

Anisotropic Adaptive Refinement Algorithms for Finite Element Methods

by

Juan Carlos Aguilar Villegas

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
Graduate School of Arts and Science
Department of Mathematics
New York University
September 2000

Approved: _____
Professor Jonathan B. Goodman

© Juan Carlos Aguilar Villegas
All Rights Reserved 2000

For my parents

Acknowledgements

I am very grateful to my advisor, Professor Jonathan Goodman, who did a great job in helping me to find a lot of directions (including directions for mesh refinement, financial help, job). It has been a very enjoyable experience to work and learn from Jonathan. Thanks to Professor Olof Widlund for his useful suggestions on this work.

I want to thank my dear friends Helga Klein, Jorge Maciel, and Martín García for their support.

My family has played always an important role in my studies, I thank my parents Ma. del Carmen and Máximo, my brothers Raúl, Arturo, Javier, and Pedro, my sisters Angélica, Yolanda, Verónica, Virginia, Angeles, Teresa, and Sandra.

Thank you all!

Contents

Dedication	iv
Acknowledgements	v
List of figures	vii
1 Introduction	1
2 Anisotropic error estimates and algorithms for the Poisson equation	4
2.1 Notation	4
2.2 The error reduction approach	5
2.3 An anisotropic a posteriori error estimator	6
2.4 Shape of triangulations	11
2.4.1 Local quasi-uniform meshes	12
2.4.2 Triangles with only one small angle	12
2.4.3 Triangles with one angle close to π	12
2.5 The estimator for a 1-D model problem	14
2.6 An anisotropic mesh refinement algorithm in 2-D	18
2.6.1 The case of two marked edges	19
2.6.2 An anisotropic refinement algorithm	21
3 An optimal mesh refinement algorithm for an interpolation problem	24
3.1 Introduction	24

3.2	The model problem	25
3.3	Regular mesh refinement	26
3.4	Anisotropic refinement	27
3.4.1	An anisotropic refinement algorithm	27
3.4.2	Analysis of the anisotropic refinement algorithm	29
4	Numerical examples	36
4.1	An example using $\widehat{\Delta e^2}_{W_4}$	40
4.2	An example with two layers that intersect	45
	Bibliography	51

List of Figures

2.1	Adding node F reduces the error more than adding node G	6
2.2	Triangles used in Theorem 2.2	9
2.3	(a) Testing a new node in the short edge gives $\Delta e^2 / \widehat{\Delta e^2}_{w_0} \approx 1$. (b) When the triangles have only one small angle, $\Delta e^2 / \widehat{\Delta e^2}_{w_0}$ has a bound independent of the aspect ratio.	13
2.4	(a) The angle C can be close to π but $\ \overline{AC}\ / \ \overline{AB}\ $ should be at most $1/4$ to have $\Delta e^2 / \widehat{\Delta e^2}_{w_0} \leq 5$. (b) Theorem 2.2 is inconclusive when the angle C is $\approx \pi$ and $\ \overline{AC}\ / \ \overline{CB}\ \approx 1$. (c) An example of a triangulation where $\Delta e^2 / \widehat{\Delta e^2}_{w_0} \rightarrow \infty$ as $h/\delta \rightarrow \infty$. In this example $\Delta e^2 / \widehat{\Delta e^2}_{w_4}$ remains bounded.	14
2.5	A function with an interior layer of width 2ϵ at $w \in (x_i, x_{i+1})$. (a) The new node \tilde{x}_i satisfies $ \tilde{x}_i - w > \epsilon$. (b) The new node \tilde{x}_i satisfies $ \tilde{x}_i - w < \epsilon$	15
2.6	Initial mesh, refined mesh, and magnification of the refined mesh around the layer located at $x = 0.4$. The exact solution is $u(x) = \sin(\pi x) \tanh((x - 0.4)/0.005)$. For the refined mesh $\frac{\sum_{i=0}^{66} \Delta e_i^2}{\ u - u_V\ _a^2} = 0.7361$	18
2.7	(a) The triangle ABC has two edges marked for refinement. (b) The first alternative of refinement consists of adding first node F followed by G . (c) In the second alternative the node G is added first followed by F	19
2.8	(a) Triangle ABC in current mesh τ has two edges marked for refinement. (b) Adding node F to τ produces a mesh $\tilde{\tau}$. (c) Adding node G to $\tilde{\tau}$ produces a mesh $\tilde{\tilde{\tau}}$	20

2.9	Refinement of a triangle for the case of 1, 2, or 3 marked edges. For the case of two marked edges we choose one of two alternatives (see Section 2.6.1).	23
2.10	When a triangle has a boundary edge \overline{AB} and a marked interior edge \overline{BC} , it is refined by choosing one of two alternatives.	23
3.1	Type of refinement for the anisotropic refinement algorithm. The triangles in quadrilateral (a) are replaced by the ones in (b).	28
3.2	Construction of the sequences $\{S_n\}$, $\{L_n\}$, $\{K_n\}$	29
3.3	Angle α_n in the proof of Lemma 3.4.1	30
3.4	The trapezoid in the $n - th$ refinement step.	31
3.5	Initial mesh and three refinement steps when σ is a circle.	33
3.6	The arc σ_{AB} is enclosed by an arc of a circle and a line segment \overline{AB} , as stated in Lemma 3.4.3	34
3.7	Construction used to show that the quadrilateral $Q_1Q_2Q_3Q_4$ encloses the arc σ_{AB}	35
3.8	The arc σ_{AB} , with curvature bounded by $1/R$, is enclosed between two arcs of circles of radii R	35
4.1	Initial grid with 25 degrees of freedom, refined grid with 1780 degrees of freedom using $\widehat{\Delta e^2}_{W_0}$, and blow up of the refined grid. The exact solution is $\sin(\pi x) \sin(\pi y) \tanh((r - 0.5)/\epsilon)$, where $r = \sqrt{(x - 0.1)^2 + (y - 0.1)^2}$ and $\epsilon = 0.005$. The refined mesh contains triangles with aspect ratio up to 137 and $1 \leq \Delta e^2 / \widehat{\Delta e^2}_{W_0} \leq 3.9$, $1 \leq \Delta e^2 / \widehat{\Delta e^2}_{W_4} \leq 1.7$ for all its interior edges.	38
4.2	Same as Fig. 4.1 but using $\widehat{\Delta e^2}_{W_4}$. The final mesh contains 1700 degrees of freedom.	39
4.3	First meshes using $\widehat{\Delta e^2}_{W_4}$	40
4.4	Next four refinements.	41
4.5	Next refinements, the last one with 1700 degrees of freedom. Also blow up of the last mesh	42
4.6	Blow up of the last mesh.	43

4.7	Blow up of the initial mesh and the first three refinements	44
4.8	Blow up of the next four refinements.	45
4.9	First meshes for the example with two layers	46
4.10	Next four refinements	47
4.11	Next four refinements. The last one has 2010 degrees of freedom . .	48
4.12	Blow up of the last mesh using the estimator $\widehat{\Delta e^2}_{W_4}$	49
4.13	The estimator $\widehat{\Delta e^2}_{W_0}$ produced the mesh shown with 1820 degrees of freedom. The other pictures are magnifications of this mesh . . .	50

Chapter 1

Introduction

This thesis is on anisotropic adaptive finite element methods for numerical solution of partial differential equations. Adaptive finite element methods place more fine scale elements where more resolution is needed. Isotropic or shape regular adaptive methods use only elements with bounded aspect ratio (stretched elements are avoided). Anisotropic adaptive methods fit high aspect ratio elements (highly stretched elements) along the regions of rapid variation of the solution for situations like shocks or boundary layers. Anisotropic adaptive methods give a bigger saving in terms of computational cost (number of elements and degrees of freedom) than the isotropic ones if stretched elements are placed appropriately. This research is about error estimates and adaptive grid generation allowing the use of anisotropic (stretched) elements.

There are two main aspects in an adaptive method: error estimation and grid generation (see Babuska [7], Johnson [20], Zienkiewicz [14], among others). Most of the theory currently available is restricted to the isotropic case, but see [1], [2], [19], [21], [23], [25], [27]. One reason that makes anisotropic refinements difficult is that placing high aspect ratio elements in the wrong direction may cause loss of convergence (see [6]). On the other hand, the amount of computing time and memory that a good anisotropic algorithm may save compared with a good isotropic algorithm (in problems with anisotropies) is the main reason that motivated this thesis.

Grid generation algorithms for isotropic refinements are designed to maintain the aspect ratio of the new elements below certain bound, see the algorithm of Bank [9]. In 2-D, for triangular meshes, some anisotropic grid generators are designed to keep the maximum angle of all elements below a bound smaller than π (see [6]). This usually results in refinement algorithms that do not produce a sequence of nested finite element spaces, which is a desirable property for error estimation and for exploitation of methods like multigrid to solve the associated system of equations in an efficient way. The algorithm presented in Chapter 2 generates triangular grids without any restriction on the angles of the elements and it produces a sequence of nested finite element spaces.

There are two main difficulties that an error estimator for anisotropic methods has to handle: (i) grids are not shape regular (ii) the estimator should in addition to providing information about location also provide information on direction of stretching in order to improve the current grid. Anisotropic refinements using rectangular elements are of great advantage when there are two directions of anisotropy (see [23], [25]); triangular elements give more geometric flexibility for more complicated situations. In this thesis we use only triangular elements.

Most error estimates for both isotropic or anisotropic refinements try to estimate the local contributions from each element to the global error. The estimators used in this thesis are constructed by trying to answer the following question : how much is the error reduced if we add a single degree of freedom to the current finite element space? Such an estimator for isotropic refinement has been proposed by Zienkiewicz (see [14]). In this thesis we prove a theorem about the effectivity of the estimator in [14] for local quasi-uniform meshes, and we give some results for more general meshes. We give more robust estimators for the error reduction in Section 2.3.

In Chapter 2 we obtain error estimates for anisotropic refinement in 2-D using

triangular elements. We construct an anisotropic refinement algorithm based on the error reduction estimators. Although for 1-D problems there is no anisotropic refinement, several aspects of our 2-D anisotropic refinement algorithm have been motivated from applying the error reduction estimators to a 1-D model problem. We present 1-D numerical experiments (see Chapter 4 for 2-D numerical tests). Different approaches can be found in the work of Goodman, Samuelsson, and Szepessy [19], and for rectangular elements in Rackowitz [23], Siebert [25].

The applications of anisotropic refinements goes beyond finite elements; as we will see in Chapter 3, there are some other problems where an appropriate anisotropic refinement strategy can produce optimal results in terms of number of triangles needed to achieve a given accuracy. In the model problem that we analyse, we prove that using N triangles the isotropic or regular refinement produces an error of approximation of order $1/N$, while the proposed anisotropic refinement algorithm gives an error of order $1/N^2$.

Chapter 2

Anisotropic error estimates and algorithms for the Poisson equation

In this chapter we analyse an error estimator which will be used in the design of anisotropic refinement algorithms. What makes this estimator different from most error estimators is that this one estimates the change of the square of the error when a single degree of freedom is introduced to the current approximating space. Another difference is the flexibility in the geometry of meshes allowed by this estimator (see Section 2.4). The estimators given in this chapter are for 2-D problems but they can be extended to 3-D. Some numerical experiments are presented in Chapter 4 to test the performance of the error estimators.

2.1 Notation

We will consider here the model problem

$$\begin{aligned} -\Delta u &= f && \text{in } \Omega, \\ u &= 0 && \text{on } \partial\Omega, \end{aligned} \tag{2.1}$$

where Ω is the unit square and the data are given such that the solution u and some of its derivatives change quickly across an interior layer. Approximate solutions of

the model problem will be obtained by the finite element method using triangular elements and continuous, piecewise linear functions.

In the usual notation of finite elements (see [11], [20]), $a(v, w)$ is the bilinear form defining the PDE, $\langle f, v \rangle$ is the L^2 inner product, and $\|v\|_a^2 = a(v, v)$. For the model problem (2.1), we have $a(v, w) = \int_{\Omega} \nabla v \cdot \nabla w dx dy$, for all $v, w \in H_0^1(\Omega)$, and (2.1) can be reformulated as: find $u \in H_0^1(\Omega)$ such that

$$a(u, v) = \langle f, v \rangle \quad \text{for all } v \in H_0^1(\Omega). \quad (2.2)$$

For the standard finite element formulation of Equation (2.2), we will need a *triangulation* of Ω , which is a collection τ of triangles contained in $\overline{\Omega}$ such that $\overline{\Omega} = \bigcup_{K \in \tau} K$ and where for any two triangles in τ one of the 3 possibilities holds: a) their intersection is empty, b) their intersection is a vertex of both triangles, c) they have an edge in common. The finite element space $V \subset H_0^1(\Omega)$ will be the space spanned by the *nodal basis functions* v_1, \dots, v_N which are defined as continuous, piecewise affine functions such that

$$v_i(b_j) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise,} \end{cases} \quad (2.3)$$

where b_1, \dots, b_N are the interior vertices of the triangulation τ . The finite element discretization of (2.2) is then: find $u_V \in V$ such that

$$a(u_V, v_j) = \langle f, v_j \rangle \quad \text{for } j = 1, \dots, N. \quad (2.4)$$

2.2 The error reduction approach

As we mentioned before, an anisotropic adaptive method requires that an error estimator should provide information about location and direction of stretching in order to refine the current mesh. This information is used to obtain a new mesh that is better adapted to the solution, which reduces the computational cost. The approach to obtain the anisotropic error estimator we use in this thesis is based on

the following question: how much is the error reduced if a single degree of freedom is added to the current approximating space V ? We want to add the degrees of freedom that produce large error reduction (Fig. 2.1). This idea is similar to a matching pursuits algorithm [22]: from a collection of candidate functions to be added to V , we select the ones that best match the difference $u - u_V$ (with respect to the bilinear form a), where u is the exact solution and u_V is the finite element approximation. Zienkiewicz proposed that we could get an approximate answer by a local computation [14]. In this thesis we prove (in Theorem 2.2) the effectivity of the estimator in [14] for local quasi-uniform meshes, and we give some results for more general meshes. More robust estimators for the error reduction are given in Section 2.3.

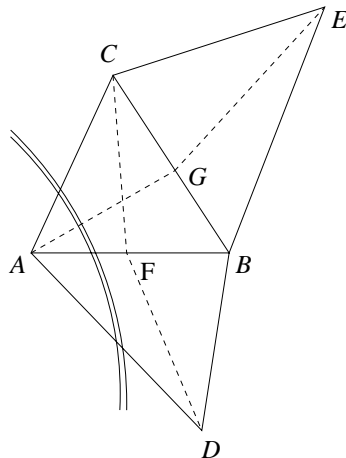


Figure 2.1: Adding node F reduces the error more than adding node G .

2.3 An anisotropic a posteriori error estimator

In this section we obtain an estimator of the reduction of the error when a single degree of freedom is added to the current triangulation. The finite element space consists of continuous piecewise linear functions. Let τ be the current grid with approximating space V and let $\tilde{\tau}$ be a refinement of τ obtained by bisecting an interior edge of τ (see Fig. 2.1) where the interior edge \overline{AB} is bisected), then the

associated finite element space \tilde{V} contains exactly one more degree of freedom than V , say $\tilde{V} = V + \text{span}\{\varphi\}$, where φ is 0 on all nodes of τ and 1 on the new node (see Fig. 2.1) where F is the new node). We want to estimate the error reduction

$$\Delta e^2 = \|u - u_V\|_a^2 - \|u - u_{\tilde{V}}\|_a^2, \quad (2.5)$$

where u is the exact solution, u_V and $u_{\tilde{V}}$ are the finite element solutions on the triangulations τ and $\tilde{\tau}$ respectively. The only solution available is u_V . We want an estimator, $\widehat{\Delta e^2}$, that is of the same order of magnitude and cheap to compute. The following lemma motivates the error estimator we use:

Lemma 2.1

$$\Delta e^2 = \frac{|\langle f, \varphi \rangle - a(u_V, \varphi)|^2}{\inf_{v \in V} \|v - \varphi\|_a^2}. \quad (2.6)$$

PROOF: Using the facts that $V \subset \tilde{V}$ and $(u - u_{\tilde{V}})$ is orthogonal to \tilde{V} (with respect to the bilinear form a), it follows that $(u - u_{\tilde{V}})$ is orthogonal to $(u_V - u_{\tilde{V}})$ and thus

$$\|u - u_V\|_a^2 = \|u - u_{\tilde{V}}\|_a^2 + \|u_V - u_{\tilde{V}}\|_a^2,$$

which means from Equation (2.5) that the error reduction can be expressed as

$$\Delta e^2 = \|u_V - u_{\tilde{V}}\|_a^2. \quad (2.7)$$

On the other hand,

$$\begin{aligned} \|u_V - u_{\tilde{V}}\|_a^2 &= \sup_{w \in \tilde{V}} \frac{|a(u_V - u_{\tilde{V}}, w)|^2}{\|w\|_a^2} \\ &= \sup_{w \in \tilde{V}} \frac{|a(u - u_{\tilde{V}}, w) - a(u - u_V, w)|^2}{\|w\|_a^2} \\ &= \sup_{w \in \tilde{V}} \frac{|a(u - u_V, w)|^2}{\|w\|_a^2}. \end{aligned}$$

Any $w \in \tilde{V}$ can be written as $w = v - \alpha\varphi$ with $v \in V$ and $\alpha \in \mathbb{R}$.

Thus

$$\begin{aligned}
\| u_V - u_{\tilde{V}} \|_a^2 &= \sup_{v \in V, \alpha \in \mathbb{R} - \{0\}} \frac{|a(u - u_V, v - \alpha\varphi)|^2}{\| v - \alpha\varphi \|_a^2} \\
&= \sup_{v \in V, \alpha \in \mathbb{R} - \{0\}} \frac{\alpha^2 |a(u - u_V, \varphi)|^2}{\alpha^2 \| v/\alpha - \varphi \|_a^2} \\
&= \sup_{v \in V} \frac{|a(u - u_V, \varphi)|^2}{\| v - \varphi \|_a^2} \\
&= \sup_{v \in V} \frac{|\langle f, \varphi \rangle - a(u_V, \varphi)|^2}{\| v - \varphi \|_a^2}.
\end{aligned} \tag{2.8}$$

From (2.7) and (2.8) we obtain Equation (2.6).

The denominator of (2.6) is the a norm distance from φ to V . Computing v requires solving a system of equations involving the stiffness matrix for V . By replacing V in (2.6) by a low dimension linear subspace $W \subset V$, we get an approximation

$$\Delta e^2 \approx \widehat{\Delta e^2}_W = \frac{|\langle f, \varphi \rangle - a(u_V, \varphi)|^2}{\inf_{v \in W} \| v - \varphi \|_a^2}. \tag{2.9}$$

We will analyse the estimator $\widehat{\Delta e^2}_W$ for the cases $W = W_0 = \{0\}$ (the Zienkiewicz estimator [14]) and $W = W_4$ = the space spanned by the nodal basis functions in V surrounding the edge under analysis (the nodes A, B, C , and D in Fig. 2.1).

An estimator $\widehat{\Delta e^2}_W$ is effective if Δe^2 and $\widehat{\Delta e^2}_W$ are of the same order of magnitude and it is cheap to compute. For us “cheap to compute” means $O(1)$ computation, and “same order of magnitude” means that the constants C_1 and C_2 with $\Delta e^2 \leq C_1 \widehat{\Delta e^2}_W \leq C_2 \Delta e^2$, depend only on the geometric constraints. In Theorem 2.2 we show that both estimators $\widehat{\Delta e^2}_{W_0}$ and $\widehat{\Delta e^2}_{W_4}$ are effective for shape regular triangulations and for triangulations containing high aspect ratio elements with some geometric constraints.

From $W_0 \subset W_4 \subset V$, it follows that $\widehat{\Delta e^2}_{W_0} \leq \widehat{\Delta e^2}_{W_4} \leq \Delta e^2$. In Section 2.4.3 we construct an example of a mesh where $\Delta e^2 / \widehat{\Delta e^2}_{W_0}$ is very big while $\Delta e^2 / \widehat{\Delta e^2}_{W_4}$ remains low. In practice, we found useful both estimators for the meshes generated

by the refinement algorithm of Section 2.6.2. See Chapter 4 for a numerical experiments.

The next result establishes the effectivity of the estimator $\widehat{\Delta e^2}_{W_0}$. The notation is as follows: given the triangle ABC , its height through C will be denoted by $\overline{CC'}$ (see Fig. 2.2); thus $\overline{CC'}$ is perpendicular to \overline{AB} and C' belongs to the line through A and B . F is the middle point of \overline{AB} , and the triangle ABC is divided obtaining the triangles AFC and FBC ; similarly the triangle ABD produces the triangles AFD and FBD . The length of the line segment \overline{AB} will be denoted by $\|\overline{AB}\|$.

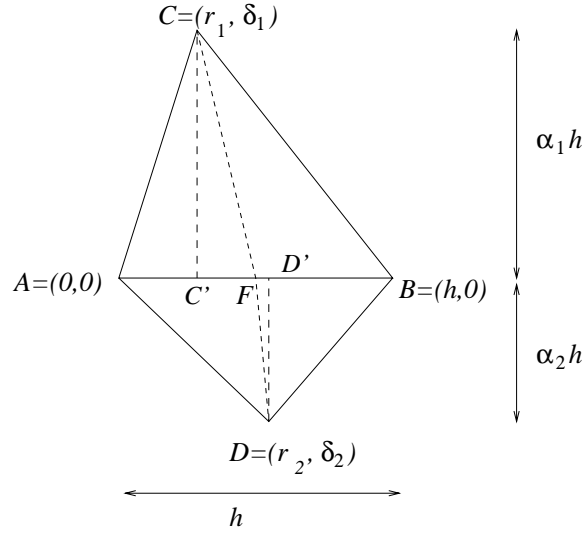


Figure 2.2: Triangles used in Theorem 2.2

Theorem 2.2 For any triangulation τ and any interior edge \overline{AB} shared by the triangles ABC and ABD , with F the middle point of \overline{AB} , $h = \|\overline{AB}\|$, $\overline{CC'}$ and $\overline{DD'}$ the heights through C and D respectively, with $\alpha_1 = \|\overline{CC'}\| / h$, and $\alpha_2 = \|\overline{DD'}\| / h$, then

$$\widehat{\Delta e^2}_{W_0} \leq \Delta e^2 \leq \left(1 + \frac{1}{4\alpha_1\alpha_2}\right) \widehat{\Delta e^2}_{W_0}. \quad (2.10)$$

If $\|\overline{C'F}\| \geq \beta h$ and $\|\overline{D'F}\| \geq \beta h$ then

$$\widehat{\Delta e^2}_{W_0} \leq \Delta e^2 \leq \left(1 + \frac{1}{4(\beta^2 + \alpha_1\alpha_2)}\right) \widehat{\Delta e^2}_{W_0}, \quad (2.11)$$

for any value of $\alpha_1\alpha_2 > 0$.

PROOF: If W is a linear subspace of V , then from (2.6) and (2.9)

$$\frac{\Delta e^2}{\widehat{\Delta e^2}_W} = \frac{\inf_{w \in W} \|w - \varphi\|_a^2}{\inf_{v \in V} \|v - \varphi\|_a^2}. \quad (2.12)$$

By replacing W by $W_0 = \{0\}$ in (2.12), it follows that

$$\frac{\Delta e^2}{\widehat{\Delta e^2}_{W_0}} = \frac{\|\varphi\|_a^2}{\inf_{v \in V} \|v - \varphi\|_a^2}. \quad (2.13)$$

Therefore, it suffices to show that

$$\inf_{v \in V} \|v - \varphi\|_a^2 \geq \frac{1}{M} \|\varphi\|_a^2, \quad \text{where } M = 1 + \frac{1}{4(\beta^2 + \alpha_1\alpha_2)}. \quad (2.14)$$

Denote by K_1 and K_2 the triangles ABC and ABD respectively. Without losing generality, assume that $A = (0, 0)$, $B = (h, 0)$, $C = (r_1, \delta_1)$, and $D = (r_2, \delta_2)$. Thus $\alpha_1, \alpha_2 > 0$ satisfy $|\delta_1| = \alpha_1 h$, $|\delta_2| = \alpha_2 h$. As before V is the finite element space associated to the current triangulation τ , and $\tilde{\tau}$ is the triangulation obtained by bisecting the interior edge \overline{AB} of τ . Let \tilde{V} be the finite element space associated with $\tilde{\tau}$. We can write $\tilde{V} = V + \text{span}\{\varphi\}$ where φ is the new nodal basis function corresponding to the vertex $F = (A + B)/2$; thus $\varphi(F) = 1$ and $\varphi(b_i) = 0$ for any node b_i in τ . For any $v \in V$ we have

$$\|v - \varphi\|_a^2 = \sum_{K \in \tau} \int_K |\nabla(v - \varphi)|^2 dx dy \geq \int_{K_1 \cup K_2} |\nabla(v - \varphi)|^2 dx dy. \quad (2.15)$$

Introducing the space $\Theta = \{p : K_1 \cup K_2 \rightarrow \mathbb{R} \mid p \text{ continuous and } p|_{K_1}, p|_{K_2} \text{ affine}\}$, we obtain from (2.15) that

$$\inf_{v \in V} \|v - \varphi\|_a^2 \geq \inf_{p \in \Theta} \int_{K_1 \cup K_2} |\nabla(p - \varphi)|^2 dx dy. \quad (2.16)$$

Some calculations show that $\tilde{p} \in \Theta$ determined by

$$\tilde{p}(A) = \tilde{p}(B) = 1, \quad \text{and} \quad \tilde{p}(C) = \tilde{p}(D) = 0,$$

minimizes the right hand side of inequality (2.16), that is,

$$\inf_{p \in \Theta} \int_{K_1 \cup K_2} |\nabla(p - \varphi)|^2 dx dy = \int_{K_1 \cup K_2} |\nabla(\tilde{p} - \varphi)|^2 dx dy. \quad (2.17)$$

Now,

$$\begin{aligned} \int_{K_1 \cup K_2} |\nabla(\tilde{p} - \varphi)|^2 dx dy &= \frac{h}{2|\delta_1|} \left(1 - \frac{2r_1}{h}\right)^2 + \frac{2(|\delta_1| + |\delta_2|)}{h} \\ &\quad + \frac{h}{2|\delta_2|} \left(1 - \frac{2r_2}{h}\right)^2, \end{aligned} \quad (2.18)$$

and

$$\begin{aligned} \|\varphi\|_a^2 &= \frac{h}{2|\delta_1|} \left(\left(1 - \frac{2r_1}{h}\right)^2 + 1 \right) + \frac{2(|\delta_1| + |\delta_2|)}{h} \\ &\quad + \frac{h}{2|\delta_2|} \left(\left(1 - \frac{2r_2}{h}\right)^2 + 1 \right) \\ &= \int_{K_1 \cup K_2} |\nabla(\tilde{p} - \varphi)|^2 dx dy + \frac{h}{2} \left(\frac{1}{|\delta_1|} + \frac{1}{|\delta_2|} \right). \end{aligned} \quad (2.19)$$

Therefore, from (2.16), (2.17) and (2.19)

$$\begin{aligned} \frac{\|\varphi\|_a^2}{\inf_{v \in V} \|v - \varphi\|_a^2} &\leq \frac{\|\varphi\|_a^2}{\int_{K_1 \cup K_2} |\nabla(\tilde{p} - \varphi)|^2 dx dy} \\ &\leq 1 + \left(\frac{h}{2}\right) \left(\frac{1/|\delta_1| + 1/|\delta_2|}{\int_{K_1 \cup K_2} |\nabla(\tilde{p} - \varphi)|^2 dx dy} \right). \end{aligned} \quad (2.20)$$

If $\|\overline{C'F}\| \geq \beta h$ and $\|\overline{D'F}\| \geq \beta h$, then $|r_1 - h/2| \geq \beta h$ and $|r_2 - h/2| \geq \beta h$, which implies that $|1 - 2r_1/h| \geq 2\beta$ and $|1 - 2r_2/h| \geq 2\beta$. Using the last two inequalities and the relations $|\delta_1| = \alpha_1 h$, $|\delta_2| = \alpha_2 h$, one obtains from (2.18)

$$\int_{K_1 \cup K_2} |\nabla(\tilde{p} - \varphi)|^2 dx dy \geq \frac{2\beta_1^2}{\alpha_1} + 2(\alpha_1 + \alpha_2) + \frac{2\beta_2^2}{\alpha_2}. \quad (2.21)$$

From (2.21) and (2.20) we conclude that

$$\frac{\|\varphi\|_a^2}{\inf_{v \in V} \|v - \varphi\|_a^2} \leq 1 + \frac{1}{4(\beta^2 + \alpha_1 \alpha_2)}. \quad (2.22)$$

This proves (2.11). Inequality (2.10) follows by setting $\beta = 0$ in (2.11).

2.4 Shape of triangulations

In this section we analyse the shape of triangles that can be exploited using Theorem 2.2. Let us assume that the triangles ABC and ABD of Theorem 2.2 belong to

a mesh τ and that a new node F is tested in the middle of the edge \overline{AB} . The triangles are divided into three groups according to the size of their interior angles:

- a) Triangles in local quasi-uniform meshes
- b) Triangles with only one small angle
- c) Triangles with one angle close to π .

2.4.1 Local quasi-uniform meshes

For local quasi-uniform meshes, any pair of triangles ABC and ABD as in Theorem 2.2 have the property that $\alpha_1 \approx 1$ and $\alpha_2 \approx 1$. Thus in this case we get from (2.10) the estimate

$$\widehat{\Delta e^2}_{W_0} \leq \Delta e^2 \leq C \widehat{\Delta e^2}_{W_0},$$

with $C \approx 1.25$.

2.4.2 Triangles with only one small angle

When the triangles ABC and ABD have only one small angle, Theorem 2.2 says that $\widehat{\Delta e^2}_{W_0}$ approximates Δe^2 well. This can be seen as follows:

- 1) If the new testing node is placed on the short edge, clearly $\alpha_1 \alpha_2 \gg 1$ which implies from Theorem 2.2 that $\Delta e^2 / \widehat{\Delta e^2}_{W_0}$ is close to 1 (see Fig. 2.3a).
- 2) When the new testing node is in one of the long sides, then β as defined in Theorem 2.2 is close to $1/2$, which implies that $\Delta e^2 / \widehat{\Delta e^2}_{W_0} \leq 2$, regardless of how close $\alpha_1 \alpha_2$ is to 0 (Fig. 2.3b).

2.4.3 Triangles with one angle close to π

When the triangles ABC and ABD have one angle close to π , Theorem 2.2 gives a small or big bound for $\Delta e^2 / \widehat{\Delta e^2}_{W_0}$ depending on the shape of the triangles. The parameter β allow us to establish the cases for which the estimate given in Theorem 2.2 is inconclusive. One example of small bound is when $\beta \geq 1/4$ as in Fig. 2.4a where the angle C can be close to π but $\|\overline{AC}\| / \|\overline{AB}\|$ should be close to $1/4$ to have $\Delta e^2 / \widehat{\Delta e^2}_{W_0} \leq 5$. An example of a big bound is when both β and $\alpha_1 \alpha_2$ are

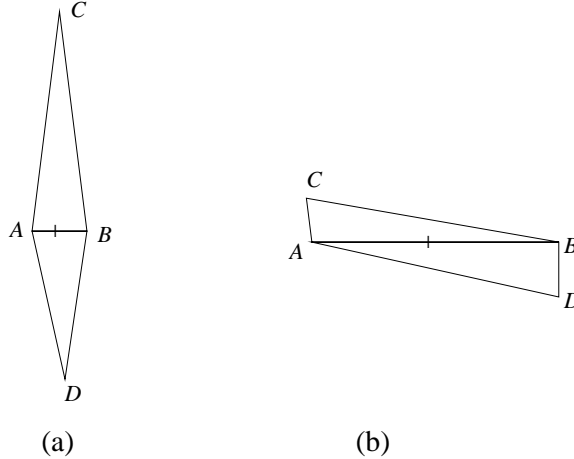


Figure 2.3: (a) Testing a new node in the short edge gives $\Delta e^2 / \widehat{\Delta e^2}_{W_0} \approx 1$. (b) When the triangles have only one small angle, $\Delta e^2 / \widehat{\Delta e^2}_{W_0}$ has a bound independent of the aspect ratio.

very close to 0 (see Fig. 2.4b where the new testing node is placed at the middle of the long side of a triangle that has two other sides of length $\approx 1/2$ of the long side) since in this case the bound $1 + 1/(4(\beta^2 + \alpha_1\alpha_2))$ becomes much greater than 1. One reason for this is that Theorem 2.2 takes into account only the 2 triangles that share the edge where a new node is being tested, without any information on the rest of the mesh τ . For this case it is possible to construct a triangulation τ where $\Delta e^2 / \widehat{\Delta e^2}_{W_0} \rightarrow \infty$ as the aspect ratio $\rightarrow \infty$. An example of this is a mesh containing eight triangles in Fig. 2.4c, where the six surrounding triangles are almost equilateral and the aspect ratio $h/\delta \rightarrow \infty$. For this same mesh of Fig. 2.4c, it can be shown that there is a constant C independent of h and δ such that $1 \leq \Delta e^2 / \widehat{\Delta e^2}_{W_4} \leq C$. On the other hand numerical experiments suggest that our refinement algorithm of Section 2.6.2 generate meshes containing triangles with very high aspect ratio, including triangles for which Theorem 2.2 is inconclusive (Fig. 2.4b), and still $\Delta e^2 / \widehat{\Delta e^2}_{W_0}$ remains low (less than 6) and $\Delta e^2 / \widehat{\Delta e^2}_{W_4}$ remains less than 2. For numerical experiments see Chapter 4.

In addition, numerous numerical experiments suggest that our refinement algorithms generate meshes containing triangles with very high aspect ratio, including triangles for which Theorem 2.2 is inconclusive (Fig.2.4 b), and still $\Delta e^2 / \widehat{\Delta e^2}_{W_4}$

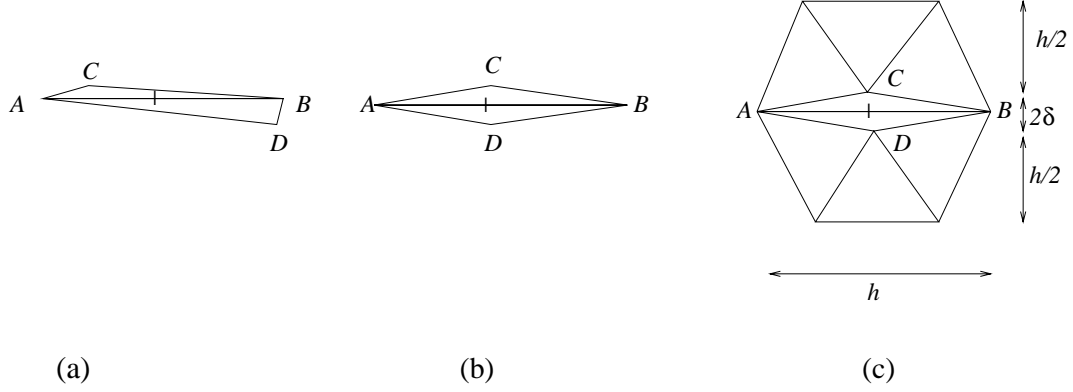


Figure 2.4: (a) The angle C can be close to π but $\|\overline{AC}\|/\|\overline{AB}\|$ should be at most $1/4$ to have $\Delta e^2/\widehat{\Delta e^2}_{W_0} \leq 5$. (b) Theorem 2.2 is inconclusive when the angle C is $\approx \pi$ and $\|\overline{AC}\|/\|\overline{CB}\| \approx 1$. (c) An example of a triangulation where $\Delta e^2/\widehat{\Delta e^2}_{W_0} \rightarrow \infty$ as $h/\delta \rightarrow \infty$. In this example $\Delta e^2/\widehat{\Delta e^2}_{W_4}$ remains bounded.

remains very low (at most 3). For numerical experiments see Chapter 4.

2.5 The estimator for a 1-D model problem

Although for 1-D problems there is no analogue of elements with high aspect ratio, the error reduction indicator $\widehat{\Delta e^2}_{W_0}$ makes sense in this case. Moreover, we can already see a possible complication of the error reduction approach.

For the 1-D problem:

$$\begin{aligned} -u'' &= f \text{ on } (0, 1), \\ u(0) &= u(1) = 0, \end{aligned}$$

we have

$$a(u, v) = \int_0^1 u'v' dx \text{ for all } u, v \in H_0^1(0, 1).$$

Let T_0 be an initial partition of $(0, 1)$ with nodes $x_1 < x_2 < \dots < x_N$ and let $x_0 = 0, x_{N+1} = 1$. Denote by $\varphi_1, \varphi_2, \dots, \varphi_N$ the corresponding continuous and

piecewise linear basis functions. Consider a point $\tilde{x}_i \in (x_i, x_{i+1})$, for $i = 0, \dots, N$, and let $\tilde{\varphi}_i$ be the nodal basis function associated to the partition $\tilde{T}_0^i = T_0 + \{\tilde{x}_i\}$ and such that $\tilde{\varphi}_i(\tilde{x}_i) = 1$. Let V and \tilde{V}_i be the finite element spaces associated to the partitions T_0 and \tilde{T}_0^i respectively. A calculation shows that $\tilde{\varphi}_i \in V^\perp$ and thus, from Equation (2.6) we obtain that for the 1-D model problem the estimator $\widehat{\Delta e^2}_{W_0}$ is exactly Δe^2 . Denoting by Δe_i^2 the error reduction Δe^2 when introducing the node \tilde{x}_i to T_0 , we have

$$\Delta e_i^2 = \frac{|\langle f, \tilde{\varphi}_i \rangle|^2}{\|\tilde{\varphi}_i\|_a^2}.$$

Consider now the situation where the solution u has an interior layer around $w \in (x_i, x_{i+1})$ of width 2ϵ and that has values close to u_L and u_R outside but near the layer. Consider two cases (see Figs. 2.5a and 2.5b).

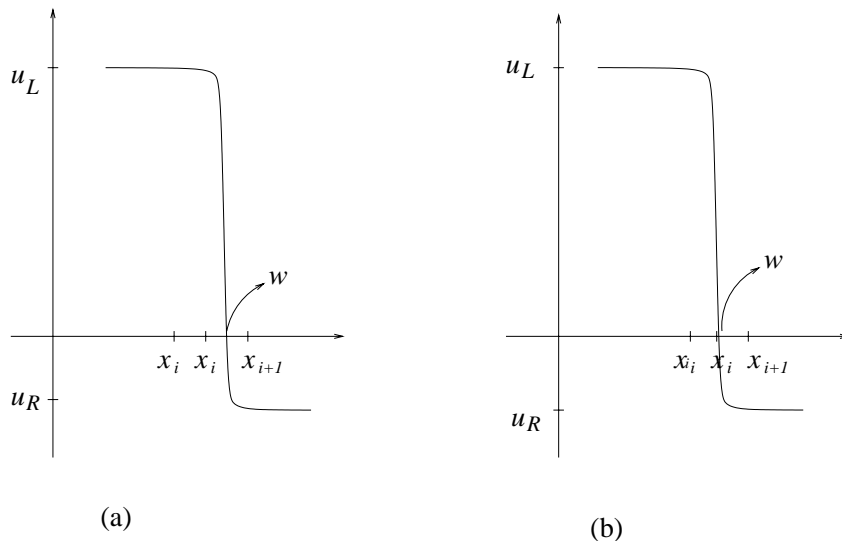


Figure 2.5: A function with an interior layer of width 2ϵ at $w \in (x_i, x_{i+1})$. (a) The new node \tilde{x}_i satisfies $|\tilde{x}_i - w| > \epsilon$. (b) The new node \tilde{x}_i satisfies $|\tilde{x}_i - w| < \epsilon$.

Case a) $\tilde{x}_i = \frac{1}{2}(x_i + x_{i+1})$ and $|\tilde{x}_i - w| > \epsilon$.

For our model problem, the finite element solutions u_V and $u_{\tilde{V}_i}$ have the prop-

erties:

$$u_V(x_j) = u_{\tilde{V}_i}(x_j) = u(x_j) \text{ for } j = 1, \dots, N \quad \text{and} \quad u_{\tilde{V}_i}(\tilde{x}_i) = u(\tilde{x}_i).$$

Since $\Delta e_i^2 = \|u_V - u_{\tilde{V}_i}\|_a^2$, we conclude that

$$\Delta e_i^2 \approx \frac{|u_L - u_R|}{h} \text{ where } h = x_{i+1} - x_i.$$

Case b) $\tilde{x}_i = \frac{1}{2}(x_i + x_{i+1})$ and $|\tilde{x}_i - w| < \epsilon$.

In this case it is possible that $\Delta e_i^2 \approx 0$, which means that the approximation error is going to remain almost the same if we introduce the new node \tilde{x}_i .

In case a) the values of Δe_j^2 for intervals (x_j, x_{j+1}) where the solution u is much smoother than in (x_i, x_{i+1}) , we have $\Delta e_i^2 / \Delta e_j^2 \gg 1$. So Δe_i^2 gives information of intervals that must be refined. In case b), on the other hand, the estimator Δe_i^2 does not give useful information of intervals that must be refined.

To remedy the situation presented in b), we will test new nodes at $1/2$, $1/3$, and $2/3$ of each interval:

$$\tilde{x}_i = \frac{1}{2}(x_i + x_{i+1}), \quad \tilde{x}_i = \frac{1}{3}(2x_i + x_{i+1}), \quad \text{and} \quad \tilde{x}_i = \frac{1}{3}(x_i + 2x_{i+1}).$$

Then, if $h/3 > 2\epsilon$ we get

$$\Delta e_i^2 \approx \frac{|u_L - u_R|}{2h} \text{ for at least one of the three test nodes.}$$

To simplify our refinement algorithm (mainly in 2-D), after comparing all the error reduction indicators Δe_i^2 , the refining will be done by adding nodes in the middle of the intervals selected for refinement.

The algorithm in 1-D is as follows: let T_0 be an initial partition of $(0,1)$ with nodes $x_1 < x_2 < \dots < x_N$, and compute the finite element solution u_{V_0} in the space V_0 of continuous piecewise linear functions associated to T_0 . For each interval $(0, x_1), (x_1, x_2), \dots, (x_N, 1)$ compute the error reduction estimator $\frac{|<f, \tilde{\varphi}_i>|^2}{\|\tilde{\varphi}_i\|_a^2}$ for the three test nodes $\tilde{x}_i = \frac{1}{2}(x_i + x_{i+1})$, $\tilde{x}_i = \frac{1}{3}(x_i + 2x_{i+1})$ and $\tilde{x}_i = \frac{1}{3}(2x_i + x_{i+1})$ and

take Δe_i^2 as the biggest of the three computed values. If C is a threshold, $0 < C \leq 1$ and $\Delta e_j^2 = \max_{i=0, \dots, N}(\Delta e_i^2)$, then if $\Delta e_r^2 \geq C\Delta e_j^2$, mark the interval (x_r, x_{r+1}) for refinement. Refine the marked intervals by adding a node in the middle. Let T_1 be the new mesh, V_1 the new finite element space, and N_1 the dimension of V_1 . The procedure is repeated.

Numerical experiments suggest that for the sequence of meshes T_0, T_1, \dots constructed by this algorithm, with associated finite element spaces V_0, V_1, \dots of dimensions N_0, N_1, \dots , the relation

$$\lim_{j \rightarrow \infty} \frac{\sum_{i=0}^{N_j} \Delta e_i^2}{\|u - u_{V_j}\|_a^2} = \frac{3}{4}$$

is satisfied. So, as stopping criterion, given a tolerance TOL for the approximation error, we will stop refining when we reach a mesh T_j such that $\sum_{i=0}^{N_j} \Delta e_i^2 < \frac{3}{4}(Tol)^2$; or when we have reached a maximum admissible number of nodes (see Fig. for a numerical example).

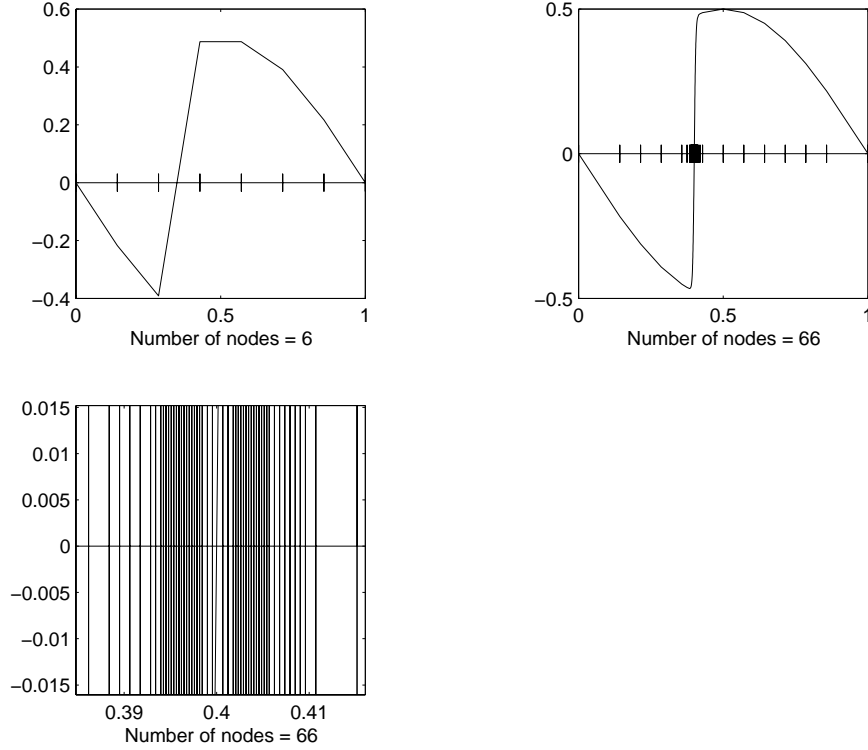


Figure 2.6: Initial mesh, refined mesh, and magnification of the refined mesh around the layer located at $x = 0.4$. The exact solution is $u(x) = \sin(\pi x) \tanh((x - 0.4)/0.005)$. For the refined mesh $\frac{\sum_{i=0}^{66} \Delta e_i^2}{\|u - u_V\|_a^2} = 0.7361$.

2.6 An anisotropic mesh refinement algorithm in 2-D

In our 2-D refinement algorithm we will compute for each interior edge η_i of the current triangulation τ the error reduction estimator Δe_W^2 , with $W = W_0$ or $W = W_4$ (see Section 2.3) for three testing nodes located at $1/3$, $1/2$, and $2/3$ of the edge η_i ; we will keep the maximum of the three computed values and denote it by Δe_i^2 . If

$$\Delta e_j^2 = \max_{i=1, \dots, M} \{\Delta e_i^2\}, \quad M = \text{Number of interior edges of } \tau,$$

and $0 < C \leq 1$ is a constant, then we mark edge η_r for refinement if $\Delta e_r^2 \geq C \Delta e_j^2$.

We now have to consider the cases of triangles with one, two, or three marked edges. To describe the refinement for the case of triangles with two marked edges we will need a few more estimates which are presented in the next section.

2.6.1 The case of two marked edges

Suppose that in the current mesh τ there is a triangle ABC with two edges, \overline{AB} and \overline{AC} , marked for refinement. We will choose one of two alternatives (shown in Figs. 2.6.1b and 2.6.1c), keeping the one that gives a bigger error reduction. We will base our selection by estimating the reduction of the error when two degrees of freedom are added to the current approximating space V .

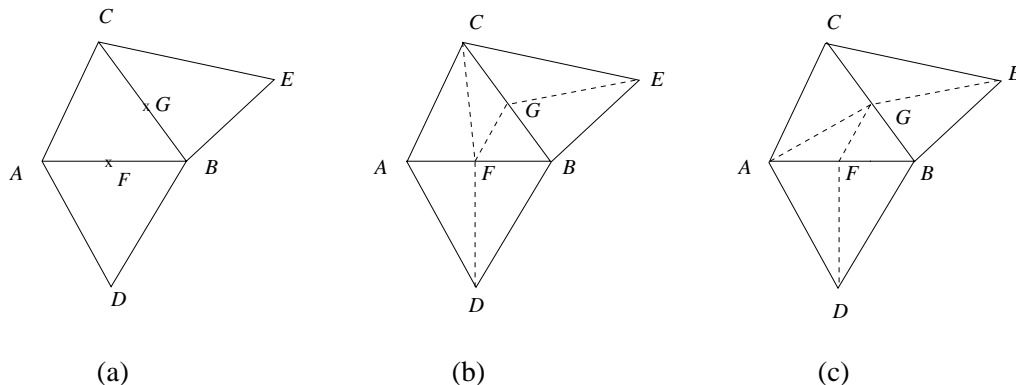


Figure 2.7: (a) The triangle ABC has two edges marked for refinement. (b) The first alternative of refinement consists of adding first node F followed by G . (c) In the second alternative the node G is added first followed by F .

Consider the alternative of Fig. b. This is done in two steps (see Fig. 2.8):

- 1) By testing a new node F on edge \overline{AB} we would obtain a new mesh $\tilde{\tau}$ and a new finite element space $\tilde{V} = V + \text{span}\{\varphi\}$ (see Section 2.3). The corresponding error reduction can be estimated using the finite element solution u_V in the mesh τ .
- 2) Going one step further, we want to estimate the error reduction obtained by adding a new node G to the mesh $\tilde{\tau}$ on edge \overline{BC} .

In order to estimate the error reduction resulting from adding the node G to $\tilde{\tau}$, we need the finite element solution $u_{\tilde{\tau}}$ which is not available. How can we get

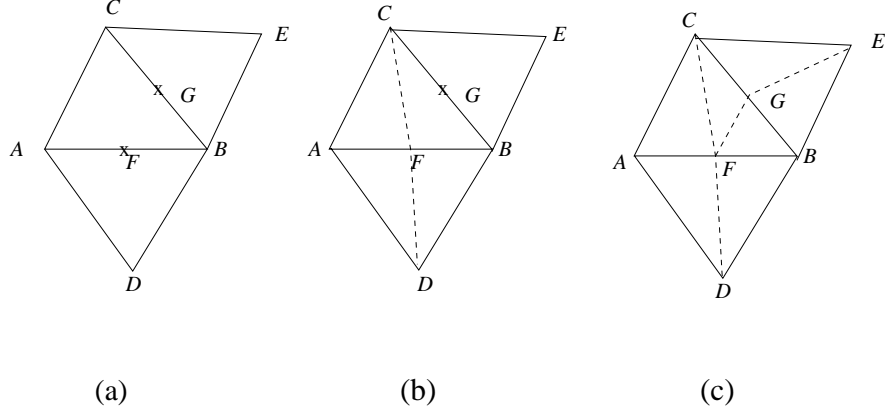


Figure 2.8: (a) Triangle ABC in current mesh τ has two edges marked for refinement. (b) Adding node F to τ produces a mesh $\tilde{\tau}$. (c) Adding node G to $\tilde{\tau}$ produces a mesh $\tilde{\tilde{\tau}}$.

an estimate of $u_{\tilde{V}}$? Let W be a linear subspace of V . It can be shown that the projection \tilde{g} of $u_V - u_{\tilde{V}}$ into the space $W + \text{span}\{\varphi\}$ with respect to the inner product a , satisfies the system of equations

$$\begin{aligned} a(\tilde{g}, v_i) &= 0 \text{ for } v_i \in W, \\ a(\tilde{g}, \varphi) &= a(u_V, \varphi) - \langle f, \varphi \rangle. \end{aligned}$$

So the difference $\tilde{u} = u_V - \tilde{g}$ is an approximation to $u_{\tilde{V}}$. Moreover \tilde{g} is related to the error reduction estimator (2.9) by

$$\|\tilde{g}\|_a = \widehat{\Delta} e^2_W.$$

In our case we have $W = W_0 = \{0\}$ or $W = W_4 = \text{span}\{v_1, v_2, v_3, v_4\}$, with v_1, v_2, v_3 , and v_4 the nodal basis functions in V associated to the nodes A, D, B , and C respectively. We now use \tilde{u} to estimate the error reduction resulting from introducing the new node G on edge \overline{BC} of the mesh $\tilde{\tau}$: let $\tilde{\tilde{g}} \in \tilde{W} + \text{span}\{\tilde{\varphi}\}$ be the solution of the system of equations

$$\begin{aligned} a(\tilde{\tilde{g}}, \tilde{v}_i) &= 0 \text{ for } \tilde{v}_i \in \tilde{W}, \\ a(\tilde{\tilde{g}}, \tilde{\varphi}) &= a(\tilde{u}, \tilde{\varphi}) - \langle f, \tilde{\varphi} \rangle, \end{aligned}$$

where $\tilde{W} = \text{span}\{\tilde{v}_1, \tilde{v}_2, \tilde{v}_3, \tilde{v}_4\}$, assuming that $\tilde{v}_1, \tilde{v}_2, \tilde{v}_3$, and \tilde{v}_4 are the basis functions in \tilde{V} associated to the nodes B, C, F , and E respectively (see Fig. 2.8b) and $\tilde{\varphi}$ the new degree of freedom with value 1 at node G . If $\tilde{\tilde{V}} = \tilde{V} + \text{span}\{\tilde{\varphi}\} = V + \text{span}\{\varphi, \tilde{\varphi}\}$, then by definition (2.5) and Equation (2.7), the exact total error reduction of adding this way the nodes F and G to the current mesh is τ is

$$\begin{aligned} \|u - u_V\|_a^2 - \|u - u_{\tilde{V}}\|_a^2 &= \left(\|u - u_V\|_a^2 - \|u - u_{\tilde{V}}\|_a^2 \right) + \\ &\quad + \left(\|u - u_{\tilde{V}}\|_a^2 - \|u - u_{\tilde{\tilde{V}}}\|_a^2 \right) \\ &= \|u_V - u_{\tilde{V}}\|_a^2 + \|u_{\tilde{V}} - u_{\tilde{\tilde{V}}}\|_a^2. \end{aligned}$$

Therefore the total error reduction can be estimated by the sum $\|\tilde{g}\|_a^2 + \|\tilde{\tilde{g}}\|_a^2$.

For the second alternative (Fig. 2.6.1c) the same procedure is applied but now we add first node G and then node F . We compare the total error reduction estimated with both alternatives and choose the one with bigger error reduction.

With respect to the function $\tilde{u} = u_V - \tilde{g}$ of the first alternative, it can be computed as follows: for the testing node $F = tB + (1-t)A$, $t = 1/3, 1/2$, and $2/3$, we have

$$\tilde{u}(F) = u_V(F) - \tilde{g}(F)$$

with $u_V(F) = tu_V(B) + (1-t)u_V(A)$ and $\tilde{g}(F) = t\beta_3 + (1-t)\beta_1 + \beta$;

$$\tilde{u}(P_i) = u_V(P_i) - \beta_i \text{ with } P_1 = A, P_2 = D, P_3 = B, P_4 = C.$$

Here we have used $\tilde{g} = \sum_{i=1}^4 \beta_i v_i + \beta\varphi$.

2.6.2 An anisotropic refinement algorithm

We now describe an algorithm for the 2-D model (3.2) problem using triangular meshes with continuous piecewise linear functions and the estimators $\widehat{\Delta e^2}_{W_0}$ and $\widehat{\Delta e^2}_{W_4}$.

i) Let M_0 be the number of interior edges of the initial mesh τ_0 . For each interior edge η_i , compute Δe^2_W for the testing nodes located at $1/3, 1/2$, and $2/3$ of η_i and

let Δe_i^2 be the biggest of the three computed values.

ii) Let $\Delta e_j^2 = \max_{i=1,\dots,M_0} \{\Delta e_i^2\}$ and $0 < C \leq 1$ a constant. Mark edge η_r for refinement if $\Delta e_r^2 \geq C\Delta e_j^2$.

iii) If a triangle in τ_0 has:

1) One marked edge, refine it by joining the middle point of the marked edge with the opposite vertex (see Fig. 2.9a).

2) Three marked edges, refine it regularly by joining the middle point of the 3 edges (see Fig. 2.9b).

3) two marked edges, choose one of the two alternatives of Sec. 2.6.1 by computing the indicators $\|\tilde{g}\|_a^2 + \|\tilde{g}\|_a^2$ for each alternative, and choosing the one with bigger indicator (see Fig. 2.9c).

iv) If a triangle has marked edges and has one edge as part of $\partial\Omega$ then it is refined as follows: if it has two marked edges, refine it regularly; for one marked edge η_r and one boundary edge η , compare the size of Δe_r^2 with the error reduction obtained if the middle point of η is joined with the opposite vertex and the middle point of η_r is joined with the middle of η (see Fig. 2.10), choose the refinement that gives a bigger error reduction.

v) Let τ_1 be the refined mesh and repeat the process until a mesh τ_j satisfies $\sum_{i=1}^{M_j} \Delta e_i^2 \leq (Tol)^2$ or the number of triangles in τ_j is bigger than a maximum number of triangles allowed.

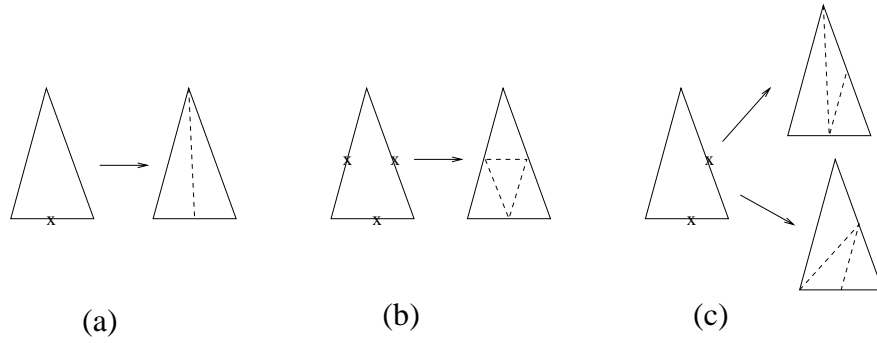


Figure 2.9: Refinement of a triangle for the case of 1, 2, or 3 marked edges. For the case of two marked edges we choose one of two alternatives (see Section 2.6.1).

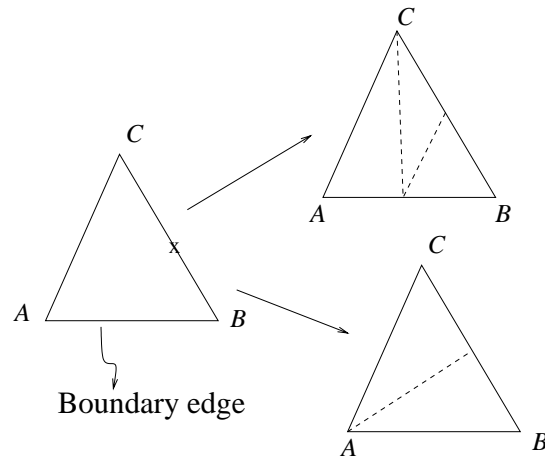


Figure 2.10: When a triangle has a boundary edge \overline{AB} and a marked interior edge \overline{BC} , it is refined by choosing one of two alternatives.

Chapter 3

An optimal mesh refinement algorithm for an interpolation problem

3.1 Introduction

There are many approximation problems for functions f of several variables (given explicitly or implicitly) e.g., as the solution of some equation, where the task is to find a sequence of triangulations $\{\tau_i\}$ of the domain of f and functions $\{f_i\}$, where f_i is linear on each triangle of mesh τ_i . The goal is to make $\|f_i - f\|$ as small as possible with a small number of triangles.

To achieve this, the meshes $\{\tau_i\}$ are generated adaptively, that is, τ_{i+1} is generated by introducing triangles only in some “appropriate” parts of τ_i . The “appropriate” parts are usually found with a criterium based on calculations on previous meshes.

As a model problem we consider here a function f with a jump discontinuity along a closed curve σ in the unit square and with constant values (1 and 0) in the two different regions separated by σ . We want to approximate the function

f with functions that are continuous and piecewise linear on triangular grids; the approximating error is measured in the L^1 norm.

For this model problem, it can be shown that if the approximating function \tilde{f} is the interpolant of f , then the optimal approximating error is $O(1/N^2)$ when using grids with N triangles, contrasting with $O(1/N)$ when using shape regular grids. For a long time it was an open problem to find a hierarchical algorithm that achieves $O(1/N^2)$. We analyse here shape regular refinement and an anisotropic refinement algorithm that achieves the optimal order.

3.2 The model problem

Our model problem is the following: Let $\sigma : [a, b] \rightarrow \mathbb{R}^2$ be a closed and smooth curve in the plane, which is 1-1 on (a, b) and enclosed in a square Ω . Let $\overline{\Omega}_I$ be the closure of the open, bounded set enclosed by σ .

Define $f : \Omega \rightarrow \mathbb{R}$ as

$$f(x, y) = \begin{cases} 1 & \text{if } (x, y) \in \overline{\Omega}_I, \\ 0 & \text{otherwise.} \end{cases} \quad (3.1)$$

Given a triangular mesh T of Ω , denote by N_T the number of triangles in T and by f_T the linear interpolant of f . Thus, f_T is the continuous function that satisfies:

- i) f_T is linear in each triangle $K \in T$.
- ii)

$$f_T(\mathbf{a}) = \begin{cases} 1 & \text{if } \mathbf{a} \in \overline{\Omega}_I, \\ 0 & \text{otherwise.} \end{cases} \quad (3.2)$$

for all vertices \mathbf{a} of all triangles $k \in T$.

The error E_T will measure how well f_T approximates f and in the L^1 norm:

$$E_T = \int_{\Omega} |f - f_T| dx dy.$$

The problem consists in finding a sequence of meshes T_0, T_1, \dots where T_j is an "improvement" of T_{j-1} and such that $E_{T_j} = O(1/N_j^2)$.

We will consider two type of meshe sequences. For the first one, T_j is obtained by refining regularly some of the triangles in the mesh T_{j-1} , and preserving a local quasi-uniformity in the mesh (see Bank [9]). For the second sequence of meshes, the shape regularity of elements is not preserved and triangles with high aspect ratio are introduced. The rate of convergence for the anisotropic case is much better than the regular one. In this construction, however, a good choice of the initial mesh is important in order to achieve an optimal rate of convergence.

3.3 Regular mesh refinement

For a regular mesh refinement algorithm that produces local quasi-uniform meshes T , the approximation error E_T is no better than $O(1/M)$, where M is the number of triangles in T . To see this, notice that for our model problem the triangles K_1, K_2, \dots, K_N in T that intersect σ are the only ones that contribute to the approximation error E_T . Regardless of the shape of K_i , with some estimates using the formulae to calculate the volume of a tetrahedrom and other geometric shapes in 3-D we can obtain the inequalities

$$\frac{1}{6} \text{Area}(K_i) \leq \int_{K_i} |f - f_T| dx dy \leq \text{Area}(K_i) \quad \text{for } i = 1, \dots, N.$$

So

$$\frac{1}{6} \sum_{i=1}^N \text{Area}(K_i) \leq E_T \leq \frac{2}{3} \sum_{i=1}^N \text{Area}(K_i).$$

Let $h_i = \text{diam}(K_i)$. Then local quasi-uniformity implies that

$$\mu_1 h_i^2 \leq \text{Area}(K_i) \leq \mu_2 h_i^2 \quad \text{for } i = 1, \dots, N,$$

where the positive constants μ_1 and μ_2 are independent of the mesh T . The approximation error is therefore proportional to $\sum_{i=1}^N h_i^2$. If $l = \text{length}(\sigma)$, then $\sum_{i=1}^N h_i$ is proportional to l . The method of Lagrange multipliers can be used to show that $h_i = l/N$ for $i = 1, \dots, N$, minimizes $\sum_{i=1}^N h_i^2$ subject to $\sum_{i=1}^N h_i = l$. And so the minimum approximation error is of order l^2/N .

3.4 Anisotropic refinement

In this section we construct an anisotropic refinement algorithm which achieves the optimal rate of convergence, $O(1/N^2)$ for meshes with N triangles, for the model problem of Section (3.2). Mesh refinement in this algorithm requires removing some triangles in order to introduce new ones as illustrated in Fig. 3.1, where the two triangles in the quadrilateral $P_1P_2P_3P_4$ (Fig. 3.1a) are decomposed producing the two quadrilaterals $C_1C_2C_3C_4$ and $C_2Q_1Q_2C_3$ of Fig. 3.1b.

3.4.1 An anisotropic refinement algorithm

Our anisotropic refinement algorithm requires a “good” initial mesh. The initial mesh has the purpose that during the refinement step the new elongated triangles enclose the curve σ which allow us to control the error. We describe now how to construct the initial mesh and how the refinement step is done. In the algorithm it is assumed that the curve σ of the model problem of Section 3.2 is smooth with curvature bounded by $1/R$. The algorithm is as follows (see Fig. 3.1):

Initial Mesh

- a) Split the curve σ into arcs of length at most $\pi R/3$
- b) For each of the arcs σ_{AB} obtained, let $P_1P_2P_3P_4$ be a quadrilateral such that A and B are the middle points of $\overline{P_1P_4}$ and $\overline{P_2P_3}$ respectively, $\overline{P_2P_3}$ and $\overline{P_1P_4}$ are both normal to the arc σ_{AB} , and $\|\overline{P_2P_3}\| = \|\overline{P_1P_4}\| \geq 0.7R$.

Refinement procedure

- 1) Let D be the middle point of the curve σ_{AB} and construct the quadrilateral $C_1C_2C_3C_4$ with the following properties :

- i) C_1 and C_4 are on the line segment $\overline{P_1P_4}$
 - ii) A is the middle point of $\overline{C_1C_4}$, D is the middle point of $\overline{C_2C_3}$
 - iii) $\|\overline{C_1C_4}\| = \frac{1}{4}\|\overline{P_1P_4}\| = \|\overline{C_2C_3}\|$
 - iv) $\overline{C_2C_3}$ is normal to σ_{AD} .
- 2) Similarly, construct the quadrilateral $C_2Q_1Q_2C_3$ for the arc segment σ_{DB}
- For the next refinements repeat steps 1) and 2) for each of the resulting quadrilaterals.

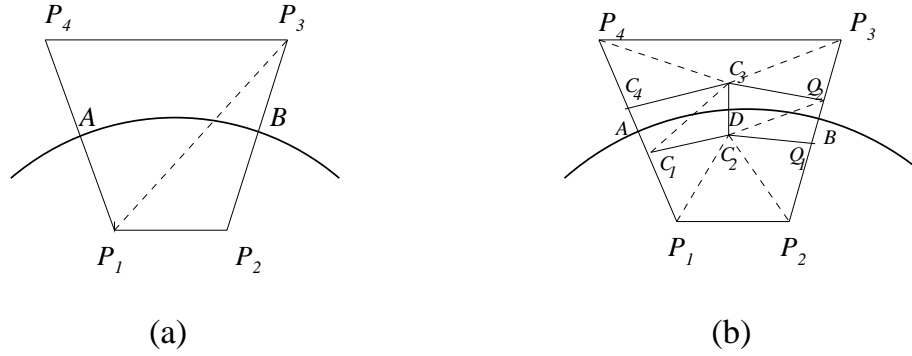


Figure 3.1: Type of refinement for the anisotropic refinement algorithm. The triangles in quadrilateral (a) are replaced by the ones in (b).

The main properties of the above refinement algorithm are established in the following

Theorem 3.4.1 *Let σ be a smooth convex closed curve inside the unit square whose length and curvature are bounded by a constant C . Let f be the function with values 1 in the region enclosed by σ and 0 otherwise. Then the refinement algorithm constructs a sequence of grids T_0, T_1, \dots such that*

$$\|f_{T_j} - f\|_{L^1} = O\left(\frac{C}{N^2}\right),$$

where N is the number of triangles in the mesh T_j and the approximating function f_{T_j} is taken as the linear interpolant of f in the mesh T_j .

Remark In Theorem 3.4.1 the assumption that σ is convex can be dropped as

long as the triangles in the initial mesh construction do not overlap. In the next section we state some lemmas which will be used to prove Theorem 3.4.1.

3.4.2 Analysis of the anisotropic refinement algorithm

In this section we prove Theorem 3.4.1. It will be useful to consider first the case when σ is a circle of radius R ; we will see that the general case follows from this particular one.

Lemma 3.4.1 *Let σ be a circle of radius R and let A and B be two points on the circle such that σ_{AB} is an arc of length at most $\pi R/3$. Define three sequences of real numbers $\{S_n\}$, $\{L_n\}$, and $\{K_n\}$ as follows (see Fig. 3.2) :*

let $D_0 = B$, and for any integer $n \geq 0$ let D_{n+1} be the middle point of σ_{AD_n} , that is, $D_{n+1} \in \sigma_{AD_n}$ and $\text{length}(\sigma_{AD_{n+1}}) = \text{length}(\sigma_{D_{n+1}D_n})$; $S_n = \text{length}(\sigma_{AD_n})$, $L_n = \text{length}(\overline{AD_n})$, and $K_n = \text{distance from } D_{n+1} \text{ to the middle point of } \overline{AD_n}$.

Then $\{L_n\}$, and $\{K_n\}$ satisfy

$$\frac{L_0}{2^n} \leq L_n < \frac{S_0}{2^n} \quad \text{for } n \geq 0, \quad (3.3)$$

$$\frac{K_0}{4^n} \leq K_n \leq \frac{S_0^2}{8R} \left(\frac{1}{4^n} \right). \quad (3.4)$$

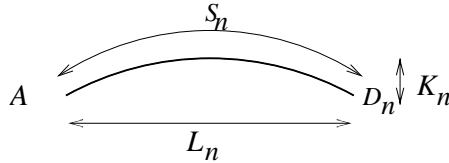


Figure 3.2: Construction of the sequences $\{S_n\}$, $\{L_n\}$, $\{K_n\}$.

PROOF: From the definition of S_n and L_n we conclude

$$L_{n+1} \geq \frac{L_n}{2} \quad \text{and} \quad S_n = \frac{S_0}{2^n} \quad \text{for } n \geq 0. \quad (3.5)$$

Then (3.3) follows from (3.5) and the fact that $L_n < S_n$.

Thus

$$\frac{L_0}{2^n} \leq L_n < \frac{S_0}{2^n} \quad \text{for } n \geq 0. \quad (3.6)$$

If O is the center of the circle, define the angle α_n as in Fig. 3.3. Then (3.4) follows from the relations

- i) $K_n = R(1 - \cos \alpha_{n+1})$
- ii) $\alpha_{n+1} = S_{n+1}/R = S_0/(2^{n+1}R)$
- iii) $1 - \cos \beta < \beta^2/2$ for any $\beta \neq 0$.

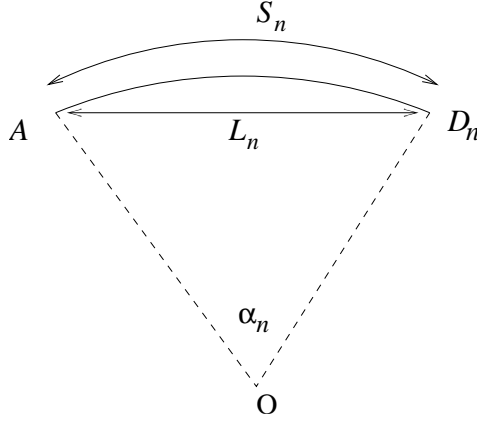


Figure 3.3: Angle α_n in the proof of Lemma 3.4.1

- Lemma 3.4.2** For the sequences of arcs $\{\sigma_{AD_n}\}$ of Lemma 3.4.1, let $\{P_{n,1}P_{n,2}P_{n,3}P_{n,4}\}$ be a sequence of trapezoids such that (see Fig. 3.4)
- i) A and D_n are the middle points of $\overline{P_{n,1}P_{n,4}}$ and $\overline{P_{n,2}P_{n,3}}$ respectively
 - ii) $\overline{P_{n,1}P_{n,4}}$ and $\overline{P_{n,2}P_{n,3}}$ are normal to the arc $\{\sigma_{AD_n}\}$
 - iii) $\|\overline{P_{n,1}P_{n,4}}\| = \frac{1}{4}\|\overline{P_{n-1,1}P_{n-1,4}}\| = \|\overline{P_{n,2}P_{n,3}}\|$.

Then the height h_n of the trapezoid $P_{n,1}P_{n,2}P_{n,3}P_{n,4}$ (that is, the distance from $\overline{P_{n,1}P_{n,2}}$ to $\overline{P_{n,4}P_{n,3}}$) satisfies

$$(0.954724)\frac{h_0}{4^n} \leq h_n \leq \left(\frac{2}{\sqrt{3}}\right)\frac{h_0}{4^n} \quad \text{for } n \geq 0, \quad \text{provided that } S_0 \leq \pi R/3. \quad (3.7)$$

If A_n is the area of the trapezoid $P_{n,1}P_{n,2}P_{n,3}P_{n,4}$ then

$$\frac{A_0}{8^n}(0.9548) \leq A_n \leq \left(\frac{2.4}{\sqrt{3}}\right)\frac{A_0}{8^n} \quad \text{for } n \geq 0, \quad \text{provided that } S_0 \leq \pi R/3. \quad (3.8)$$

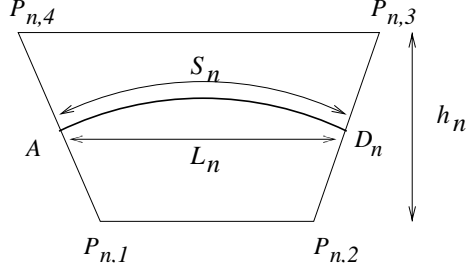


Figure 3.4: The trapezoid in the n -th refinement step.

PROOF: Let $l_n = \|\overline{P_{n,2}P_{n,3}}\|$. We have from the definition of the angle α_n that

$$\frac{h_{n+1}}{l_n} = \frac{\cos \alpha_{n+2}}{4} \geq \frac{1}{4} - \frac{1}{8}(\alpha_{n+2})^2 = \frac{1}{4} - \frac{1}{32} \left(\frac{S_0}{R}\right)^2 \frac{1}{4^{n+1}}. \quad (3.9)$$

If $C = (S_0/R)^2/8$, it follows that

$$h_{n+1} \geq \frac{h_0}{4^{n+1}} \prod_{j=1}^{\infty} \left(1 - \frac{C}{4^j}\right). \quad (3.10)$$

If $S_0/R \leq \pi/3$, then a calculation (with the help of a computer) shows that

$$h_{n+1} \geq (h_0/4^{n+1})(0.954724). \quad (3.11)$$

From Equation (3.9) we obtain

$$\frac{h_{n+1}}{l_n} = \frac{\cos \alpha_{n+2}}{4} \leq \frac{1}{4}. \quad (3.12)$$

Since $S_0 \leq \pi/3$, it follows that $\alpha_1 \leq \pi/6$. Since $l_n = l_{n-1}/4$ and $h_0 = l_0 \cos \alpha_1 \geq l_0\sqrt{3}/2$, from (3.11) and (3.12) we obtain the inequalities (3.7). Let A_n be the area of the trapezoid $P_{n,1}P_{n,2}P_{n,3}P_{n,4}$ (see Fig. 3.4). From (3.3) and (3.7), A_n satisfies

$$A_n = L_n h_n \leq \frac{2}{\sqrt{3}} \left(\frac{S_0}{2^n}\right) \left(\frac{h_0}{4^n}\right) = \frac{2}{\sqrt{3}} \frac{S_0 h_0}{8^n}, \quad (3.13)$$

and

$$A_n \geq \left(\frac{L_0}{2^n}\right) \left(\frac{h_0}{4^n}\right) (0.9548) = \frac{L_0 h_0}{8^n} (0.9548) = (0.9548) \frac{A_0}{8^n}. \quad (3.14)$$

The restriction $S_0 \leq \pi R/3$ implies that $S_0 \leq 1.2L_0$, which combined with (3.13) and (3.14) gives (3.8). This proves Lemma 3.4.2.

One of the requirements in the design of the anisotropic algorithm of Section (3.4.1) was that the trapezoid $P_{n,1}P_{n,2}P_{n,3}P_{n,4}$ encloses the arc σ_{AD_n} . This condition holds if

$$4K_n < h_n, \quad \text{for all } n \geq 0. \quad (3.15)$$

From (3.4) and (3.7), the last inequality holds if

$$4 \left(\frac{S_0^2}{8R} \right) \left(\frac{1}{4^n} \right) < (0.954724) \frac{h_0}{4^n} \quad \text{and} \quad S_0 \leq \frac{\pi R}{3},$$

i.e.,

$$h_0 \geq \frac{S_0^2}{2R}(1.06) \quad \text{provided that} \quad S_0 \leq \frac{\pi R}{3}. \quad (3.16)$$

This means that if R is big then h_0 is of size $O(1/R)$. If $R < 1$ then h_0 is of size $O(R)$.

Let us analyse the convergence of the anisotropic algorithm when σ is a circle of radius R . Let T_0, T_1, \dots be the sequence of meshes produced by the algorithm (Fig. 3.5 shows the initial mesh and 3 refinement steps). For the j -th refinement the error E_{T_j} is proportional to the sum of the areas of the triangles in T_j that intersect σ . Any of the first trapezoids $P_{0,1}P_{0,2}P_{0,3}P_{0,4}$ generated by the initial mesh has a height h_0 that satisfies (3.16) due to the fact that $\|\overline{P_{0,2}P_{0,3}}\| = \|\overline{P_{0,1}P_{0,4}}\| \geq 0.7R$. Thus the set of all trapezoids generated in the j -th refinement will contain the curve σ . Taking into account that the area of each trapezoid in the j -th refinement is bounded by $A_0/8^j$, similar calculations as in the regular refinement case of Section (3.3) allow us to conclude that

$$E_{T_j} = O \left(\frac{l^2}{(N_{T_j})^2} \right), \quad \text{where } l = \text{length}(\sigma). \quad (3.17)$$

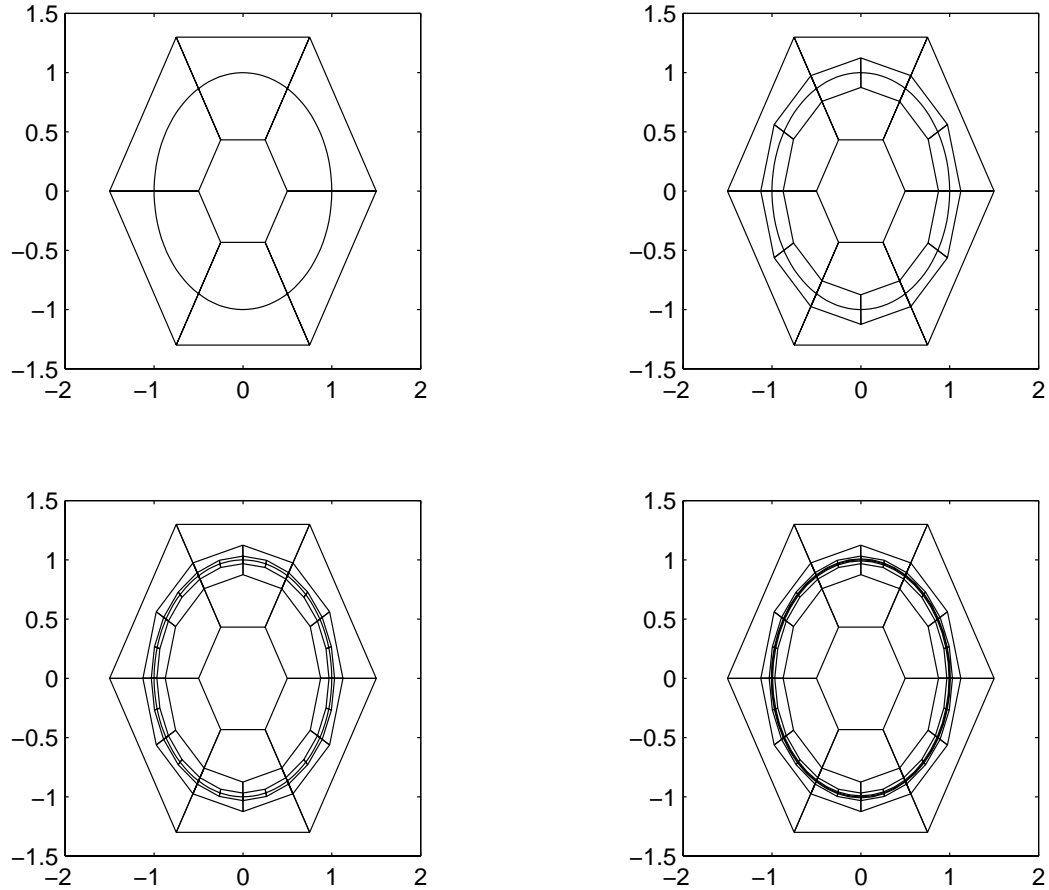


Figure 3.5: Initial mesh and three refinement steps when σ is a circle.

We turn our attention now to the anisotropic refinement algorithm applied to curves other than circles. In order to prove Theorem 3.4.1 we will use the following lemma,

Lemma 3.4.3 *Let σ be a smooth closed curve inside the unit square. Let A and B be two points on σ such that the curvature on the arc σ_{AB} has constant sign and is bounded in absolute value by $1/R$. If $\|\overline{AB}\| \leq R$, then the arc σ_{AB} is completely enclosed between the line segment \overline{AB} and the arc C_{AB} of the circle C of radius R that passes through A and B (Fig. 3.6).*

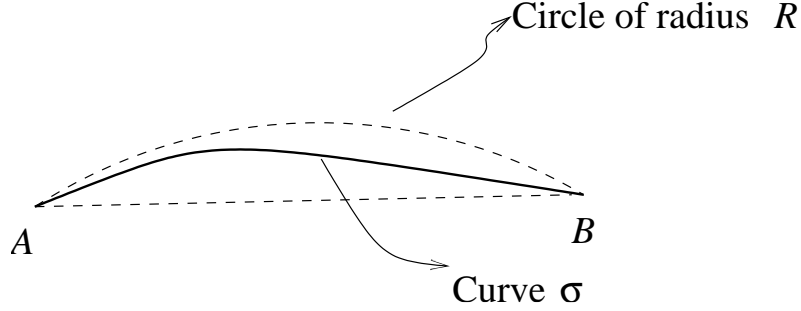


Figure 3.6: The arc σ_{AB} is enclosed by an arc of a circle and a line segment \overline{AB} , as stated in Lemma 3.4.3

We now link the above lemmas to show Theorem 3.4.1. Let σ , A , B , R , C_{AB} be as in Lemma 3.4.2. Define D and F as the middle point of the arcs C_{AB} and σ_{AB} respectively. Let $L_0 = \|\overline{AB}\|$, $L_1 = \|\overline{AD}\|$. The initial mesh construction of Section 3.4.1 for the arc σ_{AB} produces a quadrilateral $Q_1Q_2Q_3Q_4$. Similarly from the initial mesh procedure applied to the arc of circle C_{AB} the trapezoid $P_1P_2P_3P_4$ is obtained as in Fig. 3.7. As seen before, the initial mesh construction ensures that the trapezoid $Q_1Q_2Q_3Q_4$ encloses the arc C_{AB} , which, combined with Lemma 3.4.2 guarantee that the quadrilateral $Q_1Q_2Q_3Q_4$ encloses the arc σ_{AB} . This and the fact that the set of trapezoids produced during the j -th refinement procedure applied to a circle, imply that the quadrilaterals obtained during the j -th refinement procedure will enclose the curve σ . Moreover, the j -th refinement procedure applied to the arc of circle C_{AB} produces trapezoids whose areas are of the same order of magnitude as the quadrilaterals obtained after the j -th refinement is applied to σ_{AB} . Noticing now that if in Lemma 3.4.3 we allow the piece of σ to change sign then the curve will be enclosed between the two arcs of circles of radii R that pass through A and B (see Fig. 3.8). This proves Theorem 3.4.1.

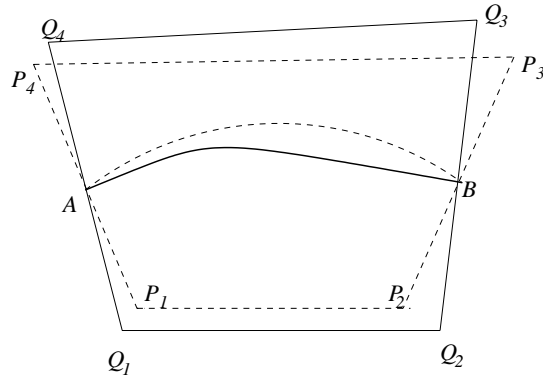


Figure 3.7: Construction used to show that the quadrilateral $Q_1Q_2Q_3Q_4$ encloses the arc σ_{AB} .

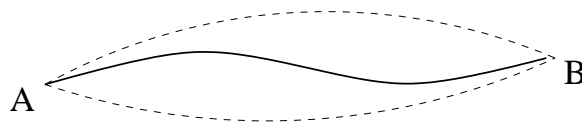


Figure 3.8: The arc σ_{AB} , with curvature bounded by $1/R$, is enclosed between two arcs of circles of radii R .

Chapter 4

Numerical examples

In this chapter we present more details of the type of meshes generated by the estimators $\widehat{\Delta e^2}_{W_0}$ and $\widehat{\Delta e^2}_{W_4}$. In numerical experiments the estimators $\widehat{\Delta e^2}_{W_0}$ and $\widehat{\Delta e^2}_{W_4}$ worked very well with the meshes generated by our refinement algorithm. We present two examples for the elliptic model problem (2.1). In the first one we selected the data f such that the exact solution is $u = \sin(\pi x) \sin(\pi y) \tanh((r - 0.5)/\epsilon)$, where $r = \sqrt{(x - 0.1)^2 + (y - 0.1)^2}$ and $\epsilon = 0.005$. This solution has rapid variations in a thin layer (of order ϵ) along an arc of a circle centered at $(0.1, 0.1)$ and with radius 0.5 (see [19]). The starting grid contains 25 degrees of freedom (see Fig. 4.1). After several refinements based on the estimator $\widehat{\Delta e^2}_{W_0}$ and the threshold $C=1/15$, we obtained a mesh with 1780 degrees of freedom. To see how well $\widehat{\Delta e^2}_{W_0}$ and $\widehat{\Delta e^2}_{W_4}$ approximate Δe^2 on the meshes generated, we computed the exact value of Δe^2 for each interior edge, obtaining the following results: 1) for the initial mesh: $1 \leq \Delta e^2 / \widehat{\Delta e^2}_{W_0} \leq 1.18$ and $1 \leq \Delta e^2 / \widehat{\Delta e^2}_{W_4} \leq 1.0036$ for all interior edges. 2) For the refined mesh: $1 \leq \Delta e^2 / \widehat{\Delta e^2}_{W_0} \leq 3.9$ and $1 \leq \Delta e^2 / \widehat{\Delta e^2}_{W_4} \leq 1.7$ for all interior edges. The aspect ratio (which for triangles can be defined as the quotient of the longest side over the shortest height) of the initial triangulation is constant=2; for the refined mesh it varies between 2 and 137. In the second example the solution has two interior layers that intersect. In general it can be noticed that $\widehat{\Delta e^2}_{W_0}$ generates triangles that are slightly more elongated than the ones generated with $\widehat{\Delta e^2}_{W_4}$. One reason for this is that $\widehat{\Delta e^2}_{W_0}$ is less accurate than $\widehat{\Delta e^2}_{W_4}$ when

a new degree of freedom is tested for the long side of a triangle with shape as in Fig. 2.4b. Even for this case, our numerical experiments show that $\widehat{\Delta e^2}_{W_4}$ remains small (at most 2) and $\Delta e^2/\widehat{\Delta e^2}_{W_0}$ at most 6 for very high aspect ratio elements. We present the results obtained for $\frac{\sum_{i=0}^N \Delta e_i^2}{\|u-u_V\|_a^2}$ for large N , where V denotes the finite element space associated to the last refined mesh; N is the number of interior edges in the last refined mesh, and Δe_i^2 is the error reduction indicator (using $\widehat{\Delta e^2}_{W_0}$ or $\widehat{\Delta e^2}_{W_4}$ applied to the the i -th interior edge. We will see that for large N , $\frac{\sum_{i=0}^{N_j} \Delta e_i^2}{\|u-u_{V_j}\|_a^2}$ is close to 3/4, in agreement to what happens in the 1-D model problem (see the end of Sec. 2.5). We have also compared the anisotropic meshes generated by our algorithm with an adaptive shape regular refinement (using the error reduction approach to detect triangles that must be refined), obtaining savings in the number of triangles by a factor of 15.

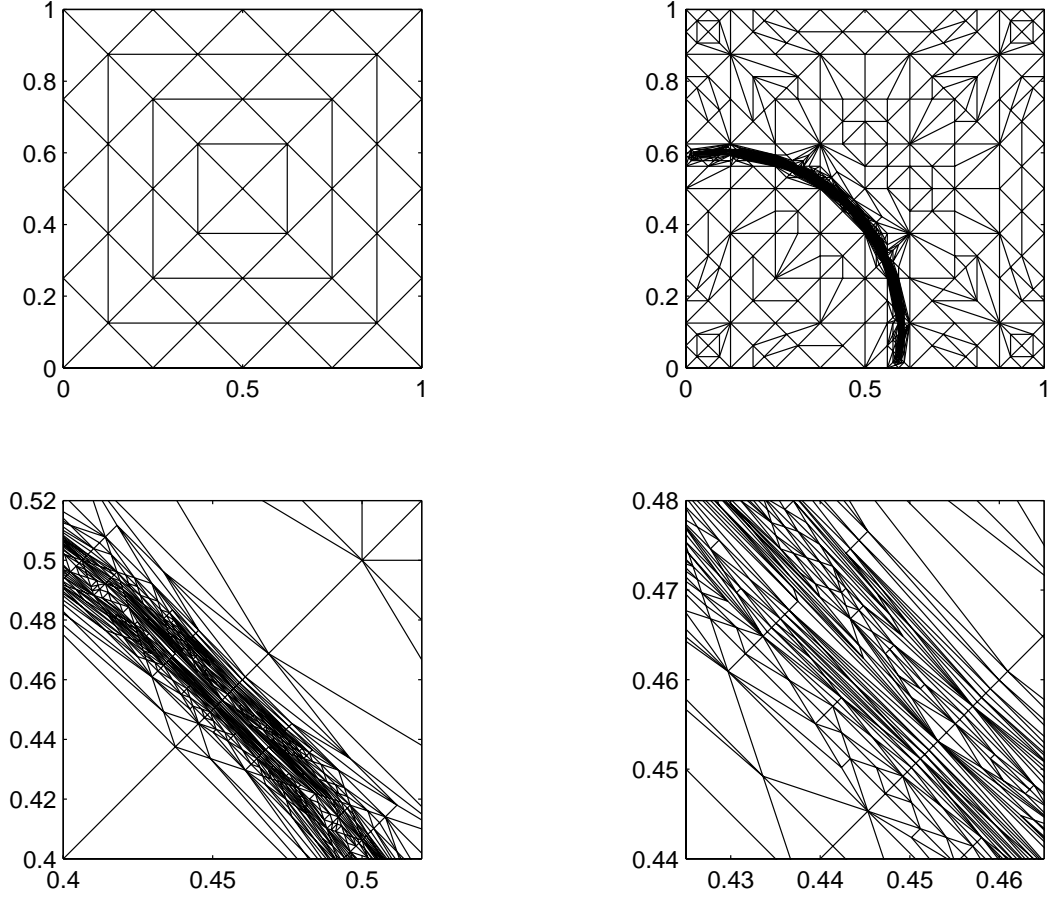


Figure 4.1: Initial grid with 25 degrees of freedom, refined grid with 1780 degrees of freedom using $\widehat{\Delta e^2}_{W_0}$, and blow up of the refined grid. The exact solution is $\sin(\pi x) \sin(\pi y) \tanh((r - 0.5)/\epsilon)$, where $r = \sqrt{(x - 0.1)^2 + (y - 0.1)^2}$ and $\epsilon = 0.005$. The refined mesh contains triangles with aspect ratio up to 137 and $1 \leq \Delta e^2 / \widehat{\Delta e^2}_{W_0} \leq 3.9$, $1 \leq \Delta e^2 / \widehat{\Delta e^2}_{W_4} \leq 1.7$ for all its interior edges.

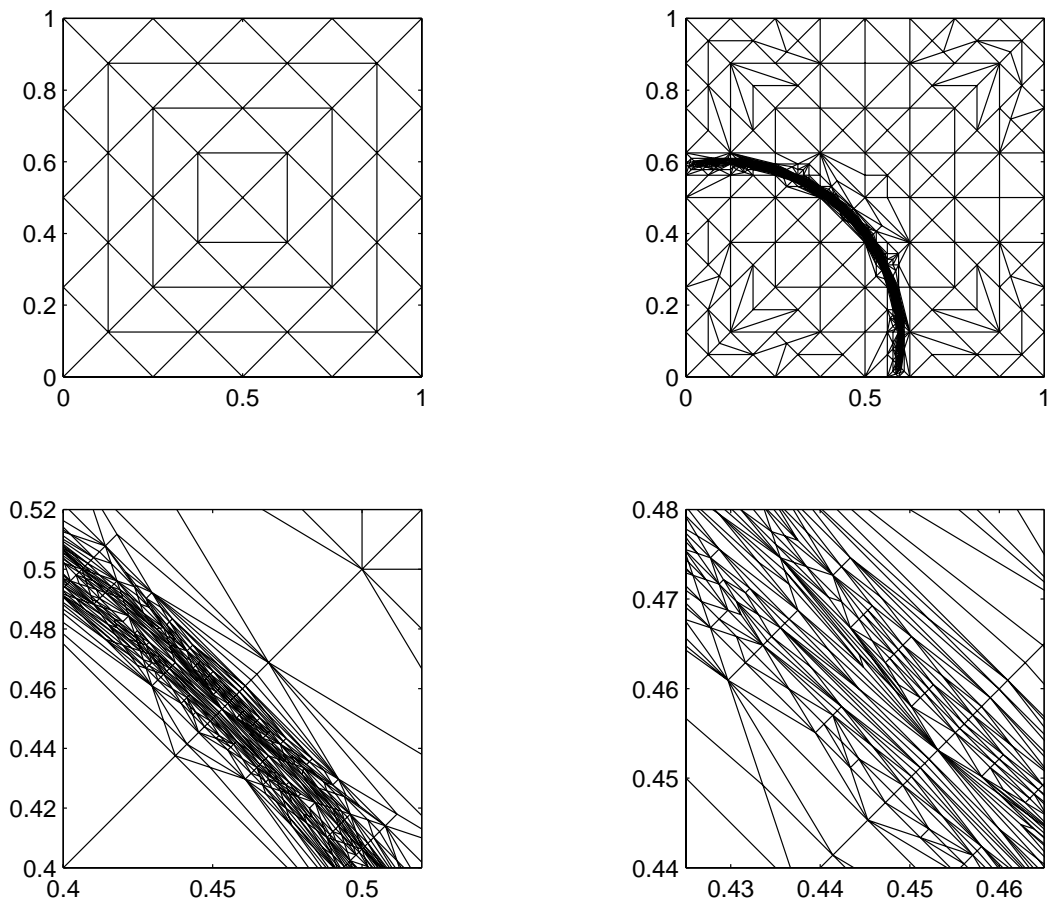


Figure 4.2: Same as Fig. 4.1 but using $\widehat{\Delta e^2}_{W_4}$. The final mesh contains 1700 degrees of freedom.

4.1 An example using $\widehat{\Delta e^2}_{W_4}$

Here we approximate the exact solution $\sin(\pi x) \sin(\pi y) \tanh((r - 0.5)/\epsilon)$ of the elliptic model problem (2.1) using the estimator $\widehat{\Delta e^2}_{W_4}$. The maximum aspect ratio in the refined mesh was 65, the last mesh contains 1700 degrees of freedom, the maximum value of $\Delta e^2/\widehat{\Delta e^2}_{W_4}$ and $\Delta e^2/\widehat{\Delta e^2}_{W_0}$ was 1.46 and 3.41 respectively. $\frac{\sum_{i=0}^{N_j} \Delta e_i^2}{\|u - u_{V_j}\|_a^2} = 0.74$.

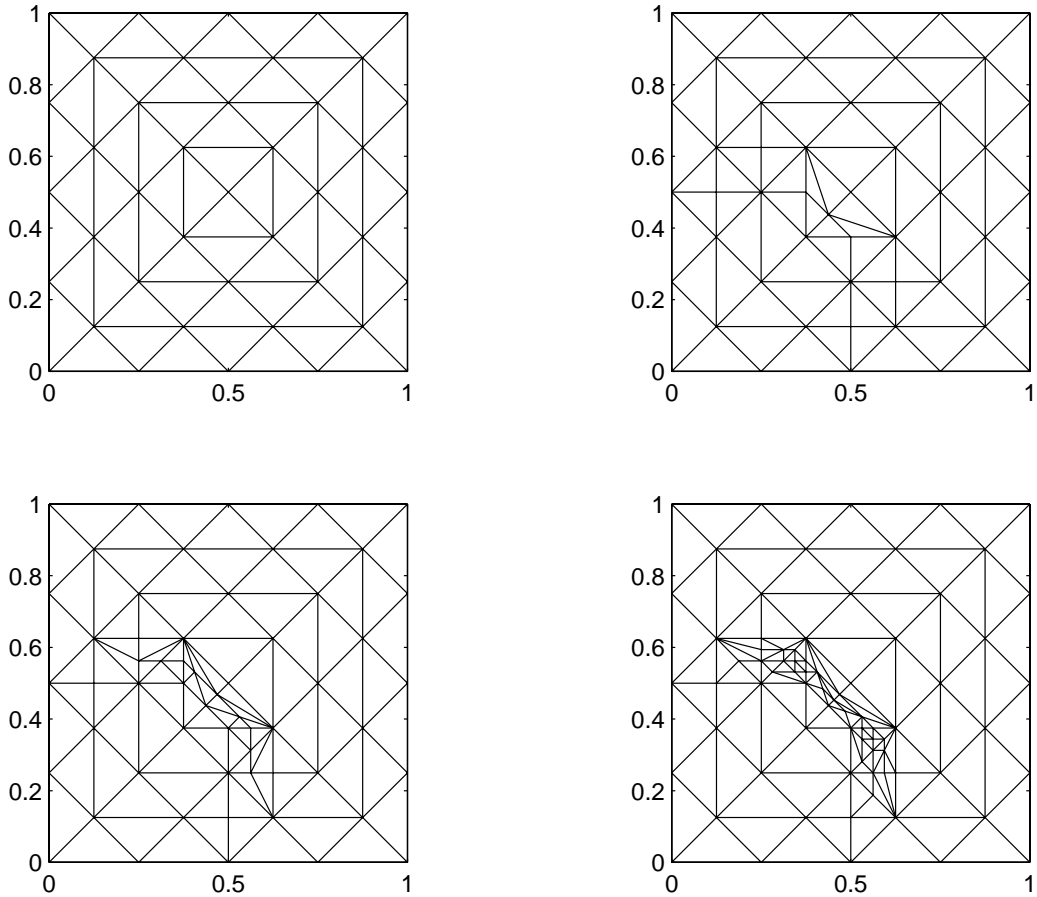


Figure 4.3: Initial mesh and the first three refinements using $\widehat{\Delta e^2}_{W_4}$. The exact solution is $\sin(\pi x) \sin(\pi y) \tanh((r - 0.5)/\epsilon)$, where $r = \sqrt{(x - 0.1)^2 + (y - 0.1)^2}$ and $\epsilon = 0.005$

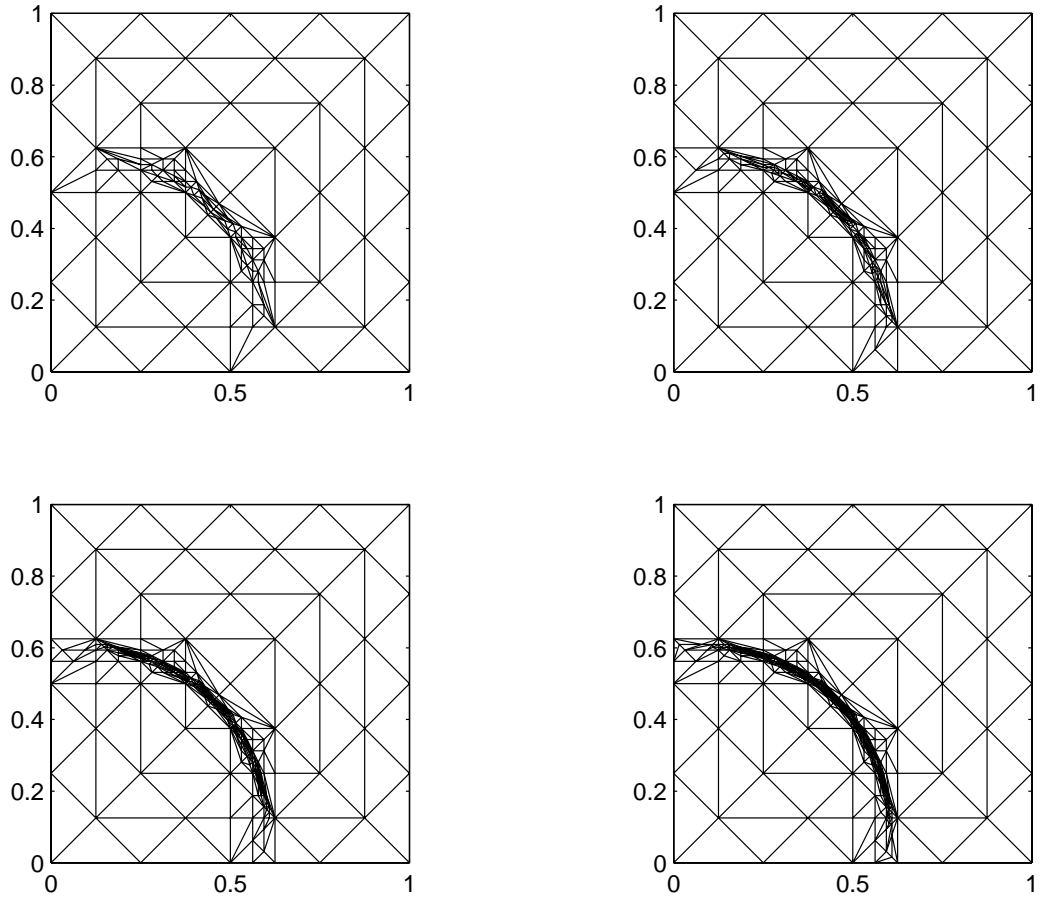


Figure 4.4: Next four refinements.

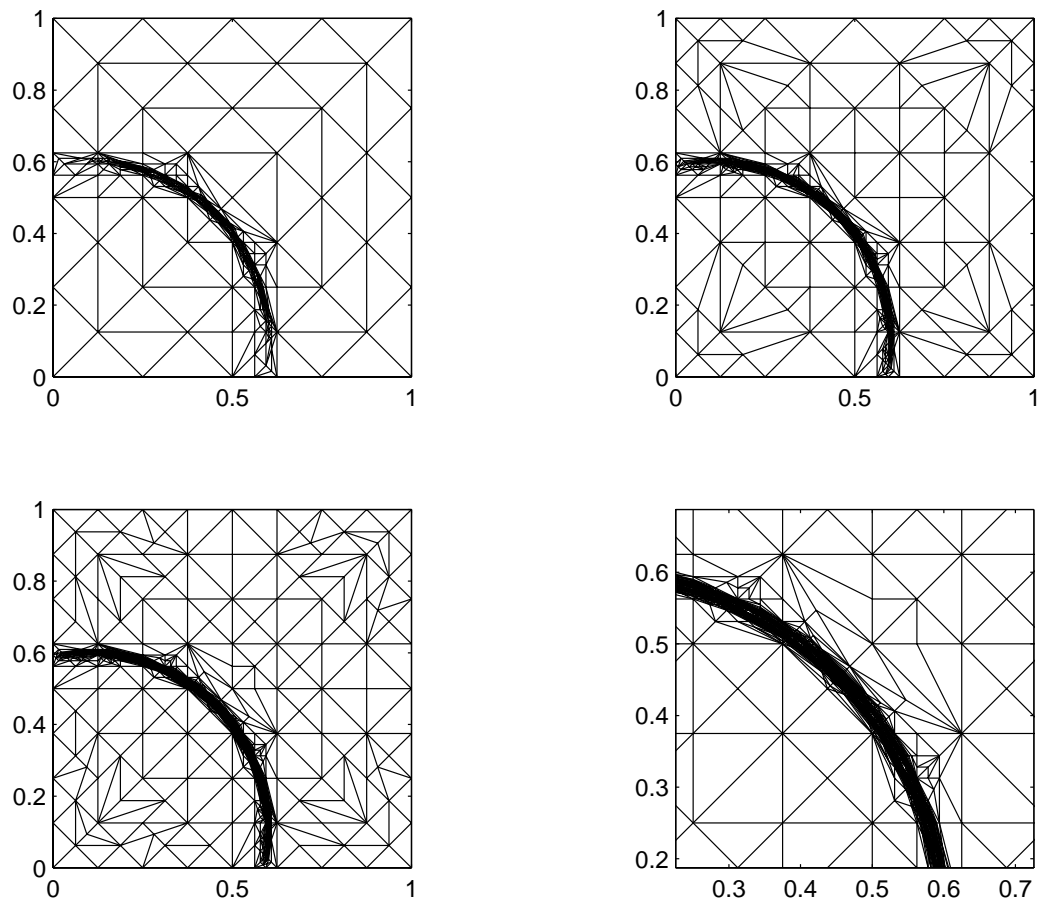


Figure 4.5: Next refinements, the last one with 1700 degrees of freedom. Also a blow up of the last mesh.

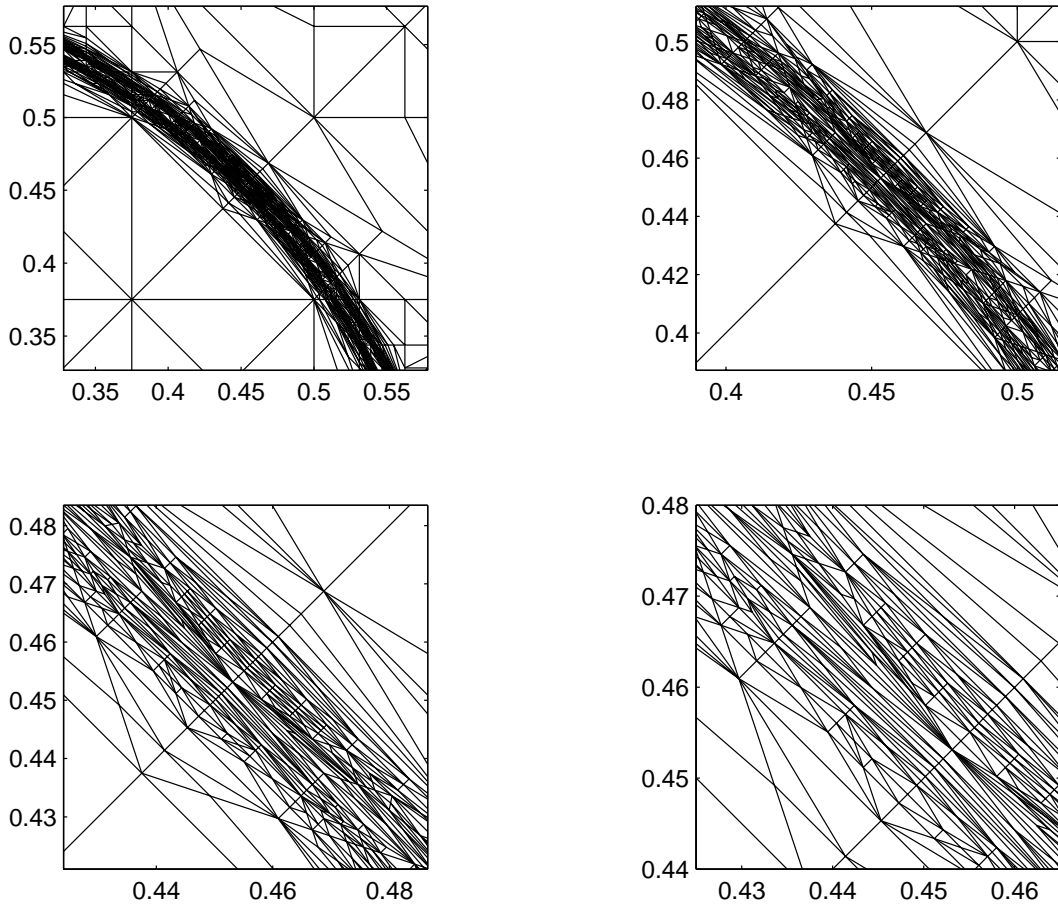


Figure 4.6: Blow up of the last mesh.

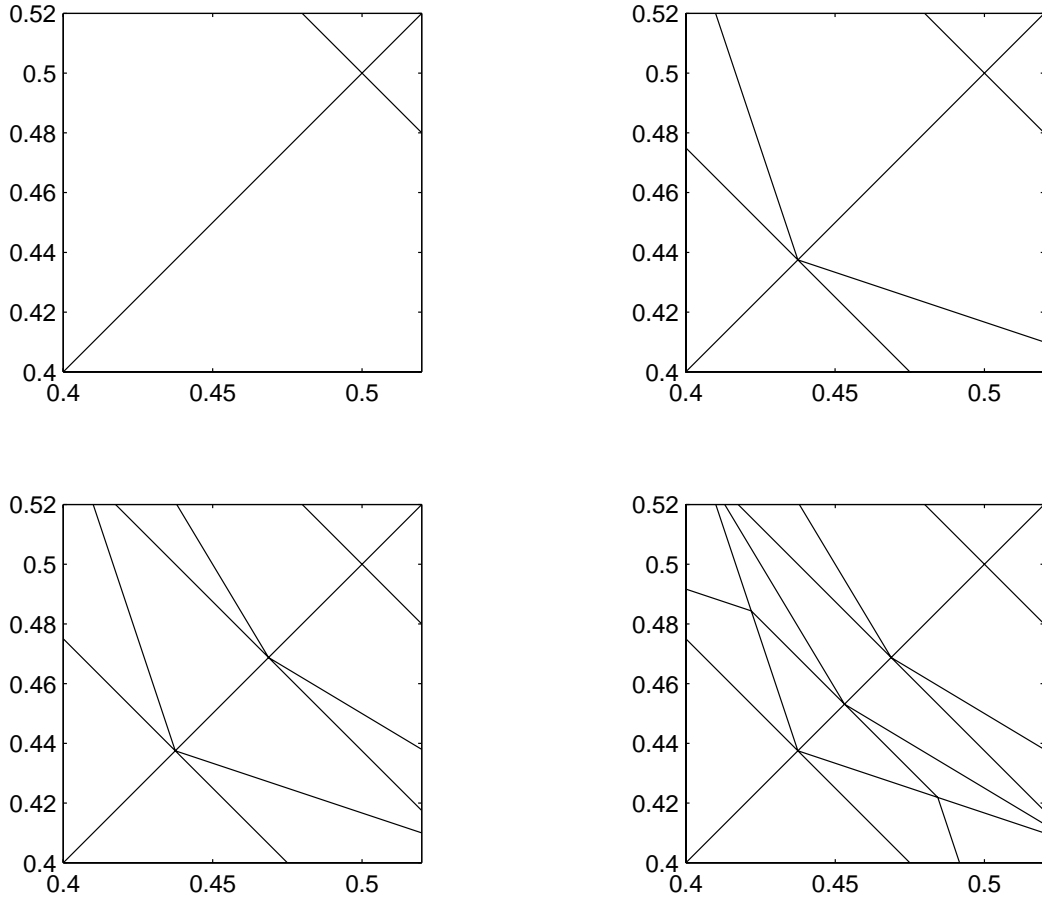


Figure 4.7: Blow up of the initial mesh and the first three refinements.

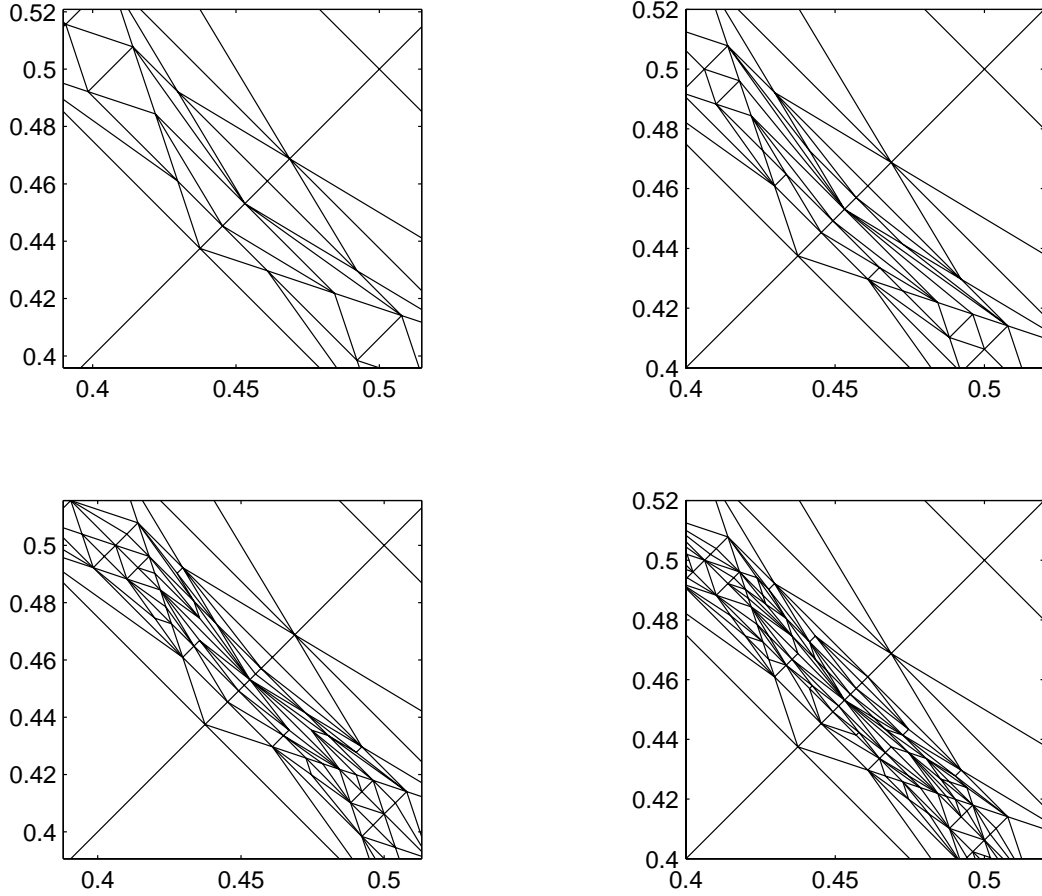


Figure 4.8: Blow up of the next four refinements.

4.2 An example with two layers that intersect

Here we approximate the exact solution $0.5 \sin(\pi x) \sin(\pi y) (\tanh((r - 0.5)/\epsilon) + \tanh((y - x + 0.2)/\epsilon))$ of the elliptic model problem (2.1) using the estimators $\widehat{\Delta e^2}_{W_4}$ and $\widehat{\Delta e^2}_{W_0}$. The maximum aspect ratio in the refined meshes was 80 and 120 respectively, the last meshes contain 2010 and 1820 degrees of freedom respectively, the maximum value of $\Delta e^2 / \widehat{\Delta e^2}_{W_4}$ and $\Delta e^2 / \widehat{\Delta e^2}_{W_0}$ were 1.9 and 6 respectively. For the refinement based on $\widehat{\Delta e^2}_{W_4}$ we got $\frac{\sum_{i=0}^{N_j} \Delta e_i^2}{\|u - u_{V_j}\|_a^2} = 0.7524$.

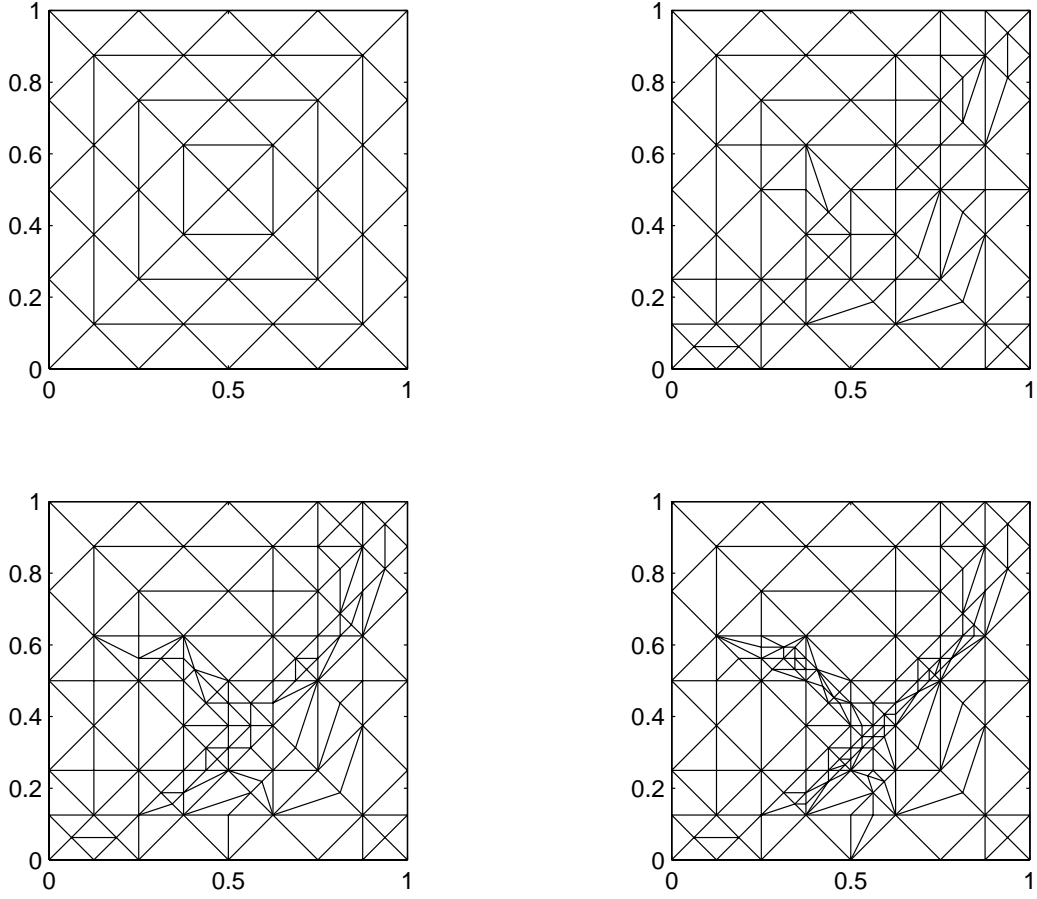
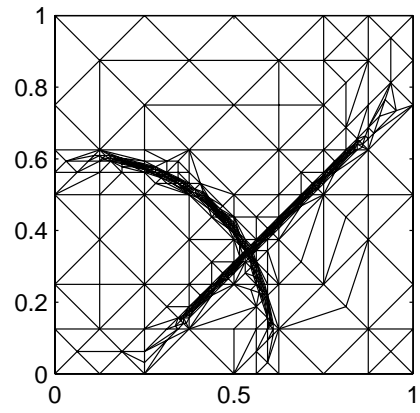
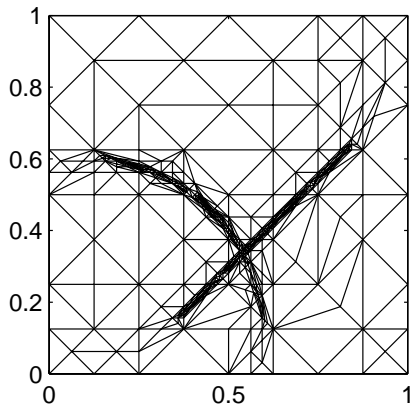
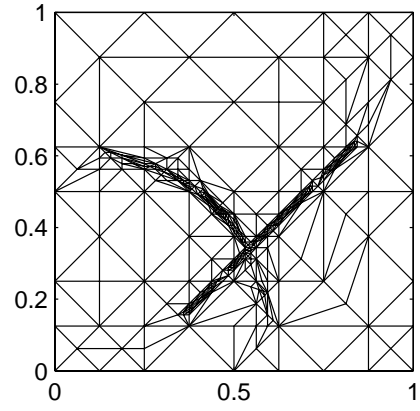
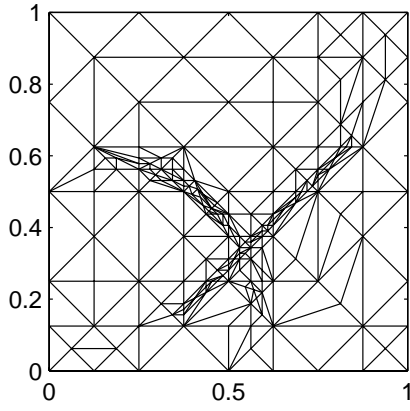


Figure 4.9: Initial mesh and the first three refinements using $\widehat{\Delta e^2}_{W_4}$. The exact solution is $0.5 \sin(\pi x) \sin(\pi y) (\tanh((r - 0.5)/\epsilon) + \tanh((y - x + 0.2)/\epsilon))$, where $r = \sqrt{(x - 0.1)^2 + (y - 0.1)^2}$ and $\epsilon = 0.005$.

Figure 4.10: Next four refinements.



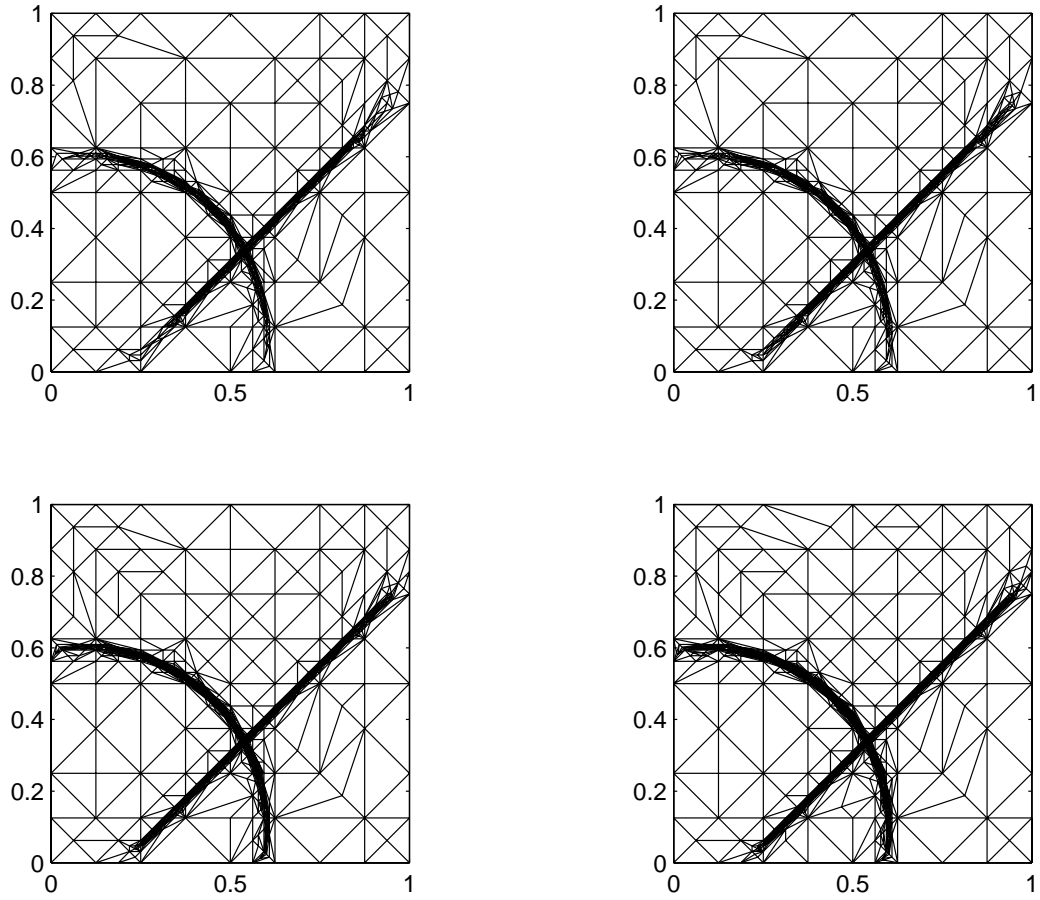


Figure 4.11: Next four refinements. The last one has 2010 degrees of freedom.

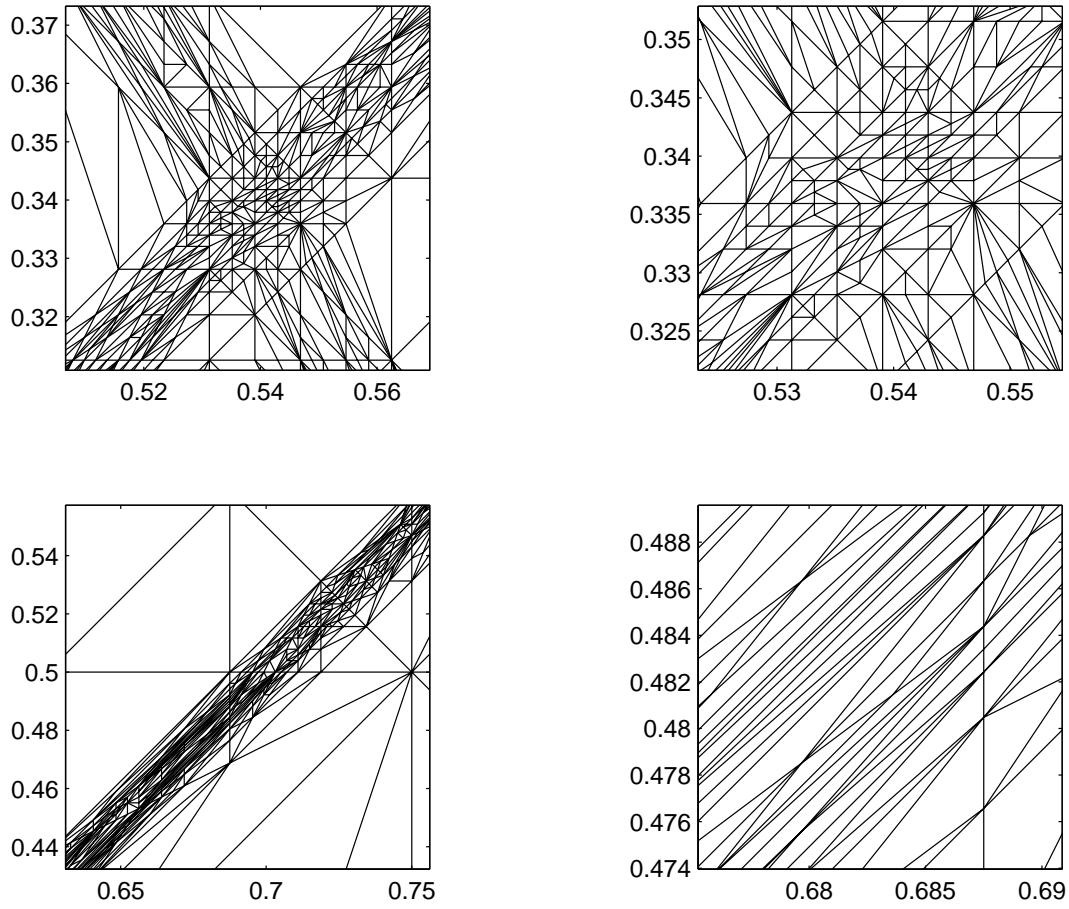


Figure 4.12: Blow up of the last mesh using the estimator $\widehat{\Delta e^2}_{W_4}$.

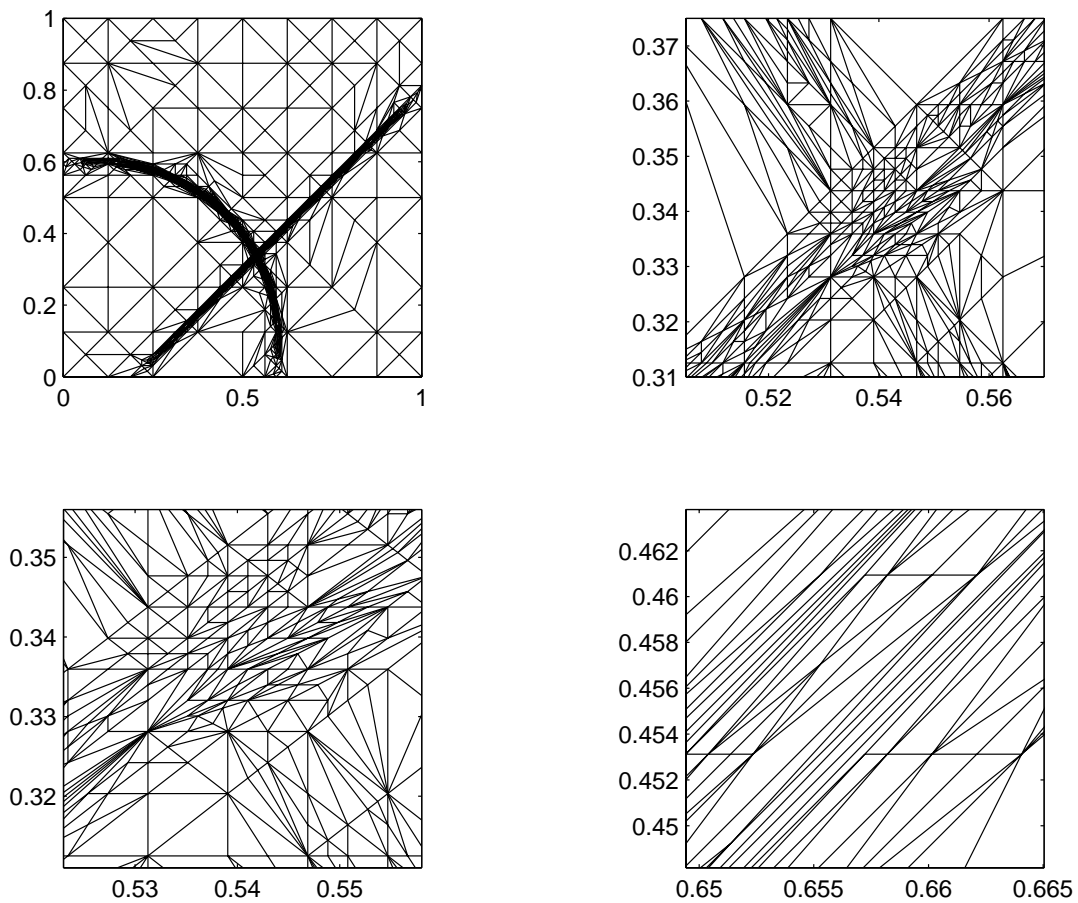


Figure 4.13: The estimator $\widehat{\Delta e^2}_{W_0}$ produced the mesh shown with 1820 degrees of freedom. The other pictures are magnifications of this mesh.

Bibliography

- [1] T. Apel, G. Lube, *Anisotropic mesh refinement in stabilized Galerkin methods*, Numer. Math. 74 (1996), pp. 261-282.
- [2] T. Apel, M. Dobrowolski, *Anisotropic interpolation with applications to the finite element method*, Computing, 47 (1992), pp. 277-293.
- [3] T. Apel, S. Nicaise, *The finite element method with anisotropic mesh grading for elliptic problems in domains with corners and edges*, Math. Meth. Appl. Sci. 21 (1998), pp. 519-549.
- [4] T. Apel, S. Nicaise, J. Schöberl *Crouzeix-Raviart type finite elements on anisotropic meshes*, Preprint SFB393 /99-10, TU Chemnitz-Zwickall, 1999.
- [5] T. Apel, *Interpolation of non-smooth functions on anisotropic finite element meshes*, Preprint SFB393 /97-6, TU Chemnitz-Zwickall, 1997.
- [6] I. Babuska, A. K. Aziz, *On the angle condition in the finite element method*, SIAM Journal of Numerical Analysis, 13 (1976), pp. 214-226.
- [7] I. Babuska, W. C. Rheinboldt, *A posteriori error estimates for the finite element method*, Internat. J. Numer. Methods Engrg., 12 (1978), pp. 1597-1615.
- [8] R.E Bank, A. Weiser, *Some a posteriori error estimators for elliptic partial differential equations*, Mathematics of Computation, 44 (1985), pp 283-301.
- [9] R.E Bank, *PLTMG, a software package for solving elliptic partial differential equations: users' guide 6.0*, Philadelphia: Society for Industrial and Applied Mathematics, 1990.
- [10] J. U. Brackbill, *An adaptive grid with directional control*, J. Comput. Physics, 108 (1993) 38-50.
- [11] S. Brenner, R Scott, *The mathematical theory of finite element methods*, New

- York, Springer-Verlag, 1994.
- [12] M. J. Castro Diaz, F. Hecht, *Anisotropic surface mesh generation*, Report 2672, INRIA, 1995.
 - [13] M. J. Castro Diaz, F. Hecht, B. Mohammadi, *New progress in anisotropic grid adaptation for inviscid and viscous flow simulations*, Proceedings, 4th International Meshing Roundtable, pp. 73-85, Albuquerque, NM, 1995. Sandia National Laboratories.
 - [14] A.W. Craig, O. C. Zienkiewicz, *Adaptive mesh refinement and a posteriori error estimation for the p-version of the finite element method*, *Adaptive computational methods for partial differential equations*, (College Park, MD, 1983) 33-56, SIAM, Philadelphia, PA, 1983.
 - [15] E. F. D’Azevedo, *On adaptive mesh generation in two dimensions*, Proceedings, 8th International Meshing Roundtable, South Lake Tahoe, CA, U.S.A., pp. 109-117, October 1999.
 - [16] E. F. D’Azevedo, *Optimal triangular mesh generation by coordinate transformation*, *SIAM J. Sci. Statist. Comput.*, 12 (4), pp. 755-786, 1991.
 - [17] E. F. D’Azevedo, R. B. Simpson, *On optimal triangular meshes for minimizing the gradient error*, *Numer. Math.*, 59 (1991), pp. 321-348.
 - [18] K. Eriksson, C. Johnson, *Adaptive finite element methods for parabolic problems I*, *SIAM Journal of Numerical Analysis*, 28 (1991), pp. 43-77.
 - [19] J. Goodman, K. Samuelsson, A. Szepessy, *Anisotropic refinement algorithms for finite elements*, Report, NADA KTH, Stockholm, March 1996.
 - [20] C. Johnson, *Numerical Solution of partial differential equations by the finite element method*, New York, Cambridge University Press, 1987.
 - [21] G. Kunert, *A posteriori error estimation for anisotropic tetrahedral and triangular finite element meshes*, Dissertation, 1999, TU Chemnitz.
 - [22] S. Mallat, Z. Zhang, *Matching pursuits with time-frequency dictionaries*, *IEEE Trans. Signal Process.*, 41 (December 1993), pp. 3397-3415.
 - [23] W. Rackowicz, *An anisotropic h-type mesh-refinement strategy*, *Computer Methods in Applied Mechanics and Engineering*, 109 (1993), pp. 169-181.
 - [24] K. Shimada, *Anisotropic triangular meshes of parametric surfaces via close*

- packing of ellipsoidal bubbles*, Proceedings, 6th International Meshing Roundtable, 1997, Park City, Utah.
- [25] K.G. Siebert, *An a posteriori error estimator for anisotropic refinements*, Numer. Math., 73 (1996), pp. 373-398.
- [26] R. B. Simpson, *Anisotropic mesh transformation and optimal error control*, Applied Numerical Mathematics, 1994.
- [27] O. C. Zienkiewicz, J. Wu, *Automatic directional refinement in adaptive analysis of compressible flows*, Int. J. Num. Meth. Eng., 37 (1994), pp. 2189-2210.

