ABSTRACT

Title of Dissertation: On the Implementation of an Accurate and Efficient Solver for Convection-Diffusion Equations

Chin-Tien Wu, Doctor of Philosophy, Nov 2003

Dissertation directed by: Dr. Howard C. Elman
Department of Computer Science

In this dissertation, we examine several different aspects of computing the numerical solution of the convection-diffusion equation. The solution of this equation often exhibits sharp gradients due to Dirichlet outflow boundaries or discontinuities in boundary conditions. Because of the singular-perturbed nature of the equation, numerical solutions often have severe oscillations when grid sizes are not small enough to resolve sharp gradients. To overcome such difficulties, the streamline diffusion discretization method can be used to obtain an accurate approximate solution in regions where the solution is smooth. To increase accuracy of the solution in the regions containing layers, adaptive mesh refinement and mesh movement based on a posteriori error estimations can be employed. An error-adapted mesh refinement strategy based on a posteriori error estimations is also proposed to resolve layers. For solving the sparse linear systems that arise from discretization, goemetric multigrid (MG) and algebraic multigrid (AMG) are compared. In addiiton, both methods are also used as

preconditioners for Krylov subspace methods. We derive some convergence results for MG with line Gauss-Seidel smoothers and bilinear interpolation. Finally, while considering adaptive mesh refinement as an integral part of the solution process, it is natural to set a stopping tolerance for the iterative linear solvers on each mesh stage so that the difference between the approximate solution obtained from iterative methods and the finite element solution is bounded by an a posteriori error bound. Here, we present two stopping criteria. The first is based on a residual-type a posteriori error estimator developed by Verfürth. The second is based on an a posteriori error estimator, using local solutions, developed by Kay and Silvester. Our numerical results show the refined mesh obtained from the iterative solution which satisfies the second criteria is similar to the refined mesh obtained from the finite element solution.

On the Implementation of an Accurate and Efficient Solver for Convection-Diffusion Equations

by

Chin-Tien Wu

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland at College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
Nov 2003

Advisory Committee:

Dr. Howard C. Elman, Chairman/advisor
Dr. Sung W. Lee
Dr.. Jian-Guo Liu
Dr. Ricardo H. Nochetto
Dr. Tobias Von Petersdorff

Dedication

To Wu, Tsing-Hua and Chen, Yi-Mei, my parents

ACKNOWLEDGEMENTS

This thesis would not have been finished without Dr. Howard Elman's the indefinite patient and his valuable suggestions in many research topics. I also would like to thank Dr. Tobias Von Petersdorff, Dr. Nochetto, Ricardo, Dr. Jian-Guo Liu and Dr. Sung W. Lee for serving in my defense committee. I deeply appreciate for all the encouragement from my friends and my family. Finally, I thank my mother Yi-Mei Chen from the bottom of my heart. Without her endless support, I would not have a chance to pursue my research and to see the world in different point of views.

Table of Contents

Chapter 1

Introduction

1.1 Problem Description

The purpose of this dissertation is to study the convection-diffusion equation

$$\begin{aligned} -\epsilon\triangle u + b\cdot\nabla u + cu &= f, \\ u &= g \quad \text{ on } \partial\Omega, \end{aligned} \tag{1.1}$$

where the domain Ω is convex with Lipschitz boundary $\partial\Omega$, and b, c, f are sufficiently smooth, $0 \leq c \ll |b| \leq 1$. In order to ensure existence and uniqueness of the solution, we assume

$$d_1 \geq c - \frac{1}{2}\nabla\cdot b \geq d_0 \geq 0, \quad \text{for some constants } d_0 \text{ and } d_1, \tag{1.2}$$

and

$$\int_{\partial\Omega} g^2(b\cdot n)dS \geq 0. \tag{1.3}$$

When $|b| \gg \epsilon$, the problem is referred to as a convection-dominated flow problem. Otherwise, the problem is diffusion-dominated.

Finite element methods are widely used to compute approximate solutions, especially for complex domains. In our analysis, we always assume the underlying meshes is quasi-uniform, i.e. the minimal angle of all elements in the underlying mesh $\Im_{h_k}$ is bounded well above 0^o and below 180^o. The mesh Peclét number is defined by

$$P_{e_T} = \frac{\|b\|_{\infty,T}\, h_T}{2\epsilon},$$

where $T \in \Im_{h_k}$ with diameter h_T. Here, we will restrict our efforts to convection-dominated flow especially when $P_{e_T} > 1$ for all $T \in \Im_{h_k}$.

1.2 Historical Overview

It is well known that the standard Galerkin discretization of (1.1) yields inaccurate, oscillatory solutions near boundary layers in convection dominated flows and, if the diffusion parameter ϵ is decreased without proportional reduction of the discretization mesh size, then these inaccuracies propagate into regions where the solution is smooth [56]. The streamline diffusion discretization method (SD) introduced by Hughes and Brooks [54] is designed to overcome these problems by introducing a small amount of artificial diffusion in the direction of streamlines. The first mathematical analysis of the SD method was given by Johnson and N$\ddot{a}$vert, who obtained local $O(h^{3/2})$ error estimates in the L^2 norm and global $O(h^{3/2})$ error estimates in a special mesh-dependent so-called SD-norm. The numerical solution obtained from the SD method has the desirable property that the accuracy in regions where the exact solution is smooth will not be degraded as a result of discontinuities and layers in the exact solution [85], [58]. However, the numerical solution obtained from the SD method can be

oscillatory in regions where there are layers, and it may also suffer from overshooting and undershooting. On the other hand, this localization property opens a possibility of reducing oscillation, overshooting and undershooting through local grid refinement. Many modified streamline diffusion methods have been proposed to improve the SD approach by adding shock-capturing term (SD-SC) or crosswind diffusion (SD-CD) [22], [55], [58], [88].

To obtain an accurate finite element solution on a given mesh, usually a so-called quasi-uniform or isotropic mesh is desirable [6]. Delaunay triangulation (DT) is one of the most important algorithms to produce such a triangulation because the DT algorithm maximizes the minimal angle of the triangulation [15], [12]. Mesh operations such as edge swapping and mesh relaxation can also be employed to improve mesh quality [31], [45], [60]. One commom technique to increase the accuracy of the finite element solution is mesh refinement, the so-called h-method. In addition to the regular mesh refinement, Rivara's longest side bisection algorithm (LSB), [83], [84], guarantees that the minimal angle of the refined mesh will not be less than one half of the minimal angle of original mesh. Moreover, the meshes generated by LSB are nested. As a result, meshes from both regular refinement and LSB refinement possess shape regularity and are suitable for multigrid algorithms.

Another grid adaptation technique is based on moving meshes. Mesh movement derived from equidistribution principle and direct minimization have been studied by many researchers such as Azevedo [28], Baines [9], Felippa [43], Huang and Russell [52], Tourigny [93], [94], and literatures cited therein. The idea to make use of a posteriori error estimator in mesh movement is presented by Bank and Smith

[10]. This approach requires the computation of approximate second derivatives for all elements and solutions of a local optimization problem at each node which is complicated and time-consuming. Here, we examine a mesh movement strategy based on equidistributing an a posteriori error estimator. How mesh movement can improve the accuracy of the numerical solutions in the adaptive process is still not clear. In particular, for convection-diffusion problems, node movement may be in the wrong direction, when approximate solutions contain serious oscillations in regions containing layers. As a result, mesh movement may actually degrade the quality of the underlying meshes and the accuracy of the numerical solutions. Nonetheless, our numerical studies suggests this simple strategy for mesh movement can significantly improve the accuracy of finite element solutions.

There are cases in which anisotropic meshes,consisting of long thin triangles, may produce more accurate solutions [76], [81] than the isotropic meshes. For the convection-diffusion problems, anisotropic mesh adaptation including Shishkin meshes have been shown to be effective [4], [23], [27], [30], [65]. However, rigorous theoretical analysis on anisotropic meshes has not been fully developed. Even though we shall not pursue any theoretical results in this area, our error-adapted mesh refinement algorithm in section 3.5 is capable of producing long-thin triangles in the layer region which cluster nodes in these regions. Moreover, in contrast to the moving mesh strategy where the nested grid structures can't be maintained and interpolation between grids has to be computed for multigrid solvers, the grids generated by the error-adapted refinement algorithm is ready to be used in multigrid solvers without any extra computation cost.

For an adaptive refinement procedure to succeed, reliable and efficient a posterori error estimators are needed. For the reliability and efficiency of a posteriori error estimators, a standard measure is the so-called effectivity index, defined as

$$eff = \frac{estimated\,error}{true\,error}$$

An estimator is called asymptotically exact if its effectivity index converges to 1 when the mesh size approaches 0. If the effectivity index is much smaller than 1, the estimator is under-estimating the error. On the other hand, if the effectivity index is much greater than 1, the estimator is over-estimating the error. If the estimator does not under-estimate or over-estimate the error globally, then the estimator is reliable, meaning the error on the global domain can be properly controlled by the estimator. If the estimator does not under-estimate or over-estimate the error locally, then the error estimator is efficient, meaning the estimator is able to pinpoint exactly where the error is large and where the error is small. For two-dimensional problems, several estimators have been shown to be asymptotically exact when used on uniform meshes provided the solution of the problem is smooth enough [7], [33], [34]. Estimators based on computing residuals, so-called residual-type estimators, and estimators based on solving a local Dirichlet problem, so-called Dirichlet-type estimators, were introduced by Babuška and Rheinboldt [8]. Estimators based on solving a local Neumann problem, so-called Neumann-type estimators, were first given by Bank and Weiser [11]. These estimators have been studied by many researchers such as Ainsworth [2], Johnson, Eriksson [40] [57] , Kay and Silvester [59] and Verfürth [96] [97]. The Zienkiewicz-Zhu (ZZ) type of estimators based on recovery of gradient and Hessian are also well developed, see [3], [73], and articles cited therein.

For convection-diffusion problems, numerical results in [72] show the residual-type error estimator and the ZZ estimator are not as reliable as the Neumann-type estimator. Here, our numerical results also show that the Neumann-type estimator introduced by Kay and Silvester is more reliable than Verfürth's residual-type estimator. One of our goals is to understand how the quality of estimators may degrade if we replace the exact finite element solution by approximate iterative solution. In other words, we are interested in finding the largest stopping tolerance for the iterative solver, such that the reliability and efficiency of error estimator will not change too much when these estimators are computed from approximate solutions obtained from iterative methods.

Multigrid methods (MG) are among the most efficient methods for solving the linear systems arising from discretization of elliptic partial differential equations. There has been intensive research on the convergence of MG since it was introduced by Fedorenko [42]. For symmetric positive-definite elliptic problems, thanks to many researchers, such as Bank, Braess, Bramble, Brandt, Dupont, Hackbusch, Mandel and McCormick, etc, the convergence theory has matured. However, for singular perturbation problems, the development of theoretical analysis is far less advanced. The difficulties arise from the weak coercivity and poor regularity in these type of problems.

The major ingredients for convergence analysis of MG are called the *approximation property* and the *smoothing property*. One approach for convergence analysis is the so-called $compact\ perturbation\ technique$, which relies on a strong approximation property and treats the lower order terms as a small perturbation of the symmetric pos-

itive define term. The technique has been successfully applied to diffusion-dominated flow problems and Bramble, Pasciak, Wang, Xu have shown robust MG uniform convergence [17], [18], [19], [20], [99]. In these studies, uniform convergence of MG can be established with one step of standard Jacobi or Gauss-Seidel smoothing even without regularity assumptions. For convection-dominated flow problems, this approach requires coarse grids with very small grid size, $h_{coarse} \ll \epsilon$, which is usually not valid in practical computations.

With realistic coarse grids in mind, matrix-dependent prolongation and restriction operators have been proposed by Dendy [29], De Zeeuw [105], Reusken [79] and Wesseling [100] to enhance the approximation property on uniform meshes. It is not clear how to generalize these results on complex domains where one can only use unstructured meshes. The algebraic multigrid method developed by Ruge and Stüben [86], [92], is readily adapted to such applications. Convergence of AMG is established when the coefficient matrix is a symmetric M-matrix. This is typically not the case for the convection-diffusion problem, but numerical studies in [92] also suggest AMG is still applicable. Both matrix-dependent operators and AMG require computing correction operators on coarse grids. These seem not to be a natural choice of methods if adaptive process is involved.

Another approach requires a strong smoothing property to compensate for poor approximation property in this type of problems. In this direction, it is very important to find a robust smoother. Robust smoothers such as the Gauss-Seidel method with flow-oriented ordering and the incomplete LU factorization (ILU) method have been studied by many researchers such as Bey [13] [14], Chernesky and Elman [37] [38],

Hackbusch and Probst [51], Wesseling [100] and Wittum [102]. Recently, researchers such as Reusken, Pflaum prove MG convergence in L^2 with the help of special gridding techniques such as semi-coarsening [75], [80]. Szepessy shows MG convergence in L^1 by residual damping through large smoothing steps [74]. Moreover, Ramage have demonstrated that MG convergence rates can be significantly improved if the SD discretization is employed with an optimal stabilization parameter [77]. Here, we would like to study MG convergence of the SD-discretized flow problems. We prove some MG convergence results for a simple constant flow problem when mesh size $h \gg \sqrt{\epsilon}$, where only standard bilinear prolongation and restriction operators are considered in MG algorithm.

For problems containing recirculating flows, it is not easy to obtain a robust smoother. As a result, MG fails to converge without special treatments on discretization methods and prolongation operators [104], [105]. However, some numerical experiments indicate that MG is a robust preconditioner in Krylov subspace solver [70]. Here, we would also like to investigate whether MG and AMG, as preconditioners of GMRES solver, are still robust in these convection-diffusion problems on adaptively refined unstructured grids.

1.3 Dissertation Outline

First we review many aspects of computing accurate finite element solutions for convection-diffusion equations and discuss some difficulties associated with using multigrid for solving the linear system that arise from discretization of (1.1). In Chapter 2, linear discretization methods are studied. We briefly review two finite element

methods, the standard Galerkin method and the streamline diffusion method. Some fundamental properties of the solutions from both methods are also presented. Our numerical results show that the Galerkin method produces oscillatory solutions globally, whereas the solution obtained from streamline diffusion method are oscillatory only in the regions where there are layers. In Chapter 3, a posteriori error estimations as well as a mesh movement strategy and an error-adapted mesh refinement strategy based on these estimations are introduced. First, theoretical results of residual-type of a posteriori error estimator by Verfürth and Neumann-type of a posteriori error estimator by Kay and Silvester are reviewed. Then a comparison of reliability and effectivity of both estimators is given. Numerical results for a mesh movement strategy, based on equidistribution of the error estimators, are also shown here after a brief overview on the mesh movement strategies based on equidistribution principles. In Section 3.5, the error-adapted mesh refinement algorithm is presented. In Chapter 4, the algorithm and convergence of several linear iterative solvers are studied. For stationary iterative methods, Jacobi, Gauss-Seidel, line Jacobi and line Gauss-Seidel as well as the Krylov subspace iterative method, GMRES, are presented. For multigrid methods, geometric multigrid (MG) and algebraic multigrid (AMG) algorithm are presented. We prove geometric multigrid will converge when the mesh size satisfies $h \gg \sqrt{\epsilon}$ for a simple constant flow problem on uniform mesh. For more difficult problems such as those with circulating flows, the performance of MG, AMG and GMRES with GS, MG and AMG preconditioners are compared. In Chapter 5, stopping criteria of the iterative linear solver in adaptive mesh refinement process are studied. We develop two stopping criteria, one associated with Verfürth's residual-type error indicator and the other associated with Kay and Silvester's Neumann-type error indicator. We show that it is necessary for the iterative solution to satisfy our stopping criteria in order

to ensure that the error arising from the iterative solution is bounded by the a posterior error estimations. Our numerical results show error estimators computed from the multigrid solution, which satisfy our stopping criteria, produce almost identical mesh refinements as error estimators computed from exact finite element solution. In Chapter 6, we draw some conclusions.

1.4 Notation and Terminology

The following notations are used in this thesis.

- The notation $x \preceq y$ for $x, y \in R$ is defined as there is a constant $0 < c \ll \infty$ such that $x \leq y$.

- Let $(\cdot, \cdot)$ be the inner product in $L^2(\Omega)$ defined by $(f, g) = \int_\Omega fg$

  The notation $\|f\|_k$ and $|f|_k$ denotes the usual Sobolev norm and semi-norm over the global domain Ω, defined by $\|f\|_k = \left(\sum_{|\alpha|\leq k} \|D^\alpha f\|_0^2\right)^{1/2}$ and $\|f\|_k = \left(\sum_{|\alpha|=k} \|D^\alpha f\|_0^2\right)^{1/2}$, respectively, where $\|f\|_0^2 = (f, f)$, for $f \in H^k$. Also, $\|f\|_{\Omega_0,k} = \left(\sum_{|\alpha|\leq k} \|D^\alpha f\|_0^2\right)^{1/2}$ is the Sobolev norm of f defined on a sub-domain $\Omega_0 \subset \Omega$.

- Let $< \cdot, \cdot >$ denote the Euclidean inner product on R^n.

  The notation $\|x\|$ is defined as $\|x\| =< x, x >^{1/2}$ for $x \in R^n$.

- The L^2 norm of a given matrix is defined as

$$\|A\| = \sup_{x \in R^n} \frac{| < Ax, x > |}{\|x\|}$$

- Let A be a matrix $A = (a_{ij})$, $1 \leq i, j \leq n$. If $x^T Ax > 0$ for all nonzero $x \in R^n$, A is called positive define.

  If $a_{i,j} \geq 0$ for all i and j, then A is called a non-negative matrix and is denoted as $A \geq 0$.

  If A is nonsingular, $a_{ij} \leq 0$ for $j \neq i$ and $A^{-1} \geq 0$, A is called a M-matrix.

  If there exist permutation matrix P such that

$$\mathrm{P^T AP} = \begin{bmatrix} \mathrm{A_{1,1}} & \mathrm{A_{1,2}} \\ 0 & \mathrm{A_{2,2}} \end{bmatrix},$$

A is called reducible. If no such permutation exists, A is called irreducible.

If $\sum_{j\neq i}|a_{i,j}| \leq |a_{i,i}| \ \ \forall i$, A is called diagonal dominant.

If A is diagonal dominant and $\sum_{j\neq i}|a_{i,j}| < |a_{i,i}|$ for some i, A is called weakly diagonal dominant.

- Let d denote the function that measures the diameter of a given domain. Let Ω be a given domain and $\Im_h$ be a mesh such that

$$\max\{d(T) : T \in \Im_h\} \leq hd(\Omega).$$

The mesh $\Im_h$ is called quasi-uniform if there exists $r > 0$ such that

$$\min\{d(B_T) : T \in \Im_h\} \geq rhd(\Omega),$$

where B_T is the largest ball contained in T.

- Let $N_i, i = 1 \cdots m$ denote the nodes of $\Im_h$. Let ϕ_i be the nodal basis function at node N_i. The nodal interpolant I is defined as

$$Iu = \sum_{i=1}^{m} u(N_i)\phi_i$$

$S_i = supp(\phi_i)$. Let π_i be the L^2 orthogonal projection onto the piecewise linear function space in S_i. The quasi-interpolant I is defined as

$$Iu = \sum_{i=1}^{m} \pi_i u(N_i)\phi_i.$$

Let $\mathcal{E}$ denote the set of edges in $\Im_h$. For any element $T \in \Im_h$ and edge $E \in \mathcal{E}$,

$$\omega_T = \bigcup_{\emptyset\neq T'\cap T\in\mathcal{E}} T', \ \tilde{\omega}_T = \bigcup_{T'\cap T\neq\emptyset} T', \ \omega_E = \bigcup_{E\subset T'} T', \ \omega_i = \bigcup_{N_i\in T'} T'$$

Chapter 2

Linear Discretization Methods

In this chapter, we review two finite element methods for discretizing the convection-diffusion equation (1.1), the standard Galerkin method (GK) and the streamline-diffusion finite element method (SDFEM). We consider finite element techniques with isoparametric bilinear elements for the convection-diffusion problem with small viscosity ϵ. We illustrate the solution behavior in both analysis and numerical experiments on some model problems.

A weak solution of (1.1)-(1.2) is given by $u \in H^1(\Omega)$ such that

$$B(u, v) = F(v), \ \forall v \in H_0^1(\Omega), \tag{2.1}$$

where the bilinear form is defined as

$$B(u, v) = \epsilon(\nabla u, \nabla v) + (b \cdot \nabla u, v) + (cu, v), \tag{2.2}$$

and the linear functional on the right hand side is defined as

$$F(v) = (f, v) + \int_{\partial\Omega} (gv)n \cdot dS. \tag{2.3}$$

The existence and uniqueness of the weak solution are established by the Lax-Milgram theorem since the bilinear form B is coercive and continuous on $H^1(\Omega)$.

Moreover, when f and g are smooth enough, a smoother solution $u \in H^2$ can be obtained if the underlying domain Ω is convex [48].

Lemma 2.0.1 (Continuity) *For all $u \in H^1(\Omega)$ and $v \in H_0^1(\Omega)$, there exists a constant $\Gamma > 0$ such that*

$$|B(u,v)| \leq \Gamma \|u\|_1 \|v\|_1 \tag{2.4}$$

Proof: From (2.2), we have

$$\begin{aligned} |B(u,v)| &= |\epsilon(\nabla u, \nabla v) + (u, b\cdot\nabla v) + ([2(c - \frac{1}{2}\nabla\cdot b) - c]u, v)| \\ &\leq \epsilon|u|_1|v|_1 + \|u\|_0 |v|_1 + (1+2d_1)\|u\|_0 \|v\|_0, \text{ , by (1.2)} \\ &\leq \Gamma \|u\|_1 \|v\|_1, \end{aligned}$$

where $\Gamma = \epsilon + 2(1+d_1)$.

□

Lemma 2.0.2 (Coercivity) *For all $u \in H_0^1(\Omega)$, there exist constant $\gamma > 0$ such that*

$$|B(u,u)| \geq \gamma \|u\|_1^2 \tag{2.5}$$

Proof: By Green's formula,

$$\int_{\partial\Omega} u^2 b\cdot n dS = \int_\Omega \nabla\cdot(u^2 b) dxdy = \int_\Omega (\nabla\cdot b)u^2 dxdy + 2\int_\Omega (b\cdot\nabla u) u dxdy$$

Therefore,

$$\begin{aligned} B(u,u) &= \epsilon\int_\Omega \nabla u\cdot\nabla u dxdy + \int_\Omega (c - \frac{1}{2}\nabla\cdot b)u^2 dxdy + \frac{1}{2}\int_{\partial\Omega} g^2 b\cdot n dS \\ &\geq \epsilon|u|_1 + d_0\|u\|_0, \text{ by (1.2) and (1.3)}, \\ &\geq \gamma\|u\|_1, \text{ for some } \gamma > 0. \end{aligned}$$

□

Remark 2.0.3 *If the energy norm, defined as* $|||u||| = \epsilon \|\nabla u\|_0 + d_0 \|u\|_0$*, is considered, the continuity and coercivity can be also estimated in terms of energy norm as in the following: for all* $u, v \in H_0^1(\Omega)$*,*

$$|B(u,v)| \leq |||u||| \, |||v||| + \epsilon^{-1/2} \, |||u||| \, \|v\|_0 \preceq \frac{1}{\epsilon} \, |||u||| \, |||v||| \tag{2.6}$$

and

$$B(u,u) \geq |||u|||^2 . \tag{2.7}$$

Moreover, the following lemma gives us an estimation of the regularity of the weak solution in terms of given data. For problems with exponential boundary layers, this estimation is sharp as mentioned in Remark 1.17 in [85]

Lemma 2.0.4 *If the weak solution* $u \in H^2(\Omega)$ *and* $u|_{\partial\Omega} = 0$*, then the following inequality holds.*

$$\epsilon^{3/2} \|u\|_2 + \epsilon^{1/2} \|u\|_1 + \|u\|_0 \leq C \|f\|_0 , \tag{2.8}$$

for some constant $C > 0$*.*

Proof: see Lemma 1.18 in p. 186 [85].

□

2.1 Galerkin Discretization

Assume we are given a quasi-uniform mesh $\Im_h$ with node points x_1,...,x_n. Let V_h be the finite-dimensional subspace consisting of piecewise linear or bilinear functions defined on $\Im_h$. The Galerkin finite element method seeks an approximate solution u_h of the weak solution u in V_h which satisfies

$$B_{gk}(u_h, v_h) = F_{gk}(v_h),\ \forall v_h \in V_h, \tag{2.9}$$

where $B_{gk} = B$ and $F_{gk}(v) = (f_h, v_h) + \int_{\partial\Omega}(g_h v_h)n \cdot dS$.

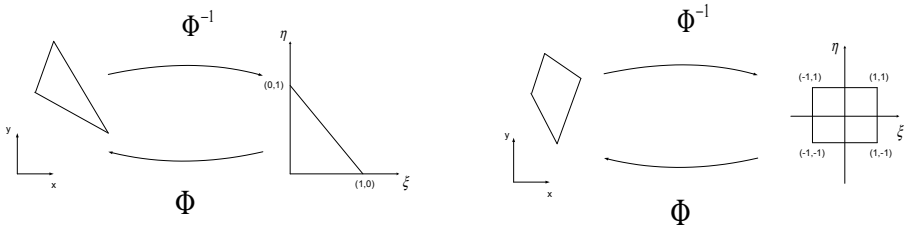
Clearly, each function $v_h \in V_h$ has a unique representation $v = \sum_{i=1}^{n} v_h^i \phi_i$, where v_h^i is the nodal value and ϕ_i is the linear nodal basis function at node x_i satisfying

$$\phi_i(x) = \begin{cases} 1 & \text{if } x = x_i \\ 0 & \text{if } x \neq x_i \end{cases}$$

Using (2.2) and (2.3), (2.9) can be rewritten as

$$\sum_{e \in \Im_h} \epsilon \underbrace{\int_e \nabla u_h \nabla v_h dxdy}_{I} + \underbrace{\int_e (b \cdot \nabla u_h) v_h dxdy}_{II} + c \underbrace{\int_e u_h v_h dxdy}_{III} = \sum_{e \in \Im_h} \underbrace{\int_e f v_h dxdy}_{IV} \tag{2.10}$$

Each term is then computed elementwise. The computation is done on a reference element ê instead of on the actual element through an isoparametric mapping Φ.



(a) Triangular Element

(b) Rectangular Element

Let ξ and η be the reference coordinates. $\Phi : (\xi, \eta) \mapsto (x, y)$ is defined by

$$(x, y) = \Phi(\xi, \eta) = \sum_{i=1}^{d} (x_i, y_i)\chi_i(\xi, \eta), \tag{2.11}$$

where d is the degree of freedom of the associated element and χ_i, i=1...d, is the linear element nodal basis function of ê. Moreover, from isoparametric formulation, u_h and

v_h can also be represented as

$$u_h = \sum_{i=1}^{d} u_h^i \chi_i(\xi, \eta) \text{ and } v_h = \sum_{i=1}^{d} v_h^i \chi_i(\xi, \eta) \tag{2.12}$$

on each element, where u_h^i ad v_h^i are the function values on node x_i.

For linear triangular elements, $d = 3$ and

$$\begin{aligned} \chi_1(\xi, \eta) &= 1 - \xi - \eta \\ \chi_2(\xi, \eta) &= \xi \\ \chi_3(\xi, \eta) &= \eta. \end{aligned} \tag{2.13}$$

For bilinear rectangular elements, $d = 4$ and

$$\begin{aligned} \chi_1(\xi, \eta) &= \frac{1}{4}(1 - \xi)(1 - \eta) \\ \chi_2(\xi, \eta) &= \frac{1}{4}(1 + \xi)(1 - \eta) \\ \chi_3(\xi, \eta) &= \frac{1}{4}(1 + \xi)(1 + \eta) \\ \chi_4(\xi, \eta) &= \frac{1}{4}(1 - \xi)(1 + \eta). \end{aligned} \tag{2.14}$$

The Jacobian matrix J arising from coordinate transformation is

$$J = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{bmatrix} \tag{2.15}$$

and can directly be computed from (2.11) and the above definitions of the nodal basis functions. Since (I) and (II) can be rewritten as

$$\int_e \nabla u_h \nabla v_h = \int_{\hat{e}} [\frac{\partial v_h}{\partial \xi}, \frac{\partial v_h}{\partial \eta}] \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \end{bmatrix}^T \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \end{bmatrix} \begin{bmatrix} \frac{\partial u_h}{\partial \xi} \\ \frac{\partial u_h}{\partial \eta} \end{bmatrix} |J| d\xi d\eta, \tag{2.16}$$

$$\int_e (b \cdot \nabla u_h) v_h = \int_{\hat{e}} v_h [b_1, b_2] \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \end{bmatrix} \begin{bmatrix} \frac{\partial u_h}{\partial \xi} \\ \frac{\partial u_h}{\partial \eta} \end{bmatrix} |J| d\xi d\eta, \text{and}, \tag{2.17}$$

and the following relationships holds

$$\frac{\partial \xi}{\partial x} = |J|^{-1}\frac{\partial y}{\partial \eta},\ \frac{\partial \xi}{\partial y} = -|J|^{-1}\frac{\partial x}{\partial \eta},\ \frac{\partial \eta}{\partial x} = -|J|^{-1}\frac{\partial y}{\partial \xi},\ \frac{\partial \eta}{\partial y} = |J|^{-1}\frac{\partial x}{\partial \xi}, \tag{2.18}$$

the associated element discrete matrices can be computed directly from (2.11), (2.12), and the definition of the nodal basis functions (2.13 and (2.14). Similarly, (III) and the righthand side of (2.10) can be rewritten as

$$\int_e u_h v_h = \int_{\hat{e}} v_h u_h |J| d\xi d\eta, \tag{2.19}$$

and

$$\int_e f_h v_h = \int_{\hat{e}} v_h f_h |J| d\xi d\eta, \tag{2.20}$$

respectively. Clearly, the discrete matrix of (2.19) and (2.20) can also be computed by the same way. Let $\mathcal{H}_e$ be the discrete matrix of (2.16), $\mathcal{C}_e$ be the discrete matrix of (2.17), and $\mathcal{M}_e$ be the discrete matrix of (2.19) and (2.20). Now, (2.9) can be written in the following matrix form

$$(\epsilon\mathcal{H} + \mathcal{C} + c\mathcal{M})u_h = \mathcal{M}f, \tag{2.21}$$

where $\mathcal{H} = \sum_{e\in\Im_h} \mathcal{H}_e$, $\mathcal{C} = \sum_{e\in\Im_h} \mathcal{C}_e$, and $\mathcal{M} = \sum_{e\in\Im_h} \mathcal{M}_e$. The matrix on the lefthand side of (2.21) is usually called the stiffness matrix and the matrix on the righthand side is called the mass matrix.

The usual stencil notation for the stiffness matrix and mass matrix at each node can be obtained by assembling the element matrices of neighbor elements of that node.

On uniform triangular meshes, the stencil notation is:

$$\mathcal{H} \sim \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}, \mathcal{C} \sim \frac{h}{6} \times \begin{bmatrix} -b_1 + b_2 & b_1 + 2b_2 & 0 \\ -(2b_1 + b_2) & 0 & 2b_1 + b_2 \\ 0 & -(b_1 + 2b_2) & b_1 - b_2 \end{bmatrix}, \text{ and}$$

$$\mathcal{M} \sim \frac{h^2}{12} \times \begin{bmatrix} 1 & 1 & 0 \\ 1 & 6 & 1 \\ 0 & 1 & 1 \end{bmatrix}.$$

On uniform rectangular mesh, the stencil notation is:

$$\mathcal{H} \sim \frac{1}{3} \times \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}, \mathcal{C} \sim \frac{h}{12} \times \begin{bmatrix} -b_1 + b_2 & 4b_2 & b_1 + b_2 \\ -4b_1 & 0 & 4b_1 \\ -(b_1 + b_2) & -4b_2 & b_1 - b_2 \end{bmatrix}, \text{ and}$$

$$\mathcal{M} \sim \frac{h^2}{36} \times \begin{bmatrix} 1 & 4 & 1 \\ 4 & 16 & 4 \\ 1 & 4 & 1 \end{bmatrix}.$$

The stiffness matrix arises from the Galerkin discretization can be denoted as

$$A_{GK} = \epsilon\mathcal{H} + \mathcal{C} + c\mathcal{M}.$$

Since we consider $\epsilon \ll h$ and $c < |b|$, clearly, $\mathcal{C}$ is the dominating term. Standard Fourier analysis suggests that the solution will contain large oscillatory modes. A detail analysis can be found at section 3.5 of [39].

Remark 2.1.1 *For $\nabla \cdot b = 0$, we have*

$$(b \cdot \nabla u, v) = -(u, b \cdot \nabla v) - ((\nabla \cdot b)u, v) = -(b \cdot \nabla v, u).$$

It follows that the matrix $\mathcal{C}$ is skew-symmetric in this case. For our problem,

$$|(b \cdot \nabla u, v) + (b \cdot \nabla v, u)| \leq 2(c - d_0)(u, v).$$

Since $|c| < 1$ and the mass matrix $\mathcal{M}$ from (u, v) is $O(h^2)$, the symmetric part of $\mathcal{C}$ is in the order of h^2. Therefore, $\mathcal{C}$ is nearly skew-symmetric. Moreover, for small ϵ, A_{GK} is also nearly skew-symmetric.

Lemma 2.1.2 *Let $v \in H^m(\Omega)$. The interpolant v^I satisfies*

$$\left\|v - v^I\right\|_k \preceq h^{m-k}|v|_m, \tag{2.22}$$

for $0 \leq k \leq m$ where $m = 0, 1$ or 2.

Proof: See [56] Theorem 4.2 or [21] Theorem 4.4.20.

□

Now, we can prove an a priori error estimation for the Galerkin finite element solution.

Theorem 2.1.3 (A priori error estimation) *If u_h satisfies (2.9) and u is the weak solution of (2.2), then there exist a constant C, independent with h and ϵ, such that*

$$\|u - u_h\|_1 \leq C(1 + \frac{h}{\epsilon})h|u|_2. \tag{2.23}$$

Proof: From coercivity, we have

$$\gamma \|u - u_h\|_1^2 \leq B_{gk}(u - u_h, u - u_h) = B_{gk}(u - u_h, u - u^I) + B_{gk}(u - u_h, u^I - u_h). \tag{2.24}$$

Since $B_{gk}(u-u_h, u^I-u_h) = 0$, from the orthogonality property of the Galerkin discretization, we only need to estimate $B_{gk}(u-u_h, u-u^I)$.

$$\begin{aligned}
|B(u-u_h, u-u^I)| &\leq \epsilon|u-u_h|_1|u-u^I|_1 + |u-u_h|_1 \left\|u-u^I\right\|_0 \\
&+ c\left\|u-u_h\right\|_0 \left\|u-u^I\right\|_0, \text{ by the Cauchy-Schwarz inequality,} \\
&\leq (c_1\epsilon h + c_2h^2 + c_3h^2)\left\|u-u_h\right\|_1 |u|_2, \\
&\quad \text{by Lemma 2.1.2 and the Poincaré inequality,}
\end{aligned}$$

for some constants, c_1, c_2, $c_3 > 0$. As a result, (2.24) implies

$$\begin{aligned}
\|u-u_h\|_1^2 &\leq \frac{1}{r}(c_1\epsilon h + c_2h^2 + c_3h^2)\left\|u-u_h\right\|_1 |u|_2 \\
, &= \frac{1}{r}\epsilon(c_1 + c_2\frac{h}{\epsilon} + c_3\frac{h}{\epsilon})h\left\|u-u_h\right\|_1 |u|_2
\end{aligned}$$

Recall that $\gamma = O(\epsilon)$. Therefore, we have

$$\left\|u-u_h\right\|_1 \preceq C(1+\frac{h}{\epsilon})h|u|_2,$$

for some constant C independent with h and ϵ.

□

The estimate (2.23) shows the Galerkin finite element solution u_h converges to the weak solution u with error of $O(h^2)$ in H^1 norm when $h \gg \epsilon$. However, the fact that the constant C is proportional to $\frac{1}{\epsilon}$, for $h \gg \epsilon$, indicates the upper bound is very poor unless u is very smooth, namely $|u|_2 \ll 1$. Unfortunately, for the convection-dominated flow problems, one can only bound $|u|_2$ in the order of $\epsilon^{-\frac{3}{2}}$ as shown in Lemma 2.0.4 or ϵ^{-1} when neither an outflow nor an inflow boundary present ([85] p.180-186).

2.2 Streamline Diffusion Discretization

As defined in [56] p.185, the streamline diffusion finite element method (SDFEM) seeks an approximate solution in $(V_h, |||\cdot|||_{sd})$ which satisfies

$$B_{sd}(u_h, v_h) = F_{sd}(v), \ \text{, for all } v \in V_h, \tag{2.25}$$

where

$$B_{sd}(u_h, v_h) = B_{gk}(u_h, v_h) + \sum_{T \in \Im_h} \delta_T (b \cdot \nabla u_h + c u_h, b \cdot \nabla v_h)_T, \tag{2.26}$$

and

$$F_{sd}(v_h) = (f_h, v_h) + \sum_{T \in \Im_h} (f, \delta_T b \cdot \nabla v_h). \tag{2.27}$$

Here, δ_T is the stabilization parameter and $|||\cdot|||_{sd}$ is defined as follows:

$$|||v|||_{sd} = (\epsilon \left\|\nabla v\right\|_0^2 + \sum_{T \in \Im_h} \delta_T \left\|b \cdot \nabla v\right\|_{0;T}^2 + d_0 \left\|T\right\|_0^2)^{1/2}, \ \forall v \in V_h.$$

Furthermore, the SDFEM discretization matrix of (2.26) has the following stencil form

$$A_{SD} = A_{GK} + \bar{\mathcal{C}} + \bar{\mathcal{M}}, \tag{2.28}$$

where $\bar{\mathcal{M}} = \delta_T \mathcal{C}^T$,

$$\bar{\mathcal{C}} \sim \delta_T \times \begin{bmatrix} b_1 b_2 & -(b_1 b_2 + b_2^2) & 0 \\ -(b_1^2 + b_1 b_2) & 2(b_1^2 + b_2^2 + b_1 b_2) & -(b_1^2 + b_1 b_2) \\ 0 & -(b_1 b_2 + b_2^2) & b_1 b_2 \end{bmatrix}, \quad \text{for triangular element}$$

and

$$\bar{\mathcal{C}} \sim \delta_T \times \begin{bmatrix} -\frac{1}{6}(b_1^2 + b_2^2) + \frac{1}{2} b_1 b_2 & \frac{1}{3} b_1^2 - \frac{2}{3} b_2^2 & -\frac{1}{6}(b_1^2 + b_2^2) - \frac{1}{2} b_1 b_2 \\ -\frac{2}{3} b_1^2 + \frac{1}{3} b_2^2 & \frac{4}{3}(b_1^2 + b_2^2) & -\frac{2}{3} b_1^2 + \frac{1}{3} b_2^2 \\ -\frac{1}{6}(b_1^2 + b_2^2) - \frac{1}{2} b_1 b_2 & \frac{1}{3} b_1^2 - \frac{2}{3} b_2^2 & -\frac{1}{6}(b_1^2 + b_2^2) + \frac{1}{2} b_1 b_2 \end{bmatrix}, \quad \text{for rectangular element.}$$

Notice that the stabilization term $\bar{\mathcal{C}}$ is in the same order $O(h)$ as the skew-symmetric $\mathcal{C}$ in A_{GK}. With the help of a proper choice on the stabilization parameter δ_T, it can be shown that the SDFEM solutions no longer suffer from large oscillation [39]. In the following, we show the existence of the SDFEM solution and derive the a priori error bound for the SDFEM solution. First let's show the coercivity of B_{sd}.

Theorem 2.2.1 *[Coercivity] If* $0 < \delta_T \leq \frac{1}{2}\frac{d_0}{c_T^2}$, *where* $c_T = \max|c|$ *for each* $T \in \Im_h$, *then*

$$B_{sd}(v,v) > \frac{1}{2}\left|\left|\left|v\right|\right|\right|_{sd}^2 \forall v \in V_h. \tag{2.29}$$

Proof: By Green's formula, (2.2) and (2.19) imply

$$\begin{aligned} B_{sd}(v,v) > \epsilon|v|_1^2 + d_0\left\|v\right\|_0^2 + \sum_{T\in\Im_h}\delta_T\left\|b\cdot\nabla v\right\|_{0,T}^2 \\ + \sum_{T\in\Im_h}\delta_T(cv, b\cdot\nabla v)_T, \end{aligned} \tag{2.30}$$

for any $v \in V_h$. Since

$$\begin{aligned} |\sum_{T\in\Im_h}\delta_T(cv, b\cdot\nabla v)_T| &\leq \sum_{T\in\Im_h}\delta_T c_T\left\|v\right\|_{0,T}\left\|b\cdot\nabla v\right\|_{0,T} \\ &\leq \sum_{T\in\Im_h}(\frac{1}{2}c_T^2\delta_T\left\|v\right\|_{0,T}^2 + \frac{1}{2}\delta_T\left\|b\cdot\nabla v\right\|_{0,T}) \\ &\leq \frac{1}{2}(d_0\left\|v\right\|_0^2 + \sum_{T\in\Im_h}\delta_T\left\|b\cdot\nabla v\right\|_{0,T}) \\ &< \frac{1}{2}\left|\left|\left|v\right|\right|\right|_{sd}, \end{aligned}$$

inequality (2.29) can be derived directly from (2.30).

□

Remark 2.2.2 *For* $P_{e_T} \gg 1$, δ_T *is usually set equal to* $\delta_0 h$ *for some constant* δ_0. *A good choices of* $\delta_0 = \frac{1}{2\|b\|}(1 - \frac{1}{P_e})$ *hsa been shown in [44]. Here, we simply set* $\delta_0 \approx \frac{1}{2\|b\|}$.

By simple calculation, (2.19) can be written as

$$B_{sd}(u_h, v_h) = \hat{B}_{sd}(u_h, v_h) + \check{B}_{sd}(u_h, v_h), \tag{2.31}$$

where $\hat{B}$ is the symmetric part of the operator and defined as

$$\begin{aligned}\hat{B}_{sd}(u_h, v_h) &= \epsilon(\nabla u_h, \nabla v_h) + ((c - \frac{1}{2}div(b))u_h, v_h) + \sum_{T\in\Im_h} \delta_T(b\cdot\nabla u_h, b\cdot\nabla v_h) \\ &+ \frac{1}{2}\sum_{T\in\Im_h} \delta_T(cb, \nabla(u_h v_h)),\end{aligned}$$

and $\check{B}_{sd}$ is the skew-symmetric part,

$$\check{B}_{sd}(u_h, v_h) = \frac{1}{2}[(b\cdot\nabla u_h, v_h) - (u_h, b\cdot\nabla v_h)] - \frac{1}{2}\sum_{T\in\Im_h} \delta_T[(cb\cdot\nabla u_h, v_h) - (u_h, cb\cdot\nabla v_h)]. \tag{2.32}$$

Since

$$\begin{aligned}\frac{1}{2}\sum_{T\in\Im_h} \delta_T(cb, \nabla(u_h u_h)) &= \frac{1}{2}|\sum_{T\in\Im_h} \delta_T \int_T 2(cu)(b\nabla u_h)| \\ &\leq \sum_{T\in\Im_h} \delta_T \int_T |b\cdot\nabla u_h||cu_h| \\ &\leq \sum_{T\in\Im_h} \delta_T[\frac{1}{2\sqrt{2}}\int_T |b\cdot\nabla u_h|^2 + \frac{1}{\sqrt{2}}\int_T |cu_h|^2], \\ &\quad \text{by arithmetic-geometric mean inequality}, \\ &\leq \frac{1}{2\sqrt{2}}[d_0\int_\Omega |u_h|^2 + \sum_{T\in\Im_h} \delta_T \|b\cdot\nabla u_h\|_T^2], \\ &\quad \text{by assumption of theorem 2.2.1},\end{aligned}$$

$\hat{B}_{sd}(u_h, u_h) \geq (1 - \frac{1}{2\sqrt{2}}) |||u_h|||_{sd}$. So, $\hat{B}_{sd}$ is positive definite. It is natural to define an energy norm $|||u|||_h = (u, u)^{1/2}_{\hat{B}_{sd}}$. Clearly,

$$(1 - \frac{1}{2\sqrt{2}}) |||u_h|||_{sd}^2 \leq |||u_h|||_h^2 \leq (1 + \frac{1}{2\sqrt{2}}) |||u_h|||_{SD}^2, \tag{2.33}$$

so, $|||\cdot|||_h$ is equivalent to $|||\cdot|||_{SD}$.

Lemma 2.2.3 *For any $u \in H^1$, we have*

$$C_{s1} \max\{\sqrt{\epsilon}, \sqrt{d_0}\} \|u\|_0 \leq |||u|||_{SD} \leq C_{s2} h^{-1/2} \|u\|_0 , \tag{2.34}$$

where C_{s1} and C_{s2} are constants.

Proof: By definition of $|||u|||_{sd}$, the upper bound can be derived from the inverse inequality and the lower bound is obvious from the Poincaré inequality.

□

Lemma 2.2.4 *For $u \in H^1(\Omega)$ and $v \in H_0^1(\Omega)$, there exist constants C_{b1} and C_{b2} such that*

$$|\check{B}_{sd}(u,v)| \leq C_{b1}(h)^{-1/2} |||u|||_{sd} \|v\|_0 . \tag{2.35}$$

$$|\check{B}_{sd}(u,v)| \leq C_{b2}(h\epsilon)^{-1/2} |||u|||_{sd} |||v|||_{sd} . \tag{2.36}$$

Proof:

$$\begin{aligned}
\tfrac{1}{2} \quad & [(b\cdot\nabla u, v) - (u, b\cdot\nabla v)] = \frac{1}{2}[2(b\cdot\nabla u, v) + (div(b)u, v)] \\
\leq \quad & (\sum_{T\in\Im_h} \frac{1}{\sqrt{\delta_T}} \sqrt{\delta_T} \|b\cdot\nabla u\|_0 \|v\|_0) + \frac{c-d_0}{\sqrt{d_0}} \sqrt{d_0} \|u\|_0 \|v\|_0 \\
\leq \quad & \max_{T\in\Im_h} (\frac{1}{\sqrt{\delta_T}}, \frac{c-d_0}{\sqrt{d_0}}) |||u|||_{sd} \|v\|_0 \\
\leq \quad & (\max_{\Omega} c) \frac{\sqrt{2}}{\sqrt{\delta_T}} |||u|||_{sd} \|v\|_0 , \text{ by } \delta_T \leq \tfrac{d_0}{2c_T^2}, \\
\leq \quad & \tilde{c} h^{-1/2} |||u|||_{sd} \|v\|_0 \quad \text{for some constant } \tilde{c}.
\end{aligned}$$

Also,

$$\begin{aligned}
& \sum_{T\in\Im_h} \delta_T[(cb\cdot\nabla u, v) - (u, cb\cdot\nabla v)] \\
\leq\ & \sum_{T\in\Im_h} \delta_T c_T(\|b\cdot\nabla u\|_0 \|v\|_0 + \|b\cdot\nabla v\|_0 \|u\|_0), \\
\leq\ & \sum_{T\in\Im_h} \frac{c_T\sqrt{\delta_T}}{\sqrt{d_0}}(\sqrt{\delta_T}\|b\cdot\nabla u\|_0 \sqrt{d_0}\|v\|_0 + \sqrt{\delta_T}\|b\cdot\nabla v\|_0 \sqrt{d_0}\|u\|_0), \text{ by } \delta_T \leq \tfrac{d_0}{2c_T^2} \\
\leq\ & \frac{1}{\sqrt{2}} |||u|||_{sd} |||v|||_{sd}, \\
\leq\ & \tilde{c}h^{-1/2} |||u|||_{sd} \|v\|_0, \text{ for some constant } \tilde{c}, \text{ by Lemma 2.2.3.}
\end{aligned}$$

After substituting the above estimations into (2.32). It follows that (2.35) holds. The inequality (2.36) then follows from Lemma 2.2.3.

□

Now, we can prove the continuity inequality.

Theorem 2.2.5 (Continuity) *For all $u, v \in V_h$, there exists some constant C such that*

$$|B_{sd}(u,v)| \leq C(h\epsilon)^{-1/2} |||u|||_{sd} |||v|||_{sd}. \tag{2.37}$$

Proof: Since $\hat{B}_{sd}$ is positive definite, by (2.33), we have

$$|\hat{B}_{sd}(u,v)| \leq |||u|||_h |||v|||_h \leq \tilde{c} |||u|||_{sd} |||v|||_{sd}, \text{ for some constant } \tilde{c}$$

Combine with (2.36), (2.31) implies

$$|B_{sd}(u,v)| \leq |\hat{B}_{sd}(u,v)| + |\check{B}_{sd}(u,v)| \leq C(h\epsilon)^{-1/2} |||u|||_{sd} |||v|||_{sd},$$

for some constant $C > 0$.

□

Again, by the Lax-Milgram Lemma, the SD finite element solution exists. An a priori error estimation can be easily obtained from Lemma 2.2.4.

Theorem 2.2.6 (A priori error estimate) *Suppose $u \in H^2(\Omega)$ is the weak solution and u_h is the discrete solution obtained from SD discretization on linear elements. Then the discretization error satisfies*

$$|||u - u_h|||_{sd} \preceq h^{k-\frac{1}{2}} |u|_k \tag{2.38}$$

for $k = 1$ or 2.

Proof:

$$\begin{aligned}
\frac{1}{2} |||u - u_h|||_{sd}^2 &\leq |B_{sd}(u - u_h, u - u_h)| \\
&= |B_{sd}(u - u_h, u - v)| \quad \forall v \in V_h \\
&= |\hat{B}_{sd}(u - u_h, u - v) + \check{B}_{sd}(u - u_h, u - v)| \\
&\preceq |||u - u_h|||_h \, |||u - v|||_h \, | + h^{-1/2} \, |||u - u_h|||_{sd} \, \|u - v\|_0 \\
&\preceq h_k^{-1/2} \, |||u - u_h|||_{sd} \inf_{v \in V_h} \|u - v\|_0 .
\end{aligned}$$

By Lemma 2.1.2, we have

$$|||u - u_h|||_{sd} \preceq h^{k-\frac{1}{2}} |u|_k, \text{ for k=1 or 2.}$$

□

From the a priori error estimation, the finite element solution obtained using SDFEM method approximates the weak solution with order only $O(h^{3/2})$ (compared to $O(h^2)$ for the Galerkin method (see Theorem 2.1.3)). On the other hand, there is no large constant of magnitude $\frac{1}{\epsilon}$ hidden inside the error bound for SDFEM. Consequently, this estimate is much more reliable than the a priori estimate from the Galerkin method.

Unfortunately, the regularity of u remains a difficulty for global convergence as discussed in the end of Section 2.1. Nevertheless, Johnson has shown $O(h^2)$ convergence on a region excluding the layers [58]. Niijima [69] proved $O(h^{11/8}log(h))$ pointwise convergence and Zhou sharpened the bound to $O(h^\alpha), \frac{3}{2} \leq \alpha \leq 2$ [107]. For simple flows with smooth domain and data, the weak solution is smooth in the interior regions, [85] pages 176-185. Moreover, the SDFEM method is capable of removing the oscillatory modes with carefully chosen stabilization parameter δ_T, [39] section 3.5. Therefore, we expect SDFEM solution to approximate the weak solution well in the region away from layers. In the next section, our numerical results support this observation.

2.3 Numerical Tests

In this section, we present two simple examples to compare the solution qualities from the SDFEM method and the GK method. Also, the convergence behavior of the SDFEM method for refined mesh is investigated. Our numerical results clearly show that the error in regions away from layers is much smaller than the global error. Moreover, the local convergence rate in regions away from layers is also faster than the global convergence rate under the SD-norm. However, the global convergence rate in our numerical tests is only $O(h^{1/2})$ instead of $O(h^{3/2})$ which is the best approximation order one can expect from the a priori error estimate. This should not be a surprise. If one combines the regularity estimate (2.8) in Lemma 2.0.4, and the a priori error estimate (2.38) in Theorem 2.2, one can bound the error $u - u_h$ in terms of the given data f as shown in the following:

$$|||u - u_h|||_{sd} \preceq (\frac{h}{\epsilon})^{k-\frac{1}{2}} \|f\|_0 , \text{ where } k = 1, 2.$$

Clearly, when $h \gg \epsilon$, we obtain a better error bound with order $O(h^{1/2})$ by letting $k = 1$.

In our test problems, we estimate the error $u - u_h$ on a very fine adaptively refined mesh, $\Im_f$, which is generated by 3 refinement steps from an initial 64x64 mesh with threshold value 0.25 in the maximum marking strategy defined in Chapter 3. The discrete solution u_h is injected to $\Im_f$ by standard bilinear interpolation. For problems whose exact solution is known, the error $u - u_h$ on $\Im_f$ is available. Otherwise, the SDFEM solution u_f on $\Im_f$ is then treated as exact solution u and the error $u_f - u_h$ is treated as the true error $u - u_h$.

Problem 1: Downstream boundary layers

Consider

$$u(x, y) = \frac{e^{\beta_1 x/\varepsilon} - 1}{e^{\beta_1/\varepsilon} - 1} + \frac{e^{\beta_2 y/\varepsilon} - 1}{e^{\beta_2/\varepsilon} - 1} \tag{2.39}$$

on the domain $\Omega = [0, 1] \times [0, 1]$, where $(\beta_1, \beta_2) = (\cos\theta, \sin\theta)$ for $0^o < \theta < 90^o$. Direct calculation shows u satisfies

$$-\varepsilon \cdot \Delta u + (\beta_1, \beta_2) \cdot \nabla u = 0.$$

with Dirichlet boundary condition $g = u$ on $\partial\Omega$. Clearly, exponential layers near boundary $x = 1$ and $y = 1$ are expected. we examine the convergence rate in regions that exclude layers. First, the region Ω_0

$$\Omega_0 = \{(x, y) \in \Omega : x < 0.9 \wedge y < 0.9\}.$$

is obtained empirically. Next, since the width of the exponential layer of the solution u is $O(\epsilon)$ and the local pointwise error, $|u(x_0) - u_h(x_0)|$ of any interior point x_0,

is usually estimated with respect to $\|u\|_{2,B_r(x_0)}$ where $B_r(x_0)$ is a ball with radius $r \sim h|\log(h)|$ [58], it is reasonable to assume that the exponential layer of u_h has width about $h|\log(h)| + \epsilon$. To define a region that does not include the layers, we exclude from Ω a region of width $2(h|\log(h)| + \epsilon)$ next to the outflow boundaries. Let $\Omega_{0,h}$ denote this region,

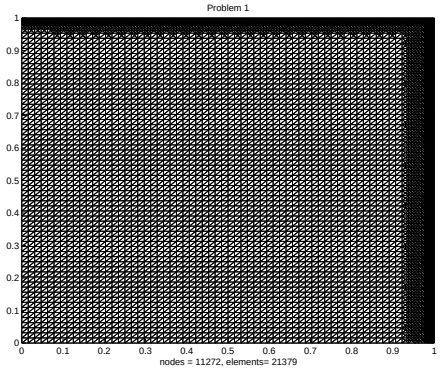
$$\Omega_{0,h} = \{(x, y) \in \Omega : x < 1 - 2(h \log h + \varepsilon) \wedge y < 1 - 2(h \log h + \varepsilon)\}.$$

The local convergence rate is then examined on both Ω_0 and $\Omega_{0,h}$.

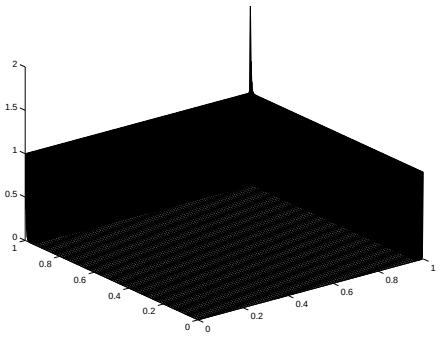
The following are numerical results for the case $\epsilon = 1e - 03$ and $\theta = 15^o$. Clearly, Figure 2.1 shows GK solution suffer serious oscillation on whole domain but SD solution maintains good solution quality with small oscillation in the layer regions. The third column of Table 2.1 shows SD solution has much smaller error in the regions away from layer comparing to the global error in the first column. On a fixed domain Ω_0 excluding layer regions, the convergence rate is better than h^2 which may due to the fact that the solution u belong to $H^\alpha(\Omega_0)$ for $\alpha > 2$.

mesh	$\|\|\|u-u_h\|\|\|_{sd,\Omega}$	$\|\|\|u-u_h\|\|\|_{sd,\Omega_0}$	$\|\|\|u-u_h\|\|\|_{sd,\Omega_{0,h}}$
8x8	3.82	3.33e-01	3.17e-03
16x16	2.69	8.82e-02	1.82e-03
32x32	1.87	9.91e-03	1.13e-04
64x64	1.26	3.41e-06	1.15e-07

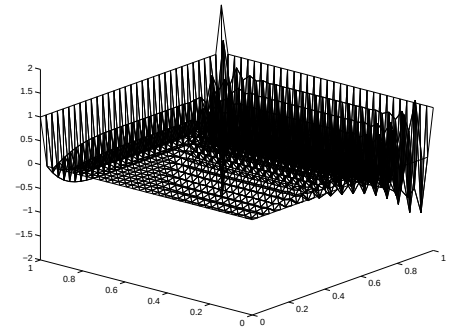
Table 2.1: Error estimation of SD solution



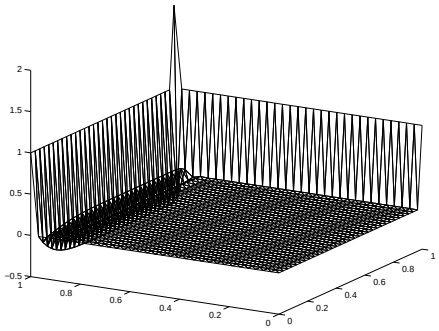
(a) Mesh $\Im_f$



(b) Solution u on $\Im_f$



(c) GK Solution on uniform 32x32 grid



(d) SD Solution on uniform 32x32 grid

Figure 2.1:

Problem 2: Characteristic and downstream layer

$$-\varepsilon \cdot \Delta u + \frac{\partial u}{\partial y} = 0$$
$$u|_{\partial\Omega} = \begin{cases} 1 & \text{if } y = 0 \text{ and } x > 0 \text{ or } x = 1, \\ 0 & \text{otherwise,} \end{cases}$$

where $\Omega = [-1, 1] \times [-1, 1]$.

This problem exhibits an internal layer in the interior region and a exponential layer on the boundary $y = 1$. The internal layer arises due to the discontinuity of the given boundary data and has width $O(\sqrt{\epsilon})$. We set the width of layers to $2(h|\log(h)| + \sqrt{\epsilon})$ and let $\Omega_{0,h}$ denote the region excluding layers,

$$\Omega_{0,h} = \{(x, y) \in \Omega : x > 2(h \log h + \varepsilon^{1/2}) \wedge y < 1 - 2(h \log h + \varepsilon^{1/2}),$$
$$\text{or } x < -2(h \log h + \varepsilon^{1/2})\}.$$

Also, another domain Ω_0 that excludes layers are empirically chosen to be

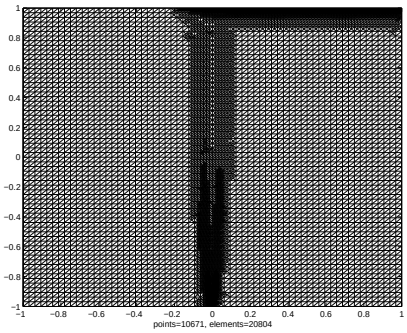
$$\Omega_0 = \{(x, y) \in \Omega : x > 0.2 \wedge y < 0.8 \cup \mathrm{x} < -0.2\}.$$

Again, the local convergence rate is examined on both Ω_0 and $\Omega_{0,h}$ and the exact error is computed on a mesh .

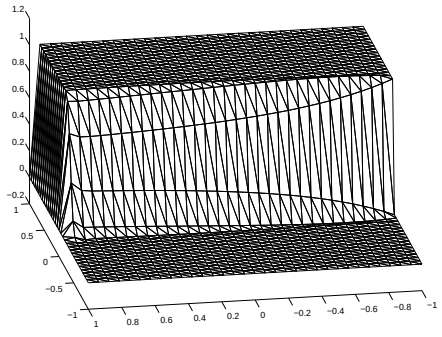
For the case $\epsilon = 10^{-3}$, figure 2.2 shows that the GK solution suffer serious oscillation on the whole domain. On the other hand, the SDFEM solution has good solution quality. The third column of table 2.2 shows the error in the region away from layers is much smaller than the global error in the first column. Also, on the fixed domain Ω_0, the convergence rate is better than h^2 as we seen in problem 1.

mesh	$\|\|\|u-u_h\|\|\|_{sd,\Omega}$	$\|\|\|u-u_h\|\|\|_{sd,\Omega_0}$	$\|\|\|u-u_h\|\|\|_{sd,\Omega_{0,h}}$
8x8	5.53	3.15e-01	7.28e-03
16x16	3.88	6.69e-02	1.40e-03
32x32	2.70	4.03e-03	3.12e-04
64x64	1.84	2.13e-04	6.78e-05

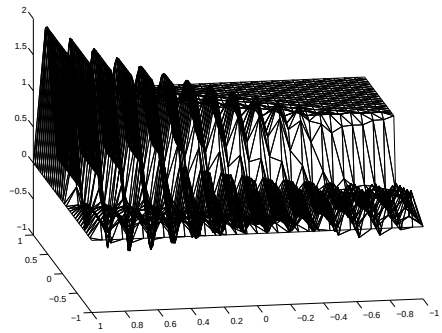
Table 2.2: Error estimate of SD solution



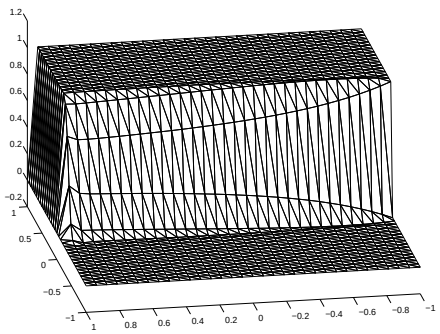
(a) Mesh $\Im_f$



(b) Solution on $\Im_f$



(c) GK Solution on 32x32 grid



(d) SD Solution on 32x32 grid

Figure 2.2:

Chapter 3

A Posteriori Error Estimations and Mesh Improvement

In Chapter 2, we have shown that the theoretical convergence rate is $h^{3/2}$ for the SD-FEM solutions, but only $h^{1/2}$ convergence rate is observed in our numerical results. This result can be explained if the error is bounded in terms of data. Even though the a priori error bound is capable of revealing the asymptotic behavior of the error, it is not computable and can't be used to estimate the exact error. On the other hand, our numerical results also show that errors in the regions excluding layers are much smaller than the global errors. This phenomenon suggests that one can increase the accuracy of the approximate solution without overloading the computational cost by placing more grid points in the regions where errors are large. Therefore, it is natural to acquire some computable error indicators to pinpoint where the error is large and, at the same time, properly bound the exact error on the whole domain. In this chapter, we consider such a posteriori error indicator.

To validate the reliability and efficiency of the error indicators, the global effectivity

index, defined as

$$E_\Omega = \frac{(\sum_{T\in\Im_h} \eta_{h,T}^2)^{1/2}}{\|u-u_h\|_\Omega},$$

and the local effectivity index, defined as

$$E_T = \max_{t\in\Im_h} \frac{\eta_{h,T}}{\|u-u_h\|_T},$$

are computed, where $\|\cdot\|$ represents the norms used to measure the exact error in these indicators. Obviously, if $E_\Omega \approx 1$, the error indicator is reliable in measuring the global error. Otherwise, if $E_\Omega \gg 1$, the error indicator under-estimates the error and if $E_\Omega \ll 1$ the error indicator over-estimates the error. Moreover, the local index E_T can be used to determine how sharp the local a posteriori lower bound is.

With an error indicator in hand, adaptive mesh refinement can be accomplished by the decision of selecting elements, the so-called marking strategy, and the refinement strategies such as the regular refinement or the longest-side bisection algorithm [83] [84]. A heuristic marking strategy is the maximum marking strategy [72] where an element T^* will be marked for refinement if

$$\eta_{T^*} > \theta \max_{T\in\Im_h} \eta_T, \tag{3.1}$$

with a prescribed threshold $0 \leq \theta \leq 1$. Some other marking strategies can also be seen in [72].

In this chapter, we study two types of a posteriori error estimators where the approximate solution is obtained from SDFEM. In Section 3.1, we introduce a residual-type of error indicator proposed by Verfürth in [97]. Hereafter, we call it the VR-indicator. In Section 3.2, instead of studying the Neumann-type of error indicator by Verfürth

in which the size of each local problem is at least 12x12 in triangular elements, we introduce a Neumann-type of error indicator proposed by Kay and Silvester in [59] where the size of each local problem is only 4x4 in triangular elements. Hereafter, we call it the KS-indicator. In Section 3.3, we present numerical results indicating that the a posteriori bounds in our studies are sharp.

3.1 Residual-type a Posteriori Error Estimation

First, let us introduce the following abbreviations:

$$\begin{aligned}
R_T(u_h) &= f + \epsilon\Delta u_h - b\cdot\nabla u_h - cu_h \\
R_{h,T}(u_h) &= f_h + \epsilon\Delta u_h - b\cdot\nabla u_h - cu_h \\
R_{h,E_\Omega}(u_h) &= -[\epsilon\nabla u_h\cdot n_E]_E \quad \text{if } E\in\Omega \\
R_{E_N}(u_h) &= \bar{g} - \epsilon\nabla u_h\cdot n_E \quad \text{if } E\in\Gamma_N \\
R_{h,E_N}(u_h) &= \bar{g}_h - \epsilon\nabla u_h\cdot n_E \quad \text{if } E\in\Gamma_N \\
R_{E_D}(u_h) &= 0 \quad \text{if } E\in\Gamma_D
\end{aligned}$$

where n_E is the unit vector normal to the edge E, $\bar{g}$ is the given Neumann condition on boundary Γ_N and $[\cdot]_E$ denotes the jump of a function across the edge E. The VR-indicator consists of the element residual component, $R_{h,T}$, and the element edge-flux components, R_{h,E_Ω} and R_{h,E_N}, and is written as

$$\eta_{h,T} = (\rho_T^2\,\|R_{h,T}(u_h)\|_{0,T}^2 + \rho_E \sum_{E\in\partial T\cap\Omega} \|R_{h,E_\Omega}(u_h)\|_{0,E}^2 + \rho_E \sum_{E\in\partial T\cap\Gamma_N} \|R_{h,E_N}(u_h)\|_{0,E}^2)^{1/2},$$

with $\rho_T = \min\{\frac{h_T}{\sqrt{\epsilon}},1\}$ and $\rho_E = \epsilon^{-1/2}\min\{\frac{h_E}{\sqrt{\epsilon}},1\}$. Let $e_h = u - u_h$ and $|||\cdot||| = (\epsilon\,\|\nabla\cdot\|^2 + d_0\,\|\cdot\|^2)^{1/2}$ denote the usual energy norm where d_0 is the constant described in (1.2). Assume $d_0 \gg \epsilon$. Verfürth's a posteriori error estimation reads as follows:

(**Global Upper Bound**):

$$|||e_h|||_\Omega \preceq \{\sum_{T\in\Im_h} \eta_{h,T}^2\} + \{\sum_{T\in\Im_h} \rho_T^2 \|f - f_h\|_{0,T}^2 + \sum_{E\in\Gamma_N} \epsilon^{-1/2}\rho_E \|\bar{g} - \bar{g}_h\|_{0,E}^2\} \quad (3.2)$$

and

(**Local Lower Bound**):

$$\begin{aligned}\eta_{h,T} \preceq &\{1 + \|c\|_{\infty,\omega_T} + \epsilon^{-1/2}\|b\|_{\infty,\omega_T}\rho_T\} |||e_h|||_{\omega_T} \\ &+ \rho_T \|f - f_h\|_{0,\omega_T} + \{\sum_{E\in\partial T\cap\Gamma_N} \epsilon^{-1/2}\rho_E \|\bar{g} - \bar{g}_h\|_{0,E}^2\}^{1/2}.\end{aligned} \quad (3.3)$$

In the following, we outline the basic proof only for problems with only Dirichlet boundary conditions. The same scheme can be extended to problems with Neumann conditions and we refer to [97] for details.

First, by integration by parts, for all $w \in H_0^1(\Omega)$, we have

$$B(e_h, w) = \sum_{T\in\Im_h} \{(R_{h,T}(u_h), w)_T + (f - f_h, w)_T\} + \sum_{E\in\Omega} (R_{h,E_\Omega}(u_h), w)_E. \quad (3.4)$$

By Cauchy-Schwarz inequality, it is clear that

$$|B(e_h, w)| \leq \sum_{T\in\Im_h} (\|R_{h,T}(u_h)\|_{0,T} + \|f - f_h\|_{0,T}) \|w\|_{0,T} + \sum_{E\in\Omega} \|R_{h,E_\Omega}(u_h)\|_{0,E} \|w\|_{0,E}$$

Let $w = e_h - I(e_h)$ where the operator I is the quasi-interpolation operator of Clément. By the interpolation estimates in Lemma 3.2 of [97],

1. $\|w - Iw\|_{0,T} \preceq \rho_T |||w|||_{\tilde{\omega}_T}$,

2. $\|w - Iw\|_{0,E} \preceq \sqrt{\rho_E} |||w|||_{\tilde{\omega}_T}$,

for all $w \in H^1(\tilde{\omega}_T)$, the above inequality implies

$$
\begin{aligned}
|B(e_h, e_h - I(e_h))| &\leq \sum_{T \in \Im_h} \rho_T (\|R_{h,T}(u_h)\|_{0,T} + \|f - f_h\|_{0,T}) \, |||e_h|||_{\tilde{\omega}_T} \\
&\quad + \sum_{E \in \omega} \rho_E^{1/2} \, \|R_{h,E_\Omega}(u_h)\|_{0,E} \, |||e_h|||_{\tilde{\omega}_T} \qquad (3.5) \\
&\preceq \{ (\sum_{T \in \Im_h} \eta_{h,T}^2)^{1/2} + (\sum_{T \in \Im_h} \rho_T^2 \, \|f - f_h\|_{0,T}^2)^{1/2} \} \, |||e_h|||_\Omega \, .
\end{aligned}
$$

Second, the bilinear form $B(e_h, w)$ can also be rewrite as

$$
B(e_h, w) = B_{sd}(e_h, w) - \sum_{T \in \Im_h} \delta_T (R_T(u_h), b \cdot \nabla w)_T \quad \forall w \in V_h
$$

Let $w = I(e_h)$. The orthogonality of B_{sd} implies $B_{sd}(e_h, w) = 0$. Therefore,

$$
\begin{aligned}
|B(e_h, I(e_h))| &\leq \sum_{T \in \Im_h} \delta_T \, \|R_T(u_h)\|_{0,T} \, \|b \cdot \nabla I(e_h)\|_{0,T} \\
&\preceq \sum_{T \in \Im_h} \delta_T (\|R_{h,T}(u_h)\|_{0,T} + \|f - f_h\|_{0,T}) \, \|b\|_{\infty,T} \, h^{-1} \, \|I(e_h)\|_{0,T},
\end{aligned}
$$

by a simple scaling argument. Again, from the interpolation estimates in Lemma 3.2 of [97],

$$
|||Iw|||_T \preceq |||w|||_{\tilde{\omega}_T} \, ,
$$

we have

$$
|B(e_h, I(e_h))| \preceq \{ (\sum_{T \in \Im_h} \eta_{h,T}^2)^{1/2} + (\rho_T^2 \sum_{T \in \Im_h} \|f - f_h\|_{0,T}^2)^{1/2} \} \, |||e_h|||_\Omega \, . \qquad (3.6)
$$

Now, from the coercivity estimate (2.29), (3.5) and (3.6), clearly the upper bound (3.2) holds.

For the local lower bound, one would require judicious cut-off functions ψ_T on each element T and cut-off functions $\psi_{E,\vartheta}$ on each interior edge E, where ϑ is a scaling

parameter between 0 and 1. The cut-off functions are essentially scaled bubble functions and are rigorously defined in [97]. First, by choosing $w = \psi_T R_{h,T}(u_h)$, (3.4) implies

$$(R_{h,T}(u_h), \psi_T R_{h,T}(u_h))_T = B(e_h, \psi_T R_{h,T}(u_h)) + (f - f_h, \psi_T R_{h,T}(u_h))_T. \quad (3.7)$$

By the following inequalities, in Lemma 3.3 of [97],

$$\begin{aligned} \|v\|_{0,T} &\preceq (v, \psi_T v)_T, \\ \|v\psi_T\|_{0,T} &\leq \|v\|_{0,T}, \\ |||v\psi_T||| &\preceq \rho_T^{-1} \|v\|_{0,T}, \end{aligned} \quad (3.8)$$

for all $v \in P_k$, one can show that

$$\begin{aligned} \|R_{h,T}(u_h)\|_{0,T} \preceq |||e_h|||_T \{(1 + \|c\|_{\infty,T})\rho_T^{-1} + \epsilon^{-1/2} \|b\|_{\infty,T}\} \\ + \|f - f_h\|_{0,T}. \end{aligned} \quad (3.9)$$

Next, by choosing test function $w = \psi_{E,\vartheta} P_E R_{h,E_\Omega}(u_h)$ with the scaling parameter $\vartheta = \min\{\frac{h_E}{\sqrt{\epsilon}}, 1\}$, where P_E is a continuation operator which extends function value on an edge E to its neighboring elements, (3.4) implies

$$\begin{aligned} (R_{h,E_\Omega}(u_h), \psi_{E,\vartheta} P_E R_{h,E_\Omega}(u_h)) &= B(e_h, \psi_{E,\vartheta} P_E R_{h,E_\Omega}(u_h)) \\ &- \sum_{T \subset \omega_E} (R_{h,T}(u_h), \psi_{E,\vartheta} P_E R_{h,E_\Omega}(u_h))_T \\ &- \sum_{T \subset \omega_E} (f - f_h, \psi_{E,\vartheta} P_E R_{h,E_\Omega}(u_h))_T \end{aligned} \quad (3.10)$$

Again, by the following inequalities, in Lemma 3.3 of [97],

$$\begin{aligned} \|v\|_{0,E} &\preceq (v, \psi_{E,\vartheta} P_E v)_E, \\ \|\psi_{E,\vartheta} P_E v\|_{0,\omega_E} &\preceq \epsilon^{1/2} {\rho_E}^{1/2} \|v\|_{0,E}, \\ |||\psi_{E,\vartheta} P_E v|||_{0,\omega_E} &\preceq \rho_E^{-1/2} \|v\|_{0,E}, \end{aligned} \quad (3.11)$$

for all $v \in P_k|_E$, one can show

$$\begin{aligned} \|R_{h,E_\Omega}(u_h)\|_{0,E} \preceq |||e_h|||_{\omega_E} \{1 + \|c\|_{\infty,\omega_E} + \epsilon^{-1/2}\rho_T \|b\|_{\infty,\omega_E}\}\rho_E^{-1/2} \\ + \rho_E^{-1/2} \min\{\frac{h_E}{\sqrt{\epsilon}}, 1\} \|f - f_h\|_{0,\omega_E} \end{aligned} \tag{3.12}$$

By combining (3.9), (3.12) and the definition of $\eta_{h,T}$, the local lower bound (3.3) holds.

Remark 3.1.1 *The parameters $\rho_T, \rho_E = \min\{\frac{h}{\sqrt{\epsilon}}, 1\}$ appearing in the VR-indicator is a direct result from scaling factors between the energy norm and the other norms, such as L^2 norm and H^1 norm, while estimating the error in terms of the residual, $R_{h,T}$, and the edge-flux, $R_{h,E_\Omega}, R_{h,E_N}$. For convection-diffusion equations with coefficient c=0 in (1.1), the energy norm is simply $|||\cdot||| = \epsilon^{1/2} \|\nabla\cdot\|$ without the L^2-norm component. Obviously, the scaling factors between the energy norm and the other norms are different and lead to different ρ_T and ρ_E in the error indicator. By following Verfürth's arguments and carefully adjusting the scaling factors in the auxiliary inequalities of [97], one can show that the same upper and lower bound holds with $\rho_T, \rho_E = \frac{h}{\sqrt{\epsilon}}$.*

3.2 Neumann-type a Posteriori Error Estimation

The basic idea of the KS-estimator is based on solving a local (element) Poisson problem over a higher order approximation space with given data from interior residuals and flux jumps along element edges. First, we introduce some abbreviations. The

interior residual of element T and the flux jump of edge E are denoted as follows:

$$
\begin{aligned}
R_T &= (f - b \cdot \nabla u_h)|_T \\
R_T^0 &= \mathcal{P}_T^0(R_T), \text{ where } \mathcal{P}_T^0 \text{ is the } L^2(T)\text{-projection onto } P^0(T) \\
R_E &= \begin{cases} [\frac{\partial u_h}{\partial n_E}]_E & \text{if } E \in \Omega \\ -2(\frac{\partial u_h}{\partial n_E}) & \text{if } E \in \Gamma_N \\ 0 & \text{if } E \in \Gamma_D \end{cases}
\end{aligned}
$$

The approximation space is denoted as $Q_T = Q_T \bigoplus B_T$, where

$Q_T = span\{\psi_E \circ \Phi^{-1} |\ \psi_E = 4\chi_i\chi_j$, i,j are the endpoints of E and $E \in \partial T \bigcap (\Omega \bigcup \Gamma_N)\}$

is the space spanned by the quadratic edge bubble functions and

$$B_T = span\{\psi_T \circ \Phi^{-1} |\ \psi_T = 27 \prod_{i=1}^{3} \chi_i\}$$

is the space spanned by cubic interior bubble function. For an element T, the estimator is given by

$$\eta_{h,T} = \|\nabla e_T\|_{0,T} ,$$

where $e_T \in Q_T$ satisfies

$$\epsilon(\nabla e_T, \nabla v)_T = (R_T^0, v)_T - \frac{1}{2}\epsilon \sum_{E \in \partial T} (R_E, v)_E \tag{3.13}$$

Let $e_h = u - u_h$. The Kay and Silvester's a posteriori error estimation can be read as following:

(**Global Upper Bound**):

$$\|\nabla(e_h)\|_{0,\Omega} \preceq (\sum_{T \in \Im_h} \eta_{h,T}^2 + \sum_{T \in \Im_h} (\frac{h}{\epsilon})^2 \left\|R_T - R_T^0\right\|_{0,T})^{1/2} \tag{3.14}$$

(**Local Lower Bound**):

$$\eta_{h,T} \preceq \|e_h\|_{0,\omega_T} + \sum_{T\subset\omega_T} \frac{h_T}{\epsilon} \|b\cdot\nabla e_h\|_{0,T} + \sum_{T\subset\omega_T} \frac{h_T}{\epsilon} \left\|R_T - R_T^0\right\|_{0,T} \tag{3.15}$$

To derive the upper bound, first, the bilinear form $B(e_h, e_h)$ is written as

$$\begin{aligned} B(e_h,e_h) &= B(e_h, e_h - Ie_h) - B(e_h, Ie_h) \\ &= B(u, e_h - Ie_h) - B(u_h, e_h - Ie_h) - \sum_{T\in\Im_h} \delta_T(f - b\cdot\nabla u_h - cu_h, b\cdot\nabla Ie_h) \\ &= \sum_{T\in\Im_h} [(R_T, e_h - Ie_h)_T - \delta_T(R_T, b\cdot\nabla Ie_h)_T + \frac{1}{2}\epsilon \sum_{E\in(\Omega\cup\Gamma_N)} (R_E, e_h - Ie_h)_E]. \end{aligned}$$

From coercivity estimate (2.29), interpolation estimates (2.1.2) and the Cauchy-Schwarz inequality, it can be shown

$$\begin{aligned} \epsilon \|\nabla e_h\|_{0,\Omega}^2 &\preceq \sum_{T\in\Im_h} h_T(\left\|R_T^0\right\|_{0,T} + \left\|R_T - R_T^0\right\|_{0,T} + \frac{\epsilon}{2} \sum_{E\in(\Omega\cup\Gamma_N)} h_E^{1/2} \|R_E\|_{0,E}) \|\nabla e_h\|_{0,\tilde{\omega}_T} \\ &\preceq \|\nabla e_h\|_{0,\Omega} \{ \sum_{T\in\Im_h} [h_T^2 \left\|R_T^0\right\|_{0,T}^2 + h_T^2 \left\|R_T - R_T^0\right\|_{0,T}^2 \\ &+ (\frac{\epsilon}{2})^2 \sum_{E\in(\Omega\cup\Gamma_N)} h_E \|R_E\|_{0,E}^2]\}^{1/2}. \end{aligned} \tag{3.16}$$

Now, it remains to bound $\|R_T^0\|_{0,T}$ and $\|R_E\|_{0,E}$ in terms of $\eta_{h,T}$. By choosing a cut-off function $\psi_T \in B_T$, (3.13) and (3.8) imply

$$\left\|R_T^0\right\|_{0,T}^2 \preceq (R_T^0, \psi_T R_T^0)_T = \epsilon(\nabla e_T, \nabla\psi_T R_T^0)_T \leq \epsilon h_T^{-1} \|\nabla e_T\|_{0,T} \left\|R_T^0\right\|_{0,T} \tag{3.17}$$

Similarly, by choosing the cut-off function $\psi_E \in Q_T$, (3.13) and (3.11) imply

$$\begin{aligned} \epsilon \|R_E\|_{0,E}^2 \preceq \epsilon(R_E, \psi_E R_E)_E &= \sum_{T'\subset\omega_T} -\epsilon(\nabla E_{T'}, \nabla\psi_E R_E)_{T'} + (R_{T'}^0, \psi_E R_E)_{T'} \\ &\preceq \|R_E\|_{0,E} \sum_{T'\subset\omega_T} [\epsilon h_E^{-1/2} \|\nabla e_{T'}\|_{0,T'} + h_{T'}^{1/2} \left\|R_{T'}^0\right\|_{0,T'}] \end{aligned} \tag{3.18}$$

By plugging (3.17) and (3.18) into (3.16), the global upper bound (3.14) holds. To show the local lower bound, first, we set $v = e_T$ in (3.13). By a standard scaling argument, it is clear that

$$\begin{aligned}\epsilon \left\|\nabla e_T\right\|_{0,T}^2 &= (R_T^0, e_T)_T - \frac{\epsilon}{2} \sum_{E \in (\Omega \cap \Gamma_N)} (R_E, e_T)_E \\ &\preceq h_T \left\|R_T^0\right\|_{0,T} \left\|\nabla e_T\right\|_{0,T} + \frac{\epsilon}{2} \sum_{T \in (\Omega \cap \Gamma_N)} h_E^{1/2} \left\|R_E\right\|_{0,E} \left\|\nabla e_T\right\|_{0,T}.\end{aligned} \tag{3.19}$$

Now we only need to bound $\|R_T^0\|_{0,T}$ and $\|R_E\|_{0,T}$ in terms of ∇e_h. Again, from a proper chosen cut-off function ψ_T, we have

$$\begin{aligned}\left\|R_T^0\right\|_{0,T}^2 &\preceq (R_T^0, \psi_T R_T^0)_T = (R_T^0 - R_T, \psi_T R_T^0)_T + B(u - u_h, \psi_T R_T^0) \\ &\preceq \left\|R_T^0 - R_T\right\|_{0,T} \left\|\psi_T R_T^0\right\|_{0,T} + \epsilon \left\|\nabla e_h\right\|_{0,T} \left\|\nabla \psi R_T^0\right\|_{0,T} + \left\|b \cdot \nabla e_h\right\|_{0,T} \left\|\psi_T R_T^0\right\|_{0,T} \\ &\preceq \left\|R_T^0 - R_T\right\|_{0,T} \left\|R_T^0\right\|_{0,T} + \frac{\epsilon}{h_T} \left\|\nabla e_h\right\|_{0,T} \left\|R_T^0\right\|_{0,T} + \left\|\nabla e_h\right\|_{0,T} \left\|R_T^0\right\|_{0,T}\end{aligned} \tag{3.20}$$

Similarly, by using the cut-off function ψ_E, it can be shown

$$\begin{aligned}\epsilon \left\|R_E\right\|_{0,E}^2 &\preceq \epsilon (R_E, \psi_E R_E)_E \\ &= \sum_{T' \subset \omega_E} \epsilon (\nabla u_h, \nabla \psi_E R_E)_{T'}, \text{ by the definition of } R_E \text{ and the Green formula} \\ &= \sum_{T' \subset \omega_E} -\epsilon (\nabla e_h, \nabla \psi_E R_E)_{T'} + \epsilon (\nabla u, \nabla \psi_E R_E)_{T'} \\ &= \sum_{T' \subset \omega_E} [(R_T - R_T^0 + R_T^0 - b \cdot \nabla e_h - c e_h, \psi_E R_E)_{T'} - \epsilon (\nabla e_h, \nabla \psi_E R_E)_{T'}].\end{aligned}$$

Therefore, by (3.11) and (3.20), we have

$$\epsilon h_E^{1/2} \left\|R_E\right\|_{0,E} \preceq \sum_{T' \subset \omega_E} [h_{T'} \left\|R_{T'}^0 - R_{T'}\right\|_{0,T'} + h_{T'} \left\|b \cdot \nabla e_h + c e_h\right\|_{0,T'} + \epsilon \left\|\nabla e_h\right\|_{0,T'}] \tag{3.21}$$

By plugging (3.20) and (3.21) into (3.19), the local lower bound (3.15) holds.

3.3 Numerical Results

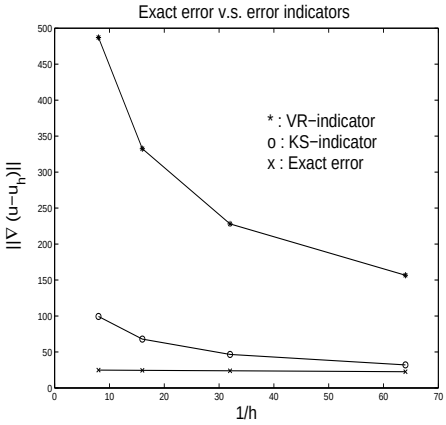
In this section, we compute both global and local effectivity indices of the VR-indicator and the KS-indicator for Problem 1 in Section 2.3. In order to see how the effectivity indices change in terms of the diffusion parameter ϵ and mesh size h, the problem is solved over uniform meshes with mesh size $h = \frac{1}{8}, \frac{1}{16}, \frac{1}{32}$ and $\frac{1}{64}$ for $\epsilon = \frac{1}{64}, \frac{1}{256}, \frac{1}{1024}$ and $\frac{1}{4096}$. Since the true solution has exponential layers along the boundary at x=1 and y=1, one requires a mesh which is fine enough in layer regions, to obtain a better approximation of the exact error. To generate such a mesh, first, the problem with $\epsilon = \frac{1}{1024}$ is solved on a 64x64 initial mesh. Three refinement steps are performed by using the maximum marking strategy on KS-indicator with threshold value $\theta = 0.75$. The mesh $\Im_f$, similar to the mesh shown in Figure 2.1 (a), consists of 11271 nodes and 21377 elements. The discrete true solution u is obtained directly by (2.39) on $\Im_f$. The SDFEM solution u_h is also prolonged by standard bilinear interpolation onto $\Im_f$. Then, an approximation to the exact error can be computed as

$$\|u - u_h\|_{0,\Omega} = (\sum_{T \in \Im_f} \|u - u_h\|_{0,T}^2)^{1/2},$$

where $\|u - u_h\|_{0,T}^2$ is calculated by 7-point Gaussian quadrature.

First, the VR-indicator, the KS-indicator and the exact error are plotted in the following figure for the case $\epsilon = \frac{1}{1024}$, where the exact error is measured in the H^1-seminorm and the VR-indicator is scaled by a factor of $\frac{1}{\sqrt{\epsilon}}$ to reflect the scaling factor between the H^1-seminorm and the energy norm. The table beside the figure contains the actual data for plotting the error and error indicators. It is clear that the KS-indicator provides a more reliable upper bound than the VR-indicator. In fact, similar results hold for $h \gg \epsilon$.

	$\|\|\|u-u_h\|\|\|$	$\|u-u_h\|_1$	$(\sum_{T\in\Im_h}\eta_{h,T}^2)^{1/2}$	
			VR	KS
1/8	0.773	24.75	15.22	99.40
1/16	0.765	24.48	10.39	67.85
1/32	0.745	23.85	7.126	46.54
1/64	0.703	22.49	4.895	31.97



Moreover, from Table 3.1, we can see that the local effectivity indices of VR-indicator and KS-indicator blow up in a rate of $O(P_e)$ as $h \gg \epsilon$. Furthermore, the numerical data in Table 3.2 also show that the global effectivity indices blow up in a rate of $O(\sqrt{P_e})$ as mentioned in [59]. The above results indicate that the local lower bounds of the a posteriori error estimation of Verfürth, Kay and Silvester, are sharp and support the well-known equivalence of residual type error indicator and local-problem type error indicator.

ϵ	8x8	16x16	32x32	64x64
$\frac{1}{64}$	12.43	8.620	8.610	7.741
$\frac{1}{256}$	45.26	22.69	12.46	8.627
$\frac{1}{1024}$	181.0	90.51	45.26	22.67
$\frac{1}{4096}$	724.1	362.0	181.0	90.54

(a) E_T of VR-indicator

ϵ	8x8	16x16	32x32	64x64
$\frac{1}{64}$	2.504	1.714	1.687	1.557
$\frac{1}{256}$	9.242	4.637	2.536	1.750
$\frac{1}{1024}$	36.95	18.48	9.239	4.629
$\frac{1}{4096}$	147.8	73.90	36.95	18.48

(b) E_T of KS-indicator

Table 3.1: Comparison of the local effectivity indices

ϵ	8x8	16x16	32x32	64x64
$\frac{1}{64}$	5.764	4.673	4.825	5.064
$\frac{1}{256}$	10.01	7.137	5.415	4.555
$\frac{1}{1024}$	19.68	13.58	9.562	6.966
$\frac{1}{4096}$	41.49	28.62	19.96	14.04

(a) E_Ω of VR-indicator

ϵ	8x8	16x16	32x32	64x64
$\frac{1}{64}$	1.156	0.951	0.979	1.022
$\frac{1}{256}$	2.044	1.457	1.105	0.929
$\frac{1}{1024}$	4.016	2.772	1.952	1.422
$\frac{1}{4096}$	8.470	5.842	4.075	2.867

(b) E_Ω of KS-indicator

Table 3.2: Comparison of the global effectivity indices

3.4 Moving Mesh

Although adaptive mesh refinement can greatly improve the accuracy of the numerical solution when a reliable a posterior error estimator is available, without proper threshold value in the marking strategies, under-refinement or over-refinement may occur in the refinement process. As a result, in order to obtain an accurate approximate solution, number of refinement steps may become too large if under-refinement occurs, or, the discrete linear system may become too large to solve if over-refinement occurs. Especially, for convection-dominant problems, i.e. the mesh Peclét number $\min_{T\in\Im_h} P_{e_T} \gg 1$, if the diffusion parameter ϵ is extremely small, it is not practical to resolve layers by simply increasing number of nodes with a regular mesh refinement process. With the above difficulties in mind, it is desirable to be able to increase the accuracy of the numerical solution in the layer regions with fix amount of nodes. A natural approach to achieve this goal is to cluster nodes in the layer regions using moving meshes.

Moving mesh methods such as moving mesh partial differential equations (MM-

PDES) by Huang and Russell [52], moving finite element (MFE) by Miller [66][67] and gradient weighted moving finite element (GWMFE) by Carlson and Miller [25] [26] are well known for solving time-dependent problems. In one-dimensional domains, these methods have been demonstrated to produce highly accurate solutions for many time-dependent problems. However, in two-dimensional and three-dimensional domains, not only more mathematic analysis is needed for unstructured grids but also carefully tuning of parameters to prevent mesh tangling is needed even for structured grids.

The basic idea of moving mesh algorithms is how best to represent the given data by a smooth function, by data points or by solution of a related PDE. One technique to develop a moving mesh algorithm is based on a so-called equidistribution principle, where nodes are relocated to equidistribute a given monitor function Υ. Many moving mesh techniques, including MMPDES, are based on this technique. If data is generated from a smooth function u, some possible candidates for monitor functions are $\Upsilon_1 = |\nabla u|$ and $\Upsilon_2 = (1 + |\nabla u|^2)^{1/2}$. In one-dimensional space, if Υ_1 is employed, the node movement tends to equidistribute function values u, and if Υ_2 is employed, the node movement tends to equidistribute the arc-length of u. Monitor functions related to some error measures are also popular [1]. In two-dimensional or three-dimensional space, there is still no rigorous definition and analysis of the equidistribution methodology.

The other technique to develop a moving mesh algorithm is based on direct minimization where nodes are relocated to minimize a measure of the error between the targeted function and its approximation. Moving mesh techniques such as MFE and

GWMFE are in this category. For self-adjoint problems, where the solutions can be obtained by minimizing a known energy functional, mesh movement based on direct minimization is a natural approach toward obtaining an optimal solution within a finite dimensional space. In [93] and [94], Baines, Tourigny and Hülsemann have shown that an energy functional decreases in a monotone fashion with their moving mesh algorithm.

For non self-adjoint problems such as convection-diffusion problems, the solutions are not derived from minimization of any energy functional. Theoretical analysis of moving mesh algorithms for such problems is an open question. Recently, Bank and Smith [10], Cao, Huang and Russell [24] have been employed an a posterior error indicator as a monitor function in their moving mesh strategies. Their approaches seem promising from the numerical results of some reaction-diffusion problems in their studies. Here, for coding simplicity, we study a moving mesh strategy proposed by Baine [53] and use the KS indicator as a monitor function.

First, let us briefly review the equidistribution principle in one-dimensional space. Let $x_j, j = 1, \cdots, n$ be a set of irregularly spaced grid points in $\Omega = [a, b]$. Suppose these points are related to the regularly spaced grid points $\xi_j, j = 1 \cdots n$ in the domain $\tilde{\Omega} = [0, 1]$ by discrete values of the continuous variable

$$\xi = \frac{\int_a^x \Upsilon(s)ds}{\int_a^b \Upsilon(s)ds}.$$

By differentiating the above equation twice, we obtain the mesh equation

$$\frac{d}{d\xi}(\Upsilon(x)\frac{dx}{d\xi}) = 0 \tag{3.22}$$

with boundary condition $x(0) = a$ and $x(1) = b$.

When $\Upsilon(x)$ is not constant, (3.22) is nonlinear and may be solved iteratively by the algorithm

$$\frac{d}{d\xi}(\Upsilon(x^p)\frac{dx^{p+1}}{d\xi}) = 0 \ (p = 0 \cdots),$$

with $x^0 = \xi$, provided it converges. When Υ is constant or piecewise constant, ie, Υ does not depend on x, the solution of (3.22) can be approximated directly by finite element or finite difference methods. Consider the monitor function

$$\Upsilon(x) = \eta_T \text{ for } x \in T.$$

Clearly, $\Upsilon(x)$ is a piecewise constant function. Linear finite element discretization of (3.22) give rises to the following tridiagonal linear system:

$$Tx = b, \text{ with } T_i = [\Upsilon(x_{i-\frac{1}{2}}), -\Upsilon(x_{j-\frac{1}{2}}) - \Upsilon(x_{j+\frac{1}{2}}), \Upsilon(x_{j+\frac{1}{2}})]. \tag{3.23}$$

If one solves (3.23) by iterative methods such as Jacobi or Gauss-Seidel, at kth iteration, node movement δx_j^k of node x_j^k, from one point Jacobi step, can be computed simply by a Υ weighted averaging on the adjacent nodes, i.e.

$$\delta x_j = x_j^{k+1} - x_j^k = \frac{\Upsilon(x^k_{j-\frac{1}{2}})(x_j^k - x_{j-1}^k) + \Upsilon(x^k_{j+\frac{1}{2}})(x_{j+1}^k - x_j^k)}{\Upsilon(x_{j-\frac{1}{2}}) + \Upsilon(x_{j+\frac{1}{2}})}.$$

The new location x_j^{k+1} of x_j^k can then be updated by $x_j^{k+1} = x_j^k + \gamma\delta x_j^k$ where $0 \leq \gamma \leq 1$ is the so-called relaxation parameter. Clearly, for $\gamma \leq \frac{1}{2}$, nodes remain ordered and mesh tangling is prevented. One step of point Gauss-Seidel is essentially the same as one point Jacobi step except node positions are updated immediately, and mesh tangling can not occur with this strategy.

For two-dimensional problems, there is no proper mathematical definition for equidistribution. On uniform grids, a useful grid adaption technique is to treat 2D "equidis-

tribution" as 1D equidistribution along the x-axis and y-axis separately [9], [106]. In other words, we have the following equations:

$$\nabla_\xi(\Upsilon(x,y)\nabla_\xi x) = 0 \tag{3.24}$$

and

$$\nabla_\eta(\Upsilon(x,y)\nabla_\eta y) = 0, \tag{3.25}$$

where x, y are coordinates on a domain $\Omega = [a_1, a_2] \times [b_1, b_2]$, and ξ, η are coordinates on the domain $\tilde{\Omega} = [0,1] \times [0,1]$. If boundary nodes are fixed, (3.24) and (3.25) have the following Dirichlet boundary conditions

$$x(0,\eta) = a_1, \quad x(1,\eta) = a_2, \quad , x(\xi,0) = x(\xi,1) = \xi,$$

and

$$y(\xi,0) = b_1, \quad y(\xi,1) = b_2, \quad , y(0,\eta) = y(1,\eta) = \eta,$$

respectively. If boundary nodes are allowed to move, the following Neumann condition can be posed:

$$x(0,\eta) = a_1, \quad x(1,\eta) = a_2, \quad , x_\eta(\xi,0) = x_\eta(\xi,1) = 0,$$

and

$$y(\xi,0) = b_1, \quad y(\xi,1) = b_2, \quad , y_\xi(0,\eta) = y_\xi(1,\eta) = 0.$$

Clearly, when Υ is piecewise constant, the analysis used in one dimensional case can be repeated here. Therefore, one step of point Jacobi or point Gauss-Seidel is again equivalent to a Υ-weighted averaging on the adjacent nodes.

Since an unstructured grid is a natural result from adaptive refinement process, we employee the following moving mesh algorithm:

1. Compute coordinates $\tilde{x}_j$, center point of T_j, and $\tilde{h}_j$, the smallest height of T_j, for all $T_j \in \omega_{x_i}$.

2. Let n_i be the number of elements in ω_{x_i} and $dx_j = \tilde{x}_j - x_i$.

3. Compute
$$\delta x_i = \frac{\sum_{j=1}^{n_i} \Upsilon(\tilde{x}_j) dx_j}{\sum_{j=1}^{n_i} \Upsilon(\tilde{x}_j)}$$

4. If $\|\delta x_i\| \leq \gamma \min_{1 \leq j \leq n_i} \tilde{h}_j$, then $x_i^{new} = x_i + \delta x_i$.
Otherwise, $x_i^{new} = x_i + (\gamma \min_{1 \leq j \leq n_i} \tilde{h}_j) \frac{\delta x_i}{\|\delta x_i\|}$, where $\gamma < 1$ is the relaxation parameter.

Algorithm 3.4.1: Moving mesh algorithm

The algorithm is basically the same as the moving mesh algorithm in [53] except the monitor function is replaced by the KS error indicator. In [53], the relaxation parameter γ is set to 0.5 and the location of each node is updated after all moving directions are calculated (Jacobi type). In our numerical tests, we set $\gamma = 0.6$ and the location of each node is updated immediately after its moving direction is computed (Gauss-Seidel type).

Two numerical tests are presented here. The first problem is Problem 2 in Chapter 2 with $\epsilon = 10^{-4}$. The second problem is a variant of the "IAHR/CEGB" workshop problem [91] as follows,

Problem 3: Flow with curved internal layer and boundary layer

$$-\epsilon \cdot \Delta u + \beta \cdot \nabla u = 0 \text{ on } \Omega = [-1, 1] \times [0, 1],$$

where $\epsilon = 10^{-4}$ and $\beta = (2y(1 - x^2), -2x(1 - y^2))$. The boundary conditions are given as $u|_{\Gamma_1} = 1$, $u|_{\Gamma_3} = 0$ and $\frac{\partial u}{\partial n}|_{\Gamma_2} = 0$, where

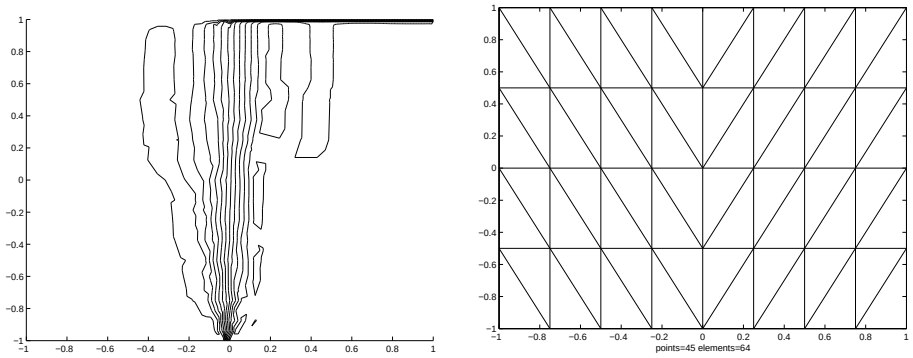
$$\begin{aligned} \Gamma_1 &= \{(x, y) \in \partial\Omega | x = 1 \text{ or } -0.5 \leq x \leq 0 \cap y = 0\} \\ \Gamma_2 &= \{(x, y) \in \partial\Omega | 0 < x \leq 1 \cap y = 0\} \\ \Gamma_3 &= \partial\Omega - (\Gamma_1 \cup \Gamma_2). \end{aligned}$$

.

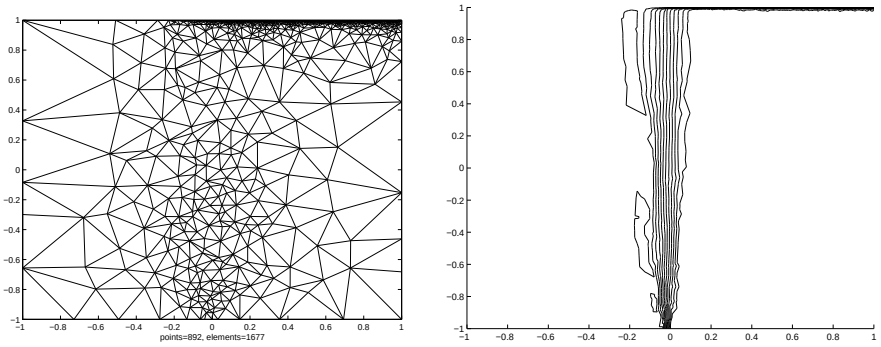
In both problems, two moving mesh steps are performed and these are followed by one local optimization procedure (LOP), so-called edge swaps strategy introduced by Lawson [64], [12], [82], before each mesh refinement step. We call this process,

$$\text{two moving mesh steps} \longrightarrow \text{LOP} \longrightarrow \text{mesh refinement},$$

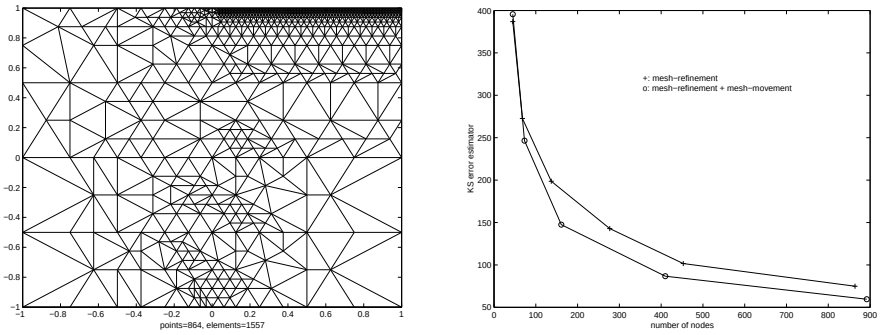
moving mesh refinement. In order to compare moving mesh refinement and regular refinement, we carefully choose refinement steps and threshold values θ in the maximum marking strategy so that both methods produce a similar number of nodes in the finest meshes. Four moving mesh refinement steps are performed for both problems, six regular refinement steps are performed for Problem 2 and seven regular refinement steps are perform for Problem 3. In both refinement methods, the threshold value θ in the maximum marking strategy (3.1) equals to 0.25. To access solution accuracy, since there is no mathematical expression for the exact solution, the KS error estimator is used to represent the true error in our tests. Clearly, from Figure 3.1 and Figure 3.2, we can see that the mesh movement strategy improves solution quality. Moreover, the error from moving mesh refinement is less than the error from regular mesh refinement.



(a) solution on $\Im_h$ (b) initial mesh

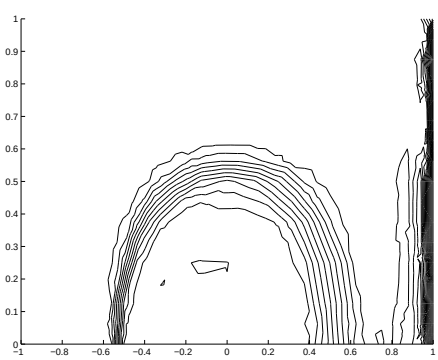


(c) moving mesh $\Im_h^m$ (d) solution on $\Im_h^m$

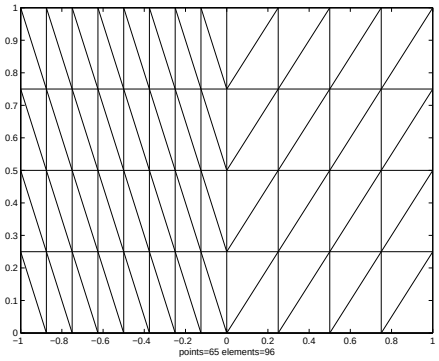


(e) regular mesh $\Im_h$ (f) error comparison

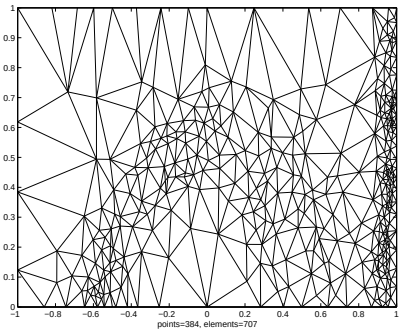
Figure 3.1: fixed mesh refinement vs moving mesh mesh refinement



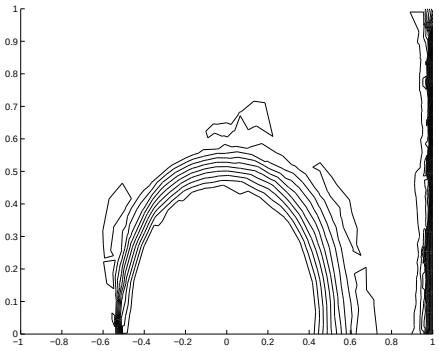
(a) solution on $\Im_h$



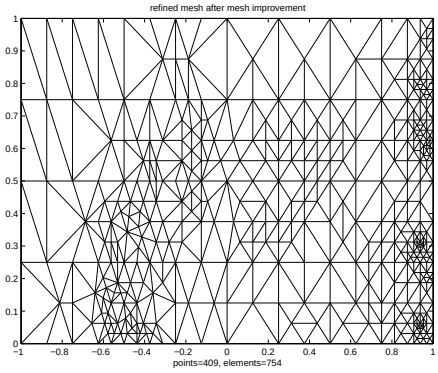
(b) initial mesh



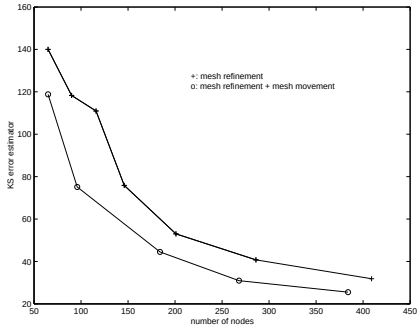
(c) moving mesh $\Im_h^m$



(d) solution on $\Im_h^m$



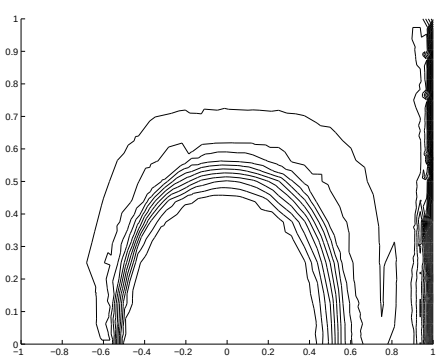
(e) regular mesh $\Im_h$



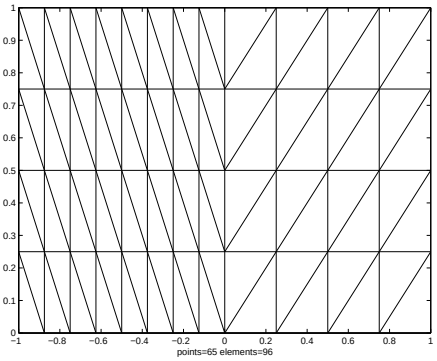
(f) error comparison

Figure 3.2: fixed mesh refinement vs moving mesh mesh refinement

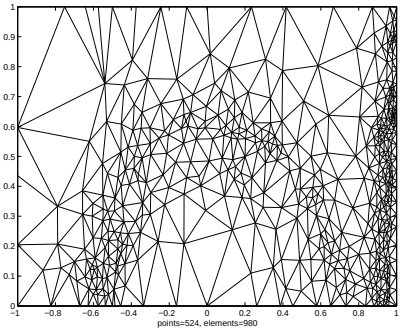
Although moving mesh strategies may increase the solution accuracy without increasing the number of nodes, there are still some disadvantages. For example, to successfully improve the solution accuracy, a carefully chosen relaxation parameter is needed especially for problems in two or more dimensions. To demonstrate the importance of choosing proper relaxation parameter, we solve Problem 3 on two meshes, one from moving mesh refinement with relaxation parameter $\gamma = 0.5$ and the other from regular mesh refinement. The errors of these two solutions are plotted in Figure 3.3 (f). Clearly, unlike what is shown in Figure 3.2 (f), the error from moving mesh refinement is no longer strictly less than the error from regular refinement along the whole refinement process. This is caused by only a small change of γ!



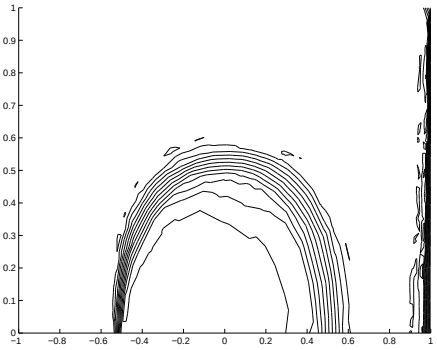
(a) solution on $\Im_h$



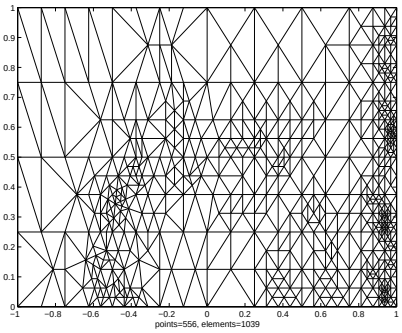
(b) initial mesh



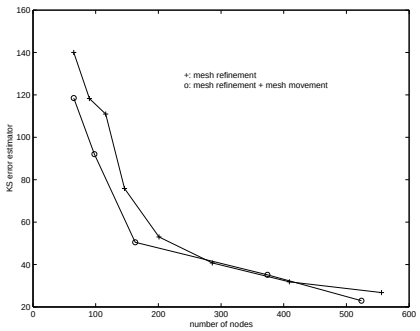
(c) moving mesh $\Im_h^m$



(d) solution on $\Im_h^m$



(e) regular mesh $\Im_h$



(f) error comparison

Figure 3.3: fixed mesh refinement vs moving mesh mesh refinement

3.5 Error-Adapted Mesh Refinement Strategy

The computational overhead of moving mesh strategies is $C_{mv} \times N$ where N is the total number of nodes and C_{mv} is the cost of computing a move direction. In addition to the drawback that the moving mesh in Section 3.4 is very sensitive to the relaxation parameter, if C_{mv} is large and, in addition, one has an efficient linear solver, such as multigrid methods, for the convection-diffusion equation, then the cost for mesh movement will be high compared to the cost of solving the linear system. Moreover, if one would like to use multigrid methods to solve the sparse linear system, expensive interpolation must be computed for each moving mesh step on all grid levels because the grids after mesh movement are no longer nested. In adaptive refinement process, it may be more desirable to increase the accuracy of the approximate solutions without reducing the efficiency of linear solvers.

In this section, we propose an error-adapted mesh refinement strategy in which new nodes are added to marked edges adaptively, according to the distribution of errors. The cost for computing interpolation in our method is basically free. We also expect nodes will cluster to the region where error is large in the adaptive refinement process. In the following, we present the idea of our error-adapted refinement strategy and some numerical tests.

Suppose $e_{i,j}$ is an edge in a marked element T with end points p_i and p_j. In regular refinement and longest side bisection method, a new node p^{mid} is always inserted in the mid-point of $e_{i,j}$. In the new algorithm, the location of new node on edge $e_{i,j}$ is determined by recovered error estimator η_i and η_j on nodes p_i and p_j respectively, where the recovered error estimator is computed from an area-weighted averaging

of η_T over its adjacent elements, $T \in \omega_i$, for any node p_i. The basic idea behind this algorithm comes from a tension spring model. One can think of each edge as a tension spring connecting its end points. The newly added node p^{new} is located at the mid-point of $e_{i,j}$, initially. When errors are uniformly distributed across edge $e_{i,j}$, no force is introduced and $p^{new} = p^{mid}$. Otherwise, we consider $F_{i,j} = (\eta_j - \eta_i)\frac{\vec{e}_{i,j}}{|e_{i,j}|}$, where $\vec{e}_{i,j} = p_j - p_i$, as an external force posed on p^{mid} and move p^{mid} to the equilibrium of the simple tension spring system on edge $e_{i,j}$. Hence, the displacement $\delta x_{i,j}$ can be computed as

$$\delta x_{i,j} = \frac{1}{2K_{i,j}} F_{i,j}, \text{ where } K_{i,j} \text{ is the tension constant of edge } e_{i,j},$$

and the location of new node p^{new} can be updated by

$$p^{new} = p^{mid} + \delta x_{i,j}.$$

It is possible that p^{new} is located outside of $e_{i,j}$ and produces mesh tangling. Here, for simplicity, we set $K_{i,j} = 1$ on all $e_{i,j}$, and modify the external force as follows:

$$F'_{i,j} = \begin{cases} (1 - (\frac{\min_{e \in E_h^*} |F_{i,j}|}{|F_{i,j}|})^\alpha)\vec{e}_{i,j}, & \text{if } e_{i,j} \in E_h^* \text{ and } \eta_j > \eta_i \\ (1 - (\frac{\min_{e \in E_h^*} |F_{i,j}|}{|F_{i,j}|})^\alpha)\vec{e}_{j,i}, & \text{if } e_{i,j} \in E_h^* \text{ and } \eta_j < \eta_i \\ 0, & \text{otherwise}, \end{cases} \tag{3.26}$$

where $0 \leq \alpha \leq 1$ is a relaxation parameter. Clearly, $0 \leq F'_{i,j} < 1$, and the displacement $\delta x_{i,j}$ can now be safely computed as

$$\delta x_{i,j} = \frac{1}{2} F'_{i,j} \vec{e}_{i,j}.$$

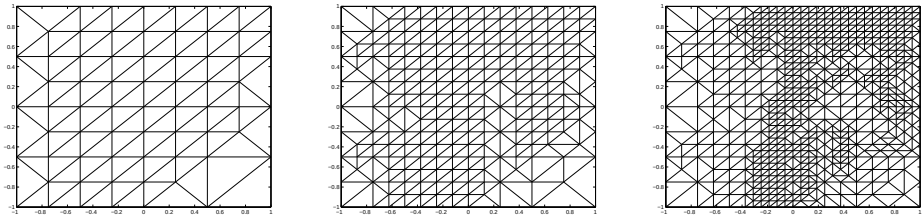
As a result, p^{new} will be always located in $e_{i,j}$.

Remark 3.5.1 *The external force in (3.26) has very little effect on determining the location of new nodes, for small α, i.e., $\alpha \to 0$. On the other hand, for large α,*

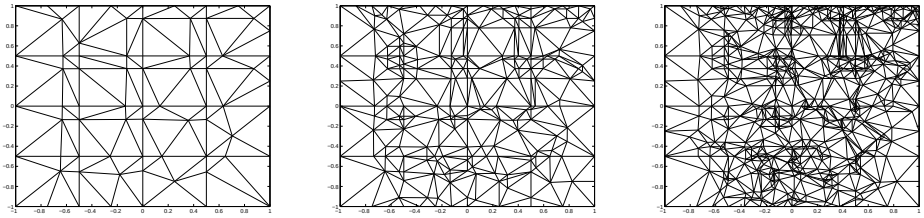
i.e., $\alpha \to 1$*, the newly inserted nodes can be moved away from the mid points of the marked edges. Hereafter, we call* α *the error-sensitivity parameter.*

To preserve the quasi-uniform structure of the refined mesh outside regions containing layer, a small threshold value in the maximum marking strategy and large error-sensitivity parameter $\alpha \approx 1$ should not be combined. Otherwise, long-thin elements may also appear outside the layer regions from our error-adapted refinement process and further degrade the solution quality. With careful chosen error-sensitivity parameter, our numerical results show the error-adapted refinement strategy quickly cluster new nodes to layer regions and still maintain good quality mesh in the other regions.

In the following, first, an example is given to demonstrate the importance of the error-sensitivity parameter in our error-adapted refinement strategy. We solve Problem 2 with $\epsilon = 10^{-4}$ on both regular refined meshes and error-adapted refined meshes generated by the KS-estimator. Three refined meshes are plotted for each refinement strategy. In Figure 3.4, a threshold value $\theta = 0.05$ is used in the maximum marking strategy and the error-sensitivity parameter α is equal to 1. One can see that serious mesh distortion appears on the whole domain. However, in Figure 3.5, with a threshold value $\theta = 0.25$ and $\alpha = \frac{1}{3}$, it is clear that the error-adapted mesh refinement clusters nodes to layer regions and still maintains good mesh-quality mesh outside layer regions.

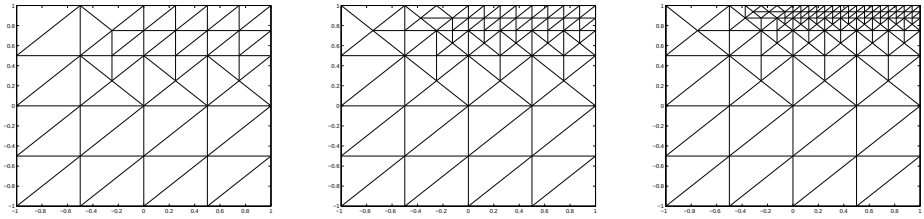


(a) Regular refinement

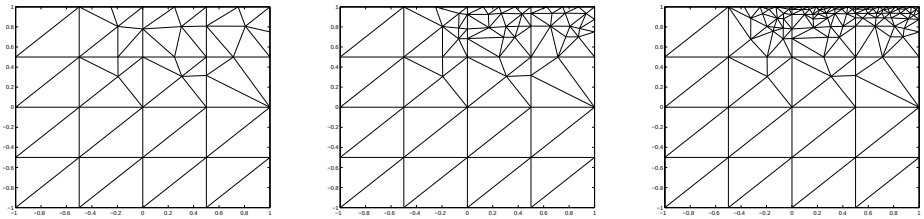


(b) Error-adapted refinement

Figure 3.4: Regular refinement vs Error-adapted refinement: $\theta = 0.05$ and $\alpha = 1$ for Problem 2



(a) Regular refinement



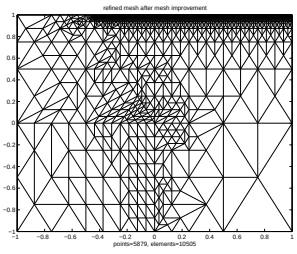
(b) Error-adapted refinement

Figure 3.5: Regular refinement vs Error-adapted refinement: $\theta = 0.25$ and $\alpha = \frac{1}{3}$ for Problem 2.

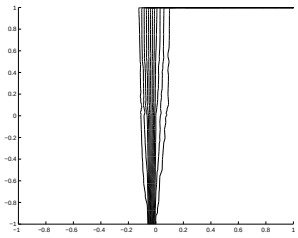
Second, we present more examples to show that the error-adapted mesh refinement is good for resolving boundary layers. In these examples, if the error-adapted refinement is employed, the refinement strategy is replaced by Longest-Side Bisection algorithm when the minimal height of the triangles is less than $\frac{\epsilon}{2}$. Also, the KS-estimator is used in these examples.

First, consider constant flow problems such as Problem 2 in Chapter 2 where both an exponential boundary layer and an parabolic internal layer exist. In this problem, since the wind β is perpendicular to the wall y=1, the term $\frac{h}{\epsilon}\left\|\beta \cdot \nabla e_h\right\|_0$ in the a posteriori lower error bound (3.15) is expected to be dominant. Therefore, it is not surprising that the error indicator η_T in the boundary layer near the wall y=1 has extremely large value compared to η_T in other regions. In this case, our error-adapted mesh refinement process is able to cluster new nodes to the boundary layer region efficiently as seen in the following results. Two test cases, $\epsilon = 10^{-4}$ and $\epsilon = 10^{-3}$, are given. In both cases, the error-sensitivity parameter α is set to $\alpha = \frac{1}{3}$.

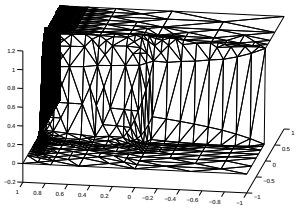
In the case of $\epsilon = 10^{-3}$, 10 refinement steps are performed with marking threshold value $\theta = 0.25$. Both algorithms are able to resolve the boundary layer. However, from Figure 3.6 , it is clear that the error-adapted mesh provides higher resolution near the boundary point $(0, 0)$, where the jump discontinuity appears. Moreover, the regular refinement algorithm generates 5979 node points whereas only 1973 nodes are generated by our refinement algorithm.



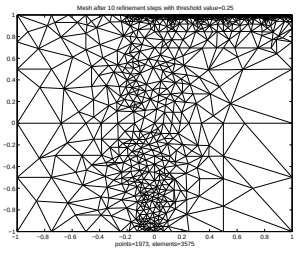
(a) Isotropic mesh



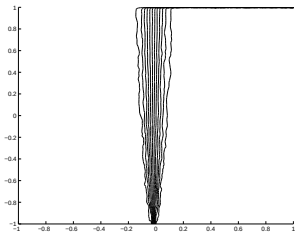
(b) Contour plot of solution



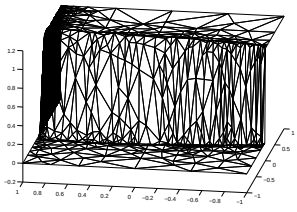
(c) 3D representation of solution



(d) Error-adapted mesh



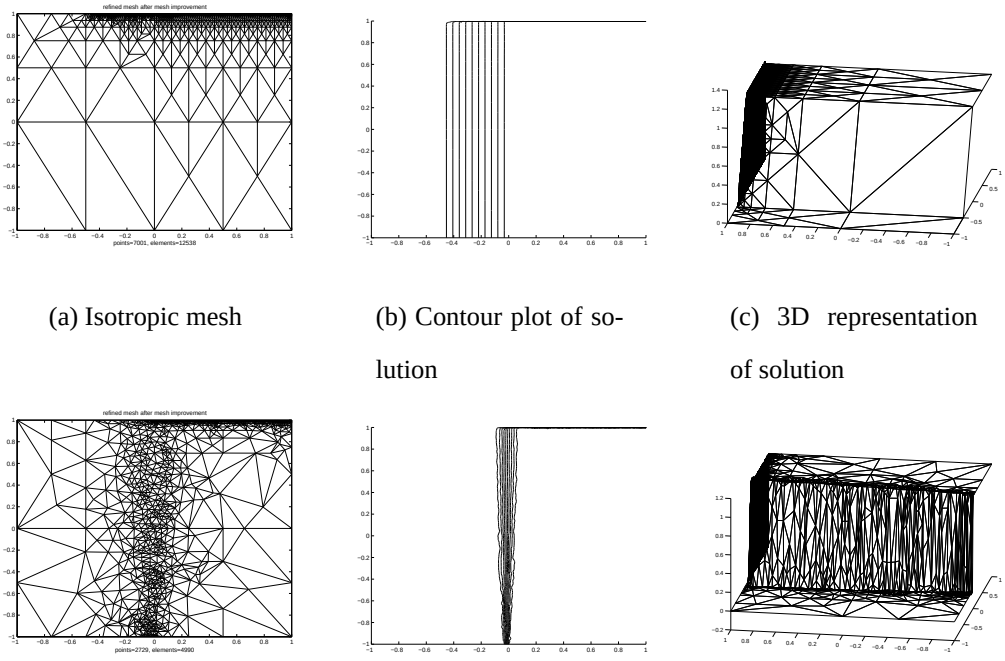
(e) Contour plot of solution



(f) 3D representation of solution

Figure 3.6: Isotropic refinement vs error-adapted refinement for the case $\epsilon = 10^{-3}$

For the case of $\epsilon = 10^{-4}$, in general, it is hard to fully resolve the boundary layer and produce an accurate internal layer without paying an extremely high computing cost. Here, from Figure 3.7, a clear internal layer can be seen from the solution on the mesh generated by the error-adapted algorithm. The solution on the mesh generated by regular refinement fails to resolve the internal layer. Again, only 2729 nodes are generated by our algorithm compared 7001 nodes from regular refinement. In this numerical test, 16 refinement steps are performed with marking threshold value $\theta = 0.5$.



(a) Isotropic mesh

(b) Contour plot of solution

(c) 3D representation of solution

(d) Error-adapted mesh

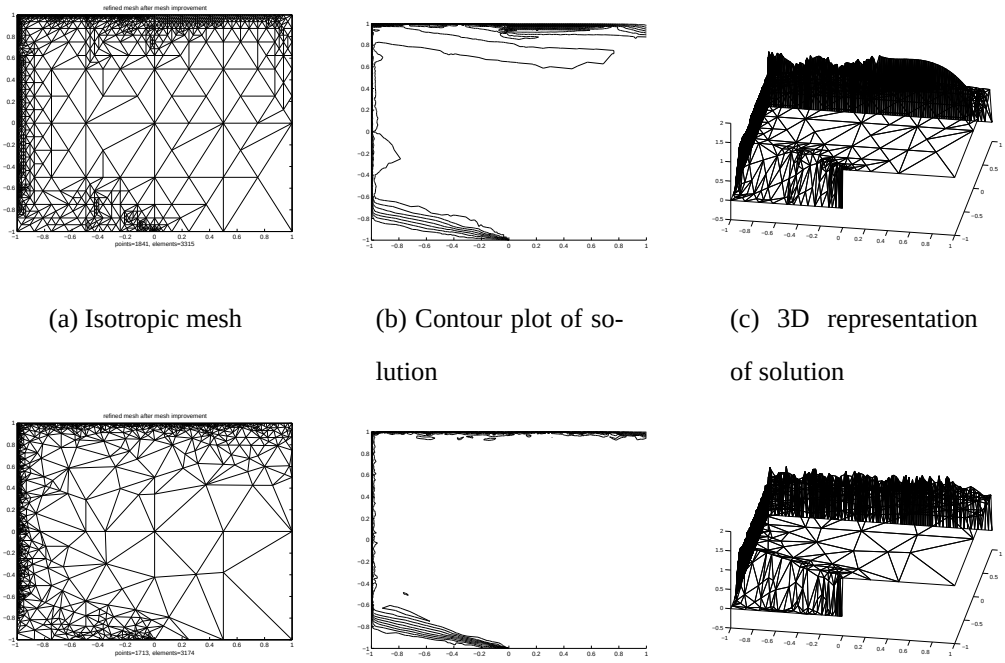
(e) Contour plot of solution

(f) 3D representation of solution

Figure 3.7: Isotropic refinement vs error-adapted refinement for the case $\epsilon = 10^{-4}$

Another constant flow problem in our tests is the same problem as Problem 2 except the wind β is $(\cos(\frac{5}{6}\pi), \sin(\frac{5}{6}\pi))$. In this example, $\epsilon = 10^{-4}$ and 12 refinement steps are performed with threshold value $\theta = 0.5$ and error-sensitivity parameter $\alpha = \frac{1}{3}$. Again, the solution from the error-adapted refinement algorithm is better as shown in Figure 3.8.

Next consider Problem 3, the "IAHR/CEGB" workshop problem [91]. With a curved internal layer due to a jump discontinuity on the Dirichlet boundary and an exponential layer on the hot wall $x = 1$, this problem not only can be used to test discretization strategies but also can be a challenge problem to our error-adaptive mesh refinement strategy. Unlike the constant flow problems, where β is perpendicular to the wall, β



(a) Isotropic mesh

(b) Contour plot of solution

(c) 3D representation of solution

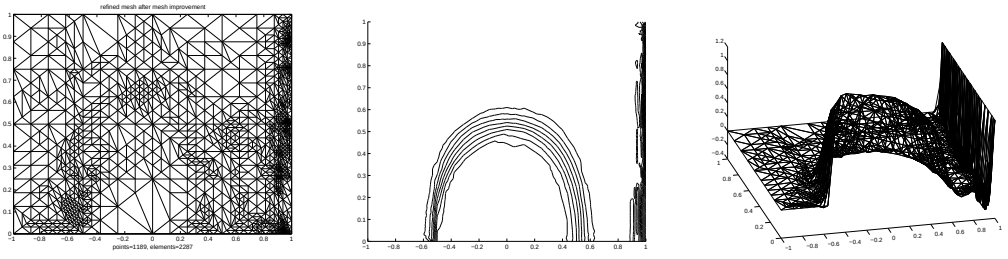
(d) Error-adapted mesh

(e) Contour plot of solution

(f) 3D representation of solution

Figure 3.8: Isotropic refinement vs error-adapted refinement for the case $\epsilon = 10^{-4}$

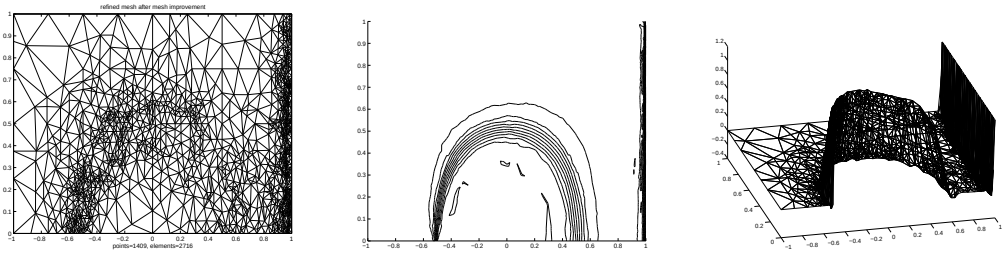
is parallel with the wall $x = 1$ in this case. Therefore, $\frac{h}{\epsilon} \|\beta \cdot \nabla e_h\|_0$ may no longer be the dominating term in the a posteriori lower error bound, i.e., $\frac{h}{\epsilon} \|R_T - R_T^0\|_{0,T}$ cannot be treated as a low order term. In this situation, we can not expect the error indicator η_T to be significantly larger in the layer region near the wall $x = 1$ than η_T in the internal layer region. As a result, if we try to resolve the exponential layer more quickly in boundary layer region by increasing the mesh error-sensitivity parameter α, some anisotropic elements may appear in the interior region, where isotropic elements are desirable, this leads to larger errors in these regions. In our numerical tests, a small $\alpha = 1/8$ is chosen and 8 mesh refinement steps are performed. Figure 3.9 shows that only errors in boundary layer are reduced significantly by our new refinement strategy in this case.



(a) Isotropic mesh

(b) Contour plot of solution

(c) 3D representation of solution



(d) Error-adapted mesh

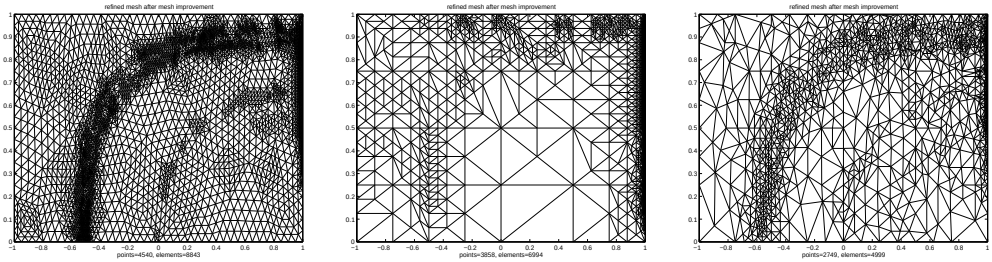
(e) Contour plot of solution

(f) 3D representation of solution

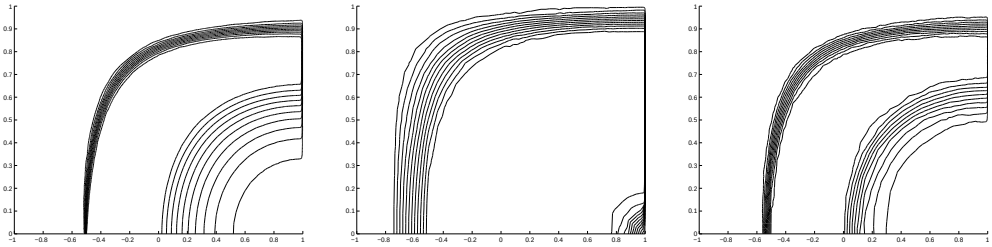
Figure 3.9: Isotropic refinement vs error-adapted refinement

The next problem is very similar to the above "IAHR/CEGB" workshop problem with $\epsilon = 10^{-3}$. The only differences are that the wind β is changed to $(y(4-(1-x)^2), 2(1-x)(1-y^2))$ and the hot wall boundary condition, $u = 1$ on $x = 1$, is replaced by a cold wall with $u = 0$ on $x = 1$. In this problem, β is now perpendicular to the wall $x = 1$. Therefore, $\frac{h}{\epsilon} \|\beta \cdot \nabla e_h\|_0$ is again the dominant term and a large error indicator η_T in boundary layer is expected. As shown in the first problem, the error-adaptive mesh refinement algorithm should be able to cluster node points in the boundary region efficiently. In this numerical test, the mesh error-sensitivity parameter $\alpha = \frac{1}{3}$. First, a fine initial mesh is generated followed by three regular refinement steps. The solution computed on this fine mesh, denoted by $\Im_0$, is then considered to

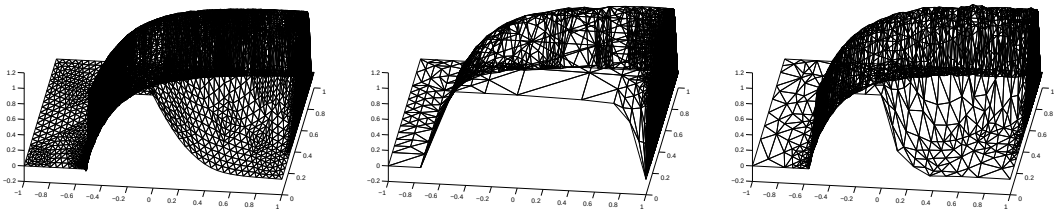
be the exact solution. Two meshes, $\Im_1$ and $\Im_2$, are generated from a 4x4 initial mesh with same threshold value $\theta = 0.25$. Eight refinement steps are performed to generate regular-refined mesh $\Im_1$ and fourteen refinement steps are performed to generate error-adapted mesh $\Im_2$. Again, from Figure 3.10, we can see the solution from regular mesh refinement, with 2858 node points, fails to present accurate internal layer structure. In contrast, the solution on error-adapted refined mesh, with 2749 node points, shows both accurate internal layer and boundary layer.



(a) mesh $\Im_0$ (b) mesh $\Im_1$ (c) mesh $\Im_2$



(d) Contour plot of solution on $\Im_0$ (e) Contour plot of solution on $\Im_1$ (f) Contour plot of solution on $\Im_2$



(g) 3D representation of solution on $\Im_0$ (h) 3D representation of solution on $\Im_1$ (i) 3D representation of solution on $\Im_2$

Figure 3.10: Isotropic mesh refinement vs error-adapted mesh refinement

Chapter 4

Methods for Solving Sparse Systems

In this chapter, we study several linear iterative methods for solving the linear system,

$$Au = f, \tag{4.1}$$

Our goal is to find out which one is the most suitable solver on the adapted refined mesh for the matrix $A = A_{SD}$ arising from SDFEM discretization of the convection-diffusion equation.

First we introduce the stationary iterative methods based on matrix splittings $A = M - N$. The popular Jacobi and Gauss-Seidel methods belong to this category. It is well known that if the matrix A is an M-matrix, these types of iteration methods converge. Moreover, the Stein-Rosenberg theorem implies the Gauss-Seidel method converges faster than the Jacobi method. However, for the convection-dominate flow problems, the matrix A_{SD} is only a positive definite matrix, due to the coercivity of B_{SD}), but not an M-matrix. As a result, it is difficult to show the stationary iterative methods converge. In fact, Bey has shown that there exists a positive define matrix for which the Gauss-Seidel method never converges but the Jacobi method converges [13]. In addition, although the flow-oriented Gauss-Seidel method shows good convergence in many numerical studies for simple flows, the node numbering becomes

more difficult for complex flows such as flows with closed characteristics.

Second, we study Krylov-subspace methods such as the generalized minimum residual method (GMRES) [87], which is a natural candidate for solving a nonsymmetric linear system. In theory, this method is guaranteed to converge and the convergence rate can be bounded in terms of the spectrum of A and the condition number of the eigenvectors. Even though the estimated convergence rate may much greater than the actual convergence rate [41], it may still reveal the fact that the convergence rate can be slow for small mesh sizes and large convection for the convection dominant problems. As a result, the computation cost may become too expensive. One way to improve the convergence rate of GMRES is by using preconditioning. Instead of solving the linear system $Au = f$, one can solve the linear system $M^{-1}Au = M^{-1}f$ where the preconditioning matrix M is nonsingular. If $M^{-1}A \approx I$ and $M^{-1}A$ is closer to a normal matrix, one would expect an improved convergence rate. Good preconditioning matrices can be derived from a convergent stationary iteration or from an incomplete LU factorization if A is an M-matrix. Although this is not the case for the convection-dominated flow, numerical studies in [89] still show these preconditioners are robust.

Unlike stationary iterative methods and Krylov subspace methods, where the convergence rates decrease as the mesh is refined, multigrid methods (MG) are well known for having a mesh-independent convergence property for self-adjoint elliptic problems if the solution u has H^2 regularity, i.e.

$$\|u\|_2 < c_0 \|f\|_0 . \tag{4.2}$$

For problems with solution $u \in H^{1+\alpha}(\Omega), 0 < \alpha < 1$, the mesh-independent conver-

gence can still be shown [19] if the bilinear form B of the associated partial differential equation has strong coercivity and continuity, namely there exist constants $0 \ll c_1$, $c_2 \ll \infty$ such that for all $v \in V_h \subset H^{1+\alpha}$

$$c_1 \|v\|_1^2 < B(v,v) < c_2 \|v\|_1^2 . \tag{4.3}$$

For non-self-adjoint elliptic problems, if the skew-symmetric part of the operator can be treated as a small perturbation term, for problems that are diffusion-dominated, the MG convergence is still mesh-independent as shown in [17], [20]. Unfortunately, in the convection dominant case, MG convergence can not be proved due to the fact that the constant $c_0 \approx P_e^{3/2}$ and $c_1 \approx \epsilon$. However, MG uniform convergence can still be achieved by using special gridding techniques, for example, using meshes obtained from semi-coarsening [80] and Shishkin meshes [46] with operator-dependent interpolations. This is because those techniques improve the regularity of the discrete solution in the sense that the coarse grid provides a better approximation for the error on the fine grid. On the other hand, without knowing such a priori formulated grids, algebraic multigrid (AMG) [86] first defines the algebraic smooth error, then selects a set of grid points to interpolate these smooth error. Although the convergence results of AMG has only been established for M-matrices with a 2-level scheme, AMG convergence still appears to be essentially independent of mesh size in many numerical studies [62].

In the following, we present implementations and convergence results of each linear solver. For simplicity, only Problem 2 in Section 2.3 on a uniform $N \times N$ rectangular meshes is discussed in our analysis. Numerical results compare the performance of these solvers on the adaptive refined mesh.

4.1 Stationary Iteration Methods

Given a matrix splitting $A = M - N$, the corresponding stationary iteration for (**??**) is written as

$$Mu_{m+1} = Nu_m + f. \tag{4.4}$$

By subtracting Mu_m from both sides of (4.4), we have the alternative form:

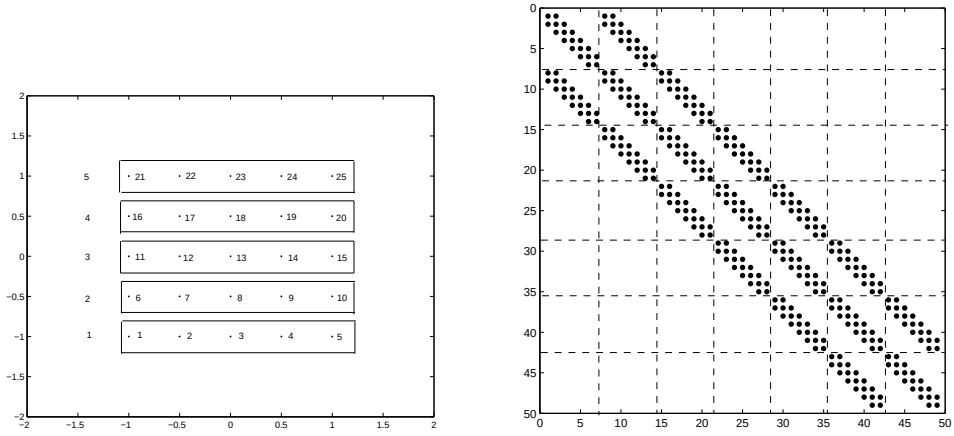
$$u_{m+1} = u_m + M^{-1}(f - Au_m). \tag{4.5}$$

One can think of the matrix M as an approximation of A. If $M = A$, $u^1 = u_h$ and the equation (4.5) represents a direct solver. Let A_D and A_L denote the diagonal matrix and the lower triangular matrix of the matrix A respectively, and let I be the identity matrix. The following Table 4.1 shows some of the well known stationary iteration methods in terms of the choice of matrix M:

Jacobi	$M = A_D$
Damped Jacobi	$M = \omega^{-1}A_D$ where $0 < \omega < 1$
Gauss-Seidel	$M = A_D + A_L$
Successive Over-Relaxation	$M = \omega^{-1}(A_D + \omega A_L)$ where $0 < \omega < 2$
Richardson	$M = \omega^{-1} \|A\|$I where $0 < \omega < 2$

Table 4.1: Stationary iterative methods

One can also partition the mesh into a set of independent blocks which induces a block partitioning of A. Table 4.1 can also represent the block-version of those iterative methods with A_D denoting the block diagonal matrix of A and A_L denote the block lower triangular matrix of A.



(a) Lexicographical node ordering (b) Block partition of matrix A

Figure 4.1:

For example, if one partitions the uniform mesh into a union of horizontal lines and numbers the grid points in lexicographical order, as shown in Figure 4.1, then the discrete matrix A_{SD} in (2.28) of Problem 2, with stabilization parameter $\delta_T = \frac{h}{2}$, can be represented in the following matrix form:

$$A_{SD} = \begin{bmatrix} D & -U & & & & \\ -L & D & -U & & 0 & \\ & -L & \ddots & \ddots & & \\ & & \ddots & \ddots & -U & \\ & 0 & & -L & D & -U \\ & & & & -L & D \end{bmatrix} \tag{4.6}$$

The blocks are tridiagonal matrices

$$\begin{aligned} D &= h \times tridiag[\tfrac{1}{6} - \tfrac{1}{3}\tfrac{\epsilon}{h}, \tfrac{2}{3} + \tfrac{8}{3}\tfrac{\epsilon}{h}, \tfrac{1}{6} - \tfrac{1}{3}\tfrac{\epsilon}{h}], \\ L &= h \times tridiag[\tfrac{1}{6} + \tfrac{1}{3}\tfrac{\epsilon}{h}, \tfrac{2}{3} + \tfrac{1}{3}\tfrac{\epsilon}{h}, \tfrac{1}{6} + \tfrac{1}{3}\tfrac{\epsilon}{h}], \\ U &= \epsilon \times tridiag[\tfrac{1}{3}, \tfrac{1}{3}, \tfrac{1}{3}]. \end{aligned} \tag{4.7}$$

The block diagonal matrix, block lower triangular matrix and block upper triangular matrix of A_{SD} is defined as follows:

$$A_D = \begin{bmatrix} \mathrm{D} & & & \\ & \mathrm{D} & & \\ & & \ddots & \\ & & & \mathrm{D} \end{bmatrix}, \; A_L = \begin{bmatrix} 0 & & & \\ -\mathrm{L} & 0 & & \\ & \ddots & \ddots & \\ & & -\mathrm{L} & 0 \end{bmatrix} \text{ and } A_U = \begin{bmatrix} 0 & -\mathrm{U} & & \\ & 0 & \ddots & \\ & & \ddots & -\mathrm{U} \\ & & & 0 \end{bmatrix}$$

The block Gauss-Seidel method is then defined by (4.5) with $M = A_D + A_L$. Because each block consists of nodes on a horizontal line, this block Gauss-Seidel method is called the horizontal line Gauss-Seidel (HGS) method. If the node ordering in HGS is reversed, we call the resulting block Gauss-Seidel method the backward HGS. Similarly, one can define another block Gauss-Seidel method where each block consists of nodes on a vertical line. This block Gauss-Seidel method is then called the vertical line Gauss-Seidel (VGS) method. Again, by reversing the node ordering, one obtains the backward VGS. For general convergence analysis of the stationary iterative methods, we refer to Chapter 4 [50] by Hackbusch.

It has been shown that the HGS method converges for our model problem on a uniform mesh with mesh size $h \ll \epsilon$ [37]; we consider this method in the following analysis and also allow mesh sizes $h > \epsilon$. In order to analyze the convergence of HGS, the equation (4.5) is rewritten in the error reduction form,

$$e_{m+1} = (I - M^{-1} A_{SD}) e_m, \tag{4.8}$$

where $e_i = u - u_i$ is the iterative error at ith iteration, by subtracting u from (4.5). By direct computing, the error reduction operator $E^s = I - (A_D + A_L)^{-1} A_{SD} = -(A_D + A_L)^{-1} A_U$ can be written in the following matrix form:

$$E^s = G_1 \cdot G_2, \tag{4.9}$$

where

$$G_1 = \begin{bmatrix} 0 & \mathrm{I} & & & \\ 0 & \mathrm{D^{-1}L} & \ddots & & \\ \vdots & \vdots & \ddots & \mathrm{I} & \\ 0 & (\mathrm{D^{-1}L})^{\mathrm{n-2}} & \cdots & \mathrm{D^{-1}L} & \mathrm{I} \\ 0 & (\mathrm{D^{-1}L})^{\mathrm{n-1}} & \cdots & (\mathrm{D^{-1}L})^2 & \mathrm{D^{-1}L} \end{bmatrix} \text{ and } G_2 = \begin{bmatrix} 0 & & & & \\ & \mathrm{D^{-1}U} & & & \\ & & \ddots & & \\ & & & \mathrm{D^{-1}U} & \\ & & & & \mathrm{D^{-1}U} \end{bmatrix}$$

From (4.9), the convergence results of the HGS iterative method can be shown by estimating $\|G_1\|$ and $\|G_2\|$. In the following, $\|\cdot\|$ represents the matrix L^2 norm or vector Euclidian norm depending on either input argument is a matrix or a vector.

From (4.7), D and U are symmetric. The following inequalities

$$\begin{aligned} \|U\| &= \rho(U) \leq \epsilon \\ \left\|D^{-1}\right\| &= \rho(D^{-1}) = \frac{1}{\lambda_{min}(D)} < \frac{1}{\frac{h}{3} + \frac{10\epsilon}{3}} \approx \frac{3}{h}, \end{aligned}$$

for $h \gg \epsilon$, follow directly from the Gerschgorin circle theorem. Therefore, we have

$$\|G_2\| \leq 3\frac{\epsilon}{h}. \tag{4.10}$$

To estimate $\|G_1\|$, the following lemmas are needed.

Lemma 4.1.1 *Given two symmetric matrices B_1 and B_2. Assume that $B_1, B_2 \geq 0$, B_1 is irreducible and B_2 is positive definite. The following properties hold.*

1. *There exist a positive eigenvector x^+ such that*

$$B_2^{-1}B_1x^+ = \rho(B_2^{-1}B_1)x^+ \tag{4.11}$$

2. *If $\alpha I - B_2^{-1}B_1$ is non-singular and $(\alpha I - B_2^{-1}B_1)^{-1} \geq 0$ then $\rho(B_2^{-1}B_1) < \alpha$.*

Proof: The existence of a positive eigenvector satisfying (4.11) is essentially a generalization of the well-known Perron and Frobenius theorem ([95] Theorem 2.7). Using (4.11), one can prove the second result using a standard argument of Perror-Frobenius theory, see Theorem 3.16 in [95].

□

Lemma 4.1.2 *Let L and D be the matrices defined in (4.7). For any $\delta \geq (1+\frac{2\epsilon}{h})\frac{h}{\epsilon}$, the matrix $\delta(D-L)-D$ is an M-matrix.*

Proof: Let us choose $\delta = \frac{h\gamma}{\epsilon}$ for some $\gamma > 0$. From (4.7), $D-L = \frac{\epsilon}{3}\times tridiag[-2,7,-2]$. Therefore, we have

$$\begin{aligned}\delta(D-L)-D &= h\times tridiag[\frac{-2\gamma}{3},\frac{7\gamma}{3},\frac{-2\gamma}{3}] - h\times tridiag[\frac{1}{6}-\frac{\epsilon}{3h},\frac{2}{3}+\frac{8\epsilon}{h},\frac{1}{6}-\frac{\epsilon}{3h}]\\ &= h\times tridiag[-(\frac{2\gamma}{3}+\frac{1}{6}-\frac{\epsilon}{3h}),\frac{7\gamma}{3}-\frac{2}{3}-\frac{8\epsilon}{3h},-(\frac{2\gamma}{3}+\frac{1}{6}-\frac{\epsilon}{3h})].\end{aligned}$$

Since

$$\frac{7\gamma}{3}-\frac{2}{3}-\frac{8\epsilon}{3h}-2(\frac{2\gamma}{3}+\frac{1}{6}-\frac{\epsilon}{3h}) = \gamma-1-\frac{2\epsilon}{h},$$

clearly, for $\gamma \geq 1+\frac{2\epsilon}{h}$, the matrix $\delta(D-L)-D$ is irreducible and weakly diagonal dominant. This implies the matrix $\delta(D-L)-D$ and $(\delta(D-L)-D)^{-1}$ are positive definite. Moreover, since the off-diagonal entries of $(\delta(D-L)-D)$ are all negative, the matrix $\delta(D-L)-D$ is an M-matrix for $\delta \geq (1+\frac{2\epsilon}{h})\frac{h}{\epsilon}$.

□

Now, we estimate $\|G_1\|$ in the following. First, let us estimate $\|D^{-1}L\|$. Considering $D^{-1}L = I - D^{-1}(D-L)$, we have

$$\alpha I - D^{-1}L = D^{-1}(D-L)-(1-\alpha)I = (1-\alpha)\{D^{-1}[\frac{1}{1-\alpha}(D-L)-D]\}. \quad (4.12)$$

Let us choose α satisfying $\frac{1}{1-\alpha} = \delta = (1 + \frac{2\epsilon}{h})\frac{h}{\epsilon}$. Lemma 4.1.2 implies the matrix $\frac{1}{1-\alpha}(D - L) - D$ is an M-matrix. Consequently, $(\frac{1}{1-\alpha}(D - L) - D)^{-1} \geq 0$. Then using the equation (4.12) and $D \geq 0$, it follows that the matrix $\alpha I - D^{-1}L > 0$. Since D is also positive definite, by Lemma 4.1.1, we can conclude that

$$\left\|D^{-1}L\right\| = \rho(D^{-1}L) < \alpha = 1 - \frac{1}{\delta} = 1 - \frac{\epsilon}{h}(\frac{1}{1 + 2\frac{\epsilon}{h}}) < 1 - \frac{\epsilon}{3h}. \tag{4.13}$$

By utilizing (4.13), we estimate $\|G_1\|$ in the following.

Let $x = (x_1, x_2, \cdots, x_N) \in V_h$, where $x_i \in R^N$ and $\sum_{i=1}^{N} \|x_i\|^2 = 1$. It is clear that $\|G_1 x\| < \|G_1 y\|$ for $y = (\|x_1\| x^+, \|x_2\| x^+, \cdots, \|x_N\| x^+)$. Therefore, the eigenvector corresponding to the maximum eigenvalue has the following form:

$$y = (0, \beta_1 x^+, \beta_2 x^+, \cdots, \beta_{N-1} x^+), \text{ where } \sum_{i=1}^{N-1} \beta_i^2 = 1.$$

By direct computing,

$$\begin{aligned} \|G_1 y\| &= \{\sum_{i=1}^{N} \left\| \sum_{k=1}^{i} \beta_k (D^{-1}L)^{i-k} x^+ \right\|^2 \}^{1/2} \\ &= \{\sum_{i=1}^{N} (\sum_{k=1}^{i} \beta_k l^{i-k})^2\}^{1/2}, \text{ where } l = \rho(D^{-1}L). \end{aligned}$$

Since $l < 1$ and $\sum_{i=1}^{N-1} \beta_i^2 = 1$, the inequalities

$$\|G_1\| \leq \{\sum_{i=1}^{N-1} (\sum_{k=1}^{i} \beta_k)^2\}^{1/2} \leq (\sum_{i=1}^{N-1} i)^{1/2} \leq N \tag{4.14}$$

and

$$\begin{aligned} \|G_1\| &\leq \{\sum_{i=1}^{N-1} [\sum_{k=1}^{i} \beta_k^2][\sum_{k=1}^{i} (l^{i-k})^2]\}^{1/2} \leq \{\sum_{i=1}^{N-1} \sum_{k=1}^{i} (l^{i-k})^2\}^{1/2} \\ &\leq \{\frac{1}{1-l^2}(\sum_{i=1}^{N-1} 1 - l^{2i})\}^{1/2} \leq (\frac{N}{1-l})^{1/2} \end{aligned} \tag{4.15}$$

hold. Recall that $l < 1 - \frac{\epsilon}{3h}$ from (4.13). By combining (4.14) and (4.15), we have

$$\|G_1\| \leq \frac{1}{h} \min\{\frac{3h}{\sqrt{\epsilon}}, 1\}. \tag{4.16}$$

Therefore, from (4.10) and (4.16), the following theorem holds.

Theorem 4.1.3 *For $h \gg \epsilon$, the error reduction matrix E^s of the line Gauss-Seidel iterative method for problem 4.1, obtained from SDFEM discretization of the convection-diffusion equation with wind b=(0,1), satisfies the following inequality:*

$$\|E^s\| \leq 3\frac{\epsilon}{h^2} \min\{\frac{3h}{\sqrt{\epsilon}}, 1\}. \tag{4.17}$$

□

Theorem 4.1.3 shows that the error reduction rate is decreased as $\epsilon \to 0$. For a given stopping tolerance, less iterative steps is expected as shown in Table 4.2, where the stopping tolerance $\|r_m\| < 10^{-6} \|r_0\|$, r_0 is the initial residual and r_m is the residual at m-th iterative step, is chosen.

Mesh	$\epsilon = 10^{-2}$	$\epsilon = 10^{-3}$	$\epsilon = 10^{-4}$	$\epsilon = 10^{-5}$
16×16	8	6	4	3
32×32	8	7	5	4
64×64	9	8	6	4

Table 4.2: HGS convergence on rectangular mesh for Problem 2

4.2 Krylov Subspace Method: GMRES

An alternative methodology for solving a linear system, $Au = f$, is based on Krylov subspaces. Iterative methods that take this approach include the well-known conjugate

gradient method (CG) [5], minimal residual method (MINRES) [71]and generalized minimal residual method (GMRES) [87]. Given an $N \times N$ matrix A and a vector $v \in R^N$, the k-dimensional Krylov subspace, $k \leq N$, generated by the matrix A with respect to vector v is defined as

$$K_k(A, v) = span\{v, Av, A^2v, \cdots, A^{k-1}v\}.$$

The above mentioned methods generate iterative solutions u_m on translated Krylov subspace $u_0 + K_m(A, r_0)$, where u_0 is the initial guess and r_0 is the initial residual, such that either the error $e_m = u - u_m$ with respect to the A-norm, defined as $\|e_m\|_A = \sqrt{< Ae_m, e_m >}$, is minimized or the l_2-norm of the residual, $\|Ae_m\|$, is minimized. For a systematic comparison on these methods, we refer to the article in [36] pages 69-118 by Elman. Here, we summarize the GMRES method as follows.

First, an l_2-orthonormal basis $\{v_0, v_1, \cdots, v_{m-1}\}$ of the Krylov subspace $K_m(A, v_0)$ is generated by the Arnoldi process as shown in Algorithm 4.2.1,

First, choose an initial vector v_0 with $\|v_0\| = 1$;

for $j = 0 : m - 1$ **do**

$h_{i,j} =< Av_j, v_i >$ for $i = 1 \cdots j$,

$\hat{v}_{j+1} = Av_j - \sum_{i=1}^{j} h_{i,j} v_i$,

$h_{j+1,i} = \|\hat{v}_{j+1}\|$,

$v_{j+1} = \frac{\hat{v}_{j+1}}{h_{j+1,j}}$.

end for

Algorithm 4.2.1: The Arnoldi process

Let V_m denote the matrix $[v_0, v_1, \cdots, v_{m-1}]$ and $H_m = (h_{i,j})$ where $0 \leq i, j \leq m-1$.

The following relation holds directly from the construction of Algorithm 4.2.1:

$$AV_m = V_{m+1}\bar{H}_m, \tag{4.18}$$

where $\bar{H}_m$ is a $(m+1) \times m$ matrix satisfying $(\bar{H}_k)_{i,j} = H_{i,j}$ for all $0 \leq i, j \leq m-1$ and $\bar{H}_m(m+1,:) = [0, 0, \cdots, 0, h_{m,m-1}]$. The GMRES method computes iterative solutions $u_m = u_0 + z_m \in u_0 + K_m(A, r_0)$ such that

$$\|f - Au_m\| = \|f - A[u_0 + z_m]\| = \min_{z \in K_m(A,r_0)} \|r_0 - Az\|. \tag{4.19}$$

Since V_m is an orthonormal basis of $K_m(A, r_0)$, we have $z = V_m y$ for some $y \in R^m$. Let $v_0 = \frac{1}{\beta} r_0$, where $\beta = \|r_0\|$. From (4.19), the GMRES iterative solution can then be obtained by finding the minimum of the following function

$$J(y) = \min_y \|\beta v_1 - AV_m y\| = \min_y \left\|V_{m+1}[\beta e^1 - \bar{H}_m y]\right\|, \quad \text{by (4.18)}, \tag{4.20}$$

on R^m, where $e^1 = (1, 0, \cdots, 0) \in R^{m+1}$. Moreover, since V_{m+1} is orthonormal, one can rewrite (4.20) as

$$J(y) = \min_y \left\|\beta e^1 - \bar{H}_m y\right\|. \tag{4.21}$$

To further simplify (4.21), let us consider the QR factorization of $\bar{H}_m$. Because $\bar{H}_m$ is an upper Hessenberg matrix, the QR factorization of $\bar{H}_m$ can be easily computed by introducing m plane-rotations [47] page 343. Let

$$\bar{H}_m = Q_m R_m \tag{4.22}$$

be the QR factorization of $\bar{H}_m$, where Q_m is an $(m+1) \times (m+1)$ matrix from plane rotations and satisfies $\|Q_m\| = 1$ and R_m is a $(m+1) \times m$ matrix with zero last row. By (4.22), (4.21) can be further transformed to the following

$$J(y) = \min_y \left\|Q_m[\beta e^1 - \bar{H}_m y]\right\| = \min_y \|g_m - R_m y\|,$$

where $g_m = \beta Q_m e^1$. Now, it becomes clear that one can simply solve the upper triangular part of

$$R_m y = g_m \tag{4.23}$$

to find a vector $y_m \in R^m$ such that $J(y_m) = \min_y \|g_m - R_m y\|$. Let $z_m = V_m y_m$. As a result, we have

$$u_m = u_0 + z_m = u_0 + V_m y,$$

which satisfies (4.19). The complete GMRES algorithm is shown in Algorithm 4.2.2. For GMRES computation cost and some cost-saving implementation issues, we refer to Saad and Schultz [87].

The convergence properties of GMRES are summarized in the following theorem.

Theorem 4.2.1 *Let u_m be the iterative solution generated after m steps of GMRES with residual $r_m = f - Au_m$.*

1. *If A is diagonalizable, $A = X\Lambda X^{-1}$, where $\Lambda = diag[\lambda_i]$ is the diagonal matrix of eigenvalues of A and X is the matrix of eigenvectors, then*

$$\|r_m\| \leq \|X\| \left\|X^{-1}\right\| \min_{\phi_m \in P_m} \max_i |\phi_m(\lambda_i)| \, \|r_0\| , \tag{4.24}$$

   where P_m denotes the set of polynomials P_m of degree m for which $P_0 = 1$.

2. *Let $\hat{A}$ and $\check{A}$ be the symmetric and skew symmetric parts of A, respectively. If $\hat{A}$ is positive definite, then*

$$\|r_m\| \leq \left(1 - \frac{\lambda_{min}(\hat{A})^2}{\lambda_{min}(\hat{A})\lambda_{max}(\hat{A}) + \rho(\check{A})^2}\right) \|r_0\| , \tag{4.25}$$

   where $\rho(\check{A})$ is the spectral radius of $\check{A}$.

Proof: See [35] and [87].

□

Choose u_0, compute $r_0 = f - Au_0$.

Let $\tau = \|r_0\|$, $\beta = \tau$ and $k = 0$.

while $\tau >$ tolerance **do**

Set $v_1 = \frac{r_0}{\beta}$ and $k = k + 1$.

for $j = 1 : m$ **do**

$h_{i,j} =< Av_j, v_i >,$ for $i = 1 \cdots, j,$

$\hat{v}_{j+1} = Av_j - \sum_{i=1}^{j} h_{i,j} v_i,$

$h_{j+1,j} = \|\hat{v}_{j+1}\|$, and

$v_{j+1} = \frac{\hat{v}_{j+1}}{h_{j+1,j}}$

Compute $\tau = \min_{y \in R^j} \left\| \beta e^1 - \bar{H}_j y \right\|$

If ($\tau <$ tolerance) break;

end for

Update $u_k = u_0 + V_j y$,

If ($\tau <$ tolerance) break;

Compute $r_0 = f - Au_k$ and set $\tau = \|r_0\|$

if ($\tau <$ tolerance) **then**

break;

else

set $u_0 = u_k$ and $\beta = \tau$

end if

end while

Algorithm 4.2.2: The GMRES method with restarts after every m steps

Notice that verifying the stopping tolerance of GMRES iteration is essentially free, because the minimum of $J(y)$ is just the m+1 entry of g_m, which is available from the QR factorization of $\bar{H}_m$. Also, the QR factorization requires much less computation time and storage than the Arnoldi process. Therefore, the amount of work and storage of GMRES is mostly determined by the Arnoldi process. Unfortunately, the computation time and storage requirement of the Arnoldi process rises in proportion to $O(m^2)$ and $O(m)$, respectively, as the iteration count m increases. As a result, unless one is fortunate enough to obtain extremely fast convergence, the cost of GMRES will rapidly become prohibitive. To overcome this drawback, a good preconditioner of A or a restarted version of GMRES [87] (see Algorithm 4.2.2) are generally considered in practice.

Let M be a preconditioning matrix of A. In Algorithm 4.2.2, if one replaces A by $M^{-1}A$ and f by $M^{-1}f$, one obtains a preconditioned version of GMRES algorithm. From Theorem 4.2.1, a good preconditioner M is one for which $M^{-1}A$ is close to a normal matrix such that the matrix X of eigenvectors of $M^{-1}A$ satisfies $\|X\|\,\|X^{-1}\| \approx 1$ and the eigenvalues of $M^{-1}A$ are close to 1. For the convection-diffusion equation discretized by the finite volume methods on a uniform mesh, Oosterlee and Washio have shown some multigrd methods with matrix-dependent prolongation operators are good preconditioners and GMRES using multigrid as a preconditioners leads to a faster convergence than the same multigrid methods as solvers [70]. For the convection-diffusion equations discretized by SDFEM on a uniform mesh, the performance of GMRES with preconditioners from different types of Gauss-Seidel methods or from incomplete block factorizations can be seen in [90]. In Section 4.5, we consider preconditioners such as one step of the Gauss-Seidel iteration with

flow-oriented node numbering, one step of the standard V-cycle multigrid with bilinear prolongation operator, and one step of the V-cycle algebraic multigrid. Performance of these methods as GMRES precondiitoners and solvers are compared for the convection-diffusion equations discretized by SDFEM on both a uniform mesh and an adaptive refined mesh.

4.3 Multigrid Method

The efficiency of the multigrid algorithm is achieved from an elegant combination of the smoothing procedure and the coarse grid correction procedure. The smoothing procedure plays the role of reducing highly oscillatory error modes, and the coarse-grid correction is used to correct the remaining smooth error modes. Hackbusch [49] and Braess [16] give the first rigorous proof on the multigrid convergence and identify that the $smoothing\ property$ and the $approximation\ property$ are the cornerstones for the convergence analysis of multigrid methods.

Let A_l and S_l be the matrix from discretization and the error reduction matrix from an iteration method on a mesh with size h_l. Let p and r be the prolongation and restriction operator. The $smoothing\ property$ is defined as

An iteration S_l satisfies the $smoothing\ property$ if there is a function $\eta(v)$ independent of S_l with

$$\|A_l S_l\| \leq \eta(v) \|A_l\| \text{ for all } 0 \leq v < \infty \text{ and } l > 0, \tag{4.26}$$

where $\lim_{v\to\infty} \eta(v) = 0$.

and the $approximation\ property$ is defined as

$$\left\| A_l^{-1} - pA_{l-1}^{-1} r \right\| \leq C_A / \|A_l\| \text{for all } l \geq 0, \tag{4.27}$$

where C_A is independent with l.

In this section, we describe the multigrid algorithm and its recurrence relations. We also give a proof of the V-cycle multigrid convergence for our model problem by establishing some inequalities similar to (4.26) and (4.27).

Assume we are given a nested sequence of finite dimensional subspaces $(V_k, \Im_k)$ for $k = 1, \cdots, J$, where $V_l \subset V_k \subset H^1$ for all $l < k$ and $\Im_k$ is a regular refinement from $\Im_{k-1}$ for all k. Let A_k denote the matrix obtained from the SDFEM discretization of the convection-diffusion equation on V_k. Clearly, for all $w_k, v_k \in V_k$,

$$(A_k w_k, v_k)_k = B_{sd}(w_k, v_k).$$

and the SDFEM solution $u_k \in V_k$ satisfies

$$(A_k u_k, v_k)_k = (I^k f, v_k), \text{ for all } v_k \in V_k,$$

where $(\cdot, \cdot)_k$ denote the L^2 inner product on V_k and $I^k f$ is the nodal interpolation of f on V_K. Since the SDFEM solution u_k is unique on each subspace V_k, the projection operator $P_k : H^1 \to V_k$ is well defined and satisfies $B_{sd}(P_k u, v) = B_{sd}(u, v), \quad \forall\, v \in V_k$. Let the prolongation operator $I_{k-1}^k : V_{k-1} \to V_k$ be the canonical bilinear interpolation. On uniform rectangular meshes, I_{k-1}^k can be represented by the following stencil notation:

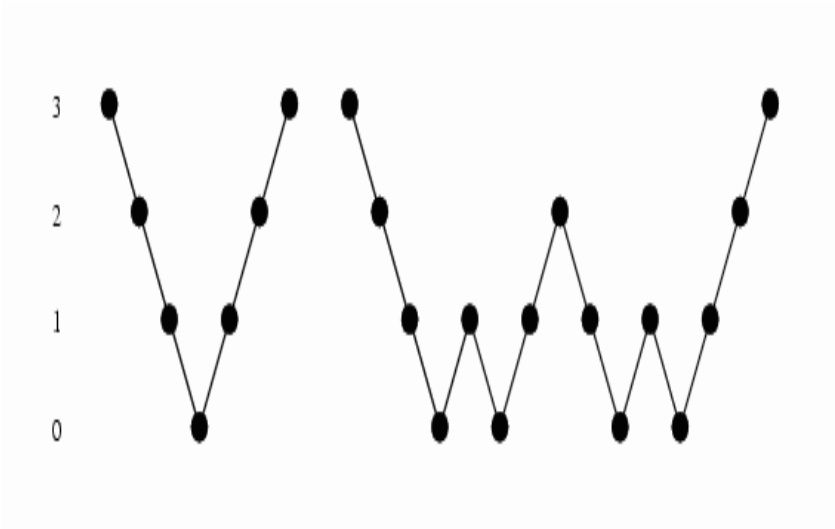
$$I_{k-1}^k = \begin{bmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 1 & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{bmatrix}.$$

Let the restriction operator $I_k^{k-1} : V_k \to V_{k-1}$ be defined by

$$(I_k^{k-1}u, v)_{k-1} = (u, I_{k-1}^k v)_k, \quad \forall\, u \in V_k \text{ and } v \in V_{k-1}.$$

Also, let $M_k^{-1} : V_k \to V_k$ represent a linear smoothing operator.

In order to define the multigrid operator, first, for $k = 1$, let us define $MG_0(w_0, g_0) = A_0^{-1}g_0$. For $k > 1$, let w_k be the initial guess, g_k be the initial righthand side and y_k be the iterative solution after one multigrid step on V_k. By defining the multigrid operator on level k, $MG_k(w_k, g_k)$, in terms of the multigrid operator on level $k-1$, MG_{k-1}, the standard multigrid algorithm can be described in Algorithm 4.3. The usual V-cycle and W-cycle multigrid algorithm are represented by setting $m = 1$ and 2, respectively, in Algorithm 4.3.



(a) V cycle and W cycle

Figure 4.2:

1. set $x_k = w_k$

2. (pre-smoothing) $x_k = w_k + M_k^{-1}(g_k - A_k w_k)$.

3. (restriction) $\bar{g}_k = I_k^{k-1}(g_k - A_k x_k)$.

4. (correction) $q_i = MG_{k-1}(q_{i-1}, \bar{g}_k)$ for $1 \leq i \leq m$, $m = 1$ or 2 and $q_0 = 0$.

5. (prolongation) $\bar{q}_m = I_{k-1}^k q_m$

6. set $x_k = x_k + \bar{q}_m$

7. (post-smoothing) $x_k = x_k + M_k^{-1}(g_k - A_k x_k)$

8. set $y_k = MG_k(w_k, g_k) = x_k$.

Algorithm 4.3.1: Multigrid Algorithm

In the following, we only discuss V-cycle multigrid without post-smoothing. Let the initial error on level k be denoted as $e_k^0 = u_k - w_k$, and the error after one step of multigrid iteration be denoted as $e_k^1 = u_k - y_k$. The error reduction operator for one multigrid iteration can be defined as

$$E_k^{mg}(e_k^0) = e_k^1. \tag{4.28}$$

To derive the recursive relation of the multigrid error reduction operators, let $\tilde{q}$ denote the exact coarse grid correction, i.e.,

$$\tilde{q} = A_{k-1}^{-1} I_k^{k-1}(g_k - A_k x_k) = A_{k-1}^{-1} I_k^{k-1} A_k(u_k - x_k). \tag{4.29}$$

Since, for all $u_k, v_k \in V_k$,

$$\begin{aligned}(I_k^{k-1} A_k u_k, I_k^{k-1} v_k)_{k-1} &= (A_k u_k, I_{k-1}^k I_k^{k-1} v_k)_k = B_{sd}(u_k, I_{k-1}^k I_k^{k-1} v_{k-1}) \\ &= B_{sd}(P_{k-1} u_k, I_k^{k-1} v_k) = (A_{k-1} P_{k-1} u_k, I_k^{k-1} v_k)_{k-1},\end{aligned}$$

we have

$$I_{k-1}^{k}(I_{k}^{k-1}A_k - A_{k-1}P_{k-1}) = 0.$$

Therefore, the relation

$$I_{k}^{k-1}A_k = A_{k-1}P_{k-1}, \tag{4.30}$$

holds on V_k because the bilinear interpolation operator I_{k-1}^{k} has full rank. By plugging (4.30) into (4.29), we have

$$\tilde{q} = P_{k-1}(u_k - x_k). \tag{4.31}$$

Moreover, since the function q in step 2 approximates the function $\tilde{q}$, by (4.28),

$$\tilde{q} - q = E_{k-1}^{mg}\tilde{q}, \tag{4.32}$$

By combining (4.31) and (4.32), the error e_k^1 can be written as

$$\begin{aligned}
e_k^1 &= u_k - y_k = u_k - x_k - I_{k-1}^{k}q = u_k - x_k - I_{k-1}^{k}(\tilde{q} - E_{k-1}^{mg}\tilde{q}) \\
&= u_k - x_k - I_{k-1}^{k}(I - E_{k-1}^{mg})P_{k-1}(u_k - x_k) \\
&= (I - I_{k-1}^{k}P_{k-1} + I_{k-1}^{k}E_{k-1}^{mg}P_{k-1})(u_k - x_k) \\
&= (I - I_{k-1}^{k}P_{k-1} + I_{k-1}^{k}E_{k-1}^{mg}P_{k-1})(I - M_k^{-1}A_k)(u_k - w_k) \quad \text{by (4.8)}, \\
&= (I - I_{k-1}^{k}P_{k-1} + I_{k-1}^{k}E_{k-1}^{mg}P_{k-1})E_k^s e_k^0,
\end{aligned}$$

where $E_k^s = I - M_k^{-1}A_k$ is the error reduction from the smoothing step. Thus, the error reduction operators of multigrid iteration satisfy the following recursive relation,

$$E_k^{mg} = [(I - I_{k-1}^{k}P_{k-1}) + I_{k-1}^{k}E_{k-1}^{mg}P_{k-1}]E_k^s. \tag{4.33}$$

Remark 4.3.1 *For 2-grid multigrid method (k=1),* $E_0^{mg} = 0$. *From (4.30),* $P_{k-1} = A_{k-1}^{-1}I_k^{k-1}A_k$. *In this case, (4.33) can be rewritten as*

$$E_1^{mg} = (I - I_{k-1}^{k}A_{k-1}^{-1}I_k^{k-1}A_k)E_1^s = (A_k^{-1} - I_{k-1}^{k}A_{k-1}^{-1}I_k^{k-1})(A_kE_1^s).$$

Clearly, the $smoothing\ property$ (4.26) and the $approximation\ property$ (4.27) can guarantee that $\|E_1^{mg}\| < 1$ with enough smoothing steps.

From (4.33), we have

$$\begin{aligned} \|E_k^{mg}\| &\leq \left\|(I - I_{k-1}^k P_{k-1})E_k^s\right\| + \left\|I_{k-1}^k E_{k-1}^{mg} P_{k-1} E_k^s\right\| \\ &\leq \left\|(I - I_{k-1}^k P_{k-1})E_k^s\right\| + \left\|E_{k-1}^{mg}\right\| \|P_{k-1}E_k^s\|. \end{aligned}$$

Here, instead of deriving (4.26) and (4.27) for the multigrid convergence, we show that,

$$\left\|(I - I_{k-1}^k P_{k-1})E_k^s\right\| + \|P_{k-1}E_k^s\| < 1, \tag{4.34}$$

hold, when $h_k \gg \sqrt{\epsilon}$ and HGS is employed in the smoothing step, for our model problem. Then, the convergence of the multigrid algorithm 4.3.1 can then be established by mathematical induction.

Theorem 4.3.2 *[Smoothing Property] Let S_k^v be the error reduction operator of v steps of HGS on V_k. The following inequality holds:*

$$\|A_k S_k^v\| \leq \epsilon(1 + 3\frac{\epsilon}{h_k^2}\min\{\frac{3h_k}{\sqrt{\epsilon}}, 1\}) \left\|S_k^{v-1}\right\|. \tag{4.35}$$

Proof: From (4.6) and (4.9), by directly computing, we have

$$\begin{aligned} A_k S_k &= \begin{bmatrix} 0 & \mathrm{U} - \mathrm{U}(\mathrm{D}^{-1}\mathrm{L})\mathrm{D}^{-1}\mathrm{U} & -\mathrm{U}\mathrm{D}^{-1}\mathrm{U} & & \\ 0 & -\mathrm{U}(\mathrm{D}^{-1}\mathrm{L})^2\mathrm{D}^{-1}\mathrm{U} & \ddots & \ddots & \\ \vdots & \vdots & \ddots & -\mathrm{U}(\mathrm{D}^{-1}\mathrm{L})\mathrm{D}^{-1}\mathrm{U} & -\mathrm{U}\mathrm{D}^{-1}\mathrm{U} \\ 0 & -\mathrm{U}(\mathrm{D}^{-1}\mathrm{L})^{\mathrm{n}-1}\mathrm{D}^{-1}\mathrm{U} & \cdots & -\mathrm{U}(\mathrm{D}^{-1}\mathrm{L})^2\mathrm{D}^{-1}\mathrm{U} & -\mathrm{U}(\mathrm{D}^{-1}\mathrm{L})\mathrm{D}^{-1}\mathrm{U} \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix} \\ &= diag[U](T_1 - T_2 G_2), \end{aligned}$$

where

$$T_1 = \begin{bmatrix} 0 & \mathrm{I} & & & \\ 0 & 0 & \ddots & & \\ \vdots & \vdots & \ddots & \mathrm{I} & \\ 0 & 0 & \cdots & 0 & \mathrm{I} \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix} \text{ and } T_2 = \begin{bmatrix} 0 & \mathrm{D}^{-1}\mathrm{L} & \mathrm{I} & & \\ 0 & (\mathrm{D}^{-1}\mathrm{L})^2 & \ddots & \ddots & \\ \vdots & \vdots & \ddots & \mathrm{D}^{-1}\mathrm{L} & \mathrm{I} \\ 0 & (\mathrm{D}^{-1}\mathrm{L})^{\mathrm{n}-1} & \cdots & (\mathrm{D}^{-1}\mathrm{L})^2 & \mathrm{D}^{-1}\mathrm{L} \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix}.$$

By using the same argument in deriving Theorem 4.1.3, one can show $\|T_2 G_2\| \leq 3\frac{\epsilon}{h_k^2} \min\{\frac{3h_k}{\sqrt{\epsilon}}, 1\}$. Therefore,

$$\begin{aligned} \|A_k S_k\| &\leq \epsilon(1 + 3\frac{\epsilon}{h_k^2} \min\{\frac{3h_k}{\sqrt{\epsilon}}, 1\}) \\ &= \epsilon(1 + 3\frac{\epsilon}{h_k^2} \min\{\frac{3h_k}{\sqrt{\epsilon}}, 1\}). \end{aligned}$$

Hence, (4.35) holds.

□

Remark 4.3.3 *From Theorem 4.1.3, we have $\|S_k\| < 1$ for $h_k > \sqrt{3\epsilon}$. Thus, (4.35) implies the smoothing property (4.26) by the fact $\epsilon \leq \|A_k\|$ for all $k > 0$.*

With the help of Theorem 4.3.2, we show (4.34) in the following. Assume v steps of HGS are employed in the smoothing step, i.e. $E_k^s = S_k^v$. For $h_k > \sqrt{3\epsilon}$, we have

$$\begin{aligned} \|P_{k-1} E_k^s\| &= \left\|A_{k-1}^{-1} I_k^{k-1} A_k S_k^v\right\|, \text{ by (4.30)}, \\ &\leq (1 + 3\frac{\epsilon}{h_k^2}) \left\|I_k^{k-1}\right\| \|S_k\|^{v-1}, \text{ by (4.35)}, \\ &\leq (1 + 3\frac{\epsilon}{h_k^2})(3\frac{\epsilon}{h_k^2})^{v-1} \left\|I_k^{k-1}\right\|, \text{ by (4.17)}. \end{aligned} \tag{4.36}$$

Moreover, let $e \in V_k \subset H^1$,

$$\begin{aligned}
\left\|(I - I_{k-1}^k P_{k-1})E_k^s(e)\right\| &\leq ch_k^{-2} \left\|(I - P_{k-1})S^v(e)\right\|_0 = ch_k^{-2}\frac{1}{\sqrt{\epsilon}} \left|\|(I - P_{k-1})S_k^v(e)\|\right| \\
&\leq ch_k^{-2}\sqrt{\frac{h_k}{\epsilon}}|S_k^v(e)|_1, \text{ by the a priori error bound (2.38)} \\
&\leq ch_k^{-2}\sqrt{\frac{h_k}{\epsilon^2}} \left\|A_k S_k^v(e)\right\|_0, \text{ by the regularity estimate (2.8)} \\
&\leq c\sqrt{h_k}(1 + 3\frac{\epsilon}{h_k^2})(3\frac{\epsilon}{h_k^2})^{v-1} \left\|e\right\|,
\end{aligned}$$

where c is a constant. Therefore, we have

$$\left\|(I - I_{k-1}^k P_{k-1})E_k^s\right\| \leq c\sqrt{h_k}(1 + 3\frac{\epsilon}{h_k^2})(3\frac{\epsilon}{h_k^2})^{v-1}. \tag{4.37}$$

From (4.36) and (4.37), the inequality (4.34) holds for $v \geq 2$ and $h_k \gg \sqrt{3\epsilon}$. Now, we can state our multigrid convergence result as follows:

Theorem 4.3.4 *If more then 2 steps of HGS are employed in the smoothing procedure of the multigrid algorithm 4.3.1, then*

$$\|E_J^{mg}\| < 1,$$

for $h_J \gg \sqrt{3\epsilon}$.

Remark 4.3.5 *By direct expansion, equation (4.33) can be rewritten as*

$$E_J^{mg} = \sum_{k=1}^{J-1}(I - P_{J-k})E_{J-k+1}^s(\prod_{l=1}^{k-1} P_{J-k+l}E_{J-k+l+1}^s), \tag{4.38}$$

where $\prod_{l=1}^{0} P_{J-k+l}E_{J-k+l+1}^s = I$. Let $\tilde{h}_i = h_i/\sqrt{6}$ for all i. By plugging (4.36) and (4.37) into (4.38), two steps of HGS smoothing imply

$$\begin{aligned}
\|E_J^{mg}\| &\leq c\sum_{k=1}^{J-1} \tilde{h}_{J-k+1}^{1/2} \left(\frac{\epsilon}{\tilde{h}_{J-k+1}^2}\right) \prod_{l=1}^{k-1} \frac{\epsilon}{\tilde{h}_{J-k+l+1}^2} \\
&< c\sum_{k=1}^{J-1} \tilde{h}_{J-k+1}^{1/2} \left(\frac{\epsilon}{\tilde{h}_{J-k+1}^2}\right) \left(\prod_{l=1}^{k-1} (\frac{1}{2^2})^l\right) \left(\frac{\epsilon}{\tilde{h}_J^2}\right)^{k-1}.
\end{aligned}$$

A sharper estimate of MG convergence,

$$\|E_J^{mg}\| \preceq \frac{\epsilon}{h_J^{3/2}} \preceq \epsilon^{1/4}, \tag{4.39}$$

for $h_J \gg \sqrt{\epsilon}$, can be obtained from directly estimating the righthand side of the above inequality.

Similar to the convergence behavior of HGS, the estimate (4.39) shows that the error reduction rate of MG is also decreased as $\epsilon \to 0$ and less iterative steps are expected for smaller ϵ. For Problem 2, two, three and four levels V-cycle MG are tested on 16×16, 32×32 and 64×64 uniform rectangular meshes, respectively, with 1 step HGS pre-smoothing and post-smoothing. The results are shown in Table 4.3 for various ϵ. The stopping tolerance in our computation is $\|r_m\| < 10^{-6}\|r_0\|$ where r_0 is the initial residual and r_m is the residual at m-th iterative step. By comparing the numerical results in Table 4.2 and Table 4.3, it is evident that MG converges faster than HGS as expected from (4.17) and (4.39).

Mesh	$\epsilon = 10^{-2}$	$\epsilon = 10^{-3}$	$\epsilon = 10^{-4}$	$\epsilon = 10^{-5}$
16×16	4	2	2	1
32×32	4	3	2	1
64×64	5	4	2	2

Table 4.3: MG convergence on uniform rectangular mesh for Problem 2

4.4 Algebraic Multigrid Method

In previous section, we have shown a MG convergence result of the Problem 2 in Section 2.3. With fixed coarse grids and interpolation operators, our MG convergence

result is essentially achieved through the robust smoothing property of HGS in Theorem 4.3.2. Unfortunately, it is hard to show that such a smoothing property still holds if the underlying mesh is unstructured or the flow is more complicated. Since the other major component of MG is the $approximation\ property$, an alternative way to achieve MG convergence is to find better interpolation operators and coarse grids. The basic idea of the algebraic multigrid method (AMG) is to employee an algebraic coarsening process (selecting coarse nodes and defining interpolation) to ensure that the algebraic smooth errors, i.e. the errors which can not be efficiently reduced by relaxation iterations, can be captured by the coarse grid correction. In order to introduce the AMG briefly, we consider two level V-cycle with post-smoothing here. Only the coarsening strategy proposed by Ruge and Stüben [86] is studied here. Similar strategies can be found in Reusken [79] and Wagner, Kinzelbach and Wittum [98].

First, let us introduce some notation. Let V_h and V_H denote the fine grid space and the coarse grid space. Let A_h denote the matrix arising from a discretization method such as Galerkin or SDFEM, and D_h the diagonal matrix of A_h. For $v \in V_h$, let $\|v\|_0 =< D_h v, v >$, $\|v\|_1 =< A_h v, v >$ and $\|v\|_2 =< D_h^{-1} A_h v, A_h v >$. Since the coarsening process does not produce a mesh in geometric sense, the coarse grid operator A_H on V_H is defined by

$$A_H = I_h^H A_h I_H^h, \tag{4.40}$$

where I_H^h is the interpolation operator to be defined by coarsening process and $I_h^H = (I_H^h)^T$. As shown in Remark 4.3.1, the two-grid error reduction operator then can be written as

$$E^{mg} = E^s E^c, \tag{4.41}$$

where E^s is the error reduction operator from the smoothing step and

$$E^c = I - I_H^h A_H^{-1} I_h^H A_h \tag{4.42}$$

is the coarse grid correction.

Multigrid methods using the coarse grid operator (4.40) are called Galerkin-type methods due to their origin in the finite element Galerkin discretization. For symmetric positive define M-matrix $A_h = (a_{i,j})$, it can be shown that the coarse grid correction E^c is an orthogonal projection from V_h to V_H ([101] Chapter 5) with respect to the inner product $< \cdot, \cdot >_1$, i.e. for all $v_h \in V_h$ and $v_H \in V_H$, $< A_h E^c v_h, v_H >= 0$. By using this orthogonal property, the $smoothing\ assumption$,

$$\exists\ \alpha > 0 \text{ such that } \|E^s e_h\|_1^2 \leq \|e_h\|_1^2 - \alpha \|e_h\|_2^2, \text{ for any } e_h \in V_h, \tag{4.43}$$

and the $approximation\ assumption$

$$\min_{e_H} \left\|e_h - I_H^h e_H\right\|_0^2 \leq \beta \|e_h\|_1^2 \text{ with } \beta > 0 \text{ independent with } e_h, \tag{4.44}$$

Ruge and Stüben [86] show the following theorem holds.

Theorem 4.4.1 *Let A_h be a symmetric positive define matrix. Suppose the smooth operator E^s satisfies (4.43) and the interpolation operator I_H^h has full rank and satisfies (4.44). Then $\beta \geq \alpha$ and the convergence rate of the two level V-cycle satisfies $\|E^s E^c\|_1 \leq \sqrt{1 - \frac{\alpha}{\beta}}$.*

Proof: By orthogonality, for any $e_h \in R(E^c) \subset V_h$ and $e_H \in V_H$, we have

$$\|e_h\|_1^2 =< A_h e_h, e_h >=< A_h e_h, e_h - I_H^h e_h > + < A_h e_h, I_H^h >=< A_h e_h, e_h - I_H^h e_h > . \tag{4.45}$$

Since $A_h > 0$ and $D_h^{-1}A > 0$, (4.45) implies

$$\begin{aligned} \|e_h\|_1 &= < A_h^{1/2}(D_h^{-1}A_h)^{1/2}e_h, A_h^{1/2}(D_h^{-1}A_h)^{-1/2}(e_h - I_H^h e_H) > \\ &\leq \left\| A_h^{1/2}(D_h^{-1}A_h)^{1/2}e_h \right\| \left\| A_h^{1/2}(D_h^{-1}A_h)^{-1/2}(e_h - I_H^h e_H) \right\|, \\ &\quad \text{by the Schwarz inequality,} \\ &= \|e_h\|_2 \left\| e_h - I_H^h e_H \right\|_0 . \end{aligned}$$

By (4.44), we have

$$\|e_h\|_1^2 \leq \beta \|e_h\|_2^2 . \tag{4.46}$$

The convergence estimate of the theorem is a direct result of (4.43) and (4.46) as shown in the following:

$$\begin{aligned} \|E^s E^c e_h\|_1^2 &\leq \|E^c e_h\|_1^2 - \alpha \|E^c e_h\|_2^2 \leq (1 - \frac{\alpha}{\beta}) \|E^c e_h\|_1^2 \\ &\leq (1 - \frac{\alpha}{\beta}) \|e_h\|_1^2 . \end{aligned}$$

□

In [86] Theorem 4.2, Ruge and Stüben also show that the usual point Guass-Seidel iteration satisfies the $smoothing\ assumption$ (4.43). In particular, if A_h is also an M-matrix, one has $\alpha = \frac{1}{4}$. Therefore, it remains to construct the interpolation operator such that (4.44) holds for the AMG to converge. The special coarsening strategy in AMG serves this purpose. First, let the set of fine grid points be denoted as F and the set of coarse grid nodes be denoted as C. The neighborhood of the ith node v_i is defined as $N_i = \{j \in F | j \neq i \text{ and } a_{i,j} \neq 0\}$. We consider the interpolation operator

I_H^h which has the following form:

$$\left(\mathrm{I}_{\mathrm{H}}^{\mathrm{h}}\mathrm{e}_{\mathrm{H}}\right)_{\mathrm{i}} = \sum_{v_{\mathrm{j}}\in\mathrm{C}} \mathrm{w}_{\mathrm{i,j}} \left(\mathrm{e}_{\mathrm{H}}\right)_{\mathrm{j}} \tag{4.47}$$

where $w_{i,j} = \delta_{i,j}$ for $v_i \in C$ and $\delta_{i,j}$ denotes the Kronecker symbol. On any given set of coarse grid points C, for any error $e = (e_1, e_2, \cdots e_n) \in V_h$, if the interpolation weights $w_{i,j}$, for all $i, j \in F$, satisfy the following two conditions:

$$\sum_{v_i\in F}\sum_{v_j\in C} a_{i,i}w_{i,j}(e_i - e_j)^2 \leq \frac{\beta}{2}\sum_{i,j} -a_{i,j}(e_i - e_j)^2, \tag{4.48}$$

and

$$\sum_{v_i\in F} a_{i,i}(1 - s_i)e_i^2 \leq \beta \sum_i \left(\sum_j a_{i,j}\right) e_i^2, \tag{4.49}$$

where $s_i = \sum_{v_j\in C} w_{i,j} \leq 1$, then the *approximation assumption* (4.44) holds ([86] Theorem 5.3). For the case that A_h is a M-matrix and diagonal dominant, one can consider the interpolation weights $w_{i,j} = \eta_i|a_{i,j}|$ where $0 \leq \eta_i \leq \frac{1}{\sum_{v_j\in C}|a_{i,j}|}$, which ensures $s_i \leq 1$. Obviously, it is sufficient to require that for every $v_i \in F$ and $v_j \in C_i \subseteq N_i \cap C$

$$0 \leq a_{i,i}w_{i,j} \leq \beta|a_{i,j}| \tag{4.50}$$

and

$$0 \leq a_{i,i}(1 - s_i) \leq \beta \sum_j a_{i,j}, \tag{4.51}$$

for (4.48) and (4.49) to hold.

With the above simple inequalities (4.50) and (4.51), more practical conditions in the coarsening strategies which use β as an input parameter can be derived. For example, given $\beta \geq 1$, if the coarse grid C is selected such that for each $v_i \in F$,

$$\beta(a_{i,i} - \sum_{\substack{v_j\notin C_i \\ j\neq i}} a_{i,j}) = \beta \sum_{v_j\notin C_i} a_{i,j} \geq a_{i,i} \text{ where } C_i \subseteq N_i \cap C. \tag{4.52}$$

and the interpolation weight is defined as $w_{i,j} = \frac{|a_{i,j}|}{\sum_{v_j \notin C_i} a_{i,j}}$, clearly, we have

$$a_{i,i} w_{i,j} = \frac{a_{i,i}}{\sum_{v_j \notin C_i} a_{i,j}} |a_{i,j}| \leq \beta |a_{i,j}|, \text{ by (4.52)}$$

and

$$a_{i,i}(1 - s_i) = a_{i,i} \left(1 - \frac{\sum_{v_j \in C_i} |a_{i,j}|}{\sum_{v_j \notin C_i} a_{i,j}} \right) = a_{i,i} \frac{\sum_j a_{i,j}}{\sum_{v_j \notin C_i} a_{i,j}} \leq \beta \sum_j a_{i,j}.$$

Therefore, by (4.50) and (4.51), the $approximation\ assumption$ holds.

Recall that the algebraic smooth error e^s is more slowly reduced by the smoother E^s, i.e. $\|E^s e^s\|_1 \approx \|e^s\|_1$. By the $smoothing\ assumption$ (4.43), the error e^s has to satisfy $\|e^s\|_2 \ll \|e^s\|_1$ or more explicitly $\sum_i \frac{r_i^2}{a_{i,i}} \ll \sum_i r_i e_i^s$ where $r = (r_1, r_2, \cdots, r_n) = A_h e^s$. Therefore, on average, one can expect $|r_i| \ll a_{i,i}|e_i|$ for all i. Consequently, one obtains a good approximation for e_i,

$$a_{i,i} e_i \approx \sum_{j \in N_i} -a_{i,j} e_j, \tag{4.53}$$

through its neighboring error values e_j, $j \in N_i$. Moreover, since

$$\|e^s\|_1 \;=\; < D_h^{-1/2} A_h e^s, D_h^{1/2} e^s > \leq \left\| D_h^{-1/2} A_h e^s \right\| \left\| D^{1/2} e^s \right\| = \|e^s\|_2 \|e^s\|_0 ,$$

$\|e^s\|_2 \ll \|e^s\|_1$ implies $\|e^s\|_1 \ll \|e^s\|_0$ or, explicitly,

$$\begin{aligned} < A_h e^s, e^s > \;&=\; \frac{1}{2} \sum_{i,j} -a_{i,j} (e_i^s - e_j^s)^2 + \sum_i \left(\sum_j a_{i,j} \right) (e_i^s)^2 \\ &\ll\; < D_h e^s, e^s > = \sum_i a_{i,i} (e_i^s)^2. \end{aligned}$$

For the important case $\sum_{i \neq j} |a_{i,j}| \approx a_{i,i}$, the above inequality means that, on average for each i,

$$\frac{1}{2} \sum_{i \neq j} -a_{i,j} (e_i^s - e_j^s)^2 \ll a_{i,i} (e_i^s)^2. \tag{4.54}$$

In other word, the smooth error generally varies slowly in the so-called strong-connected direction where $\frac{|a_{i,j}|}{a_{i,i}}$ is relative large. The condition (4.52), in turn, suggests that the algebraic coarsening should be done in the direction of strong connections.

Now we can introduce the coarsening strategy proposed by Ruge and Stüben. First, let us introduce some definitions.

Definition 4.4.2 *A node $v_i \in F$ is strongly connected to a node $v_j \in F$ with respect to A_h, denoted as $v_i \to v_j$ if*

$$-a_{i,j} \geq \mu \max_{m \neq i}(-a_{i,m}),$$

where the strong connection parameter μ satisfies $0 \leq \mu \leq 1$. Let N_i^S denote the set of all strongly connected neighbors of v_i, i.e.,

$$N_i^S = \{v_j \in N_i | v_i \to v_j\},$$

and $(N_i^S)^T$ denote the set of nodes which are strongly connected to v_i, i.e.,

$$(N_i^S)^T = \{v_j \in F | v_j \to v_i\} = \{v_j \in F | v_i \in N_j^S\}.$$

The interpolatory nodes C_i in (4.52) are defined as strong C-node neighbors, i.e., $C_i = N_i^S \cap C$. Also, the noninterpolatory nodes D_i are split into strong D_i^S and weak D_i^W noninterpolatory nodes where

$$D_i = N_i \setminus C_i,\ D_i^S = D_i \cap N_i^S \text{ and } D_i^W = D_i \setminus D_i^S.$$

□

Since a large set of coarse grid points C is not practical due to expensive computation cost and memory requirement on the coarse grids, one would like C to be as small as

possible with $C_i \neq \emptyset$ for all i. If condition (4.52) is employed, this means the input parameter β may become very large. As a result, a slow convergence rate of AMG is expected according to Theorem 4.4.1. To get good interpolations and maintain a reasonable complexity of coarse grid the following two criteria are used.

Criterion 4.4.3 *For each node $v_i \in F$, each node $v_j \in N_i^S$ should be either in C or should be strongly connected to at least one node in C_i.*

Criterion 4.4.4 *C should be a maximal subset of all nodes with the property that no two C-nodes are strongly connected to each other.*

The criterion 4.4.3 shall ensure that the interpolation is good enough. The criterion 4.4.4 is taken as a guideline to force the generated coarse grid to significant fewer nodes than the fine grid. In fact, the criterion 4.4.3 arises naturally from the following analysis. First, equation (4.53) can be rewritten as

$$a_{i,i}e_i = \sum_{j\in C_i} -a_{i,j}e_j + \sum_{j\in D_i^S} -a_{i,j}e_j + \sum_{j\in D_i^W} -a_{i,j}e_i - \sum_{j\in D_i^W} -a_{i,j}(e_i - e_j). \quad (4.55)$$

Since we have $\sum_{j\in D_i^W} -a_{i,j}(e_i - e_j) \ll a_{i,i}e_i$ by (4.54), equation (4.55) implies

$$(a_{i,i} + \sum_{j\in D_i^W} -a_{i,j})e_i \approx \sum_{j\in C_i} -a_{i,j}e_j + \sum_{j\in D_i^S} -a_{i,j}e_j. \quad (4.56)$$

Recall that the smooth error varies slowly in the direction of strong connection. As a result, for $j \in D_i^S$, the error value e_j can be replaced by

$$e_j = \frac{\sum_{k\in C_i} |a_{j,k}|e_k}{\sum_{m\in C_i} |a_{j,m}|} \quad (4.57)$$

as long as there exist strong connections $v_j \to v_k$ for some $k \in C_i$. Plugging (4.57) into (4.56), the following formula for computing interpolation weighst in Ruge and Stüben's AMG coarsening algorithm is obtained

$$w_{i,j} = -\frac{1}{\tilde{a}_{i,i}} \left(a_{i,j} + \sum_{k\in D_i^S} \frac{a_{i,k}a_{k,j}}{\sum_{m\in C_i} a_{k,m}} \right), \quad (4.58)$$

where $\tilde{a}_{i,i} = a_{i,i} + \sum_{k \in D_i^W} a_{i,k}$. Based on the criterion 4.4.3 and the criterion 4.4.4, the Ruge and Stüben coarsening strategy consists of two coarsening steps as outlined in Algorithm 4.4.1 and Algorithm 4.4.2. Algorithm 4.4.1 tends to produce grids with very few strong C-node to C-node connection. Algorithm 4.4.2 ensures that the criterion 4.4.3 holds and computes the interpolation weight according to (4.58).

$C = \emptyset; F = \emptyset; U = \{1, 2, \cdots, n\}$;

For $(i = 1 : n)$, $z_i = |(N_i^S)^T|$;

while $(U \neq \emptyset)$ **do**

  get $i \in U$ with maximal z_i then set $C = C \cup \{i\}$ and $U = U \setminus \{i\}$;

  for $(j \in (N_i^S)^T \cap U)$ **do**

    $F = F \cup \{j\}; U = U \setminus \{j\}$;

    For $(k \in N_j^S)$, $z_k = z_k + 1$;

  end for

  For $(j \in N_i^S \cap U)$ $z_j = z_j - 1$;

end while

Algorithm 4.4.1: Preliminary C-point selection

$T = \emptyset$;

while $(F \setminus T \neq \emptyset)\{$

pick $i \in F \setminus T$; set $T = T \cup \{i\}$ and $done = 0$;

$C_i = N_i^S \cap C$; $D_i^S = N_i^S \setminus C_i$; $D_i^W = N_i \setminus N_i^S$; $\tilde{C}_i = \emptyset$;

while $(done == 0)\{$

$d_i = a_{i,i} + \sum_{k \in D_i^W} a_{i,k}$; $d_j = a_{i,j} \forall j \in C_i$

for $(k \in D_i^S)\{$

if $(N_k^S \cap C_i \neq \emptyset)$ $d_j = d_j + \frac{a_{i,k} a_{k,j}}{\sum_{m \in C_i} a_{k,m}} \forall j \in C_i$;

else $\{$

if $(\tilde{C}_i \neq \emptyset)\{C = C \cup \{i\}$; $F = F \setminus \{i\}$; break; $\}$

else $\{$

$\tilde{C}_i = \{k\}$; $C_i = C_i \cup \{k\}$; $D_i^S = D_i^S \setminus \{k\}$;

$done = 0$; break;

$\}$

$\}$

$\}$

$\}$

if $(i \in F)$ $\{C = C \cup \tilde{C}_i$; $F = F \setminus \tilde{C}_i$; $w_{i,j} = -d_j/d_i \forall j \in C_i\}$

$\}$

Algorithm 4.4.2: Final C-point selection and definition of interpolation weights

Although most of the theoretical analysis of AMG is limited to M-matrices, numerical studies in [92] show fast convergence of AMG even if the matrix A_h is not symmetric, such as in the case of finite difference discretization of the convection-diffusion equation. Numerical studies of the AMG convergence for the matrix from

SDFEM discretization of the convection-diffusion equation will be presented below in Section 4.5. Here, we show coarse grids from the AMG coarsening on two problems. The first is Problem 2 in Section 2.3. The second problem is the convection-diffusion problem with closed characteristic as follows:

Problem 4: Flows with closed characteristics

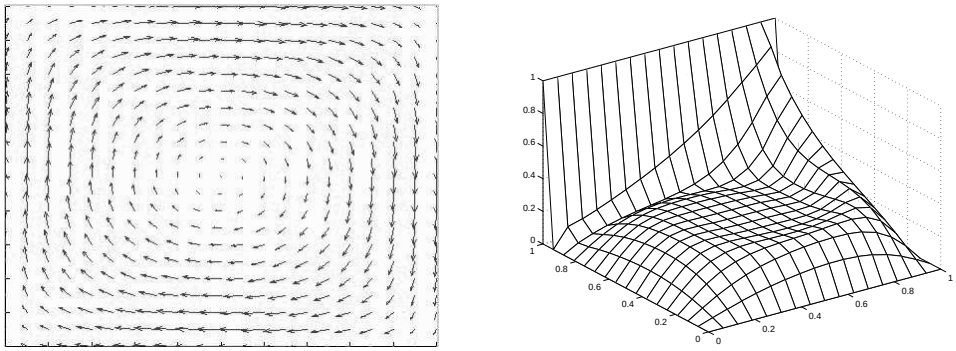
$$-\epsilon \cdot \Delta u + (b_1, b_2) \cdot \nabla u = 0, \text{ with}$$

$$(b_1, b_2) = (2(2y-1)(1-(2x-1)^2), -2(2x-1)(1-(2y-1)^2)) \text{ and,}$$

$$u|_{\partial\Omega} = \begin{cases} 1 & \text{if } y = 1, \\ 0 & \text{otherwise,} \end{cases}$$

where $\Omega = [0,1] \times [0,1]$.

A sample solution is shown in Figure 4.3.

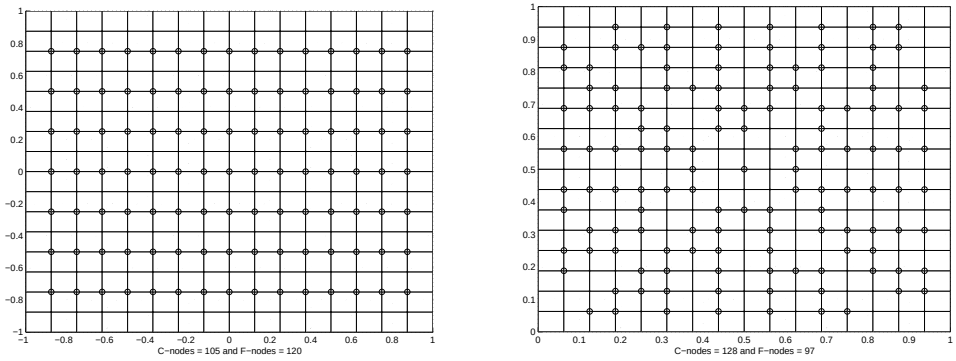


(a) Flow field (b_1, b_2) (b) 3D representation of solution

Figure 4.3: Flow field and solution of Problem 4

In both problems, the diffusion parameter ϵ is 10^{-2} and the input parameter μ, used to define the strong connection in Definition 4.4.2, is set to 0.25. Figure 4.4 (a) shows that the coarse grid obtained from AMG coarsening is the same as the coarse grid

obtained from semi-coarsening the fine grid in the y-direction for Problem 2. Figure 4.4 (b) shows that the coarse grid from AMG coarsening tends to be symmetric with respect to the center of the domain Ω in Problem 4. These results suggest that AMG coarsening strategies coarsen the fine grid in a direction that follows the flow field (b_1, b_2). In fact, with the help of standard matrix-dependent interpolation I_H^h and restriction I_h^H defined in [78], MG convergence on a mesh obtained from semi-coarsening is proved by Reusken [80] for Problem 2. Naturally, one may conjecture that β in the $approximation\ assumption$ (4.44) can be small in AMG and a faster MG convergence rate can be obtained. Our numerical studies in Section 4.5 give an answer to this question.



(a) Problem 2 (b) Problem 4

Figure 4.4: Coarse grids from AMG coarsening

4.5 Numerical Comparisons of GMRES, MG and AMG

In this section, we compare the performance of different linear solvers for the discrete convection-diffusion equation, including MG, AMG, GMRES and preconditioned GMRES. Two test problems, Problem 2 and Problem 4, are discretized on both an uniform 32×32 triangular mesh and an adaptively refined mesh for $\epsilon = 10^{-2}, 10^{-3}$

and 10^{-4}. The adaptively refined mesh is generated by refining an initial 8×8 uniform mesh four times based on the KS error indicator and the maximum marking strategy. The threshold value θ in the maximum marking strategy is chosen such that elements in the layer regions can be refined for both problems. For Problem 2, $\theta = 0.1$, 0.01 and 0.001 for $\epsilon = 10^{-2}, 10^{-3}$ and 10^{-4}, respectively. The adaptive meshes and solutions of Problem 2 are shown in Figure 4.5. For Problem 4, $\theta = 0.1$ for all ϵ. The adaptive meshes and solutions of Problem 4 are shown in Figure 4.6.

In Section 4.1 and Section 4.2, it has been shown that the horizontal line Gauss-Seidel method (HGS) converges and MG converges with HGS smoother, when $h \gg \epsilon^{1/2}$, for Problem 2. On uniform meshes, we would also like to use one step of HGS as a smoother and a preconditioner in Problem 2. For Problem 4, because the flow field has closed characteristics, our strategy is to use four Gauss-Seidel sweeps,

$$\text{HGS} \rightarrow \text{VGS} \rightarrow \text{backward HGS} \rightarrow \text{backward VGS},$$

as a smoother of MG and AMG, and preconditioner of GMRES. We call the above four sweep Gauss-Seidel method the alternating direction Gauss-Seidel method (ADGS). On unstructured meshes, there is no natural horizontal line or vertical line. However, one can order the nodes by using the y-coordinate as the primary key and the x-coordinate as the secondary key to obtain a node ordering similar to the node ordering in HGS. Here, we call the point Gauss-Seidel method, associated with this node ordering, HGS. Similarly, if one orders the nodes by using x-coordinate as primary key and y-coordinate as secondary key, one obtain a node ordering similar to the node ordering in VGS. We call the point Gauss-Seidel method, associated with such ordering, VGS. By reversing the node numbering, the backward HGS and backward VGS on unstructured grids can be defined from HGS and VGS respectively. Again, on the un-

structured meshes, one step of HGS is used both as a smoother of MG and AMG, and preconditioner of GMRES for Problem 2. For Problem 4, ADGS is used as a smoother of MG and AMG, and preconditioner for GMRES. In addition, MG and AMG, with the above Gauss-Seidel smoothers, are also used as preconditioners of GMRES for both problems. In the following, GMRES with MG preconditioner is denoted as GMRES-MG and GMRES with AMG preconditioner is denoted as GMRES-AMG.

To compare the performance of MG and AMG as solvers or preconditioner of GMRES, four levels of V-cycle are performed in our computation. In MG, the coarse grids are either 4×4, 8×8, 16×16 uniform meshes, or meshes generated during the refinement process. In AMG, the coarse grids are generated from AMG coarsening of the finest adaptive mesh. The comparison of coarse grid complexity of MG and AMG on both uniform mesh and unstructured mesh is shown in Table 4.5 and Table 4.7, respectively. Our results show that, with heuristic strong connection parameter $\mu = 0.25$, the number of coarse grid points generated from AMG coarsening process is greater than the number of grid points on the adaptive mesh at the same mesh level, if the the 32×32 uniform mesh is the finest mesh. However, fewer coarse grid points are generated by AMG coarsening compared to the number of coarse grid points on the meshes from adaptive refinement. As a result, we do not expect AMG and GMRES-AMG to perform well if the problems are solved on the adaptive meshes.

In Table 4.4 and Table 4.6, one can see that AMG and GMRES-AMG converge faster than MG and GMRES-MG, respectively, for Problem 2 especially on the uniform mesh. On the other hand, MG and GMRES-MG outperform AMG and GMRES-AMG for Problem 4 on the adaptive mesh. Both MG and AMG produce better pre-

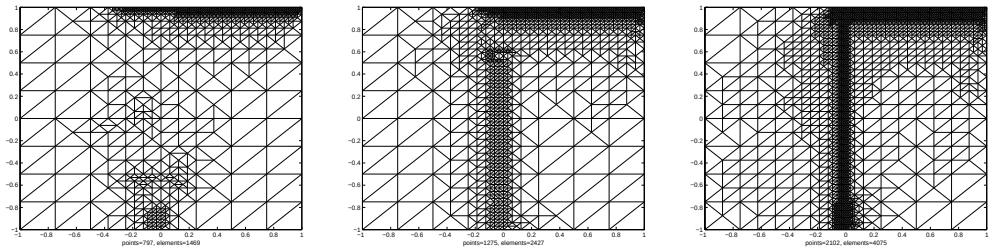
conditioning for GMRES than Gauss-Seidel methods. If the problems are solved on both uniform meshes and adaptive refined meshes, GMRES-MG and GMRES-AMG are the best choices among these solvers. However, one should be reminded that AMG involves more preprocessing time and may also need a carefully chosen strong connection parameter. On the other hand, these problems are usually solved on an mesh similar to the adaptive refined mesh to obtain more accurate solutions in those regions. Under this circumstance, our numerical studies suggest that MG or GMRES with MG preconditioner are the best choices in solving the test problems. Overall, GMRES-MG seems to be a good choice of linear solver for the convection-diffusion problems when solution accuracy, numerical stability (on both uniform and adaptive meshes) and computation cost are our concerns.

In the following tests, the stopping tolerance for iterative methods is set to be

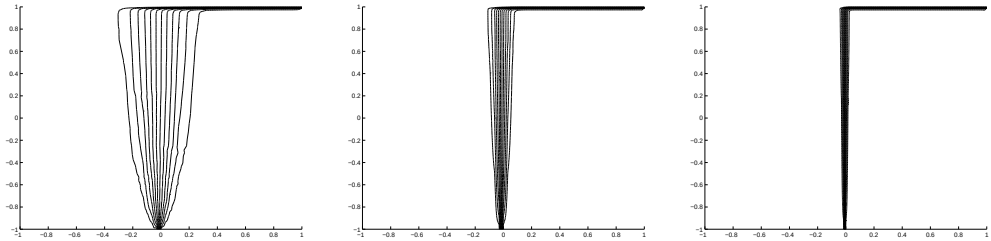
$$\|r_m\| \leq 10^{-6} \|r_0\|,$$

where r_0 is the initial residual and r_m is the residual at m-th iteration. Also, the notation " $-$ " represents that the number of iterations is greater than 200.

Numerical results for Problem 2:



(a) Mesh: $\epsilon = 10^{-2}$ (b) Mesh: $\epsilon = 10^{-3}$ (c) Mesh: $\epsilon = 10^{-4}$



(d) Solution: $\epsilon = 10^{-2}$ (e) Solution: $\epsilon = 10^{-3}$ (f) Solution: $\epsilon = 10^{-4}$

Figure 4.5: Solutions and adaptive meshes for various ϵ

ϵ	10^{-2}	10^{-3}	10^{-4}
GMRES	58	75	94
MG	13	27	51
AMG	7	7	9
GMRES-GS	26	32	43
GMRES-MG	14	20	28
GMRES-AMG	8	9	12

(a) Iterative steps on uniform mesh

ϵ	10^{-2}	10^{-3}	10^{-4}
GMRES	65	146	-
MG	4	22	59
AMG	4	8	14
GMRES-GS	11	31	59
GMRES-MG	5	16	36
GMRES-AMG	4	8	14

(b) Iterative steps on adaptive mesh

Table 4.4: Iteration steps for various iteration methods

	MG	AMG		
ϵ	10^{-2},10^{-3},10^{-4}	10^{-2}	10^{-3}	10^{-4}
level=1	1089	1089	1089	1089
level=2	289	480	479	479
level=3	81	307	331	231
level=4	25	157	108	108

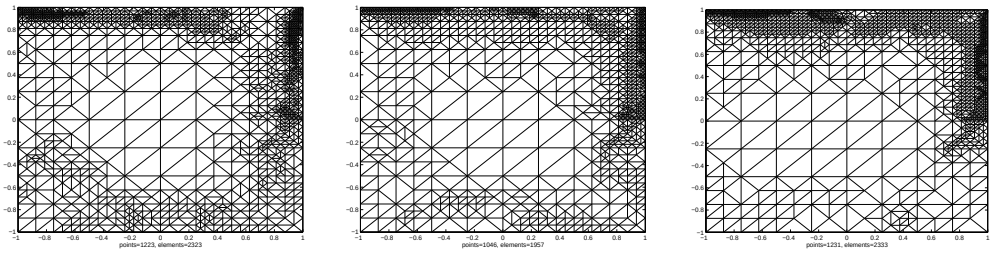
(a) Number of points in coarse grids from uniform mesh

	MG			AMG		
ϵ	10^{-2}	10^{-3}	10^{-4}	10^{-2}	10^{-3}	10^{-4}
level=1	797	1275	2102	797	1275	2102
level=2	410	649	1047	348	580	996
level=3	215	320	528	159	304	523
level=4	122	176	239	88	166	281

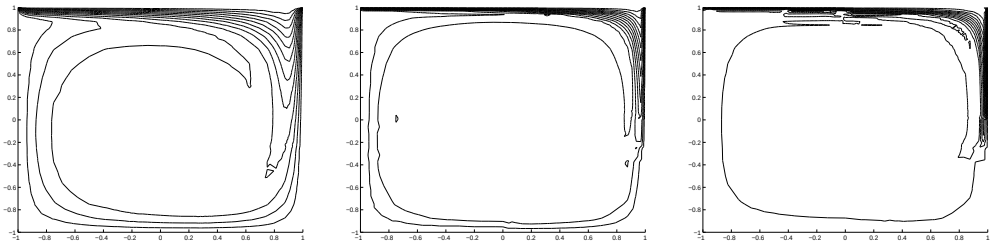
(b) Number of points in coarse grids from adaptive mesh

Table 4.5: Comparison on coarse grids from MG and AMG

Numerical results for Problem 4:



(a) Mesh: $\epsilon = 10^{-2}$ (b) Mesh: $\epsilon = 10^{-3}$ (c) Mesh: $\epsilon = 10^{-4}$



(d) Solution: $\epsilon = 10^{-2}$ (e) Solution: $\epsilon = 10^{-3}$ (f) Solution: $\epsilon = 10^{-4}$

Figure 4.6: Solutions and adaptive meshes for varies ϵ

ϵ	10^{-2}	10^{-3}	10^{-4}
GMRES	-	-	-
MG	26	187	-
AMG	29	-	-
GMRES-GS	37	59	77
GMRES-MG	13	32	45
GMRES-AMG	11	24	33

(a) Iterative steps on uniform mesh

ϵ	10^{-2}	10^{-3}	10^{-4}
GMRES	-	-	-
MG	8	19	13
AMG	23	142	-
GMRES-GS	34	42	40
GMRES-MG	8	12	14
GMRES-AMG	10	16	16

(b) Iterative steps on adaptive mesh

Table 4.6: Iteration steps for various iteration methods

	MG	AMG		
ϵ	10^{-2},10^{-3},10^{-4}	10^{-2}	10^{-3}	10^{-4}
level=1	1089	1089	1089	1089
level=2	289	502	500	498
level=3	81	289	288	280
level=4	25	168	146	151

(a) Number of points in coarse grids from uniform mesh

	MG			AMG		
ϵ	10^{-2}	10^{-3}	10^{-4}	10^{-2}	10^{-3}	10^{-4}
level=1	1223	1046	1231	1223	1046	1231
level=2	629	645	824	573	461	565
level=3	315	381	390	311	254	323
level=4	161	203	202	171	127	179

(b) Number of points in coarse grids from adaptive mesh

Table 4.7: Comparison on coarse grids from MG and AMG

Chapter 5

Stopping Criteria for Iterative Linear Solvers

Let $\eta_{h,T}$ denote an error indicator for a finite element solution of the convection-diffusion equation discussed in Chapter 2. In this chapter, we seek some stopping criteria for the iterative solutions such that meshes generated from $\eta^n_{h,T}$ will not be too different with the mesh generated from $\eta_{h,T}$, where $\eta_{h,T}$ is the error indicator computed from the SD-solution u_h on each element $T \in \Im_h$ and $\eta^n_{h,T}$ is the error indicator computed from the solution u^n_h obtained after n steps of an iterative solution algorithm.

It is natural to require large enough n such that

$$|||u_h - u^n_h|||^2 \preceq c_0 \sum_{T\in\Im_h} {\eta_{h,T}}^2, \tag{5.1}$$

where constant $c_0 > 0$ is small. In other word, $|||u - u^n_h|||$ is still bounded by the same a posteriori upper bound. On the other hand, it is also desirable to have sufficient iteration steps so that $\eta^n_{h,T}$ is close to $\eta_{h,T}$, i.e. there exist constants $c_1, c_2 \approx 1$ such that

$$c_1\eta_{h,T} < \eta^n_{h,T} < c_2\eta_{h,T}. \tag{5.2}$$

As a result, $\eta^n_{h,T}$ can still produce similar mesh refinement as $\eta_{h,T}$ for any refinement

strategy. In the following, we first assume

$$|||u_h - u_h^n|||_{\omega_T} \preceq c_{h,\omega_T}\eta_{h,T} \tag{5.3}$$

on each mesh level, where c_{h,ω_T} are constants to be determined. In Lemma 5.1.1 and 5.2.1 , we show what order of magnitude of c_{h,ω_T}, in terms of h and ϵ, is needed for (5.2) to hold. Obviously, for developing computable stopping criteria, (5.3) is not enough because $\eta_{h,T}$ is still an unknown quantity. It will be more satisfactory if one can replace $\eta_{h,T}$ by the error indicator η_{h_p,T_p} where T_p is the parent element of element T and h_p is the diameter of T_p. In other word, if there exists a constant $\alpha \gg 0$ independent with mesh size h such that

$$\alpha < \min_{T\in\Im_h} \frac{\eta_{h,T}}{\eta_{h_p,T_p}},$$

then (5.3) can be replaced by the following inequality

$$|||u_h - u_h^n|||_{\omega_T} \preceq \alpha c_{h,\omega_T}\eta_{h_p,T_p},$$

Then, one can have computable stopping criteria, as shown in Theorem 5.1.4 and 5.2.4, that imply (5.1) and (5.2). Unfortunately, although the global error reduction rate has been studied by Dörfer and Nochetto [32] [68] and papers cited therein for some self-adjoin problems, there is still no known estimation of the local reduction rate for the error estimators.

Nevertheless, the stopping criteria in Theorem 5.1.5 and 5.2.5 are given to ensure that (5.1) holds for the iterative solutions satisfying these criteria under the assumption that the adaptive refinement process converges at a rate slower than h^σ, $\sigma \leq 2$. This assumption is generally true since the underlying weak solutions are generally not in $H^2(\Omega)$. In addition, in Theorem 5.1.6 and Theorem 5.2.6, we show that when the

maximum marking strategy is employed for the mesh refinement, (5.2) holds in the marked regions for the iterative solutions satisfying our stopping criteria and severe over-refinement will not occur, under the assumption

$$\frac{\max_{T\in\Im_h}\eta_{h,T}}{\max_{T_p\in\Im_{h_p}}\eta_{h_p,T_p}}\gg 0,$$

is a constant independent with mesh size h. Our numerical studies support this assumption. Furthermore, we also derive computable stopping criteria in Theorem 5.1.7 and 5.2.7 and show that both (5.1) and (5.2) hold without any assumption on $\eta_{h,T}$ when the iterative solutions satisfying these stopping criteria and the marking strategy in [68] is employed. In section 5.3, stopping criteria in Theorem 5.1.6 and 5.2.6 are used in our numerical tests. Our numerical results show that almost identical meshes are produced by $\eta^n_{h,T}$ and $\eta_{h,T}$. For simplicity, only Dirichlet boundary condition is considered and the interpolation errors from data and boundary conditions are high order terms that can be ignored. Moreover, only one level of mesh refinement is considered in our analysis.

5.1 Stopping Criteria Associated with Residual-Type a Posteriori Error Estimation

Recall that for any function u in the finite element space V_h of $\Im_h$, Verfürth's error indicator is

$$\begin{aligned} {\eta_T}^2 &= \alpha_T^2 \left\| f_h - \epsilon\triangle u - b\cdot\nabla u - c\cdot u\right\|_{0;T}^2 \\ &+ \frac{1}{2}\sum_{E\in\partial T\cap\Omega} \epsilon^{-1/2}\alpha_E \left\|[\epsilon\partial_{n_E}u]_E\right\|_{0;E}^2 \\ &+ \sum_{E\in\partial T\cap\Gamma_N} \epsilon^{-1/2}\alpha_E \left\|g_h - \epsilon\partial_{n_E}u\right\|_{0;E}^2 \end{aligned}$$

where $\alpha_T = \min\{h\epsilon^{-1/2}, 1\}, T\in\Im_h$ and $\alpha_E = \min\{|E|\epsilon^{-1/2}, 1\}, E\in\partial T\cap\Omega$.

Let u_1, u_2 be any two functions in V_h. The following lemma gives a measure on how close u_2 has to be with u_1 so that the associated error indicators will have the same profile.

Lemma 5.1.1 *Let η_T^1 and η_T^2 be the error indicator of u_1 and u_2 on element T respectively. If*

$$|||u_1 - u_2|||_{\omega_T} \le \frac{1}{2\sqrt{2}\left\|b\right\|_\infty} c_{h,w_T}\eta_T^1, \text{ where } c_{h,\omega_T} = \epsilon^{1/2}\max\{\tfrac{\sqrt{\epsilon}}{h}, 1\} \tag{5.4}$$

then

$$\frac{1}{2}\eta_T^1 \le \eta_T^2 \le \frac{3}{2}\eta_T^1. \tag{5.5}$$

Proof: From the definition of η_T^1 and η_T^2,

$$\begin{aligned} &|\eta_T^1 - \eta_T^2| \le \\ &\underbrace{\{[\alpha_T(\left\|f_h - \epsilon\triangle u_1 - b\cdot\nabla u_1 - cu_1\right\|_{0,T} - \left\|f_h - \epsilon\triangle u_2 - b\cdot\nabla u_2 - cu_2\right\|_{0,T})]^2}_{I} \\ &\underbrace{+\frac{1}{2}\epsilon^{-1/2}[\sum_{E\in\partial T}\alpha_E(\left\|[\epsilon\partial_{n_E}u_1]\right\|_{0,E} - \left\|[\epsilon\partial_{n_E}u_2]\right\|_{0,E})^2]\}^{1/2}}_{II}. \end{aligned}$$

Now let us estimate (I) and (II):

$$\begin{aligned}
(I) &\leq \alpha_T^2 \left\| b \cdot \nabla(u_2 - u_1) + c(u_2 - u_1) \right\|_{0;T}^2 \\
&\leq \alpha_T^2 (\left\| b \cdot \nabla(u_2 - u_1) \right\|_{0;T} + \left\| c(u_2 - u_1) \right\|_{0;T})^2 \\
&\leq \alpha_T^2 (\left\| b \right\|_{\infty;T} \left\| \nabla(u_2 - u_1) \right\|_{0;T} + \left\| c \right\|_{0;T} \left\| u_2 - u_1 \right\|_{0;T})^2 \\
&= \alpha_T^2 (\frac{\left\| b \right\|_{\infty;T}}{\sqrt{\epsilon}} \sqrt{\epsilon} \left\| \nabla(u_2 - u_1) \right\|_{0;T} + \frac{\left\| c \right\|_{0;T}}{\sqrt{d_0}} \sqrt{d_0} \left\| u_2 - u_1 \right\|_{0;T})^2 \\
&\leq \alpha_T^2 (\frac{\left\| b \right\|_{\infty;T}^2}{\epsilon} + \frac{\left\| c \right\|_{0;T}^2}{d_0}) (\epsilon \left\| \nabla(u_2 - u_1) \right\|_{0;T}^2 + d_0 \left\| u_2 - u_1 \right\|_{0;T}^2) \\
&= C_I \left| \left\| u_2 - u_1 \right\| \right|_{0;T}^2 ,
\end{aligned}$$

where $C_I = \alpha_T^2 (\frac{\|b\|_{\infty;T}^2}{\epsilon} + \frac{\|c\|_{0;T}^2}{d_0})$.

By applying the trace inequality (Lemma 3.1 [97]),

$$\begin{aligned}
(II) &\leq \frac{1}{2} \epsilon^{-1/2} \sum_{E \in \partial T} \alpha_E \left\| \epsilon [\partial_{n_E} (u_2 - u_1)]_E \right\|_{0;E}^2 \\
&\leq \frac{1}{2} \epsilon^{-1/2} \sum_{E \in \partial T} \alpha_E \{ h_T^{-1/2} \left\| \epsilon [\partial_{n_E} (u_2 - u_1)]_E \right\|_{0;T} \\
&+ \left\| \epsilon [\partial_{n_E} (u_2 - u_1)]_E \right\|_{0;T}^{1/2} \left\| \nabla(\epsilon [\partial_{n_E} (u_2 - u_1)]_E) \right\|_{0;T}^{1/2} \}^2 \\
&\leq \frac{1}{2} \epsilon^{-1/2} \sum_{E \in \partial T} \alpha_E \{ 2 h_T^{-1/2} \left\| \epsilon [\partial_{n_E} (u_2 - u_1)]_E \right\|_{0;T} \}^2 , \\
&\qquad \text{by inverse estimate, Lemma 4.5.3 [21],} \\
&\leq 6 \epsilon^{1/2} \max_{E \in \partial T} \{\alpha_E\} h_T^{-1} (\epsilon \left\| \nabla(u_2 - u_1) \right\|_{0;\omega_T}^2) \\
&= C_{II} \left| \left\| u_2 - u_1 \right\| \right|_{0;\omega_T}^2 ,
\end{aligned}$$

where $C_{II} = 6\epsilon^{1/2} \max_{E \in \partial T} \{\alpha_E\} h_T^{-1}$.

Clearly, when $h < \sqrt{\epsilon}$, $\alpha_T \approx \alpha_E \approx \frac{h}{\sqrt{\epsilon}}$. we have $C_I \approx (\frac{h}{\epsilon})^2$, and $C_{II} \approx 1$. Also, when $h > \sqrt{\epsilon}$, $\alpha_T \approx \alpha_E \approx 1$, we have $C_I \approx \frac{1}{\epsilon}$, and $C_{II} \approx \frac{\sqrt{\epsilon}}{h} < 1$. Therefore, C_I is always greater than C_{II}. As a result, for convection-dominated flows, we have

$$|\eta_T^2 - \eta_T^1| \leq \sqrt{2 C_I} \left| \left\| u_2 - u_1 \right\| \right|_{\omega_T} . \tag{5.6}$$

From (5.4), this implies

$$\eta_T^2 \leq \eta_T^1 + \sqrt{2C_I} C_{h,\omega_T} \eta_T^1 = (1 + \sqrt{2C_I} C_{h,\omega_T}) \eta_T^1. \tag{5.7}$$

$$\eta_T^2 \geq \eta_T^1 - \sqrt{2C_I} C_{h,\omega_T} \eta_T^1 = (1 - \sqrt{2C_I} C_{h,\omega_T}) \eta_T^1. \tag{5.8}$$

Let's choose $C_{h,\omega_T} = \frac{1}{2\sqrt{2}\|b\|_\infty} \epsilon^{1/2} \max\{\frac{\sqrt{\epsilon}}{h}, 1\} \preceq \frac{1}{2}(\sqrt{2C_I})^{-1}$. it is then clear that

$$\frac{1}{2}\eta_T^1 \leq \eta_T^2 \leq \frac{3}{2}\eta_T^1.$$

□

Clearly, if one replaces η_T^1 by $\eta_{h,T}$ and η_T^2 by $\eta_{h,T}^n$, the following corollary holds.

Corollary 5.1.2 *Let u_h be the finite element solution and u_h^n be the iterative solution. If the number of iterations is large enough that*

$$|||u_h - u_h^n|||_{\omega_T} \leq \frac{1}{2\sqrt{2}\,\|b\|_\infty} c_{h,\omega_T} \eta_{h,T}, \tag{5.9}$$

where $c_{h,\omega_T} = \epsilon^{1/2} \max\{\frac{\sqrt{\epsilon}}{h}, 1\}$, then

$$\frac{1}{2}\eta_{h,T} \leq \eta_{h,T}^n \leq \frac{3}{2}\eta_{h,T}. \tag{5.10}$$

Moreover, for some marking strategies such as the marking strategy in [32], one may not particularly require the values of error indicators from the exact solution and iterative solution to be similar on each element but only requires that the L^2 norm of the error indicator from exact solution is close to the L^2 norm of the error indicator computed from iterative solution in a set of elements. The result in Corollary 5.1.2 can be easily generalized for a set of elements.

Corollary 5.1.3 *Let u_h be the finite element solution and u_h^n be the iterative solution. If the number of iterations is large enough such that*

$$(\sum_{T \in \Im_h^*} |||u_h - u_h^n|||_{\omega_T}^2)^{1/2} \leq \frac{1}{2\sqrt{2}\,\|b\|_\infty} \min_{T \in \Im_H^*} \{c_{h,\omega_T}\} (\sum_{T \in \Im_h^*} \eta_{h,T}^2)^{1/2}, \tag{5.11}$$

where $\Im_h^* \subset \Im_h$ *and* $c_{h,\omega_T} = \epsilon^{1/2} \max\{\frac{\sqrt{\epsilon}}{h}, 1\}$*, then*

$$\frac{1}{2}(\sum_{T\in\Im_h^*} \eta_{h,T}^2)^{1/2} \leq (\sum_{T\in\Im_h^*} {\eta_{h,T}^n}^2)^{1/2} \leq \frac{3}{2}(\sum_{T\in\Im_h^*} \eta_{h,T}^2)^{1/2}. \tag{5.12}$$

Proof: By the triangle inequality and (5.6),

$$\begin{aligned} |(\sum_{T\in\Im_h^*} \eta_{h,T}^2)^{1/2} - (\sum_{T\in\Im_h^*} {\eta_{h,T}^n}^2)^{1/2}| &\leq (\sum_{T\in\Im_h^*} |\eta_{h,T} - \eta_{h,T}^n|^2)^{1/2} \\ &\leq (\sum_{T\in\Im_h^*} 2C_I \, |||u_h - u_h^n|||_{\omega_T}^2)^{1/2}. \end{aligned}$$

The result follows from the same arguments used to establish Lemma 5.1.1.

□

Now, let γ_h^n be the residual of the solution obtained after n step of an iterative solver. Since

$$\begin{aligned} \|\gamma_h^n\|_T &= \langle f_h - A_h u_h^n, f_h - A_h u_h^n\rangle_T^{1/2} \\ &= \langle A_h(u_h - u_h^n), A_h(u_h - u_h^n)\rangle_T^{1/2} \\ &\geq \min\{\Lambda(A_h A_h^*)\}^{1/2} \|u_h - u_h^n\|_T \\ &\geq \epsilon^{1/2} h^{-1} \|u_h - u_h^n\|_{0;\omega_T} \\ &\geq \epsilon^{1/2} h^{-1} (\frac{\epsilon}{h^2} + d_0)^{-1/2} \, |||u_h - u_h^n|||_{\omega_T}, \end{aligned}$$

we have,

$$|||u_h - u_h^n|||_{\omega_T} \leq \kappa \|\gamma_h^n\|_T, \text{ where } \kappa = \max\{\tfrac{h}{\sqrt{\epsilon}}, 1\}. \tag{5.13}$$

Similarly, by the same argument,

$$|||u_h - u_h^n|||_{\Omega} \leq \kappa \|\gamma_h^n\|_{0,\Omega}, \text{ where } \kappa = \max\{\tfrac{h_{max}}{\sqrt{\epsilon}}, 1\}. \tag{5.14}$$

Clearly, if n is large enough such that

$$\|\gamma_h^n\|_T \leq \frac{c_{h,\omega_T}}{\kappa}(\min_{T\in\Im_h} \frac{\eta_{h,T}}{\eta_{h_p,T_p}})\eta_{h_p,T_p},$$

the inequality,

$$|||u_h - u_h^n|||_{\omega_T} \quad \leq \quad c_{h,\omega_T}\eta_{h,T},$$

holds. As a result of Corollary 5.1.2, (5.1) and (5.2) holds for the iterative solutions satisfying the following stopping criterion.

Theorem 5.1.4 *Let* $\alpha = \min_{T\in\Im_h} \frac{\eta_{h,T}}{\eta_{h_p,T_p}}$. *If the number of iterations n is large enough such that the residual*

$$\|\gamma_h^n\|_T \preceq \alpha_\eta \frac{\epsilon}{h}\eta_{h_p,T_p}, \quad \forall T \in \Im_h, \tag{5.15}$$

then

$$\frac{1}{2}\eta_{h,T} \leq \eta_{h,T}^n \leq \frac{3}{2}\eta_{h,T}, \tag{5.16}$$

and

$$|||u_h - u_h^n|||_\Omega \leq (\sum_{T\in\Im_h} \eta_{h,T}^2)^{1/2}. \tag{5.17}$$

□

In next theorem, we provide a stopping criterion such that the global a posteriori error bound won't be affected by the iterative solution satisfying this stopping criterion. For this purpose, we assume that the finite element solutions strictly converge to the weak solution u with a rate slower than $h^{3/2}$ along the adaptive mesh refinement process, i.e.

$$\frac{1}{2\sqrt{2}} < \frac{|||u - u_h|||_\Omega}{|||u - u_{h_p}|||_\Omega}, \tag{5.18}$$

where u_h is the finite element solution on $\Im_h$ and u_{h_p} is the finite element solution on the parent mesh $\Im_{h_p}$ of $\Im_h$. This assumption is generally true because the a priori error estimation in Chapter 2 only shows $h^{3/2}$ convergence and the numerical studies

in Chapter 2 even suggest the convergence rate is only $h^{1/2}$.

From the local lower bound in [97] Proposition 4.1, we have

$$\begin{aligned}
\sum_{T_p \in \Im_{h_p}} {\eta_{h_p,T_p}}^2 &\leq \sum_{T \in \Im_{h_p}} \{1 + \|c\|_{L^\infty_{\omega_T}} + \epsilon^{-1/2} \|b\|_{\infty,\omega_T} \alpha_T\}^2 \left|\left|\left|u - u_{h_p}\right|\right|\right|^2_{\omega_{T_p}} \\
&\leq 4\{1 + \|c\|_\infty + \epsilon^{-1/2} \|b\|_\infty \max_{T \in \Im_{h_p}} \alpha_{T_p}\}^2 \left|\left|\left|u - u_{h_p}\right|\right|\right|^2_\Omega \\
&\leq 32\{1 + \|c\|_\infty + 2\epsilon^{-1/2} \|b\|_\infty \max_{T \in \Im_{h_p}} \alpha_T\}^2 |||u - u_h|||^2_\Omega \,, \text{ by (5.18)}, \\
&\approx 64 C'^2 \sum_{T \in \Im_h} \eta^2_{h,T},
\end{aligned} \tag{5.19}$$

where

$$C' = \epsilon^{-1/2} \|b\|_\infty \max_{T \in \Im_{h_p}} \alpha_T = \begin{cases} \|b\|_\infty \epsilon^{-1/2} & \text{if } h_{max} > \sqrt{\epsilon}. \\ \|b\|_\infty h_{max} \epsilon^{-1} & \text{otherwise} \end{cases} \tag{5.20}$$

Clearly, if n is large enough such that $\|r^n_h\|_\Omega \leq \frac{1}{8\kappa C'}(\sum_{T \in \Im_h} \eta^2_{h_p,T_p})^{1/2}$, by (5.14), the above inequality implies

$$|||u_h - u^n_h|||_\Omega \;\preceq\; \sum_{T \in \Im_h} \eta^2_{h,T}.$$

Therefore, from (5.14) and (5.20), the following theorem holds.

Theorem 5.1.5 *Assume (5.18) holds. If n is large enough such that*

$$\|r^n_h\|_\Omega \leq \frac{1}{8\|b\|_\infty} C(\sum_{T \in \Im_{h_p}} \eta^2_{h_p,T_p})^{1/2}, \tag{5.21}$$

where $C = \epsilon/h_{max}$. We have

$$|||u_h - u^n_h|||_\Omega \leq (\sum_{T \in \Im_h} \eta^2_{h,T})^{1/2}.$$

Although the a posteriori upper bound is maintained for iterative solutions satisfying the stopping criterion (5.21), without Theorem 5.1.4, there is no guarantee that the marking strategies will select the same elements that would be selected if the error indicator is computed from the exact solution. Unfortunately, the constant α_η in Theorem 5.1.4 is unknown due to lack of estimations on local error reduction rate. To deal with this difficulty, the marking strategy has to be taken into consideration in the search for a stopping criterion. In the following theorem, we show that the error indicators $\eta^n_{h,T}$ and $\eta_{h,T}$ are similar in regions where elements are selected by the maximum marking strategy, and that serious over-refinement will not occur when the iterative solutions satisfy our stopping criterion.

Theorem 5.1.6 *Let $\alpha_{\eta,\infty}$ be a constant satisfying*

$$\alpha_{\eta,\infty} \leq \frac{\max_{T\in\Im_h} \eta_{h,T}}{\max_{T_p\in\Im_{h_p}} \eta_{h_p,T_p}}. \tag{5.22}$$

Assume the maximum marking strategy is used with threshold value θ. If

$$\|\gamma^n_h\|_T \preceq (\frac{\epsilon}{4h_p})\alpha_{\eta,\infty}\theta \max_{T_p\in\Im_{h_p}} \eta_{h_p,T_p}, \quad \textit{for all } T \in \Im_h, \tag{5.23}$$

then

$$\frac{1}{2}\eta_{h,T} \leq \eta^n_{h,T} \leq \frac{3}{2}\eta_{h,T}, \tag{5.24}$$

for any marked element T. On the other hand, for element $\bar{T}$ satisfying

$$\eta_{h,\bar{T}} < \frac{\theta}{4} \max_{T\in\Im_h} \eta_{h,T} \tag{5.25}$$

will not be marked by the same marking strategy with $\eta_{h,T}$ replaced by $\eta^n_{h,T}$.

Proof: First, for any element $\bar{T} \in \Im_h$, (5.13) and (5.22) imply

$$\begin{aligned} |||u_h - u^n_h|||_{\omega_{\bar{T}}} &\preceq \frac{1}{4}\epsilon^{1/2} \max\{\frac{\sqrt{\epsilon}}{h}, 1\}\alpha_{\eta,\infty}\theta \max_{T_p\in\Im_{h_p}} \eta_{h_p,T_p} \\ &< \frac{1}{4}\epsilon^{1/2} \max\{\frac{\sqrt{\epsilon}}{h}, 1\}\theta \max_{T\in\Im_h} \eta_{h,T}. \end{aligned} \tag{5.26}$$

Let $\bar{T}$ be a marked element satisfying

$$\eta_{h,\bar{T}} \geq \theta \max_{T \in \Im_h} \eta_{h,T}. \tag{5.27}$$

From (5.26), we have

$$|||u_h - u_h^n|||_{\omega_{\bar{T}}} < \frac{1}{4}\epsilon^{1/2} \max\{\frac{\sqrt{\epsilon}}{h}, 1\}\eta_{h,\bar{T}}.$$

By Corollary 5.1.2, the inequality (5.24) holds. Now, let $\bar{T}$ be an element satisfying (5.25). Recalling (5.6), we have

$$|\eta_{h,\bar{T}} - \eta_{h,\bar{T}}^n| \leq (\epsilon^{-1/2} \min\{\frac{h}{\sqrt{\epsilon}}, 1\}) \, |||u_h - u_h^n|||_{\omega_{\bar{T}}} . \tag{5.28}$$

By combining (5.26) and (5.28), we have

$$|\eta_{h,\bar{T}} - \eta_{h,\bar{T}}^n| \leq \frac{\theta}{4} \max_{T \in \Im_h} \eta_{h,T}.$$

Therefore,

$$\begin{aligned} \eta_{h,\bar{T}}^n &\leq \eta_{h,\bar{T}} + \frac{\theta}{4} \max_{T \in \Im_h} \eta_{h,T} \\ &\leq \frac{\theta}{2} \max_{T \in \Im_h} \eta_{h,T} \text{ by (5.25)}, \\ &\leq \theta \max_{T \in \Im_h} \eta_{h,T}^n, \text{ by (5.24)}. \end{aligned}$$

The second part of the theorem is proved.

□

Now, let us consider the marking strategy in [32] where a set of elements, $\Im_h^*$ are marked such that

$$(\sum_{T \in \Im_h^*} \eta_{h,T}^2)^{1/2} \geq \theta (\sum_{T \in \Im_h} \eta_{h,T}^2)^{1/2}, \tag{5.29}$$

where $0 < \theta \leq 1$. Assume n is large enough such that

$$|||u_h - u_h^n|||_\Omega^2 \leq (\alpha c_{h,\omega_T})^2 \sum_{T_p \in \Im_{h_p}} \eta_{h_p,T_p}^2. \tag{5.30}$$

We have

$$\begin{aligned}
\sum_{T \in \Im_h^*} |||u_h - u_h^n|||_{\omega_T}^2 &\leq 4\,|||u_h - u_h^n|||_\Omega^2 \\
&\preceq (\alpha c_{h,\omega_T})^2 \sum_{T_p \in \Im_{h_p}} \eta_{h_p,T_p}^2 \\
&\leq 64(\alpha c_{h,\omega_T})^2 C'^2 \sum_{T \in \Im_h} \eta_{h,T}^2, \text{ by (5.19)}, \\
&\leq 64(\alpha c_{h,\omega_T} C')^2 \frac{1}{\theta^2} \sum_{T \in \Im_h^*} \eta_{h,T}^2.
\end{aligned}$$

Let us choose $\alpha = \frac{\theta}{16\sqrt{2}C'\|b\|_\infty} = \frac{\theta}{16\sqrt{2}\|b\|_\infty^2}\epsilon^{1/2}\max\{\frac{\sqrt{\epsilon}}{h_{max}}, 1\}$. Obviously, (5.19) and (5.30) implies

$$|||u_h - u_h^n|||_\Omega \ll (\sum_{T \in \Im_h} \eta_{h,T}^2)^{1/2}.$$

Moreover, from Corollary 5.1.3, we have

$$\frac{1}{2}(\sum_{T \in \Im_h^*} \eta_{h,T}^2)^{1/2} \leq (\sum_{T \in \Im_h^*} {\eta_{h,T}^n}^2)^{1/2} \leq \frac{3}{2}(\sum_{T \in \Im_h^*} \eta_{h,T}^2)^{1/2}.$$

Using similar argument, the following inequality also holds:

$$(1 - \frac{\theta}{2})(\sum_{T \in \Im_h} \eta_{h,T}^2)^{1/2} \leq (\sum_{T \in \Im_h} {\eta_{h,T}^n}^2)^{1/2} \leq (1 + \frac{\theta}{2})(\sum_{T \in \Im_h} \eta_{h,T}^2)^{1/2}. \tag{5.31}$$

Recalling (5.14), we have

$$|||u_h - u_h^n|||_\Omega \leq \kappa \left\|\gamma_h^n\right\|_\Omega, \text{ where } \kappa = \max\{\frac{h}{\sqrt{\epsilon}}, 1\}. \tag{5.32}$$

A computable stopping criterion similar to (5.15) can be shown in the following, without assuming $\min_{T \in \Im_h} \frac{\eta_{h,T}}{\eta_{h_p,T_p}} = O(1)$.

Theorem 5.1.7 *Suppose the marking strategy (5.29) is used. If the iteration number is large enough such that*

$$\|\gamma_h^n\|_\Omega \preceq \frac{\theta}{8\sqrt{2}\,\|b\|_\infty^2}\frac{\epsilon^{3/2}}{h}C(\sum_{T_p\in\Im_{h_p}}{\eta_{h_p,T_p}}^2)^{1/2}, \tag{5.33}$$

where $C=\max\{\frac{2\sqrt{\epsilon}}{h_p},1\}$*, then there exist a small constant* c_0 *such that*

$$|||u_h-u_h^n|||_\Omega \le c_0(\sum_{T\in\Im_h}\eta_{h,T}^2)^{1/2}, \tag{5.34}$$

and

$$\frac{1}{2}(\sum_{T\in\Im_h^*}\eta_{h,T}^2)^{1/2}\le(\sum_{T\in\Im_h^*}{\eta_{h,T}^n}^2)^{1/2}\le\frac{3}{2}(\sum_{T\in\Im_h^*}\eta_{h,T}^2)^{1/2}. \tag{5.35}$$

Proof: From (5.30) and (5.32), if

$$\|\gamma_h^n\|_\Omega \le \frac{\alpha c_{h,\omega_T}}{\kappa}(\sum_{T_p\in\Im_{h_p}}\eta_{h_p,T_p}^2)^{1/2},$$

then (5.30) holds. As a result, from the above argument, (5.34) and (5.35) hold. Since

$$\kappa=\max\{\frac{h}{\sqrt{\epsilon}},1\},\ \alpha=\frac{\theta}{16\sqrt{2}\,\|b\|_\infty}\epsilon^{1/2}\max\{\frac{\sqrt{\epsilon}}{h},1\}\text{ and }c_{h,\omega_T}=\epsilon^{1/2}\max\{\frac{\sqrt{\epsilon}}{h},1\},$$

we have

$$\frac{\alpha c_{h,\omega_T}}{\kappa}=\frac{\theta}{8\sqrt{2}\,\|b\|_\infty}\frac{\epsilon^{3/2}}{h_p}\max\{\frac{2\sqrt{\epsilon}}{h_p},1\}.$$

Therefore, (5.33) implies (5.34) and (5.35).

□

Remark 5.1.8 *From (5.31) and (5.35),*

$$(\sum_{T\in\Im_h^*}{\eta_{h,T}^n}^2)^{1/2}\ge\frac{1}{2}(\sum_{T\in\Im_h^*}{\eta_{h,T}}^2)^{1/2}>\frac{\theta}{2}(\sum_{T\in\Im_h}{\eta_{h,T}}^2)^{1/2}\ge\frac{\theta}{2+\theta}(\sum_{T\in\Im_h}{\eta_{h,T}^n}^2)^{1/2}.$$

Therefore, for any subset $\Im_h^* \subset \Im_h$, which satisfies (5.29), can also be marked by the following marking strategy:

$$(\sum_{T\in\Im_h^*} {\eta_{h,T}^n}^2)^{1/2} \geq \frac{\theta}{2+\theta}(\sum_{T\in\Im_h} {\eta_{h,T}^n}^2)^{1/2}. \tag{5.36}$$

However, for large θ, (5.36) results in under-refinement comparing to the mesh generated from marking strategy (5.29). Hence, more iterative steps are needed to overcome this drawback. On the other hand, one can also employee the following strategy:

Let $\bar{\Im}_h$ be the maximal subset such that

$$(\sum_{T\in\bar{\Im}_h} {\eta_{h,T}^n}^2)^{1/2} < \frac{\theta}{2+\theta}(\sum_{T\in\Im_h} {\eta_{h,T}^n}^2)^{1/2}.$$

Elements in the the complement set of $\bar{\Im}_h$ are marked for mesh refinement.

For large θ, the above marking strategy produces less under-refinement. For example, for $\theta = 1$, (5.29) produces fully refinement and, obviously, the above marking strategy marks more elements than (5.36).

5.2 Stopping Criteria Associated with Neumann-Type a Posteriori Error Estimation

Using the same analysis as in section 5.1, we can derive a similar stopping criterion for iterative solvers when the Kay-Silvester error indicator is employed for mesh refinement. Recall we assume the interpolation errors are high order terms and can be ignored. Hence, in the following analysis, the second term in the a posteriori upper

bound will be ignored. Again, one would like to have enough iterations such that the following inequalities hold,

$$\|\nabla(u_h - u_h^n)\|_0 \leq c_0(\sum_{T\in\Im_h} \eta_{h,T}^2)^{1/2}, \text{ for a small constant } c_0, \tag{5.37}$$

and

$$c_1\eta_{h,T} < \eta_{h,T}^n < c_2\eta_{h,T}, \text{ for some small constants } c_1 \text{ and } c_2, \tag{5.38}$$

where u_h is the exact finite element solution, u_h^n is the iterative solution and $\eta_{h,T}$, $\eta_{h,T}^n$ are the corresponding error indicators. In this case, first, we also assume

$$\|\nabla(u_h - u_h^n)\|_{0,T} \leq c_{h,\omega_T}\eta_{h,T}. \tag{5.39}$$

Here $\eta_{h,T} = \|\nabla e_T\|_{0,T}$ is the error indicator with $e_T \in Q_T$ satisfying

$$\epsilon\left(\nabla e_T, \nabla v\right)_T = \left(R_T^0, v\right)_T - \frac{1}{2}\epsilon \sum_{E\in\mathcal{E}(T)} \left(R_E, v\right)_E, \tag{5.40}$$

where

$$\begin{aligned} R_T &= f - b\cdot\nabla u_h, \\ R_T^0 &= \pi^0(R_T), \\ R_E &= \begin{cases} [|\ \frac{\partial u_h}{\partial n_E}\ |]_E & E \in \mathcal{E}_{h,\Omega} \\ -2(\frac{\partial u_h}{\partial n_E}) & E \in \mathcal{E}_{h,N} \\ 0 & E \in \mathcal{E}_{h,D}, \end{cases} \end{aligned}$$

and π^0 is the L^2 projection onto constant function space $\mathbf{P}_0(T)$.

Let $u_i \in H^1(\Omega)$ and $e_{i,T} \in Q_T$ satisfy

$$\epsilon\left(\nabla e_{i,T}, \nabla v\right)_T = \left(R_{i,T}^0, v\right)_T - \frac{1}{2}\epsilon \sum_{E\in\mathcal{E}(T)} \left(R_{i,E}, v\right)_E, \text{ for } i = 1, 2, \tag{5.41}$$

where

$$\begin{aligned} R_{i,T} &= f - b \cdot \nabla u_i, \\ R^0_{i,T} &= \pi^0(R_{i,T}), \\ R_{i,E} &= \begin{cases} [|\ \frac{\partial u_i}{\partial n_E}\ |]_E & E \in \mathcal{E}_{h,\Omega} \\ -2(\frac{\partial u_i}{\partial n_E}) & E \in \mathcal{E}_{h,N} \\ 0 & E \in \mathcal{E}_{h,D}. \end{cases} \end{aligned}$$

From (5.41), we have

$$\epsilon \left(\nabla(e_{1,T} - e_{2,T}), \nabla v\right)_T = \left(R^0_{1,T} - R^0_{2,T}, v\right)_T - \frac{1}{2}\epsilon \sum_{E \in \mathcal{E}(T)} \left(R_{1,E} - R_{2,E}, v\right)_E . \tag{5.42}$$

Let $v = e_{1,T} - e_{2,T}$. From the Schwartz inequality, (5.42) implies

$$\begin{aligned} \epsilon \left\|\nabla(e_{1,T} - e_{2,T})\right\|^2_{0,T} \leq & \underbrace{\left\|R^0_{1,T} - R^0_{2,T}\right\|_{0,T} \left\|e_{1,T} - e_{2,T}\right\|_{0,T}}_{I} \\ & + \underbrace{\frac{1}{2}\epsilon \sum_{E \in \mathcal{E}(T)} \left\|R_{1,E} - R_{2,E}\right\|_{0,E} \left\|e_{1,T} - e_{2,T}\right\|_{0,E}}_{II} . \end{aligned} \tag{5.43}$$

First, let's estimate $\left\|R^0_{1,T} - R^0_{2,T}\right\|_{0,T}$:

$$\begin{aligned} \left\|R^0_{1,T} - R^0_{2,T}\right\|_{0,T} &= \left\|\pi^0(f - b \cdot \nabla u_1) - \pi^0(f - b \cdot \nabla u_2)\right\|_{0,T} \\ &= \left\|\pi^0(b \cdot (\nabla(u_2 - u_1)))\right\|_{0,T} \\ &\preceq \left\|b \cdot \nabla(u_1 - u_2)\right\|_{0,T} \\ &\leq \left\|b\right\|_{\infty,T} \left\|\nabla(u_1 - u_2)\right\|_{0,T} . \end{aligned} \tag{5.44}$$

Since $e_{1,T} - e_{2,T} \in Q_T$, from a scaling argument, we have

$$\left\|e_{1,T} - e_{2,T}\right\|_{0,T} \leq C(\theta_T) h_T \left\|\nabla(e_{1,T} - e_{2,T})\right\|_{0,T} . \tag{5.45}$$

From (5.44) and (5.45), it is clear that

$$(I) \leq C(\theta_T) \left\|b\right\|_{\infty,T} h_T \left\|\nabla(u_1 - u_2)\right\|_{0,T} \left\|\nabla(e_{1,T} - e_{2,T})\right\|_{0,T} \tag{5.46}$$

Now, let's estimate $\left\|R_{1,E} - R_{2,E}\right\|_{0,E}$. For $E \in \mathcal{E}_{h,n}$, using the trace inequality,

$$\begin{aligned}
\left\|R_{1,E} - R_{2,E}\right\|_{0,E} &= \left\| [\!| \frac{\partial u_1}{\partial n_E} |\!]_E - [\!| \frac{\partial u_2}{\partial n_E} |\!]_E \right\|_{0,E} \\
&= \left\| [\!| \frac{\partial u_1}{\partial n_E} - \frac{\partial u_2}{\partial n_E} |\!]_E \right\|_{0,E} \\
&\leq h_T^{-1/2} \left\| [\!| \frac{\partial (u_1 - u_2)}{\partial n_E} |\!]_E \right\|_{0,T} \\
&\leq h_T^{-1/2} (\left\|\nabla(u_1 - u_2)\right\|_{0,T} + \left\|\nabla(u_1 - u_2)\right\|_{0,T_{nb}}),
\end{aligned} \tag{5.47}$$

where T_{nb} is the triangle sharing edge E with T, ie, $T_{nb} \cap T = E$.

A similar result holds for $E \in \mathcal{E}_{h,N}$. Again, from a scaling argument, we have

$$\left\|e_{1,T} - e_{2,T}\right\|_{0,E} \leq C(\theta) h_E^{1/2} \left\|\nabla(e_{1,T} - e_{2,T})\right\|_{0,T}. \tag{5.48}$$

By (5.47) and (5.48), we have

$$\begin{aligned}
(II) &\leq \frac{1}{2} \epsilon \sum_{E \in \mathcal{E}} C(\theta_T) h_E^{1/2} h_T^{-1/2} [\left\|\nabla(u_1 - u_2)\right\|_{0,T} + \left\|\nabla(u_1 - u_2)\right\|_{0,T_{nb}}] \left\|\nabla(e_{1,T} - e_{2,T})\right\|_{0,T} \\
&\leq \frac{3}{2} C(\theta_T) \max_{E \in \mathcal{E}} (\frac{h_E}{h_T})^{1/2} \epsilon \left\|\nabla(u_1 - u_2)\right\|_{0,\omega_T} \left\|\nabla(e_{1,T} - e_{2,T})\right\|_{0,T}.
\end{aligned} \tag{5.49}$$

Let $C_I = C(\theta_T) \left\|b\right\|_{\infty,T} (\frac{h_T}{\epsilon})$ and $C_{II} = \frac{3}{2} C(\theta_T) \max_{E \in \mathcal{E}(T)} (\frac{h_E}{h_T})^{1/2}$. By combining (5.43), (5.46) and (5.49), we have

$$\left\|\nabla(e_{1,T} - e_{2,T})\right\|_{0,T} \leq [C_I + C_{II}] \left\|\nabla(u_1 - u_2)\right\|_{0,\omega_T} \approx C_I \left\|\nabla(u_1 - u_2)\right\|_{0,\omega_T},$$

because C_I is the dominating term when $\epsilon \ll h$. Recall $\eta^1_{h,T} = \|\nabla e_{1,T}\|_{0,T}$ and $\eta^2_{h,T} = \|\nabla e_{2,T}\|_{0,T}$. The above inequality implies

$$|\eta^1_{h,T} - \eta^2_{h,T}| \preceq C_I \|\nabla(u_1 - u_2)\|_{0,\omega_T} . \tag{5.50}$$

Clearly, if $\|\nabla(u_1 - u_2)\|_{0,\omega_T} \leq \frac{1}{2C_I}\eta^1_{h,T}$, we have

$$\frac{1}{2}\eta^1_{h,T} \leq \eta^2_{h,T} \leq \frac{3}{2}\eta^1_{h,T}. \tag{5.51}$$

Now a result analogous to Lemma 5.1.1 can be written as follows:

Lemma 5.2.1 *Let $\eta^1_{h,T}$ and $\eta^2_{h,T}$ be the error indicator of u_1 and u_2 on element T respectively. If*

$$\|\nabla(u_1 - u_2)\|_{0,\omega_T} \leq c_{h,\omega_T}\eta^1_{h,T}, \text{ where } c_{h,\omega_T} = O(\frac{\epsilon}{h}), \tag{5.52}$$

then

$$\frac{1}{2}\eta^1_{h,T} \leq \eta^2_{h,T} \leq \frac{3}{2}\eta^1_{h,T}. \tag{5.53}$$

By replacing $\eta^1_{h,T}$ and $\eta^2_{h,T}$ and by $\eta_{h,T}$ and $\eta^n_{h,T}$, the following corollary holds.

Corollary 5.2.2 *Let u_h be the finite element solution and u^n_h be the iterative solution. If the iteration steps are large enough such that*

$$\|\nabla(u_h - u^n_h)\|_{0,\omega_T} \leq c_{h,\omega_T}\eta_{h,T}, \text{ where } c_{h,\omega_T} = O(\frac{\epsilon}{h}), \tag{5.54}$$

then

$$\frac{1}{2}\eta_{h,T} \leq \eta^n_{h,T} \leq \frac{3}{2}\eta_{h,T}. \tag{5.55}$$

Of course, one can also obtain a similar result as in Corollary 5.1.3.

Corollary 5.2.3 *Let u_h be the finite element solution and u_h^n be the iterative solution. If the iterative steps are large enough such that*

$$(\sum_{T\in\Im_h^*} \|\nabla(u_h - u_h^n)\|_{0,\omega_T}^2)^{1/2} \leq \min_{T\in\Im_h^*}\{c_{h,\omega_T}\}(\sum_{T\in\Im_h^*}\eta_{h,T}^2)^{1/2} \tag{5.56}$$

, where $\Im_h^* \subset \Im_h$ and $c_{h,\omega_T} = O(\frac{\epsilon}{h})$ then

$$\frac{1}{2}(\sum_{T\in\Im_h^*}\eta_{h,T}^2)^{1/2} \leq (\sum_{T\in\Im_h^*}{\eta_{h,T}^n}^2)^{1/2} \leq \frac{3}{2}(\sum_{T\in\Im_h^*}\eta_{h,T}^2)^{1/2}. \tag{5.57}$$

□

Let r_h^n be the residual of the nth iterative solution. Since

$$\begin{aligned}
\|r_h^n\|_T &= \|f_h - A_h u_h^n\|_T \\
&= \|A_h(u_h - u_h^n)\|_T \\
&\geq \min\Lambda(A_h A_h^*)^{1/2}\|u_h - u_h^n\|_T \\
&\succeq \sqrt{\epsilon}h^{-1}\|u_h - u_h^n\|_{0,\omega_T} \\
&\succeq \sqrt{\epsilon}\|\nabla(u_h - u_{h,n})\|_{0,\omega_T} \quad \text{, by inverse inequality,}
\end{aligned}$$

we have

$$\|\nabla\cdot(u_h - u_h^n)\|_{0,\omega_T} \preceq \epsilon^{-1/2}\|\gamma_h^n\|_T. \tag{5.58}$$

The same analysis also gives

$$\|\nabla\cdot(u_h - u_h^n)\|_{0,\Omega_T} \preceq \epsilon^{-1/2}\|\gamma_h^n\|_\Omega. \tag{5.59}$$

From Corollary 5.2.2 and (5.58), obviously, the following theorem holds.

Theorem 5.2.4 *Let $\alpha_\eta = \min_{T\in\Im_h}\frac{\eta_{h,T}}{\eta_{h_p,T_p}}$, where $T_p \in \Im_{h_p}$ is the parent triangle of T with diameter h_p. If the number of iteration is large enough such that the residual*

$$\|\gamma_h^n\|_T \preceq \alpha_\eta\frac{\epsilon^{3/2}}{h_p}\eta_{h_p,T_p}, \quad \forall T\in\Im_h, \tag{5.60}$$

then

$$\frac{1}{2}\eta_{h,T} < \eta_{h,T}^n < \frac{3}{2}\eta_{h,T}, \tag{5.61}$$

and

$$\|\nabla(u_h - u_h^n)\|_{0,\Omega} \preceq (\sum_{T\in\Im_h} \eta_{h,T}^2)^{1/2}. \tag{5.62}$$

In next theorem, we provide a computable stopping criterion such that (5.37) holds by assuming

$$\frac{1}{2} \leq \frac{|u - u_h|_{1,\Omega}}{|u - u_{h_p}|_{1,\Omega}}, \tag{5.63}$$

where u is the weak solution and u_{h_p} is the finite element solution on parent mesh $\Im_{h_p}$. Since the interpolation error is only O(h) in H^1 norm, this assumption is reasonable.

Theorem 5.2.5 *Assume (5.63) holds. If n is large enough such that the residual r_h^n of nth iterative solution satisfying*

$$\|r_h^n\|_\Omega \preceq \frac{\epsilon}{h_{max}}(\sum_{T\in\Im_{h_p}} \eta_{h_p,T_p}^2)^{1/2}, \tag{5.64}$$

where h_{max} is the maximum diameter of triangles in $\Im_h$, we have

$$\|\nabla(u_h - u_h^n)\|_{0,\Omega} \preceq (\sum_{T\in\Im_h} \eta_{h,T}^2)^{1/2}.$$

Proof: From the local lower bound in [59] Theorem 1,

$$\begin{aligned}
(\sum_{T_p\in\Im_{h_p}} {\eta_{h_p,T_p}}^2) &\preceq \sum_{T\in\Im_{h_p}} (\left\|\nabla(u - u_{h_p})\right\|_{0,\omega_{T_p}} + \frac{h_p}{\epsilon}\sum_{T\in\omega_{T_p}} \|b\|_{\infty,T} \left\|\nabla(u - u_{h_p})\right\|_{0,T})^2 \\
&\preceq (\frac{2h_{max}}{\epsilon}\|b\|_\infty)^2 \sum_{T_p\in\Im_{h_p}} \left\|\nabla(u - u_{h_p})\right\|_{0,\omega_{T_p}}^2 \\
&\leq 16(\frac{h_{max}}{\epsilon}\|b\|_\infty)^2 \left\|u - u_{h_p}\right\|_\Omega^2 \\
&\leq 64(\frac{h_{max}}{\epsilon}\|b\|_\infty)^2 \|u - u_h\|_\Omega^2, \text{ by (5.63)} \\
&\leq 64(\frac{h_{max}}{\epsilon}\|b\|_\infty)^2 \sum_{T\in\Im_h} \eta_{h,T}^2.
\end{aligned} \tag{5.65}$$

By plugging the above estimate (5.64) and (5.65) into (5.59), the theorem holds.

□

Although, the Theorem 5.2.5 provide a computable stopping criterion such that the global posteriori upper bound will not be violated, the refined mesh generated from such iterative solution can be very different from the refined mesh generated from exact solution without Theorem 5.2.4. On the other hand, in Theorem 5.2.4, $\frac{\epsilon^{3/2}}{h}$ and η_{h_p,T_p} can be very small in the regions where elements have never been refined by the mesh refinement process. As a result, one may need a very large iteration number for (5.60) to be satisfied on elements in these regions. Therefore, even α_η can be estimated, (5.60) may still not be a proper stopping criterion for iterative solvers in real applications, especially when ϵ is small. Again, one needs to take the marking strategy into consideration in finding a suitable stopping criterion. In mesh refinement point of view, intuitively, it is not necessary to keep the same profile between $\eta_{h,T}$ and $\eta^n_{h,T}$ in the unmarked regions. The stopping criterion in the following lemma guarantees that when the maximum marking strategy is used, the mesh generated from $\eta^n_{h,T}$ will not produce serious over-refinement compared to the mesh generated from $\eta_{h,T}$. Moreover, the same profile is kept between $\eta_{h,T}$ and $\eta^n_{h,T}$ in the marked regions.

Theorem 5.2.6 *Let $\alpha_{\eta,\infty}$ be a constant satisfying*

$$\alpha_{\eta,\infty} \leq \frac{\max_{T\in\Im_h} \eta_{h,T}}{\max_{T_p\in\Im_{h_p}} \eta_{h_p,T_p}}. \tag{5.66}$$

Assume the maximum marking strategy is used with threshold value θ. If

$$\|\gamma^n_h\|_T \preceq (\frac{\epsilon^{3/2}}{4h_p})\alpha_{\eta,\infty}\theta \max_{T_p\in\Im_{h_p}} \eta_{h_p,T_p}, \quad \textit{for all } T \in \Im_h, \tag{5.67}$$

then

$$\frac{1}{2}\eta_{h,T} \leq \eta^n_{h,T} \leq \frac{3}{2}\eta_{h,T}, \tag{5.68}$$

for any marked element T. On the other hand, for element $\bar{T}$ satisfying

$$\eta_{h,\bar{T}} < \frac{\theta}{4}\max_{T\in\Im_h}\eta_{h,T} \tag{5.69}$$

will not be marked by the same marking strategy with $\eta_{h,T}$ replaced by $\eta^n_{h,T}$.

Proof: First, for any element $\bar{T} \in \Im_h$, (5.58) and (5.67) imply

$$\begin{aligned} \|\nabla(u_h - u^n_h)\|_{0,\omega_{\bar{T}}} &\preceq \frac{\epsilon}{4h_p}\alpha_{\eta,\infty}\theta \max_{T_p\in\Im_{h_p}} \eta_{h_p,T_p} \\ &< \frac{\epsilon}{4h}\theta\max_{T\in\Im_h}\eta_{h,T}. \end{aligned} \tag{5.70}$$

Let $\bar{T}$ be a marked element satisfying

$$\eta_{h,\bar{T}} \geq \theta\max_{T\in\Im_h}\eta_{h,T}. \tag{5.71}$$

From (5.70), we have

$$\|\nabla(u_h - u^n_h)\|_{0,\omega_{\bar{T}}} < \frac{\epsilon}{h}\eta_{h,\bar{T}}.$$

By Corollary 5.2.2, the inequality (5.68) holds. Now, let $\bar{T}$ be an element satisfying (5.69). Recall that (5.50) implies

$$\frac{\epsilon}{h}|\eta_{h,\bar{T}} - \eta^n_{h,\bar{T}}| \leq \|\nabla(u_h - u^n_h)\|_{0,\omega_{\bar{T}}}. \tag{5.72}$$

By combining (5.70) and (5.72), we have

$$|\eta_{h,\bar{T}} - \eta^n_{h,\bar{T}}| \leq \frac{\theta}{4}\max_{T\in\Im_h}\eta_{h,T}.$$

Therefore,

$$\begin{aligned} \eta^n_{h,\bar{T}} &\leq \eta_{h,\bar{T}} + \frac{\theta}{4}\max_{T\in\Im_h}\eta_{h,T} \\ &\leq \frac{\theta}{2}\max_{T\in\Im_h}\eta_{h,T} \text{ by (5.69)}, \\ &\leq \theta\max_{T\in\Im_h}\eta^n_{h,T}, \text{ by (5.68)}. \end{aligned}$$

The second part of the theorem is proved.

□

Next, let's consider the marking strategy (5.29). By using an argument similar to that used in Theorem 5.1.7, we show that iterative solutions satisfying the following stopping criterion can safely replace the exact solution.

Theorem 5.2.7 *If the marking strategy (5.29) is used and the number of iterative steps is large enough such that*

$$\|\gamma_h^n\|_\Omega \preceq \frac{\theta}{64\,\|b\|_\infty^2}\frac{\epsilon^{5/2}}{h_p^2}(\sum_{T_p\in\Im_{h_p}} {\eta_{h_p,T_p}}^2)^{1/2}, \tag{5.73}$$

then there exist a small constant c_0 such that

$$\|\nabla(u_h-u_h^n)\|_\Omega \le c_0(\sum_{T\in\Im_h}\eta_{h,T}^2)^{1/2}, \tag{5.74}$$

and

$$\frac{1}{2}(\sum_{T\in\Im_h^*}\eta_{h,T}^2)^{1/2} \le (\sum_{T\in\Im_h^*}{\eta_{h,T}^n}^2)^{1/2} \le \frac{3}{2}(\sum_{T\in\Im_h^*}\eta_{h,T}^2)^{1/2}. \tag{5.75}$$

Proof: First, it is clear (5.74) holds from Theorem 5.2.5. Now, since

$$\|\nabla(u_h-u_h^n)\|_\Omega \preceq \epsilon^{-1/2}\,\|\gamma_h^n\|_\Omega\,,$$

we have

$$\begin{aligned}
\sum_{T\in\Im_h^*}\|\nabla(u_h-u_h^n)\|_{\omega_T}^2 &\le 4\|\nabla(u_h-u_h^n)\|_\Omega^2\\
&\le 4\epsilon^{-1}\,\|\gamma_h^n\|_\Omega^2\\
&\le 4(\frac{\theta}{64\,\|b\|_\infty^2})^2\frac{\epsilon^4}{h_p^4}\sum_{T_p\in\Im_{h_p}}{\eta_{h_p,T_p}}^2\\
&\le (\frac{\theta^2}{64\,\|b\|_\infty^2})\frac{\epsilon^2}{h^2}\sum_{T\in\Im_h}\eta_{h,T}^2\\
&\le (\frac{1}{64\,\|b\|_\infty^2})\frac{\epsilon^2}{h^2}\sum_{T\in\Im_h^*}\eta_{h,T}^2.
\end{aligned}$$

Therefore, (5.75) is a direct result from Corollary 5.2.3.

□

5.3 Numerical Results

In this section, we compare the refined meshes of Problems 1, 2 and 3 for different values of $\epsilon = 10^{-2}, 10^{-3}$ and 10^{-4}. The iteration steps among different linear solvers and stopping criteria are also compared. In each problem, first the linear systems are directly solved on the coarsest 4x4 grid. Then the following procedures are followed:

1. Compute error estimator η.
2. Select elements according to the maximum marking strategy.
3. Refine selected elements and generate a new mesh.
4. Obtain the initial guess by interpolating the current solution to the new mesh.
5. Solve linear system so that a given stopping criterion S_i is satisfied.

Three different $S_i,\ i = 0, 1, 2$, are chosen. If S_0 is given, the linear systems are solved directly. S_1 is the heuristic stopping tolerance, i.e., the L^2 norm of the residual of iterative solutions less than 10^{-6}. S_2 is the stopping criterion in Theorem 5.1.6 and 5.2.6. The threshold θ in the maximum strategy is carefully chosen so that more detail layer structures of the solutions can be seen during each refinement step in both interior and boundary layer regions. The threshold is set to 0.25 for Problem 1. For Problem 2 and 4, the threshold is set to 0.1. For the number of refinement steps, four steps are performed for the case $\epsilon = 10^{-2}$, seven steps are performed for the case $\epsilon = 10^{-3}$, and eight steps are performed for the cases 10^{-4}. Both MG and GMRES with the same

Gauss-Seidel smooother or preconditioner are employed as the iterative solvers. One HGS step is applied on Problem 1, one VGS step is applied on problem 2 and one ADGS step, consisting of HGS, VGS, backward HGS and backward VGS, is applied on Problem 4.

As shown in the following numerical results, the meshes, generated from MG or GMRES iterative solutions that satisfying our stopping criteria, are almost the same as the mesh generated from exact finite element solutions in all cases. Not surprisingly, MG requires fewer iterations to reach the stopping criteria than GMRES, especially when our stopping criteria is used. The total amount of work of MG with our stopping criteria is about half of the amount of work of MG with the heuristic stopping criterion. However, no such saving can be seen from GMRES. Our numerical results indicate MG iterative methods with the stopping criteria in previous sections are the method of choice if fast and reliable iterative solutions are expected in the adaptive refinement process.

Problem 1 with VR error estimator:

ϵ	$\frac{\max_{T\in\Im_h}\eta_T}{\max_{T_p\in\Im_{h_p}}\eta_{T_p}}$ on refined meshes
10^{-2}	0.481 0.483 0.459 0.412
10^{-3}	0.487 0.509 0.490 0.491 0.482 0.469 0.429
10^{-4}	0.485 0.532 0.476 0.505 0.493 0.495 0.491 0.485

Table 5.1: Verification of the assumption (5.22) of the new stopping criteria

ϵ	Tol	Iterations
10^{-2}	S_1	9 11 10 9
	S_2	2 3 2 1
10^{-3}	S_1	10 15 15 15 11 9 8
	S_2	3 5 4 4 2 1 1
10^{-4}	S_1	10 16 17 21 21 17 13 10
	S_2	4 7 6 9 7 5 2 1

(a) MG iteration steps

ϵ	Tol	Iterations
10^{-2}	S_1	9 11 14 20
	S_2	25 26 27 30
10^{-3}	S_1	10 12 14 19 20 23 26
	S_2	25 27 29 30 29 31 30
10^{-4}	S_1	10 12 15 19 24 23 26 26
	S_2	25 27 28 30 33 33 33 31

(b) GMRES iteration steps

Table 5.2: Comparison of iteration counts for different stopping criteria

ϵ	Tolerance	Node number
10^{-2}	S_0, S_1, S_2	47 102 218 442
10^{-3}	S_0, S_1	47 102 220 464 940 1879 3736
	S_2	47 102 220 464 941 1880 3737
10^{-4}	S_0, S_1	47 102 221 474 980 1950 3835 7582
	S_2	47 102 221 473 976 1949 3834 7561

Table 5.3: Comparison of number of nodes of refined meshes from MG solutions

Problem 1 with KS error estimator:

ϵ	$\frac{\max_{T\in\Im_h}\eta_T}{\max_{T_p\in\Im_{h_p}}\eta_{T_p}}$ on refined meshes
10^{-2}	0.481 0.483 0.459 0.413
10^{-3}	0.487 0.509 0.489 0.491 0.483 0.469 0.428
10^{-4}	0.485 0.532 0.477 0.505 0.492 0.495 0.492 0.485

Table 5.4: Verification of the assumption (5.66) of the new stopping criteria

ϵ	Tol	Iterations
10^{-2}	S_1	8 11 10 9
	S_2	3 4 4 3
10^{-3}	S_1	10 15 15 15 11 9 8
	S_2	4 7 7 5 4 3 2
10^{-4}	S_1	10 16 17 21 21 17 13 10
	S_2	5 9 9 12 9 7 4 3

(a) MG iteration steps

ϵ	Tol	Iterations
10^{-2}	S_1	9 11 14 20
	S_2	25 26 27 29
10^{-3}	S_1	10 12 14 19 20 23 26
	S_2	25 27 28 30 30 29 28
10^{-4}	S_1	10 12 15 19 24 23 26 26
	S_2	25 27 28 31 33 33 33 32

(b) GMRES iteration steps

Table 5.5: Comparison of iteration steps for different stopping criteria

ϵ	Tolerance	Node number
10^{-2}	S_0, S_1, S_2	47 102 218 442
10^{-3}	S_0, S_1, S_2	47 102 220 464 940 1879 3736
	S_2	47 102 220 464 944 1883 3740
10^{-4}	S_0, S_1	47 102 221 474 980 1950 3835 7582
	S_2	47 102 221 474 980 1951 3836 7575

Table 5.6: Comparison of number of nodes of refined meshes from MG solutions

Problem 2 with VR error estimator:

ϵ	$\frac{\max_{T\in\Im_h}\eta_T}{\max_{T_p\in\Im_{hp}}\eta_{T_p}}$ on refined meshes
10^{-2}	0.5 0.5 0.5 0.5
10^{-3}	0.5 0.5 0.5 0.5 0.5 0.5 0.5
10^{-4}	0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5

Table 5.7: Verification of the assumption (5.22) of the new stopping criteria

ϵ	Tol	Iterations
10^{-2}	S_1	12 10 9 12
	S_2	3 3 3 6
10^{-3}	S_1	16 13 11 9 8 8 15
	S_2	4 3 3 3 2 3 11
10^{-4}	S_1	16 14 12 10 9 8 8 8
	S_2	6 4 4 3 3 2 2 2

(a) MG iteration steps

ϵ	Tol	Iterations
10^{-2}	S_1	11 12 13 17
	S_2	26 26 26 30
10^{-3}	S_1	11 12 12 12 13 15 28
	S_2	26 26 26 26 27 27 35
10^{-4}	S_1	11 12 12 12 13 15 16 17
	S_2	26 26 26 26 27 27 28 28

(b) GMRES iteration steps

Table 5.8: Comparison of iteration steps for different stopping criteria

ϵ	Tolerance	Node number
10^{-2}	S_0, S_1, S_2	50 97 190 394
10^{-3}	S_0, S_1	50 91 174 343 697 1350 2702
	S_2	50 91 174 343 683 1359 2705
10^{-4}	S_0, S_1	50 91 174 343 679 1346 2674 5331
	S_2	50 91 174 343 679 1346 2688 5369

Table 5.9: Comparison of number of nodes of refined meshes from MG solutions

Problem 2 with KS error estimator:

ϵ	$\frac{\max_{T\in\Im_h}\eta_T}{\max_{T_p\in\Im_{h_p}}\eta_{T_p}}$ on refined meshes
10^{-2}	0.5 0.5 0.5 0.499
10^{-3}	0.5 0.5 0.5 0.5 0.5 0.5 0.5
10^{-4}	0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5

Table 5.10: Verification of the assumption (5.66) of the new stopping criteria

ϵ	Tol	Iterations
10^{-2}	S_1	12 10 9 12
	S_2	4 4 4 7
10^{-3}	S_1	16 13 11 9 8 8 15
	S_2	7 4 4 4 3 4 13
10^{-4}	S_1	16 14 12 10 9 8 8 8
	S_2	9 6 5 5 4 3 3 3

(a) MG iteration steps

ϵ	Tol	Iterations
10^{-2}	S_1	11 12 13 17
	S_2	26 26 26 30
10^{-3}	S_1	11 12 12 12 13 15 28
	S_2	26 26 26 26 27 27 35
10^{-4}	S_1	11 12 12 12 13 15 16 17
	S_2	26 26 26 26 27 27 28 28

(b) GMRES iteration steps

Table 5.11: Comparison of iteration steps for different stopping criteria

ϵ	Tolerance	Node number
10^{-2}	S_0, S_1, S_2	50 97 190 394
10^{-3}	S_0, S_1	50 91 174 343 697 1350 2702
	S_2	50 91 174 343 683 1359 2702
10^{-4}	S_0, S_1	50 91 174 343 679 1346 2674 5331
	S_2	50 91 174 343 679 1346 2688 5334

Table 5.12: Comparison of number of nodes of refined meshes from MG solutions

Problem 4 with VR error estimator:

ϵ	$\frac{\max_{T\in\Im_h}\eta_T}{\max_{T_p\in\Im_{h_p}}\eta_{T_p}}$ on refined meshes
10^{-2}	0.35 0.29 0.26 0.26
10^{-3}	0.35 0.29 0.75 0.65 0.84 0.65 0.31
10^{-4}	0.35 0.29 0.48 1.84 0.77 0.78 0.69 0.38

Table 5.13: Verification of the assumption (5.22) of the new stopping criteria

ϵ	Tol	Iterations
10^{-2}	S_1	19 10 8 5
	S_2	3 3 3 2
10^{-3}	S_1	36 25 21 12 9 11 10
	S_2	6 8 9 4 3 5 4
10^{-4}	S_1	36 44 28 19 18 19 19 16
	S_2	11 14 14 9 6 5 6 6

(a) MG iteration steps

ϵ	Tol	Iterations
10^{-2}	S_1	16 15 32 35
	S_2	27 30 32 36
10^{-3}	S_1	28 35 41 41 48 53 57
	S_2	29 35 44 41 46 55 58
10^{-4}	S_1	29 37 48 39 35 44 67 76
	S_2	29 37 48 38 35 44 67 76

(b) GMRES iteration steps

Table 5.14: Comparison of iteration steps for different stopping criteria

ϵ	Tolerance	Node number
10^{-2}	S_0, S_1	72 167 415 1134
	S_2	72 171 423 1138
10^{-3}	S_0, S_1	73 197 453 699 1113 1754 2815
	S_2	73 197 459 705 1131 1779 2839
10^{-4}	S_0, S_1	73 205 459 790 1154 1785 2753 4144
	S_2	73 205 457 787 1148 1783 2728 4119

Table 5.15: Comparison of number of nodes of refined meshes from MG solutions

Problem 4 with KS error estimator:

ϵ	$\frac{\max_{T\in\Im_h}\eta_T}{\max_{T_p\in\Im_{h_p}}\eta_{T_p}}$ on refined meshes
10^{-2}	0.34 0.30 0.36 0.24
10^{-3}	0.34 0.37 0.51 0.68 0.75 0.44 0.17
10^{-4}	0.34 0.31 0.33 0.24 0.71 0.54 0.82 0.52

Table 5.16: Verification of the assumption (5.66) of the new stopping criteria

ϵ	Tol	Iterations
10^{-2}	S_1	20 10 8 6
	S_2	6 5 4 3
10^{-3}	S_1	41 21 16 14 18 18 10
	S_2	13 11 10 8 11 10 6
10^{-4}	S_1	52 27 22 24 17 15 15 25
	S_2	22 17 16 16 9 9 11 21

(a) MG iteration steps

ϵ	Tol	Iterations
10^{-2}	S_1	15 23 20 36
	S_2	27 30 33 36
10^{-3}	S_1	28 34 37 39 45 54 53
	S_2	28 34 37 39 45 54 53
10^{-4}	S_1	28 24 39 32 35 42 56 69
	S_2	28 35 39 32 35 42 55 69

(b) GMRES iteration steps

Table 5.17: Comparison of iteration steps for different stopping criteria

ϵ	Tolerance	Node number
10^{-2}	S_0, S_1, S_2	70 168 390 911
10^{-3}	S_0, S_1	70 176 345 592 948 1458 2391
	S_2	70 176 345 592 948 1458 2391
10^{-4}	S_0, S_1	70 176 354 764 1143 1752 2674 4093
	S_2	70 176 354 764 1143 1750 2688 4082

Table 5.18: Comparison of number of nodes of refined meshes from MG solutions

Chapter 6

Conclusions, Summary and Future Research

In this thesis, we concentrate on finding an accurate and efficient solver for solving the convection-diffusion equations. To achieve this goal involves accurate discretization methods, regularity estimates, a priori error estimations, reliable a posteriori error estimations and fast linear solvers. In this work, we have found that an accurate approximate solution of the convection-diffusion equation can be obtained by SDFEM discretization on adaptive refinement meshes. In this scenario, the question of how to compute the approximate solution that satisfies a pre-described accuracy efficiently can be broken into the following three questions:

1. How reliable is the a posteriori error estimation?

2. How fast and accurate can one refine the meshes to resolve boundary and interior layers?

3. What is the most efficient linear solver under the adapted refined meshes?

Our studies do not answer the first question and only show that the Kay and Silvester's a posteriori error estimation is more reliable than the Verfürth's error estimation. For question 2, with a carefully chosen error-adaptive sensitivity parameter, our error-

adapted mesh refinement strategy can be a remedy in case regular refinement fails to resolve the sharp gradient layers of the solution. Finally, the multigrid method combined with our stopping criteria seems to be a promising answer to the second and third questions. We summarize our results in the following.

In Chapter 2, we study the well known Galerkin discretization method (GK) and the streamline upwinding finite element discretization method (SDFEM). For both methods, the existence of the approximate solution and the a priori error estimation between the approximation solution u_h and the exact solution u are proved. Our numerical results in Section 2.3 show that SDFEM produces more accurate solutions. Furthermore, the error $|||u - u_h|||$ decreases in the order of $O(h^{1/2})$ is observed and suggests that the a priori error estimation (2.38) in terms of $\|\nabla u\|_0$ may provide a better error bound. The theoretical impact from this observation is reflected on the proof of our multigrid convergence result, Theorem 4.3.4.

In Chapter 3, we study the a posteriori error estimations including the residual type of error estimation (VR) proposed by Verfürth and the Neumann-type of error estimation (KS) proposed by Kay and Silvester. Our numerical results in Section 3.3 shows that the KS error estimation is more reliable than the VR error estimation. In addition, the local lower bounds of both error estimations are sharp and can be considered as efficient error indicators to pinpoint where the exact error is large. In order to increase the accuracy of the approximate solution, we use the KS indicator to refine meshes and move grid points to where the value of KS error indicator is large. First, our numerical results in Section 3.4 show that a simple moving mesh strategy, Algorithm 3.4.1, is able to increase the solution accuracy. However, drawbacks of the moving

mesh strategy include the necessity of a carefully chosen relaxation parameter and expensive computation overhead if fast multigrid linear solvers are desired. Second, the regular refinement strategy either requires too many refinement steps or generates too many grid points to resolve layers, or even fails to resolve the layer when the convection term is strongly dominant. We introduce an error-adapted mesh refinement strategy in Section 3.5 to overcome these difficulties. The meshes generated from the error-adapted refinement strategy are nested and can be directly used in multigrid solvers. Moreover, our numerical results show that the error-adapted refinement strategy generates significantly fewer nodes than regular refinement strategy and is capable of quickly resolving the boundary layer.

In Chapter 4, first, we prove the convergence of horizontal line Gauss-Seidel method (HGS) for the convection-diffusion problem with vertical wind (Problem 2). Theorem 4.1.3 shows the error reduction factor of HGS is proportional to $O(\frac{\epsilon}{h^2})$ for $h \gg \sqrt{\epsilon}$. In asymptotical limit $\epsilon \to 0$, HGS is the exact solver. This suggests that HGS is a good smoother if multigrid method is employed to solve the sparse linear system of Problem 2. Moreover, since, HGS is a convergent iterative method, HGS may as well be a good preconditioner for the GMRES method. Second, in Theorem 4.3.2 and Remark 4.3.3, we show that HGS satisfies the usual $smoothing\ property$ (4.26). The convergence of the V-cycle multigrid with HS smoother is then proved in Theorem 4.3.4 by utilizing the $smoothing\ property$, the a priori error estimate and the regularity estimates. Moreover, we conclude that MG converges faster than HGS for Problem 2, since the MG convergence factor is $O(\frac{\epsilon}{h^{3/2}})$ as stated in Remark 4.3.5. The numerical results in Section 4.1 and Section 4.3 support our theoretical analysis. Finally, in the search of a fast linear solver for the convection-diffusion equations, our

numerical studies in Section 4.5 show that GMRES with multigrid preconditioner is the best choice among the linear solvers: standard multigrid (MG), algebraic multigrid (AMG), GMRES, GMRES with Gauss-Seidel preconditioner and GMRES with AMG preconditioner. Here, we like to note that MG with Gauss-Seidel smoother can as well be a fast solver for the convection diffusion problems on adaptive mesh when using the stopping criteria we propose in Chapter 5.

In Chapter 5, we give two stopping criteria for the iterative linear solvers. The error indicator computed from iterative solutions satisfying the stopping criteria in Theorem 5.1.6 and Theorem 5.2.6 will generate a mesh similar to the mesh generated by the error indicator computed from exact solution. Furthermore, if the iterative solutions satisfy the stopping criteria in Theorem 5.1.5 and Theorem 5.2.5, then the error between iterative solution and exact solution is bounded below by the upper bound in the a posteriori error estimation. If the upper bound of the a posteriori error estimation is optimal, then one can not distinguish the exact solution and iterative solution in the sense of measuring the true error. we suggest that the stopping criteria in Theorem 5.1.5 and Theorem 5.2.5 only need to be verified at the finest mesh where a reliable solution is expected. For the purpose of accelerating the mesh refinement process and avoiding refinement over wrong locations, a linear solver which can more quickly satisfy our stopping criteria is preferred. Our numerical studies in Section 5.3 indicate that MG with Gauss-Seidel smoother requires fewer iterative steps to satisfy our stopping criteria than to satisfy the heuristic stopping tolerance, residual less than 10^{-6}. However, no such savings is seen if GMRES is used to solve the linear system.

It is important to realize that different discretization schemes directly affect the fundamental property of the discrete matrix and the error estimations. A good property of the discrete matrix such as M-matrix is a foundation of choosing and developing fast and stable linear solvers. Recently, Xu and Zikatanov propose a new edge-averaged finite element discretization scheme (EAFE) [103] which guarantees the resulting discrete matrix is an M-matrix. A multigrid linear solver based on EAFE and graph matching is proposed in [61]. It will be our interests to know a posteriori error estimations for this discretization scheme and see how different linear solvers perform for the linear systems arising from EAFE. Moreover, since anisotropic meshes are generally generated for real applications in computational fluid dynamics and our error-adapted refinement strategy also tends to produce anisotropic meshes in boundary layer regions, the a posteriori error estimation for the convection-diffusion equation on anisotropic meshes, such as the error estimation by Kunert [63], are topics of our future work. We also wish to explore how iterative solvers, in particular multigrid methods, perform for the anisotropic meshes generated from refinement process. To search fast linear solvers for solving more difficult problems such as the Navier-Stokes equations will always be our long-term goals. Hopefully, we can find stopping criteria for these iteration methods and apply the error-adapted mesh refinement strategy to these problems.

Bibliography

[1] S. Adjerid and J. E. Flaherty. A moving-mesh finite element method with local refinement for parabolic partial differential equations. *Comput. Meth. Appl. Mech. Engrg.*, 55:3–26, 1986.

[2] M. Ainsworth and I. Babuška. Reliable and roubst a posteriori error estimation for singular perturbed reaction-diffusion problems. *SIAM J. Numer. Anal.*, 36:331–353, 1999.

[3] R. C. Almeida, R. A. Feijóo, A. C. Gale ao, C. Padra, and R. S. Silva. Adaptive finite element computational fluid dynamics using an anisotropic error estimator. *Comput. Methods Appl. MEch. Engrg.*, 182:379–400, 2000.

[4] T. Apel and G. Lube. Anisotropic mesh refinement in stabilized galerkin methods. *Numer. Math.*, 74:261–282, 1996.

[5] O. Axelsson. Conjugate gradient type methods for unsymmetric and inconsistent systems of linear equations. *Lin. Alg. Math.*, 29:1–16, 1980.

[6] I. Babuška and A. K. Aziz. On the angle condition in the finite element method. *SIAM J. Numer. Anal.*, 13:214–226, 1976.

[7] I. Babuška, R. Durán, and R. Rodríguez. Analysis of the efficiency of an a posteriori error estimator for linear triangular finite elements. *SIAM J. Numer. Anal.*, 29:947–964, 1992.

[8] I. Babuška and W. C. Rheinboldt. Error estimates for adaptive finite element computations. *SIAM J. Numer. Anal.*, 15:736–754, 1978.

[9] M. J. Baines. Grid adaptation via node movement. *Appl. Numer. Math.*, 26:77–96, 1998.

[10] R. E. Bank and R. K. Smith. Mesh smoothing using a posteriori error estimates. *SIAM J. Numer. Anal.*, 34:979–997, 1997.

[11] R. E. Bank and A. Weiser. Some a posteriori error estimators for elliptic partial differential equations. *Math. Comp.*, 44:283–301, 1985.

[12] M. De. Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf. *Computational Geometry*. Springer-Verlag Berling Heidelberg New York, 1997.

[13] J. Bey. On the convergence of basic iterative methods for convection-diffusion equations. *Numer. Linear. Algebra. Appl.*, 1:1–7, 1993.

[14] J. Bey and G. Wittum. Downwind numbering: A robust multigrid method for convection diffusion problems on unstructured grids. *Appl. Numer. Math.*, 23:177–192, 1997.

[15] H. Borochaki and P. L. George. Aspects of 2-D Delaunay generation. *Int. J. Numer. Methods Eng.*, 40:1957–1975, 1997.

[16] D. Braess and W. Hackbusch. A new convergence proof for the multigrid methods including the v-cycle. *SIAM J. Numer. Anal.*, 20:967–975, 1983.

[17] J. H. Bramble, D. A. Kwak, and J. E. Pasciak. Uniform convergence of multigrid V-cycle iterations for indefinite and nonsymmetric problems. *SIAM J. Numer. Anal.*, 31:1746–1763, 1994.

[18] J. H. Bramble and J. E. Pasciak. New estimates for multilevel algorithms including V-cycle. *Math. Comp.*, 60:447–471, 1993.

[19] J. H. Bramble, J. E. Pasciak, J. Wang, and J. Xu. Convergence estimates for multigrid algorithms without regularity assumptions. *Math. Comp.*, 57:23–45, 1991.

[20] J. H. Bramble, J. E. Pasciak, and J. Xu. The analysis of multigrid algorithms for nonsymmetric and indefinite elliptic problems. *Math. Comp.*, 51:398–414, 1988.

[21] S. C. Brenner and L. R. Scott. *The Mathematical Theory of Finite Element Methods*. Springer-Verlag, NewYork, 1994.

[22] A. Brooks and T. Hughes. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Naiver-Stokes equations. *Comput. Methods Appl Mech. Engrg.*, 32:199–259, 1982.

[23] G. C. Buscaglia and E. A. Dari. Anisotropic mesh optimization and its application in adaptivity. *Int. J. Numer. Methods Engrg.*, 40:4119–4136, 1997.

[24] W. Cao, W. Huang, and R. D. Russell. An error indicator monitor function for an r-adaptive finite-element method. *J. Comput. Physics*, 170:871–892, 2001.

[25] N. N. Carlson and K. Miller. Design and application of a gradient-weighted moving finite element code I : in one dimension. *SIAM J. Sci. comput.*, 19:728–765, 1998.

[26] N. N. Carlson and K. Miller. Design and application of a gradient-weighted moving finite element code II : in two dimension. *SIAM J. Sci. comput.*, 19:766–798, 1998.

[27] M. J. Castro-Diaz, F. Hecht, B. Mohammadi, and O. Pironneau. Anisotropic unstructure mesh adaption for flow simulation. *Int. J. Numer. Methods in Fluids*, 25:475–491, 1997.

[28] E. F. D'Azevedo and R. B. Simpson. On optimal triangular meshes for minimizing the gradient error. *Numer. Math.*, 59:321–348, 1991.

[29] J. E. Dendy. Blackbox multigrid for nonsymmetric problems. *Appl. Math. Comput.*, 13:261–283, 1983.

[30] Vit Dolejší. Anisotropic mesh adaptation for finite volumn and finite element methods on triangular meshes. *Comput. Visual Sci*, 1:165–178, 1998.

[31] J. Dompierre, P. Labbé, J. Trépanier, and F. Fortin. Local remeshing techiniques around geometries for store release. Technical report, Creca, 03 2000.

[32] W. Dörfler and R. H. Nochetto. Small data oscillation implies the saturation assumption. *Numer. Math.*, 91:1–12, 2002.

[33] R. Durán, M. A. Muschietti, and R. Rodríguez. On the asymptotic exactness of error estimators for linear triangular finite elements. *Numer. Math.*, 59:107–127, 1991.

[34] R. Durán and R. Rodríguez. On the asymptotic exactness of Bank-Weiser's estimator. *Numer. Math.*, 62:297–303, 1992.

[35] S. C. Eisenstat, H. C. Elman, and M. H. Schultz. Variational iterative methods for nonsymmetric systems of linear equations. *SIAM J. Numer. Anal.*, 20:345–357, 1983.

[36] H. C. Elman. *Large-Scale Matrix Problems and the Numerical Solution of Partial Differential Equations*. Oxford, 1994.

[37] H. C. Elman and M. P. Chernesky. Ordering effects on relaxation methods applied to the discrete one-dimensional convection-diffusion equation. *SIAM J. Numer. Anal.*, 30:1268–1290, 1993.

[38] H. C. Elman and M. P. Chernesky. Ordering effects on relaxation methods applied to the discrete convection-diffusion equation. *Recent advances in iterative methods, Springer Berlin*, pages 45–57, 1994.

[39] H. C. Elman, D. J. Silvester, and A. J. Wathen. Finite Elements and Fast Iterative Solvers. preprint, 2003.

[40] K. Eriksson and C. Johnson. Adaptive streamline diffusion finite element methods for stationary convection-diffusion problems. *Math. Comp.*, 60:167–188, 1993.

[41] Oliver G. Ernst. Residual-minimizing Krylov subspace methods for stabilized discretizations of convection-diffusion equations. *SIAM J. Matrix Anal.Appl.*, 21:1079–1101, 2000.

[42] R. P. Fedorenko. A relaxation method for solving elliptic difference equations. *USSR Comput. Math. Phys.*, 1:1092–1096, 1961.

[43] C. A. Felippa. Optimization of finite element grids by direct energy search. *Appl. Math. Modelling*, 1:239–244, 1977.

[44] B. Fischer, A. Ramage, D. Silvester, and A. J. Wathen. On parameter choice and iterative convergence for stabilised discretisations of advection-diffusion problems. *Comput. Methods Appl. Mech. Engrg.*, 179, 1999.

[45] W. H. Frey and D. A. Field. Mesh relaxation: A new technique for improving triangulations. *Int. J. Numer. Methods Eng.*, 31:1121–1133, 1991.

[46] E. J. Gaspar, C. Clavero, and F. Lisbona. Some numerical experiments with multigrid methods on shishkin meshes. *J. Comput. Appl. Math*, 138(1):21–35, 2002.

[47] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Johns Hopkins, 1996.

[48] P. Grisvard. *Elliptic Problems in Nonsmooth Domains*. Pitman, Boston, 1985.

[49] W. Hackbusch. *Multi-grid methods and applications*. Comput. Math. Springer-Verlag (Berlin, Heidelberg, New York), 1985.

[50] W. Hackbusch. *Iterative Solution of Large Sparse Systems of Equations*. Springer Verlag New York, 1994.

[51] W. Hackbusch and T. Probst. Downwind Gauss-Seidel smoothing for convection dominated problems. *Numer. Linear Algebra Appl.*, 4:85–102, 1997.

[52] W. Huang, Y. Ren, and R. D. Russell. Moving mesh partial differential equations (MMPDES) based on the equidistribution principle. *SIAM J. Numer. Anal.*, 31:709–730, 1994.

[53] M. E. Hubbard and M. J. Baines. Multidimensional upwinding and grid adaptation. *Numerical Methods for Fluid Dynamics, Oxford Univ. Press*, V:431–438, 1995.

[54] T. Hughes and A. Brooks. A multidimensional upwind scheme with no crosswind diffusion. *Finite Element Methods for Convection Dominated Folws, AMSE, New York*, 34, 1979.

[55] T. J. R. Hughes, M Mallet, and A. Mizukami. A new finite element formulation for computational fluid dynamics: II. Beyond SUPG. *Comput. Methods Appl. Mech. Engrg.*, 54:485–501, 1986.

[56] C. Johnson. *Numerical Solution of Partial Differential Equations by the Finite Element Method*. Cambridge University Press, New York, 1987.

[57] C. Johnson. The streamline diffusion finite element method for compressible and incompressible fluid flow. *Finite elements in fluids*, 8:75–95, 1989.

[58] C. Johnson, A. H. Schatz, and L. B. Wahlbin. Crosswind smear and pointwise errors in streamline diffusion finite element methods. *Math. Comput.*, 49:25–38, 1987.

[59] D. Kay and D. Silvester. The reliability of local error estimators for convection-diffusion equations. *IMA. Journal of Num. Anal.*, 21:107–122, 2001.

[60] A. Khamayseh and A. Kuprat. Anisotropic smoothing and solution adaption for unstructured grids. *Int. J. Numer. Methods Eng.*, 39:3163–3174, 1996.

[61] H. Kim, J. Xu, and L. Zikatanov. A monotone finite element scheme for convection diffusion equation. *Numer. Linear Algebra Appl.*, 10:181–195, 2003.

[62] A. Krechel and K. Stüben. Operator dependent interpolation in algebraic multigrid. multigrid methods v. *Lect. Notes Comput. Sci. Eng.*, 3:189–211, 1998.

[63] J. Kunert, R. J. Martin, and S. Pooladsaz. A posteriori error estimation for the convection dominated problems on anisotropic meshes. *Math. Methods Appl. Sci.*, 07:589–617, 2003.

[64] C. L. Lawson. *Software for C^1 interpolation*. J. R. Rice, ed. Mathematical Software III, Academic Press, New York, 1977.

[65] T. Linß and M. Stynes. The SDFEM on Shishkin Meshes for linear convection-diffusion problems. *Numer. Math.*, 87:457–484, 2001.

[66] K. Miller and R. Miller. Moving finite Eelements, Part I. *SIAM J. Numer. Anal.*, 18:1019–1032, 1981.

[67] K. Miller and R. Miller. Moving finite elements, Part II. *SIAM J. Numer. Anal.*, 18:1033–1057, 1981.

[68] P. Morin, R. H. Nochetto, and G. K. Siebert. Data oscillation and convergence of adaptive FEM. *SIAM J. Numer. Anal.*, 38:466–488, 2000.

[69] K. Niijima. Pointwise error estimates for a streamline diffusion finite element scheme. *Numer. Math.*, 56:707–719, 1990.

[70] C. W. Osterlee and T. Washio. An evaluation of parallel multigrid as a solver and a preconditioner for singularly perturbed problems. *SIAM J. Sci. Comput.*, 19:87–110, 1998.

[71] C. C. Paige and M. A. Saunders. Solution of sparse indefinite systems of linear equations. *SIAM J. Numer. Anal.*, 12:617–624, 1975.

[72] A. Papastavrou and R. Verfürth. A posteriori error estimators for stationary convection-diffusion problems: a computational comparison. *Comput. Methods Appl. Mech. Engrg.*, 189:449–462, 2000.

[73] J. Peraire, J. Peiró, and K. Morgan. Adaptive remeshing for three-dimentional compressible flow computations. *J. Comp. Physics*, 103:269–285, 1992.

[74] I. Persson, K. Samuelsson, and A. Szepessy. On the convergence of multigrid methods for flow problems. *Electron. Trans. Numer. Anal.*, 8:46–87, 1999.

[75] C. Pflaum. Robust convergence of multilevel algorithms for convection-diffusion equations. *SIAM J. Numer. Anal.*, 37:443–469, 2000.

[76] W. Rachowicz. An anisotropic h-type mesh refinement strategy. *Comput. Methods. Appl. Mech. Engrg.*, 109:169–181, 1993.

[77] A. Ramage. A multigrid preconditioner for stabilised discretizations of advection-diffusion problems. *J. Comput. Appl. Math.*, 110:187–203, 1999.

[78] A. Reusken. Multigrid with matrix-dependent transfer operators for a singular perturbation problem. *Computing*, 1994.

[79] A. Reusken. *Multigrid with matrix-dependent transfer operators for convection-diffusion problems*. Seventh International Symposium on Domain Decomposition Methods for Partial Differential Equation. Birkhäuser, Basel, 1994.

[80] A. Reusken. Convergence analysis of a multigrid method for convection-diffusion equations. *Numer. Math.*, 91:323–349, 2002.

[81] S. Rippa. Long and thin triangles can be good for linear interpolation. *SIAM J. Numer. Anal.*, 29:257–270, 1992.

[82] S. Rippa and B. Schiff. Minimum energy triangulations for elliptic problems. *Comput. Methods in Appl. Mech. Engrg.*, 84:257–274, 1990.

[83] M. C. Rivara. Algorithms for refining triangular grids suitable for adaptive and multigrid techniques. *Int. J. Numer. Methods Eng.*, 20:745–756, 1984.

[84] M. C. Rivara. Using longest-side bisection techniques for the automatic refinement of Delaunay triangulation. *Int. J. Numer. Methods Eng.*, 40:581–597, 1997.

[85] H. G. Roos, M. Stynes, and L. Tobiska. *Numerical Methods for Singularly Perturbed Differential Equations*. Springer-Verlag, New York, 1996.

[86] J. W. Ruge and K. Stüben. Algebraic Multigrid (AMG). Multigrid Methods. *SIAM, Frontiers in Appl. Math. Philadelphia.*, 5, 1985.

[87] Y. Saad and M. H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7:856–869, 1986.

[88] Y. T. Shih and H. C. Elman. Modified streamline diffusion schemes for convection-diffusion problems. *Comput. Methods. Appl. Mech. Engrg.*, 174:137–151, 1999.

[89] Y. T. Shih and H. C. Elman. Iterative methods for stabilized discrete convection-diffusion problems. *IMA J. Numer. Anal.*, 20:333–358, 2000.

[90] Yin-Tzer Shih. *Upwind Finite Element Solutions for Convection-Diffusion Problems*. PhD thesis, University of Maryland at College Park, Department of Mathematics, 1998.

[91] R. M. Smith and A. G. Hutton. The numerical treatment of advection - a performance comparison of current methods. *Numer. Heat Transfer*, 5:439–461, 1982.

[92] K. Stüben. Algebraic multigrid (AMG) experiences and comparisons. *Appl. Math. and Comp.*, 13:419–451, 1983.

[93] Y. Tourigny and M. J. Baines. Analysis of an algorithm for generating locally optimal meshes for l_2 approximation by discontinuous piecewise polynomials. *Math. of Comp.*, 66:623–650, 1997.

[94] Y. Tourigny and F. Hülsemann. A new moving mesh algorithm for the finite element solution of variational problems. *SIAM J. Numer. Anal.*, 35:1416–1438, 1998.

[95] R. S. Varga. *Matrix Iterative Analysis*. Springer-Verlag Berlin Heidelberg, 2000.

[96] R. Verfürth. A posteriori error estimation and adaptive mesh-refinement techniques. *J. Comput. Appl. Math.*, 50:67–83, 1994.

[97] R. Verfürth. A posteriori error estimators for convection-diffusion equations. *Numer. Math.*, 80:641–663, 1998.

[98] C. Wagner, Kinzelback W, and G. Wittum. Schur-Complement multigrid - a robust method for groundwater flow and transport problems. *Numer. Math.*, 75:523–545, 1997.

[99] J. Wang. Convergence analysis without regularity assumptions for multigrid algorithms based on sor smoothing. *SIAM J. Number.*, 29:987–1001, 1992.

[100] P. Wesseling. Theoretical and pratical aspects of a multigrid method. *SIAM J. Sci. Statist. Comput.*, 3:387–407, 1982.

[101] L. B. William. *A multigrid tutorial*. SIAM, 1987.

[102] G. Wittum. On the robustness of ILU smoothing. *SIAM J. Sci. Stat. Comput.*, 10:699–717, 1989.

[103] J. Xu and L. Zikatanov. A monotone finite element scheme for convection diffusion equation. *Math. Comp.*, 68:1429–1446, 1999.

[104] I. Yavneh, C. H. Venner, and A. Brandt. Fast multigrid solution of the advection problem with closed characteristics. *SIAM J. Sci. Comput*, 19:111–125, 1998.

[105] P.M. De Zeeuw. Matrix-dependent prolongations and restrictions in a blackbox multigrid solver. *J. Comp. and Appl. Math.*, 33:1–27, 1990.

[106] P.A. Zegeling. A dynamically moving adaptive grid method based on a smoothed equidistribution principle along coordinate lines. *5th International*

Conference on Numerical Grid Generation in Computational Field Simulation, Starksville, MSU, 1996.

[107] G. Zhou. How accurate is the streamline diffusion finite element method? *Math. Comp.,* 66:31–44, 1997.