution by meshing to the cost of the computation. I.e. for two meshes that produce acceptably accurate approximations, the smaller mesh is the more efficient one.

We look for accuracy in both the valuation function and the hedging parameters. The pointwise error in the computed portfolio function, $Err^{(P)}(S,t)$, combines the pointwise errors in the valuation , $Err^{(V)}(S,t)$, and the deltas, $Err^{(\Delta)}(S,t)$, in a natural way

$$Err^{(P)}(S,t) = -Err^{(V)}(S,t) + S^T Err^{(\Delta)}(S,t) \tag{2}$$

so $Err^{(P)}(S,t)$ is a useful measure of accuracy here. Control of $Err^{(P)}(S,t)$ is not sought uniformly across Σ. The extent of Σ is typically governed by modeling boundary conditions that apply in ranges of the S plane which have a very low probability of being visited by a Black-Scholes price history; see (3) in §2. We assume that the accuracy requirements can be relaxed in parts of Σ that have a low probability of being visited so that mesh spacings, and errors, can be larger there. In §4, an error weighting function that reflects this assumption is introduced.

One might think that an efficient unstructured mesh would use just enough vertices, strategically places, so that the contributions to $Err^{(P)}$ from $Err^{(V)}$ and the two hedging parameter errors, $Err^{(\Delta)}$, would be roughly comparable and such that their combination in (2), ignoring cancellations, meets the accuracy requirement on $Err^{(P)}$. I.e. one might think that an efficient mesh can be devised that roughly balances $Err^{(V)}$ and $Err^{(\Delta)}$. Simple realizations of this idea are confounded by the fact that $Err^{(V)}$ and $Err^{(\Delta)}$ are coupled by the mesh and also by the observation that, generally, achieving accuracy in the approximation of the derivative of a function is more costly than approximating the function itself. In §3, we provide an overview of the substantial theory of convergence of approximate function values and gradients for functions specified by pdes. This theory quantifies, for such functions, the previous observation about gradient accuracy being more costly to attain than function value accuracy. The overview provides some of the rationale for the meshing technique presented in this paper.

The theory studies the convergence of a sequence of pwlinear approximate functions, $\{V^{(j)}\}$, and recovered gradients, $\{grad^R V^{(j)}\}$, computed on a sequence of meshes $\{M_j\}$. It predicts the asymptotic behaviour of the errors in $V^{(j)}$ and $grad^R V^{(j)}$ as $j \to \infty$, assuming $h(M_j) \to 0$ for $h(M_j) =$ the longest triangle edge in M_j. These results, which we paraphrase without full qualifications, are expressed in the usual asymptotic terminology; a sequence of error sizes, or norms, $\{||Err^{(j)}||\}$, converges to zero:

- linearly if $||Err^{(j)}||/h(M_j)$ is bounded ; $||Err^{(j)}|| = O(h(M_j))$

- quadratically if $||Err^{(j)}||/h(M_j)^2$ is bounded ; $||Err^{(j)}|| = O(h(M_j)^2)$

[2]A separate and somewhat historical issue is that structured methods are simpler to program, at least for simple problems. We assume that the difficulty in implementing the methods is no longer an issue.

- strictly linearly if the sequence converges linearly, but $||Err^{(j)}||/h(M_j)^{1+\beta}$ is unbounded for every $\beta > 0$

For sequences $\{M_j\}$ of general unstructured meshes, the theory predicts that the valuation function errors, $Err^{(V,j)}$, converge quadratically to zero, but that the recovered gradient errors, $Err^{(\Delta,j)}$, converge linearly. (Recall $\Delta = grad^R V$ as per (1).) This prediction implies a fundamental incompatibility between using general unstructured mesh generation techniques and the tactic of seeking efficiency by balancing the contributions of $Err^{(V,j)}$ and $Err^{(\Delta,j)}$ to $Err^{(P,j)}$. If, as predicted, $Err^{(V,j)}$ converge quadratically and if $Err^{(\Delta,j)}$ exhibit strictly linear convergence, then the hedging parameter error will dominate $Err^{(P,j)}$, for stringent enough accuracy requirements, and the convergence of $Err^{(P,j)}$ will be strictly linear.

For sequences of structured meshes, on the other hand, the theory predicts quadratic convergence of both $Err^{(V,j)}$ and $Err^{(\Delta,j)}$, which would support the efficiency tactic of balancing the error contributions to $Err^{(P,j)}$ and also ensure quadratic convergence of $Err^{(P,j)}$. Furthermore, the theory supports the prediction of localized superlinear convergence in the following sense. Let Σ_1 be a subdomain of Σ, and, for each M_j of the mesh sequence, let the restriction of M_j to Σ_1 be uniform. Then the hedging errors, $Err^{(\Delta,j)}(S,t)$, converge superlinearly for S sufficiently interior to Σ_1. It is this prediction of the theory that we propose to exploit.

1.2 Outline of the paper

In §2, we review the specifications of the option valuation, the delta hedging parameters, and the hedging portfolio for a generic two factor option by the Black-Scholes model. We review the computation of these quantities for an unstructured mesh for 1.a), and gradient recovery methods for 1.b). We then introduce weighted error measures for the computed quantities.

In §3, we present an overview of, and references for, unstructured meshing and related pwlinear approximation theory and practice as it relates to the option calculations of §2. Using this, §3.2 presents the Delaunay refinement based meshing technique for option calculations that is demonstrated in §4. In principle, this technique can produce sequences of nested meshes, $\{M_j\}$, for which the weighted pointwise errors of the computed portfolio values show a strong form of quadratic convergence locally on uniform submeshes. A computable characterization of the asymptotic form associated with this convergence is discussed. §4 demonstrates that this characterization is present in the values of the portfolio function computed on three successive meshes of $\{M_j\}$. Using this as empirical evidence of the asymptotic behaviour of the weighted error, we extrapolate this error behaviour to the rest of the mesh sequence and draw conclusions about the mesh efficiency of this approach.

2 Pricing, Hedging and Computation for the Black-Scholes model

The contract for an option specifies the expiry date $t = t_{exp}$ of the option and its pay-off value at t_{exp} in terms of the underlying asset prices $S = (S_1, S_2)^T$ at this time. The Black-Scholes model assumes that the price histories of the underlying assets can be modeled as solutions of the simple stochastic differential equations determined by a Weiner process, W_t, as

$$dS_i = \mu_i S_i dt + \sigma_i S_i dW_t \quad \text{for} \quad i = 1, 2. \tag{3}$$

In the use of this pricing model, the valuation function, $V(S,t)$ satisfies the partial differential equation[3]

$$-\partial V/\partial t = \frac{1}{2}\sigma_1^2 S_1^2 \frac{\partial^2 V}{\partial S_1^2} + \frac{1}{2}\sigma_2^2 S_2^2 \frac{\partial^2 V}{\partial S_2^2} + \rho\sigma_1\sigma_2 S_1 S_2 \frac{\partial^2 V}{\partial S_1 \partial S_2} + r S_1 \frac{\partial V}{\partial S_1} + r S_2 \frac{\partial V}{\partial S_2} - rV \tag{4}$$

for the contract at times $t < t_{exp}$, subject to

$$V(S, t_{exp}) = V_{pay}(S). \tag{5}$$

The significance of the parameters in (3) and (4) are discussed in Wilmott, [24], Chapter 11.

To complete the specification of $V(S,t)$ from (5) and (4) for computation, it is necessary to limit S to a finite domain Σ. For this purpose, 'boundary' values of S_1 and S_2 are specified. The boundary values of S_i are typically taken at extreme prices of the underlying asset, i.e. $S_i = 0$ and $S_i = S_i^{(large)}$. So Σ is this square (or rectangular) solution domain At the boundaries of Σ, conditions on V are posed which are reasonable and which do not inappropriately influence V in the ranges of S_i where the underlying asset price histories evolve. See Zvan, Forsyth, and Vetzal, [26]. Equation (4) can be integrated backwards in time to provide $V(S,t)$.

Figures 1 A and 1 B show the payoff and valuation function at $t =$ three months prior to expiry for a European call on the maximum of two assets; see Stulz, [22]. This option is one of two used in our demonstration computations of §4, where more details of its parameters are given.

A hedging strategy is a tactic which can be adopted by the issuer for reducing the risk incurred under the contract. In general, the risk depends on the actual price history, $S(t)$, taken by the underlying assets, which is assumed to be an instance of the random processes of (3). A class of simple dynamic hedging strategies involves maintaining a portfolio consisting of the contract value, $V(S(t),t)$, as a liability and holdings in the amounts $\Delta_i(S(t),t)$ of the ith asset; i.e. holdings of assets worth $S^T\Delta(S(t),t) = S_1(t)\Delta_1(S(t),t) + S_2(t)\Delta_2(S(t),t)$. Using the sign conventions of [24], this portfolio is described mathematically by

$$P(S,t) = -V(S,t) + S^T\Delta(S,t). \tag{6}$$

[3] usually written in terms of a 'backward' time variable also designated 't'.

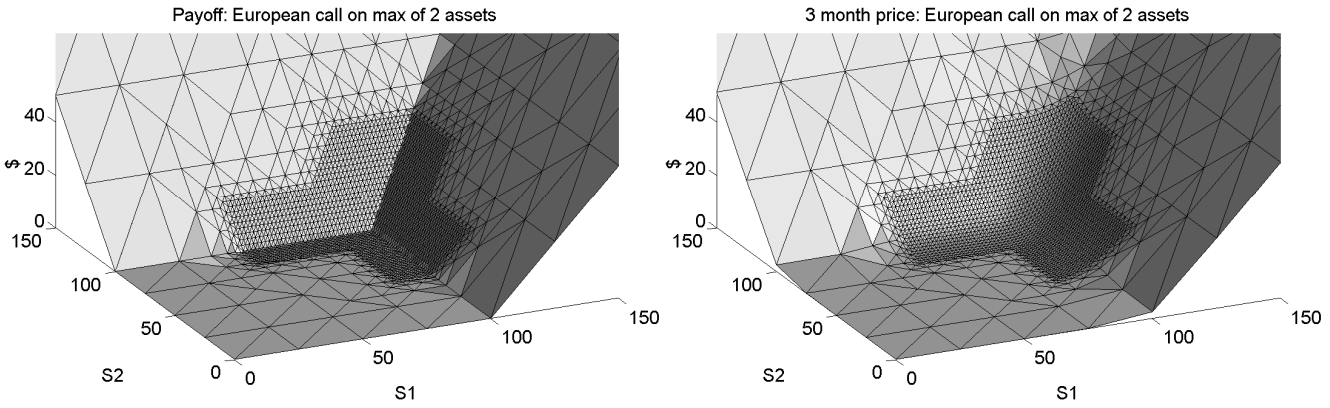


Figure 1: European call : payoff and issuing prices

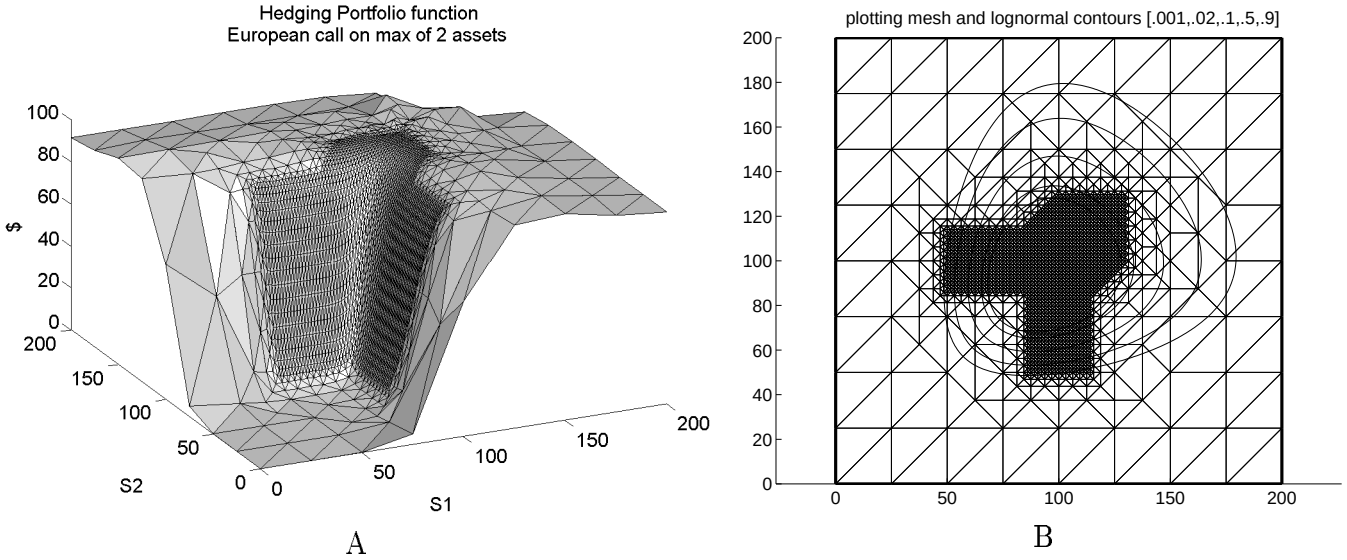


Figure 2: Portfolio function corresponding to Figure (1) and a computational mesh

Various choices for $\Delta_i(S(t),t)$ make different tactics within this general strategy. Under the assumptions of the Black-Scholes theory, the choice of

$$\Delta_i(S(t),t) = \partial V(S(t),t)/\partial S_i \text{ for } i = 1,2 \tag{7}$$

will completely eliminate the risk to the issuer, regardless of the pricing history path, [24]. Figure 2A shows the portfolio function at t = three months prior to t_{exp} for Black-Scholes hedging of the European call shown in Figure 1B.

2.1 Computing the model valuation and portfolio functions

Piecewise linear approximations for the valuation and portfolio functions can be computed using the two stage computation (1) described in §1. Stage (1.a) can be accomplished by the finite element or volume methods, integrating (4) backwards in time, using implicit time stepping. References for such methods are given in §1.

For our purposes, the inputs to the method are: the mesh M, the Black-Scholes equation parameters, the pay-off function, the length of the contract, the boundary conditions and, possibly, additional constraints such as early exercise option. The output is an approximate valuation function, $V(S,t)$, that is piecewise linear over M. To distinguish it from the exact solution of (4), we will denote the exact solution of the initial-boundary value problem for (4) by $V^{(exact)}(S,t)$ for the rest of the paper. The demonstrations presented in this paper use t = three months prior to t_{exp}.

The computation of the delta hedging parameters for stage (1.b) can be accomplished by a gradient recovery technique. Consider a bivariate function, $W^{(exact)}(S)$, and a piecewise linear approximation, $W(S)$, to $W^{(exact)}$ which is defined using a mesh, M. $grad\ W(S)$ is a vector function which is piecewise constant on the triangles of M. A gradient recovery technique post-processes the data of $grad\ W(S)$ to extend its definition to all values of S, usually as a piecewise linear vector function. We denote the result of this post-processing by $grad^R\ W(S)$, and we expect that it is a significantly better approximation to $grad\ W^{(exact)}(S)$ than $grad\ W(S)$ is.

The method used in the demonstrations of this paper is a standard FEM gradient averaging technique:

8.a) Let $S^{(k)}$ be a vertex of mesh M. Let $grad^R\ W(S^{(k)})$ be the simple average of the constant values of the gradients $grad\ W$, for each triangle incident on $S^{(k)}$

8.b) Let $grad^R\ W(S)$ be the piecewise linear vector interpolant of the $grad^R\ W(S^{(k)})$ data on M (8)

Gradient recovery methods and the theory of their error behaviour will be reviewed in §3.1.2; we note here, however, that getting significant accuracy gains by this (or any other) gradient recovery technique seems to be contingent on imposing some structure on the mesh, M.

2.2 Meshing error definitions

Computing solutions of (4) involves two discretizations: the mesh discretization of Σ and the time stepping technique. In this paper, we are focusing on the former; so we will assume that the time stepping errors are neglible, i.e. that very small time steps are used. Let $w(S)$ be a positive error weighting function. The role of $w(S)$ is primarily to limit the domain over which the error behaviour is to be studied. The weighted absolute meshing errors in the value and hedging parameters are

$$\begin{aligned} Err^{(V)}(S,t) &= w(S)(V^{(exact)}(S,t) - V(S,t)) \\ Err^{(\Delta)}(S,t) &= w(S)(grad\ V^{(exact)}(S,t) - grad^R V(S,t)). \end{aligned} \tag{9}$$

These are pointwise errors in the underlying asset price variables S, as well as time. The theory of error behaviour is developed for general pdes and not explicitly for weighted errors. So when we review this theory, we will assume $w(S) = 1$. However, in the Black-Scholes theory for the quantities that we are discussing, the underlying price histories are random walks in the S plane based on (3). Since the boundaries of the computational domain are deliberately set in regions of very low probability of encountering a price path, it is not necessary to maintain uniform accuracy across the computational domain for most end uses of the calculations. In §4, we demonstrate a choice of error weight function, $w(S)$ which limits the visualization of errors appropriately for each of the options presented.

As presented in (2), the weighted error in the portfolio function is

$$Err^{(P)}(S,t) = -Err^{(V)}(S,t) + S^T Err^{(\Delta)}(S,t). \tag{10}$$

From (10) we can see that $Err^{(P)}(S,t)$ provides a natural way to look at balancing $Err^{(V)}$ from stage 1.a) with $Err^{(\Delta)}$ from stage 1.b) of (1). Note that controlling the absolute portfolio error requires controlling the *absolute* error of the value function and controlling the error in Δ_i that is *relative* to the price, S_i of the ith underlying asset.

3 Unstructured Meshing and Piecewise Linear Functions

Some of the features of efficient meshes for these computations can be seen directly from Figure 1 and Figure 2, at least in general qualitative terms.

11.a) There are large regions of Σ in which the functions are essentially planar. In these regions, accuracy can be obtained by relatively large triangles.

11.b) There are regions where a price history is unlikely to visit. We assume that accuracy can be relaxed there, which means that large triangles can be used. (11)

11.c) There are regions where the surfaces show significant curvature and price histories are likely to visit. Small triangles are required for accuracy in this zone.

Imagine that such a mesh has been generated which is reasonably efficient by virtue of the distribution of triangle sizes that we have described, and that the valuation and portfolio functions have been computed. If we now wish to increase the accuracy of our computation, we would have to use smaller triangles in the small triangle zone, i.e. we would need a second mesh that is larger than the first. How do the contributions of $Err^{(V)}$ and $Err^{(\Delta)}$ to $Err^{(P)}$ change from the first mesh to the second? It is probably not possible to answer this vague question for an arbitrary pair of meshes. However, there is a substantial theory of pwlinear approximations which predicts that in general, using a finer unstructured mesh to achieve more accuracy will likely result in the gradient error dominating the portfolio error. We review this theory in §3.1.1. The theory of recovered gradient superlinear convergence suggests a two part remedy for this potential imbalance, as we review in §3.1.2. We describe how this remedy can be implemented for our option computations in §3.2.

3.1 Review of convergence theory

Convergence studies for FEM and FVM methods are usually based on a sequence of meshes $\{M_k\}$ such that $h(M_k) \to 0$ as $k \to \infty$ where $h(M_k) = \max_{T \in M_k}(h(T))$ and $h(T)$ is the length of a longest edge of triangle T. Such a sequence is referred to as a convergent mesh sequence. As the triangles of M_k get smaller with $k \to \infty$, their shape should not degenerate indefinitely in the sense that there should be a bound on the smallest angle in any T in $\{M_k\}$. This is guaranteed if the M_k are created by quality mesh generation methods as described above. In the convergence literature, this requirement is usually stated equivalently as requiring an upper bound on the ratio of the circumcircle of T to the shortest edge of T for all T in $\{M_k\}$. Mesh sequences with this property are called quasi-uniform. We note that for a quasi-uniform mesh sequence, the number of vertices in M_k, $N_V(M_k)$, grows asymptotically like $N_V(M_k) = O(h(M_k)^{-2})$.

3.1.1 A basic convergence rate estimate

Let $W^{(exact)}(S)$ be a function with continuous second derivatives in Σ and let $\{M_k\}$ be a quasi-uniform sequence of meshes for Σ. Let $W^{(k)}(S)$ be one of the common piecewise linear approximations to $W^{(exact)}$ on M_k. I.e. $W^{(k)}(S)$ could be computed by interpolation, or least squares fitting, or numerical solution of a partial differential equation. There is

a comprehensive theory for the convergence of $W^{(k)}$ to $W^{(exact)}$; two of the fundamental features of it are:

a) the convergence of $W^{(k)}$ to $W^{(exact)}$ is quadratic in $h(M_k)$, i.e.

$$Err^W = W^{(exact)} - W^{(k)} = O(h^2(M_k)) \tag{12}$$

b) the convergence of the piecewise constant functions $\partial W^{(k)}/\partial S_j$ to $\partial W^{(exact)}/\partial S_j$ is linear in $h(M_k)$, i.e

$$Err^{\partial W/\partial S_j} = \partial W^{(exact)}/\partial S_j - \partial W^{(k)}/\partial S_j = O(h(M_k)) \tag{13}$$

Precise statements of these features involve a variety of size measures for Err^W and $Err^{\partial W/\partial S_j}$. Rigorous statements of (13) are complicated by the fact that $\partial W^{(k)}/\partial S_j$ are, in general, only pointwise defined in the interior of the triangles of the mesh.

The monograph [17] by Schatz, Thomée and Wendland provides a useful rigorous presentation of this theory[4]. In particular, (12) and (13) hold in the L_2 norm when $W^{(exact)}(S)$is the solution of a parabolic pde in d dimensions at some fixed time level and $W^{(k)}$ is the corresponding approximate spatial profile computed by a FEM method using M_k; see [17], Theorems 1.1, page 140, and 1.2, page 147.

The key aspect of these results for our purposes is that they suggest that the rate at which the piecewise constant values of $gradV(S,t)$ converge to $gradV^{(exact)}(S,t)$ can be significantly slower than the rate at which $V(S,t)$ converges to $V^{(exact)}$. Consequently, if $gradV(S,t)$ were used for the deltas of the portfolio, (6), then the term $S^T Err^{(\Delta)}$ of (10) would dominate $Err^{(P)}$, at least for fine meshes. This suggests that a balance between the errors in the terms of the portfolio function as described in §2.2 could not be achieved using these delta values.

This imbalance between the convergence rates of $W^{(k)}$ and $gradW^{(k)}$ was observed in stress analysis, and other applications that require gradient values early in the use of the FEM. Computational techniques for improving the computation of approximate gradient values has about a 35 year history. In particular, gradient averaging as presented at (8) has been used with the FEM since 1974, [6]. But are they good enough to support a balance of the value and delta errors in the portfolio computation? In the following subsection, we summarize the theory for approximation properties of gradient recovery methods.

3.1.2 Gradient Recovery, superconvergence, and asymptotic error behaviour

One might hope that a gradient recovery technique, like the one described at (8), could be found that would produce approximate gradient component functions that converge quadratically to the exact gradient components. Unfortunately, there is no established gradient

[4] In [17], page 9 and sequel, and other convergence literature, these convergence rates are combined into the single expression

$$\max|W^{(exact)}(S) - W^{(k)}(S)| + h(M_k)\max||grad\, W^{(exact)}(S) - grad\, W^{(k)}(S)|| = O(h(M_k)^2). \tag{14}$$

E.g.see Corollary 1.2.2

recovery scheme that is demonstrated to be quadratically convergent for a general sequence of quasi-uniform meshes, $\{M_j\}$ as we have described above.

The theory of gradient recovery techniques has focused on demonstrating convergence rates that are better then linear, i.e. $Err^{grad\,W} = O(h^{1+\beta}(M_k))$ for some β between 0 and 1. In appendix A, we argue that full quadratic convergence, $\beta = 1$, is necessary to get the error balancing we seek. These methods do not apply to sequences of arbitrary quasi-uniform sequences of unstructured meshes, however. In fact, in their most basic form they apply to essentially structured meshes; i.e. meshes in which the two triangles that are incident on any internal edge almost form a parallelogram. See the review [10], cited in the introduction. On such meshes, gradient recovery methods may be viewed as generalizations of symmetric finite difference methods. We know that these methods typically compute approximate gradient values that are quadratically convergent. Consider the simple centered difference

$$\delta_{1,h}\, W(S) = (W(S_1 + h, S_2) - W(S_1 - h, S_2)/(2h).$$

Using Taylor's expansions, we can see that

$$\partial W(S)/\partial S_1 \; - \delta_{1,h}\, W(S) = O(h^2) \tag{15}$$

if $\partial^3 W(S)/\partial S_1^3$ is bounded. This simple observation has been generalized, albeit in a reduced form, in [18], by Schatz, Sloan and Wahlbin. They show how local symmetry in the mesh can be combined with gradient recovery to yield superconvergence. Their results are quite general and technically complex to state with precision. They apply to a convergent sequence of quasi-uniform meshes, $\{M_j\}$ which cover designated parts of Σ with submeshes that have forms of symmetry in the distribution of neighbouring vertices, including locally uniform submeshes of uniform spacings h_j. Pointwise superconvergent gradient error estimates are presented for a class of gradient recovery methods that includes simple gradient averaging at the vertices. These estimates apply, in particular, in the case of a sequence of meshes, $\{M_j\}$ on Σ which are locally uniform on a subdomain Σ_1 of Σ as we describe below in §3.2. They show that the recovered gradient error is $O(h^{4/3})$ at points sufficiently interior to Σ_1. While this is local superconvergence resulting from local symmetry among the mesh vertices, it is not the quadratic convergence that we observe in §4; it seems that the theory is too pessimistic in its application to our computations.

3.2 Generating efficient meshes and identifying quadratic convergence

We will demonstrate a simple class of meshes generated by Delaunay refinement that The meshes to be used in our demonstrations are generated by Delaunay refinement, which is a standard technique discussed in George and Borouchaki, [4]. The input to Delaunay refinement is:

- a coarse Delaunay mesh on Σ,

- a distributed target size for triangles, $sizeTol(S)$. We will use the length of a longest edge of T as its size measure, $h(T)$.

- a constant tolerance (lower bound) for the minimum angle of a triangle, $angleTol$.

The result of a Delaunay refinement method is a CDT, M such that for every triangle, T, in M

a) $h(T) \approx sizeTol(S_{mid})$, where S_{mid} is the midpoint of the longest edge of T,

b) the minimum angle of T is not less than $angleTol$.

Property b) of this description of the results is typically realizable for $angleTol$ up to $\pi/6$ radians $= 30°$ degrees. Mesh generation methods that satisfy b) are often referred to as 'quality' mesh generation methods. Quality Delaunay refinement methods have been described by Rivara, Hitschfeld, and Simpson, in [15], by Ruppert [16] and by Shewchuk, [19][5]. support efficient computation and empirically exhibit locally quadratic convergence of the portfolio values. We use the theory reviewed in the preceding subsections as guidance, but this theory does not rigorously extend to the proposed meshes.

In §2.1 at (11), we noted several subregions in the underlying asset price region, Σ, which had implications for mesh efficiency and computed solution accuracy. The subregion of Σ for which $V^{(exact)}(S,t)$ is expected to have significant second derivatives depends on t. Since we want the simplicity of a static mesh, we will identify it with its widest extent, which occurs at the issuing time of the option. Let Σ_A enclose this subregion, so that $\Sigma - \Sigma_A$ is of type 11.a). Let Σ_B be the type 11.b) zones in which accuracy is not a concern (i.e. low probability of a price history). Then $\Sigma_1 = \Sigma_A \bigcap (\Sigma - \Sigma_B)$ is the subregion of Σ in which small mesh spacing is needed for accuracy generally, and mesh symmetry is needed for quadratic convergence of $grad^R V(S,t)$. A mesh that is uniform in Σ_1 of triangle size h and appropriate for numerically solving (4) in all of Σ can be generated by Delaunay refinement using

$$\begin{aligned} sizeTol(x,y)(S) &= h \text{ for } S \in \Sigma_1 \qquad (16) \\ &= H \text{ for } S \in \Sigma - \Sigma_1 \end{aligned}$$

$$(17)$$

for $H >> h$. This approach can be extended to produce sequences of meshes $\{M_j\}$ that are nested on Σ_1 by setting

$$\begin{aligned} sizeTol(x,y)(S) &= h/2^j \text{ for } S \in \Sigma_1 \qquad (18) \\ &= H \text{ for } S \in \Sigma - \Sigma_1 \end{aligned}$$

and using M_{j-1} as the initial mesh for refinement of M_j. These meshes are nested in the sense that if $P_k \in M_{j-1} \cap \Sigma_1$, then $P_k \in M_{j+n}, n = 0, 1, 2, \ldots$. From a theoretical view point,

[5] A URL for a freely distributed implementation by Shewchuk, named `Triangle`, is included with this reference.

the sequence of locally uniform, locally nested meshes just described are quasi-uniform as per §3.1 since they are produced by a quality mesh generation algorithm. We will demonstrate that values and portfolios computed on Σ_1 will converge quadratically.

To empirically exhibit local quadratic convergence, we use long established ideas about asymptotic error behaviour, e.g. Keller [9], page 78. We review this topic by returning to the example (15) of centered differencing. Assuming that $\partial^4 W(S)/\partial S_1^4$ is bounded, then the Taylor's expansion analysis of $\partial W^{(exact)}/\partial S_1 - \delta_{1,h} W(S)$ can be extended to show

$$Err^{\partial W/\partial S_1}(S,h) = \partial W^{(exact)}/\partial S_1 - \delta_{1,h} W(S) = \phi(S) h^2 + O(h^3), \tag{19}$$

where

$$\phi(S) = 1/6 \; \partial^3 W^{(exact)}(S)/\partial S_1^3.$$

This form is familiar as the basis of extrapolation techniques for increasing accuracy or estimating errors, cf. Morton and Mayers, §6.6 [13] or Keller, [9]. We will refer to the function $\phi(S)$ as the magnified error function[6]. If, in an interval $0 < h \leq \bar{h}$, the term denoted $O(h^3)$ of (19) is indeed small relative to the principal error, $\phi(S)h^2$, then $Err^{\partial W/\partial S_1}(S,h)$ is in the asymptotic error region of h. For this range of values of h, the variation of $Err^{\partial W/\partial S_1}(S,h)$ with S is described by the magnified error function, $\phi(S)$ independent of h, and the size of the error is scaled by h^2, to good approximation. Observe that for $j = 0, 1, 2, \ldots$

$$\begin{aligned} \delta_{1,h/2^{j+1}} W(S) - \delta_{1,h/2^j} W(S) &= Err^{\partial W/\partial S_1}(S, h/2^j) - Err^{\partial W/\partial S_1}(S, h/2^{j+1}) \\ &= \phi(S)(3/4)h^2/4^j + O(h^3) \\ &= \phi(S)(3h^2/4^{j+1}) + O(h^3). \end{aligned} \tag{20}$$

So, for h in the asymptotic error range,

$$\phi(S) \approx 4^{j+1}(\delta_{1,h/2^{(j+1)}} W(S) - \delta_{1,h/2^j} W(S))/(3h^2) \tag{21}$$

i.e. in the asymptotic error range the quantity on the right side of (21) is essentially independent of h and j.

We can push this simple example further to explain our empirical tests presented in the next section. Consider a horizontal line segment in the (S_1, S_2) plane, $R = \{(S_1, S_2) | a \leq S_1 \leq b, S_2 = constant\}$. An empirical test to indicate whether a value of h is in the asymptotic error range for $\delta_{1,h} W(S)$ for $S \in R$ would be to compute

$$\phi_C(S) = 4(\delta_{1,h/2} W(S) - \delta_{1,h} W(S))/(3h^2), \tag{22}$$

and

$$\phi_B(S) = 16(\delta_{1,h/4} W(S) - \delta_{1,h/2} W(S))/(3h^2). \tag{23}$$

If $\phi_C(S) \approx \phi_B(S)$ for $S \in R$, then we would conclude that, locally in R

- h is in the asymptotic error range,

[6]This is the name introduced by Henrici,1962, §2.2-6 [5]

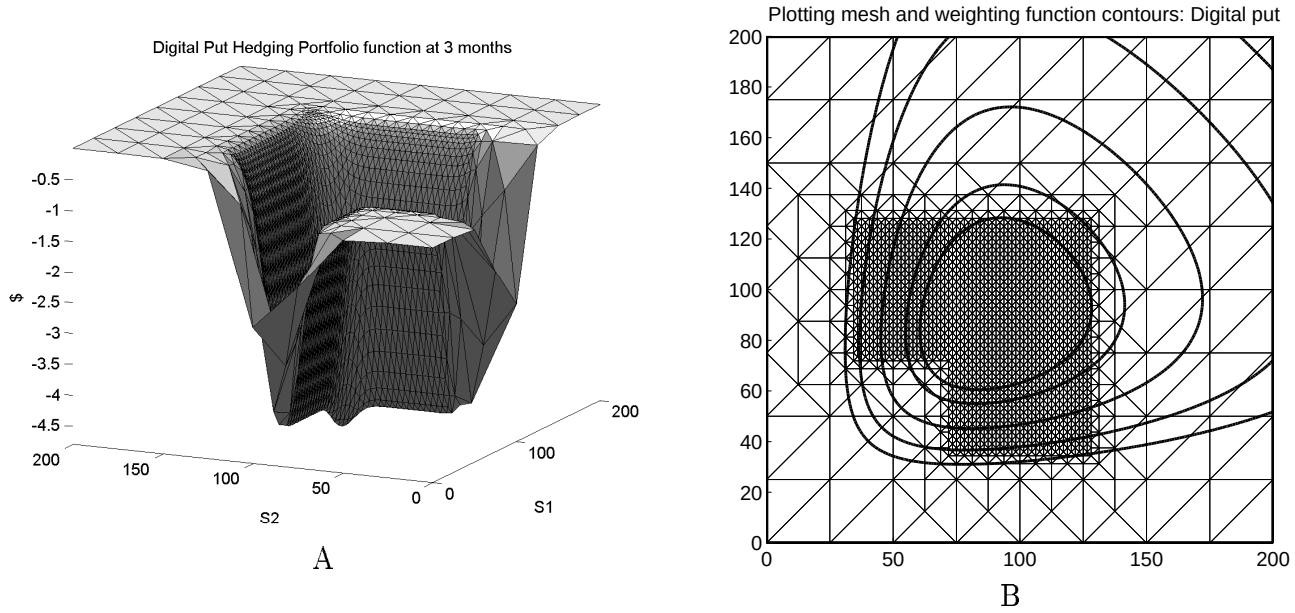


Figure 3: Digital put option portfolio and mesh showing region Σ_1

- $\delta_{1,h}W(S)$ converges quadratically to $\partial W(S)/\partial S_1$,
- $Err^{\partial W/\partial S_1}(h', S) \approx \phi_C(S)h'^2$ for $h' \leq h$.

Of course, these are empirical conclusions, not mathematical facts.

In §4, we apply these ideas to the weighted option value and portfolio errors, $Err^{(V)}(S,t)$ and $Err^{(\Delta)}(S,t)$ of (9). I.e. we hypothesize a magnified error function, $\psi^{(V)}(S,t)$ such that for mesh sequence $\{M_j\}$ of (18),

$$Err^{(V)}(S) = \psi^{(V)}(S,t)\, h_j^2 + O(h_j^3) \quad \text{for } S \in \Sigma_1, \tag{24}$$

and similarly for $Err^{(\Delta)}(S,t)$. We provide empirical evidence to support this hypothesis by computing the analogues of (22) and (23) and observing that they are approximately the same. Recently, Ma, Mao, and Zhou, [11], have validated error expansions like (24) for the FEM and FVM. They consider variable coefficient elliptic pdes and families of piecewise uniform triangular meshes parameterized by a uniform refinement parameter, h. In particular, quadratic error expansions are established for recovered gradients.

4 Demonstrations

In this section, we demonstrate the error behaviour for computing the valuation and portfolio functions of a European call, and a digital put, on the maximum of two assets, using the simple uniform core meshing strategy described in §3.2. The primary computational difference between these two options is that the payoff function for the call option is continuous, with discontinuities in its gradient, while the payoff for the digital put is discontinuous.

Consequently, the hedging parameters, and the portfolio function, become unbounded as the option expiry time is approached. For times prior to expiry, the portfolio function is mathematically well defined, which we compute. The general character of this portfolio is shown in Figure 3 A. The fluctuations of this function make it a useful test for the computation of hedging parameters, even if it is not a practical vehicle for risk reduction.

After describing some details of these two options, in §4.1, we demonstrate magnified error functions for the errors in these option functions in the subdomain Σ_1 of uniform mesh spacings described by (17). This is the evidence that, internal to Σ_1, we have quadratic convergence of the computed valuation and portfolio functions, on mesh sequences, $\{M_j\}$ generated as per §3.3. In §4.2, we use these magnified error functions to discuss the degree of balance achieved between the weighted valuation errors and the hedging parameter errors using this class of meshes.

The demonstrations impose two different payoffs for the same model of underlying assets. The asset parameters for the Black-Scholes model (4) are $\sigma_i = 0.2$ for $i = 1, 2$; their correlation coeficient is $\rho = 0.5$ and the risk free interest rate is $r = 5\ \% = .05$. The strike price for each is \$100 and Σ is the square $\{0 \leq S_i \leq 200\ | i = 1, 2\}$. The payoff for the call option is piecewise linear $V_{call}(S) = \text{Max}\ (0, S_1 - K, S_2 - K)$ and the discontinuous payoff for the digital put is $V_{dput} = -1$ if $\max(S_1, S_2) < 100,\ = 0$ otherwise.

The different payoffs for these two options lead to different subregions, Σ_1 in which small uniform triangles are used, as discussed in §3.2. The general character of the call option valuation function is shown in Figure 1 B of §2. The regions of the (S_1, S_2) plane in which $grad\ V$ is essentially constant are

$$\begin{aligned} grad\ V(S,t) &\approx (0,0)^t \text{ for } 0 \leq S_k << K,\ k = 1,2 \\ grad\ V(S,t) &\approx (1,0)^t \text{ for } S_1 >> K,\ \ S_2 < S_1\ \text{, and} \\ grad\ V(S,t) &\approx (0,1)^t \text{ for } S_2 >> K,\ \ S_1 < S_2. \end{aligned}$$

There are three relatively narrow 'gradient transition' zones separating these regions in which $grad\ V(S,t)$ varies significantly with S. We assume that the purpose of the computation is pricing and portfolio management for histories of the underlying asset values that are not highly improbable. Hence accuracy is also not required in the three extremes of the transition zones; i.e. $S_1 \to 0$, S_2 near K; $S_2 \to 0$, S_1 near K; and $S_i \to \infty$, $S_1 \approx S_2$ These zones have dynamic width; at the contract expiry time they shrink to zero width and they grow in size as t moves to earlier times. As mentioned above, we define the zones to have their widest extent, which occurs at issuing time. So, for the call option, we identify the subdomain of Σ where small triangles are needed for accuracy, as

$$\begin{aligned} \Sigma_1 \ = \ & (85 \leq S_1 \leq 115; 50 \leq S_2 \leq 115) \\ & \cup (85 \leq S_2 \leq 115; 50 \leq S_1 \leq 115) \\ & \cup (|S_1 - S_2|/\sqrt{2} \leq 22; 85 \leq S_1 \leq 130; S_2 \leq 130). \end{aligned} \tag{25}$$

A coarse version of the mesh for the call demonstration is shown in Figure 2 B, superimposed on this mesh are the contours of the error weighting function, $w(S)$ discussed further below.

The payoff for the digital put of our second demonstration is discontinuous on the two line segments $0 \leq S_1 \leq K$, $S_2 = K$ and $0 \leq S_2 \leq K$, $S_1 = K$. So, as can be seen from Figure 3 A, this option has transition zones centered on these two line segments. Consequently, we have defined Σ_1 as an L-shaped subregion

$$\Sigma_1 = (33 \leq S_1 \leq 129\ ; 73 \leq S_2 \leq 129) \cup (33 \leq S_2 \leq 129\ ; 73 \leq S_1 \leq 129). \qquad (26)$$

4.1 Estimating the magnified error functions and visualization

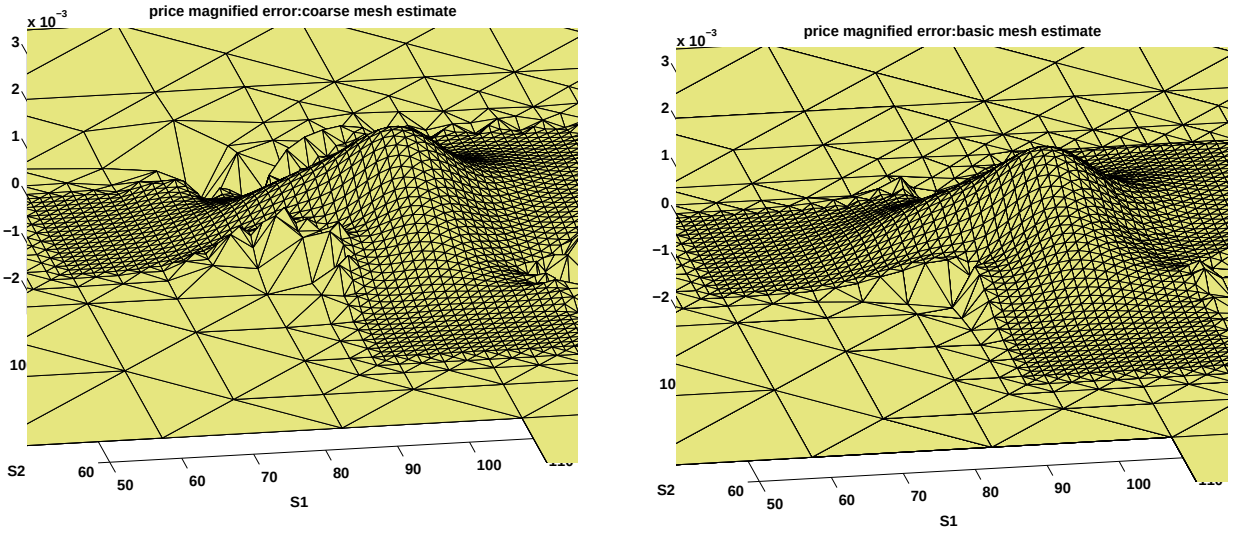
Our claim to quadratic convergence in Σ_1 is based on the similarity of estimates for the magnified error functions based on halving the mesh spacings in Σ_1. The basic idea of this comparison was discussed using an elementary example at (22) and (23). One estimate for a magnified error function compares data computed using a uniform mesh on Σ_1 of spacing h with data computed on a similar mesh with spacing $h/2$. A comparison of two such estimates can be made using three nested meshes with spacings h, $h/2$ and $h/4$. Consequently, the data for the call and the digital put options are computed on three nested meshes using (18) with a basic spacing parameter, h, of (16) set to a basic mesh spacing h_b, a coarse spacing, $h_c = 2h_b$, and a fine spacing, $h_f = h_b/2$. We will also index the computed values and the errors for each mesh by $j = c, b, f$ for 'coarse', 'basic' and 'fine'. E.g. $V^{(f)}(S)$ is the valuation function computed on the fine mesh; $P^{(b)}(S)$ is the portfolio computed on the basic mesh; $Err^{(V,b)}(S)$ is the valuation error on the basic mesh and $Err^{(P,c)}(S)$ is the portfolio error on the coarse mesh. The mesh sizes are given in Table (1.

mesh index	call		digital put	
x	h_x	no. of vertices	h_x	no. of vertices
coarse	3.0	1959	4	1741
basic	1.5	7234	2	6844
fine	.75	27363	1	26071

Table 1: Mesh Sizes

The coarse mesh for an option is also used for plotting the data surfaces in the figures for that option. The data computed using a finer mesh is adequately represented visually using the coarse mesh nodal values and this provides efficiency in data presentation. Consequently, all data for each option appears plotted on the coarse mesh, although it may have been computed using a finer mesh. E.g. Figure 3 A, shows the values of $P^{(b)}$ displayed on the mesh with uniform spacing h_c shown in Figure 3 B.

We are primarily concerned with the error behaviour in Σ_1; the weighting function, $w(S)$, introduced in (9) is the mechanism for limiting our view of the errors over all of Σ . One choice of $w(S)$ for this purpose could have been the characteristic function of Σ_1, i.e. $w(S) = 1$ for $S \in \Sigma_1$ and $w(S) = 0$ otherwise. It seems to us more informative to define $w(S)$ to be proportional to the probability density for a price path visiting S if it started



A : $\psi_{est}^{(V,c)}(S)$ B : $\psi_{est}^{(V,b)}(S)$

Figure 4: Call option valuation magnified error function estimates

near the double strike price (K, K). Our choice of weighing function, then, is a two asset log normal probability density function renormalized to have value 1 at the double strike price. Details are given in Appendix B. A comparison of the contours of $w(S)$ with Σ_1 for each case is shown in Figures 2 B and 3 B .

The asymptotic error theory reviewed in §3.3 refers to a mesh sequence, $\{M_j\}$ for $j = 0, 1, \ldots$ where M_j is characterized by a mesh spacing parameter, h_j. The corresponding sequence of valuation errors, $\{Err^{(V,j)}\}$, exhibits a strong form of quadratic convergence if (24) holds, i.e. there is a magnified error function, $\psi^{(V)}(S)$, such that

$$Err^{(V,j)}(S) = \psi^{(V)}(S)\, h_j^2 \; + \; O(h_j^3).$$

If $Err^{(V,j)}(S)$ is the weighted error, $w(S)(V^{(exact)} - V^j)$, and h_0 is in its asymptotic range, and $h_j = h_0/2^j$ then (24) implies that the computable quantity

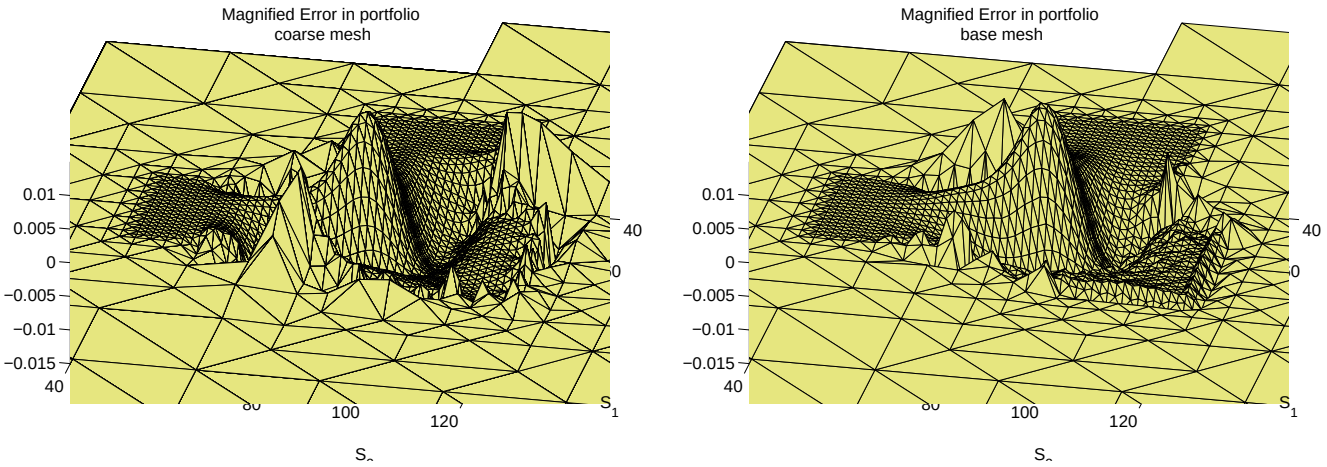
$$4^j\, w(S)(V^{(j)} - V^{(j-1)}(S))/(3\, h_0^2) \equiv \psi_{est}^{(V,j)}(S) \tag{27}$$

is approximately independent of j. Conversely, if the functions $\psi_{est}^{(V,j)}(S)$ are essentially independent of j, for $j = 1, 2, \ldots, J$, it is strong empirical evidence for the asymptotic behaviour (24). This observation applies also to the portfolio function.

To relate this theory to our demonstration, we identify $h_0 \simeq h_c$, $h_1 \simeq h_b$ and $h_2 \simeq h_f$ and we compute and display

$$\psi_{est}^{(V,c)}(S) \;=\; 4w(S)(V^{(b)} - V^{(c)})/(3\, h_c^2) \tag{28}$$

$$\psi_{est}^{(V,b)}(S) \;=\; 4w(S)(V^{(f)} - V^{(b)})/(3\, h_b^2). \tag{29}$$



A : $\psi_{est}^{(P,c)}(S)$ B : $\psi_{est}^{(P,b)}(S)$

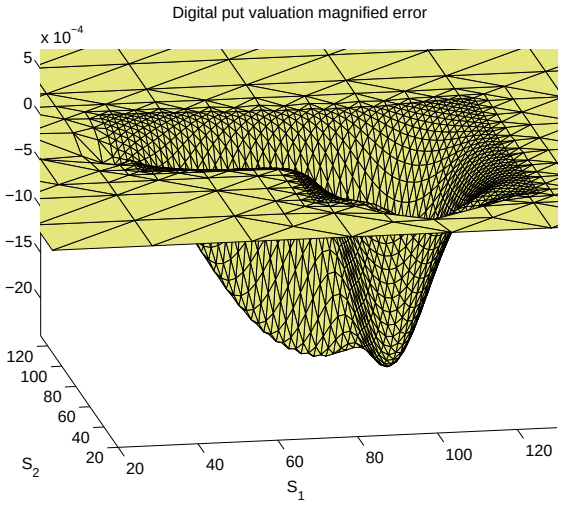
Figure 5: Call option portfolio magnified error function estimates

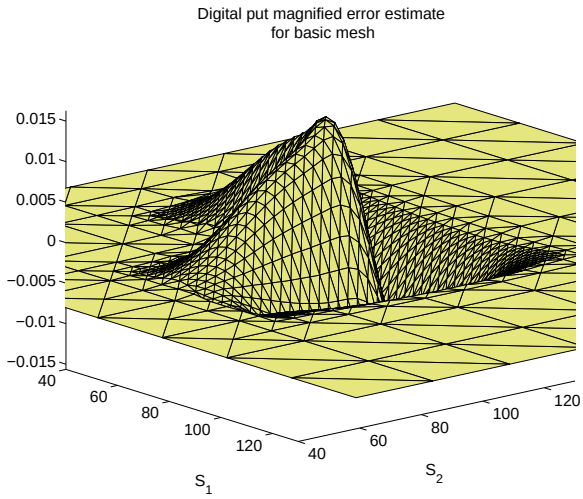
They are presented in Figures 4A and B. The size and features of $\psi_{est}^{(V,c)}$ and $\psi_{est}^{(V,b)}$ are essentially the same, which is the support for our claim of the presence of a magnified error function for $Err^{(V)}(S)$. Figure 5 compares $\psi_{est}^{(P,c)}(S)$ on the left with $\psi_{est}^{(P,b)}(S)$ on the right[7]. Again, the similarity of the two estimates in Σ_1 provide strong evidence of the presence of a magnified error function $\psi^{(P)}(S)$. In both Figures 4 and 5, some irregular differences in the distribution of error can be seen at the unstructured part of the mesh adjacent to Σ_1, The fact that these irregularities are not consistent in the two surfaces is evidence against the hypothesis of a magnified error function that extends outside Σ_1.

The apparent presence of a magnified error function for $V(S,t)$ in Σ_1 empirically implies quadratic convergence of $Err^{(V)}(S)$ to zero for $S \in \Sigma_1$. This asymptotic behaviour of the error can be anticipated as a minor strengthening of the prediction of the general theory of convergence reviewed in §3.1.1 that $Err^{(V,j)}(S)$ converges to zero quadratically on all of Σ for a convergent sequence of quasi-uniform meshes, $\{M_j\}$. This general convergence theory predicts only linear convergence of $Err^{(P,j)}$ to zero on Σ for these general mesh sequences. The more specific theory on meshes with localized symmetry properies and the use of gradient recovery reviewed in §3.1.2 predicts superlinear convergence of $Err^{(P,j)}$ in the interior of Σ_1. On this basis, the presence of a magnified error function for $Err^{(P)}$, at least in the interior of Σ_1 could have been anticipated.

The errors for the digital put option exhibit similar behaviour. In Figures 6 A and B, we show $\psi_{est}^{(V,b)}$ and $\psi_{est}^{(P,b)}$ for this option. $\psi_{est}^{(V,c)}$ and $\psi_{est}^{(P,c)}$ are essentially the same, respectively;

[7]The viewpoint in these figures is from deep in the first quadrant, i.e. from (S_{view}, S_{view}) for large S_{view}, to show the surface features above the double strike price, which would be hidden if the view point were the same as used for Figure 4.





A : $\psi_{est}^{(V,b)}(S)$ B : $\psi_{est}^{(P,b)}(S)$

Figure 6: Put option magnified error function estimates

option	$\min(\psi_{est}^{(V,b)})$	$\max(\psi_{est}^{(V,b)})$	$\min(\psi_{est}^{(P,b)})$	$\min(\psi_{est}^{(P,b)})$	$\|\|\psi_{est}^{(P,b)}\|\|_\infty / \|\|\psi_{est}^{(V,b)}\|\|_\infty$
call	-.0004	.0020	-.0035	.0151	7.5
digital put	-.0024	0	-0152	.0160	6.667

Table 2: Extremes of magnified error functions

they are not displayed. The apparent presence of magnified error functions for the valuation and portfolio on the digital put again empirically indicates quadratic convergence of the weighted errors on Σ_1.

The portfolio errors for the digital put near the boundary between the unstructured mesh and Σ_1 shows an interesting regular behaviour, which is suppressed in Figure 6B by the weighting function. It can be seen in Figure 7, which shows $-(P^{(f)}(S) - P^{(b)}(S))$, i.e. the unweighted, unscaled difference = (basic - fine) mesh portfolio values. This difference is an estimate of the negative of the portfolio error on the basic mesh. Large spikes occur at the truncated ends of the gradient transition zones for the put option discussed above, see (26) for Σ_1. The negative error estimate is displayed to simplify the orientation of the surface for the viewer; i.e. the spikes are upwards. These error spikes induce a sharp gradient in the error surface in Σ_1 near its boundary. This appears to be a form of error boundary layer near this part of the boundary of Σ_1. We recall that the theory of §3.1.2 predicts quadratic convergence of the recovered gradient values only at points sufficiently interior to Σ_1. We speculate that the error behaviour close to the spikes at the ends of the truncated transition zones illustrates that, indeed, practical quadratic convergence does not necessarily extend in Σ_1 uniformly to its boundary.

In Table 2, we list the extreme values of the magnified error functions, estimated using

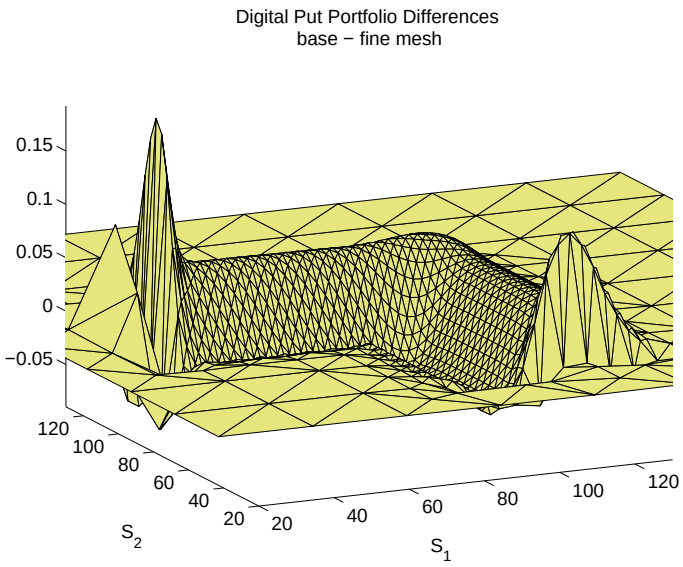


Figure 7: Difference between basic and fine portfolio values displayed on plotting mesh. Note error behaviour at uniform core - unstructured mesh boundary.

the basic mesh, for the two options demonstrated. The table introduces the maximum norm, $||\psi||_\infty \equiv \max_{S\in\Sigma} |\psi(S)|$. If the errors follow the asymptotic behaviour of (24), then $||Err^{(P)}||_\infty \approx ||\psi^{(P)}||_\infty h^2$ and we can expect that for a given mesh of the class with uniform mesh spacing h in Σ_1

$$||Err^{(P)}||_\infty / ||Err^{(V)}||_\infty \approx ||\psi^{(P)}||_\infty / ||\psi^{(V)}||_\infty. \tag{30}$$

Note that the right hand side of (30) is independent of h; so $r_{bal} = ||\psi^{(P)}||_\infty / ||\psi^{(V)}||_\infty$ could be viewed as a ratio quantifying the degree of balance between these errors that can be obtained using this type of mesh. As the table shows, $r_{bal} \approx 7$ for these two options. From the point of view of the size of the computations required to reach a prescribed error tolerance, $errTol$, this ratio implies that if a particular mesh of this type produces $||Err^{(V)}||_\infty$ of size $errTol$, then a mesh that is about 7 times as big is required to ensure $||Err^{(P)}||_\infty \approx errTol$, regardless of the size of $errTol$. This follows because the maximum weighted error is proportional to h^2 and the size of the mesh is proportional to h^{-2}.

5 Summary and Future Directions

We have proposed a simple technique for the design of efficient meshes to support the numerical evaluation in the two factor domain Σ of the Black-Scholes valuation function and risk free portfolio function for two factor options. The computations can be viewed as occurring in two stages; the computation of a pwlinear approximation to the valuation function, and the computation of a pair of pwlinear approximate deltas using gradient recovery (§2). The error in the computed portfolio function has contributions from both the error in the

valuation function and in the deltas, (§2.2). It provides a natural summary of the errors in the two stages of the computation.

The general qualitative premises for the efficiency of unstructured meshing for these computations are that

a) accuracy is not required uniformly throughout Σ, and,

b) within the region where accuracy is required, there are zones where attaining accuracy requires small mesh spacing. We have designated the subdomain comprising these zones by Σ_1.

The meshing technique presented covers Σ_1 with a square grid of vertices spaced h apart horizontally, and a general Delaunay unstructured mesh on $\Sigma - \Sigma_1$.

What is the basis for claiming that this technique is efficient? The common measure of mesh efficiency is the size of the mesh required to meet a given error requirement. The theoretical arguments that are reviewed in §3 are based on the asymptotic behaviour of the valuation and portfolio function errors, and the mesh sizes, for a convergent sequence of meshes $\{M_j\}$. Appendix A presents an argument to support the claim that any method such that the errors converge to zero quadratically will be more efficient than any method with linear convergence[8], for stringent enough error requirements. While quadratic convergence of the valuation function errors is generally assured for arbitrary sequences $\{M_j\}$, only linear convergence of the portfolio function errors is predicted. The theory does predict that superlinear convergence for the portfolio function approximations can be obtained for the interior of subdomains, like Σ_1, which have uniformly distributed vertices, or have similarly symmetric placements of vertices. The significant restrictions on meshes that seem to be required in order to realize quadratic convergence of the approximate portfolio values reflects the general observation that computing approximate derivatives is typically more difficult that computing function approximations in numerical computation.

In §4, we have demonstrated the error behaviour of the valuation and portfolio functions for two types of options on the maximum of two assets, a call and a digital put computed using the meshes of §4. In particular, the computations demonstrate quadratic convergence of the weighted error for both computed functions on the interior of Σ_1 for each option.

These errors show a strong asymptotic form of quadratic convergence in Σ_1. In this form, the pointwise error, $Err(S)$, follows a relatively smooth distribution, $\psi(S)$, referred to as the magnified error functio, and the size of the error is scaled by h^2, i.e. $Err(S) \approx \psi(S)h^2$. §4 provides empirical evidence for the presence of magnified error functions, $\psi^{(V)}$, for the valuation error and $\psi^{(P)}$ for the portfolio error. This is the basis for our claim for quadratic convergence of both these errors in Σ_1 and for the efficiency of this meshing tactic. Of course, it takes a larger mesh to achieve the same level of accuracy in the portfolio function than is required for the valuation function alone. The estimated sizes of the $\psi^{(V)}$ and $\psi^{(P)}$ indicate that the mesh would need to be about 7 times larger, at least for the options that were examined. Note that this multiple is independent of the accuracy sought.

[8] or superlinear convergence with rate $\beta < 1$

5.1 Further Directions

Within the scope of unstructured meshes and piecewise polynomial approximation, our approach to efficient computation of the quantities described above has been a conservative one, based on a single tool, i.e. local quadratic convergence of the hedging parameters. Meshing techniques are highly developed, however and we now comment on how further efficiencies could be sought.

The paper describes construction of static meshes. However, we know that the small mesh spacing zones for option valuations are dynamic, even for the simplest vanilla options. Dynamic meshing for computing the value and portfolio functions for one factor options have been studied by Huen, [7].

Two more aggressive strategies for mesh efficiency, static or dynamic, are the use of anisotropic meshing and adaptive techniques. Both of these strategies are well established for efficient computation of function values, but less well established for efficiency in the computation of recovered gradient values. Our demonstration has been limited to isotropic meshes, i.e. meshes for which the triangles are 'well shaped' so that they have essentially one length scale. As Figures 2B and 3B show, all the triangles in the meshes that we generate have the same shape, i.e. they are isosceles right angle triangles. This class of meshes is discussed further in Simpson, [21]. This triangle shape is efficient if the error associated with a mesh edge does not vary significantly with the direction of the edge. For the option data calculated here, the errors associated with an edge depend strongly on the direction of the edge in some parts of the small mesh spacing zones. In these parts of the mesh, it would be more efficient to use mesh spacings that vary with the directions of error dependence, i.e. anisotropic meshes, see Simpson, [20], George and Borouchaki, [4].

In unstructured meshing research, efficiency has been, and continues to be, sought from adaptive meshing, a large topic that we do not review. See Verfurth, [23] and Morin et al, [12]. For computing solutions of pdes, these techniques usually sense solution features near each mesh vertex and use the flexibility of unstructured meshes to add or remove vertices in the vicinity of the vertex under examination. In the case that the driving solution feature is a measure of local solution error, the method has been extensively studied and demonstrated. The adaptive meshes provide quadratic convergence of the function errors, quite generally. It is less well developed for controlling gradient error. As we have noted, the goal of quadratic convergence of the gradient error places structural as well as size requirements on the mesh. This suggests that the flexibility to make local mesh modifications to control errors is reduced. Buss and Simpson [2] have shown that there are difficulties in devising refinement techniques that are both local and respect structural properties required in the mesh.

A final comment: we have assumed the use of piecewise linear approximation for the computations under discussion. At least for the relatively smooth surfaces of many two-factor options, piecewise quadratic approximations, or higher orders, can be expected to be more efficient.

Appendix A: Efficiency of meshes with gradient quadratic convergence of the portfolio errors

Consider two computational schemes for approximately evaluating the portfolio function, each having its own class of meshes. For one of them, the errors for a convergent sequence of meshes converge quadratically in some measure, i.e. $||Err^{(j)}|| = O(h_j^2)$. The other has superlinear convergence of fractional order $0 \leq \beta < 1$, i.e. $||Err^{(j)}|| = O(h_j^{1+\beta})$. Intuitively, it is probably convincing that the quadratically convergent method will be more efficient that the superlinear convergent one; in this appendix we provide some details for an argument based on asymptotic error behaviour to support this claim. Our discussion relies heavily on standard assumptions of asymptotic discretization error behaviour.

We compare two computational schemes using meshes generated as described in §3 and gradient recovery techniques as described in §3.1.2. We will designate the computational schemes by A and B. For each scheme, we consider a convergent quasi-uniform sequence of meshes generated by quality mesh generation as described in §3, $\{MA_k\}$ for A and $\{MB_k\}$ for B. By virtue of the quality mesh generation techniques,

$$N(MA_k) = O(h(MA_k)^{-2}) \; ; \quad N(MB_k) = O(h(MB_k)^{-2}). \tag{31}$$

We extend the error definitions of §2.2; let $Err^{(\Delta)}(MA_k) = \max_{S \in MA_k}(|Err^{(\Delta)}(S,t)|)$, for the *max* described at (14), and similarly for $Err^{(V)}(MB_k)$. For a summary of the portfolio function error on mesh MA_k, let

$$Err^{(P)}(MA_k) = \max(|Err^{(V)}(S,t)| + |S^T Err^{(\Delta)}(S,t)|).$$

The only explicit difference between schemes A and B is that scheme B is assumed to result in quadratic gradient superconvergence i.e.

$$Err^{(\Delta)}(MB_k) = O(h(MB_k)^2). \tag{32}$$

Scheme A is assumed to result in gradient superconvergence with rate $\beta < 1$ and this rate is sharp , i.e. $Err^{(\Delta)}(MA_k) = O(h(MA_k)^{1+\beta}$, and

$$Err^{(\Delta)}(MA_k)/h(MA_k)^2 \text{ is unbounded as } k \to \infty. \tag{33}$$

We now argue that the theory predicts scheme B to be more efficient than scheme A. (12) implies that

$$Err^{(V)}(MA_k) = O(h(MA_k)^2) \text{ and } Err^{(V)}(MB_k) = O(h(MB_k)^2).$$

From (33), we conclude that the gradient error, $Err^{(\Delta)}(MA_k)$, eventually dominates $Err^{(P)}(MA_k)$ as $k \to \infty$. In fact, combining (31) with (33), we have

$$N(MA_k)/Err^{(P)}(MA_k) \text{ is unbounded as } k \to \infty. \tag{34}$$

However, from (32), we conclude that the gradient error, $Err^{(\Delta)}(MB_k)$, does not dominate $Err^{(P)}(MB_k)$ as $k \to \infty$ and we can expect

$$Err^{(P)}(MB_k) \leq Err^{(V)}(MB_k) + \max(|S|)(Err^{(\Delta)}(MB_k)) = O(h(MB_k)^2)$$

and consequently

$$N(MB_k)/Err^{(P)}(MB_k) \text{ is bounded as } k \to \infty. \tag{35}$$

We continue to assume that the portfolio errors follow the monotone behaviour of the asymptotic error estimates. Let us pick a small error tolerance, $pTol$, for the portfolio error, and let $k_A(pTol)$ and $k_B(pTol)$ be mesh sequence indices such that

$$Err^{(P)}(MA_{k_A(pTol)}) \approx pTol \quad \text{and} \quad Err^{(P)}(MB_{k_B(pTol)}) \approx pTol.$$

Since we have effectively normalized the error size, scheme B will be more efficient than scheme A if $N(MB_{k_B(pTol)}) < N(MA_{k_A(pTol)})$. From (34) and (35), we can conclude that

$$\frac{N(MA_{k_A(pTol)})Err^{(P)}(MB_{k_B(pTol)})}{N(MB_{k_B(pTol)})Err^{(P)}(MA_{k_A(pTol)})} \approx \frac{N(MA_{k_A(pTol)})}{N(MB_{k_B(pTol)})} \tag{36}$$

is unbounded as $pTol \to 0$. Consequently, even allowing for all the unknown constants of this argument, scheme B is more efficient than scheme A for small enough $pTol$ by virtue of the quadratic gradient recovery of scheme B.

Appendix B: The two asset lognormal probability density function

The weighting function, $w(S)$, used in the presentation of error data in §4 is derived from the two asset lognormal probability density function. This probability density function can be expressed using the stochastic differential equation parameters introduced at (3). The following description is patterned on Wilmott, [24], pages 159, and 171. Consider a two asset random process following (3) that starts at $(S_1, S_2) = (K, K)$ at forward time $t_s < 0$, which is some time prior to the issuing time of our options, at $t_{start} = 0$. The probability density function for this process in S space is

$$\phi(S_1, S_2, t) = exp(-z)/(\sigma_1\sigma_2 2\pi(t - t_s)S_1S_2(1 - \rho_{1,2}^2))$$

where

$$\begin{aligned} z &= [\alpha_1,\ \alpha_2]\Sigma^{-1}\begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} \\ &= \alpha_1^2 + \alpha_2^2 - \rho_{1,2}\alpha_1\alpha_2/(1 - \rho_{1,2}^2) \end{aligned}$$

and

$$\alpha_k = (log(K/S_k) + (\mu_k - \sigma_k^2/2)(t - t_s)) \ \text{ for } \ k = 1, 2$$

$$\Sigma = \begin{bmatrix} 1 & \rho_{1,2} \\ \rho_{1,2} & 1 \end{bmatrix} \ ; \ \Sigma^{-1} = \begin{bmatrix} 1 & -\rho_{1,2}/(1 - \rho_{1,2}^2) \\ -\rho_{1,2}/(1 - \rho_{1,2}^2) & 1 \end{bmatrix}.$$

Our weighting function is $\phi(S_1, S_2, 0)$ normalized to 1 at (K, K), i.e.

$$w(S) = \phi(S_1, S_2, 0)/\phi(K, K, 0).$$

A selection of five contours of $w(S)$ at levels, .9, .5, .1, .01, .001, are shown in Figures 2B and 3b , indicating how this function weights the errors. The numerical values of the parameters for $w(S)$ have been chosen for each option to give a somewhat broader density function that would result from the parameters given in §4 for the option calculations. They are listed in the following table.

	σ_k for $k = 1, 2$	μ_k for $k = 1, 2$	$\rho_{1,2}$	$-t_s$ (days)
call	0.3	.04	.25	3
digital put	0.7	.04	.25	9

References

[1] R E Bank and J Xu. Asymptotically exact a posteriori error estimators, part i: Grids with superconvergence. *SIAM J. Numer. Analysis*, 41:2313– 2332, 2003.

[2] J F Buss and R B Simpson. Planar mesh refinement cannot be both local and regular. *Numerische Mathematik*, pages 1–10, 1998.

[3] P A Forsyth, K R Vetzal, and R Zvan. A finite element approach to the pricing of discrete lookbacks with stochastic volatility. *Applied Num. Fin.*, 6:87–106, 1999.

[4] P L George and H Borouchaki. *Delaunay Triangulation and Meshing.* Hermes, 1998.

[5] P Henrici. *Discrete Variable Methods in Ordinary Differential Equations.* John Wiley, New York, 1962.

[6] E Hinton and J S Campbell. Local and global smoothing of discontinuous finite element function using a least squares method. *International Journal for Numerical Methods in Engineering*, 8:461–480, 1974.

[7] S Huen. Non-uniform adaptive meshing for one-asset problems in finance. Master's thesis, University of Waterloo, 2003.

[8] S.R. Idelsohn and E. Oñate. Finite volumes and finite elements:'two good friends'. *International Journal for Numerical Methods in Engineering*, 37:3323–3341, 1994.

[9] H B Keller. *Numerical Methods for Two Point Boundary Value Problems.* Blaisdell, Waltham, Mass., 1968.

[10] A M Lakhany, I Marek, and J R Whiteman. Superconvergence results on mildy structured triangulations. *Computer Methods in Applied Mechanics and Engineering*, 189:1–75, 2000.

[11] X Ma, D Mao, and A Zhou. Extrapolation for finite volume approximations. *SIAM J. Sci. Comput.*, 24:1974–1993, 2003.

[12] P Morin, R H Nochetto, and K G Siebert. Convergence of adaptive finite element methods. *SIAM Review*, 44:631–658, 2002.

[13] K W Morton and D F Mayers. *Numerical Solution of Partial Differential Equations.* Cambridge University Press, Cambridge, 1994.

[14] D M Pooley, P A Forsyth, K R Vetzal, and R B Simpson. Unstructured meshing for two asset barrier options. *Applied Math. Finance*, 7:33–60, 2000.

[15] M. C. Rivara, N. Hitschfeld, and R. B. Simpson. Terminal edges Delaunay (small angle based) algorithm for the quality triangulation problem. *Computer Aided Design*, 33:263–277, 2001.

[16] J Ruppert. A Delaunay refinement algorithm for quality 2-dimensional mesh generation. *J of Algorithms*, 18:548–585, 1995.

[17] A Schatz, V Thomee, and B Wendland. *Mathematical Theory of Finite and Boundary Element Methods*. Birkhauser Verlag, Basel, 1990.

[18] A H Schatz, I.H.Sloan, and L B Wahlbin. Superconvergence in finite element methods and meshes that are locally symmetric with respect to a point. *SIAM Journal of Numerical Analysis*, 33:505–521, 1996.

[19] J R Shewchuk. Triangle: Engineering a 2d quality mesh generator and Delaunay triangulator. In ACM, editor, *First Workshop on Applied Computational Geometry*, pages 124–133. (Philadelphia, Pennsylvania), 1996. http://www-2.cs.cmu.edu/ quake/triangle.html.

[20] R B Simpson. Anisotropic mesh transformations and optimal error control. *Applied Num. Math.*, 14:183–198, 1994.

[21] R B Simpson. How efficient is Delaunay refinement? In Phillipe P Pebay, editor, *Proc. of the 15th Int'l Meshing Roundtable*. Springer, 2006.

[22] R M Stulz. Options on the minimum or the maximum of two risky assets. *J. Fin. Econ.*, 10:161–185, 1982.

[23] R. Verfurth. *A Review of a posteriori error estimation and adaptive refinement techniques*. Wiley, Teubner, New York, 1996.

[24] P Wilmott. *Paul Wilmott on Quantitative Finance*. John Wiley and Sons Ltd, West Sussex, England, 2000.

[25] Z Zhu and N Stokes. A finite element platform ofr pricing path-dependent exotic options. In CSIRO Math & Info. Sciences, editor, *Proc. Quant. Methods in Finance*. (Sydney, Australia), 1999.

[26] R Zvan, P A Forsyth, and K R Vetzal. A finite volume approach for contigent claims valuation. *IMA Journal of Numerical Analysis*, 21:703–731, 2001.

[27] R Zvan, P A Forsyth, and K R Vetzal. Negative coefficients in two factor option pricing models. *Journal of Computational Finance*, 7:37–74, 2003.