Residual-Based Isotropic and Anisotropic Mesh Adaptation for Computational Fluid Dynamics

by

Amir R. Baserinia

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Mechanical Engineering

Waterloo, Ontario, Canada, 2008

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Amir R. Baserinia

Abstract

The accuracy of a fluid flow simulation depends not only on the numerical method used for discretizing the governing equations, but also on the distribution and topology of the mesh elements. Mesh adaptation is a technique for automatically modifying the mesh in order to improve the simulation accuracy in an attempt to reduce the manual work required for mesh generation. The conventional approach to mesh adaptation is based on a feature-based criterion that identifies the distinctive features in the flow field such as shock waves and boundary layers. Although this approach has proved to be simple and effective in many CFD applications, its implementation may require a lot of trial and error for determining the appropriate criterion in certain applications. An alternative approach to mesh adaptation is the residual-based approach in which the discretization error of the fluid flow quantities across the mesh faces is used to construct an adaptation criterion. Although this approach provides a general framework for developing robust mesh adaptation criteria, its incorporation leads to significant computational overhead.

The main objective of the thesis is to present a methodology for developing an appropriate mesh adaptation criterion for fluid flow problems that offers the simplicity of a feature-based criterion and the robustness of a residual-based criterion. This methodology is demonstrated in the context of a second-order accurate cell-centred finite volume method for simulating laminar steady incompressible flows of constant property fluids. In this methodology, the error of mass and momentum flows across the faces of each control volume are estimated with a Taylor series analysis. Then these face flow errors are used to construct the desired adaptation criteria for triangular isotropic meshes and quadrilateral anisotropic meshes. The adaptation results for the lid-driven cavity flow show that the solution error on the resulting adapted meshes is 80 to 90 percent lower than that of a uniform mesh with the same number of control volumes.

The advantage of the proposed mesh adaptation method is the capability to produce meshes that lead to more accurate solutions compared to those of the conventional methods with approximately the same amount of computational effort.

Acknowledgements

This thesis would not have been realized without the guidance, suggestions, criticisms, and support of a few people and organizations to whom I wish to express my gratitude.

I would like to thank my research supervisor, Professor Gordon Stubley, who proposed the topic of this work and supported me to do it during the past five years. However, Professor Stubley's role goes beyond the technical aspects of of this thesis. Professor Stubley taught me that the most important virtue of an engineer is not the technical knowledge, but the integrity in the incorporation of that knowledge. In this sense, he has been and will be my role model.

I would also like to thank my fellow graduate students, especially Andrea Scott, Brian McLaughlin, and Jonathan Jilesen for the conversations that we had and their suggestions. I would specifically like to thank Shahzma Jaffer for her support during the past year and the plenty of time that we spent together.

This work was funded by the Natural Sciences and Engineering Research Council of Canada, the University of Waterloo, and the Department of Mechanical and Mechatronics Engineering in the form scholarships, research assistantship, and teaching assistantships. Their financial support is gratefully acknowledged.

The numerical simulations of this work were made possible by the facilities of the Shared Hierarchical Academic Research Computing Network (SHARCNET).

I would also like to acknowledge my appreciation to the free software movement for creating many software tools without which this work would have been a less enjoyable pursuit.

Last but not least, I want to thank my family, especially my mom, for being supportive during the years that I have been far from them. Without their support and sacrifice it is hard to imagine that I would be where I am today.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Fluid flow phenomena occur in many scientific and engineering applications; understanding the interstellar flows, predicting the weather, and designing a gas turbine engine are just three examples of these applications. Therefore scientists and engineers need to study and understand the physics of fluid flows, which is the subject of fluid dynamics.

A significant breakthrough in fluid dynamics occurred in the second half of the twentieth century along with the advent of computers. Until the mid twentieth century, the majority of works on fluid dynamics were experimental. This approach was costly and time consuming. The advent of digital computers revolutionized fluid dynamics by enabling engineers to deal with fluid flows through numerical methods. This new approach, which is called computational fluid dynamics (CFD), became increasing popular in industry as a less costly and more accessible alternative to the traditional experimental approach. The trend continues today and CFD is becoming increasingly popular as a tool for fluid flow analysis and design [52]. However, the widespread use of CFD would not have been possible without progress in *mesh generation* technology.

Mesh generation is the process of discretizing the geometry of a physical problem. It has been an integral component of numerical methods for solving partial differential equations since the earliest applications. In the earliest applications, the emphasis was on the simulation of simple flow models in simple two-dimensional geometries using finite difference methods. In these situations, very simple mesh designs were adequate. Figure 1.1 exhibits an example, used by Richardson in his pioneering work in numerical

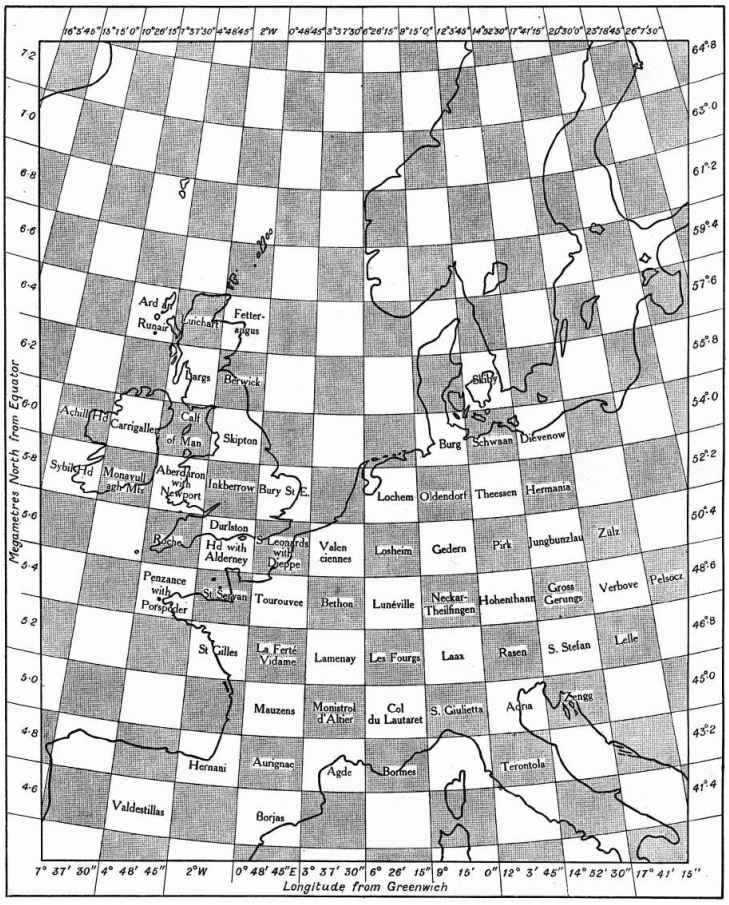


Figure 1.1: The mesh used by Richardson in numerical weather prediction [72]

meteorology [72]. With the desire to simulate more complex fluid flow models in more complicated geometries, mesh generation technology has had to evolve. This evolution continues today and CFD is the key driver in stimulating the development of high quality mesh generators [11].

Mesh generation is important since the stability and accuracy of a CFD simulation depend partially on the mesh. A mesh is called high quality if the simulation on it is accurate enough. In the earlier applications of CFD, the quality of meshes were judged through visual assessment [11]. In these situations, the CFD practitioner had to use

her or his expertise to incorporate fluid flow simulation concepts into mesh generation. In contrast in modern applications of CFD that involve complex flow fields in complicated geometries, the visual assessment is neither adequate nor feasible. Therefore *mesh adaptation* methods emerged in order to automate the mesh generation process.

This thesis is an attempt towards automating the process of mesh generation in CFD applications. But before delving into the details of mesh generation and discussing the specific objectives of the thesis, let us briefly review the process of fluid flow simulation.

1.1 Numerical Simulation: A Background

Numerical simulation is the process of solving a physical problem using numerical methods. Such a process consists of four steps: establishing a boundary value problem that models the problem, generating a mesh, discretizing the boundary value problem on the mesh, and solving the discretized equations. Let us explain each step in more detail.

1.1.1 Establishing a Boundary Value Problem

The first step in a fluid flow simulation is to establish a boundary value problem (BVP) that describes the physics of the flow field. The BVP consists of two components: the mathematical model and the boundary conditions.

The mathematical model is the set of partial differential equations that represent the behaviour of a fluid flow. These equations are referred to as the governing equations. In fluid dynamics, the governing equations are based on the conservation of mass, momentum, energy, and other species. For example, the governing equations for a steady-state incompressible isothermal flow of a constant-property Newtonian fluid in the absence of gravity are:

$$\nabla \cdot \mathbf{v} = 0 \tag{1.1}$$

$$\nabla \cdot (\rho\, \mathbf{v} \otimes \mathbf{v}) = -\nabla p + \nabla \cdot \left[\mu \left(\nabla \mathbf{v} + \nabla \mathbf{v}^T \right) \right] \tag{1.2}$$

where $\mathbf{v}$ is the flow velocity vector, p is the flow pressure, ρ is the fluid density, and μ is the fluid viscosity. The superscript T in Equation (1.2) represents the transpose of

the velocity gradient tensor, $\nabla \mathbf{v}$. Note that the solution of the set of Equations (1.1) and (1.2) is not unique unless we impose appropriate boundary conditions on them.

The boundary conditions are the additional equations that we impose on the mathematical model in order to obtain the solution of a certain flow field. For example, if we want to solve the set of Equations (1.1) and (1.2) on a given domain, Ω, it is necessary to prescribe the values or gradients of flow velocity and pressure at the domain boundary, $\partial\Omega$. Once the boundary conditions are prescribed, we can solve the BVP and obtain the *exact solution*.

Definition 1.1 *The exact solution of a flow field is the solution of the boundary value problem, associated with the flow field.*

The exact solution does not necessarily correspond to the physical solution. The difference between the two is called the *modelling error*, which is due to the discrepancy between the predictions of the mathematical model and the behaviour of the fluid flow in reality. However, the modelling error is not the emphasis of the present study and we assume that the mathematical model is representative of the physical model.

After establishing the BVP for a flow field, we can proceed to solve the problem. In the case of simple flow fields in simple geometries, we might be able to solve the BVP analytically and obtain a closed-form solution. However, analytical solutions are not available for most flows of practical interest. In these cases, we have to proceed to the second step of flow simulation and solve the BVP numerically.

1.1.2 Generating a Mesh

The second step in a fluid flow simulation is mesh generation. In mesh generation, we subdivide the physical domain, Ω, into simple polygons in $\mathbb{R}^2$ or polyhedrons in $\mathbb{R}^3$, called *cells*. We assume that cells satisfy the following conditions [42]:

- The union of all cells is equal to the physical domain, Ω.
- The overlap of any two cells is empty.
- The volume of each cell is non-zero and finite.

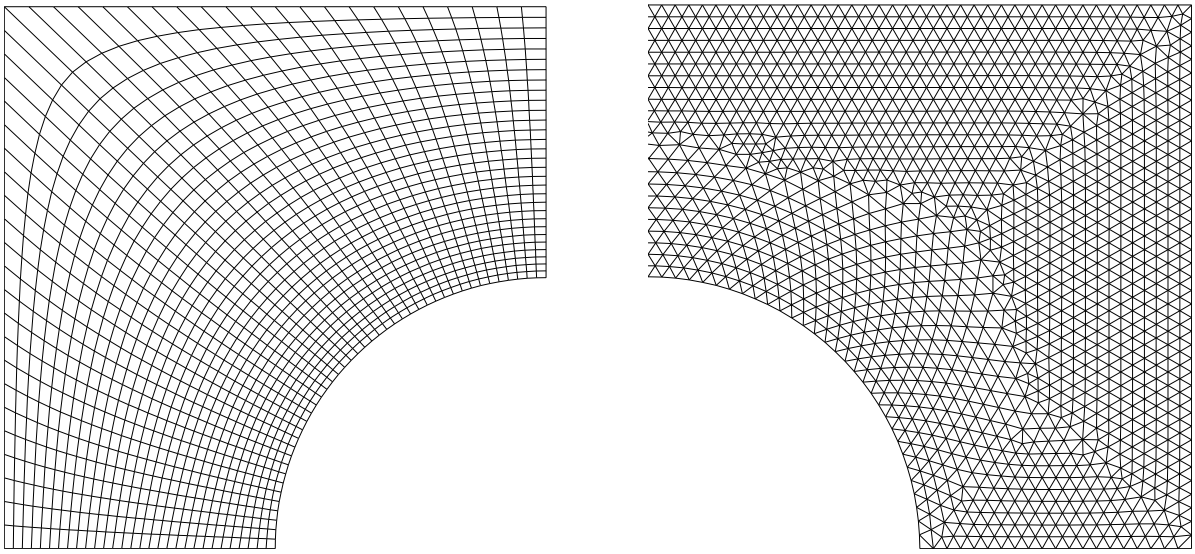


Figure 1.2: Examples of a 2-D structured mesh (left) and a 2-D triangular unstructured mesh (right) [17]

Figure 1.2 indicates two examples of meshes that satisfy the above conditions [17]. Note that the second condition in the above list excludes the case of *composite* meshes that have overlapping cells [84].

In numerical methods, meshes are generally categorized as structured or unstructured. A structured mesh is formed by two families of curves in $\mathbb{R}^2$ or three families of surfaces in $\mathbb{R}^3$ so that the members of a family do not intersect each other and the members of different families intersect each other only once [39]. In contrast, an unstructured mesh has no general definition and not surprisingly any mesh that is not structured is unstructured [42]. Figure 1.2 shows examples of structured and an unstructured meshes [17]. Note that this kind of categorization is more about the way that we implement the mesh on a computer than its appearance. For example, we can always handle a structured mesh as if it were unstructured. In this study, our emphasis is on unstructured meshes because of their capability of spanning complicated domains with limited user input and also their increasing popularity in the CFD community.

The generation of unstructured meshes can be performed using a few standard methods, discussed in the literature [42]. However, before generating the mesh we may have to answer a few questions. For example, how large the size of the mesh cells should be? Should all the cells be of the same size? Triangular cells or quadrilateral ones? In the case of triangular cells, equilateral triangles or right-angled ones? At this stage, we are not going to answer these questions. However, a brief literature review shows that triangular meshes with equilateral cells or quadrilateral meshes with orthogonal angles are usually preferred. Thus let us assume for the moment that our mesh is triangular with equilateral cells of the same size.

1.1.3 Discretizing the BVP on the Mesh

The third step in a fluid flow simulation is the discretization of the BVP on the mesh. In this step, we have to discretize both the governing equations and the boundary conditions. Although, there are various methods for discretizing a BVP, the preferred method in CFD applications is the finite volume method (FVM) [51, 60]. The reason for such a preference is that the finite volume method retains the conservation property in the discretized equations.

The basic idea in the finite volume method is to subdivide the domain into *control volumes* and then to apply the conservation equations to every single control volume. This process leads to a system of equations for the average value of the unknown solution variables within the control volumes. There are various ways to implement the above process. In particular, we must specify the arrangement of the control volumes on the mesh and also the way that we apply the conservation equations to them.

In a finite volume method, the arrangement of the control volumes may be *vertex-based* or *cell-based*, shown in Figure 1.3. In the vertex-based arrangement [12, 24, 59, 77], each control volume surrounds a mesh vertex and the control volume boundary is constructed by joining the centroids of the neighbouring cells and faces to each other. In contrast, in a cell-based arrangement [33, 67], the control volumes are the same as the mesh cells. In this study, we use the cell-based finite volume method because it is easier to implement. In addition, the cell-based arrangement facilitates the study of the effect of mesh geometry on the simulation quality.

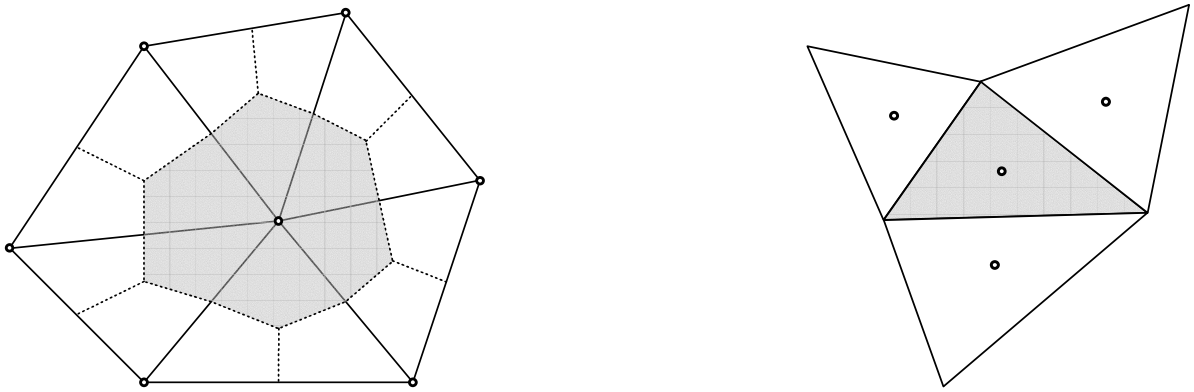


Figure 1.3: Schematics of vertex-based (left) and cell-based (right) arrangements in the finite volume method.

After specifying the arrangement of control volumes, we need to apply the conservation equations to them. For this purpose, we integrate the conservation equations on each control volume in order to obtain the integral conservation equations. These integral equations establish a relationship between the average value of the unknown solution variables within each control volume and the flow of those variables across the control volume faces. The unknown solution variables are stored at the centroid of the control volume, called the *node*. For calculating the flow across the control volume faces we have to interpolate the solution variables from its node to its faces. Although there are many interpolation schemes for this purpose, the most common schemes are the first and second order accurate interpolation schemes [39]. In this work, we use a second order accurate interpolation scheme.

The interpolation scheme in a finite volume method is based on the Taylor series expansion about the control volume nodes. For example, to calculate the flow of some variable, ϕ, across the face between two control volumes, shown in Figure 1.4, we need to use the values of ϕ at the neighboring nodes 1 and 2. For the sake of clarity, let us

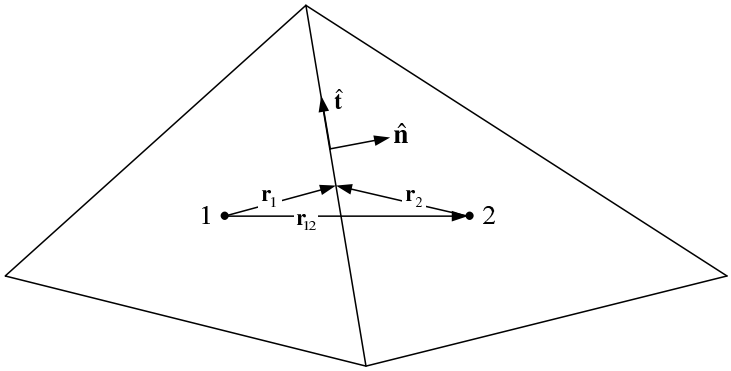


Figure 1.4: Schematic of two control volumes and their common face

assume ϕ to be a scalar quantity. The flow of ϕ, denoted by $\dot{F}_\phi$, across the face follows,

$$\dot{F}_\phi = \left[\underbrace{\dot{m}\phi}_{\text{advection}} - \underbrace{\Gamma A \nabla\phi \cdot \hat{\mathbf{n}}}_{\text{diffusion}} \right]_{\text{face}} \tag{1.3}$$

where $\dot{m}$ is the mass flow rate, Γ is the diffusion coefficient, and A is the face area. As seen, to calculate the flow of ϕ across the face, we need to evaluate the value of ϕ and its face-normal directional derivative, $\nabla\phi \cdot \hat{\mathbf{n}}$, at the face. For calculating the value of ϕ at the face we use an upwind scheme,

$$\phi_{\text{face}} = \phi_1 + \nabla\phi_1 \cdot \mathbf{r}_1 + \cdots \tag{1.4}$$

and for the value of $\nabla\phi \cdot \hat{\mathbf{n}}$ we use a central scheme based on the values of ϕ_1 and ϕ_2.

Once we write the flow across a face, we obtain an algebraic relation between nodal values of its neighbouring control volumes. Adding up the flows across all the faces of a control volume and equating it to the rate of change of the nodal value within the control volume results in the discretized form of the conservation equation. The approximations that we use to derive the discretized equations introduce some error into the solution, which is called the *discretization error*.

Definition 1.2 *Discretization error is the difference between the solution of the discretized equations and the exact solution of the BVP, associated with the flow field.*

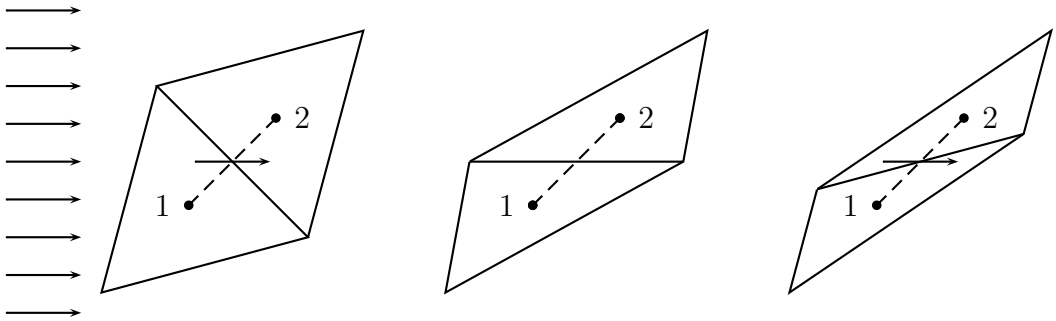


Figure 1.5: Effect of mesh geometry on the discretization of face flows

An examination of the steps that we took for discretizing the BVP reveals that the discretization error depends on two factors: the discretization method and the mesh geometry. The effect of the discretization method can be seen in Equation (1.4). If we use a higher order Taylor series expansion on a fine enough mesh, the interpolations become more accurate and the overall solution accuracy is improved. The effect of the mesh can also be seen in Equation (1.4). If we use a finer mesh, the magnitude of the vector $\mathbf{r}_1$ would be smaller and the interpolations become more accurate. However, there are also other characteristics in a mesh that may affect the discretization error.

The geometry of a mesh can have a significant impact on the discretization error. This impact is illustrated in Figures 1.5. These figures show three different meshes, with the nodes located at the same position. As seen, in Figure 1.5(a) the advective flow is from the node 1 to the node 2. However, in Figure 1.5(b), there is no advective flow between the two control volumes and in Figure 1.5(c), the direction of the advective flow changes altogether and goes from the node 2 to the node 1. As Reuss argues [69], such a non-physical phenomenon can be detrimental to the simulation accuracy. This observation partially explains why equilateral triangular cells are often preferred over skewed triangular cells. We will discuss this issue with more details in Chapter 4.

1.1.4 Solution of the Discretized Equations

The fourth and last step in a fluid flow simulation is the solution of the discretized system of equations. In this step, we linearize the discretized algebraic equations and assemble

them in the form of a linear system of equations. Then we solve the resulting system of equations using a linear solver.

Choosing the suitable linear solver depends partially on the mesh due to the impact of the mesh on the discretized equations. For example, using an unstructured mesh results in a linear system with sparse matrix of coefficients. Therefore we may have to use an iterative linear solver since the performance of direct solvers for sparse systems is poor. The other important factor is the quality of the mesh. For example, the use of skewed cells, such as the ones in Figure 1.5(c), can result in an ill-conditioned system of equations.

The emphasis of this study is not on the solution techniques for linear systems of equations. Therefore it is sufficient to use one of the freely available linear algebra systems for sparse matrices. In this study, we use the Unsymmetric Multifrontal Sparse LU Factorization Package (UMFPACK), developed by Davis [28, 29, 30, 31].

After solving the equation system and obtaining the numerical solution, the simulation process is complete. However, the numerical solution is not necessarily the same as the exact solution at the nodes. The difference between the two is called the *numerical error*. Note that the numerical error is different from the discretization error since iterative solvers also introduce some error into the final solution. However, in most practical cases we can control the contribution from the iterative solver by performing more iterations.

Definition 1.3 *Numerical error is the difference between the numerical solution and the exact solution of the BVP, associated with the flow field.*

Numerical error is an inevitable consequence in any numerical simulation and we have to examine it in order to verify the validity of a simulation.

1.2 Impact of Mesh on Discretization Error

Discretization of the conservation equations on the mesh, which is the source of discretization error, is a major source of error in a fluid flow simulation. In a real-world simulation, it is crucial to examine this error in order to verify the solution validity. As we mentioned in the previous section, the discretization error depends on both the discretization method and the mesh that the discretization is performed on.

In the finite volume method on unstructured meshes, the second-order accurate discretization is the most common option. Unfortunately, the implementation of higher order schemes on an unstructured mesh is difficult. Therefore it is not practical to improve the solution accuracy by using a higher order discretization scheme.

The discretization error also depends on the mesh. Although, there is no rigorous recipe to determine the discretization error on a given mesh, there are various techniques to reduce it. For example, we can reduce the discretization error by using a finer mesh or avoiding stretched and skewed control volumes. But is there any systematic way to reduce the discretization error? The answer is yes and the simplest method for this purpose is a uniform mesh refinement.

1.2.1 Uniform Mesh Refinement

The obvious way to improve the solution accuracy is to use a uniformly refined mesh. Figure 1.6 shows a simple example of a uniform mesh refinement. The solution on the refined mesh is more accurate provided that both meshes are fine enough and their features are the same. For example in Figure 1.6, both meshes are triangular with equilateral cells.

The advantage of uniform mesh refinement is its simplicity. In theory, we can arbitrarily refine a mesh until we obtain the desired accuracy. The problem with this method is the rapid increase in the required computer resources as the mesh becomes finer. Since computer resources are limited, uniform mesh refinement becomes intractable at some point. Therefore we need to consider an alternative approach; an approach that reduces the error while ensuring that the computer resources remain affordable.

1.2.2 Optimal Mesh

Mesh optimization simply implies reducing the solution error through mesh modification while the computational cost is retained. In a CFD simulation, the computational cost depends on the complexity of the numerical method and the number of unknowns in the problem. The number of unknowns is a multiple factor of the number of control volumes. Therefore if we use a certain numerical method, the computational cost would be a function of the number control volumes. In this work, we define the concept of

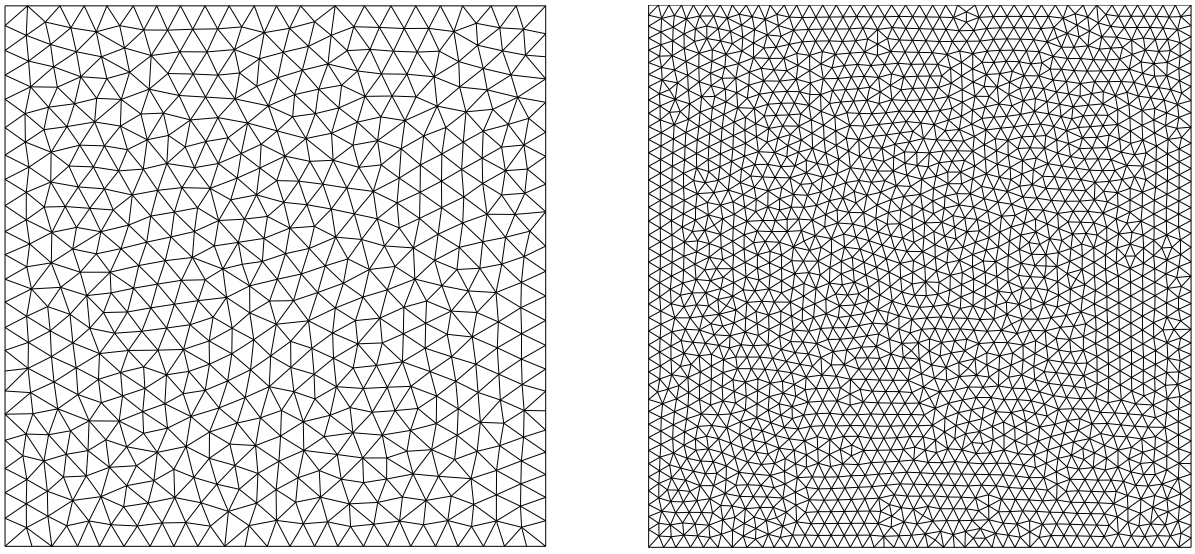


Figure 1.6: Example of uniform mesh refinement

optimal mesh based on the number of control volumes.

Definition 1.4 *We call a mesh optimal if it minimizes the discretization error for a prescribed number of control volumes.*

One of the major challenges in CFD is to develop methods for generating meshes that approximate the optimal mesh. For a variety of reasons, including the fact that the exact solution is unknown, it is impossible to generate the optimal mesh in most situations. In practice it is only feasible to generate a nearly optimal mesh, which approximates the optimal mesh. The methods used for generating these nearly optimal meshes are called *mesh adaptation* methods. The purpose of mesh adaptation methods is to modify or adapt an initial mesh in attempt to improve its approximation of the optimal mesh.

Definition 1.5 *Mesh adaptation is a process that modifies a given mesh to improve its approximation of the optimal mesh in the absence of solution error information.*

The above definition implies that although the solution error information is not available, we have to estimate it in order to be able to adapt the mesh. In the following section, we elaborate on this idea.

1.3 Solution Based Mesh Adaptation

Mesh generation is generally an *a priori* stage in the simulation process. In other words, we generate a mesh for a given geometry without knowing exactly the flow field properties. Therefore there is no guarantee that the mesh would be suitable for resolving the important features in the flow. This shortcoming in mesh generation brings up the concept of solution-based mesh adaptation. In a solution-based adaptation method, we start the simulation with an initial mesh. After obtaining the solution on this initial mesh and analysing its characteristics, we identify the important features of the flow field and modify the mesh such that the solution accuracy is improved. Let us clarify the idea using a simple example.

1.3.1 Example: Flow in Lid-Driven Cavity

To demonstrate the mesh adaptation methodology, we use the classic two-dimensional square lid-driven cavity flow [13, 22, 36, 37, 44, 66, 78]. Although the cavity flow is not of major engineering importance, it is of significant scientific importance because it exhibits many important fluid flow phenomena in a very simple geometry [78]; phenomena like boundary layers, free shear layers, impinging jets, and recirculating flows. Therefore in this work, we emphasize on the lid-driven cavity flow as the major test case. Figure 1.7 shows the schematic of the problem. The Reynolds number of the flow, based on the lid velocity, U, and the cavity size, L, is defined by $\mathrm{Re} = \rho UL/\mu$, where ρ and μ are the density and viscosity of the fluid, respectively. The boundary conditions of the flow are no-slip walls. To set the pressure level, the pressure at the lower left corner is set to zero.

Figure 1.8 shows the flow streamlines at $\mathrm{Re} = 1600$ based on the solution on a uniform Cartesian mesh with 256×256 control volumes. In this figure, the mass flow rate between any two streamlines is the same and consequently, flow velocity is inversely proportional to spacing between streamlines. As seen, close to the lid surface there is a layer of high velocity flow which produces a strong shear layer. There are three recirculating bubbles in the flow; two big bubbles in the bottom corners and one small one close to the top left corner. The flow inside these bubbles is slow and viscous-dominated. These observations suggest that a non-uniform mesh might be better for resolving the flow features. For

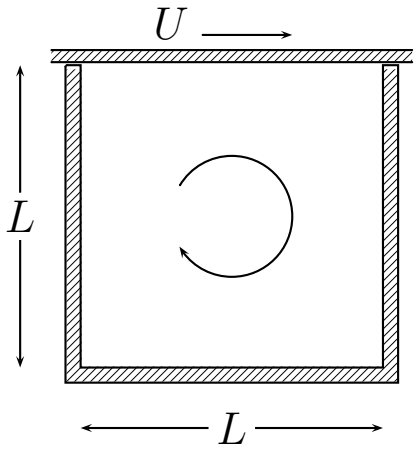


Figure 1.7: Schematic of a lid-driven cavity flow

example, a non-uniform mesh with small cells at the top is required to resolve the strong shear layer and a coarse mesh at the bottom is sufficient to resolve the low-velocity recirculation bubbles.

Figure 1.9 shows an example of a non-uniform mesh refinement. In this figure, both meshes have approximately the same number of control volumes. However, the mesh on the left is uniform, but the other one is not. The fact that both meshes in Figure 1.9 have approximately the same number of control volumes implies that the simulation cost on these meshes is roughly the same but the solution on the non-uniform mesh may be more accurate, which is the objective of mesh adaptation.

The adaptation method outlined above is an example of manual mesh adaptation. Although the solution on the non-uniform mesh in Figure 1.9 is as costly as that of the uniform mesh, generating the non-uniform mesh involves extra effort from the user since the adaptation process is time consuming. The other problem with the manual approach is its dependence on the level of user expertise. In other words, the user must have a good understanding of the physics of the flow and the effect of the mesh on the solution error in order to enhance the mesh. These limitations urge us to consider automatic mesh adaptation.

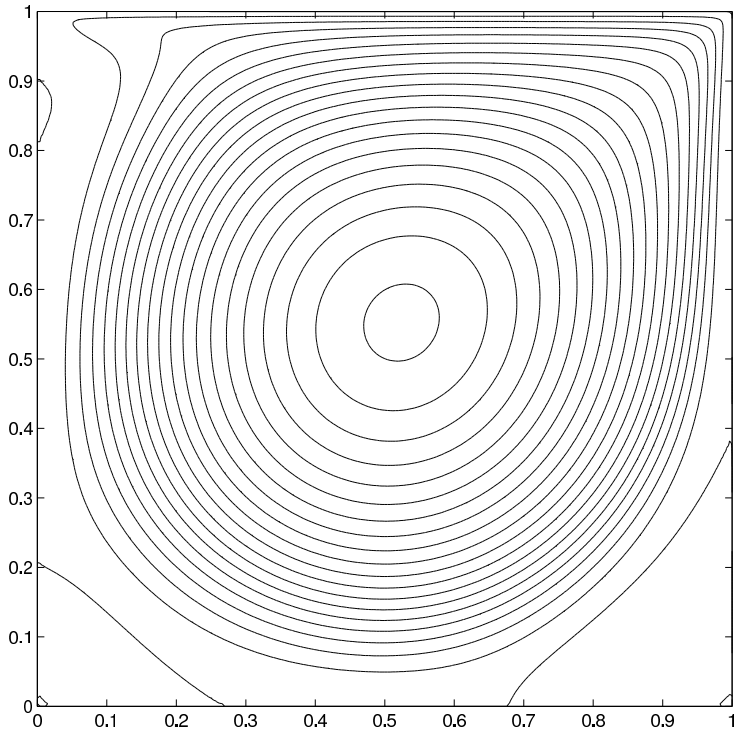


Figure 1.8: Streamlines of the lid-driven cavity flow at Re = 1600

1.3.2 Automating the Adaptation Process

Mesh adaptation is a tedious and lengthy process that is better automated. An automatic mesh adaptation method has the capability to detect important flow features in the same way that a user does and adapts the mesh accordingly. Despite the great variety of the existing adaptation methods in the literature, all of them share three common components [69]: An error indicator that discerns important features in the flow, a mesh alteration mechanism that changes the mesh geometry, and a criterion that relates the former two components.

The error indicator is a measure that designates the regions with higher contribution to the solution error. This higher contribution may be due to critical features in the flow field or poor mesh geometry. In the cavity flow example, we used strain rate as an indicator. The strain rate is not the only possible indicator and we can propose other

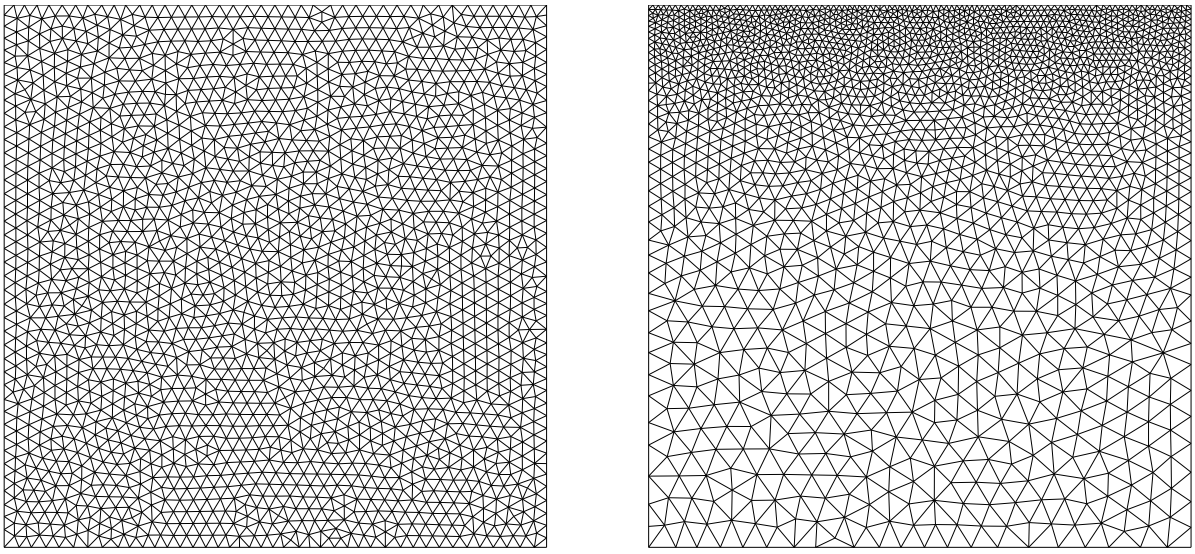


Figure 1.9: Example of non-uniform mesh refinement for the lid-driven cavity flow; Both meshes have roughly the same number of control volumes.

plausible indicators such as vorticity.

Definition 1.6 *Error indicator is a measure based on a numerical solution that quantifies the effect of fluid flow features and mesh geometry on the discretization error.*

A mesh alteration mechanism or simply an adaptation mechanism is the component that modifies the mesh geometry in an attempt to improve its capability to resolve the critical features in the flow field and to reduce the discretization error. Subdividing control volumes into smaller ones or merging a few control volumes to form a larger control volume are examples of adaptation mechanisms.

Adaptation criterion is the component that relates the adaptation indicator to the mesh geometric characteristics. Intuitively we know that the mesh must be refined in the regions of high error indicator. However, adaptation criterion determines the quantitative relationship between the two. For example in the cavity flow, we may compare the strain rate within each control volume to a threshold and subdivide the control volume if the the strain rate is larger than the threshold.

Definition 1.7 *Adaptation criterion is a relationship between the error indicator and the mesh geometric characteristics.*

A review of the adaptation methods in the literature reveals that researchers have proposed many automatic mesh adaptation methods based on the above three components. However, there are still unresolved issues that need to be addressed.

1.4 Motivation and Objectives

Although researchers have proposed many adaptation methods during the past thirty years, mesh adaptation is still an active area of research [11] especially in the context of finite volume method. Historically, the finite volume method was developed and used by engineers. As a consequence, the method has not been rigorously analysed on a mathematical basis until recently. In a recent work, Nicaise [61] reviews the present mathematical understanding of the finite volume method. Therefore the impact of mesh geometry on the solution accuracy is still not a well understood issue and it is relevant to ask how a mesh affects the simulation quality and how the solution quality can be improved by modifying the mesh.

The traditional approach to mesh adaptation is based on the solution features. This approach proves to be very effective for the flows that are dominated by discontinuities and steep gradients, such as shock waves and strong shear layers. But in the case of flows with multiple competing mechanisms and smooth gradients, the quality of the feature based adaptation methods degrades. Low speed incompressible flows are a good example of this kind of flow.

In recent years, a few researchers started analysing the impact of mesh geometry on the solution accuracy in the context of finite volume methods. Some of these works show promising results for low speed flows. However, this work is mainly motivated by the observation that none of the previous works in the literature indicates the capability of fully automatic solution-based mesh design without requiring user intervention and tuning.

The objective of this work is to propose an automatic mesh adaptation method which can improve the solution accuracy of a wide range of physical problems in the context of

steady-state laminar incompressible flows. In this sense, the thesis addresses the following issues:

- A mathematically rigorous adaptation indicator that can identify the important features in the flow and can be applied to a wide range of physical problems without needing user intervention;
- An adaptation criterion that can use the adaptation indicator in order to extract the mesh parameters including:
  - The local mesh characteristic size;
  - The local characteristics of the cells (aspect ratio, orientation, and skewness); and
  - The topology of the mesh (triangular, quadrilateral, or mixed).
- And an adaptation mechanism that can generate a target mesh so that the conditions of the adaptation criterion are met.

1.5 Scope of the Thesis

This thesis is in the context of two-dimensional steady-state incompressible laminar flows of constant-property Newtonian fluids. The governing equations are the conservation of mass and momentum, expressed by the continuity and Navier-Stokes equations, respectively. To solve these equations numerically, we use second-order accurate finite volume method with a cell-centered collocated arrangement on an unstructured mesh. The mesh topology may be triangular, quadrilateral, or mixed.

As we will see by the end of this work, the above assumptions are not conceptual restrictions. Nevertheless, we use these assumption for the sake of simplicity.

1.6 Outline of the Thesis

This thesis consists of six chapters. After this introduction, in Chapter 2, we discuss the conventional approaches toward mesh adaptation for low-speed flows and review the

literature on the subject. We also highlight the issues associated with the existing methods and set the direction for a general adaptation method.

In Chapter 3, we propose a general adaptation indicator for low-speed flows and its application in mesh adaptation. We also point out the restrictions of such an indicator, which leads us to the less restrictive isotropic and anisotropic error indicators in Chapters 4 and 5.

In Chapter 4, we propose a simplified version of the adaptation indicator, proposed in Chapter 3, for isotropic meshes and devise an isotropic mesh adaptation method for triangular meshes.

In Chapter 5, we extend the method of Chapter 4 to the more general case of anisotropic mesh adaptation. We show that in the case of anisotropic meshes, the performance of a mixed triangular-quadrilateral mesh may be superior to that of a purely triangular mesh. However, a mixed mesh introduces additional issues in the adaptation process. We address these issues and evaluate the performance of the method.

In the final chapter, Chapter 6, we summarize the thesis by a conclusion. We highlight the strengths and shortcomings of the proposed adaptation method and also give some suggestions for future works.

Chapter 2

Conventional Mesh Adaptation Methods

Researchers have proposed many mesh adaptation methods during the past thirty years. Although these methods differ in particular details, all of them are developed for a single purpose: improving the accuracy of a CFD simulation by modifying the distribution and topology of mesh elements in an attempt to reduce the manual work required for mesh generation. To fulfill this purpose, an adaptation method has to be capable of identifying and quantifying the characteristic scales in a flow in order to generate a mesh that can resolve these scales accurately. As a result, the CFD analyst can save time and effort on generating the proper mesh. Unfortunately, the existing mesh adaptation methods may not be able to fulfill this purpose for certain CFD applications.

The traditional mesh adaptation methods prove to be effective for flow fields that are dominated by discontinuities and steep gradients. However, if a flow field involves multiple competing mechanisms with different characteristic scales, the quality of the traditional adaptation methods degrades. This degradation is mainly due to the use of improper criteria in mesh adaptation. In these situations, the CFD analyst may have to tune the adaptation scheme in order to obtain a reasonable mesh. Unfortunately, such tunings usually involve a time intensive trial and error process. The question is how to avoid such pitfalls in mesh adaptation?

In this chapter, we review the common approaches to mesh adaptation and highlight

their advantages and disadvantages. The chapter begins with an example of a feature-based mesh adaptation method. In this example, we demonstrate the mesh adaptation methodology and discuss the impact of error indicators on its performance. Then we review the common alternative approaches for developing error indicators in the mesh adaptation literature. In the last two sections, we will discuss the shortcomings of the existing error indicators in order to set the direction of the thesis towards developing a more comprehensive error indicator.

2.1 An Example of Feature-Based Adaptation

This section demonstrates the traditional feature-based mesh adaptation method using the example of a lid-driven cavity flow at Re = 1600, shown in Figure 1.7. In the previous chapter, we presented a preliminary analysis of this flow and suggested an adapted mesh based on its qualitative features, Figure 1.9. Although such a qualitative approach is insightful, it is not feasible for complex flow fields. Therefore we need an adaptation method that can systematically identify and quantify the characteristic scales in the flow and adapt the mesh accordingly. As mentioned in the previous chapter, an adaptation scheme has three components: an error indicator, an adaptation criterion, and an adaptation mechanism.

The first component of an adaptation method is the error indicator. In feature-based mesh adaptation methods, the role of the error indicator is to detect the important features in the flow. Therefore the error indicator is chosen based on the dominant features in the flow field. In low speed incompressible flows, which are dominated by viscous effects, velocity derivatives are commonly used as error indicators since they can detect boundary layers and free shear layers [63, 32]. Even if the indicator is based on velocity derivatives, there would still be a range of possible error indicators. For example, Yang [83] uses a combination of pressure gradient and vorticity but Nithiarasu and Zienkiewicz use the magnitude of velocity gradient [63]. To understand the impact of error indicators on mesh

adaptation performance let us consider the following four different indicators:

$$e_1 = \frac{L|\varepsilon|}{2U} = \frac{L}{U}\left|\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right| \tag{2.1}$$

$$e_2 = \frac{L|\zeta|}{U} = \frac{L}{U}\left|\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\right| \tag{2.2}$$

$$e_3 = \frac{L}{U}\left[\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 + \left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2\right]^{\frac{1}{2}} \tag{2.3}$$

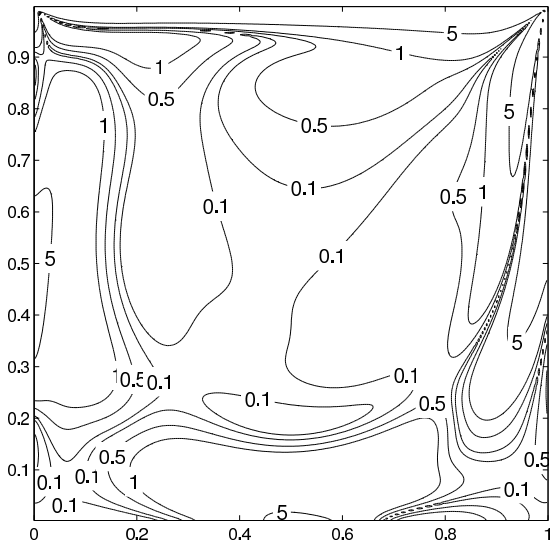
$$e_4 = \frac{L}{\rho U^2}\left[\left(\frac{\partial p}{\partial x}\right)^2 + \left(\frac{\partial p}{\partial y}\right)^2\right]^{\frac{1}{2}} \tag{2.4}$$

where ε is the strain rate and ζ is the vorticity of the flow. In the above relations, each indicator is nondimensional and scaled by the global flow parameters. The first three indicators are based on the velocity gradient and the last one, e_4, is based on the pressure gradient. We use the pressure gradient based indicator in order to compare its effectiveness to that of the velocity gradient based indicators. Figures 2.1(a) through 2.1(d) show the contour plots of the indicators e_1 through e_4, respectively, for the cavity flow at $\mathrm{Re} = 1600$, calculated on a uniform Cartesian mesh with 256×256 control volumes. As seen, the average levels of the indicators are different. Even amongst velocity gradient based indicators the distribution of the indicators vary. All the velocity gradient based indicators show a higher level close to the driven lid. In contrast, the pressure-based indicator shows a totally different distribution, especially near the driven lid and in the middle of the cavity.

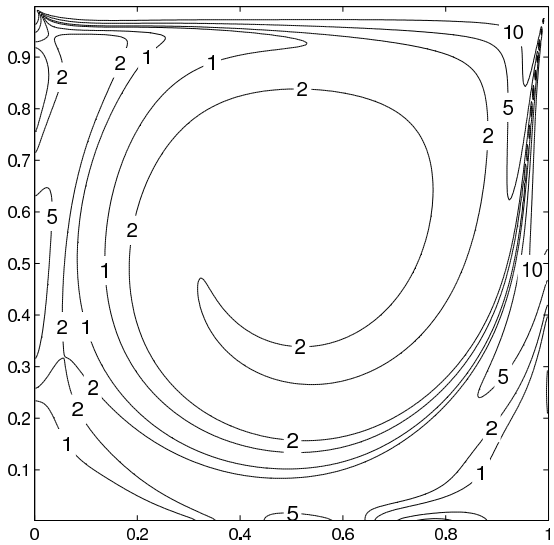
The second component of an adaptation scheme is the adaptation criterion. This component determines how the mesh size depends on the error indicator. Intuitively we know that the mesh size must scale inversely with the error indicator. One common criterion for this purpose is the equidistribution principle [3, 5, 11, 18, 20, 41]. For example, in a second-order method where the error scales with the square of the mesh size, h^2, equidistribution implies:

$$h^2 \propto \frac{1}{e}$$

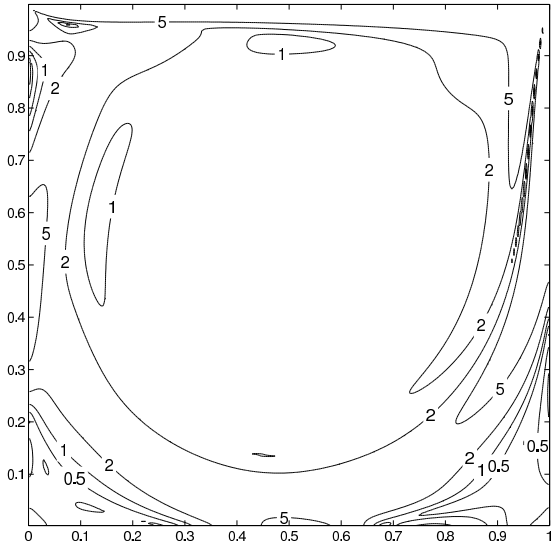
where e is the error indicator. Unfortunately, the above formula may result in very small or very large mesh cells depending on the value of the error indicator. A typical way for



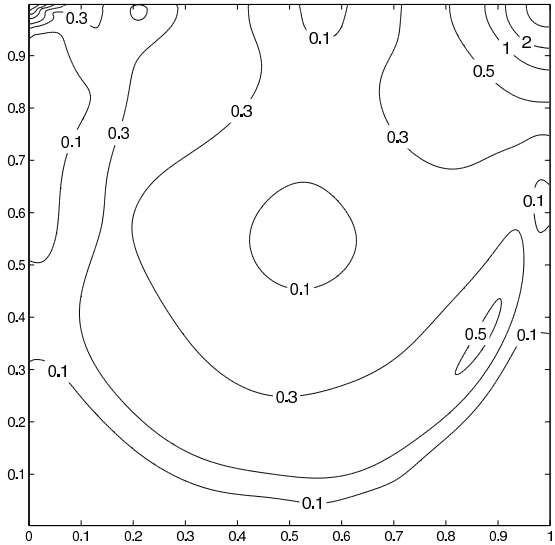
(a) Strain rate (e_1)



(b) Vorticity (e_2)

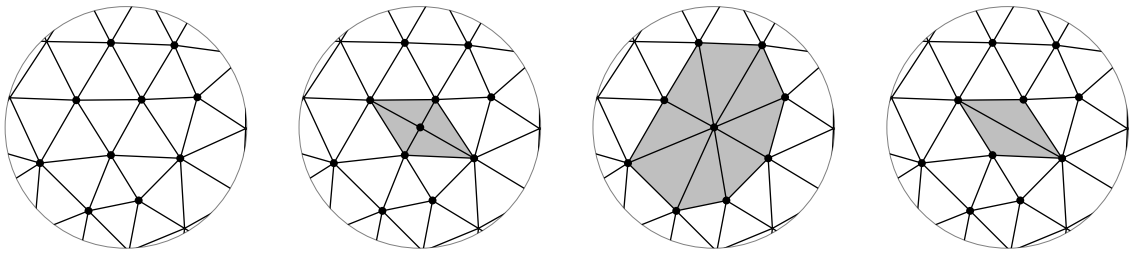


(c) Velocity gradient magnitude (e_3)



(d) Pressure gradient magnitude (e_4)

Figure 2.1: Contours of error indicators e_1 through e_4 for the lid-driven cavity flow at $\mathrm{Re} = 1600$, calculated on a 256×256 Cartesian mesh.



(a) Original mesh (b) Face splitting (c) Face collapsing (d) Face swapping

Figure 2.2: Typical adaptation mechanisms for triangular meshes; the shaded areas indicate the affected regions of the mesh.

ruling out the very small or very large cells is to limit the range of h as follows:

$$h = \text{median}\left(h_{\min}, \frac{C}{\sqrt{e}}, h_{\max}\right) \tag{2.5}$$

where $h_{\min}$ and $h_{\max}$ are the minimum and maximum permitted cell sizes in the adapted mesh, respectively, and C is a scaling factor that determines the average cell size in the adapted mesh. Equation (2.5) represents a typical heuristic adaptation criterion in the literature. The form of this criterion is not unique and there are many other alternative forms in the literature as well. However, most of these criteria are conceptually similar.

The third component of an adaptation scheme is the adaptation mechanism. The adaptation mechanism is the algorithm for modifying the mesh geometry in order to satisfy the criterion of Equation (2.5). There are many adaptation mechanisms in the literature and we discuss them in detail in Chapter 4. For the moment, we use an iterative mechanism that involves vertex movement, face splitting, face collapsing, and face swapping, illustrated in Figure 2.2. These mechanisms are the most common adaptation mechanisms for triangular meshes [19, 20].

Figure 2.3 shows adapted triangular meshes based on the above method. To generate these meshes we use the values of $h_{\min} = 0.001L$ and $h_{\max} = 0.05L$ in Equation (2.5). As seen, all the meshes are triangular and nearly isotropic with roughly the same number but different distribution of cells. In Figures 2.3(a) through 2.3(c) in which the underlying indicator is based on the velocity gradient, mesh refinement mostly occurs near the walls.

This near wall refinement leads to an improved resolution of the wall induced shear layers. In contrast, in Figure 2.3(d) in which the underlying indicator is based on the pressure gradient, the refinement occurs in the top corners and in the inner region of the cavity. This distribution leads to an improved resolution of the pressure field within the cavity. Therefore the adapted meshes reflect some qualitative features of the flow field. However, the significant qualitative difference among these adapted meshes raises a few questions. Which of these adapted meshes provide the most accurate simulation? Does the answer depend on the number of mesh cells? The following section addresses these questions.

2.2 Analysis of the Accuracy Improvement

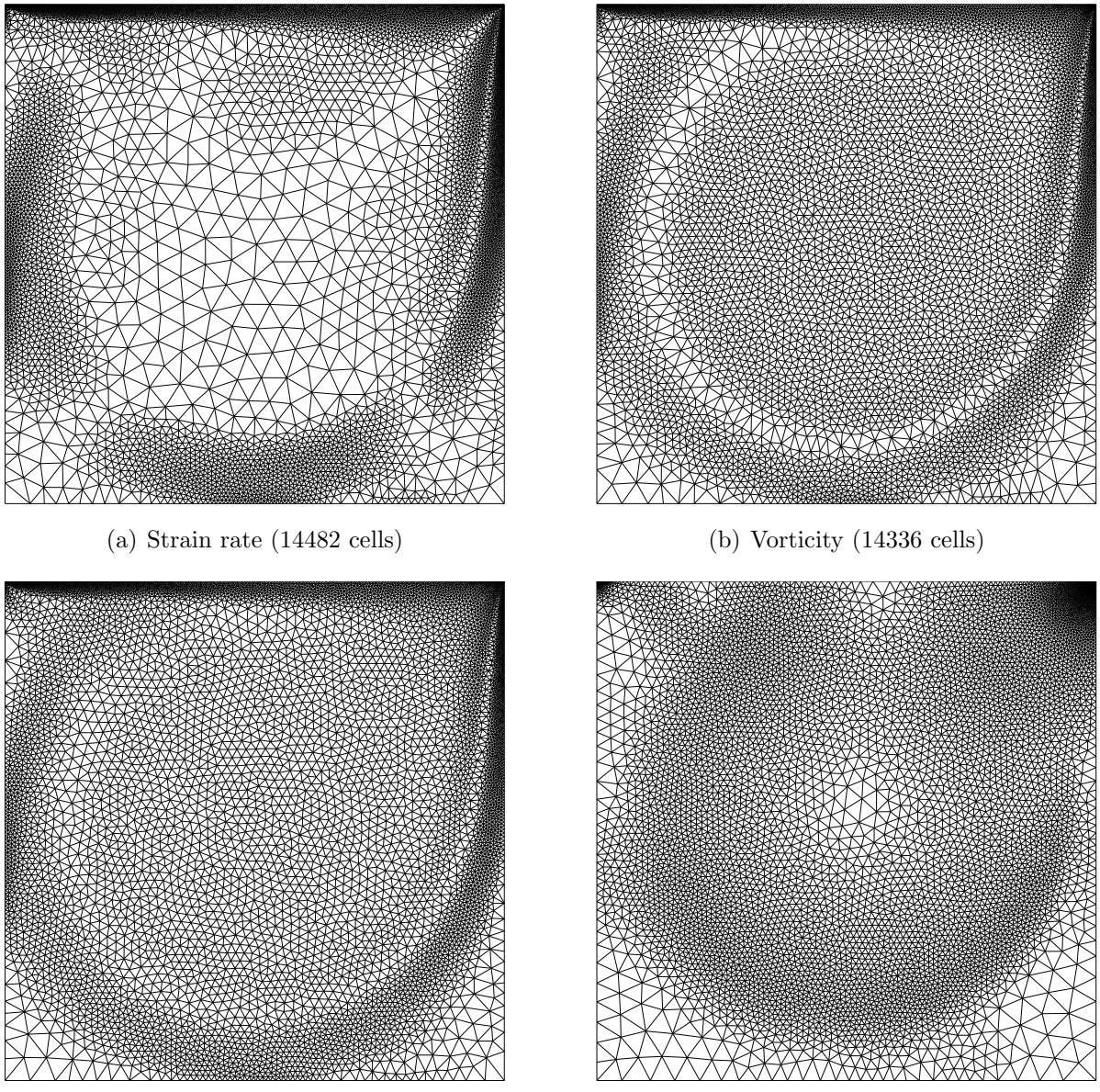
The goal of mesh adaptation is to improve the solution accuracy. Therefore the performance of an adaptation scheme must be judged based on its impact on the solution error. Unfortunately, evaluating the solution error is not easy since in many practical CFD applications the exact solution is unknown. However, we can use the mesh independent solution as an approximation to the exact solution. To find the mesh independent solution we need to solve the problem on a series of successively refined meshes and to extrapolate the results using techniques such as the Richardson extrapolation [73, 74].

In mesh refinement studies, we need a parameter for evaluating the mesh convergence of the solution. For example, Erturk *et al.* [36] use the vorticity and the stream-function at the centre of the primary vortex as the mesh convergence parameters. In this work, we use the total kinetic energy of the flow within the cavity as the mesh convergence parameter:

$$K = \int_{\Omega} \frac{1}{2}\rho(\mathbf{v}\cdot\mathbf{v})dV \approx \sum_{i=1}^{N} \frac{1}{2}\rho(\mathbf{v}_i\cdot\mathbf{v}_i)V_i \tag{2.6}$$

where V is the volume, $\mathbf{v}$ is the velocity vector, ρ is the fluid density, and i is the index of the i-th control volume, which ranges from one to the number of mesh cells, N. For the sake of convenience, we may prefer to use the nondimensional form of the kinetic energy, K^*, defined by:

$$K^* = \frac{K}{\frac{1}{2}\rho U^2 L^2} \tag{2.7}$$



(a) Strain rate (14482 cells)

(b) Vorticity (14336 cells)

(c) Velocity gradient magnitude (14268 cells)

(d) Pressure gradient magnitude (14137 cells)

Figure 2.3: Adapted meshes for the lid-driven cavity flow at Re = 1600 with $h_{\min} = 0.001L$ and $h_{\max} = 0.05L$ based on various error indicators.

Table 2.1: Nondimensional total kinetic energy, K^*, for the cavity flow at Re = 1600 on three different uniform Cartesian meshes; the last row is the extrapolated value to an asymptotic fine mesh.

CVs (n)	Mesh Size (h)	K^*	Error
16384	0.0078125	0.089941	1.82%
65536	0.0039062	0.091149	0.504%
262144	0.0019531	0.091493	0.129%
$n \to \infty$	$h \to 0$	0.091611	

In this work, we use uniform Cartesian meshes for obtaining the mesh independent solution. Table 2.1 shows the values of K^* based on the solutions on three different uniform Cartesian meshes with 128×128, 256×256, and 512×512 cells. Since the discretization for these solutions is second order accurate, the discretization error of K^* on an adequately fine mesh should satisfy a power series of the form:

$$K^*_{\text{exact}} = K^*_h + a_1 \left(\frac{h}{L}\right)^2 + a_2 \left(\frac{h}{L}\right)^4 + a_3 \left(\frac{h}{L}\right)^6 + \cdots \tag{2.8}$$

where h is the mesh characteristic size, K^*_h, is the value of K^* on such a mesh, and $a_1, a_2, \cdots$ are constant coefficients. In practice, we cannot use Equation (2.8) for calculating the exact solution, K^*_{exact}, since the solutions on an infinite number of meshes are required. However, we can use the following truncated series:

$$K^*_{\text{MI}} \approx K^*_h + a_1 \left(\frac{h}{L}\right)^2 + a_2 \left(\frac{h}{L}\right)^4$$

where K^*_{MI} is the estimated mesh independent solution. To calculate K^*_{MI}, we use the latter formula and substitute the values of h and K^* from Table 2.1. The last row of Table 2.1 shows the mesh independent value of K^*, which is equal to 0.091611. We use this value as an estimate of the exact solution.

Table 2.2 shows the values of K^* and their errors on the adapted meshes and Figure 2.4 summarizes the results by plotting the error of K^* versus the number of control volumes

for all the uniform and adapted meshes. As seen, the results for the uniform Cartesian and triangular meshes exhibit second order convergence as expected. In Figure 2.4, we expect the solid points associated with the adapted meshes to lie below the line for the uniform triangular mesh. The vertical distance between the solid points and the line for uniform triangular meshes indicates the reduction in the solution error. As seen, all the adapted meshes exhibit some improvement in the solution accuracy except the three coarsest pressure-gradient-based adapted meshes. In the case of the coarse adapted meshes, the adaptation indicator based on the strain rate shows the best performance followed by the indicators based on the vorticity, velocity gradient, and pressure gradient. However, in the case of the finest adapted meshes, the pressure gradient based indicator results in the smallest error. Therefore we cannot draw any general conclusion about the quality of these indicators. Moreover, a visual assessment of the meshes in Figure 2.3 may suggest that the vorticity based or the velocity gradient based adapted meshes are the best since they look pleasing. Apparently this is not the case in Figure 2.4 and these two indicators lead to poorer results compared to those of the strain rate based indicator.

The above example shows that choosing the proper error indicator is not trivial and the results may turn out to be counter-intuitive. These observations raise an important question about the foundations of error indicators in mesh adaptation methods. How can we know that a specific error indicator is going to work for a certain CFD simulation? The following section presents a brief review of the common approaches for establishing error indicators in the mesh adaptation literature.

2.3 Approaches to Establish Error Indicators

Although the purpose of mesh adaptation is to reduce the discretization error of a CFD simulation, in practice it is only feasible to reduce the error by indirect means using an error indicator. The common approaches for this purpose in the literature are based on the equidistribution or minimization of the error indicator. The error indicator is based on the information extracted from the numerical solution. In other words, once we obtain the numerical solution to a flow field on an initial mesh, we analyse it in order to determine the effect of the mesh distribution and topology on the numerical error and establish

Table 2.2: Nondimensional total kinetic energy, K^*, and its error for the cavity flow at $\mathrm{Re} = 1600$ based on four different adaption indicators

Indicator	# of CVs (n)	K^*	Error
Strain Rate (e_1)	1486	0.091068	0.593%
	3947	0.093078	1.60%
	14482	0.092248	0.695%
	56800	0.091893	0.308%
Vorticity (e_2)	1183	0.090884	0.793%
	3707	0.093286	1.83%
	14336	0.092320	0.774%
	56914	0.091893	0.308%
Velocity Gradient (e_3)	1220	0.094629	3.29%
	3632	0.093306	1.85%
	14268	0.092578	1.05%
	56716	0.091873	0.286%
Pressure Gradient (e_4)	1118	0.065132	28.9%
	3668	0.086725	5.33%
	14137	0.090321	1.41%
	54950	0.091412	0.217%

bounds on it. A review of the mesh adaptation literature shows that there are four major approaches for characterizing the discretization error and establishing an error indicator:

- Detecting certain features in the flow field;
- Evaluating the sensitivity of an integral quantity;
- Evaluating the interpolation errors; and
- Evaluating a residual-based indicator.

Let us discuss each approach in more detail.

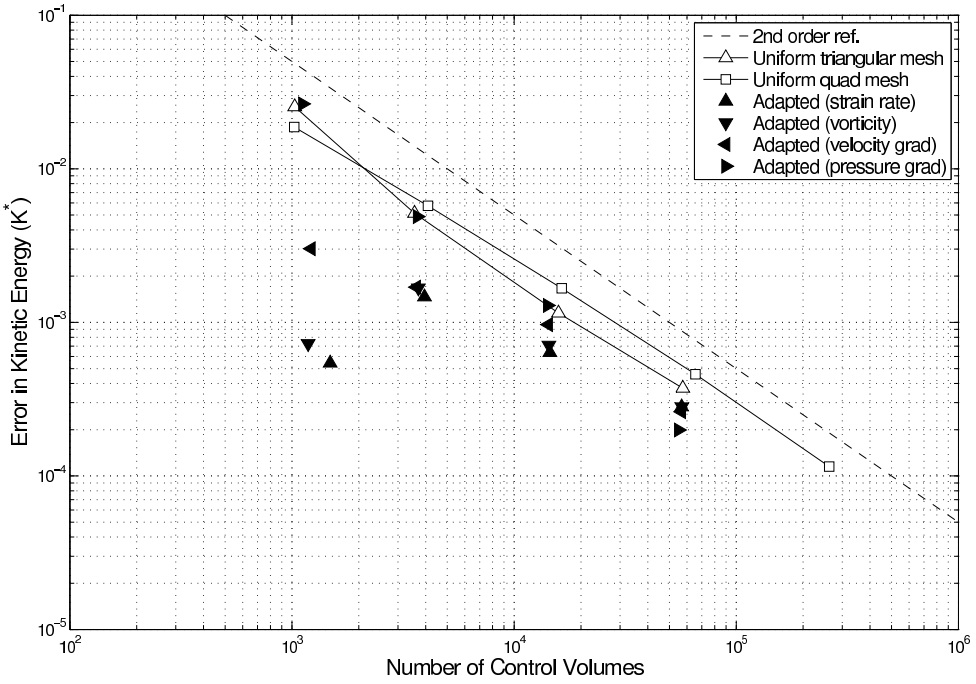


Figure 2.4: Comparison between the error on uniform and adapted meshes for the lid-driven cavity flow at Re = 1600.

In the traditional mesh adaptation methods, the error indicator is based on detecting important features in the flow field; features like jets, boundary layers, shock waves, and contact surfaces. These features can be characterized by primitive variables such as velocity and pressure or by derived variables such as velocity gradient or total pressure. For example, a jet can be characterized by its velocity and a boundary layer can be characterized by its velocity gradient. In the context of incompressible flows, velocity and pressure based indicators are the most common ones for mesh adaptation. Lin *et al.* [56] use total pressure and Nithiarasu and Zienkiewicz [63] use velocity gradient. In these works, the equidistribution principle is an indispensable component. In other words, mesh elements are distributed so that their size would be inversely proportional to the error indicator. Although this method has been successfully applied to many CFD applications during the past three decades, it is not very suitable for automatic mesh adaptation due

to its flawed premises. The problem is that there is no clear relationship between the solution error and the flow features. In addition, the success of such a method strongly depends on the proper choice of error indicator. Therefore the CFD analyst needs to have a good understanding of the physics of the flow and the processes that lead to the production of error. Despite these shortcomings, feature-based indicators are widely used due to their simplicity and ease of implementation.

The second approach for characterizing the numerical error of a CFD simulation is based on evaluating the sensitivity of an integral parameter to mesh refinement. A good example of an integral parameter in these methods is the lift or drag of an airfoil or the total kinetic energy of the flow within a lid driven cavity. The basic idea is to evaluate the sensitivity of the integral parameter to local mesh refinement in order to designate the mesh elements that need refinement. If the mesh in a certain region is fine enough, the integral parameter shows no sensitivity to the refinement, otherwise the parameter changes. As a result we can detect the mesh regions that need refinement. In practice, actual mesh refinement is not necessary and one can solve an adjoint problem to evaluate the sensitivity [2, 81, 82]. Although this approach is ideal for optimizing a certain parameter, there is no guarantee that the overall solution accuracy is improved as well. For example, optimizing the mesh around an airfoil for the lift calculation does not necessarily lead to an accurate calculation of the drag. Moreover, simultaneous optimization of more than one parameter is difficult. Therefore this approach is used only in applications where the accuracy of a certain parameter is crucial. But as Baker points out [9], this method is not suitable for improving the overall solution accuracy.

The third approach for characterizing the numerical error is based on the concept of interpolation error. In this approach a higher order interpolation operator is applied to a discrete solution in order to recover a higher order solution. For example, if the initial solution is second-order accurate, a third- or a fourth-order interpolation operator can be applied to it. The difference between the original solution and the recovered one is called the interpolation error [9, 62]. Figure 2.5 illustrates the interpolation error for a one-dimensional problem. In mesh adaptation methods that use interpolation-based error indicators, the goal is to redistribute the mesh cells so that the error indicator is minimized throughout the domain. However, there is no rigorous mathematical procedure for this

purpose and most researchers simply use the equidistribution principle [26, 45]. The main advantage of this method is its simplicity and flexibility for various applications. For example, many researchers in computer science use this method for surface meshing [43] and image compression [27]. In fluid mechanics, this approach became very popular with the emergence of anisotropic mesh adaptation methods [4, 6, 18, 23, 26, 45]. We will discuss anisotropic mesh adaptation methods in more detail later on in Chapter 5. But this approach has the same shortcomings as those of the feature-based approach. In other words, the choice of the proper adaptation indicator is application dependent. It is also hard to justify that the equidistribution or even minimization of the interpolation error leads to the minimization of the discretization error.

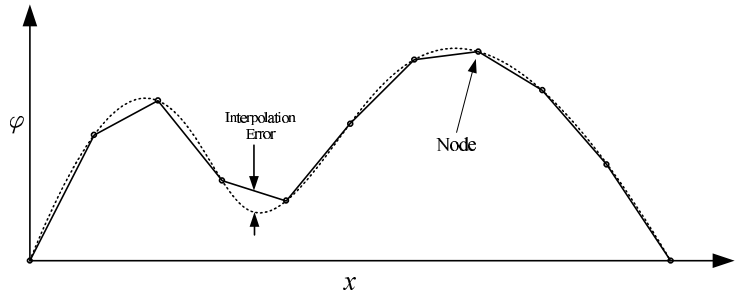


Figure 2.5: Interpolation error in a one dimensional problem; solid line is the numerical solution and dashed line is a higher order interpolation.

The fourth approach for characterizing the numerical error in a CFD simulation is to use the concept of residual. This approach is based on the fact that residual is the source of numerical error and its reduction leads to error reduction as Reuss [69], Reuss and Stubley [70], and Hay and Visonneau [46, 49] point out. This approach is popular in the finite element context since residual is an offshoot of the numerical solution. In contrast, in the finite volume context an explicit additional equation must be used to evaluate the solution residual [70, 85, 75, 53]. Once the residual is evaluated, it can be used to designate the mesh elements that need refinement. In the literature, there are a few works that use this approach and confirm its effectiveness. For example, Zhang *et al.* [85] apply a residual-based indicator for mesh adaptation in one-dimensional hyperbolic problems and Reuss [69] uses a similar approach for repositioning the mesh points in two-

dimensional flow of scalars and fluids. More recently, Hay and Visonneau [49] use a rigorous method for estimating the solution residual and show that residual is an effective indicator for mesh adaptation. The only noticeable disadvantage of this approach is that the calculation of a residual-based indicator is more complicated than a feature-based or an interpolation-based indicator especially on unstructured meshes.

The above review indicates that the feature detection is not the only approach towards proposing error indicators. There are other approaches that lead to more promising error indicators for mesh adaptation. In particular, the residual-based approach provides a solid foundation for proposing indicators that are tightly integrated with the mathematical model of the CFD problem. These observations raise the important question that why feature-based indicators are still widely used in mesh adaptation? Could we simply use a residual-based indicator instead of the gradient-based or pressure-based indicators in the example of lid-driven cavity flow? Unfortunately, replacing a feature-based indicator by a residual-based one is not a trivial issue since they have different characteristics. The following section compares these differences.

2.4 Residual-Based versus Feature-Based Indicators

Let us examine why the feature detection approach is more common in the mesh adaptation literature whereas the residual-based approach seems to be based on a stronger premise. The short answer is that the feature-based indicators are easier to incorporate in adaptation methods. In contrast, residual-based indicators, at least in their current form in the literature, are only applicable within certain mesh adaptation schemes. The following subsections compare the various aspects of these two approaches in more detail.

2.4.1 Complexity of Calculation

In terms of ease of calculation, feature-based indicators are usually preferred since they are based on solution variables and gradients. However, in situations that the error indicator is based on second- or higher-order derivatives, additional effort is required to recover these derivatives from a second-order accurate discrete solution. In contrast, residual-based indicators are more complicated since evaluating the solution residual for

a control volume involves calculating the errors of flow quantities across the faces of the control volume. These face flow errors can be estimated using a Taylor series analysis or an accurate interpolation of the discrete solution [49]. In both cases, the process of calculating residual is more involved than evaluating a feature-based indicator.

2.4.2 Dependence on Initial Mesh

A desirable feature of any mesh adaptation method is its capability to produce the same final adapted mesh independent of the initial mesh. This feature, which we call *robustness with respect to the initial mesh*, is important since we prefer to obtain the adapted mesh with as little effort as possible applied to generating the initial mesh. In this regard, feature-based adaptation methods have the inherent advantage of producing more consistent results compared to residual-based adaptation methods. This difference is due to the fact that the feature-based indicators are designed for detecting certain *features in the flow* whereas the residual-based indicators are designed for estimating the solution *residual on a given mesh*. Therefore the distribution of a feature-based indicator is roughly the same on various initial meshes while the distribution of a residual-based indicator varies. As a result, the residual-based adaptation methods are more dependent on the initial mesh and consequently less robust with respect to the initial mesh.

2.4.3 Extension to Anisotropic Adaptation

In the past decade there has been a clear trend towards using anisotropic mesh adaptation methods. A mesh is called anisotropic if the aspect ratio of its cells is much larger than one. Anisotropic meshes are beneficial for resolving anisotropic flow features that have different characteristic scales in various directions. For example, a boundary layer is an anisotropic feature since its characteristic scale in the streamwise direction is much larger than its characteristic scale in the transverse direction.

The most common error indicators for anisotropic mesh adaptation in the literature are based on solution features and interpolation errors. These error indicators are mostly related to the tensor of second order derivatives, called the Hessian, of physical variables [26, 35, 57]. For example, if in Equation (2.4) we use the pressure Hessian as the

error indicator instead of the pressure gradient, the error indicator would be:

$$\mathbf{E} = \frac{L^2}{\rho U^2} \begin{pmatrix} \frac{\partial^2 p}{\partial x^2} & \frac{\partial^2 p}{\partial x \partial y} \\ \frac{\partial^2 p}{\partial x \partial y} & \frac{\partial^2 p}{\partial y^2} \end{pmatrix}$$

and the equidistribution criterion of Equation (2.5) changes to:

$$\mathbf{h}^T \mathbf{E} \mathbf{h} = C$$

where C is a constant and $\mathbf{h}$ is the mesh characteristic size vector. As seen, the mesh characteristic size is a vector as opposed to equation (2.5) where it is a scalar quantity. Therefore the orientation of mesh cells determines their characteristic size depending on the eigenvalues and eigenvectors of the tensor $\mathbf{E}$.

Although the above technique is the basis of many anisotropic mesh adaptation methods in the literature, it has still the same shortcomings as those of the feature-based methods. On the other hand, the use of residual-based indicators for anisotropic mesh adaptation is not very straightforward. The reason is the nonlinear effects of mesh adaptation on a residual-based error indicator that slow down the adaptation process. We will explain this issue in detail in Chapter 3. Therefore the feature-based and interpolation-based error indicators are the only existing options in the context of anisotropic mesh adaptation.

2.4.4 Dependence on Boundary Conditions

A desirable feature of mesh adaptation methods is to improve the solution accuracy for all CFD applications that are governed by a certain mathematical model. Note that specific applications using a certain mathematical model are distinguished by their boundary conditions. If a mesh adaptation method can effectively improve the accuracy of all reasonable boundary conditions, we call it *robust with respect to the boundary conditions*.

The brief literature review of the previous section shows that the residual-based approach is the only promising approach towards developing an adaptation method that is robust with respect to the boundary conditions. However, the residual-based approach has other shortcomings, as discussed above. The question is how to alleviate these shortcomings.

2.5 Towards a General Residual-Based Error Indicator

The literature review of this chapter shows that although feature detection is not the only approach for establishing error indicators, none of the other approaches are as easy to implement and affordable. However, the feature detection approach has the disadvantage of being vulnerable to a poor choice of error indicator in complex applications. In other words, the feature-based mesh adaptation methods are not robust with respect to boundary conditions. In contrast, residual-based mesh adaptation methods prove to be robust for a wider range of applications. However, the residual-based methods are more complicated and less flexible than the feature-based methods. The immediate question is how to develop a residual-based error indicator that is as easy to implement and affordable as a feature-based error indicator, which brings us to the objectives of this thesis.

The main objective of this thesis is to develop a residual-based error indicator that not only retains its robustness with respect to boundary conditions, but also possesses the advantages of feature-based error indicators. In summary, the proposed residual-based error indicator should enable us to construct a mesh adaptation method with the following properties:

- The proposed adaptation method must be as easy to implement as a feature-based method.
- The proposed adaptation method must be as robust as a feature-based method with respect to the initial mesh.
- The proposed adaptation method must be applicable to anisotropic mesh adaptation.

In the following three chapters, we propose a residual-based adaptation method that has the above characteristics. In the following chapter, we develop a method for estimating the residual of a finite volume simulation. We derive the proposed method by analysing the sources of error in the governing equations. Then we use this estimate as the indicator in a classic mesh refinement and investigate its effectiveness. In Chapter 4 we simplify the

residual estimator, developed in Chapter 3, and derive a residual-based error indicator that is suitable for isotropic mesh adaptation. In Chapter 5, we extend the proposed indicator to anisotropic mesh adaptation methods and propose an anisotropic error indicator.

Chapter 3

Application of Residual in Mesh Adaptation

In the previous chapter, we discussed the importance of a residual-based error indicator that is as simple and robust as a feature-based error indicator. However, first we need to clarify the precise concept of residual and the reason that it is a good candidate for mesh adaptation. The objective of this chapter is to explain the concept of residual in the context of the finite volume method and the techniques for its estimation. We also investigate the possibility of using residual as an error indicator in mesh adaptation and discuss its advantages and disadvantages.

This chapter is divided into five sections. In the first section, we demonstrate the concept of residual using a simple one-dimensional finite difference example. This example helps us to understand the impact of residual on solution error and its merit in mesh adaptation. In the second section, we develop a method for estimating the residual of a second order finite volume solution of the system of mass and momentum equations. We use this residual estimate in the third section as an error indicator in classic h-refinement. In the last two sections, we show that the classic h-refinement is not the most effective strategy for mesh adaptation since it fails to take full advantage of the information provided by the error indicator. This discussion sets the stage for proposing more flexible residual-based mesh adaptation methods in Chapters 4 and 5.

3.1 The Concept of Residual: A Finite-Difference Example

Let us demonstrate the concept of residual using a simple one-dimensional boundary value problem. Consider the following second order linear ordinary differential equation with two Dirichlet boundary conditions:

$$\frac{d^2\Phi}{dx^2} + 30x^4 = 0 \qquad \text{BCs} : \Phi(-1) = \Phi(1) = 0 \tag{3.1}$$

The analytic solution to the above boundary value problem is:

$$\Phi(x) = 1 - x^6 \qquad x \in [-1, 1] \tag{3.2}$$

We can also solve Equation (3.1) numerically using the finite difference method. The first step in the process of numerical solution is to generate a mesh. In this example, we simply subdivide the interval $[-1, 1]$ into n evenly spaced sub-intervals as the mesh. Therefore we have to find the solution at the discrete set of points, $x_0, \ldots, x_n$. For this purpose we need to discretize Equation (3.1) using the finite difference method. If we apply a second-order central difference scheme to Equation (3.1), the following finite difference equation is obtained:

$$\frac{\phi_{i-1} - 2\phi_i + \phi_{i+1}}{\Delta x^2} + 30x_i^4 = 0 \qquad i = 1, \ldots, n-1 \tag{3.3}$$

where ϕ_i is the numerical solution at the point x_i and Δx is the size of each mesh sub-interval. Assembling Equation (3.3) for $i = 1, \ldots, n-1$ and using the boundary conditions, $\phi_0 = \phi_n = 0$, we obtain a linear system of $n + 1$ equations for the values of ϕ_i, which can be solved using a linear solver. Figure 3.1 shows the analytical solution, Equation (3.2), and the numerical solution based on the above method for $n = 20$. As seen, the numerical solution is slightly different from the exact solution due to the discretization error.

The discretization error, as discussed in the first chapter, is the difference between the exact solution to a boundary value problem and the numerical solution to its corresponding discretized equations. In mathematical terms, the discretization error at the point x_i is:

$$e_i = \phi_i - \Phi(x_i)$$

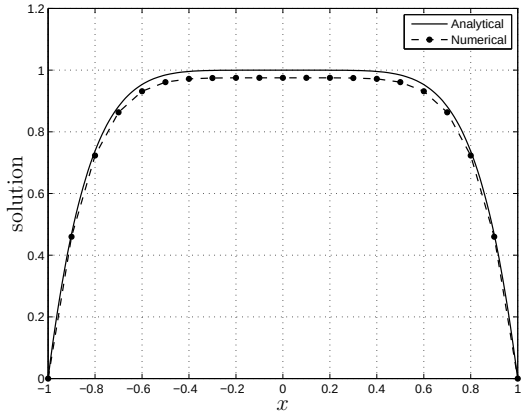


Figure 3.1: Analytical and numerical solution of the one-dimensional boundary value problem of Equation (3.1)

or equivalently, $\phi_i = \Phi(x_i) + e_i$. If we substitute this relation into Equation (3.3) and rearrange the result, we obtain:

$$\frac{\Phi(x_{i-1}) - 2\Phi(x_i) + \Phi(x_{i+1})}{\Delta x^2} + 30x_i^4 = -\frac{e_{i-1} - 2e_i + e_{i+1}}{\Delta x^2} \qquad i = 1, \ldots, n-1$$

The left hand side of the above equation has the same form as that of Equation (3.3), however, the numerical solution ϕ_i is replaced by the exact solution $\Phi(x_i)$. As seen, the exact solution does not satisfy the discretized equation since the right hand side is not zero. In other words, the discretized equation needs an apparent source term in order to satisfy the exact solution. This apparent source term is called residual:

$$\delta_i = -\frac{e_{i-1} - 2e_i + e_{i+1}}{\Delta x^2}$$

Note that the above relation is the central difference formula for the second derivative of error, d^2e/dx^2. This is not a coincidence since Equation (3.1) is a linear equation and its left hand side involves the second derivative operator. In nonlinear equations such as the Navier-Stokes equation, the residual would bear less resemblance to the discretized form of the differential operator in the original equation.

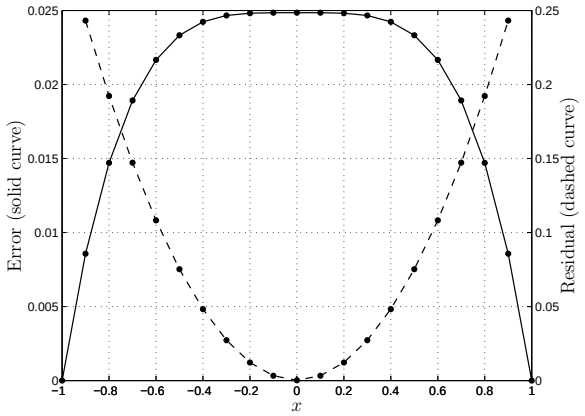


Figure 3.2: Distribution of error and residual in the finite difference solution of the one-dimensional boundary value problem of Equation (3.1)

The concept of residual is of great importance in mesh adaptation since it provides information about error production and the contribution of each control volume to the overall discretization error. Figure 3.2 illustrates this issue by showing the distribution of error and residual for the above finite difference example. As seen, the maximum error occurs at the middle of the domain while the maximum residual occurs at the domain end-points. We can use this information to shrink the mesh size around the domain end-points and expand it in the central region. The idea of residual-based mesh adaptation for finite volume fluid flow simulations is very similar to the above finite difference example.

3.2 Residual Estimation in Fluid Flow Simulation

In fluid flow simulation, we use the finite volume method to discretize the system of mass and momentum equations and then we solve the discretized equations. The numerical solution obtained from the finite volume method is approximate, which means the exact solution of the mass and momentum equations does not satisfy the discretized equations. Therefore substituting the exact solution into the discretized mass and momentum equations results in a residual. In practice, we cannot use this method for evaluating the

solution residual since in the most practical applications the exact solution is unknown. Therefore we need a method for estimating the solution residual in the absence of the exact solution.

In this section, we develop a method for estimating the residual of the finite volume solution to a flow field without knowing its exact solution. For this purpose, we start with discretizing the governing equations and solving them using the finite volume method. Then we use the numerical solution for estimating the solution residual through analysing face flow errors. Once the residual estimation is complete, we can use the results in mesh adaptation.

3.2.1 The Governing Equations

The governing equations for incompressible flows are conservation equations for mass and momentum, which were presented in Equations (1.1) and (1.2), respectively. For the sake of convenience, we repeat these equations for a steady-state incompressible isothermal flow of a Newtonian fluid in the absence of gravity:

$$\nabla \cdot \mathbf{v} = 0 \tag{3.4}$$

$$\nabla \cdot (\rho\, \mathbf{v} \otimes \mathbf{v}) = -\nabla p + \nabla \cdot \Big[\mu \left(\nabla \mathbf{v} + \nabla \mathbf{v}^T\right)\Big] \tag{3.5}$$

where $\mathbf{v} = (u, v)$ is the flow velocity vector, p is the flow pressure, ρ is the fluid density, μ is the fluid viscosity, and the superscript $(\quad)^T$ is the transpose operator. Note that the mass equation (3.4) is scalar while the momentum equation (3.5) is vectorial. Therefore the entire system of mass and momentum equations for a two-dimensional flow field consists of three equations, namely for mass, momentum in the x direction, and momentum in the y direction. For the sake of brevity, we may summarize these equations in the following symbolic form.

$$\mathscr{L}(\Phi) = 0 \tag{3.6}$$

where $\mathscr{L}$ is the differential operator for the conservation of mass and momentum, presented in Equations (3.4) and (3.5), acting on the vector of primitive variables, Φ.

$$\mathbf{\Phi} = \begin{pmatrix} p \\ u \\ v \end{pmatrix} \tag{3.7}$$

In the most practical applications, Equation (3.6) has no analytical solution and we have to solve it numerically. Therefore we have to discretize Equation (3.6). In this work we use the finite volume method.

3.2.2 Finite Volume Discretization and Numerical Solution

In the finite volume method, the discretization process starts with integrating Equation (3.6) over each control volume in order to obtain the integral conservation equations,

$$\int_{\Omega_i} \mathscr{L}(\Phi)\, d\Omega = 0 \tag{3.8}$$

where Ω_i represents the volume of the i-th control volume, shown in Figure 3.12. Note that by applying the Gauss Divergence theorem [7], we can change the volume integrals into surface integrals and write Equations (3.4) and (3.5) in terms of mass and momentum flows across the faces of the control volumes.

$$\oint_{\partial\Omega_i} \mathbf{v} \cdot d\mathbf{s} = 0 \tag{3.9}$$

$$\oint_{\partial\Omega_i} (\rho \mathbf{v}\mathbf{v}) \cdot d\mathbf{s} = -\oint_{\partial\Omega_i} p\, d\mathbf{s} + \oint_{\partial\Omega_i} \mu(\nabla\mathbf{v} + \nabla\mathbf{v}^T) \cdot d\mathbf{s} \tag{3.10}$$

where $\partial\Omega_i$ is the boundary, or rather the faces, of the i-th control volume. The next step is to approximate the above surface integrals in terms of the average flow across every single face of the control volumes; thus we need to substitute the surface integrals by summation over the faces.

$$\sum_{\text{face}} J_{\text{vol}} = 0 \tag{3.11}$$

$$\sum_{\text{face}} \mathbf{J}_{\text{mom}} = \sum_{\text{face}} \mathbf{F}_{\text{pres}} + \sum_{\text{face}} \mathbf{F}_{\text{visc}} \tag{3.12}$$

where J_{vol} is the volumetric flow rate, $\mathbf{J}_{\text{mom}}$ is the vector of momentum flow rate, and $\mathbf{F}_{\text{pres}}$ and $\mathbf{F}_{\text{visc}}$ are the pressure and viscous forces, respectively, acting on the control volume faces. Although these flows and forces are unknown, we can calculate them based on the values of ϕ at the nodes of the neighbouring control volumes. The core idea is to use an n-th order accurate interpolation scheme in order to estimate the value of ϕ at the control volume faces. Then we can use this estimate for calculating the flow of mass and momentum across the faces.

The derivation of the fully discretized equations is tedious and we leave it to Appendix A for the sake of brevity. However, the final result for the i-th control volume would be of the following form:

$$\phi_i + \sum_{j \in \text{nb}} c_{ij}\phi_j = b_i \tag{3.13}$$

where ϕ is the numerical solution at the i-th control volume and j is the index of its neighbouring control volumes, denoted by 'nb' in Figure 3.12. Note that in Equation (3.13), the coefficients c_{ij} and b_i may generally be functions of ϕ and as a result Equation (3.13) would be a nonlinear equation. for the sake of brevity, we write this equation in the following symbolic form:

$$\mathcal{L}_h^n(\phi_i) = 0 \tag{3.14}$$

where $\mathcal{L}_h^n$ is the n-th order accurate discrete form of $\mathscr{L}$ on a mesh with the characteristic size h. The order of accuracy, n, depends on the order of accuracy of the interpolation scheme that we use for calculating the face flows and forces in Equations (3.11) and (3.12).

Writing Equation (3.13) for all control volumes leads to a system of nonlinear algebraic equations for the values of ϕ at the nodes of the control volumes.

$$[\mathbf{I} + \mathbf{C}(\boldsymbol{\phi})]\boldsymbol{\phi} = \mathbf{b}(\boldsymbol{\phi})$$

where $\mathbf{I}$ is the identity matrix, $\mathbf{C}(\boldsymbol{\phi})$ is a sparse matrix containing c_{ij} coefficients in Equation (3.13), $\mathbf{b}$ is the vector of b_i coefficients in Equation (3.13), and $\boldsymbol{\phi}$ is the solution vector. In spite of the nonlinearity of the latter, we can linearize it by lagging the coefficient matrix $\mathbf{C}$ and the right hand side vector $\mathbf{b}$.

$$\mathbf{A}(\boldsymbol{\phi}^k)\boldsymbol{\phi}^{k+1} = \mathbf{b}(\boldsymbol{\phi}^k) \tag{3.15}$$

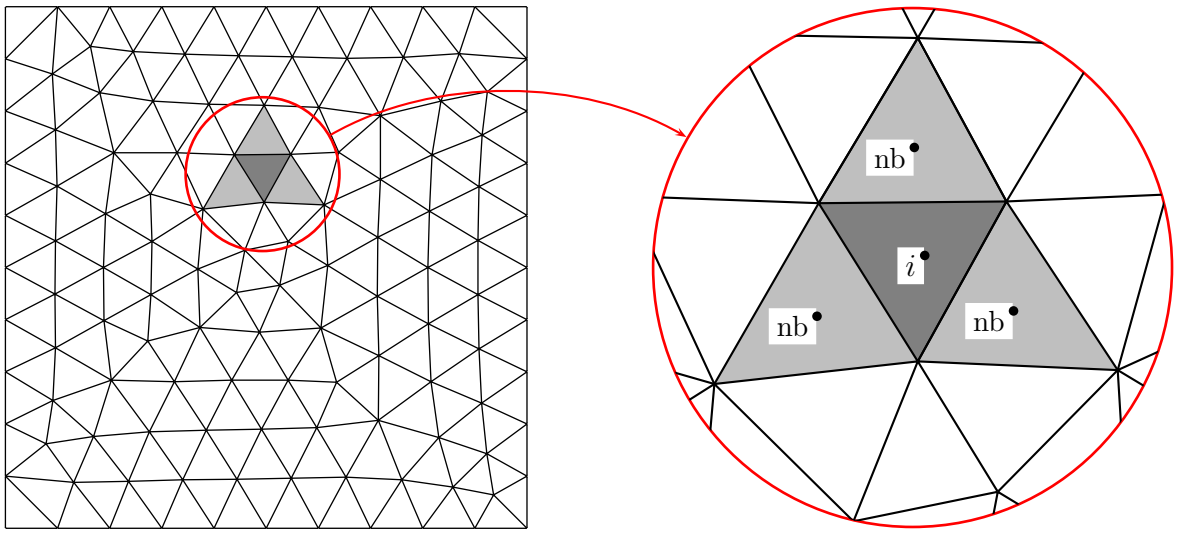


Figure 3.3: Schematic of a control volume and its neighbours in the cell-centred finite volume method on an unstructured triangular mesh

where $\mathbf{A} = \mathbf{I} + \mathbf{C}$. As seen, the matrix $\mathbf{A}$ and the vector $\mathbf{b}$ are evaluated based on the solution at the previous iteration, $\boldsymbol{\phi}^k$. Therefore we have to start with an initial guess $\boldsymbol{\phi}^0$ and solve Equation (3.15) iteratively to find $\boldsymbol{\phi}^1, \boldsymbol{\phi}^2, \ldots$ until convergence.

The numerical solution obtained from the above procedure differs from the exact solution due to discrete approximations. In the next section, we examine the difference between the two and discuss the concept of residual.

3.2.3 Error and Residual of the Numerical Solution

Error is the difference between the exact and the numerical solution as mentioned before. Assuming that the exact and numerical solutions at the node of the i-th control volume are Φ_i and ϕ_i, respectively, the solution error is:

$$e_i = \phi_i - \Phi_i \tag{3.16}$$

Note that the exact solution, Φ, does not necessarily satisfy the discrete equation (3.14). In other words, substituting the exact solution into the discrete equation results in:

$$\mathcal{L}_h^n(\Phi_i) = \delta_h^n|_i \tag{3.17}$$

where δ_h^n is the residual for an n-th order accurate scheme on a mesh with the characteristic size h. Equation (3.17) reveals the residual as the apparent source of error.

Definition 3.1 *Residual is the apparent source term needed to satisfy the discrete conservation equations for the exact solution.*

Equation (3.17) provides a method for calculating the residual of a finite volume solution. The only problem is that the exact solution, Φ, is unknown. However, we can still use the mesh independent solution as an approximation to the exact solution. In other words, we need to find the solution on an extremely fine mesh:

$$\mathcal{L}_{h'}^n(\Phi') = 0 \quad , \quad h' \ll h$$

where h' is the characteristic size of an extremely fine mesh and Φ' is the mesh independent solution on such a mesh. The next step is to substitute the mesh independent solution, Φ', into Equation (3.17). For this purpose, we have to interpolate the mesh independent solution to the control volume nodes of the original mesh and substitute the result into Equation (3.17).

$$\delta_h^n|_i \approx \mathcal{L}_h^n(\Phi_i') \tag{3.18}$$

where Φ_i' is the value of the mesh independent solution, interpolated to the node of the i-th control volume. Writing the above equation for all control volumes and assembling them results in the following compact formula based on Equation (3.15).

$$\boldsymbol{\delta} \approx \mathbf{A}(\boldsymbol{\Phi}')\boldsymbol{\Phi}' - \mathbf{b}(\boldsymbol{\Phi}') \tag{3.19}$$

Therefore we only need to substitute the interpolated mesh independent solution in the discrete equations and find the residual of the algebraic system.

In spite of the strength of the above approach for research purposes, it is of little practical importance. If we could find the mesh independent solution, there would be no

need to solve the problem on a coarse mesh in the first place. Therefore we need to find a more practical approach towards residual estimation.

In the context of the finite volume method, we can show that the solution residual within a control volume is equal to the net flow imbalance across the faces of the control volume. Let us prove this statement using the integral form of the mass equation, Equation (3.11):

$$\sum_{\text{face}} J_{\text{vol}} = 0$$

If the exact solution were available, we could use it in order to evaluate the exact volumetric flow rate across the faces, J_{vol}. Unfortunately, the exact solution is not available and we can only use an approximation of it. As mentioned earlier in this section, in this work we use a second-order finite volume method for obtaining the numerical solution. In other words, we use a second order interpolation scheme to calculate the face flows based on the discrete numerical solution at the control volume nodes. Therefore the approximate face flow would have a linear distribution along the face. Figures 3.4(a) and 3.4(b) show an example of the difference between the exact and the approximate face flows. As seen, the second-order accurate face flow smears out many details of the exact flow. We may introduce ΔJ_{vol} as the difference between the exact and the approximate face flows.

$$J_{\text{vol}}^{\text{exact}} = J_{\text{vol}}^{\text{approx}} + \Delta J_{\text{vol}}$$

If we apply the above equation to all the faces of a control volume and add the results together, we obtain the integral form of the mass equation.

$$\underbrace{\sum_{\text{face}} J_{\text{vol}}^{\text{exact}}}_{\mathcal{L}_h^n(\Phi_i)} = \underbrace{\sum_{\text{face}} J_{\text{vol}}^{\text{approx}}}_{\mathcal{L}_h^n(\phi_i)=0} + \underbrace{\sum_{\text{face}} \Delta J_{\text{vol}}}_{\delta_{\text{vol}}} \tag{3.20}$$

Ideally we want the left hand side of the above equation to be zero. However, in practice we can only set the first term on the right hand side to zero. Therefore the residual is equal to the net flow error across the boundaries of a control volume.

$$\delta_{\text{vol}} = \sum_{\text{face}} \Delta J_{\text{vol}} \tag{3.21}$$

A similar argument holds for the momentum equation, Equation (3.12), provided that the error of the forces acting on the control volume boundary is also taken into account.

$$\boldsymbol{\delta}_{\text{mom}} = \sum_{\text{face}} \Delta \mathbf{J}_{\text{mom}} - \sum_{\text{face}} \Delta \mathbf{F}_{\text{pres}} - \sum_{\text{face}} \Delta \mathbf{F}_{\text{visc}} \tag{3.22}$$

From Equations (3.21) and (3.22) we can conclude that the solution residual in the context of the finite volume method is essentially the net flow imbalance across the faces of each control volume. Therefore we can develop more affordable methods for estimating the solution residual compared to the exact method, which requires the mesh independent solution. In the next subsection, we present a brief literature review on the subject.

3.2.4 A Review of Residual Estimation Methods

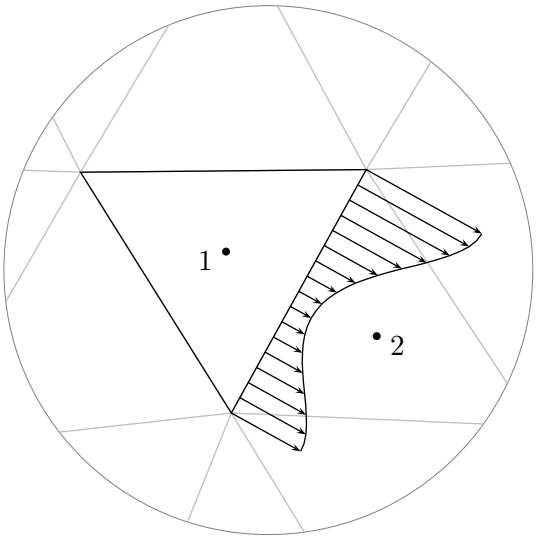
In the context of the finite volume method, the solution residual is equal to the net imbalance of mass and momentum flows across the faces of each control volume. This concept is expressed in Equations (3.21) and (3.22). Therefore we need a method for estimating the error of mass and momentum flows across the faces of each control volume. In the literature, there are three methods for estimating the face flow errors:

- Estimating the neglected terms in the discretization scheme;
- Recovering a higher order accurate solution based on the discrete solution; and
- Use of a higher order accurate discretization scheme.

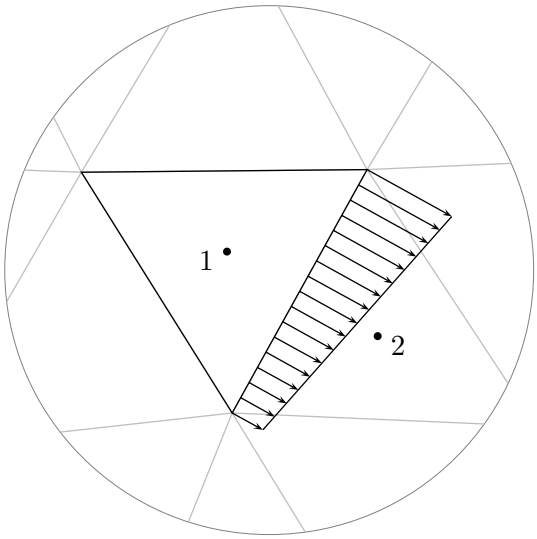
Let us discuss each method in more detail.

The first method for estimating the face flow errors is based on analysing the neglected terms in the Taylor series expansion in the discretization scheme. In this method, the neglected terms are estimated and used for establishing a higher order accurate face flow calculation. Therefore these neglected terms determine the face flow errors for mass and momentum equations. Examples of this method in the literature are the works of Ilinca *et al.* [50], Reuss and Stubley [70], and Zhang *et al.* [85].

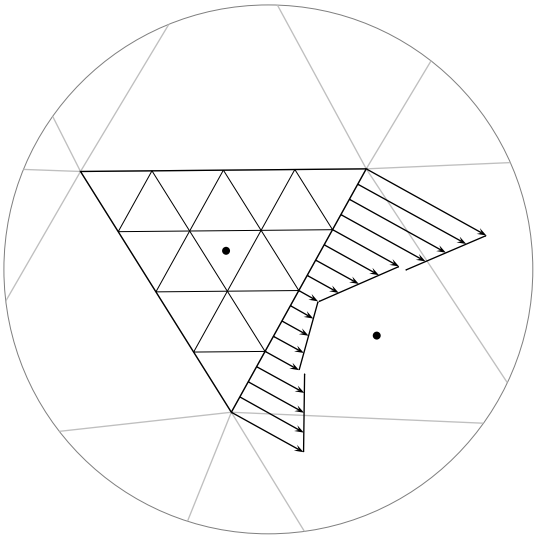
The second method for estimating the face flow errors is based on recovering a higher order accurate solution from the available discrete solution in order to obtain a more



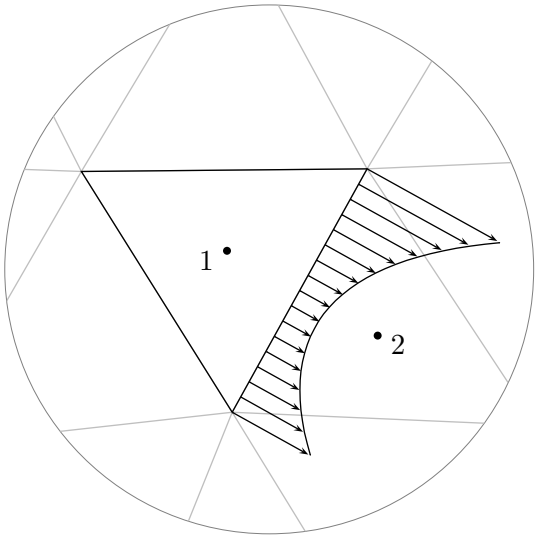
(a) Exact flow



(b) Approximate flow, 2nd order method



(c) Approximate flow, 2nd order method on a refined mesh



(d) Approximate flow, 3rd order method

Figure 3.4: Comparison of exact face flow and its approximation based on various numerical schemes and mesh sizes

accurate face flow estimation. In this method, the discrete solution on an initial mesh is used along with a higher order accurate interpolation method. Since the discrete solution is only given at the control volume nodes, the higher order interpolation provides a more accurate estimate of the solution variables at the faces. Therefore an estimate for the face flow error can be obtained. Examples of this method can be seen in recent works of Hay and Visonneau [49] and Hay *et al.* [47]

The third method for estimating the face flow errors is based on using a higher order discretization scheme. In this method, we use the numerical solution, obtained from an n-th order of accuracy discretization scheme along with another discretization scheme whose order of accuracy is m where $m > n$. Substitution of the n-th order solution in the m-th order discretization scheme results in an estimate of the residual. This method, which is traditionally referred to as the defect correction method, is elaborated on by Ervin and Layton [38] and Pierce and Giles [68].

Note that all the above methods are conceptually equivalent and the difference is only in the implementation. Therefore the choice of the proper method is a matter of convenience for a certain purpose. The advantage of the first two methods is the ease of implementation while the third method is more cumbersome to develop and implement. In contrast, the third method is specifically useful in mesh adaptation since the mesh geometry parameters can also be taken into account in the final formulation. As a result the use of a higher order discretization scheme enables us to deal with various kinds of mesh without major modification to the method. Therefore we use this method for estimating the face flow errors.

3.2.5 Adopted Residual Estimation Scheme

In this section, we develop a residual estimation technique based on using a higher order discretization scheme. In general, if the solution to a flow field is smooth enough, we can enhance the solution accuracy using a higher-order accurate discretization scheme.

$$\lim_{n\to\infty} \mathcal{L}_h^n(\phi) = 0 \Rightarrow \phi \to \Phi$$

Let us focus on the particular case of a second-order accurate discretization scheme. In this case, Equation (3.17) reads:

$$\mathcal{L}_h^2(\Phi) = \delta_h^2 \tag{3.23}$$

If the mesh size h is fine enough, a third order accurate scheme would provide a more accurate solution.

$$\mathcal{L}_h^3(\phi_h^3) = 0 \tag{3.24}$$

Comparison between Figures 3.4(a) and 3.4(d) provides a conceptual account for the above argument. As seen, the face flow based on the third order scheme is a better approximation to the exact flow compared to that of the second order scheme in 3.4(b). Therefore the third order accurate solution, ϕ_h^3, can be used as an approximation to the exact solution, Φ.

$$\mathcal{L}_h^2(\phi_h^3) \approx \delta_h^2 \tag{3.25}$$

But the latter involves calculating the third order accurate solution, ϕ_h^3, which is not available. Fortunately, we can use the following approximate formula, which is proved in Appendix B. This formula, which is applicable to solutions on a fine enough mesh, is:

$$\mathcal{L}_h^2(\phi_h^3) \approx -\mathcal{L}_h^3(\phi_h^2) \tag{3.26}$$

Substituting Equations (3.26) into (3.25) gives rise to:

$$\delta_h^2 \approx -\mathcal{L}_h^3(\phi_h^2) \tag{3.27}$$

Therefore applying the third-order discrete operator, $\mathcal{L}_h^3$, on the second-order accurate solution, ϕ_h^2, provides the solution residual.

3.2.6 Application of the Proposed Scheme and Discussion

Let us analyse the solution residual of the lid-driven cavity flow at $\mathrm{Re} = 1600$, presented in Subsection 1.3.1, using both the exact method based on the grid independent solution and the adopted method of the previous subsection. To examine the performance of the proposed method, we apply it to the solution on uniform Cartesian and triangular meshes, shown in Figure 3.5.

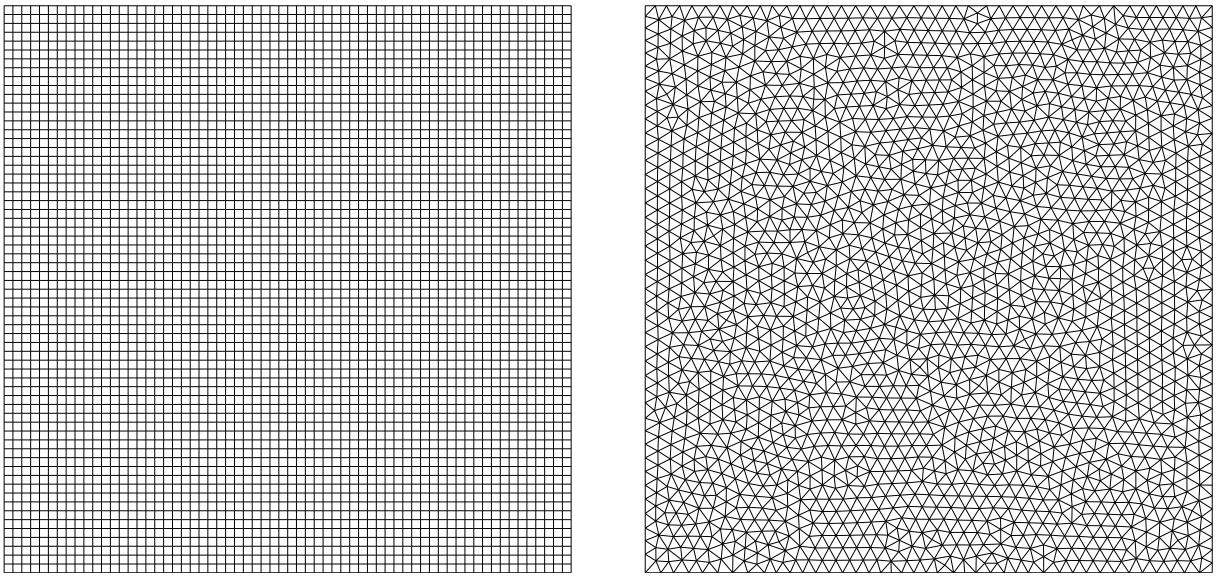


Figure 3.5: Uniform Cartesian mesh with 4096 control volumes (left) and uniform unstructured triangular mesh with 3827 control volumes(right)

The first step is to simulate the flow field within the cavity. For this purpose, we use the second-order accurate finite volume method, presented in Appendix A. The simulation involves solving the system of mass, x-momentum, and y-momentum equations. Therefore we have to evaluate the residual of each equation separately.

After simulating the flow field we can use the exact method of Equation (3.18) or the method of Equation (3.27) for calculating the solution residual. Figures 3.6 through 3.11 show the results. As seen, each figure consists of four plots. The first plot, labelled (a), shows the distribution of the true residual logarithm based on the exact method. In this method as mentioned before, we use the mesh independent solution as an approximation to the exact solution and substitute it in Equation (3.18). In this work, we calculate the mesh independent solution on a uniform Cartesian mesh with 512×512 control volumes. The solution error on this mesh, given in Table 2.1, is about 0.13%.

The second plot in Figures 3.6 through 3.11, labelled (b), shows the distribution of the estimated residual logarithm based on the proposed method in the previous subsection. To

Table 3.1: The slope of the linear least squares fit and the correlation factor of the scatter plot in Figures 3.6(d) through 3.11(d).

Mesh Type	Equation	Correlation	Fitted Line Slope
Cartesian	mass	0.7242	1.0047
Cartesian	x-momentum	0.7468	1.0580
Cartesian	y-momentum	0.7595	1.0554
Triangular	mass	0.6938	1.1041
Triangular	x-momentum	0.6207	0.9244
Triangular	y-momentum	0.6058	0.8923

obtain these plots, we substitute the second order accurate solution into Equation (3.27). A comparison between the plots (a) and (b) reveals the difference between the average level of residual throughout the domain. For the sake of clarity, plot (c) in each figure shows the scaled estimated residual so that its average throughout the domain is the same as that of the true residual in plot (a).

A comparison between plots (a) and (c) reveals the strong resemblance of the actual and the estimated residuals. For example, in all figures the residual in the top corners is high and the variation of residual order of magnitude is roughly the same. Although this qualitative comparison is instructive, we need a better quantitative measure to evaluate the performance of the method.

Plot (d) in Figures 3.6 through 3.11, shows the scatter plot of the estimated residual logarithm versus the true residual logarithm for each control volume. If the proposed method is of high quality, the data should scatter around the $y = x$ line. In other words, both the correlation coefficient and the slope of the scatter plot must be close to one.

The coefficient of correlation is a measure for the linear relationship between the two variables [1]. The definition of the correlation coefficient for two sets of data is:

$$r_{x,y} = \frac{N \sum x_i y_i - \sum x_i \sum y_i}{\sqrt{N \sum x_i^2 - (\sum x_i)^2}\ \sqrt{N \sum y_i^2 - (\sum y_i)^2}} \tag{3.28}$$

where x and y are the data sets and N is the number of samples. if $r_{x,y} = 0$ there is no

relationship between x and y and if $r_{x,y} = 1$ there is a perfect linear relationship between the two. Table 3.1 lists the correlation coefficients between the true and the estimated residuals. As seen, the correlation on the Cartesian mesh is generally larger than that of the triangular mesh. However, the minimum correlation, associated to the y-momentum equation on the triangular mesh, is still significantly large and equal to 0.6058. Therefore there is a strong relationship between the estimated and the true residual. Although a linear relationship is necessary, it is not sufficient and we must examine the slope of the data as well. The slope of the data is a measure of the quality of the proposed method. In this study, we use a normal-distance linear least squares fit to find the slope of the data and Table 3.1 lists the results. As seen, the slope is generally close to one. The above results show that the proposed residual estimation method is effective for estimating the solution residual.

3.3 Residual as the Error Indicator in Mesh Adaptation

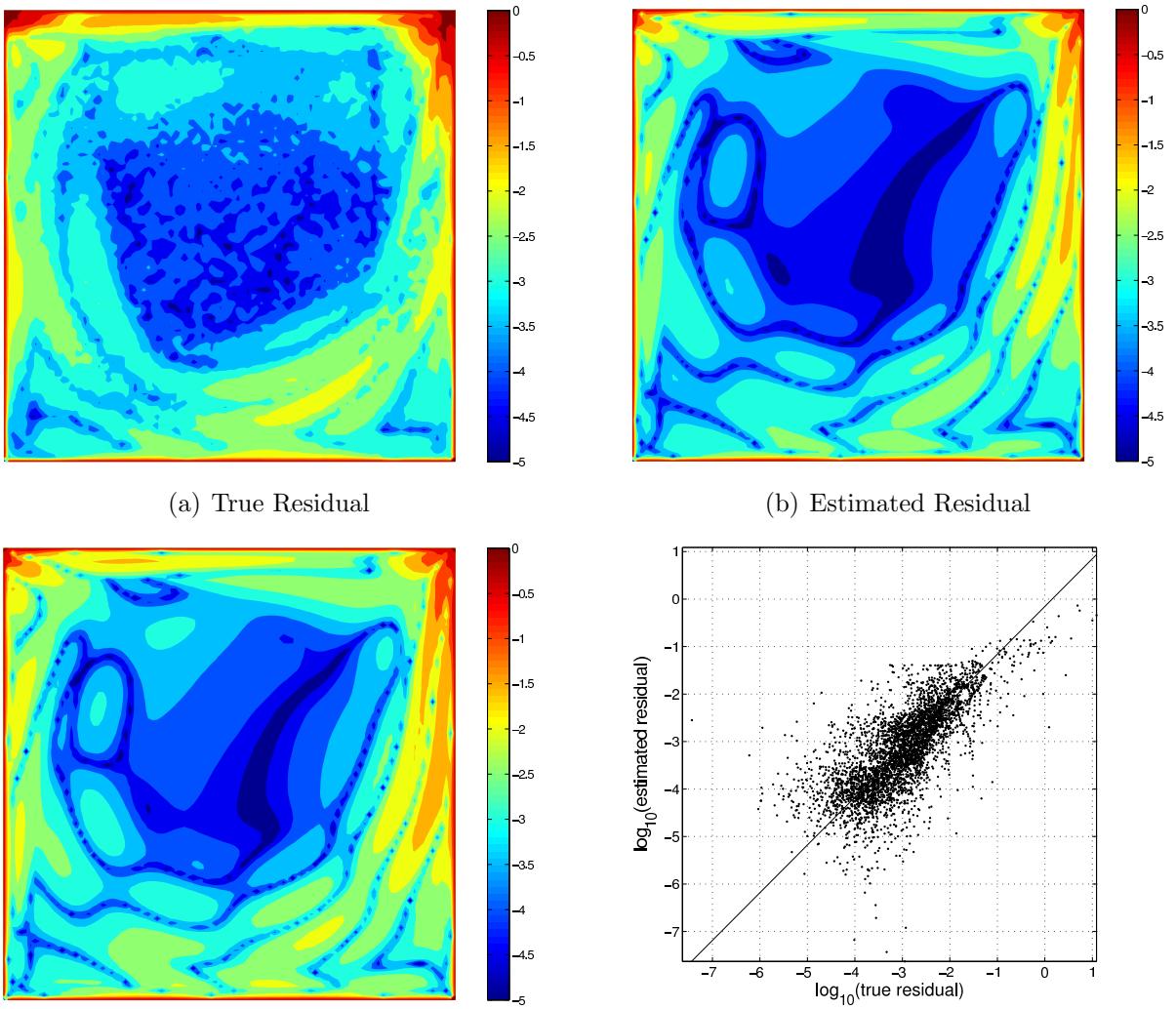
The main objective of this chapter is to demonstrate that the solution residual is a promising error indicator for mesh adaptation purposes. In the previous section, we developed a method for estimating the solution residual. The next question is how to take advantage of the information provided by this residual estimate in mesh adaptation applications. The simplest idea is to use residual as an error indicator.

3.3.1 Residual-Based Classic h-Refinement

The main premise in this section is to use the estimated residual as the error indicator in the same way that we used velocity and pressure gradients in Chapter 2. Therefore

$$\varepsilon = |\delta| \tag{3.29}$$

where $|\delta|$ is the absolute value of the solution residual estimation. Nevertheless, the adaptation indicator, ε, is not sufficient for determining the mesh size and we also need an adaptation criterion. Let us use an equidistribution principle similar to the one in



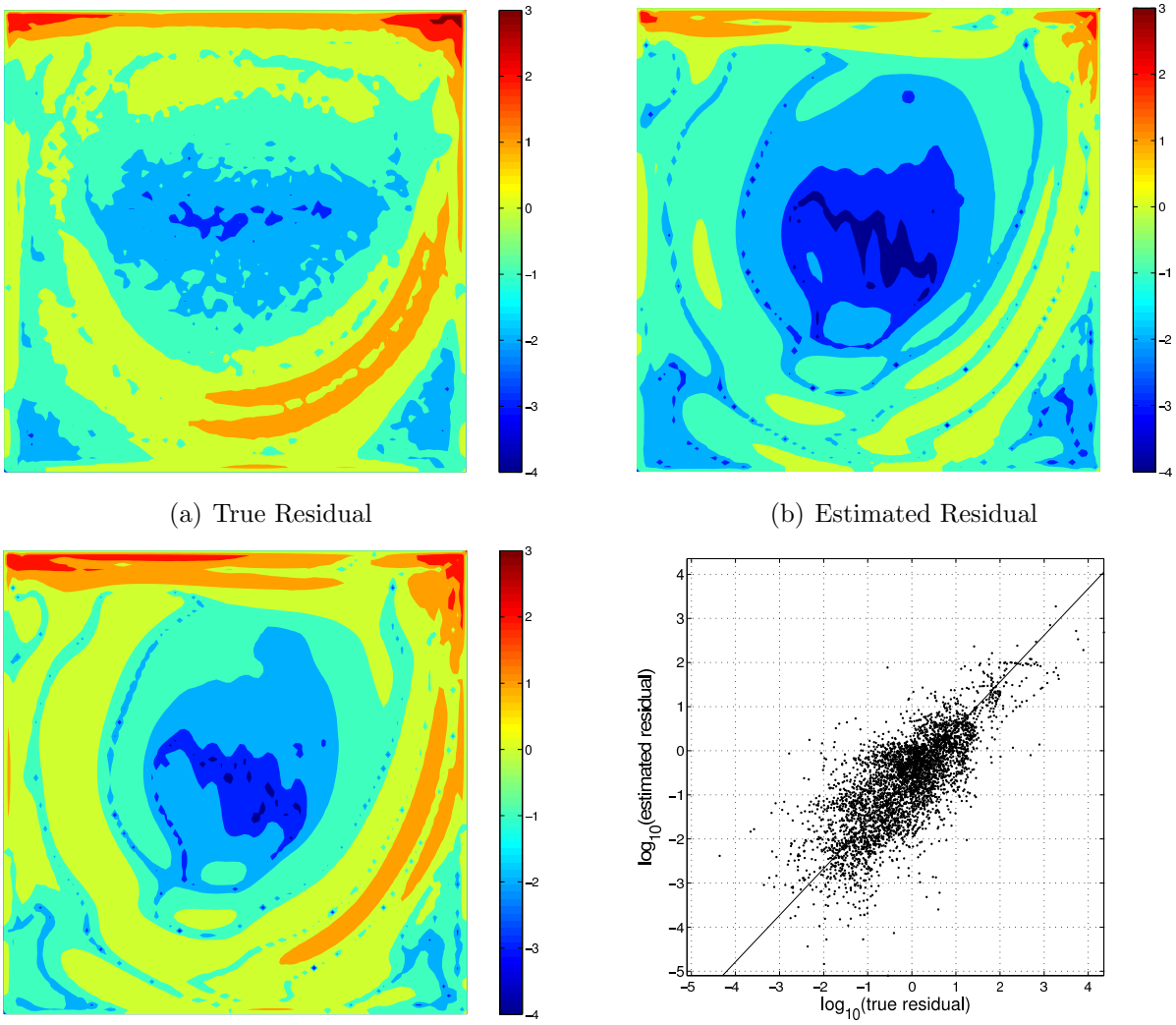
(a) True Residual

(b) Estimated Residual

(c) Scaled Estimated Residual

(d) Correlation

Figure 3.6: Residual distribution of the mass equation on the Cartesian mesh, shown in Figure 3.5, for the lid-driven cavity flow at Re = 1600.



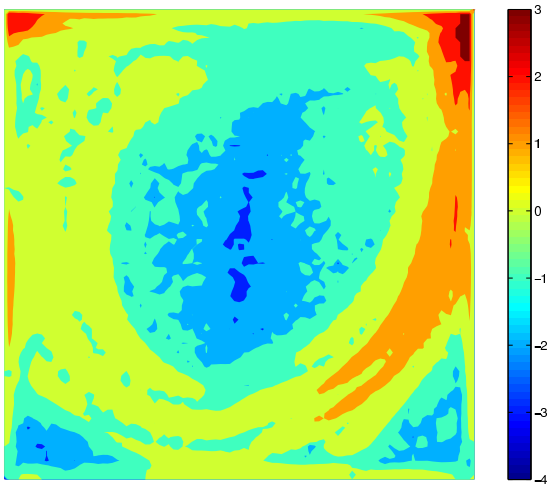
(a) True Residual

(b) Estimated Residual

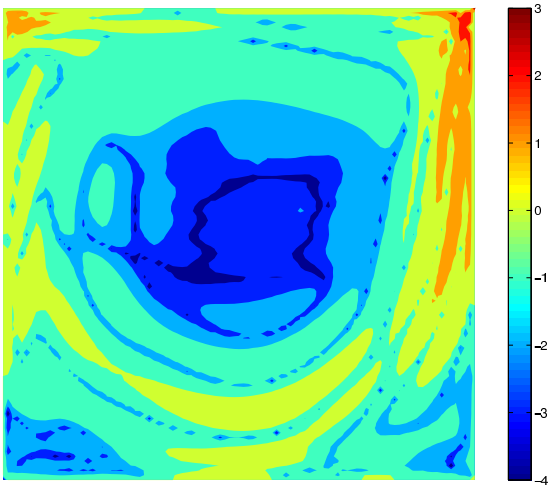
(c) Scaled Estimated Residual

(d) Correlation

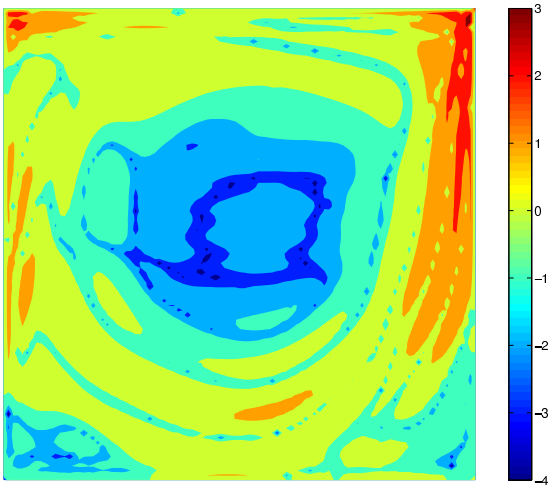
Figure 3.7: Residual distribution of the x-momentum equation on the Cartesian mesh, shown in Figure 3.5, for the lid-driven cavity flow at $\mathrm{Re} = 1600$.



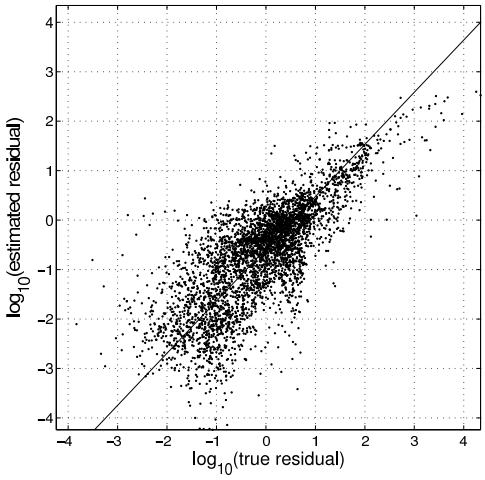
(a) True Residual



(b) Estimated Residual

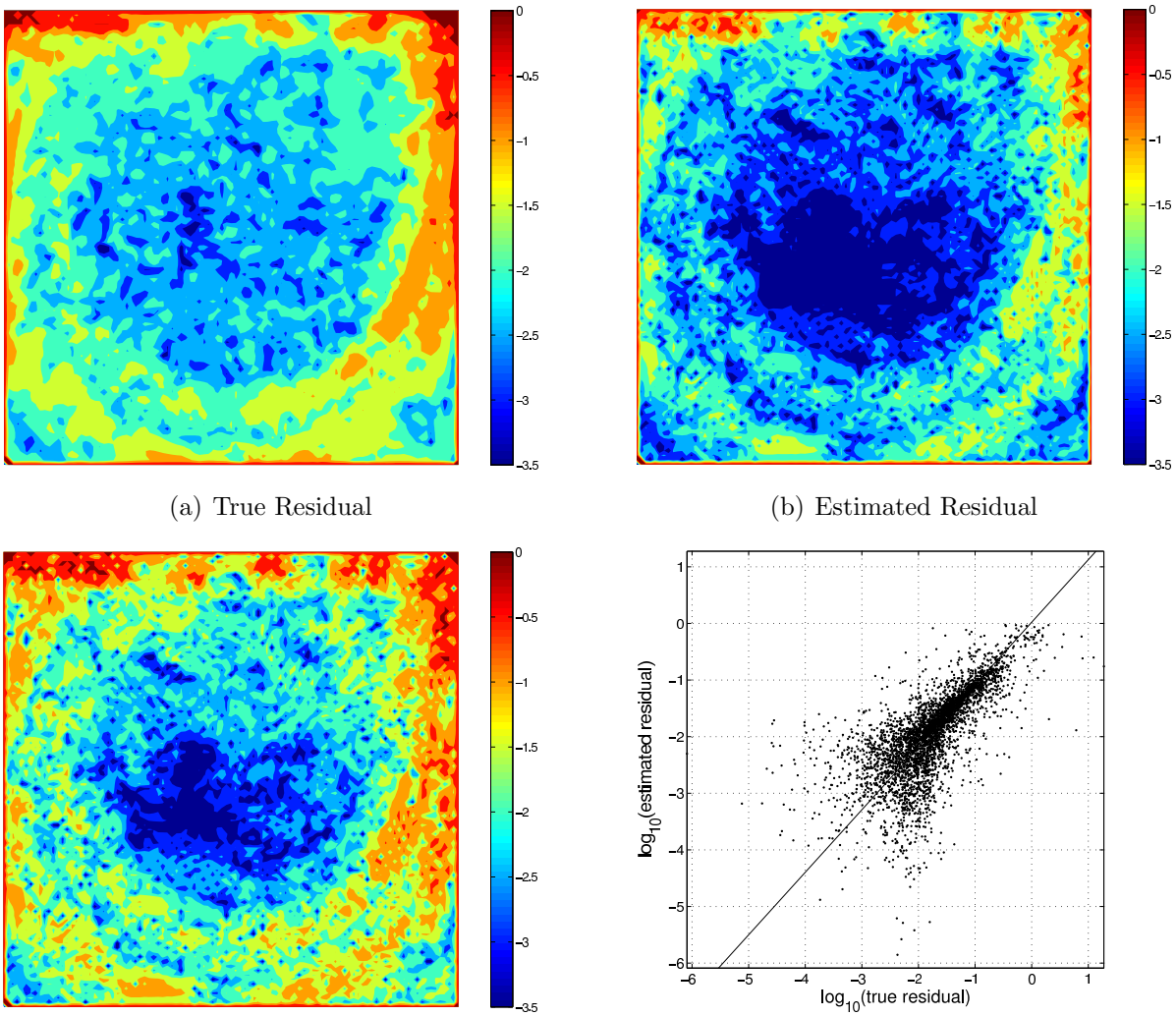


(c) Scaled Estimated Residual



(d) Correlation

Figure 3.8: Residual distribution of the y-momentum equation on the Cartesian mesh, shown in Figure 3.5, for the lid-driven cavity flow at $\mathrm{Re} = 1600$.



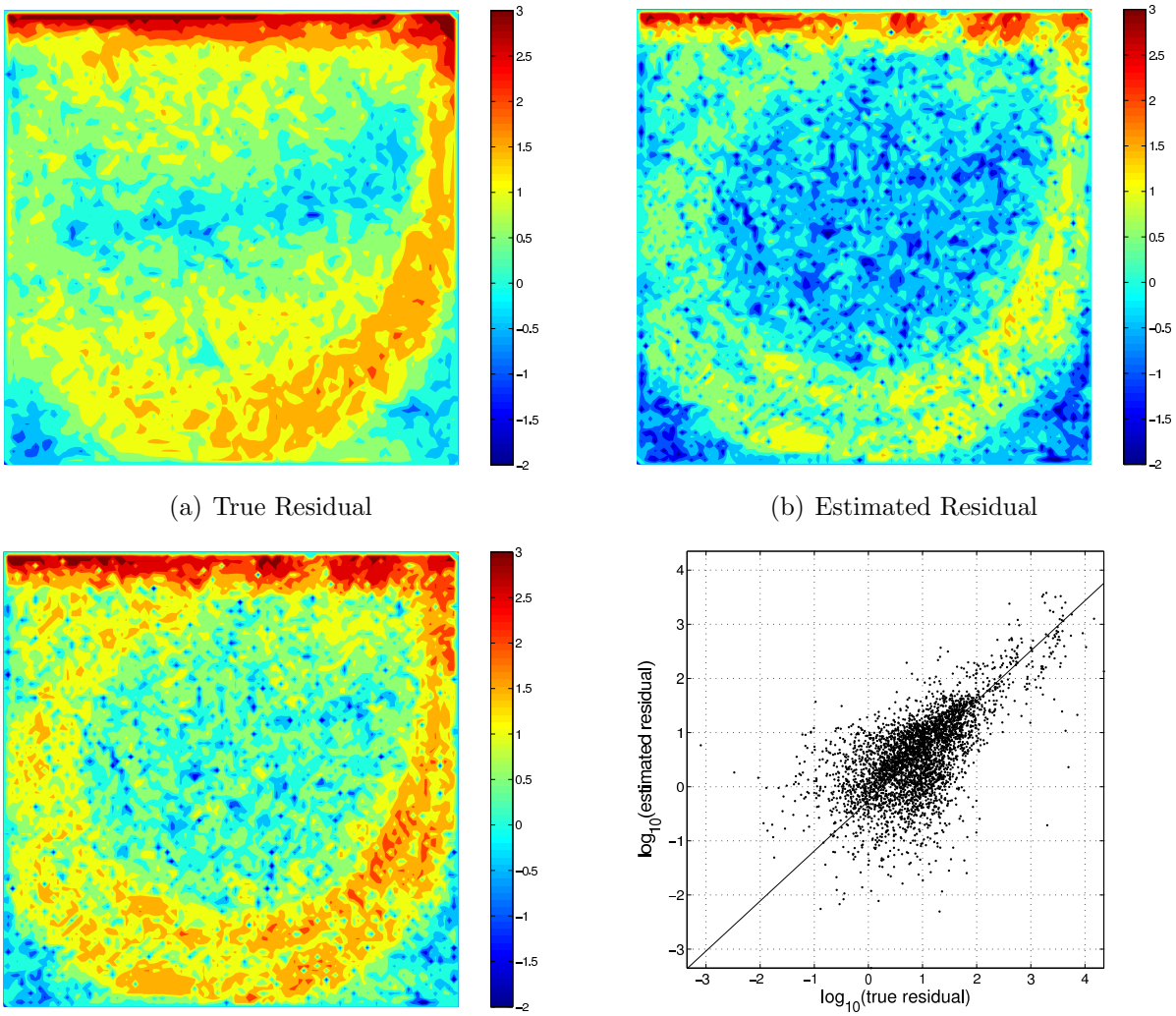
(a) True Residual

(b) Estimated Residual

(c) Scaled Estimated Residual

(d) Correlation

Figure 3.9: Residual distribution of the mass equation on the unstructured triangular mesh, shown in Figure 3.5, for the lid-driven cavity flow at Re = 1600.



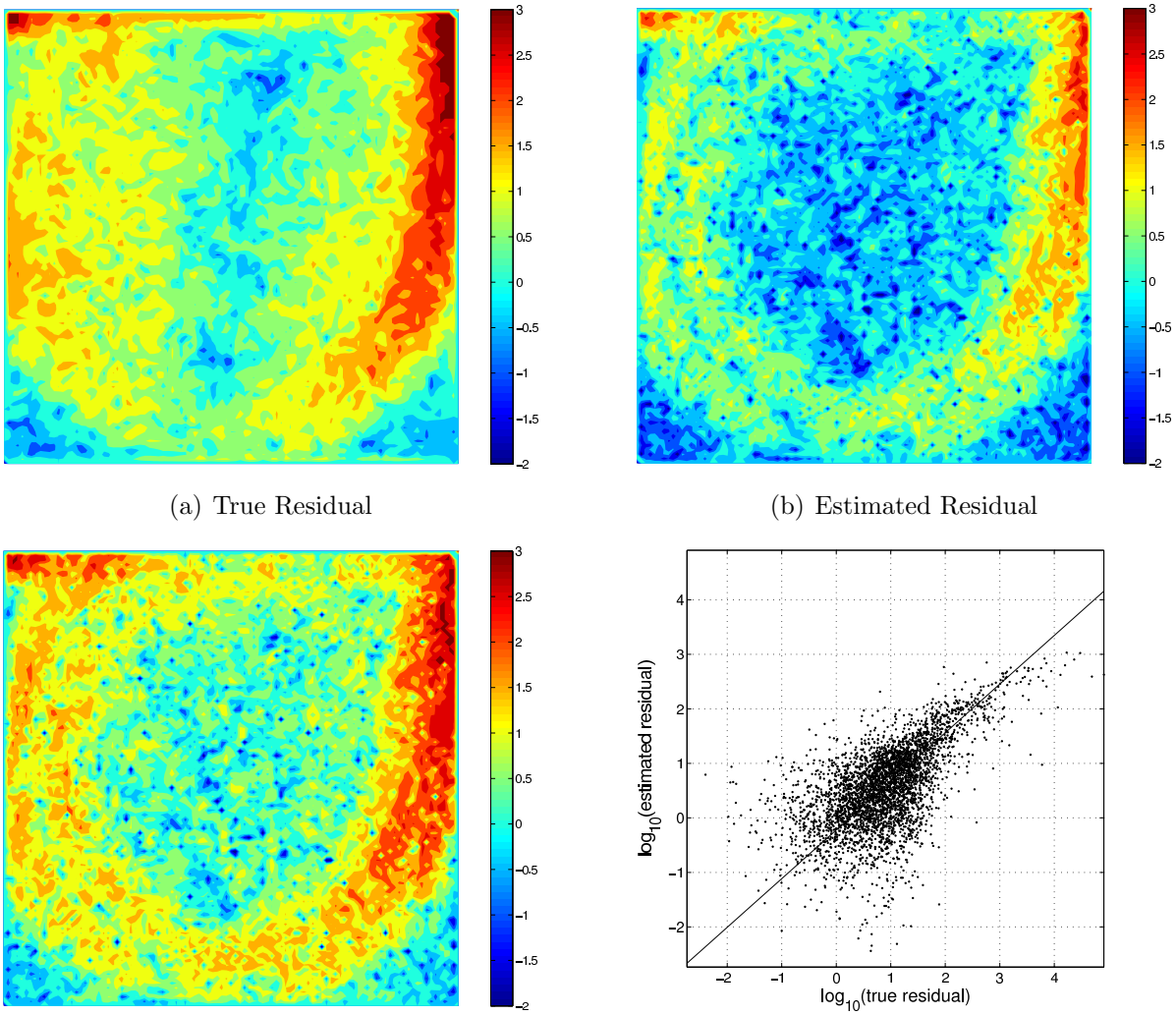
(a) True Residual

(b) Estimated Residual

(c) Scaled Estimated Residual

(d) Correlation

Figure 3.10: Residual distribution of the x-momentum equation on the unstructured triangular mesh, shown in Figure 3.5, for the lid-driven cavity flow at Re = 1600.



(a) True Residual

(b) Estimated Residual

(c) Scaled Estimated Residual

(d) Correlation

Figure 3.11: Residual distribution of the y-momentum equation on the unstructured triangular mesh, shown in Figure 3.5, for the lid-driven cavity flow at Re = 1600.

Equation (2.3) as the adaptation criterion. We must also note that the discretization scheme is second order accurate, which implies halving the mesh size results in error reduction by a factor of four. Therefore error indicator equidistribution across the domain is realized by applying the following formula:

$$h_i = h_0 \left(\frac{\bar{\varepsilon}}{\varepsilon_i} \right)^{\frac{1}{2}} \tag{3.30}$$

where h_0 is the average size of the original mesh, which the residual is estimated on, and $\bar{\varepsilon}$ is the average error indicator across the domain.

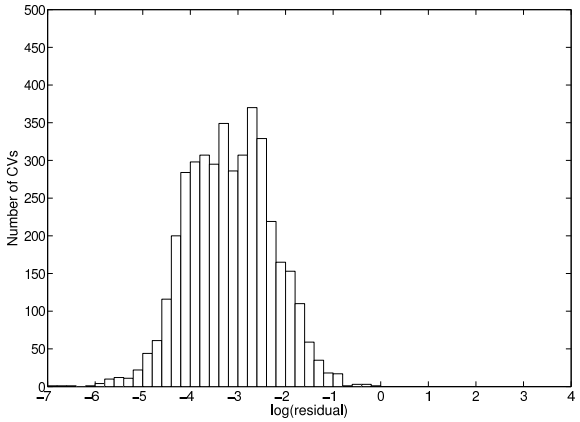
$$\bar{\varepsilon} = \left(\prod_i \varepsilon_i^{\Omega_i} \right)^{1/\Omega} = \exp\left(\frac{1}{\Omega} \sum_i \log(\varepsilon_i) \Omega_i \right) \tag{3.31}$$

where the subscript i represents the i-th control volume and Ω is the volume of the entire physical domain. Note that in the latter equation, we define $\bar{\varepsilon}$ as the geometric average of ε throughout the domain. The reason lies in the residual scaling behaviour, shown in Figures 3.12(a) through 3.12(f) for the cavity flow at Re = 1600. As seen, the histograms of the residual reveals a Gaussian-like pattern in logarithmic scale. In other words, residual has a logarithmic distribution rather than a linear distribution. Therefore it is more reasonable to apply the arithmetic averaging in the logarithmic scale, which is equivalent to a geometric averaging in Equation (3.31).

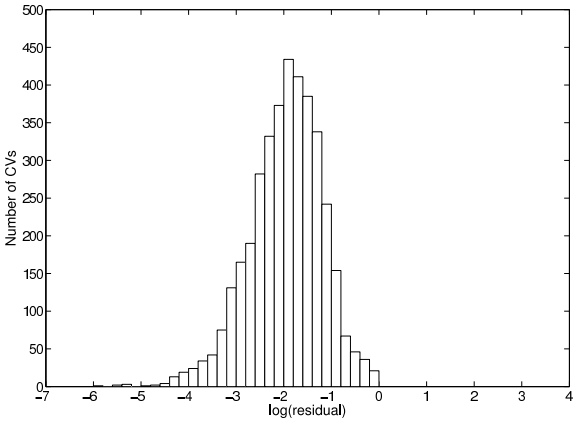
Equation (3.30) determines the desirable characteristic size of the adapted mesh based on the solution residual. However, there are three residual components associated with the mass, x-momentum, and y-momentum equations. Therefore we have to apply Equation (3.30) to all those three components and calculate the corresponding values of h, which are h_{mass}, $h_{x-\text{mom}}$, and $h_{y-\text{mom}}$, respectively. Then we pick the smallest of the three as the desirable mesh size.

$$h = \min\left(h_{\text{mass}}, h_{x-\text{mom}}, h_{y-\text{mom}} \right) \tag{3.32}$$

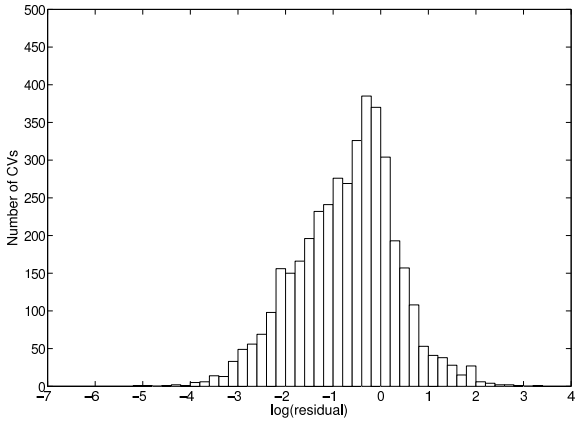
The last component in an adaptation method is the adaptation mechanism, which is a mechanism to modify the original mesh in order to obtain the adapted mesh whose characteristic size satisfies Equation (3.32). In this case, the only feasible mechanism is



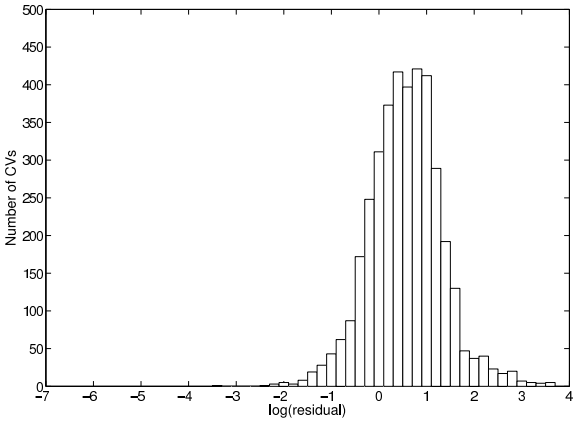
(a) Residual of mass on Cartesian mesh



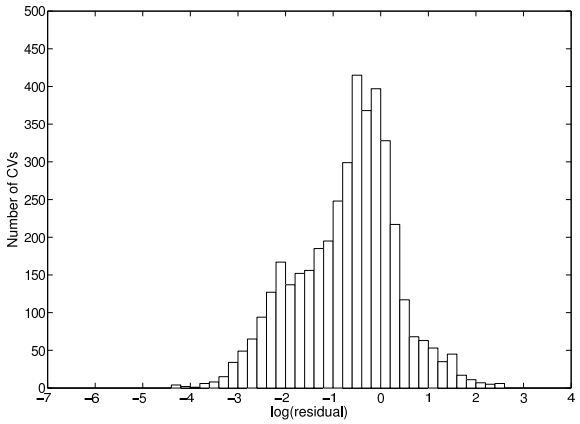
(b) Residual of mass on triangular mesh



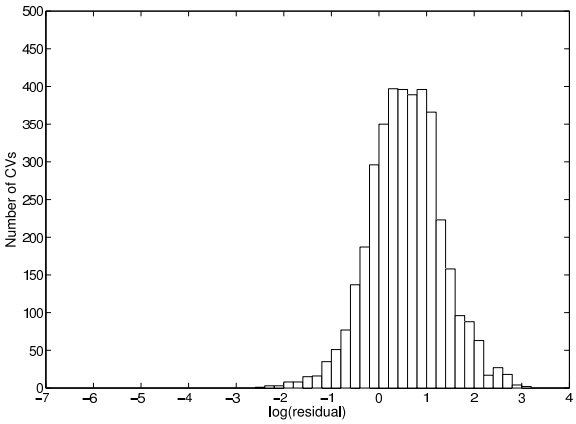
(c) Residual of x-momentum on Cartesian mesh



(d) Residual of x-momentum on triangular mesh



(e) Residual of y-momentum on Cartesian mesh



(f) Residual of y-momentum on triangular mesh

Figure 3.12: Histograms of the residuals of the mass, x-momentum, and y-momentum equations on Cartesian and triangular meshes for the cavity flow at $\mathrm{Re} = 1600$.

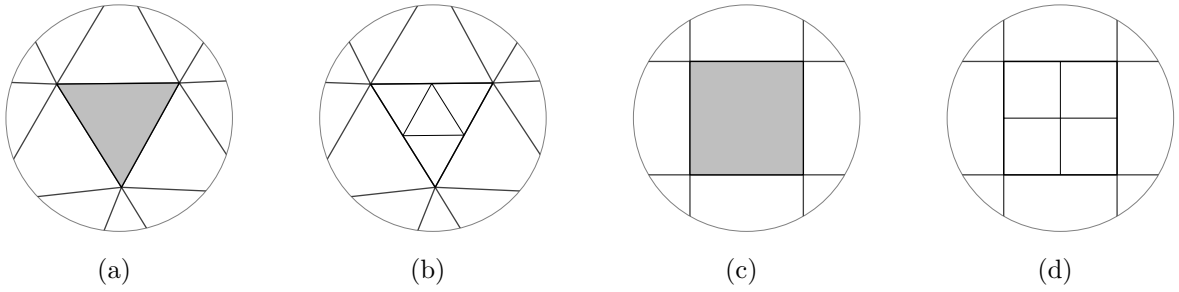


Figure 3.13: Schematic of classic h-refinement mechanism for triangular mesh, (a) to (b), and quadrilateral mesh, (c) to (d).

a classic h-refinement, which splits a control volume into four smaller control volumes if its residual is large but does not change the position of mesh vertices. Let us explain why the classic h-refinement mechanism is the only feasible mechanism in this case.

The method that we developed in the previous section can only estimate the solution residual on a *given mesh*. Therefore the estimated residual is associated with the geometry of the mesh. This issue will be better understood in the discussion on the mesh independent residual estimator of Chapter 4. Indeed any mesh alteration that moves the mesh vertices invalidates the residual estimate. In this regard the residual-based indicator is different from a feature-based indicator since a feature-based indicator depends weakly on the mesh. The only way to circumvent this problem is to retain the geometry of the original mesh and split some control volumes into smaller ones.

Figure 3.4 shows that mesh refinement through subdivision of control volumes improves the solution accuracy. As seen, the face flow calculation on the original mesh is still valid on the refined mesh provided that the original mesh vertices are retained. The only way that we can refine the mesh while retaining the geometry is to split each control volume into smaller ones. Figure 3.13 shows this splitting process for both triangular and quadrilateral control volumes. Unfortunately, it is difficult to implement mesh coarsening in this mechanism. However, Hay and Visonneau [48] propose a method that can coarsen the mesh through agglomerating neighbouring control volumes.

Figure 3.13 shows only one level of refinement. We can apply the above adaptation

mechanism recursively to the refined control volumes as well to reach two or more levels of refinement. Therefore it may be more convenient to write Equation (3.30) in terms of the number of required refinement levels.

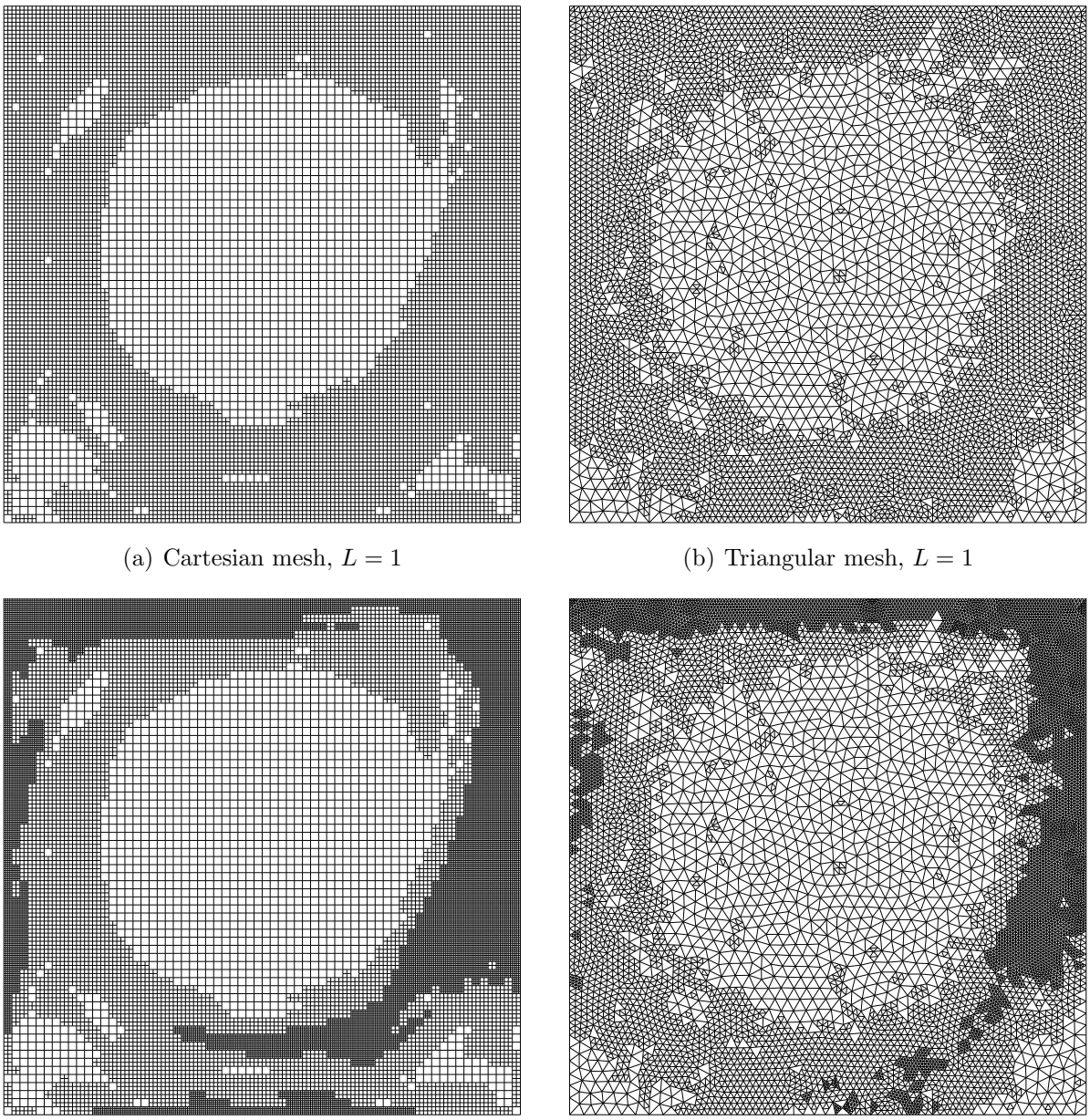
$$L_i = \left\lfloor \log_4 \left(\frac{\varepsilon_i}{\bar{\varepsilon}} \right) \right\rfloor \tag{3.33}$$

where L is the number of refinement levels and $\lfloor\ \rfloor$ is the floor function. The above formula tells us the required number of refinement levels for the each control volume as a function of its residual.

3.3.2 Application Results and Discussion

In this section, we apply the method of the previous section to the lid-driven cavity flow at $\mathrm{Re} = 1600$ on two different meshes, shown in Figure 3.14. Figures 3.14(a) through 3.14(d) show the refined meshes up to two levels of refinement. As seen, all meshes exhibit similar features. The residual is large at the top and right walls and consequently the mesh is refined in those regions. In contrast, the resolution of the original mesh is sufficient for the central region and the bottom corners of the cavity and therefore no refinement occurs in those regions. Although these observations are consistent with what we expected, the visual assessment of the meshes in Figure 3.14 is not sufficient and we need to analyse the error reduction on the refined meshes. This quantitative analysis of the error helps us to evaluate the performance of the h-refinement method.

To analyse the error reduction on the refined meshes we may follow the same technique as of Section 2.2. In this technique, we compare the kinetic energy, K^*, error on the refined meshes to that of a uniform mesh with the same number on control volumes. Tables 3.2 and 3.3 show the results for the refined triangular and Cartesian meshes, respectively. The parameter L represents the maximum number of refinement levels; $L = 0$ for the case of no refinement, $L = 1$ for the case of one level of control volume subdivision, and so on. We may summarize the results of Tables 2.1, 3.2, and 3.3 into Figure 3.15. This figure shows the error of K^* versus the number of control volumes for both the uniform and refined Cartesian and triangular meshes. As seen, there is a consistent error reduction for the refined meshes. However, the the performance of the triangular mesh with two level



(a) Cartesian mesh, $L = 1$

(b) Triangular mesh, $L = 1$

(c) Cartesian mesh, $L = 2$

(d) Triangular mesh, $L = 2$

Figure 3.14: Classic h-refinement for the cavity flow at $\mathrm{Re} = 1600$ on uniform Cartesian and unstructured triangular meshes.

of refinement, $L = 2$, is slightly better than that of the Cartesian mesh with two levels of refinement.

Table 3.2: Nondimensional total kinetic energy, K^*, for the cavity flow at Re = 1600 on two refined triangular meshes

Level	Number of CVs (n)	K^*	Error
$L = 0$	3827	0.083155	9.23%
$L = 1$	10157	0.090220	1.52%
$L = 2$	17913	0.091228	0.418%

Table 3.3: Nondimensional total kinetic energy, K^*, for the cavity flow at Re = 1600 on two refined Cartesian meshes

Level	Number of CVs (n)	K^*	Error
$L = 0$	4096	0.085868	6.27%
$L = 1$	11485	0.089875	1.94%
$L = 2$	23113	0.090980	0.689%

A comparison between the performance of the residual-based h-refinement of Figures 3.14 and the feature-based mesh adaptation of Figure 2.3 reveals that in spite of the promising results, the residual-based h-refinement method is not as effective as a much simpler gradient-based adaptation method. The problem is that in the above h-refinement method, the vertices of the original mesh must be retained and only subdivision of control volumes is permitted. Therefore we cannot take advantage of other adaptation mechanisms such as mesh coarsening and vertex repositioning. In other words, the above adaptation method is not robust enough with respect to the initial mesh. The natural extension to the above line of argument is the development of a residual-based mesh adaptation method that can produce the adapted mesh independent of the original mesh.

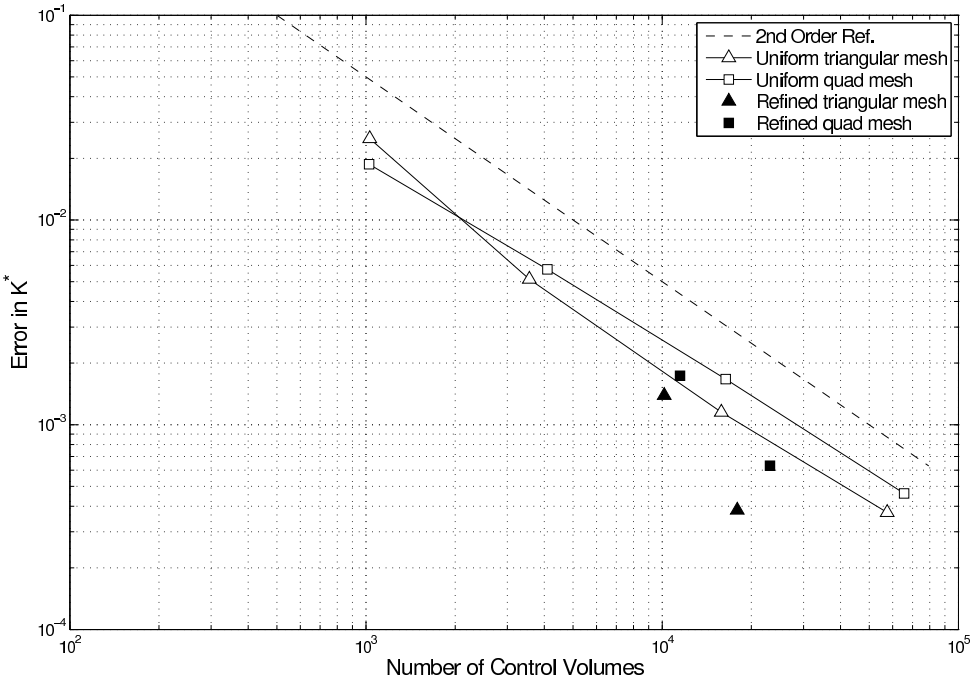


Figure 3.15: Comparison between the error on uniform and refined meshes for the lid-driven cavity flow at Re = 1600.

3.4 Residual-Based Adaptation Independent of Initial Mesh

The example of the previous section shows that the reconciliation of a residual-based error indicator and a geometry altering adaptation mechanism that removes or repositions the mesh vertices is a challenging issue. In the mesh adaptation literature, the common approach for addressing this incompatibility is through using an iterative residual estimation method. Let us explain this approach in more detail.

3.4.1 Iterative Residual Estimation Approach

Some researchers use an iterative approach along with a residual-based adaptation indicator to drive the adaptation process. In this approach, the mesh may change during the adaptation process while the residual is updated at each iteration [75, 69, 70].

Roe and Nishikawa [75] apply the above method to inviscid compressible and incompressible flows. In the case of incompressible flows, they use the Cauchy-Riemann system as the model equations and use a second order accurate vertex-centred finite volume method on unstructured meshes for the numerical solution. Their residual estimation is based on a third-order accurate Fluctuation-Splitting Method [25]. Then, they move the mesh vertices (r-adaptation) in order to minimize the residual. They show that residual minimization is effective in enhancing the solution accuracy. However, the convergence rate is slow and takes many thousands of iterations.

Reuss [69] and Reuss and Stubley [70] use an iterative residual-based adaptation method for incompressible viscous flows. Their solution method is based on a first or second-order accurate cell-centred finite volume method on unstructured meshes. They estimate face flow errors based on a Taylor series analysis and try to minimize it through adaptation. They use vertex movement, r-adaptation mechanism, along with a steepest descent method to minimize the residual. They also report that the convergence of the adaptation method is slow.

Although the above iterative methods prove to be effective, they suffer from a few shortcomings. The iterative methods are usually slow and it takes thousands of iterations until convergence. Considering the fact that at every iteration we have to update the residual, we can conclude that iterative methods are inefficient for real-world problems.

We can accelerate iterative methods by adding h-refinement. However, the calculation of residual for such a method would be even more complicated. The question is how to estimate and minimize the solution residual on the adapted mesh without iteration. Mesh independent residual estimation method addresses this issue and enables us to implement a direct mesh adaptation method.

3.4.2 Direct Residual Estimation Approach

The alternative approach to the iterative residual estimation method is the direct approach, which takes advantage of a mesh independent residual estimator. In this approach, we derive a residual estimator for certain kinds of meshes. Let us explain the idea of mesh independent residual estimator based on mesh geometric characteristics.

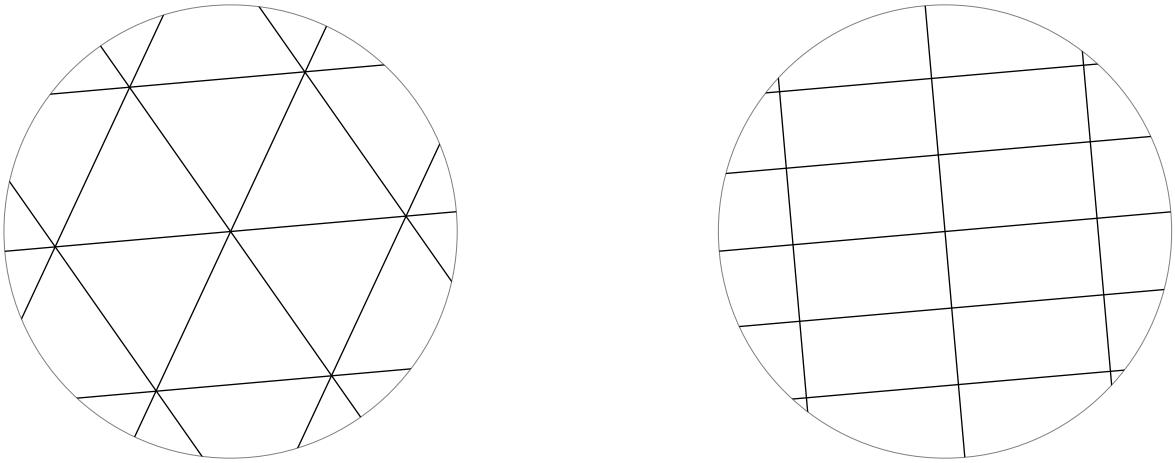
The solution residual depends on both the physical solution and the mesh geometry. Nonetheless, on a given mesh the residual only depends on the physical solution since the mesh geometry is fixed. Therefore we can propose a residual estimator on this mesh. Unfortunately, such a residual estimator is not suitable for mesh adaptation since it only works on a single given mesh. A better approach is to encapsulate the mesh geometric characteristics into a handful of parameters and develop the residual estimator based on those parameters.

Consider the example of a triangular uniform isotropic mesh. We say a mesh is uniform and isotropic if and only if the characteristic size of its cells is translation and rotation invariant, respectively. In other words, the cells must be equilateral triangles of the same size throughout the domain, as shown in Figure 3.16(a). The only parameter required for characterizing such a mesh is the size of cells, h. As a result, we can take advantage of this kind of simplification to devise a mesh independent residual estimator.

3.5 Towards a Direct Residual-Based Mesh Adaptation

A prerequisite for direct residual based adaptation methods is to put some constraints on mesh geometry. In the context of CFD, two types of mesh geometries are very popular: triangular isotropic and quadrilateral anisotropic meshes, shown in Figures 3.16(a) and 3.16(b), respectively.

A triangular isotropic mesh is composed of nearly equilateral triangles. As mentioned above, the only required parameter to characterize such a mesh is the cell size, h, as a function of position. Therefore we can define a residual-based isotropic error indicator of



(a) Uniform isotropic triangular mesh

(b) Uniform anisotropic quadrilateral mesh

Figure 3.16: Examples of uniform triangular and quadrilateral meshes

the form:

$$\varepsilon_{\text{iso}} = f(h, \text{physical solution})$$

where ε_{iso} is the error indicator. We will discuss how to use the above equation in mesh adaptation in Chapter 5. However, the basic idea is to use the above equation for determining mesh characteristic size, h, so that the error indicator, ε_{iso}, is minimized.

A quadrilateral anisotropic mesh is composed of elongated rectangles. This kind of mesh is very popular for resolving highly directional flow features such as shear layers and shock waves. We cannot characterize this kind of mesh using only a single parameter since the aspect ratio and orientation of each cell are important. Therefore we need at least three parameters to characterize such a mesh and the residual estimator would be a function of the physical solution and these three mesh related parameters.

$$\varepsilon_{\text{anis}} = g(h, A, \theta, \text{physical solution})$$

where $\varepsilon_{\text{anis}}$ is the anisotropic error indicator. We will discuss how to use the above equation in anisotropic mesh adaptation in Chapter 6. However, the basic idea is very similar to that of the isotropic adaptation; we use the above equation for determining h, A, and θ so that the error indicator, $\varepsilon_{\text{anis}}$, is minimized.

Chapter 4

Residual-Based Isotropic Mesh Adaptation

In the previous chapter, we outlined a method for estimating the residual of a finite volume solution. We explained how to use such a residual estimate as the error indicator in the classic h-refinement. In spite of the effectiveness of the outlined residual-based error indicator, its performance was inferior to the simpler velocity-gradient-based error indicators of Chapter 2. However, this inferior performance was not due to an inherent limitation of the residual-based error indicator but was rather due to the rigidity of the mesh adaptation mechanism, classic h-refinement, which prevented the adaptation method from taking full advantage of the information provided by the residual-based error indicator. In this chapter, we try to integrate the residual-based error indicator of the previous chapter with more flexible adaptation mechanisms in order to improve the mesh adaptation performance.

The residual estimation method, presented in the previous chapter, is based on calculating face flow errors. A brief examination of the face flow error equations, derived in Appendix C, shows that face flow errors are not only functions of physical variables and their derivatives, but also functions of mesh geometry. Therefore the residual estimation method of the previous chapter cannot be used during the mesh adaptation process since the mesh geometry changes. To address this issue we have to characterize mesh geometry using some parameters and develop a residual-based error indicator based on these

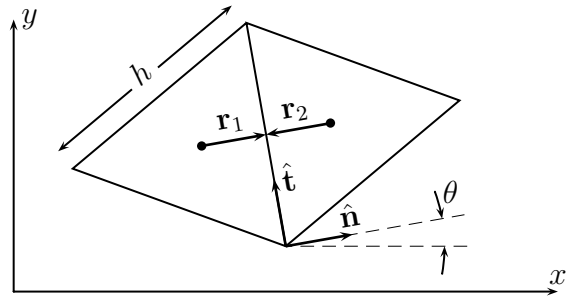


Figure 4.1: Schematic of an interior face between two control volumes in an isotropic triangular mesh

parameters.

In this chapter, we develop a residual-based error indicator for isotropic triangular meshes. The reason that we concentrate on isotropic triangular meshes is that they are easy to generate and popular in CFD applications. In addition, the geometric characterization of isotropic triangular meshes is straightforward using the scalar field of local cell sizes, h, throughout the physical domain. Once we develop the residual-based error indicator, we use it in mesh adaptation and evaluate its performance.

4.1 Residual Estimator for Isotropic Triangular Meshes

An isotropic mesh is a mesh in which the characteristic size of mesh cells is rotation invariant. In other words, the cells must be nearly regular polygons in two dimensions or polyhedrons in three dimensions. The advantage of isotropic meshes from the geometrical point of view is that we can characterize them using a single quantity such as the local cell size. The simplest case of an isotropic mesh is a triangular mesh with nearly equilateral triangular cells. Figure 3.5 shows an example of an isotropic triangular mesh. Isotropic triangular meshes are very popular in CFD since they are easy to generate and also their properties are well-understood thanks to the huge body of research on them in the CFD literature. Therefore in this section, we limit our discussion to residual estimation for isotropic triangular meshes and its use in isotropic mesh adaptation.

The method for estimating the solution residual on an isotropic triangular mesh is a

simplified variant of the more general residual estimation method of Chapter 3. In this simplified variant, we use mesh geometric parameters to simplify the formulation of the residual estimation method. Let us explain the concept using Figure 4.1. This figure shows the schematic of an interior face in an isotropic triangular mesh. The assumption of mesh isotropy implies that the mesh cells are nearly equilateral triangles. We can use this assumption to simplify the residual estimation method. For example in Figure 4.1, the nodes of control volumes 1 and 2 and the face integration point lie on a straight line, which is perpendicular to the face. This arrangement, which is called *orthogonal*, greatly simplifies the residual estimation and causes a few terms to vanish in the residual estimation method, presented in Appendix C. The other assumption that we make for simplifying the residual estimation method is that the local distribution of solution variables is quadratic. In other words, we neglect the cubic and higher order terms in the Taylor series expansion of the solution variables about the node of each control volume. The details of this simplification is presented in Appendix G and we do not repeat it here for the sake of brevity. The final results of these simplifications are residual estimation formulae in which the mesh geometry is characterized by two parameters:

$$\delta \approx h^3 \sin(3\theta + \theta_0)\mathbf{f}(\text{physical solution})$$

where h is the local cell characteristic size, θ is the local cell orientation, and θ_0 is a solution dependent orientation angle at which the residual is zero. If $\sin(3\theta + \theta_0) = 1$ the residual δ becomes locally maximized, which is the most critical orientation for mesh cells. Therefore we can find an upper bound for the local residual as a function of mesh size and physical solution. Note that the physical solution is a strong a function of position and only a weak function of mesh geometry. As a result if we have the physical solution on a reasonable initial mesh, we can interpolate the solution to any other mesh and find the physical solution at a specific position (x, y). Therefore we obtain the following formula for the upper bound for the solution residual of an isotropic triangular cell.

$$\boldsymbol{\delta} \approx h^3 \mathbf{f}(\text{physical solution}) \tag{4.1}$$

Note that the function $\mathbf{f}$ in the latter is a vector function with three components for the

mass, x-momentum, and y-momentum equations:

$$\mathbf{f} = \begin{pmatrix} f_{\text{mass}} \\ f_{x-\text{mom}} \\ f_{y-\text{mom}} \end{pmatrix} \tag{4.2}$$

Functions f_{mass}, $f_{x-\text{mom}}$, and $f_{y-\text{mom}}$ are derived in Appendix G and expressed in Equations (G.9), (G.13), and (G.17), respectively. For the sake of clarity, let us repeat the final results here.

$$f_{\text{mass}} = \frac{\rho}{32}\Bigg[\left(-\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + 4\frac{\partial^2 u}{\partial x \partial y} + \frac{\partial^2 v}{\partial x^2} - \frac{\partial^2 v}{\partial y^2}\right)^2 + \left(-\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + 4\frac{\partial^2 v}{\partial x \partial y} + \frac{\partial^2 v}{\partial x^2} - \frac{\partial^2 v}{\partial y^2}\right)^2 \Bigg]^{\frac{1}{2}} \tag{4.3}$$

$$f_{x-\text{mom}} = \frac{1}{16}\Bigg\{ \Bigg[\rho\left(\frac{\partial u}{\partial y}^2 - \frac{\partial u}{\partial x}^2 + \frac{\partial u}{\partial y}\frac{\partial v}{\partial x} + \frac{\partial u}{\partial x}\frac{\partial v}{\partial y}\right) + \rho u\left(\frac{\partial^2 u}{\partial y^2} - \frac{\partial^2 u}{\partial x^2} + 2\frac{\partial^2 v}{\partial x \partial y}\right) + \left(\frac{\partial^2 p}{\partial y^2} - \frac{\partial^2 p}{\partial x^2}\right)\Bigg]^2 + \Bigg[\rho\left(\frac{\partial u}{\partial x}\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\frac{\partial v}{\partial y} + 2\frac{\partial u}{\partial x}\frac{\partial u}{\partial y}\right) + \rho u\left(u\frac{\partial^2 v}{\partial x^2} - u\frac{\partial^2 v}{\partial y^2} + 2u\frac{\partial^2 u}{\partial x \partial y}\right) + \left(2\frac{\partial^2 p}{\partial x \partial y}\right)\Bigg]^2 \Bigg\}^{\frac{1}{2}} \tag{4.4}$$

$$f_{y-\text{mom}} = \frac{1}{16}\Bigg\{ \Bigg[\rho\left(\frac{\partial u}{\partial y}\frac{\partial v}{\partial y} - \frac{\partial u}{\partial x}\frac{\partial v}{\partial x} + 2\frac{\partial v}{\partial x}\frac{\partial v}{\partial y}\right) + \rho v\left(\frac{\partial^2 u}{\partial y^2} - \frac{\partial^2 u}{\partial x^2} + 2\frac{\partial^2 v}{\partial x \partial y}\right) + \left(2\frac{\partial^2 p}{\partial x \partial y}\right)\Bigg]^2 + \Bigg[\rho\left(\frac{\partial v}{\partial x}^2 - \frac{\partial v}{\partial y}^2 + \frac{\partial u}{\partial x}\frac{\partial v}{\partial y} + \frac{\partial v}{\partial x}\frac{\partial u}{\partial y}\right) + \rho v\left(\frac{\partial^2 v}{\partial x^2} - \frac{\partial^2 v}{\partial y^2} + 2\frac{\partial^2 u}{\partial x \partial y}\right) + \left(\frac{\partial^2 p}{\partial x^2} - \frac{\partial^2 p}{\partial y^2}\right)\Bigg]^2 \Bigg\}^{\frac{1}{2}} \tag{4.5}$$

One interesting fact about the above equations is the absence of viscosity coefficient, μ, in the results. Although this result is not very intuitive, it shows that the effect of viscosity on the solution residual is not through the discretization of the viscous term. The effect of viscosity on the solution residual is mediated by the velocity and pressure fields, which are in turn affected by the viscous term.

Equations (4.3) through (4.5) provide an easy way for estimating the solution residual on an isotropic mesh. Therefore we can use it as an error indicator for mesh adaptation purposes, the same way that we used velocity gradient in Chapter 2. However, let us examine the performance of the above residual estimator before moving on to the topic of mesh adaptation.

4.2 Performance of the Isotropic Residual Estimator

To examine the performance of the proposed residual estimator, we apply it to the lid-driven cavity flow at Re = 1600. Since the residual estimator of the previous section is developed for isotropic meshes, we use it for estimating the residual on a uniform triangular isotropic mesh, shown in Figure 3.5.

The process of residual estimation is straightforward using Equations (4.1) through (4.5). We need to evaluate Equation (4.1) at the node of every single control volume. This equation consists of two parts: the geometry dependent part, h^3, and the solution dependent part, $\mathbf{f}(x, y)$. The geometry dependent part is a constant for the mesh of Figure 3.5 since it is uniform. This mesh consists of 3827 control volumes. therefore $h \approx 0.024565$ assuming that the cavity size is equal to unity.

The solution dependent part of Equations (4.1) can be calculated using Equations (4.3) through (4.5). As seen in these equations, the function $\mathbf{f}$ depends on the solution and its first and second derivatives. Therefore it is not necessary to have the solution on the same mesh that we are estimating the residual on. Instead we can solve for physical variables on a different mesh and interpolate the results to the mesh that the residual is being estimated on. To rule out the effect of solution errors on coarse meshes, we use the solution on a fairly fine Cartesian mesh with 256×256 control volumes. Then we interpolate the solution obtained on this mesh to the uniform isotropic triangular mesh of

Figure 3.5. Having the interpolated values of the solution variables and derivatives at the nodes of all control volumes, we can calculate the function $\mathbf{f}(x, y)$ using Equations (4.3) through (4.5) and estimate the residual using Equation (4.1).

Figure 4.2 compares the results based on the proposed method and the true residual based on the mesh independent solution, presented in Chapter 3. As seen, there is a good qualitative agreement between the results. In all cases, the estimated residual exhibits the same qualitative features as those of the true residual. However, the estimated residual fields based on the above proposed method are smoother. This smoothness is mainly due to the simplifying assumptions for mesh geometry that eliminate any noise due to mesh irregularities.

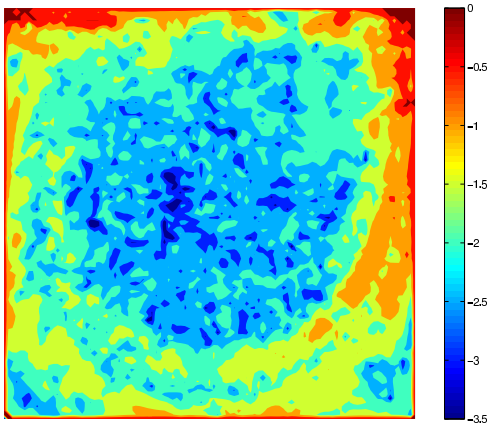
Note that in the above example, we use the solution on a 256×256 Cartesian mesh for estimating the residual on an isotropic triangular mesh with 3827 control volumes. This is an interesting observation since we estimate the residual on a given mesh based on the solution on a different mesh. In the next section, we explain how to take advantage of this property in mesh adaptation.

4.3 Criterion for Isotropic Mesh Adaptation

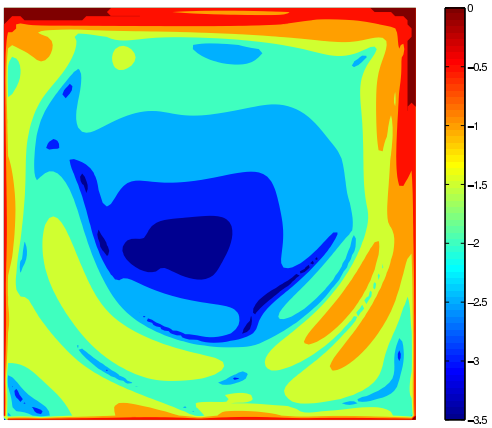
In the example of the previous section, we estimated the residual on a given mesh based on the solution on a different mesh. Indeed such an approach has strong parallels with mesh adaptation. In Chapter 3, we argued that in mesh adaptation we look for a target mesh that minimizes the solution residual based on the solution on an initial mesh. In other words, we solve a CFD problem on an initial mesh in order to obtain the physical solution and then we can use the above method to estimate the residual on any other isotropic mesh including the one that minimizes the residual. However, for identifying the mesh that minimizes the solution residual we need a criterion. In this section we develop such a criterion.

Equation (4.1) is the main tool that we need for developing the appropriate adaptation criterion. In this equation, the residual on an isotropic triangular mesh is estimated as a function of the mesh size and the physical solution.

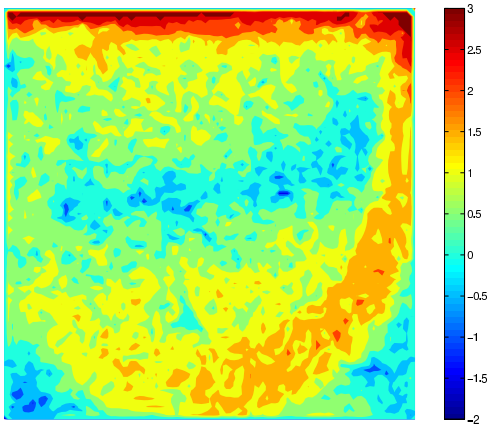
$$\delta = h^3 f(x, y)$$



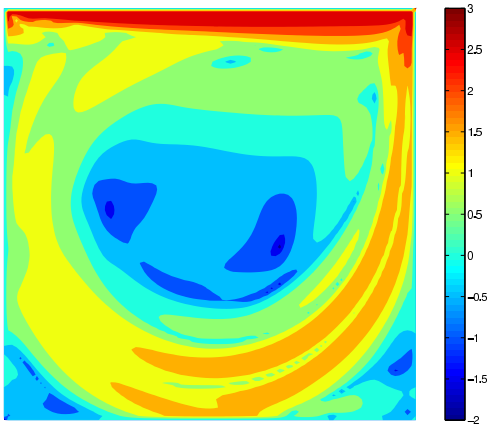
(a) True mass residual



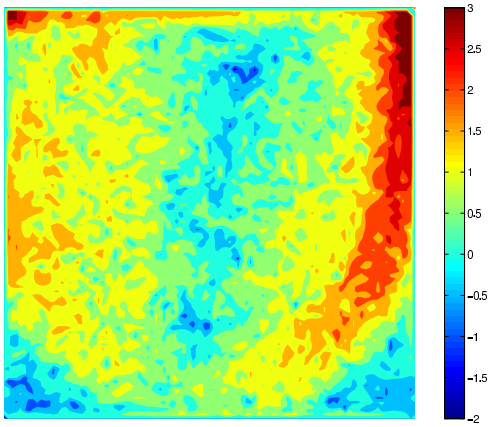
(b) Estimated mass residual



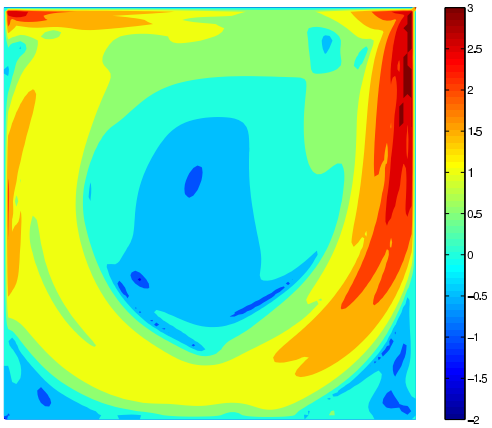
(c) True x-momentum residual



(d) Estimated x-momentum residual



(e) True y-momentum residual



(f) Estimated y-momentum residual

Figure 4.2: Comparison between the true and estimated residual distributions for the cavity flow at $\mathrm{Re} = 1600$ on an isotropic triangular mesh.

We can also use the above equation inversely to find the local mesh size, h, as a function of residual, δ. The objective here is to find $h(x, y)$ so that the residual becomes minimized throughout the domain. Then we may generate a mesh where the distribution of cell sizes complies with $h(x, y)$. In this sense, the function $h(x, y)$ is called the *size map* of the target mesh.

Let us define the total residual throughout the domain as the weighted summation of the residuals of all control volumes.

$$\Delta = \sum_i \delta_i V_i$$

where the subscript i acts on the control volume indices. Assuming that δ is a piecewise constant function, we can rewrite the above equation in the following form for a two-dimensional problem.

$$\Delta = \iint_\Omega \delta \, dxdy = \iint_\Omega h^3 f(x, y) \, dxdy \tag{4.6}$$

The objective is to find the size map h so that Δ becomes minimized. However, we must note that there is a geometric constraint on h; the total volume of the mesh cells must be equal to the domain volume. The volume of each cell in a two-dimensional isotropic mesh is approximately equal to $h^2\sqrt{3}/4$ and the following formula holds for every control volume.

$$\iint_{\mathrm{CV}} \frac{dxdy}{h^2} = \frac{\sqrt{3}}{4}$$

Therefore, for the entire domain the above formula results in the following constraint.

$$\iint_\Omega \frac{dxdy}{h^2} = \frac{\sqrt{3}}{4} N = P \tag{4.7}$$

where N is the number of control volumes. Combining Equations (4.6) and (4.7) results in,

$$\Delta + \lambda P = \iint_\Omega \left(h^3 f(x, y) + \frac{\lambda}{h^2} \right) dxdy$$

where λ is a constant, called the *Lagrange multiplier*. Minimizing the left hand side of the latter results in the minimization of Δ, since λP is a constant for meshes with a certain

number of control volumes. Therefore, our new objective is to minimize the right hand side integral. Let us define the functional ψ in the following form,

$$\psi(x, y, h) = h^3 f(x, y) + \frac{\lambda}{h^2}$$

To minimize the residual, the functional ψ must satisfy the Euler's variational formula [7],

$$\frac{\partial \psi}{\partial h} - \frac{d}{dx}\frac{\partial \psi}{\partial h_x} - \frac{d}{dy}\frac{\partial \psi}{\partial h_y} = 0 \tag{4.8}$$

In this work, we assume h is a piecewise constant function and neglect its gradient and higher order derivatives. In other words, we use a first order approximation to h. Therefore in the above variational formula, the second and third terms vanish and we obtain,

$$\frac{\partial \psi}{\partial h} = 0 \quad \Rightarrow \quad 3h^2 f(x, y) - \frac{2\lambda}{h^3} = 0$$

which results in the following solution.

$$h(x, y) = \frac{C}{f^{1/5}(x, y)} \tag{4.9}$$

where C is a constant that controls the number of cells in the target mesh. To calculate the value of C, we substitute the latter into the equation for geometric constraint, Equation (4.7), and evaluate the result.

$$C = \left(\frac{\sqrt{3}}{4N} \int f^{2/5}(x, y)\, dxdy \right)^{\frac{1}{2}} \tag{4.10}$$

Equations (4.9) and (4.10) determine the size map of an isotropic triangular mesh with N control volumes that minimizes the solution residual. In other words, Equation (4.9) is the adaptation criterion.

In Equation (4.9), it is clear that the residual distribution on the adapted mesh in not uniform. In other words, the adaptation criterion of Equation (4.9) does not follow the equidistribution principle. However, we can easily obtain the equidistribution counterpart of the above criterion by equating the residual δ in Equation (4.1) to a constant, D,

$$\delta = D \quad \Rightarrow \quad h(x, y) = \frac{D}{f^{1/3}(x, y)} \tag{4.11}$$

In the following sections we compare the performance of the above criteria.

The function $h(x, y)$ in Equations (4.9) and (4.11) determines the characteristic size of control volumes at the position (x, y) but does not provide the actual mesh. Therefore we need an adaptation mechanism, which is explained in the following section.

4.4 Isotropic Triangular Mesh Adaptation Mechanism

In a mesh adaptation method we need an adaptation mechanism to modify the mesh geometry so that the adaptation criterion is satisfied. In this section, we discuss the common adaptation mechanisms for isotropic triangular meshes in the literature. Then we explain the method that we use in this work.

4.4.1 Review of Existing Mesh Adaptation Mechanisms

In general, we can categorize mesh adaptation mechanisms into two groups: topology maintaining mechanisms and topology changing mechanisms [57].

In topology maintaining mechanisms, we reposition the mesh vertices based on some criterion in order to enhance the solution accuracy. These mechanisms, which are also called r-method, allow some mesh cells to contract while the others to expand. Therefore the average mesh characteristic size remains unchanged. In the mesh adaptation literature, there are two approaches towards implementing the criterion that drives a vertex repositioning process [57]: optimization techniques and spring analogy. In optimization techniques, a functional is defined and associated with the mesh. Then the mesh vertices are moved using optimization techniques so that the functional is minimized. For example, Reuss [69], Reuss and Stubley [70], and Roe and Nishikawa [75] use a steepest descent method for this purpose. This approach is usually preferred in the cases that we have an error indicator but no information about the geometric characteristics of the adapted mesh. In the spring analogy method, the mesh edges are modelled as elastic springs that apply force on the mesh vertices. Therefore the vertices are moved so that the network of springs reaches equilibrium. An example of this method is the adaptive moving mesh method of Palmerio [65]. The spring analogy approach is usually preferred in the cases that the mesh size map is available.

The vertex movement methods are easy to implement and can be equally applied to structured and unstructured meshes. However, these methods suffer from a few disadvantages. The biggest disadvantage of these methods is the slow convergence rate. This is due to the nonlinear effects of vertex repositioning on the solution error, which is usually treated by heavy under relaxation of the process. Moreover, vertex movement methods cannot improve the overall mesh quality since any refinement in one region comes at the cost of coarsening and even degradation of the mesh cells in other regions. In practice, there are many works in the literature that use vertex movement as a smoothing mechanism along with topology changing mechanisms [35, 45]. Therefore vertex movement is rarely used as the sole adaptation mechanism.

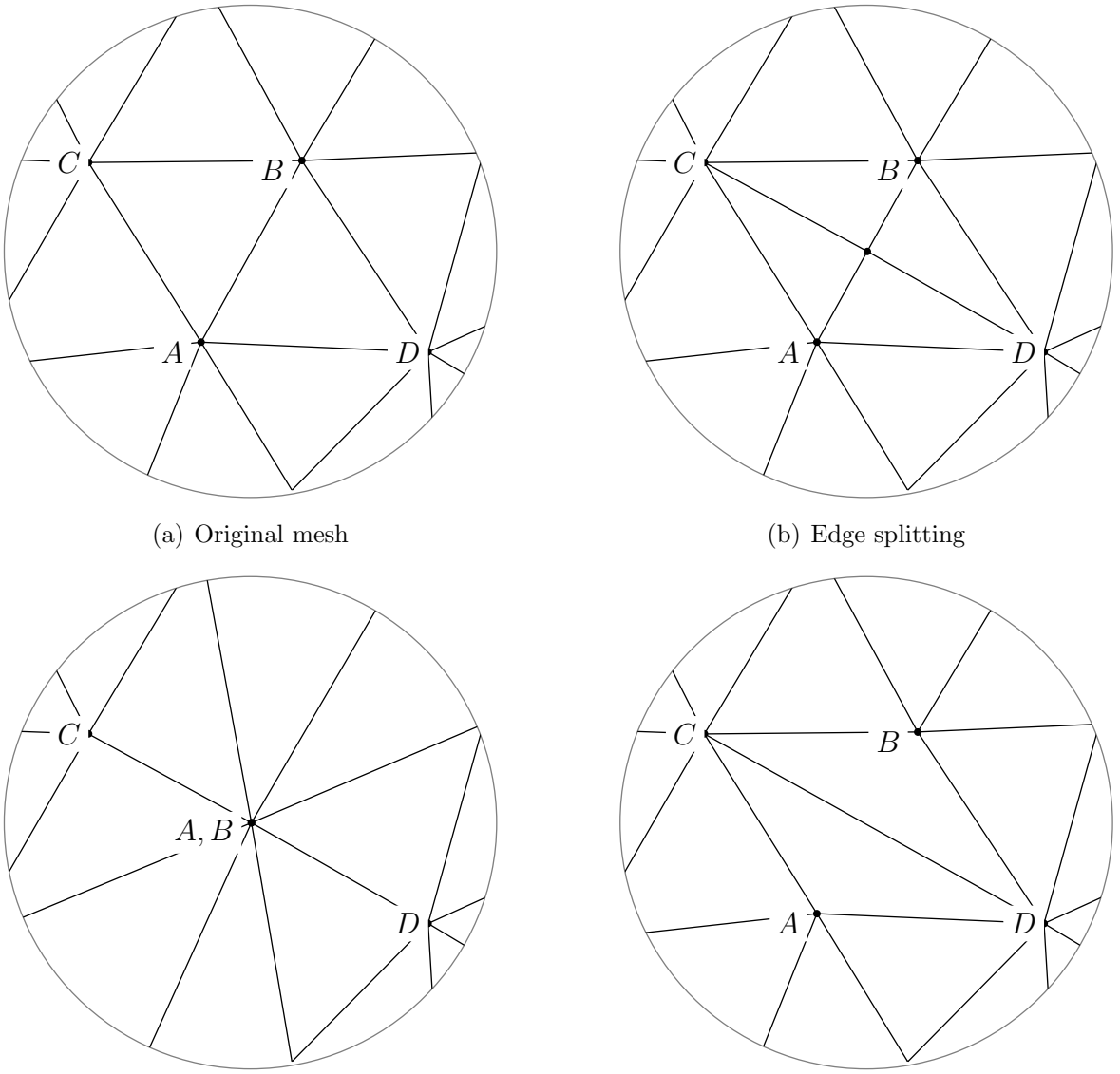
In topology changing mechanisms, also called h-methods, we insert or remove mesh entities in order to obtain the desired mesh density. The mesh adaptation method of chapter 3 is an example of this method, which is usually referred to as the classic h-refinement. However, the classic h-refinement is not the only available mechanism. Vertex insertion, vertex removal, edge subdivision, edge collapse, and cell agglomeration are the other common techniques in the literature. These mechanisms are only applicable to unstructured meshes since they alter the mesh topology. A good review of these mechanisms is given by Mavriplis [58].

The advantages and disadvantages of the above methods lead to hybrid mechanisms that are a combination of topology maintaining and topology changing mechanisms. In hybrid mechanisms, mesh refinement and coarsening is used for accelerating the mesh adaptation process while vertex movement is used for smoothing and fine tuning the mesh. Therefore we use a hybrid mechanism in this work as explained in the following subsection.

4.4.2 Implementation of the Isotropic Adaptation Mechanism

To obtain a high quality mesh, we use a hybrid method that consists of vertex repositioning, edge splitting, edge collapsing, and edge swapping. In this subsection, we explain how to combine the above components such that the resulting mesh satisfies the size map given by Equation (4.9).

For mesh refinement and coarsening, we use the edge splitting and collapsing mech-



(a) Original mesh

(b) Edge splitting

(c) Edge collapsing

(d) Edge swapping

Figure 4.3: Typical adaptation mechanisms for triangular mesh adaptation

anisms, respectively. Figures 4.3(b) and 4.3(c) illustrate these mechanisms for the edge AB in Figure 4.3(a). For this purpose, we compare the length of edge AB, to the local size map, h, and apply the above mechanisms based on the following criteria:

$$l_{\text{edge}} < \frac{h_{\text{edge}}}{\sqrt{2}} \qquad \Rightarrow \qquad \text{Collapse the edge} \tag{4.12}$$

$$l_{\text{edge}} > \sqrt{2} h_{\text{edge}} \qquad \Rightarrow \qquad \text{Split the edge} \tag{4.13}$$

where l_{edge} is the Euclidean length of the edge

$$l_{\text{edge}} = |\mathbf{r}_B - \mathbf{r}_A| = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}$$

where A and B are the end vertices of the edge, shown in Figure 4.3(a), and $\mathbf{r}_A$ and $\mathbf{r}_B$ are the position vectors of these points. In Equations (4.12) and (4.13), h_{edge} is the desired length of the edge, defined by:

$$h_{\text{edge}} = \int_0^1 h(\mathbf{r}_A + t(\mathbf{r}_B - \mathbf{r}_A))dt$$

where t is a parameter along the AB edge ranging from zero at vertex A to one at vertex B. Using these formulae, we can keep collapsing and splitting the mesh edges until there is no edge that satisfies the criteria of Equations (4.12) and (4.13). At this point the mesh may look erratic and we have to smooth it.

For mesh smoothing, we use the spring analogy technique. In this technique, we adjust the spring coefficients so that they produce attraction or repulsion between mesh vertices based on the following criteria:

$$l_{\text{edge}} < h_{\text{edge}} \qquad \Rightarrow \qquad \text{repulsion}$$

$$l_{\text{edge}} > h_{\text{edge}} \qquad \Rightarrow \qquad \text{attraction}$$

The simplest strategy for this purpose is the Laplacian smoothing [40], which assumes that the spring coefficients follow the Hooke's Law. Although this method is widely used in mesh adaptation literature, the resulting mesh may not be of desirable quality. A better approach is to use nonlinear coefficients for the springs. This approach is mainly inspired by the forces between molecules in crystals [79]. In crystals, the nonlinear forces between

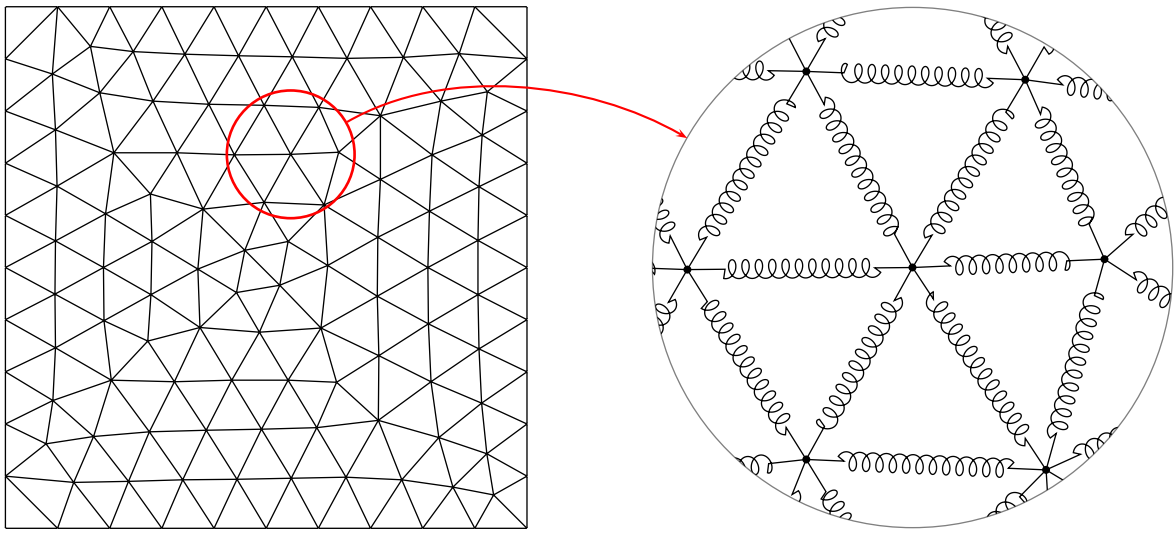


Figure 4.4: Schematic of a mesh as a network of springs

the molecules lead to regular structures that resemble isotropic meshes. Therefore we can use the same kind of forces for the springs in a mesh. A simple model for the forces between molecules is the Lennard-Jones model [55]. However, the repulsive force in the springs diverges to infinity as the spring length approaches zero. To address this issue Shimada [79] and Bossen and Heckbert [21] propose models that qualitatively behave like the Lennard-Jones model but take a finite value as the spring length approaches zero. The behaviour of these models is shown in Figure 4.5. In this work we use a modified version of the Bossen and Heckbert model that reads:

$$F = -\left[1 - \left(\frac{l_{\text{edge}}}{h_{\text{edge}}}\right)^2\right] \exp\left[-\left(\frac{l_{\text{edge}}}{h_{\text{edge}}}\right)^2\right] \tag{4.14}$$

where F is the spring force between any two vertices joined by an edge. Therefore if we pick a vertex we can calculate the force acting on it by the edges and move it accordingly.

The last components that we use in adaptation mechanism for further mesh smoothing is edge swapping, shown in Figure 4.3(d). In this mechanism, we use the Delaunay criterion [10] that minimizes the maximum angle in the mesh. In mathematical term the

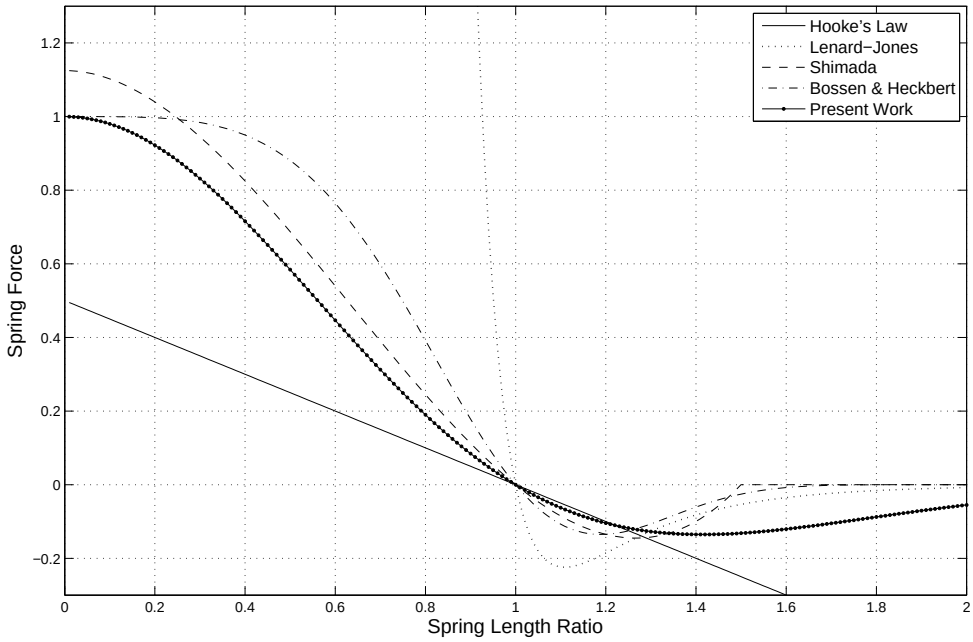


Figure 4.5: Spring force versus edge length in various Lenard-Jones like models.

Delaunay criterion is:

$$\angle CAD + \angle CBD > \pi \qquad \Rightarrow \qquad \text{swap edge } AB \text{ with } CD \tag{4.15}$$

where the AB is the edge between the vertices A and B, shown in Figure 4.3(a), and CD is the edge between the vertices C and D, shown in Figure 4.3(d).

We also need to determine how the various components are combined in the adaptation mechanism. As mentioned above, we use edge collapsing, edge splitting, vertex repositioning, and vertex swapping. In this work, we check the criteria of Equations (4.12) and (4.13) for all edges in the mesh and collapse/split them if necessary. Then we check the criterion of Equation (4.15) for all the edges and swap them if necessary. This process is followed by five levels of vertex movement and one level of edge swapping. Therefore we can summarize the process as follows:

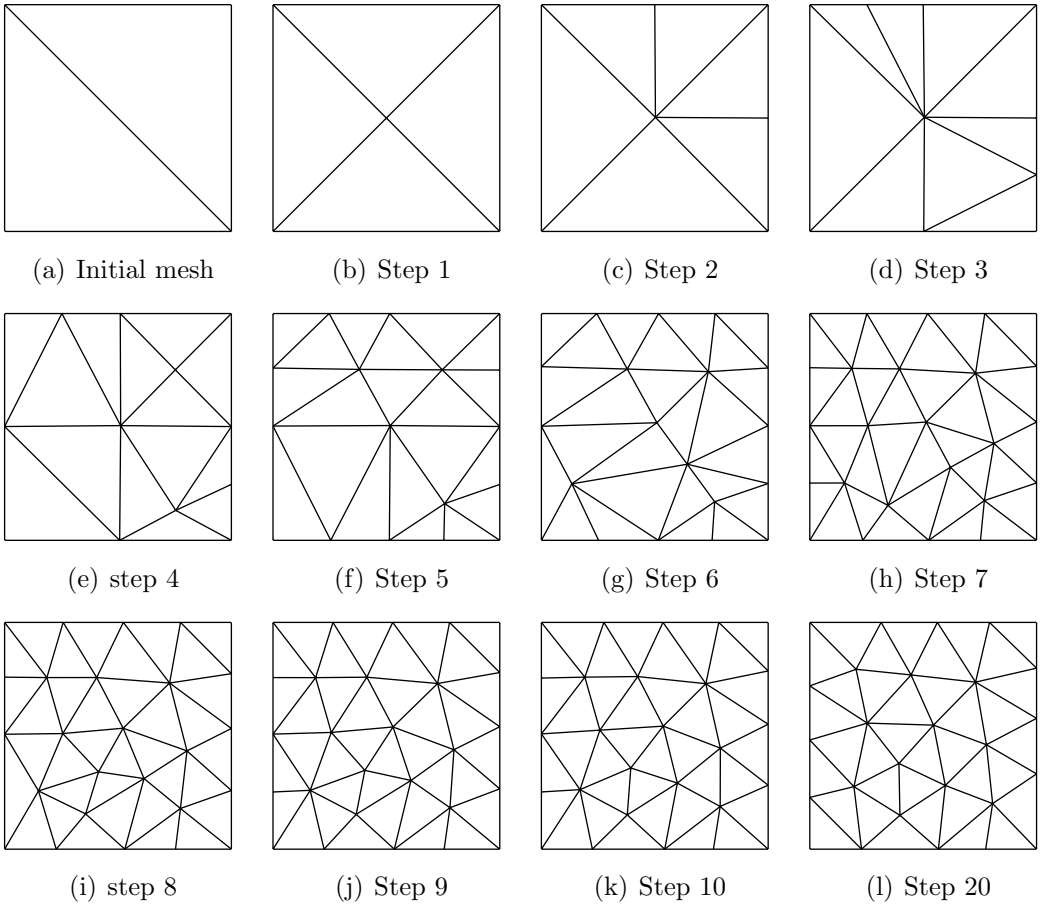
$$\text{one step of adaptation} = (\text{DE} + \text{CE}) + \text{SE} + 5 \times \text{MV} + \text{SE}$$

where DE is edge splitting, CE is edge collapsing, SE is edge swapping, and MV is vertex moving. Figure 4.6 illustrates this process using a simple example. In this example, we start the adaptation mechanism with a simple mesh with two cells. The target mesh is a uniform triangular mesh with the uniform size map of $h(x, y) = 0.25$. The figure shows the evolution of the mesh towards the target mesh.

It is important to note that the adaptation mechanisms based on spring analogy in general and the above isotropic adaptation mechanism in particular are not the most preferred mechanisms in terms of efficiency. This is due to the slow convergence of these mechanisms, which makes them undesirable for large scale simulations. However, the emphasis of this work is not on developing efficient mesh adaptation mechanisms. In this sense, the above mesh adaptation mechanism is adequate since it is easy to implement and can generate the desired mesh with minimum user input.

4.5 Application Results and Discussion

In this section, we apply the isotropic triangular mesh adaptation method, proposed in the previous sections, to the lid driven cavity flow in order to examine its performance.

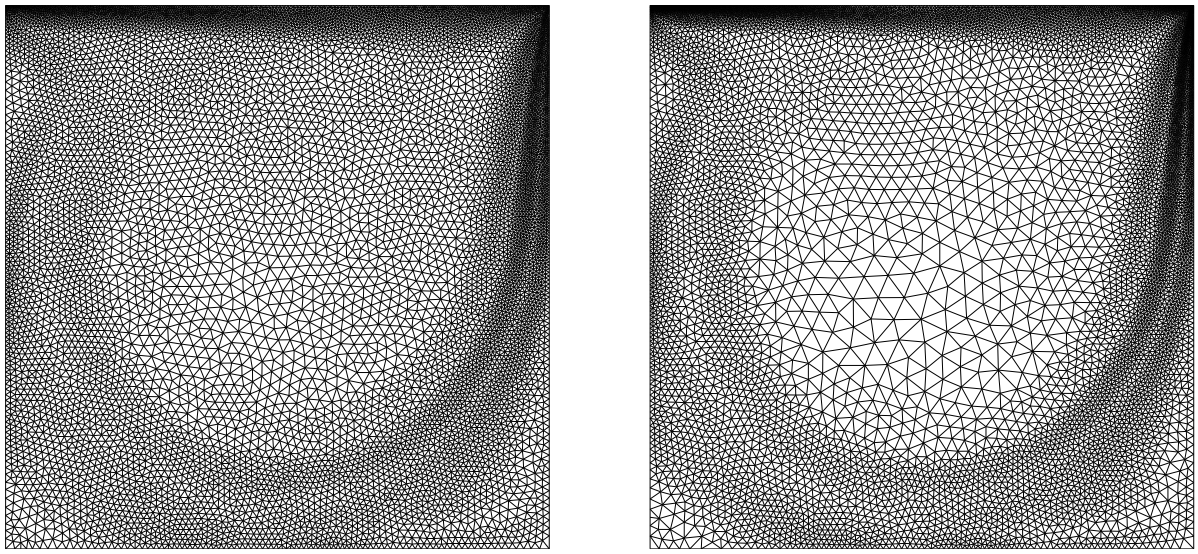


(a) Initial mesh (b) Step 1 (c) Step 2 (d) Step 3

(e) step 4 (f) Step 5 (g) Step 6 (h) Step 7

(i) step 8 (j) Step 9 (k) Step 10 (l) Step 20

Figure 4.6: Evolution of a uniform isotropic triangular mesh with the characteristic size $h = 0.25$



(a) Residual minimization (20005 cells) (b) Residual equidistribution (19926 cells)

Figure 4.7: Residual based isotropic triangular adapted meshes for the lid-driven cavity flow at Re = 1600; the left mesh is based on residual minimization and the right mesh is based on residual equidistribution

4.5.1 Lid-Driven Cavity Flow at Re = 1600

Let us apply the isotropic triangular mesh adaptation method of this chapter to the lid driven cavity flow at Re = 1600. The process of mesh adaptation is fairly straightforward. The first step is to solve the flow field on an initial mesh to obtain the solution variables and their derivatives. Here we use a uniform isotropic triangular mesh with the cell characteristic size of $0.00625d$ as the initial mesh, where d is the size of the cavity. Then we use the criteria, given in Equations (4.9) and (4.11), to obtain the target size map of the adapted mesh. We can also adjust the constants C and D in these criteria to get various meshes with different number of cells. Then we use this target size map on the initial mesh and use the adaptation mechanism to generate the target mesh.

Figure 4.7 shows the adapted meshes based on the residual minimization criterion of Equation (4.9) and residual equidistribution criterion of Equation (4.11). As expected,

Table 4.1: Nondimensional total kinetic energy, K^*, for the cavity flow at Re = 1600 on adapted triangular meshes

Criterion	Number of CVs (n)	K^*	Error
Residual Minimization Equations (4.9)	1283	0.089789	1.99%
	5146	0.090830	0.852%
	20005	0.091389	0.242%
	76781	0.091551	0.0655%
Residual Equidistribution Equations (4.11)	1422	0.089108	2.73%
	5930	0.091005	0.661%
	19926	0.091405	0.224%
	73450	0.091553	0.0633%

the general features of both meshes are the same but the residual minimization criterion results in a smoother mesh. We can easily justify this observation by examining Equations (4.9) and (4.11). In the residual minimization criterion, Equation (4.9), the size map is proportional to $1/f^{1/5}(x, y)$ while in the residual equidistribution criterion, Equation (4.11), the size map is proportional to $1/f^{1/3}(x, y)$. The power of one-fifth in the residual minimization criterion squeezes the residual spectrum throughout the domain more aggressively and leads to a more uniform size map compared to that of the residual equidistribution. Although the visual assessment of the meshes in Figure 4.7 provides some insight, we cannot rely on it for evaluating the performance of the proposed adaptation method. Therefore we need a quantitative analysis for this purpose.

To analyse the error reduction on the refined meshes we may follow the technique used in Section 2.2. In this technique, we compare the kinetic energy, K^*, error of the adapted meshes to that of a uniform mesh with the same number of control volumes. Table 4.1 shows the solution errors on the adapted meshes based on residual minimization and equidistribution criteria. The results of this table along with the solution errors on uniform Cartesian and triangular meshes are shown in Figure 4.8. As seen, error reduction in both cases is significant. A comparison between Figures 4.8, 3.6, and 2.4 shows that the

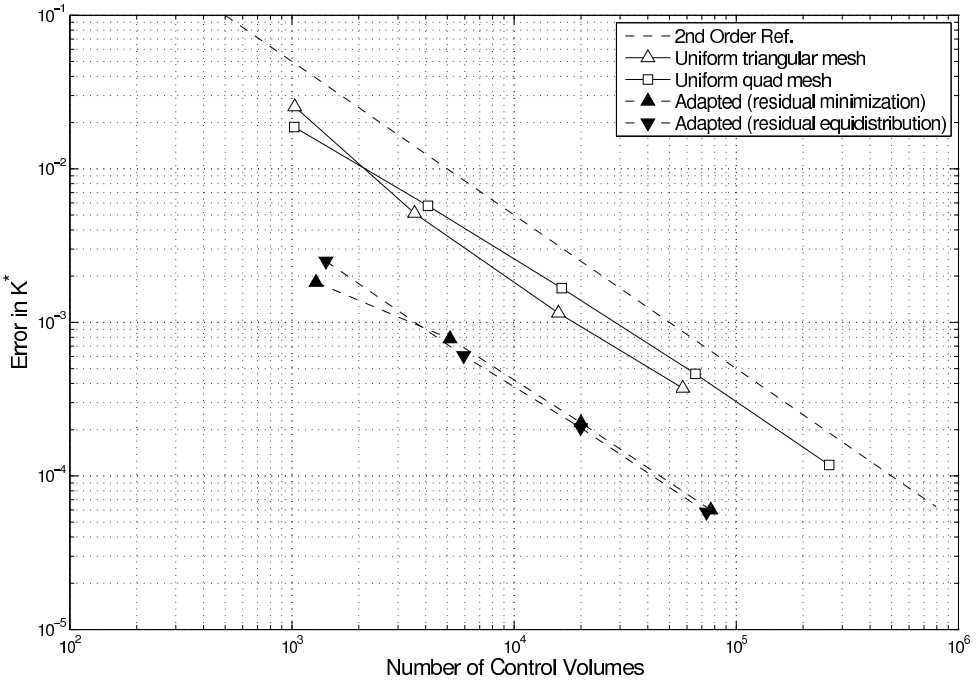


Figure 4.8: Comparison between error on uniform and adapted meshes for the lid-driven cavity flow at Re = 1600.

above residual based isotropic triangular mesh adaptation method outperforms both the traditional feature-based method and the residual-based classic h-refinement. Therefore we have been partially successful in meeting the original objectives of this work. However, the use of residual minimization criterion does not offer any advantage over the residual equidistribution criterion.

Figure 4.8 shows that the performance of the residual-equidistribution-based criterion is slightly better than that of the residual-minimization-based criterion. However, the difference is insignificant. To explain this observation we may suggest two hypotheses. One hypothesis is that the residual-equidistribution is the preferred criterion since it leads to lower solution discretization error. This hypothesis, if true, undermines the residual-minimization criterion of Equation (4.9). The alternative hypothesis is that the cavity flow at Re = 1600 is a special case that is not sensitive enough to the adaptation criterion

due to mild variations of the flow variables. If this hypothesis is true, we must see a difference between the results of these two criteria at higher Reynolds numbers where the variation of solution variables is intensified. In the following subsection, we test the second hypothesis.

4.5.2 Lid-Driven Cavity Flow at Re = 7500

In this subsection, we apply the isotropic triangular mesh adaptation method of this chapter to the lid-driven cavity flow at Re = 7500. The reason that we choose this specific Reynolds number is that the wall shear layers are stronger than those at Re = 1600. Also Re = 7500 is nearly the highest Reynolds number that creates a steady laminar flow. Although there is no clear condition for the lid-driven cavity flow to be laminar and steady, some researchers manage to obtain a steady laminar solution up to Re = 20000 [37]. However, a study by Auteri *et al.* [8] shows that the flow inside cavity undergoes a Hopf bifurcation in the range Re $\in [8017.6, 8018.8]$. Therefore we use the cavity flow at Re = 7500 as the steady laminar test case.

Tables 4.2 and 4.3 show the results on uniform Cartesian and uniform isotropic triangular meshes. As seen, the second order convergence is realized and the mesh independent value of the nondimensional kinetic energy of the flow in the cavity, K^*, is 0.095192. We use this value as a reference for evaluating the performance of the mesh adaptation method.

The adapted meshes based on the isotropic triangular method, proposed in this chapter, are shown in Figure 4.9. As seen, the general features of the adapted meshes are the same as those of adapted meshes for the flow at Re = 1600, shown in Figure 4.7. However, the effect of thinner shear layer near the driven lid and the right wall can easily be seen in Figure 4.9. To evaluate the performance of the adaptation method, let us perform an error analysis on the adapted meshes.

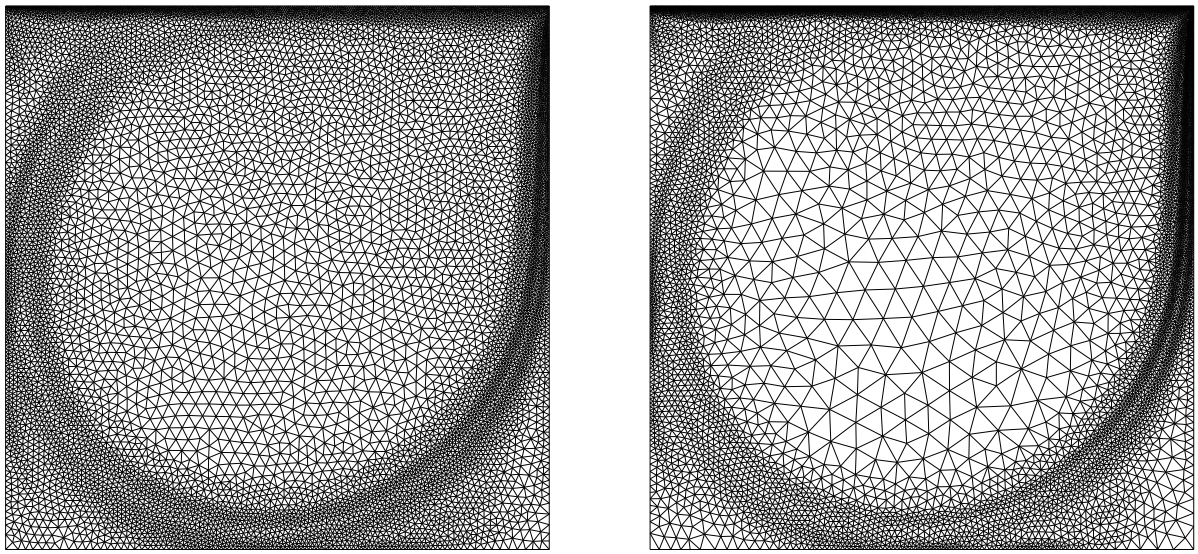
Table 4.4 shows the solution errors on the adapted meshes based on residual minimization and equidistribution criteria. The results of this table along with the solution errors on uniform Cartesian and triangular meshes are shown in Figure 4.10. As seen, the residual-minimization-based criterion reduces the error significantly better than the residual-equidistribution-based criterion. As mentioned before, this is probably due to ag-

Table 4.2: Nondimensional total kinetic energy, K^*, for the cavity flow at Re = 7500 on four different uniform Cartesian meshes; the last row shows the extrapolated value to an asymptotic fine mesh.

CVs (n)	Mesh Size (h)	K^*	Error
4096	0.015625	0.082854	13.0%
16384	0.0078125	0.092510	2.82%
65536	0.0039062	0.094290	0.947%
262144	0.0019531	0.094952	0.252%
$n \to \infty$	$h \to 0$	0.095192	

Table 4.3: Nondimensional total kinetic energy, K^*, for the cavity flow at Re = 7500 on three different uniform triangular isotropic meshes. The error is calculated based on the extrapolated value of K^* on Cartesian meshes, Table 4.2

CVs (n)	Mesh Size (h)	K^*	Error
3560	0.025470	0.079817	16.1%
15836	0.012076	0.092805	2.51%
57464	0.0063429	0.094364	0.870%



(a) Residual minimization (19686 cells)

(b) Residual equidistribution (19847 cells)

Figure 4.9: Residual based isotropic adapted meshes for the lid-driven cavity flows at $\mathrm{Re} = 1600$

Table 4.4: Nondimensional total kinetic energy, K^*, for the cavity flow at Re = 7500 on adapted triangular meshes

Criterion	Number of CVs (n)	K^*	Error
Residual Minimization Equation (4.9)	1394	0.081377	14.5%
	4941	0.093195	2.10%
	19686	0.094834	0.376%
	77720	0.095135	0.0599%
Residual Equidistribution Equation (4.11)	1473	0.081886	14.0%
	5079	0.089748	5.72%
	19847	0.093642	1.63%
	76469	0.095007	0.194%

gressive adaptation performed by the residual-equidistribution criterion that refines the mesh excessively in the regions of high residual and coarsens it in the regions of low residual. Therefore residual-minimization is the preferred criterion for the proposed isotropic mesh adaptation method.

4.5.3 Other Test Cases

In addition to the lid-driven cavity flow at Re = 1600 and 7500, we applied the isotropic mesh adaptation method of this chapter to other test cases including the laminar channel flow, laminar flat-plate boundary layer flow, and two-dimensional stagnation flow.

In the case of laminar channel flow, the adapted mesh turns out to be approximately uniform. The reason is the simple structure of the laminar channel flow. In this flow, the transverse velocity is zero and the streamwise velocity profile is parabolic. As a result, a uniform mesh is adequate for resolving the flow features and the proposed isotropic adaptation method cannot improve the simulation accuracy. Note that the above observation is only valid for laminar channel flows. In a turbulent channel flow, there is a severe velocity gradient near the wall. Therefore we expect mesh adaptation to be beneficial for a turbulent channel flow, which is beyond the scope of this work.

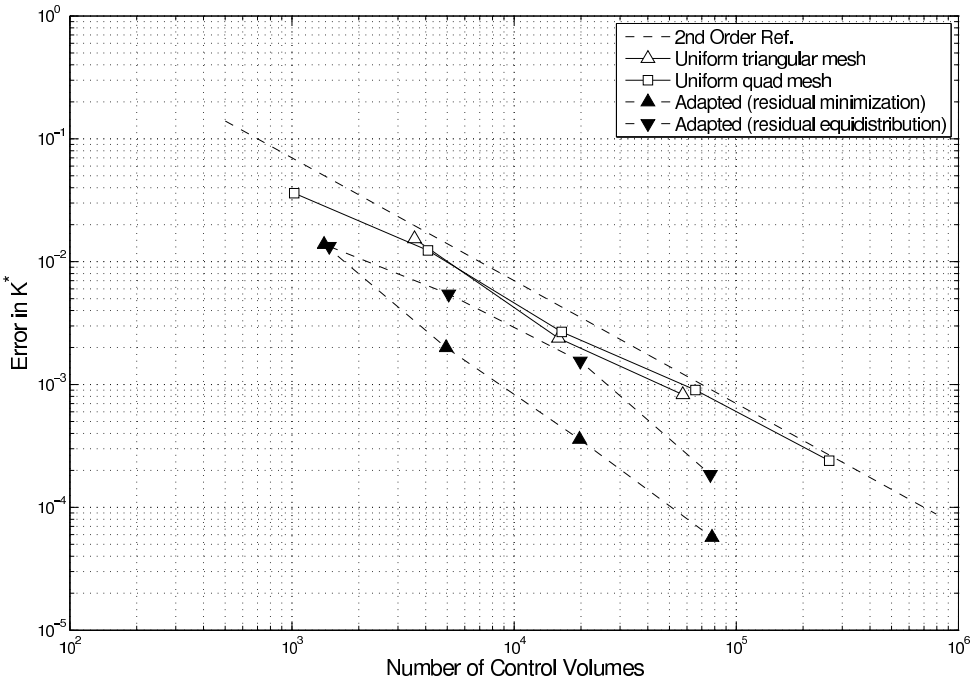


Figure 4.10: Comparison between the error on uniform and adapted meshes for the lid-driven cavity flow at Re = 7500.

In the case of laminar flat-plate boundary layer flow and two-dimensional stagnation flow, the proposed adaptation method proves to be effective. In these flows, the proposed method refines the mesh only within the boundary layer region and the simulation accuracy is significantly improved. Here we do not present the adaptation result for these flows because they exhibit some of the same features that we observed in the lid-driven cavity flow. For example, the two-dimensional stagnation flow is very similar to the flow near the top right corner of a lid-driven cavity flow. Therefore we used the lid-driven cavity flow as the main test case in this work.

4.6 Beyond the Isotropic Triangular Mesh Adaptation

The results presented in this chapter indicate that the isotropic triangular mesh adaptation method proposed in this chapter outperforms both the traditional feature-based method and classic h-refinement method for the lid-driven cavity flow. In addition, the proposed method is comparable to a simpler feature-based mesh adaptation method in terms of cost and complexity of implementation. This method also provides a reasonable adaptation criterion that does not require ad-hoc assumptions. In this sense we have already met some of the objectives set in Chapter 2. However, the beauty of residual-based mesh adaptation is the capability to go beyond simple isotropic triangular mesh adaptation.

In the next chapter, we extend the adaptation method of this chapter to more general case of anisotropic mesh adaptation. In anisotropic mesh adaptation, the concept is very similar to the isotropic method, presented in this chapter. However, we cannot characterize an isotropic mesh exclusively with a size map field. As we will show in the next chapter, for designing an anisotropic mesh at least we need to determine three scalar fields as opposed to one in the isotropic adaptation. These extra fields bring new unknowns and we have to close the mesh adaptation equations by introducing additional equations.

Chapter 5

Residual-Based Anisotropic Mesh Adaptation

In the past decade, there has been a clear trend towards research on anisotropic meshes in the CFD literature. The main advantage of anisotropic meshes is the improved capability to resolve highly directional features in the flow field without using as many cells as required in an isotropic mesh. This advantage is realized by using mesh cells that are stretched based on the local solution variations. For example, in a boundary layer where the velocity gradient in the flow transverse direction is much larger than that of the flow streamwise direction, the use of mesh cells elongated in the streamwise direction is beneficial. However, in general the solution features may not be known in advance. In these situations, we need an algorithm to detect these directional features automatically in order to adapt the mesh without user intervention.

The objective of this chapter is to propose a residual-based anisotropic mesh adaptation method, which is conceptually similar to the isotropic adaptation method of Chapter 4. In the first section, we present a brief review of the existing anisotropic mesh adaptation methods in the literature. In the second section, we highlight the potential differences between anisotropic and isotropic mesh adaptation methods. Based on these differences, we establish a conceptual framework for the anisotropic mesh adaptation method presented in this chapter. We use this conceptual framework in the third section for proposing a residual-based error indicator for quadrilateral anisotropic meshes. In

the next two sections, we present the adaptation criterion and mechanism, respectively, and propose a complete anisotropic mesh adaptation method with the capability of creating mixed triangular-quadrilateral anisotropic meshes. In the last section, we apply this method to the lid-driven cavity flow and analyse its performance.

5.1 Existing Anisotropic Adaptation Methods in the Literature

In the mesh adaptation literature, there are two general approaches towards developing anisotropic mesh adaptation methods. The first approach is based on iterative optimization methods and the second one is based on the concept of anisotropic metrics. Let us review these methods briefly.

5.1.1 Iterative Optimization-Based Mesh Adaptation

Anisotropic meshes might be the natural outcome of iterative optimization-based mesh adaptation methods. In these methods, an initial mesh is modified iteratively using an adaptation mechanism until an adaptation criterion is satisfied. In these situations, if the aspect ratio of mesh cells is not constrained by the adaptation mechanism, the final optimized mesh may turn out to be anisotropic. Examples of the above approach are the residual-based adaptation methods of Reuss and Stubley [70] and Roe and Nishikawa [75], discussed in Subsection 3.4.1. In these works, the mesh vertices are moved so that a residual-based error indicator is minimized. As a result, some of mesh cells are stretched and the mesh becomes anisotropic. Another example of this method is the work by Chong *et al.* [76], which optimizes the mesh using a genetic algorithm.

The advantage of iterative optimization-based mesh adaptation methods is their robustness in anisotropic mesh generation. In other words, these methods can handle isotropic and anisotropic meshes equally without user intervention. However, as mentioned in Subsection 3.4.1, the convergence of these methods is poor and it may take thousands of iterations to obtain the adapted mesh. Therefore these methods are not convenient for large scale CFD applications.

5.1.2 Metric-Based Anisotropic Mesh Adaptation

The second approach for developing anisotropic mesh adaptation methods in the literature is the metric-based adaptation. In this approach, the regular definition of length and angle is replaced with a generalized definition so that anisotropic meshes can be generated using the same principles used for generating isotropic meshes. For this purpose, a tensor field is required throughout the domain for evaluating lengths and angles in the mesh.

Let us demonstrate the concept of metric-based anisotropic mesh adaptation using a simple example. Figure 5.1(a) shows a perfectly uniform isotropic triangular mesh. If we apply an Affine transformation [7] to the plane of this mesh, we obtain the mesh shown in Figure 5.1(b). The Affine transformation used in this example is:

$$\mathbf{h}_b = \mathbf{T}\,\mathbf{h}_a \tag{5.1}$$

where $\mathbf{h}_a$ and $\mathbf{h}_b$ are the vectors along the mesh edges in Figures 5.1(a) and 5.1(b), respectively, and $\mathbf{T}$ is the transformation tensor:

$$\mathbf{T} = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}$$

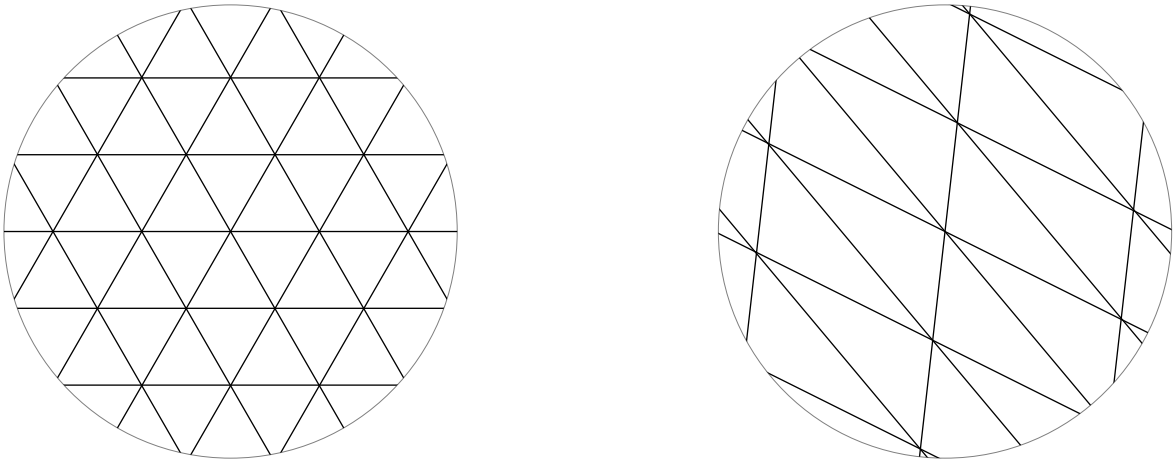
The mesh of Figures 5.1(a) satisfies the relationship:

$$\mathbf{h}_a^T \mathbf{h}_a = h^2 \tag{5.2}$$

where the superscript T is the transpose operator and h is the length of mesh edges. If the transformation tensor $\mathbf{T}$ is symmetric, substituting Equation (5.1) into (5.2) results in:

$$\mathbf{h}_b^T \underbrace{\mathbf{T}^{-2}}_{\mathbf{M}} \mathbf{h}_b = h^2 \tag{5.3}$$

where the tensor $\mathbf{M}$ is called the metric tensor. Therefore we can treat the mesh in Figure 5.1(b) like an isotropic mesh provided that we measure the length using the proper metric tensor. In the above example, the metric tensor, $\mathbf{M}$, is constant throughout the domain. However, in general $\mathbf{M}$ may vary throughout the domain and as a result we can also generate non-uniform anisotropic meshes. Therefore the problem of anisotropic



(a) Triangular uniform isotropic mesh

(b) Triangular uniform anisotropic mesh

Figure 5.1: Examples of triangular uniform isotropic and anisotropic meshes

mesh generation in a physical plane reduces to a problem of isotropic mesh generation in a metric space.

In anisotropic mesh adaptation using the metric-based method, we need to determine the metric field throughout the physical domain. The most common approach for this purpose in the CFD literature is the use of Hessian-based error indicators of the form:

$$\mathbf{M} = \mathbf{H}[\phi] = \begin{pmatrix} \frac{\partial^2 \phi}{\partial x^2} & \frac{\partial^2 \phi}{\partial x \partial y} \\ \frac{\partial^2 \phi}{\partial x \partial y} & \frac{\partial^2 \phi}{\partial y^2} \end{pmatrix} \tag{5.4}$$

where $\mathbf{H}$ is the Hessian operator, and ϕ is some solution variable. The above equation can be used along Equation (5.3) for generating the adapted mesh.

The Hessian-based anisotropic mesh adaptation method is very common in the literature [4, 6, 18, 23, 26, 45]. The reason is that this method is easy to implement. Also the mesh adaptation process based on this method is much faster than that of the iterative adaptation method. However, this method suffers from the same shortcomings as of the feature-based isotropic mesh adaptation methods in Chapter 2. The choice of the proper error indicator is not obvious and the mesh adaptation method may need some tuning to work properly.

It is also remarkable that the solution Hessian can be interpreted as the curvature of

the distribution of solution variables [34]. Therefore feature-based anisotropic mesh adaptation methods are suitable for resolving high curvature regions of the solution. Although this is a sound strategy for surface modelling and computer graphics as Tchon *et al.* [80] suggest, its benefit in CFD simulations is questionable. This argument brings us to the objective of this chapter.

The objective of this chapter is to answer the question of how to replace the common Hessian-based anisotropic error indicators in the literature with a residual-based error indicator. Since we did answer a similar question in the previous chapter for isotropic meshes, let us highlight the potential differences between anisotropic and isotropic mesh adaptation methods.

5.2 Residual-Based Anisotropic versus Isotropic Mesh Adaptation

In the previous chapter, we proposed a residual-based error indicator that led to more promising results compared to feature-based error indicators. The main assumptions that we made for developing such a residual-based error indicator were the isotropy of mesh cells and local quadratic distribution of solution variables. These assumptions led to a simplified variant of the residual estimator, expressed in Equation (4.1), of the form:

$$\varepsilon_{\text{iso}} \approx h^3 f(\text{physical solution})$$

where ε_{iso} was the isotropic error indicator and h was the mesh characteristics size. Then we manipulated the above equations to find the size map, h, so that the error indicator, ε_{iso}, becomes minimized. The basic idea of residual-based anisotropic mesh adaptation is similar to the above with two differences: the way that we represent the size map and also the mesh topology.

5.2.1 Tensor versus Scalar Size Map

In isotropic mesh adaptation method of Chapter 4, we outlined the concept of size map as a scalar field that determines the local characteristic size of the mesh cells. In an anisotropic

mesh, we cannot simply use a scalar size map since the geometric characterization of a cell not only depends on its size but also its aspect ratio and orientation. The simplest form of a size map must include all these three components and as a result an anisotropic error indicator must be of the following form:

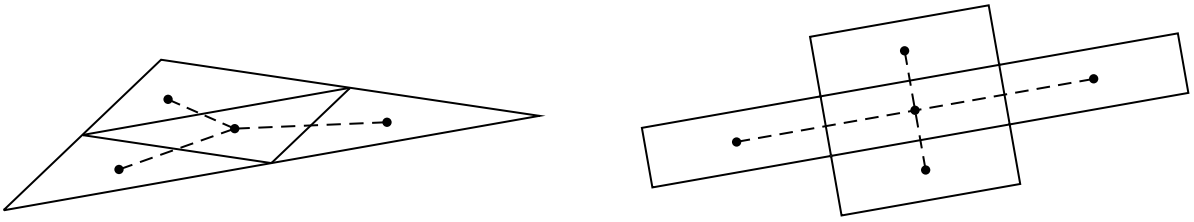
$$\varepsilon_{\text{anis}} \approx g(h, A, \theta, \text{physical solution})$$

where $\varepsilon_{\text{anis}}$ is the anisotropic error indicator, h is the cell characteristic size, A is the square root of the cell aspect ratio, and θ is the cell orientation. The goal of this chapter is to find the appropriate criteria for h, A, and θ so that the residual-based error indicator, $\varepsilon_{\text{anis}}$, becomes minimized throughout the domain. Once the fields of h, A, and θ are obtained, we can use them to construct the metric tensor $\mathbf{M}$, expressed in Equation (5.3), throughout the domain for mesh adaptation. We will derive the metric tensor based on h, A, and θ later in this chapter.

5.2.2 Quadrilateral versus Triangular Cells

In isotropic mesh adaptation method of Chapter 4, we assumed that the mesh was triangular and locally isotropic. This assumption helped us to simplify the equations for face flow errors since the line joining the nodes of two neighbouring control volumes was passing through the face integration point and was also perpendicular to face, shown in Figure 4.1. In other words, we assumed that the mesh was *orthogonal*. In contrast, it is impossible for an anisotropic triangular mesh to be orthogonal. The only way that we can generate an orthogonal anisotropic mesh is to change the mesh topology from triangular to quadrilateral.

Figure 5.2 shows simple examples of uniform anisotropic triangular and quadrilateral mesh cells. The cells in both meshes are of the same size, aspect ratio, and orientation. However, the cells in Figure 5.2(a) are triangular while the ones in Figure 5.2(b) are quadrilateral. As seen, the quadrilateral mesh is orthogonal and as a result we can simplify the flow face error estimates. Therefore in this chapter we assume that anisotropic cells are exclusively quadrilateral. Note that this is not only a matter of convenience since the anisotropic cells also reduce the residual as we prove in the following section.



(a) Uniform anisotropic triangular mesh (b) Uniform anisotropic quadrilateral mesh

Figure 5.2: Examples of uniform anisotropic triangular and quadrilateral meshes

5.3 Error Indicator for Anisotropic Quadrilateral Meshes

The procedure of deriving an error indicator for anisotropic quadrilateral meshes is similar to that of the isotropic error indicator discussed in the previous chapter. Let us explain the procedure using Figure 5.3. This figure shows a locally uniform anisotropic quadrilateral mesh. We can use the method of Appendix C to estimate the flow errors across the faces of control volume 1. However, we can simplify these estimates assuming that the distribution of solution variables is locally quadratic. Appendix H presents the details of the simplification process and also the final formulae for the flow errors across each face for the mass and momentum equations. Unfortunately, we cannot use these face flow estimates for calculating the residual of the control volume since the flow errors of opposite faces cancel out each other:

$$\Delta J_{1,2} = -\Delta J_{1,4}$$
$$\Delta J_{1,3} = -\Delta J_{1,5}$$

where $\Delta J_{1,2}$ is the flow error across the common face between the control volumes 1 and 2 in Figure 5.3 and so on for the other terms. On the other hand, in Chapter 3 we showed that the residual of a control volume is equal to the sum of flow errors across its faces. Apparently, in the case of a quadrilateral control volume this sum is zero.

$$\delta = |\Delta J_{1,2} + \Delta J_{1,3} + \Delta J_{1,4} + \Delta J_{1,5}| = 0$$

In this situation, the only way to estimate the residual is to take into account the cubic terms. In earlier chapters, we assumed that the cubic terms are locally negligible. As a result the residual estimation method of Chapter 3 fails.

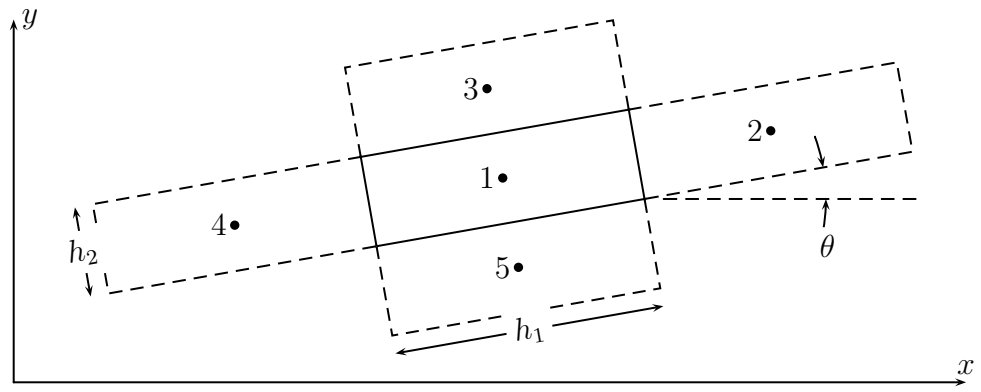


Figure 5.3: Schematic of a uniform anisotropic quadrilateral mesh with the characteristic size $h \equiv \sqrt{h_1 h_2}$, aspect ratio $A \equiv \sqrt{h_1/h_2}$, and orientation θ.

Although the above result suggests a problem in our residual estimation method, it justifies our argument for using anisotropic quadrilateral cells instead of triangular cells. Note that the residual estimation method of Chapter 3 always provides a non-zero estimate for the residual. This proves that the use of anisotropic quadrilateral cells is more beneficial in reducing the solution residual.

The above argument suggests that we cannot use the residual estimate as the error indicator. However we can use the L_2-norm of face flow errors as the error indicator.

$$\varepsilon_{\text{anis}} = \sqrt{\Delta J_{1,2}^2 + \Delta J_{1,3}^2 + \Delta J_{1,4}^2 + \Delta J_{1,5}^2} \tag{5.5}$$

This error indicator is conceptually similar to the error indicator proposed by Reuss [69] and Reuss and Stubley [70]. Appendix H presents the details of the derivation of the above error indicator for anisotropic quadrilateral meshes. The results are provided in Equations (H.9), (H.13), and (H.17) for the mass, x-momentum, and y-momentum equations, respectively. In the next section, we discuss how to use the above error indicator for specifying the anisotropic size map of the adapted mesh.

5.4 Criterion for Anisotropic Mesh Adaptation

In the previous section we proposed an anisotropic error indicator, which was based on the L_2-norm of face flow errors. The formulae for this error indicator are presented in Appendix H for the mass, x-momentum, and y-momentum equations. A simple examination of the results shows that the general form of the above error indicator is:

$$\varepsilon_{\text{anis}} = h^3 f(A, \theta, \text{physical solution}) \tag{5.6}$$

where $h \equiv \sqrt{h_1 h_2}$ is the characteristic size, $A \equiv \sqrt{h_1/h_2}$ is the square root of the aspect ratio, and θ is the orientation of mesh cells, shown in Figure 5.3. The objective is to find h, A, and θ in the above equation so that the volume-weighted error indicator becomes minimized throughout the domain. For this purpose, we need to find A and θ so that the function f is minimized. Unfortunately, the mathematical form of function f is complicated and we cannot minimize it using analytical methods. Therefore we have to find the values of A and θ numerically. Although there are many optimization techniques for finding the values of A and θ, in this work we use a simple exhaustive search. An exhaustive search is not necessarily optimal but it is easy to implement and adequate for simple applications. Once we obtain the values of of A and θ, we can substitute them in Equation (5.6) in order to find the minimum value of function f, which is only a function of physical solution.

$$\varepsilon_{\text{anis}} = h^3 f_{\min}(\text{physical solution})$$

The criterion for finding h in anisotropic mesh adaptation is the same as that of the isotropic mesh adaptation, discussed in Section 4.3. Therefore we can use Equation (4.9) for this purpose:

$$h(x, y) = \frac{C}{f_{\min}^{1/5}} \tag{5.7}$$

where C is obtained from Equation (4.10). The above equations provides a set of maps for $h(x, y)$, $A(x, y)$, and $\theta(x, y)$ throughout the domain.

The next step in the process of the criterion derivation is to introduce the proper metric tensor, as in Equation (5.3). This metric tensor will help us in the next section to develop the anisotropic adaptation mechanism. To obtain the metric tensor, $\mathbf{M}$, we can

use the following formula:

$$\mathbf{M} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} \frac{1}{h_1^2} & 0 \\ 0 & \frac{1}{h_2^2} \end{pmatrix} \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$

where h_1 and h_2 are the cell characteristic lengths and θ is its orientation, shown in Figure 5.3. Note that:

$$h_1 = hA \qquad h_2 = \frac{h}{A}$$

Therefore:

$$\mathbf{M}(x,y) = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} \frac{1}{h^2A^2} & 0 \\ 0 & \frac{A^2}{h^2} \end{pmatrix} \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \tag{5.8}$$

The above formula provides the metric field and we can use it to generate the proper anisotropic mesh. We need to use the above tensor field along with the following criterion:

$$\mathbf{h}^T\mathbf{M}\mathbf{h} = 1 \tag{5.9}$$

where $\mathbf{h}$ is a vector along mesh edges. Note that we need to perform the above calculations for the mass, x-momentum, and y-momentum equations separately. Therefore we obtain three different tensor fields, namely $\mathbf{M}_{\text{mass}}$, $\mathbf{M}_{x-\text{mom}}$, and $\mathbf{M}_{y-\text{mom}}$. In Chapter 4 where the size maps, obtained from the mass and momentum equations, were scalar fields, we picked the smallest value as the target characteristic size of the adapted mesh. Unfortunately, this simple method is not applicable to the metric fields based on the mass and momentum equations since we cannot simply compare their magnitudes. Therefore we have to use a more general method for combining the three metric fields $\mathbf{M}_{\text{mass}}$, $\mathbf{M}_{x-\text{mom}}$, and $\mathbf{M}_{y-\text{mom}}$.

Let us discuss the geometric interpretation of Equation (5.9) in order to find a method for combining the metric fields. Equation (5.9), is a criterion for the length of mesh edges. Assume that one end-point of a mesh edge is at the origin of the coordinate system. Equation (5.9) implies that the locus of the other end-point of the edge is an ellipse. The size of the minor and major radii of this ellipse depends on the eigenvalues of the metric tensor $\mathbf{M}$. The orientation of the ellipse depends on the orientation of eigenvectors of the metric tensor $\mathbf{M}$. In the case that we have more than one metric tensor, the size and

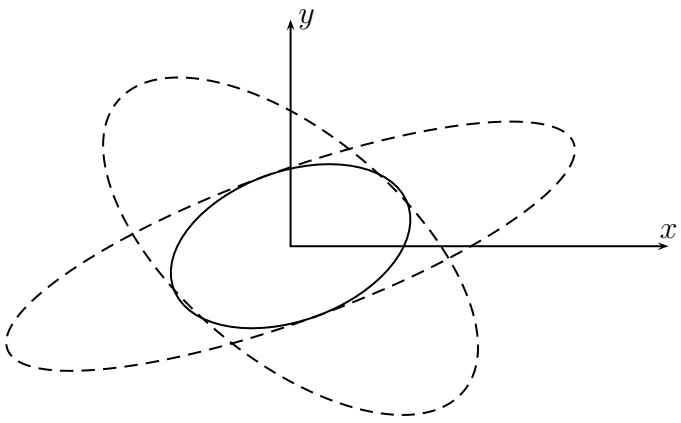


Figure 5.4: The ellipses associated with two metric tensors (dashes ellipses) and their intersection based on the method of Appendix I (solid ellipse).

orientation of these ellipses may be different. Figure 5.4 illustrates this situation for the case of two different metric tensors that are shown by two dashed ellipses.

The general method for combining more than one metric tensor is to find the intersection of their associated ellipses as Borouchaki *et al.* [19] and Castro-Diaz *et al.* [26] point out. The intersection of two ellipses is the largest ellipse that is inscribed to both those ellipses. We can also generalize this concept to the intersection of three and more ellipses. Unfortunately, the calculation of the exact intersection ellipse is difficult. However in this work, we use a modified variation of the approximate method proposed by Borouchaki *et al.* [19] and Castro-Diaz *et al.* [26] to find the intersection ellipse. Figure 5.4 shows an example of the result of this method. The details of the procedure for finding the intersection ellipse is presented in Appendix I. Once we calculate the intersection ellipse, we can calculate a new metric field based on it, which would be the intersection of the three metric fields $\mathbf{M}_{\text{mass}}$, $\mathbf{M}_{x-\text{mom}}$, and $\mathbf{M}_{y-\text{mom}}$. We use this metric field in Equation (5.9) for anisotropic mesh adaptation.

$$\mathbf{M} = \text{intersection}(\mathbf{M}_{\text{mass}}, \mathbf{M}_{x-\text{mom}}, \mathbf{M}_{y-\text{mom}}) \tag{5.10}$$

In the following section, we propose an anisotropic mesh adaptation mechanism that

satisfies the criterion, expressed in Equation (5.9).

5.5 Mechanism for Anisotropic Mesh Adaptation

The anisotropic adaptation mechanism in this work is similar to that of the isotropic adaptation, discussed in Subsection 4.4.2. However, there are two additional components in the anisotropic mesh adaptation mechanism that do not exist in the isotropic mechanism. The first component, mentioned earlier in this chapter, is the metric tensor for evaluating lengths and angles and the second one is a mechanism for switching anisotropic triangular cells to quadrilateral cells.

The first additional component of an anisotropic adaptation mechanism compared to the isotropic mechanism of Subsection 4.4.2 is the metric-based evaluation of lengths and angles. In this technique, the length of mesh edges is calculated using the following formula:

$$|\mathbf{h}|_{\mathbf{M}} = \sqrt{\mathbf{h}^T \mathbf{M} \mathbf{h}} = 1 \tag{5.11}$$

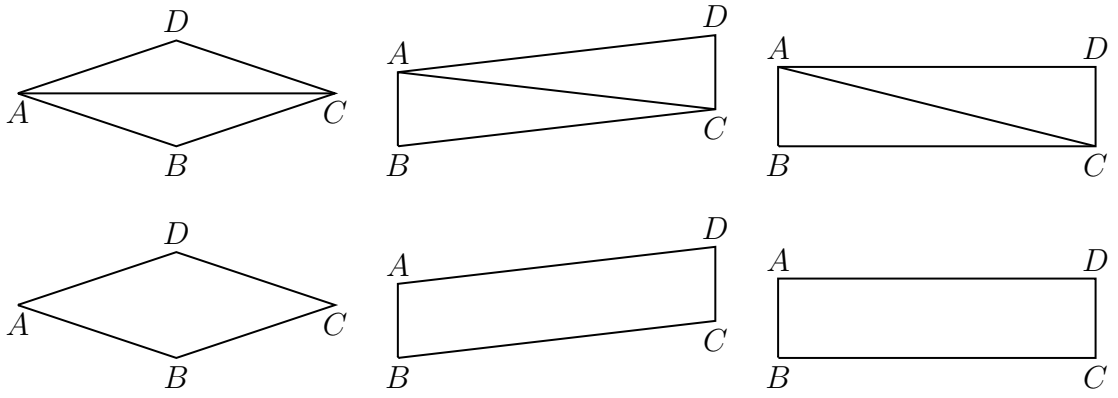
We also need to measure angles in order to satisfy the Delaunay criterion, discussed in the previous chapter.

$$\angle(\mathbf{h}_1, \mathbf{h}_2)_{\mathbf{M}} = \cos^{-1}\left(\frac{\mathbf{h}_1^T \mathbf{M} \mathbf{h}_2}{|\mathbf{h}_1|_{\mathbf{M}} |\mathbf{h}_2|_{\mathbf{M}}}\right) \tag{5.12}$$

Using the above formulae, we can easily extend the isotropic mesh adaptation mechanism, presented in Subsection 4.4.2, to anisotropic cases. The result would be an anisotropic triangular mesh in which $|\mathbf{h}|_{\mathbf{M}} = 1$.

The second additional component of an anisotropic adaptation mechanism compared to the isotropic mechanism of Subsection 4.4.2 is an algorithm for switching triangular cells to quadrilateral cells since we originally developed the anisotropic error indicator for quadrilateral cells. The key element in switching triangles to quads is to merge two triangular cells by removing their common face. Although this concept appears to be simple, it is difficult to implement. To understand the reason let us examine various situations that we may encounter in merging anisotropic triangular cells.

Figure 5.5 shows three possible situations that we encounter in the process of merging anisotropic triangular cells into anisotropic quadrilateral cells. In each case, if we remove



(a) Two obtuse triangles (b) Two acute triangles (c) Two right-angled triangles

Figure 5.5: Examples of merging two anisotropic triangular cells into one quadrilateral cell

the face AC, we obtain the quad $ABCD$. Note that we are only interested in quadrilateral cells that provide an orthogonal mesh. As seen, the quads of Figures 5.5(a) and Figures 5.5(b) cannot be used in an orthogonal anisotropic quadrilateral mesh and our only option is the quad of Figure 5.5(c). However, there is no way to produce the right-angled triangles of Figure 5.5(c) in the first place using the above anisotropic adaptation mechanism. The reason is that the triangles ABC and ACD in Figure 5.5(c) violate the adaptation criterion of Equation (5.11). To understand the reason, assume that the faces AB and BC satisfy the criterion of Equation (5.11). Therefore:

$$(\overrightarrow{AB})^T\mathbf{M}(\overrightarrow{AB}) = 1$$
$$(\overrightarrow{BC})^T\mathbf{M}(\overrightarrow{BC}) = 1$$

Using the Pythagorean theorem we can show that:

$$(\overrightarrow{AC})^T\mathbf{M}(\overrightarrow{AC}) = 2$$

Therefore we have no chance to obtain any anisotropic right-angled triangle based on criterion of Equation (5.11). To address this issue we have to modify this criterion.

To generate anisotropic right-angled triangular cells, we have to modify the criterion

of Equation (5.11) using the following formula:

$$|\mathbf{h}|_{\mathbf{M}} = \frac{\sqrt{\mathbf{h}^T\mathbf{M}\mathbf{h}}}{|\cos\psi| + |\sin\psi|} = 1 \tag{5.13}$$

where ψ is the generalized angle between the mesh edge vector, $\mathbf{h}$, and an eigenvector of the metric tensor, $\mathbf{M}$.

$$\psi = \angle(\mathbf{h}, \mathbf{v}_{\mathbf{M}}) = \cos^{-1}\left(\frac{\mathbf{h}^T\mathbf{M}\mathbf{v}_{\mathbf{M}}}{|\mathbf{h}||\mathbf{v}_{\mathbf{M}}|}\right) \tag{5.14}$$

where $\mathbf{v}_{\mathbf{M}}$ is an eigenvector of the metric tensor, $\mathbf{M}$. Let us illustrate the concept of Equation (5.13). In Figure 5.6, the dashed ellipse represents a typical metric tensor. However, we are interested in a rectangular cell with the same aspect ratio. The dotted rectangle in the figure has the same aspect ratio as that of the dashed ellipse. The rose-shaped solid curve in Figure 5.6 is the plot of Equation (5.13). This rose-shaped curve gives the ideal size of a mesh edge as a function of its orientation angle. If an edge is parallel to the eigenvectors of the metric tensor, $\mathbf{v}_1$ and $\mathbf{v}_2$, its ideal size would be the same as that given by the dashed ellipse. Otherwise, the ideal length of an edge would be longer than the length given by the ellipse. Note that in the case that the generalized angle between an edge and an eigenvector is $\pi/4$, the ideal length would be equal to half of the diagonal of the dotted rectangle. As a result, the spring analogy, presented in the previous chapter, forces mesh vertices to move towards the corners of dotted rectangle or the mid-point of its edges.

Unfortunately, the numerical experiments show that we cannot use criterion (5.13) throughout the entire domain. The reason is that the mesh becomes over constrained and the adaptation mechanism does not converge. To address this problem, we only use the criterion of Equation (5.13) in the regions where the local mesh aspect ratio is larger than a certain threshold. In this work, the threshold is three.

$$\begin{aligned} A(x,y) < 3.0 \quad &\Rightarrow \quad \sqrt{\mathbf{h}^T\mathbf{M}\mathbf{h}} = 1 \\ A(x,y) \geq 3.0 \quad &\Rightarrow \quad \frac{\sqrt{\mathbf{h}^T\mathbf{M}\mathbf{h}}}{|\cos\psi| + |\sin\psi|} = 1 \end{aligned} \tag{5.15}$$

The use of the above criterion results in an anisotropic triangular mesh. However, the high aspect ratio triangles in the resulting mesh are nearly right-angled like Figure 5.5(c).

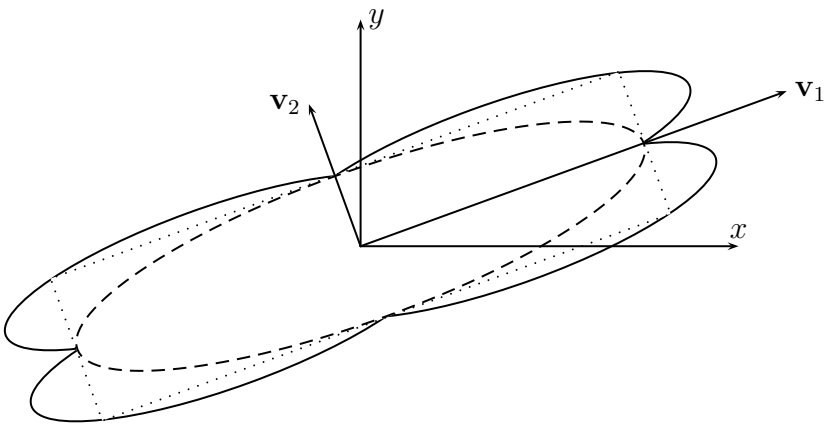


Figure 5.6: Original metric ellipse (dashed) and its modified form (solid) for anisotropic mesh adaptation

Then we look at the mesh edges and remove the ones that make a $\pi/4$ angle with the eigenvectors of the local metric tensor. The resulting mesh would be an anisotropic mixed triangular-quadrilateral mesh.

Note that the anisotropic error indicator, proposed in the previous sections, was designed exclusively for anisotropic quadrilateral control volumes. However, in the above adaptation mechanism we compromised this condition in order to obtain a well-behaved adaptation mechanism. In other words, we used anisotropic mesh cells only in the regions where mesh aspect ratio is high. This compromise might look detrimental to the validity of the adaptation method. However, a simple examination of Figures 4.8 and Figures 4.10 shows that the error on an isotropic triangular mesh is not very different from the error on a quadrilateral mesh with the same characteristic size. Therefore we use this assumption as an approximation. In the next section we analyse the performance of the anisotropic mesh adaptation method proposed in this chapter.

5.6 Application Results and Discussion

Let us apply the anisotropic mixed triangular-quadrilateral mesh adaptation method of this chapter to the lid driven cavity flow at Re = 1600 and Re = 7500. The process of

Table 5.1: Nondimensional total kinetic energy, K^*, for the cavity flow at Re = 1600 on adapted triangular-quadrilateral meshes

Number of CVs (n)	K^*	Error
2890	0.090079	1.67%
18686	0.091351	0.284%
68742	0.091542	0.0753%

mesh adaptation is the same as that of the isotropic adaptation in the previous chapter. The first step is to solve the flow field on an initial mesh to obtain the solution variables and their derivatives. Here we use a uniform isotropic triangular mesh with the cell characteristic size of $0.00625d$ as the initial mesh, where d is the size of the cavity. Then we use the criteria, given in Equations (5.15), to obtain the target size of edges in the adapted mesh. The next step is to interpolate this target size on the initial mesh and use the adaptation mechanism to generate the target mesh. At the end, we remove some mesh edges in order obtain a mixed triangular-quadrilateral mesh.

Figure 5.7 shows the adapted mesh for the cavity flow at Re = 1600 based on the minimization of the error indicator. As seen, the mesh is highly anisotropic near the driven lid and the right wall. But it is mostly isotropic in the rest of the domain. Although visual assessment provides some useful information, we cannot rely on it for evaluating the performance of the proposed adaptation method. Let us analyse the performance of the adapted mesh quantitatively.

To analyse the error reduction on the refined meshes we may follow the technique used in Section 2.2. In this technique, we compare the kinetic energy, K^*, error on the adapted meshes to that of a uniform mesh with the same number of control volumes. Table 5.1 shows the solution errors on the anisotropic adapted meshes. The results of this table along with the solution errors on uniform Cartesian and triangular meshes and also the isotropic adapted meshes of the previous chapter are shown in Figure 5.8. As seen, error reduction is significant but not as good as that of the isotropic method of the previous chapter. However, the performance of the finest anisotropic adapted mesh is almost as

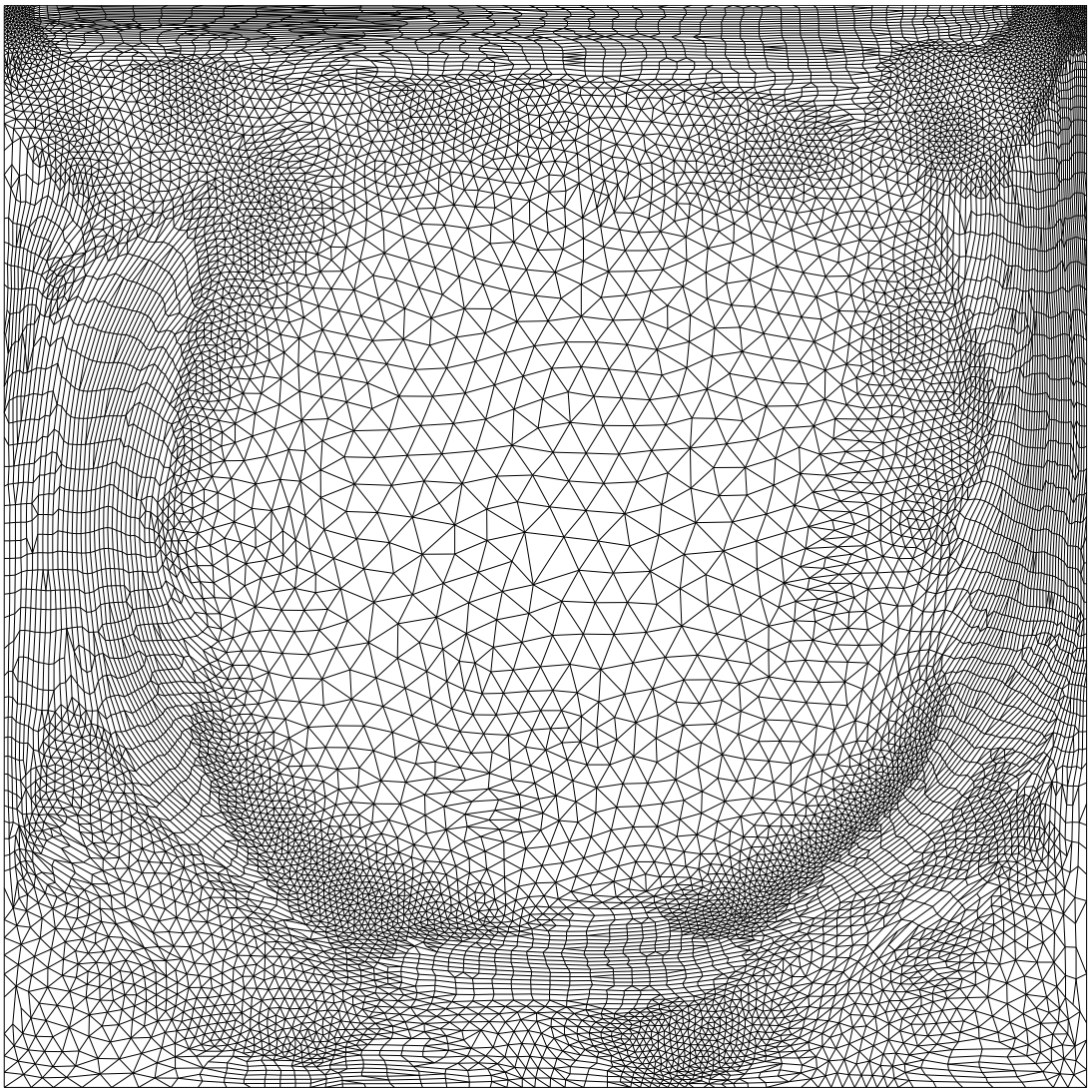


Figure 5.7: Anisotropic mesh adaptation for the lid-driven cavity flow at $\mathrm{Re} = 1600$.

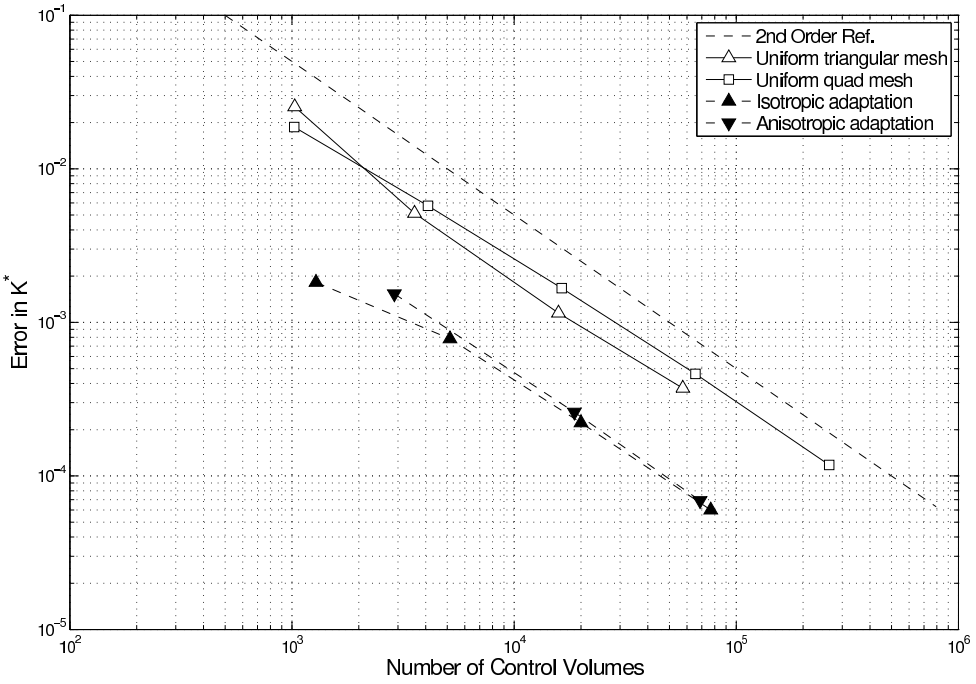


Figure 5.8: Comparison between the error on uniform and adapted meshes for the lid-driven cavity flow at Re = 1600.

good as the performance of an isotropic mesh with the same number of control volumes. Let us apply the proposed adaptation method to the cavity flow at Re = 7500 as well to investigate any potential difference in the performance.

The anisotropic adapted mesh for the cavity flow at Re = 7500 is shown in Figure 5.9. As seen, the general features of the adapted mesh are the same as those of adapted mesh for the flow at Re = 1600, shown in Figure 5.7. However, the effect of the thinner shear layer near the driven lid and the right wall can easily be seen in Figure 5.9. To evaluate the performance of the adaptation method, let us perform an error analysis on the adapted meshes.

Table 5.2 shows the solution errors on the adapted mesh of Figure 5.9. The results of this table along with the solution errors on uniform Cartesian and triangular meshes and also isotropic adapted meshes of the previous chapter are shown in Figure 5.10. As seen,

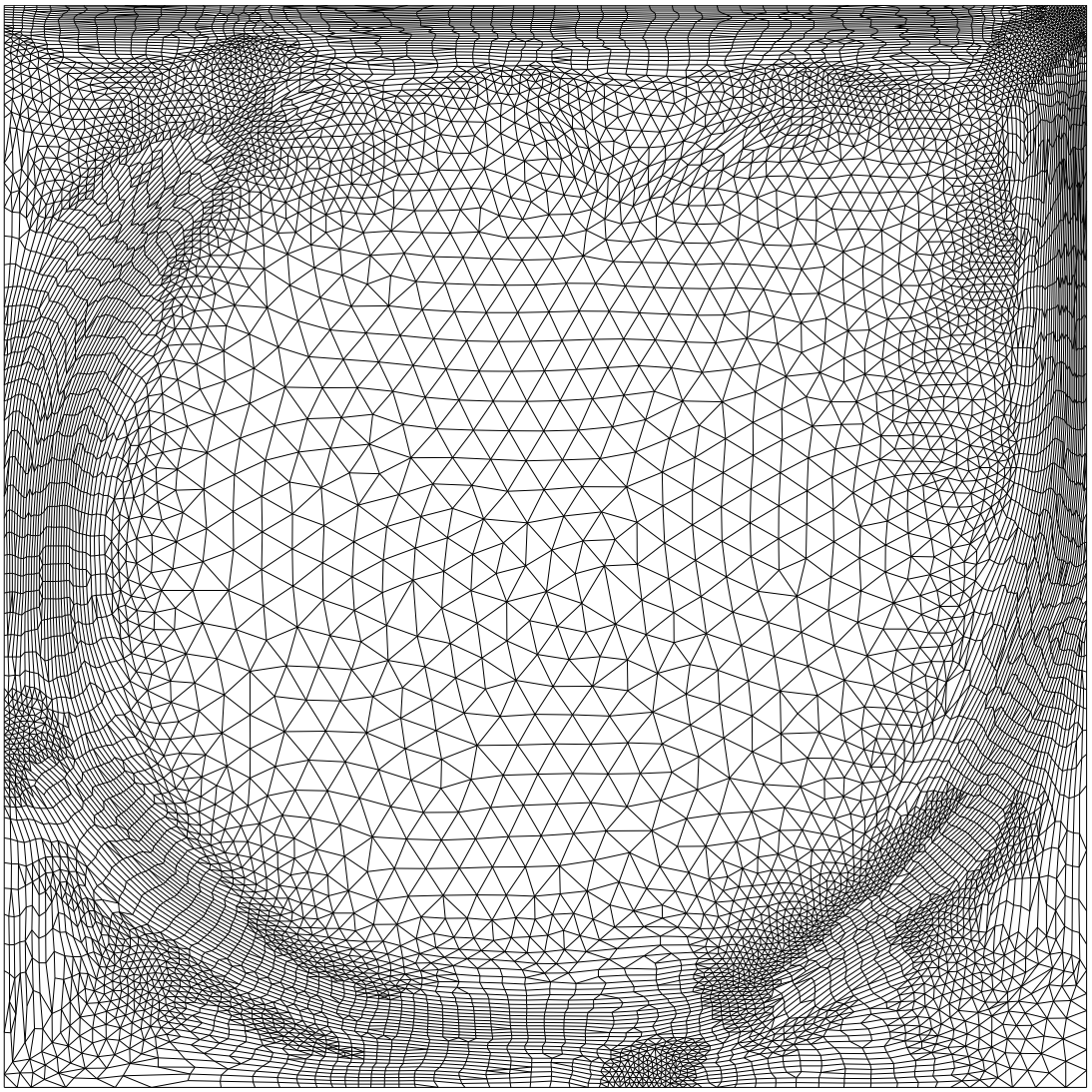


Figure 5.9: Anisotropic mesh adaptation for the lid-driven cavity flow at $\mathrm{Re} = 7500$.

Table 5.2: Nondimensional total kinetic energy, K^*, for the cavity flow at Re = 7500 on adapted triangular-quadrilateral meshes

Number of CVs (n)	K^*	Error
3854	0.088993	6.51%
14860	0.094517	0.709%
51073	0.095087	0.110%

the results are consistent with the results of the flow at Re = 1600. The performance of the anisotropic adapted mesh is close to that of the isotropic adapted mesh but not as good.

We also applied the above anisotropic mesh adaptation method to a few other test cases including the laminar channel flow, two-dimensional stagnation flow, and backward facing step flow [16]. In the case of the laminar channel flow, the above method results in a uniform anisotropic mesh in the streamwise direction. The reason is that in a laminar channel flow, the streamwise gradients are zero while the transverse gradients are not zero. Therefore it is preferred to have a mesh with a large characteristic size in the streamwise direction and a small characteristic size in the transverse direction. In the case of the backward facing step flow, the results of the above anisotropic adaptation method is very similar to that of a channel flow except within the separation bubble behind the backward facing step. In this bubble, the flow is viscous dominated and the mesh turns out to be isotropic. This case is very similar to the separation bubbles near the bottom corners of the lid-driven cavity flow. In the case of the two-dimensional stagnation flow, the proposed anisotropic mesh adaptation method results in anisotropic cells in the boundary layer region and isotropic cells outside the boundary layer. This case is very similar to the flow in the vicinity of the top right corner of the lid-driven cavity flow.

The results presented in this chapter indicate that the anisotropic triangular-quadrilateral mesh adaptation method proposed in this chapter is effective. The method can calculate the suitable aspect ratio of mesh cells and change their topology from triangular to quadrilateral if necessary. However, the performance of this method is not as good as

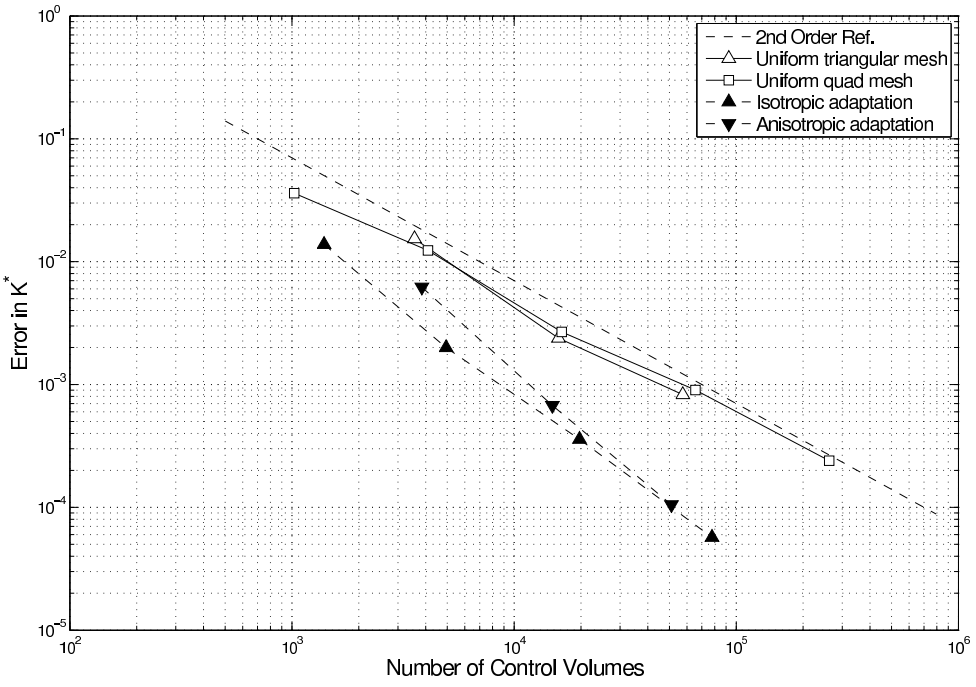


Figure 5.10: Comparison between the error on uniform and adapted meshes for the lid-driven cavity flow at Re = 7500.

the simpler isotropic mesh adaptation method of Chapter 4. This observation may have a few reasons. The most important reason is that we developed the error indicator and adaptation criterion for perfectly anisotropic rectangular cells. However, in practice we used a mixed triangular-quadrilateral mesh. In other words, we applied the error indicator in a situation that it was not designed for. In addition, it is hard to generate a purely quadrilateral mesh with orthogonal angles since there is little control over cell angles. As a result, a few orphan triangular cells are formed in the quadrilateral regions of Figures 5.7 and 5.9, which may adversely affect the solution accuracy. The last noticeable reason for the suboptimal performance of the proposed anisotropic adaptation method is the error generated at the interface of triangular and quadrilateral cells. It is very hard to control the quality of mesh cells near this interface and as a result it can be a major source of error.

Although the proposed anisotropic mesh adaptation method does not outperform the isotropic method of the previous chapter, it still has a remarkable advantage over anisotropic triangular mesh adaptation methods in the literature, which is the stability of the numerical solution. One problem with high aspect ratio triangular meshes is the instability of the numerical solution due to the stiffness of the discretized equations. Using anisotropic quadrilateral cells partially addressees this issue. Another advantage of the proposed anisotropic method is its potential capability to determine the thickness of wall inflation for mesh generators. We will discuss this point in the next chapter.

Chapter 6

Closure

Modern CFD applications involve complicated physical phenomena in complex geometries. In these applications, it is beneficial to optimize the mesh using automatic methods in order to take full advantage of computer resources. This thesis has been an attempt towards developing a fully automatic residual-based two-dimensional unstructured mesh adaptation method for steady state laminar flows. In this chapter, we discuss the contributions of this work to the mesh adaptation literature and also recommendations for future work.

6.1 Contributions of the Thesis

6.1.1 Contribution to the Methodology of Mesh Adaptation

The most important contribution of this work is the introduction of a methodology for developing a residual-based error indicator that is as robust as other residual-based error indicators in the literature and also as easy to use as feature-based error indicators. The key idea to the proposed error indicator is that it not only takes into account the effect of physical solution, but also the effect of mesh geometry. Therefore it provides an integrated framework for determining the mesh geometric properties as a function of physical solution without using heuristic criteria.

An error indicator is a measure that quantifies the contribution of each control volume

to the overall solution error. The most common error indicators in the mesh adaptation literature are based on solution features. These indicators are usually based on values or derivatives of physical variables. Although these error indicators are easy to use, they are not very robust and usually require user adjustment. To address this problem researchers have proposed residual-based error indicators.

The basic approach for developing a residual-based error indicator is to evaluate the error of mass and momentum flows across the faces of each control volume by estimating the neglected terms in the Taylor series expansion associated with the discretized equations of face flows. Then the error indicator is defined based on the estimated face flow errors. This approach has been used in numerous works in the mesh adaptation literature including the works by Reuss and Stubley [70], Hay and Visonneau [49], Roe and Nishikawa [75]. However the error indicators in these works do not account for the geometry of the mesh. In other words, the error indicator is only dependent on the physical solution on a given mesh.

$$\varepsilon = f(\text{physical solution})$$

Apparently, such an error indicator is not suitable for mesh adaptation since the geometry of the mesh varies significantly in the course of adaptation. To address this issue the mesh adaptation process has to be heavily under-relaxed, which leads to a slow and inefficient adaptation method.

In this work, we demonstrated a methodology for developing an error indicator that offers the advantages of both feature-based and residual-based error indicators without having their disadvantages. In this methodology, a higher order discretization scheme is used for estimating face flow errors. Then these face flow errors are simplified by assuming that the mesh is locally uniform and the solution distribution is locally quadratic. The result of this procedure is a simplified formula for face flow errors, which can be used for establishing an error indicator of the following form.

$$\varepsilon = g(\text{mesh geometry}, \text{physical solution})$$

An error indicator of the above form is valuable since the mesh geometry can be devised so that the error indicator is minimized. In this sense, the adaptation criterion is an integral part of the error evaluation process, as opposed to the traditional mesh adaptation methods in which the adaptation criterion is based on heuristic formulae.

6.1.2 Contributions to the Implementation of Mesh Adaptation

In the previous section, we discussed the major contribution of this work to the methodology of residual-based mesh adaptation. To prove that the proposed methodology works in practice we applied it to mesh adaptation for steady state incompressible laminar flows of constant property fluids. This process led to a few contributions in the implementation of an adaptation method, which are:

- A feature-based-like error indicator for isotropic mesh adaptation;
- A technique for minimizing the error indicator; and
- A mechanism for generating mixed triangular-quadrilateral meshes.

Let us explain each item in more detail.

In this work, we proposed a residual-based error indicator for isotropic mesh adaptation in Equations (4.3) through (4.5), which is as easy to use as a feature-based error indicator. This indicator involves only the derivatives of the physical solution. Therefore in certain circumstances, the proposed methodology of the previous section may lead to very simple error indicators. This simplicity is very beneficial since the error indicator in an existing mesh adaptation code can be easily replaced by the proposed error indicator.

The second implementation contribution of this thesis is a method for minimizing a certain norm of the error indicator throughout the domain using the calculus of variations. For example, in both isotropic and anisotropic cases, the error indicators reduced to the following form:

$$\varepsilon = h^3 g(\text{physical solution})$$

We can use the above formula to determine the mesh size, h, so that a certain norm of the error indicator becomes minimized. For example, for minimizing the volume weighted average of the error indicator, ε, we obtained:

$$h \propto \frac{1}{g^{1/5}}$$

Therefore there is no need to use an equidistribution principle or any other heuristic criterion for minimizing the error indicator.

The third implementation contribution of this thesis is a method for generating mixed triangular-quadrilateral meshes in anisotropic mesh adaptation mechanism. The basic idea of anisotropic adaptation mechanism in this work is the same as other metric-based anisotropic mechanisms in the literature. However, the existing methods in the literature cannot properly generate mixed triangular-quadrilateral meshes. In this work, we proposed a simple modification of the metric-based length evaluation in Equation (5.13), which can be used to generate quadrilateral cells. Unfortunately, the proposed method does not generate very high aspect ratio elements. As a rule of thumb, the quality of quadrilateral elements starts degrading for aspect ratios larger than ten.

This thesis also contains some minor contributions to the calculation of second order derivatives, Hessian, and the procedure for averaging the solution error and residual.

6.2 Suggestions for Future Work

In this work, we presented an automatic residual-based mesh adaptation method for steady-state laminar incompressible flows. The adaptation method presented in this thesis has certain limitations both in terms of applicability and efficiency. The performance of the adaptation method can be improved by using more efficient algorithms. Its applicability can also be extended to more general situations by using other mathematical models. In this section, we discuss a few suggestions for future work.

6.2.1 Short-term Suggestions

The short term suggestions are the ones that can be implemented in the framework of this thesis without a major rework. These suggestions include:

- Accounting for non-conservative terms;
- Minimizing the anisotropic error indicator with respect to aspect ratio and orientation using a more efficient algorithm;
- Using other adaptation criteria;
- Using more efficient adaptation mechanisms;

- Using the anisotropic indicator for wall inflation; and
- Investigating the impact of the Hessian reconstruction method.

In the following paragraph, we explain each item in more detail.

An important extension to this work is to consider the effect of non-conservative terms in the mathematical model. In this work, we used the system of steady state incompressible mass and momentum equations as the mathematical model. In this system, all terms are conservative. In other words, with the Gauss divergence theorem we can transform all volume integrals to surface integrals. However, this is not generally possible for source terms in the momentum equation. The only way that we can write a source term in the form of a surface integral is when the source term is curl free. Unfortunately, this is not always the case. This problem becomes crucial when the fluid density or viscosity vary throughout the domain. Therefore one important extension to this work is to include the non-conservative source terms.

The second suggestion is to use a more efficient algorithm for minimizing the anisotropic error indicator with respect to aspect ratio and orientation angle in Chapter 5. In Equation (5.6), we showed that the anisotropic error indicator was of the following form:

$$\varepsilon = h^3 g(A, \theta, \text{physical solution})$$

where A and θ where the local cell aspect ratio and orientation, respectively. To minimize the function g with respect to A and θ, we used an exhaustive search. However, we could use more accurate and efficient algorithms for this purpose. For example, we could use a steepest descent or a conjugate gradient algorithm.

The third suggestion is to use other adaptation criteria. In this work, the minimization of the volume weighted error indicator was the adaptation criterion. Although this option proved to be effective, there is no guarantee that it is the best option. For example, we could use the residual second-norm minimization as the adaptation criterion.

The fourth suggestion is to use more efficient adaptation mechanisms, especially for mixed triangular-quadrilateral meshes. In this thesis, we used an iterative adaptation mechanism based on vertex repositioning, face splitting, face removing, and face swapping. The results proved that the implemented mechanism was effective and other researchers

have also used this mechanism successfully. However, generating the proper mesh using this method involves a lot of iterations, especially for anisotropic mixed triangular-quadrilateral meshes.

The fifth short term suggestion is to use the proposed anisotropic error indicator for determining the thickness of wall inflation in CFD applications. In wall inflation, a few layers of quadrilateral/prismatic cells are constructed over the surface of solid walls as if the walls were inflated. The main advantage of wall inflation is the improved resolution for boundary layers, especially in turbulent flows. However, the thickness of the inflated region is usually unknown. Therefore it may be beneficial to use the anisotropic error indicator, presented in Chapter 5, to determine the thickness of the inflated region.

The last short term suggestion is to investigate the impact of the Hessian reconstruction scheme on the overall performance of the adaptation method. The Hessian reconstruction scheme adopted in this work is presented in Appendix F. In this scheme, the Hessian tensor is estimated based on a given second order solution at the control volume nodes. In a second order solution, the values of a physical variable at the node of a control volume is equal to the average value of that variable throughout the control volume. However, in a third order accurate solution in which the Hessian tensor is non-zero, the effect of the Hessian tensor on the average values of physical variables should be taken into account, as suggested by Ollivier-Gooch and Van Altena [64]. In this work, we used the Hessian tensor to estimate the residual of a second order solution. Therefore the effect of the Hessian reconstruction scheme on the residual estimation should be investigated. However the residual estimates, shown in Figures 3.6 through 3.11 , do not suggest any significant problem with the current Hessian reconstruction scheme.

6.2.2 Long-Term Suggestions

The long term suggestions are the ones that need more in depth analysis of the basic concepts and also major reworks. These suggestions are:

- Mixed order of accuracy methods;
- Three-dimensional flows; and
- Turbulent flows.

In the following paragraph, we explain each item in more detail.

One important strategy in commercial CFD codes is to blend different order of accuracy schemes. The main rationale for this strategy is to eliminate numerical oscillations in order to obtain a stable code. For example, one may discretize the pressure and viscous terms in the momentum equation using second order central differencing scheme and the momentum flow term using a first order upwind scheme. In this work we assumed that the discretization was clean and second order. This assumption puts limitations on the applicability of the proposed method and it is necessary to investigate the possibility of developing a similar adaptation method for blended schemes.

The second important extension of this work is the extension to three-dimensional problems. Although in this work we focused on two dimensional problems, there is hardly any two dimensional problem of significant engineering importance in real world applications. The extension of the proposed method of this work to three dimensional problems is straightforward but tedious. In three dimensions, all the face flow errors must be evaluated across triangular or quadrilateral faces. Therefore the evaluation of face integrals become more complicated. However, the basic concept does not change and we still have to assume that the mesh is locally uniform and the solution variation is locally quadratic.

The last and supposedly the most important extension of this work is the extension to turbulent flows. This is especially important due to the fact that the most practical applications involve turbulent flows. In this case, the basic idea could be to take into account the turbulence model equations. In this approach, the coefficient of viscosity, μ, is replaced by the eddy viscosity, which accounts for the enhanced mixing in the flow due to turbulence velocity fluctuations. Therefore the overall picture of the mesh adaptation method for turbulent flows is the same. However, the details would probably be more challenging than those we presented in this thesis.

Bibliography

[1] H. Abdi. Coefficients of correlation, alienation and determination. In N. J. Salkind, editor, *Encyclopedia of Measurement and Statistics*. Sage Publications, Inc, 2007.

[2] M. J. Aftosmis and M. J. Berger. Multilevel error estimation and adaptive h-refinement for Cartesian meshes with embedded boundaries. In *AIAA Paper 2002-0863*. 40th AIAA Aerospace Sciences Meeting and Exhibit, 2002.

[3] F. Alauzet, P. J. Frey, P. L. George, and B. Mohammadi. 3D transient fixed point mesh adaptation for time-dependent problems: Application to CFD simulations. *Journal of Computational Physics*, 222(2):592–623, 2007.

[4] F. Alauzet, P. L. George, B. Mohammadi, P. Frey, and H. Borouchaki. Transient fixed point-based unstructured mesh adaptation. *International Journal for Numerical Methods in Fluids*, 43:729–745, 2003.

[5] F. Alauzet, X. R. Li, E. S. Seol, and M. S. Shephard. Parallel anisotropic 3D mesh adaptation by mesh modification. *Engineering with Computers*, 21(3):247–258, 2006.

[6] R. C. Almeida, R. A. Feijoo, A. C. Galeao, C. Padra, and R. S. Silva. Adaptive finite element computational fluid dynamics using an anisotropic error estimator. *Computer Methods in Applied Mechanics and Engineering*, 182:379–400, 2000.

[7] G. B. Arfken. *Mathematical Methods for Physicists*. Elsevier Publishing Company, international student ed. edition, 2005.

[8] F. Auteri, N. Parolini, and L. Quartapelle. Numerical investigation on the stability of singular driven cavity flow. *Journal of Computational Physics*, 183:1–25, 2002.

[9] T. J. Baker. Mesh adaptation strategies for problems in fluid dynamics. *Finite Elements in Analysis and Design*, 25:243–273, 1997.

[10] T. J. Baker. Delaunay Vononoi mehtods. In J. F. Thompson, B. K. Soni, and N. P. Weatherill, editors, *Handbook of Grid Generation*. CRC Press, 1999.

[11] T. J. Baker. Mesh generation: Art or science? *Progress in Aerospace Sciences*, 41(1):29–63, 2005.

[12] B. R. Baliga and S. V. Patankar. A control volume finite-element method for two-dimentional fluid flow and heat transfer. *Numerical Heat Transfer*, 6:245–262, 1983.

[13] E. Barragy and G. F. Carey. Stream function-vorticity driven cavity solution using p finite element. *Computers and Fluids*, 26:453–468, 1997.

[14] T. J. Barth. Recent developments in high order k-exact reconstruction on unstructured meshes. In *AIAA Paper 1993-0668*. AIAA 31th Aerospace Sciences Meeting, 1993.

[15] T. J. Barth and D. C. Jespersen. The design and application of upwind schemes on unstructured meshes. In *AIAA Paper 1989-0366*. AIAA 27th Aerospace Sciences Meeting, 1989.

[16] I. E. Barton. A numerical study of flow over a confined backward facing step. *International Journal for Numerical Methods in Fluids*, 21:653–665, 1995.

[17] A. R. Baserinia and G. D. Stubley. Improved Hessian reconstruction scheme for cell-centred finite volume method. In *14th Annual Conference of CFD Society of Canada*, St. John's, Canada, July 2005.

[18] L. Borges, R. Feijoo, and N. Zouain. A directional error estimator for adaptive limit analysis. *Mechanics Research Communications*, 26(5):555–563, 1999. 13.

[19] H. Borouchaki and P. L. George. Aspects of 2-D delaunay mesh generation. *International Journal for Numerical Methods in Engineering*, 40(11):1957–1975, 1997. 24.

[20] H. Borouchaki, P. L. George, F. Hecht, P. Laug, and E. Saltel. Delaunay mesh generation governed by metric specifications .1. algorithms. *Finite Elements in Analysis and Design*, 25(1-2):61–83, 1997. 20.

[21] F. J. Bossen and P. S. Heckbert. A pliant method for anisotropic mesh generation. In *5th International Meshing Roundtable*, pages 63–74, October 1996.

[22] O. Botella and R. Peyret. Benchmark specteral results on the lid-driven cavity flow. *Computers and Fluids*, 27:421–433, 1998.

[23] G. C. Buscaglia and E. A. Dari. Anisotropic mesh optimization and its application in adaptivity. *International Journal for Numerical Methods in Engineering*, 40:4119–4136, 1997.

[24] Z. Cai. On the finite volume element method. *numerische mathematik*, 58:713–735, 1991.

[25] D. Caraeni, M. Caraeni, and L. Fuchs. A parallel multidimensional upwind algorithm for LES. In *15th AIAA Computational Fluid Dynamics Conference*, Anaheim, CA, June 2001.

[26] M. J. Castro-Diaz, F. Hecht, B. Mohammadi, and O. Pironneau. Anisotropic unstructured mesh adaption for flow simulations. *International Journal for Numerical Methods in Fluids*, 25:475–491, 1997.

[27] F. Courty, D. Leservoisier, P. L. George, and A. Dervieux. Continuous metrics and mesh adaptation. *Applied Numerical Mathematics*, 56:117–145, 2006.

[28] T. A. Davis. Algorithm 832: UMFPACK, an unsymmetric-pattern multifrontal method. *ACM Transactions on Mathematical Software*, 30(2):196–199, 2004.

[29] T. A. Davis. A column pre-ordering strategy for the unsymmetric-pattern multifrontal method. *ACM Transactions on Mathematical Software*, 30(2):165–195, 2004.

[30] T. A. Davis and I. S. Duff. An unsymmetric-pattern multifrontal method for sparse LU factorization. *SIAM Journal on Matrix Analysis and Applications*, 18(1):140–158, 1997.

[31] T. A. Davis and I. S. Duff. A combined unifrontal/multifrontal method for unsymmetric sparse matrices. *ACM Transactions on Mathematical Software*, 25(1):1–19, 1999.

[32] G. de With and A. E. Holdo. The use of solution adaptive grid for modeling small scale turbulent structures. *Journal of Fluids Engineering*, 127:936–944, 2005.

[33] I. Dermirdžić, Ž. Lilek, and M. Perić. A collocated finite volume method for predicting flows at all speeds. *International Journal for Numerical Methods in Fluids*, 16:1029–1050, 1993.

[34] M. P. Do Carmo. *Differential Geometry of Curves and Surfaces*. Prentice Hall, 1976.

[35] V. Dolejší and J. Felcman. Anisotropic mesh adaptation for numerical solution of boundary value problems. *Numerical Methods for Partial Differential Equations*, 20:576–608, 2004.

[36] E. Erturk, T. C. Corke, and C. Gökçöl. Numerical solutions of 2-D steady incompressible driven cavity flow at high Reynolds numbers. *International Journal for Numerical Methods in Fluids*, 48:747–774, 2005.

[37] E. Erturk and C. Gökçöl. Fourth-order compact formulation of NavierStokes equations and driven cavity flow at high Reynolds numbers. *International Journal for Numerical Methods in Fluids*, 50:421–436, 2006.

[38] V. Ervin and W. Layton. An analysis of a defect-correction method for e model convection-diffusion equation. *SIAM Journal of Numerical Analysis*, 26:169–179, 1989.

[39] J. H. Ferziger and M. Perić. *Computational Methods for Fluid Dynamics*. Springer, 3 edition, 2002.

[40] D. A. Field. Laplacian smoothing and Delaunay triangulations. *Communications in Applied Numerical Methods*, 4:709–712, 1988.

[41] P. J. Frey and F. Alauzet. Anisotropic mesh adaptation for CFD computations. *Computer Methods in Applied Mechanics and Engineering*, 194(48-49):5068–5082, 2005.

[42] P. J. Frey and P. L. George. *Mesh Generation*. HERMES Science Publishing, 2000.

[43] P. L. George, H. Borouchaki, and P. Laug. An efficient algorithm for 3D adaptive meshing. *Advances in Engineering Software*, 33:377–387, 2002.

[44] U. Ghia, K. N. Ghia, and C. T. Shin. High-Re solutions for incompressible-flow using the Navier Stokes equations and a multigrid method. *Journal of Computational Physics*, 48(3):387–411, 1982. 23.

[45] W. G. Habashi, J. Dompierre, Y. Bourgaulta, D. Ait-Ali-Yahiaa, M. Fortinb, and M. G. Vallet. Anisotropic mesh adaptation: towards user-independent, mesh-independent and solver-independent CFD - part I: general principles. *International Journal for Numerical Methods in Fluids*, 32:725–744, 2000.

[46] A. Hay. *Étude des Stratégies dEstimation dErreur Numérique et dAdaptation Locale de Maillages Non-Structurés Pour les Équations de Navier-Stokes en Moyenne de Reynolds*. PhD thesis, Ecole Centrale de Nantes, Nantes, France, 2004.

[47] A. Hay, A. Leroyer, and M. Visonneau. H-adaptive Navier-Stokes simulations of free-surface flows around moving bodies. *Journal of Marine Sciences and Technology*, 11:1–18, 2006.

[48] A. Hay and M. Visonneau. Adaptive mesh strategy applied to turbulent flows. *Comptes Rendus Mecanique*, 333:103–110, 2005.

[49] A. Hay and M. Visonneau. Error estimation using the error transport equation for finite-volume methods and arbitrary meshes. *International Journal of Computational Fluid Dynamics*, 20:463–479, 2006.

[50] C. Ilinca, X. D. Zhang, J.-Y. Trépanier, and R. Camarero. A comparison of three error estimation techniques for finite-volume solutions of compressible flows. *International Journal for Numerical Methods in Engineering*, 189:1277–1294, 2000.

[51] A. Jameson. Acceleration of transonic potential flow calculations on arbitrary meshes by the multiple grid method. In *AIAA Paper 79-1458*. AIAA 4th Computational Fluid Dynamics Conference, 1979.

[52] A. Jameson. CFD for aerodynamic design and optimization: Its evolution over three decades. In *AIAA Paper 2003-3438*. AIAA 16th Computational Fluid Dynamics Conference, 2003.

[53] H. Jasak and A. D. Gosman. Element residual error estimate for the finite volume method. *Computers and Fluids*, 32:223–248, 2003.

[54] P. Jawahar and H. Kamath. A high-resolution procedure for Euler and Navier-Stokes computations on unstructured grids. *Journal of Computational Physics*, 164(1):165–203, 2000.

[55] J. E. Lennard-Jones. Cohesion. *Proceedings of the Physical Society*, 43(5):461–482, 1931.

[56] P. Lin, L. Martinelli, and T. J. Baker. Two-dimensional implicit time dependent calculations for incompressible flows on adaptive unstructured meshes. In *AIAA Paper 2001-2655*. AIAA 15th Computational Fluid Dynamics Conference, 2001.

[57] R. Lohner. Mesh adaptation in fluid mechanics. *Engineering Fracture Mechanics*, 50:819–847, 1995.

[58] D. J. Mavriplis. Adaptive meshing techniques for viscous flow calculations on mixed element unstructured meshes. *International Journal for Numerical Methods in Fluids*, 34:93–111, 2000.

[59] K. W. Morton, M. Stynes, and E. Süli. Analysis of a cell-vertex finite volume method for convection-diffusion problems. *Mathematics of Computation*, 66:1389–1406, 1997.

[60] K. W. Morton and E. Süli. Finite volume methods and their analysis. *IMA Journal of Numerical Analysis*, 11:241–260, 1991.

[61] S. Nicaise. A posteriori error estimations of some cell centered finite volume methods for diffusion-convection-reaction problems. *SIAM Journal on Numerical Analysis*, 44:949–978, 2006.

[62] P. Nithiarasu. An adaptive remeshing technique for laminar natural convection problems. *Heat and Mass Transfer*, 38:243–250, 2002.

[63] P. Nithiarasu and O. C. Zienkiewicz. Adaptive mesh generation for fluid mechanics problems. *International Journal for Numerical Methods in Engineering*, 47(1–3):629–662, 2000. 55.

[64] C. F. Ollivier-Gooch and M. Van Altena. A high-order-accurate unstructured mesh finite-volume scheme for the advection-diffusion equation. *Journal of Computational Physics*, 181(2):729–752, 2002.

[65] B. Palmerio. A two-dimentional FEM adaptive moving node mehtod for steady Euler flow simulations. *Computer Methods in Applied Mechanics and Engineering*, 71:315–340, 1988.

[66] Y. Peng, Y. Shiau, and R. R. Hwang. Numerical solutions of 2-D steady incompressible driven cavity flow at high Reynolds numbers. *Computers and Fluids*, 32:337–352, 2003.

[67] M. Perić. *A Finite Volume Method for the Prediction of Three-Dimensional Flow in Complex Ducts*. PhD thesis, University of London, 1985.

[68] N. Pierce and M. Giles. Adjoint and defect error bounding and correction for functional estimates. *Journal of Computational Physics*, 200:769–794, 2004.

[69] S. Reuss. *A Generally Applicable Mesh Adaptation Criterion*. PhD thesis, University of Waterloo, Waterloo, Canada, 2002.

[70] S. Reuss and G. D. Stubley. An improved error indicator for mesh adaptation. *Numerical Heat Transfer, Part B: Fundamentals*, 43:1–18, 2003.

[71] C. M. Rhie and W. L. Chow. Numerical study of the turbulent-flow past an airfoil with trailing edge separation. *AIAA Journal*, 21(11):1525–1532, 1983.

[72] L. F. Richardson. *Weather Prediction by Numerical Process*. Cambridge University Press, 1922.

[73] P. J. Roache. Perspective - a method for uniform reporting of grid refinement studies. *Journal of Fluids Engineering-Transactions of the ASME*, 116(3):405–413, 1994. 34.

[74] P. J. Roache. Quantification of uncertainty in computational fluid dynamics. *Annual Review of Fluid Mechanics*, 29:123–160, 1997. 101.

[75] P. Roe and H. Nishikawa. Adaptive grid generation by minimizing residuals. *International Journal for Numerical Methods in Fluids*, 40(1-2):121–136, 2002. 11.

[76] Chong C. S., Lee H. P., and Kumar A. S. Genetic algorithms in mesh optimization for visualization and finite element models. *Neural Computing and Applications*, 15:366–372, 2006.

[77] G. E. Schneider and M. J. Raw. Control volume finite-element method for heat-transfer and fluid-flow using colocated variables .1. computational-procedure. *Numerical Heat Transfer*, 11(4):363–390, 1987. 18.

[78] P. N. Shankar and M. D. Deshpande. Fluid mechanics in the driven cavity. *Annual Review of Fluid Mechanics*, 32:93–136, 2000.

[79] K. Shimada. *Physically-Based Mesh Generation: Automated Triangulation of Surfaces and Volumes via Bubble Packing*. PhD thesis, Massachusetts Institute of Technology, 1993.

[80] K. F. Tchon, M. Khachan, F. Guibault, and R. Camarero. Three-dimensional anisotropic geometric metrics based on local domain curvature and thickness. *Computer-Aided Design*, 37:173–187, 2005.

[81] D. A. Venditti and D. L. Darmofal. Grid adaptation for functional outputs: Application to two-dimensional inviscid flows. *Journal of Computational Physics*, 176:40–69, 2002.

[82] D. A. Venditti and D. L. Darmofal. Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows. *Journal of Computational Physics*, 187:22–46, 2003.

[83] S. Yang. Adaptive strategy of the supersonic turbulent flow over a backward-facing step. *International Journal for Numerical Methods in Fluids*, 44:1163–1184, 2004.

[84] Y. Zang and R. L. Street. A composite multigrid method for calculating unsteady incompressible flows in geometrically complex domains. *International Journal for Numerical Methods in Fluids*, 20:341–361, 1995.

[85] X. D. Zhang, J.-Y. Trépanier, and Camarero. R. A posteriori error estimation for finite-volume solutions of hyperbolic conservation laws. *Computer Methods in Applied Mechanics and Engineering*, 185:1–19, 2000.

Appendix A

Discretization of the Governing Equations

The governing equations for fluid flows are the conservation of mass and momentum equations. In the case of a steady-state incompressible isothermal flow of a Newtonian fluid in the absence of gravity, the mass and momentum equations are:

$$\nabla \cdot \mathbf{v} = 0$$
$$\nabla \cdot (\rho \, \mathbf{v} \otimes \mathbf{v}) = -\nabla p + \nabla \cdot \left[\mu \left(\nabla \mathbf{v} + \nabla \mathbf{v}^T \right) \right]$$

In most practical cases there is no analytical solution to the above equations. Therefore we have to discretize and solve them numerically. In this appendix, we present the details of the discretization process of the mass and momentum equations using the second-order accurate cell-centred finite volume method on an unstructured mesh.

A.1 Discretization of the Mass Equation

The differential form of the mass equation, presented in Equation (3.4), reads:

$$\nabla \cdot \mathbf{v} = 0 \tag{A.1}$$

Integrating the latter over some domain Ω using Gauss-Ostrogradsky Divergence Theorem [7] gives rise to the integral form of the mass equation.

$$\int_{\partial\Omega} \mathbf{v} \cdot \hat{\mathbf{n}}\, dA = 0 \tag{A.2}$$

where $\partial\Omega$ is the boundary of Ω and $\hat{\mathbf{n}}$ is the unit normal vector to $\partial\Omega$. Note that the choice of the domain Ω is entirely arbitrary. Assuming that Ω is a polygonal control volume, Equation (A.2) reduces to:

$$\sum_{\text{faces}} J_{\text{vol}} = \sum_{\text{faces}} \tilde{V}_n A_f = 0 \tag{A.3}$$

where the summation is over the control volume faces. In Equation (A.3), A_f is the face area and $\tilde{V}_n$ is the face-normal velocity that carries mass, called the *advecting velocity*. In this equation, $\tilde{V}_n$ is the only parameter that needs to be discretized. In this work, we use Rhie and Chow velocity pressure interpolation method [71] for discretizing $\tilde{V}_n$. Note that the discretization process for the boundary faces is slightly different from that of the interior faces. In the following subsections, we present the discretization details for each case, separately.

A.1.1 Volumetric Flow Rate across an Interior Face

An interior face is a face that has two neighbouring control volumes. The purpose of discretization is to establish a relationship between the flow across the face and the solution at the nodes of the neighbouring control volumes. Rhie and Chow velocity pressure interpolation scheme provides such a relationship [71].

$$\tilde{V}_n = \bar{V}_n - d_f \left(\left.\frac{\partial p}{\partial n}\right|_{\text{face}}^{\text{active}} - \left.\frac{\partial p}{\partial n}\right|_{\text{face}}^{\text{lagged}} \right) \tag{A.4}$$

where $\bar{V}_n$ is the velocity that carries momentum, called the *advected velocity*. The discretization of $\bar{V}_n$ is based on the central differencing scheme, presented in Appendix D.

$$\bar{V}_n = \frac{1}{2}\left(V_{n_1} + V_{n_2}\right)^{k+1} + \frac{1}{4}\left(\nabla V_{n_1} + \nabla V_{n_2}\right)^{k} \cdot (\mathbf{r}_1 + \mathbf{r}_2) \tag{A.5}$$

The subscripts 1 and 2 represent the indexes of the neighbouring control volumes and the superscripts k and $k+1$ represent the previous and current iterations, respectively.

The second term on the right hand side of Equation (A.4) prevents the so called *checkerboard problem* from happening. The coefficient d_f is the pressure dissipation coefficient at the face, defined by:

$$d_f = \frac{1}{2}\left(\frac{V_1}{a_1} + \frac{V_2}{a_2}\right)$$

where V_1 and V_2 are the volumes and a_1 and a_2 are the average coefficients of the momentum equation associated with the control volumes 1 and 2, respectively. In Equation (A.4), the terms within the parentheses are the active and lagged pressure gradients normal to the face, evaluated at the current and previous iterations, respectively.

$$\left.\frac{\partial p}{\partial n}\right|_{\text{face}}^{\text{active}} = \alpha\left(\frac{p_2 - p_1}{|\mathbf{s}|}\right)^{k+1} + \frac{1}{2}(\nabla p_1 + \nabla p_2)^k \cdot (\hat{\mathbf{n}} - \alpha\hat{\mathbf{s}}) \tag{A.6}$$

$$\left.\frac{\partial p}{\partial n}\right|_{\text{face}}^{\text{lagged}} = \frac{1}{2}(\nabla p_1 + \nabla p_2)^k \cdot \hat{\mathbf{n}} \tag{A.7}$$

where $\alpha = \hat{\mathbf{n}} \cdot \hat{\mathbf{s}}$ is the nonorthogonality factor, explained in Appendix E. Substituting Equations (A.5) through (A.7) back into Equation (A.4) yields:

$$\begin{aligned}\tilde{V}_n = \frac{1}{2}\left(V_{n_1} + V_{n_2}\right)^{k+1} + \frac{1}{4}\left(\nabla V_{n_1} + \nabla V_{n_2}\right)^k \cdot (\mathbf{r}_1 + \mathbf{r}_2) - \\ \alpha\, d_f \left[\left(\frac{p_2 - p_1}{|\mathbf{s}|}\right)^{k+1} - \frac{1}{2}\left(\nabla p_1 + \nabla p_2\right)^k \cdot \hat{\mathbf{s}}\right]\end{aligned} \tag{A.8}$$

Note that the volumetric flow rate, J_{vol}, in Equation (A.3), is equal to $\tilde{V}_n A_f$. Therefore, we may manipulate Equation (A.8) and write it in the following form.

$$\begin{aligned}J_{\text{vol}}^{\text{int}} = \frac{A_f}{2}(\mathbf{v}_1 + \mathbf{v}_2)^{k+1} \cdot \hat{\mathbf{n}} + \frac{\alpha d_f A_f}{|\mathbf{s}|}(p_1 - p_2)^{k+1} + \\ \frac{A_f}{4}(\nabla \mathbf{v}_1 + \nabla \mathbf{v}_2)^k : [\hat{\mathbf{n}} \otimes (\mathbf{r}_1 + \mathbf{r}_2)] + \frac{\alpha d_f A_f}{2}(\nabla p_1 + \nabla p_2)^k \cdot \hat{\mathbf{s}}\end{aligned} \tag{A.9}$$

where the operator $(\otimes)$ represents the *tensor product* of two vectors

$$\mathbf{a} \otimes \mathbf{b} = \begin{pmatrix} a_x \\ a_y \end{pmatrix} \begin{pmatrix} b_x & b_y \end{pmatrix} = \begin{pmatrix} a_x b_x & a_x b_y \\ a_y b_x & a_y b_y \end{pmatrix}$$

and the operator (:) represents the *Frobenius inner product* of two tensors.

$$\mathbf{A} : \mathbf{B} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} : \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix} = A_{11}B_{11} + A_{12}B_{12} + A_{21}B_{21} + A_{22}B_{22}$$

Equation (A.9) is the discretized form of the volumetric flow rate across an interior face. In the following subsection, we present the discretized form of the volumetric flow rate across a boundary face

A.1.2 Volumetric Flow Rate across a Boundary Face

The discretization of the volumetric flow rate across a boundary face is different from that of an interior face. This difference is due to the prescription of the boundary conditions. In the following paragraphs, we derive the discrete form of the volumetric flow rate across a boundary face with various boundary conditions.

Inflow and No-Slip Wall Boundary Conditions

For the inflow and no-slip wall boundary conditions, the flow velocity at the boundary face, $\mathbf{v}_b$, is prescribed. Therefore the volumetric flow rate reads the following formula:

$$J_{\text{vol}}^{\text{in}} = J_{\text{vol}}^{\text{wall}} = V_n A_f = (\mathbf{v}_b \cdot \hat{\mathbf{n}}) A_f \tag{A.10}$$

Note that the flow velocity at a no-slip wall is not necessarily zero since the wall might be moving. However, the the flow velocity at the wall must be equal to the velocity of the wall.

Outflow Boundary Condition

In the case of outflow boundary condition, we assume that the pressure at the outflow boundary is prescribed. Therefore the formula of the advecting velocity, Equation (A.8), changes to the following equation:

$$\tilde{V}_n = V_{n_1}^{k+1} + \nabla V_{n_1}^{k} \cdot \mathbf{r}_1 - \alpha\, d_f \left[\left(\frac{p_b - p_1}{|\mathbf{r}_1|} \right)^{k+1} - (\nabla p_1 \cdot \hat{\mathbf{s}})^k \right]$$

Therefore,

$$J_{\text{vol}}^{\text{out}} = A_f \mathbf{v}_1^{k+1} \cdot \hat{\mathbf{n}} + \frac{\alpha\, d_f A_f}{|\mathbf{r_1}|}(p_1^{k+1} - p_b) + A_f \nabla(\mathbf{v}_1^k \cdot \hat{\mathbf{n}}) \cdot \mathbf{r}_1 + \alpha\, d_f A_f \nabla p_1^k \cdot \hat{\mathbf{s}} \qquad \text{(A.11)}$$

Symmetry Boundary Condition

In the case of a symmetry boundary condition, the face normal component of velocity is zero.

$$J_{\text{vol}}^{\text{sym}} = 0 \qquad \text{(A.12)}$$

A.1.3 Assembling the Discrete Mass Equation

The discrete form of the volumetric flow rate across a face is expressed in Equations (A.9) for interior faces and Equations (A.10) through (A.12) for various kinds of boundary conditions. Substituting these equations into Equation (A.3) results in the discrete form of the conservation of mass for the control volumes.

A.2 Discretization of the Momentum Equation

The differential form of the momentum equation reads:

$$\nabla \cdot (\rho\, \mathbf{v} \otimes \mathbf{v}) = -\nabla p + \nabla \cdot \left[\mu \left(\nabla \mathbf{v} + \nabla \mathbf{v}^T \right) \right] \qquad \text{(A.13)}$$

Integrating the above equation over some domain Ω using Gauss-Ostrogradsky Divergence Theorem [7] gives rise to the integral form of the momentum equation.

$$\oint_\Omega (\rho \mathbf{v}\mathbf{v}) \cdot \hat{\mathbf{n}}\, dA + \oint_\Omega p \hat{\mathbf{n}}\, dA - \oint_\Omega \mu(\nabla \mathbf{v} + \nabla \mathbf{v}^T) \cdot \hat{\mathbf{n}}\, dA = 0$$

Assuming that Ω is a control volume, we may break the above integrals into separate integrals over the faces of the control volume,

$$\sum_{\text{faces}} \left(\int_{\text{face}} (\rho \mathbf{v}\mathbf{v}) \cdot \hat{\mathbf{n}}\, dA + \int_{\text{face}} p \hat{\mathbf{n}}\, dA - \int_{\text{face}} \mu(\nabla \mathbf{v} + \nabla \mathbf{v}^T) \cdot \hat{\mathbf{n}}\, dA \right) = 0 \qquad \text{(A.14)}$$

and discretize the values of u, v, p, and their derivatives at the faces of the control volume. In the following subsections, we present the details of the discretization for interior faces and boundary faces, respectively.

A.2.1 Momentum Flow Rate across an Interior Face

The x component of the momentum flow rate across an interior face reads the following formula:

$$J_{x-\text{mom}} = \underbrace{\int_{\text{face}} (\rho \tilde{V}_n) u \, dA}_{J_{\text{adv}}} + \underbrace{\int_{\text{face}} p n_x \, dA}_{F_{\text{pres}}} - \underbrace{\int_{\text{face}} \mu \left(\nabla u + \frac{\partial \mathbf{v}}{\partial x} \right) \cdot \hat{\mathbf{n}} \, dA}_{F_{\text{visc}}} \tag{A.15}$$

Note that the mass flow rate across a face is,

$$\dot{m} = \rho J_{\text{vol}} = \rho \tilde{V}_n A_f$$

The discreitzation of the other terms in Equation (A.19) is as follows:

$$\begin{aligned}
u|_{\text{face}}^{\text{int}} &= u_{\text{up}}^{k+1} + \nabla u_{\text{up}}^{k} \cdot \mathbf{r}_{\text{up}} \\
p|_{\text{face}}^{\text{int}} &= \frac{1}{2}(p_1 + p_2)^{k+1} + \frac{1}{4}(\nabla p_1 + \nabla p_2)^k \cdot (\mathbf{r}_1 + \mathbf{r}_2) \\
\nabla u \cdot \hat{\mathbf{n}}|_{\text{face}}^{\text{int}} &= \alpha \left(\frac{u_2 - u_1}{|\mathbf{s}|} \right)^{k+1} + \frac{1}{2}(\nabla u_1 + \nabla u_2)^k \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) \\
\frac{\partial \mathbf{v}}{\partial x} \cdot \hat{\mathbf{n}} \Big|_{\text{face}}^{\text{int}} &= \frac{1}{2}(\nabla \mathbf{v}_1 + \nabla \mathbf{v}_2)^k : (\hat{\mathbf{n}} \otimes \hat{\mathbf{i}})
\end{aligned}$$

Substituting the above relations into Equation (A.15) gives rise to the discretized form of $J_{x-\text{mom}}$.

$$\begin{aligned}
J_{x-\text{mom}}^{\text{int}} = \rho J_{\text{vol}}^{\text{int}} \, u_{\text{up}}^{k+1} &+ \frac{\mu \alpha A_f}{|\mathbf{s}|}(u_1 - u_2)^{k+1} + \frac{n_x A_f}{2}(p_1 + p_2)^{k+1} + \\
&\rho J_{\text{vol}}^{\text{int}} (\nabla u_{\text{up}} \cdot \mathbf{r}_{\text{up}})^k - \frac{\mu A_f}{2}(\nabla \mathbf{v}_1 + \nabla \mathbf{v}_2)^k : (\hat{\mathbf{n}} \otimes \hat{\mathbf{i}}) + \\
&\frac{n_x A_f}{4}(\nabla p_1 + \nabla p_2)^k \cdot (\mathbf{r}_1 + \mathbf{r}_2) - \frac{\mu A_f}{2}(\nabla u_1 + \nabla u_2)^k \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}})
\end{aligned} \tag{A.16}$$

where $J_{\text{vol}}^{\text{int}}$ can be obtained from Equation (A.9). The discretization of the y component of the momentum flow across an interior face is almost identical to that of the x component. The y component of the momentum flow is:

$$J_{y-\text{mom}} = \underbrace{\int_{\text{face}} (\rho \tilde{V}_n) v \, dA}_{J_{\text{adv}}} + \underbrace{\int_{\text{face}} p n_y \, dA}_{F_{\text{pres}}} - \underbrace{\int_{\text{face}} \mu \left(\nabla v + \frac{\partial \mathbf{v}}{\partial y} \right) \cdot \hat{\mathbf{n}} \, dA}_{F_{\text{visc}}} \tag{A.17}$$

The discreitzation of the terms in the above equation is as follows:

$$
\begin{aligned}
v|_{\text{face}}^{\text{int}} &= v_{\text{up}}^{k+1} + \nabla v_{\text{up}}^{k} \cdot \mathbf{r}_{\text{up}} \\
p|_{\text{face}}^{\text{int}} &= \frac{1}{2}(p_1 + p_2)^{k+1} + \frac{1}{4}(\nabla p_1 + \nabla p_2)^{k} \cdot (\mathbf{r}_1 + \mathbf{r}_2) \\
\nabla v \cdot \hat{\mathbf{n}}|_{\text{face}}^{\text{int}} &= \alpha \left(\frac{v_2 - v_1}{|\mathbf{s}|} \right)^{k+1} + \frac{1}{2}(\nabla v_1 + \nabla v_2)^{k} \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) \\
\frac{\partial \mathbf{v}}{\partial y} \cdot \hat{\mathbf{n}} \Big|_{\text{face}}^{\text{int}} &= \frac{1}{2}(\nabla \mathbf{v}_1 + \nabla \mathbf{v}_2)^{k} : (\hat{\mathbf{n}} \otimes \hat{\mathbf{j}})
\end{aligned}
$$

Substituting these relations into Equation (A.17) gives rise to the discretized form of $J_{y-\text{mom}}$.

$$
\begin{aligned}
J_{y-\text{mom}}^{\text{int}} = \rho J_{\text{vol}}^{\text{int}} v_{\text{up}}^{k+1} &+ \frac{\mu \alpha A_f}{|\mathbf{s}|}(v_1 - v_2)^{k+1} + \frac{n_y A_f}{2}(p_1 + p_2)^{k+1} + \\
&\rho J_{\text{vol}}^{\text{int}} (\nabla v_{\text{up}} \cdot \mathbf{r}_{\text{up}})^{k} - \frac{\mu A_f}{2}(\nabla \mathbf{v}_1 + \nabla \mathbf{v}_2)^{k} : (\hat{\mathbf{n}} \otimes \hat{\mathbf{j}}) + \\
&\frac{n_y A_f}{4}(\nabla p_1 + \nabla p_2)^{k} \cdot (\mathbf{r}_1 + \mathbf{r}_2) - \frac{\mu A_f}{2}(\nabla v_1 + \nabla v_2)^{k} \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}})
\end{aligned}
\tag{A.18}
$$

Equations (A.16) and (A.18) represent the discretized form of the x and y components of the momentum flow across an interior face, respectively.

A.2.2 Flow of Momentum across a Boundary Face

The discretization of the momentum flow across a boundary face depends on the boundary condition, prescribed on the face. In the following paragraphs, we discuss how to discretize the momentum flow across a boundary face with various boundary conditions.

No-Slip Wall Boundary Conditions

In the case of no-slip wall boundary condition, it is more convenient to resolve the momentum equation into wall-tangential and wall-normal components rather than x and y components.

Equations (A.15) and (A.17) express the x and y components of the integral momentum equation. Combining these equations results in a formula for the vector form of the momentum flow across a no-slip wall.

$$\mathbf{J}_{\text{mom}} = \int_{\text{face}} \left[(\rho \tilde{V}_n)\mathbf{v} + p\hat{\mathbf{n}} - \mu(\nabla \mathbf{v} + \nabla \mathbf{v}^T) \cdot \hat{\mathbf{n}} \right] dA \tag{A.19}$$

We can resolve the latter into wall-normal and wall-tangential components. Let us start with the wall-normal direction.

$$J_{n-\text{mom}} = \mathbf{J}_{\text{mom}} \cdot \hat{\mathbf{n}} = \int_{\text{face}} \left[(\rho \tilde{V}_n) V_n + p - \mu(\nabla \mathbf{v} + \nabla \mathbf{v}^T) : (\hat{\mathbf{n}} \otimes \hat{\mathbf{n}}) \right] dA \tag{A.20}$$

The last term in the square bracket is the wall-normal viscous stress. The expanded form of this term in the wall-normal coordinate system is:

$$(\nabla \mathbf{v} + \nabla \mathbf{v}^T) : (\hat{\mathbf{n}} \otimes \hat{\mathbf{n}}) = \begin{bmatrix} 2\frac{\partial V_n}{\partial n} & \frac{\partial V_n}{\partial t} + \frac{\partial V_t}{\partial n} \\ \frac{\partial V_n}{\partial t} + \frac{\partial V_t}{\partial n} & 2\frac{\partial V_t}{\partial t} \end{bmatrix} : \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} = 2\frac{\partial V_n}{\partial n}$$

where the subscripts n and t designate the wall-normal and wall-tangential components, respectively. The right hand side derivative, $\partial V_n / \partial n$, is zero for a solid wall due to the conservation of mass. Therefore Equation (A.20) reduces to the following equation:

$$J^{\text{wall}}_{n-\text{mom}} = \int_{\text{face}} \left[(\rho \tilde{V}_n)(\mathbf{v} \cdot \hat{\mathbf{n}}) + p \right] dA \tag{A.21}$$

Similarly, the wall-tangential component of Equation (A.19) is:

$$J_{t-\text{mom}} = \mathbf{J}_{\text{mom}} \cdot \hat{\mathbf{t}} = \int_{\text{face}} \left[(\rho \tilde{V}_n)(\mathbf{v} \cdot \hat{\mathbf{t}}) - \mu(\nabla \mathbf{v} + \nabla \mathbf{v}^T) : (\hat{\mathbf{t}} \otimes \hat{\mathbf{n}}) \right] dA \tag{A.22}$$

The last term on the right hand side is the viscous shear force at the wall that can be simplified as follows:

$$\left(\nabla \mathbf{v}_f + \nabla \mathbf{v}_f^T\right) : (\hat{\mathbf{t}} \otimes \hat{\mathbf{n}}) = \begin{bmatrix} 2\frac{\partial V_n}{\partial n} & \frac{\partial V_n}{\partial t} + \frac{\partial V_t}{\partial n} \\ \frac{\partial V_n}{\partial t} + \frac{\partial V_t}{\partial n} & 2\frac{\partial V_t}{\partial t} \end{bmatrix} : \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} = \frac{\partial V_n}{\partial t} + \frac{\partial V_t}{\partial n}$$

Therefore the face-tangential component of the momentum flow across the wall becomes:

$$J^{\text{wall}}_{t-\text{mom}} = \int_{\text{face}} \left[(\rho \tilde{V}_n) V_t - \mu \left(\frac{\partial V_n}{\partial t} + \frac{\partial V_t}{\partial n} \right) \right] dA \tag{A.23}$$

In Equations (A.21) and (A.23), $J_{n-\text{mom}}$ and $J_{t-\text{mom}}$ are the wall-normal and wall-tangential components of the momentum flow. However, we need to discretize the momentum flow in x and y directions. Therefore we have to combine Equations (A.21) and (A.23) and then resolve the result into x and y components.

$$\mathbf{J}_{\text{mom}}^{\text{wall}} = J_{n-\text{mom}}^{\text{wall}}\hat{\mathbf{n}} + J_{t-\text{mom}}^{\text{wall}}\hat{\mathbf{t}} = \int_{\text{face}} \left[(\rho\tilde{V}_n)\mathbf{v} + p\hat{\mathbf{n}} - \mu \left(\frac{\partial V_n}{\partial t} + \frac{\partial V_t}{\partial n} \right) \hat{\mathbf{t}} \right] dA \qquad \text{(A.24)}$$

Note that the velocity at a no-slip wall is prescribed, $\mathbf{v} = \mathbf{v}_b$. The discretization of the other terms in the above equation is as follows.

$$\begin{aligned} p|_{\text{face}}^{\text{wall}} &= p_1^{k+1} + \nabla p_1^k \cdot \mathbf{r}_1 \\ \left.\frac{\partial V_n}{\partial t}\right|_{\text{face}}^{\text{wall}} &= \nabla \mathbf{v}_1^k : (\hat{\mathbf{n}} \otimes \hat{\mathbf{t}}) \\ \left.\frac{\partial V_t}{\partial n}\right|_{\text{face}}^{\text{wall}} &= \alpha \left(\frac{\mathbf{v}_b - \mathbf{v}_1^{k+1}}{|\mathbf{s}|} \right) \cdot \hat{\mathbf{t}} + \nabla \mathbf{v}_1^k : [\hat{\mathbf{t}} \otimes (\hat{\mathbf{n}} - \alpha\hat{\mathbf{s}})] \end{aligned}$$

Substituting the above relations into Equation (A.24) and simplifying the result yields to:

$$\begin{aligned} J_{x-\text{mom}}^{\text{wall}} &= n_x A_f p_1^{k+1} + \frac{\mu t_x \alpha A_f}{|\mathbf{s}|}(\mathbf{v}_1^{k+1} - \mathbf{v}_b) \cdot \hat{\mathbf{t}} + \rho J_{\text{vol}}^{\text{wall}} u_b + \\ &\quad n_x A_f (\nabla p_1^k \cdot \mathbf{r}_1) - \mu t_x A_f \nabla \mathbf{v}_1^k : [\hat{\mathbf{n}} \otimes \hat{\mathbf{t}} + \hat{\mathbf{t}} \otimes (\hat{\mathbf{n}} - \alpha\hat{\mathbf{s}})] \qquad \text{(A.25)} \\ J_{y-\text{mom}}^{\text{wall}} &= n_y A_f p_1^{k+1} + \frac{\mu t_y \alpha A_f}{|\mathbf{s}|}(\mathbf{v}_1^{k+1} - \mathbf{v}_b) \cdot \hat{\mathbf{t}} + \rho J_{\text{vol}}^{\text{wall}} v_b + \\ &\quad n_y A_f (\nabla p_1^k \cdot \mathbf{r}_1) - \mu t_y A_f \nabla \mathbf{v}_1^k : [\hat{\mathbf{n}} \otimes \hat{\mathbf{t}} + \hat{\mathbf{t}} \otimes (\hat{\mathbf{n}} - \alpha\hat{\mathbf{s}})] \qquad \text{(A.26)} \end{aligned}$$

Inflow Boundary Conditions

The discretization of the inflow boundary condition is very similar to that of an interior face, presented in Subsection A.2.1. However, the velocity at the inlet is prescribed and the face pressure is extrapolated from the node of the only neighbouring control volume of the face. Let us perform the discretization for the x and y components, separately.

The x component of the momentum flow at the inlet is expressed in Equation (A.15).

$$J_{x-\text{mom}} = \int_{\text{face}} (\rho\tilde{V}_n) u \, dA + \int_{\text{face}} p n_x \, dA - \int_{\text{face}} \mu \left(\nabla u + \frac{\partial \mathbf{v}}{\partial x} \right) \cdot \hat{\mathbf{n}} \, dA$$

where

$$\begin{aligned} u|_{\text{face}}^{\text{in}} &= u_b \\ p|_{\text{face}}^{\text{in}} &= p_1^{k+1} + \nabla p_1^k \cdot \mathbf{r}_1 \\ \nabla u \cdot \hat{\mathbf{n}}|_{\text{face}}^{\text{in}} &= \alpha \left(\frac{u_b - u_1^{k+1}}{|\mathbf{s}|} \right) + \nabla u_1^k \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) \\ \frac{\partial \mathbf{v}}{\partial x} \cdot \hat{\mathbf{n}} \Big|_{\text{face}}^{\text{in}} &= \nabla \mathbf{v}_1^k : (\hat{\mathbf{n}} \otimes \hat{\mathbf{i}}) \end{aligned}$$

Substituting the above relations into the equation for $J_{x-\text{mom}}$ results in,

$$\begin{aligned} J_{x-\text{mom}}^{\text{in}} &= \frac{\mu \alpha A_f}{|\mathbf{s}|} u_1^{k+1} + n_x A_f p_1^{k+1} + \left(\rho J_{\text{vol}}^{\text{in}} - \frac{\mu \alpha A_f}{|\mathbf{s}|} \right) u_b + \\ & \quad n_x A_f (\nabla p_1^k \cdot \mathbf{r_1}) - \mu A_f \left[\nabla u_1^k \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) + \nabla \mathbf{v}_1^k : (\hat{\mathbf{n}} \otimes \hat{\mathbf{i}}) \right] \end{aligned} \tag{A.27}$$

The y component of the momentum flow at the inlet is expressed in Equation (A.17).

$$J_{y-\text{mom}} = \int_{\text{face}} (\rho \tilde{V}_n) v \, dA + \int_{\text{face}} p n_y \, dA - \int_{\text{face}} \mu \left(\nabla v + \frac{\partial \mathbf{v}}{\partial y} \right) \cdot \hat{\mathbf{n}} \, dA$$

where

$$\begin{aligned} v|_{\text{face}}^{\text{in}} &= v_b^{k+1} \\ p|_{\text{face}}^{\text{in}} &= p_1^{k+1} + \nabla p_1^k \cdot \mathbf{r}_1 \\ \nabla v \cdot \hat{\mathbf{n}}|_{\text{face}}^{\text{in}} &= \alpha \left(\frac{v_b - v_1^{k+1}}{|\mathbf{s}|} \right) + \nabla v_1^k \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) \\ \frac{\partial \mathbf{v}}{\partial y} \cdot \hat{\mathbf{n}} \Big|_{\text{face}}^{\text{in}} &= \nabla \mathbf{v}_1^k : (\hat{\mathbf{n}} \otimes \hat{\mathbf{j}}) \end{aligned}$$

Substituting the above relations into the equation for $J_{y-\text{mom}}$ results in,

$$\begin{aligned} J_{y-\text{mom}}^{\text{in}} &= \frac{\mu \alpha A_f}{|\mathbf{s}|} v_1^{k+1} + n_y A_f p_1^{k+1} + \left(\rho J_{\text{vol}}^{\text{in}} - \frac{\mu \alpha A_f}{|\mathbf{s}|} \right) v_b + \\ & \quad n_y A_f (\nabla p_1^k \cdot \mathbf{r_1}) - \mu A_f \left[\nabla v_1^k \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) + \nabla \mathbf{v}_1^k : (\hat{\mathbf{n}} \otimes \hat{\mathbf{j}}) \right] \end{aligned} \tag{A.28}$$

Outflow Boundary Conditions

The discretization of the outflow boundary condition is similar to that of an interior face. However, the face pressure is prescribed at the outlet boundary. The x component of the momentum flow, expressed in Equation (A.15), is:

$$J_{x-\text{mom}} = \int_{\text{face}} (\rho \tilde{V}_n) u \, dA + \int_{\text{face}} p n_x \, dA - \int_{\text{face}} \mu \left(\nabla u + \frac{\partial \mathbf{v}}{\partial x} \right) \cdot \hat{\mathbf{n}} \, dA$$

where

$$\begin{aligned} u|_{\text{face}}^{\text{out}} &= u_1^{k+1} + \nabla u_1^k \cdot \mathbf{r}_1 \\ p|_{\text{face}}^{\text{out}} &= p_b \\ \nabla u \cdot \hat{\mathbf{n}}|_{\text{face}}^{\text{out}} &= \nabla u_1^k \cdot \hat{\mathbf{n}} \\ \left. \frac{\partial \mathbf{v}}{\partial x} \cdot \hat{\mathbf{n}} \right|_{\text{face}}^{\text{out}} &= \nabla \mathbf{v}_1^k : (\hat{\mathbf{n}} \otimes \hat{\mathbf{i}}) \end{aligned}$$

Therefore,

$$J_{x-\text{mom}}^{\text{out}} = \rho J_{\text{vol}}^{\text{out}} u_1^{k+1} + p_b n_x A_f + \rho J_{\text{vol}}^{\text{out}} (\nabla u_1^k \cdot \mathbf{r}_1) - \mu A_f \left[\nabla u_1^k \cdot \hat{\mathbf{n}} + \nabla \mathbf{v}_1^k : (\hat{\mathbf{n}} \otimes \hat{\mathbf{i}}) \right] \quad \text{(A.29)}$$

Similarly the y component of the momentum flow, expressed in Equation (A.17), is:

$$J_{y-\text{mom}} = \int_{\text{face}} (\rho \tilde{V}_n) v \, dA + \int_{\text{face}} p n_y \, dA - \int_{\text{face}} \mu \left(\nabla v + \frac{\partial \mathbf{v}}{\partial y} \right) \cdot \hat{\mathbf{n}} \, dA$$

where

$$\begin{aligned} v|_{\text{face}}^{\text{out}} &= v_1^{k+1} + \nabla v_1^k \cdot \mathbf{r}_1 \\ p|_{\text{face}}^{\text{out}} &= p_b \\ \nabla v \cdot \hat{\mathbf{n}}|_{\text{face}}^{\text{out}} &= \nabla v_1^k \cdot \hat{\mathbf{n}} \\ \left. \frac{\partial \mathbf{v}}{\partial y} \cdot \hat{\mathbf{n}} \right|_{\text{face}}^{\text{out}} &= \nabla \mathbf{v}_1^k : (\hat{\mathbf{n}} \otimes \hat{\mathbf{j}}) \end{aligned}$$

Therefore,

$$J_{y-\text{mom}}^{\text{out}} = \rho J_{\text{vol}}^{\text{out}} v_1^{k+1} + p_b n_y A_f + \rho J_{\text{vol}}^{\text{out}} (\nabla v_1^k \cdot \mathbf{r}_1) - \mu A_f \left[\nabla v_1^k \cdot \hat{\mathbf{n}} + \nabla \mathbf{v}_1^k : (\hat{\mathbf{n}} \otimes \hat{\mathbf{j}}) \right] \quad \text{(A.30)}$$

Outflow Boundary Conditions without Normal Stress

Sometimes we need to suppress the normal stress at the outflow boundary since it might adversely affect the simulation stability and convergence. In these cases we have to modify the discretization formulae, presented in the previous subsection, in order to eliminate the face-normal viscous stress. The momentum flow vector across a face, Equation (A.23), is:

$$\mathbf{J}_{\text{mom}} = \int_{\text{face}} \left[(\rho \tilde{V}_n)\mathbf{v} + p\hat{\mathbf{n}} - \mu(\nabla \mathbf{v} + \nabla \mathbf{v}^T) \cdot \hat{\mathbf{n}} \right] dA$$

We need to resolve the above flow vector into face-normal and face-tangential components and eliminate the normal-stress in the face-normal direction. This process is very similar to the discretization of the no-slip wall boundary condition, presented in Equations (A.21) and (A.23).

$$\begin{aligned} J^{\text{out}}_{n-\text{mom}} &= \int_{\text{face}} \left[(\rho \tilde{V}_n) V_n + p \right] dA \\ J^{\text{out}}_{t-\text{mom}} &= \int_{\text{face}} \left[(\rho \tilde{V}_n) V_t - \mu \left(\frac{\partial V_n}{\partial t} + \frac{\partial V_t}{\partial n} \right) \right] dA \end{aligned}$$

Combining the above formulae results in the momentum flow vector without normal stress.

$$\mathbf{J}^{\text{out}}_{\text{mom}} = J^{\text{out}}_{n-\text{mom}} \hat{\mathbf{n}} + J^{\text{out}}_{t-\text{mom}} \hat{\mathbf{t}} = \int_{\text{face}} \left[(\rho \tilde{V}_n)\mathbf{v} + p\hat{\mathbf{n}} - \mu A_f \left(\frac{\partial V_n}{\partial t} + \frac{\partial V_t}{\partial n} \right) \hat{\mathbf{t}} \right] dA \quad \text{(A.31)}$$

As seen, the latter is the same as momentum flow across a no-slip wall, Equation (A.24). However, the discretized form of the latter and Equation (A.24) are different because the velocity at the outflow boundary is not prescribed and must be extrapolated from within the domain.

$$\begin{aligned} p|^{\text{out}}_{\text{face}} &= p_b \\ \mathbf{v}|^{\text{out}}_{\text{face}} &= \mathbf{v}_1^{k+1} + \nabla \mathbf{v}_1^k \cdot \mathbf{r}_1 \\ \left. \frac{\partial V_n}{\partial t} \right|^{\text{out}}_{\text{face}} &= \nabla \mathbf{v}_1^k : (\hat{\mathbf{n}} \otimes \hat{\mathbf{t}}) \\ \left. \frac{\partial V_t}{\partial n} \right|^{\text{out}}_{\text{face}} &= \nabla \mathbf{v}_1^k : (\hat{\mathbf{t}} \otimes \hat{\mathbf{n}}) \end{aligned}$$

Resolving Equation (A.31) along x and y directions and substituting the above relations in it yields the discretized form of the momentum flow across the outflow boundary.

$$\begin{aligned} J^{\text{out}}_{x-\text{mom}} &= \rho J^{\text{out}}_{\text{vol}} u_1^{k+1} + p_b n_x A_f + \rho J^{\text{out}}_{\text{vol}} (\nabla u_1^k \cdot \mathbf{r}_1) - \\ &\quad \mu A_f t_x \nabla \mathbf{v}_1^k : (\hat{\mathbf{n}} \otimes \hat{\mathbf{t}} + \hat{\mathbf{t}} \otimes \hat{\mathbf{n}}) \qquad \text{(A.32)} \\ J^{\text{out}}_{y-\text{mom}} &= \rho J^{\text{out}}_{\text{vol}} v_1^{k+1} + p_b n_y A_f + \rho J^{\text{out}}_{\text{vol}} (\nabla v_1^k \cdot \mathbf{r}_1) - \\ &\quad \mu A_f t_y \nabla \mathbf{v}_1^k : (\hat{\mathbf{n}} \otimes \hat{\mathbf{t}} + \hat{\mathbf{t}} \otimes \hat{\mathbf{n}}) \qquad \text{(A.33)} \end{aligned}$$

Symmetry Boundary Conditions

Symmetry boundary condition is the last kind of boundary condition that we discuss in this appendix. Although this kind of boundary condition appears to be simple, its discretization takes some effort. Let us start with the equation for the momentum flow across a face, Equation (A.23).

$$\mathbf{J}_{\text{mom}} = \int_{\text{face}} \left[(\rho \tilde{V}_n)\mathbf{v} + p\hat{\mathbf{n}} - \mu(\nabla \mathbf{v} + \nabla \mathbf{v}^T) \cdot \hat{\mathbf{n}} \right] dA$$

Let us resolve the momentum flow vector, $\mathbf{J}_{\text{mom}}$, into face-normal and face tangential directions.

$$\begin{aligned} J^{\text{sym}}_{n-\text{mom}} &= \int_{\text{face}} \left[(\rho \tilde{V}_n) V_n + p - 2\mu \frac{\partial V_n}{\partial n} \right] dA = \int_{\text{face}} \left(p - 2\mu \frac{\partial V_n}{\partial n} \right) dA \\ J^{\text{sym}}_{t-\text{mom}} &= \int_{\text{face}} \left[(\rho \tilde{V}_n) V_t - \mu \left(\frac{\partial V_n}{\partial t} + \frac{\partial V_t}{\partial n} \right) \right] dA = 0 \end{aligned}$$

Note that the advecting velocity, $\tilde{V}_n$, across a symmetry plane is zero. In addition, the shear stress at a symmetry surface is zero as well. Therefore the momentum flow vector reduces to:

$$\mathbf{J}^{\text{sym}}_{\text{mom}} = J^{\text{sym}}_{n-\text{mom}} \hat{\mathbf{n}} = \int_{\text{face}} \left(p - 2\mu \frac{\partial V_n}{\partial n} \hat{\mathbf{n}} \right) dA \qquad \text{(A.34)}$$

where,

$$\begin{aligned}
p\big|_{\text{face}}^{\text{sym}} &= p_1^{k+1} + \nabla p_1^k \cdot \mathbf{r}_1 \\
\frac{\partial V_n}{\partial n}\bigg|_{\text{face}}^{\text{sym}} &= n_x \left.\frac{\partial u}{\partial n}\right|_{\text{face}}^{\text{sym}} + n_y \left.\frac{\partial v}{\partial n}\right|_{\text{face}}^{\text{sym}} \\
\frac{\partial u}{\partial n}\bigg|_{\text{face}}^{\text{sym}} &= \alpha \left(\frac{u_b - u_1^{k+1}}{|\mathbf{s}|} \right) + \nabla u^k \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) \\
\frac{\partial v}{\partial n}\bigg|_{\text{face}}^{\text{sym}} &= \alpha \left(\frac{v_b - v_1^{k+1}}{|\mathbf{s}|} \right) + \nabla v^k \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}})
\end{aligned}$$

Note that the velocity at a symmetry boundary, $\mathbf{v}_b$, is not given. However, the velocity must be parallel to the symmetry boundary.

$$\mathbf{v}_b \cdot \hat{\mathbf{n}} = 0$$

Substituting these relations into Equation (A.34) results in the disretized form of the momentum flow across a symmetry boundary.

$$\begin{aligned}
J_{x-\text{mom}}^{\text{sym}} &= p_1^{k+1} n_x A_f - \left(\frac{2\mu\alpha n_x A_f}{|\mathbf{s}|} \right) (\mathbf{v}_1^{k+1} \cdot \hat{\mathbf{n}}) + n_x A_f (\nabla p^k \cdot \mathbf{r}_1) - \\
&\quad 2\mu n_x A_f \left[n_x \nabla u^k \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) + n_y \nabla v^k \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) \right] \qquad \text{(A.35)} \\
J_{y-\text{mom}}^{\text{sym}} &= p_1^{k+1} n_y A_f - \left(\frac{2\mu\alpha n_y A_f}{|\mathbf{s}|} \right) (\mathbf{v}_1^{k+1} \cdot \hat{\mathbf{n}}) + n_y A_f (\nabla p^k \cdot \mathbf{r}_1) - \\
&\quad 2\mu n_y A_f \left[n_x \nabla u^k \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) + n_y \nabla v^k \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) \right] \qquad \text{(A.36)}
\end{aligned}$$

Appendix B

Estimating Residual of a Second-Order Accurate Method

The objective of this appendix is to prove Equation (3.26).

$$\mathcal{L}_h^2(\phi_h^3) \approx -\mathcal{L}_h^3(\phi_h^2)$$

Suppose we want to solve the following differential equation, as expressed in Equation (3.6):

$$\mathscr{L}(\mathbf{\Phi}) = 0 \tag{B.1}$$

where $\mathscr{L}$ is the differential operator for the mass and momentum equations and $\mathbf{\Phi}$ is the exact solution.

$$\mathbf{\Phi} = \begin{pmatrix} p \\ u \\ v \end{pmatrix} \qquad \mathscr{L}(\mathbf{\Phi}) = \begin{pmatrix} \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \\ \rho u \frac{\partial u}{\partial x} + \rho v \frac{\partial u}{\partial y} + \frac{\partial p}{\partial x} - \mu(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}) \\ \rho u \frac{\partial v}{\partial x} + \rho v \frac{\partial v}{\partial y} + \frac{\partial p}{\partial y} - \mu(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}) \end{pmatrix}$$

To solve the above equation numerically, we have to discretize it. In this work, we use a second-order cell-centred finite volume method for this purpose.

$$\mathcal{L}_h^2(\boldsymbol{\phi}_h^2) = 0 \tag{B.2}$$

where $\mathcal{L}_h^2$ is the second-order accurate discrete operator for the mass and momentum equations. Conceptually, this operator represents the net flow of mass and momentum

across the faces of each control volume based on the discrete solution at the control volume nodes. For example, Equations (A.9), (A.16), and (A.18) represent the discrete form of the mass, x-momentum, and y-momentum flows across an interior face. If we assemble these equations for all faces of a certain control volume, we obtain the operator $\mathcal{L}_h^2$ for that control volume. The solution of the above equation, $\boldsymbol{\phi}_h^2$, is the second-order accurate solution.

$$\boldsymbol{\phi}_h^2 - \boldsymbol{\Phi} = \mathcal{O}(h^2) \tag{B.3}$$

where h is the cell characteristic size. We can also discretize Equation (B.1) using a third-order accurate finite volume method.

$$\mathcal{L}_h^3(\boldsymbol{\phi}_h^3) = 0 \tag{B.4}$$

where $\mathcal{L}_h^3$ is the third-order accurate discrete operator for the mass and momentum equations and $\boldsymbol{\phi}_h^3$ is the third-order accurate solution.

$$\boldsymbol{\phi}_h^3 - \boldsymbol{\Phi} = \mathcal{O}(h^3) \tag{B.5}$$

A comparison between Equations (B.3) and (B.5) shows that:

$$\boldsymbol{\phi}_h^2 - \boldsymbol{\phi}_h^3 = \mathcal{O}(h^2) \tag{B.6}$$

Therefore we can consider $\boldsymbol{\phi}_h^3$ to be an approximation to the exact solution.

An examination of the operators $\mathcal{L}_h^2$ and $\mathcal{L}_h^3$ shows that if we apply them to an arbitrary vector, $\boldsymbol{\phi}_h$, the right hand side would be proportional to the mesh size.

$$\mathcal{L}_h^2(\boldsymbol{\phi}_h) = \mathcal{O}(h) \tag{B.7}$$

$$\mathcal{L}_h^3(\boldsymbol{\phi}_h) = \mathcal{O}(h) \tag{B.8}$$

To understand the above relations we need to analyse the structure of the operators $\mathcal{L}_h^2$ and $\mathcal{L}_h^3$. these operators consist of a sum of terms that are expressions in $\boldsymbol{\phi}_h$ and its derivatives, multiplied by geometric factors. These geometric factors, which scale with the powers of cell sizes, determine the asymptotic behaviour of the discretization scheme on fine meshes. In two-dimensional problems, the leading geometric factor is proportional to the cell size, h. The reason is that in a two-dimensional problem, the flow across a

face is proportional to the face area, which is proportional to the cell size. Therefore the application of a discrete operator $\mathcal{L}_h$ on an arbitrary vector is always proportional to the cell size h, expressed in Equations (B.7) and (B.8).

Note that the second-order solution, $\boldsymbol{\phi}_h^2$, and the third-order solution, $\boldsymbol{\phi}_h^3$, are close on a fine mesh since they are solutions of the same problem. Therefore we can linearize Equations (B.2) and (B.4) about $\boldsymbol{\phi}_h^3$ using a first-order Taylor series expansion:

$$\underbrace{\mathcal{L}_h^2(\boldsymbol{\phi}_h^2)}_{=0} \approx \mathcal{L}_h^2(\boldsymbol{\phi}_h^3) + \underbrace{\left.\frac{d\mathcal{L}_h^2(\boldsymbol{\phi}_h)}{d\boldsymbol{\phi}_h}\right|_{\boldsymbol{\phi}_h=\boldsymbol{\phi}_h^3}}_{\mathcal{O}(h)} \underbrace{(\boldsymbol{\phi}_h^2 - \boldsymbol{\phi}_h^3)}_{\mathcal{O}(h^2)} \tag{B.9}$$

$$\mathcal{L}_h^3(\boldsymbol{\phi}_h^2) \approx \underbrace{\mathcal{L}_h^3(\boldsymbol{\phi}_h^3)}_{=0} + \underbrace{\left.\frac{d\mathcal{L}_h^3(\boldsymbol{\phi}_h)}{d\boldsymbol{\phi}_h}\right|_{\boldsymbol{\phi}_h=\boldsymbol{\phi}_h^3}}_{\mathcal{O}(h)} \underbrace{(\boldsymbol{\phi}_h^2 - \boldsymbol{\phi}_h^3)}_{\mathcal{O}(h^2)} \tag{B.10}$$

The expression $d\mathcal{L}_h^n(\boldsymbol{\phi}_h)/d\boldsymbol{\phi}_h$ for $n = 2$ or 3 is a matrix in which the elements show the sensitivity of the net flow across the faces of each control volume with respect to the variation of $\boldsymbol{\phi}_h$ in other control volume. For example, the element (i, j) of this matrix indicates the rate of change of the net flows of mass and momentum equations across the faces of control volume i when the value of $\boldsymbol{\phi}_h$ in the control volume j changes. The structure of each element of the matrix $d\mathcal{L}_h^n(\boldsymbol{\phi}_h)/d\boldsymbol{\phi}_h$ is similar to the structure of the matrix $\mathcal{L}_h^n(\boldsymbol{\phi}_h)$. Therefore we can conclude that based on the same assumptions, the elements of this matrix are proportional to the cell size, h, and Equations (B.9) and (B.10) result in the following relations:

$$\mathcal{L}_h^2(\boldsymbol{\phi}_h^3) = \mathcal{O}(h^3) \tag{B.11}$$

$$\mathcal{L}_h^3(\boldsymbol{\phi}_h^2) = \mathcal{O}(h^3) \tag{B.12}$$

Now, let us subtract Equation (B.9) from (B.10):

$$\mathcal{L}_h^3(\boldsymbol{\phi}_h^2) + \mathcal{L}_h^2(\boldsymbol{\phi}_h^3) \approx \frac{d}{d\boldsymbol{\phi}_h} \underbrace{\left[\mathcal{L}_h^3(\boldsymbol{\phi}_h) - \mathcal{L}_h^2(\boldsymbol{\phi}_h)\right]_{\boldsymbol{\phi}_h=\boldsymbol{\phi}_h^3}}_{\mathcal{L}_h^{3-2}(\boldsymbol{\phi}_h)} (\boldsymbol{\phi}_h^2 - \boldsymbol{\phi}_h^3) \tag{B.13}$$

where $\mathcal{L}_h^{3-2}(\boldsymbol{\phi}_h)$ is the difference between the second and the third order operators, defined based on Equations (C.11), (C.22), and (C.33) for the mass and momentum equations,

respectively. The terms in the operator $\mathcal{L}_h^{3-2}$ contain the derivatives of $(\boldsymbol{\phi}_h)$ multiplied by the geometric factors. However in this operator, the leading geometric factors scale with h^3. The reason is that the operator $\mathcal{L}_h^{3-2}$ represents the flow of quadratic terms across the faces of each control volume. The quadratic terms scale with h^2 and the area of control volume faces is proportional to h. Therefore

$$\mathcal{L}_h^{3-2}(\boldsymbol{\phi}_h) = \mathcal{O}(h^3) \tag{B.14}$$

Also note that the structure of the elements of the matrix $d\mathcal{L}_h^{3-2}(\boldsymbol{\phi}_h)/d\boldsymbol{\phi}_h$ is similar to the structure of the elements of $\mathcal{L}_h^{3-2}(\boldsymbol{\phi}_h)$. Therefore we can conclude that

$$\frac{d\mathcal{L}_h^{3-2}(\boldsymbol{\phi}_h)}{d\boldsymbol{\phi}_h} = \frac{d}{d\boldsymbol{\phi}_h}\Big[\mathcal{L}_h^3(\boldsymbol{\phi}_h) - \mathcal{L}_h^2(\boldsymbol{\phi}_h)\Big] = \mathcal{O}(h^3) \tag{B.15}$$

Substituting the results of Equations (B.6) and (B.15) into Equation (B.13) yields:

$$\mathcal{L}_h^3(\boldsymbol{\phi}_h^2) + \mathcal{L}_h^2(\boldsymbol{\phi}_h^3) = \mathcal{O}(h^5) \tag{B.16}$$

A comparison between the latter and Equations (B.11) and (B.12) reveals that on a fine mesh the magnitude of $\mathcal{L}_h^3(\boldsymbol{\phi}_h^2) + \mathcal{L}_h^2(\boldsymbol{\phi}_h^3)$ is smaller than that of either $\mathcal{L}_h^3(\boldsymbol{\phi}_h^2)$ or $\mathcal{L}_h^2(\boldsymbol{\phi}_h^3)$. Therefore we conclude that

$$\mathcal{L}_h^3(\boldsymbol{\phi}_h^2) \approx -\mathcal{L}_h^2(\boldsymbol{\phi}_h^3) \tag{B.17}$$

and the proof is complete.

Appendix C

Face Flow Error and Residual Estimation

In Chapter 3, we defined the concept of residual as the source of discretization error in a numerical solution. In this appendix, we derive the equations that provide an estimate of the solution residual.

The first step towards estimating the residual is the evaluation of face flow errors. Equations (3.21) and (3.22) express the relation between face flow errors and residual as follows.

$$\delta_{\text{mass}} = \sum_{\text{face}} \Delta J_{\text{vol}} \tag{C.1}$$

$$\boldsymbol{\delta}_{\text{mom}} = \sum_{\text{face}} (\Delta \mathbf{J}_{\text{adv}} - \Delta \mathbf{F}_{\text{pres}} - \Delta \mathbf{F}_{\text{visc}}) \tag{C.2}$$

Therefore the solution residual of a control volume is equal to the sum of the flow and force errors across the faces of the control volume.

In the following sections, we discuss how to estimate the face flow errors for the mass and momentum equations. The approach is based on the estimation of the neglected terms in the Taylor series expansions of the discretized equations. For example in the case that the discretization scheme is second-order accurate, presented in Appendix A, we may estimate the face flow errors by estimating the neglected third-order terms in the discretized equations. Then we can calculate the residual using Equations (C.1) and (C.2).

C.1 Error in Discretization of Mass Flow

The discretized form of the incompressible mass equation for a control volume, Equation (A.3), is

$$\sum_{\text{faces}} \int \tilde{V}_n \, dA = 0 \tag{C.3}$$

where $\tilde{V}_n$ is the advecting velocity, defined based on the Rhie-Chow velocity pressure interpolation method [71].

$$\tilde{V}_n = \bar{V}_n - d_f \left(\left.\frac{\partial p}{\partial n}\right|^{\text{active}} - \left.\frac{\partial p}{\partial n}\right|^{\text{lagged}} \right) \tag{C.4}$$

where $\bar{V}_n$ and $\partial p/\partial n$ are the average advected velocity and the face normal pressure gradient, respectively. We need to discretize the above parameters using a third order scheme. The third order discretization of the face advected velocity is based on a central differencing scheme explained in Appendix D.

$$\begin{aligned} \bar{V}_n|_{\text{face}} = \frac{1}{2}\left(V_{n_1} + V_{n_2}\right) + \frac{1}{4}\left(\nabla V_{n_1} + \nabla V_{n_2}\right) \cdot \left(\mathbf{r'}_1 + \mathbf{r'}_2\right) + \\ \frac{1}{8}\left(\nabla\nabla V_{n_1} + \nabla\nabla V_{n_2}\right) : \left(\mathbf{r'}_1 \otimes \mathbf{r'}_2 + \mathbf{r'}_2 \otimes \mathbf{r'}_1\right) \end{aligned} \tag{C.5}$$

where $\mathbf{r'}_1$ and $\mathbf{r'}_2$ join the nodes of the neighbouring control volumes to an arbitrary point on the face. For the discretization of the active and lagged face-normal pressure gradient terms in Equation (C.4) we use Equations (E.9) and (D.4), respectively.

$$\begin{aligned} \left.\frac{\partial p}{\partial n}\right|_{\text{face}}^{\text{active}} = \alpha \left(\frac{p_2 - p_1}{|\mathbf{s}|}\right) + \frac{1}{2}(\nabla p_1 + \nabla p_2) \cdot (\hat{\mathbf{n}} - \alpha\hat{\mathbf{s}}) + \\ \frac{1}{4}(\nabla\nabla p_1 + \nabla\nabla p_2) : \left[\frac{\alpha}{|\mathbf{s}|}(\mathbf{r}'_1 \otimes \mathbf{r}'_2 - \mathbf{r}'_2 \otimes \mathbf{r}'_1) + (\mathbf{r}'_1 + \mathbf{r}'_2) \otimes \hat{\mathbf{n}}\right] \end{aligned} \tag{C.6}$$

$$\left.\frac{\partial p}{\partial n}\right|_{\text{face}}^{\text{lagged}} = \frac{1}{2}(\nabla p_1 + \nabla p_2) \cdot \hat{\mathbf{n}} + \frac{1}{4}(\nabla\nabla p_1 + \nabla\nabla p_2) : [(\mathbf{r}'_1 + \mathbf{r}'_2) \otimes \hat{\mathbf{n}}] \tag{C.7}$$

where $\alpha = \hat{\mathbf{n}} \cdot \hat{\mathbf{s}}$ is the nonorthogonality factor, explained in Appendix E. Substituting Equations (C.5), (C.6) and (C.7) into Equation (C.4) results in the advecting velocity at

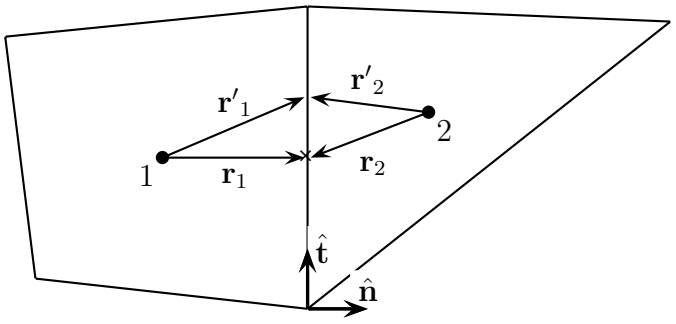


Figure C.1: Schematic of an interior face and its neighbouring control volumes. The vectors $\mathbf{r}_1$ and $\mathbf{r}_2$ are from the nodes of neighbouring control volumes 1 and 2, respectively, to the face integration point ($\times$) and the vectors $\mathbf{r'}_1$ and $\mathbf{r'}_2$ are to an arbitrary point on the face.

an arbitrary point on the face.

$$\begin{aligned}\tilde{V}_n = &\frac{1}{2}\left(V_{n_1} + V_{n_2}\right) + \frac{1}{4}\left(\nabla V_{n_1} + \nabla V_{n_2}\right)\cdot\left(\mathbf{r'}_1 + \mathbf{r'}_2\right) - \\ &\alpha d_f\left[\left(\frac{p_2 - p_1}{|\mathbf{s}|}\right) - \frac{1}{2}(\nabla p_1 + \nabla p_2)\cdot\hat{\mathbf{s}}\right] + \\ &\frac{1}{8}\left(\nabla\nabla V_{n_1} + \nabla\nabla V_{n_2}\right) : \left(\mathbf{r'}_1 \otimes \mathbf{r'}_2 + \mathbf{r'}_2 \otimes \mathbf{r'}_1\right) - \\ &\frac{\alpha d_f}{4|\mathbf{s}|}(\nabla\nabla p_1 + \nabla\nabla p_2) : \left(\mathbf{r'}_1 \otimes \mathbf{r'}_2 - \mathbf{r'}_2 \otimes \mathbf{r'}_1\right)\end{aligned} \tag{C.8}$$

Note that the last two terms of the latter are quadratic. Therefore we cannot simply assume the average velocity across the face to be equal to $\tilde{V}_n$ at the face integration point. That said, we have to integrate Equation (C.8) over the face to find the total volumetric flow rate across the face.

$$J_{\text{vol}} = \int_{\text{face}} \tilde{V}_n \, dA \tag{C.9}$$

However, the evaluation of the above integral involves using the following identities,

$$\begin{aligned}\int \left(\mathbf{r'}_1 \otimes \mathbf{r'}_2\right) dA &= \left(\mathbf{r}_1 \otimes \mathbf{r}_2 + \frac{1}{12}\mathbf{t} \otimes \mathbf{t}\right) A_f \\ \int \left(\mathbf{r'}_2 \otimes \mathbf{r'}_1\right) dA &= \left(\mathbf{r}_2 \otimes \mathbf{r}_1 + \frac{1}{12}\mathbf{t} \otimes \mathbf{t}\right) A_f\end{aligned}$$

where $\mathbf{t}$ is a vector tangent to the face in direction and equal to A_f in magnitude. Therefore the total volumetric flow across a face is:

$$\begin{aligned} J_{\text{vol}} = & \frac{A_f}{2}(\mathbf{v}_1 + \mathbf{v}_2) \cdot \hat{\mathbf{n}} + \frac{\alpha d_f A_f}{|\mathbf{s}|}(p_1 - p_2) + \\ & \frac{A_f}{4}\left[\nabla(\mathbf{v}_1 \cdot \hat{\mathbf{n}}) + \nabla(\mathbf{v}_2 \cdot \hat{\mathbf{n}})\right] \cdot (\mathbf{r}_1 + \mathbf{r}_2) + \frac{\alpha d_f A_f}{2}(\nabla p_1 + \nabla p_2) \cdot \hat{\mathbf{s}} + \\ & \frac{A_f}{8}\left[\nabla\nabla(\mathbf{v}_1 \cdot \hat{\mathbf{n}}) + \nabla\nabla(\mathbf{v}_2 \cdot \hat{\mathbf{n}})\right] : (\mathbf{r}_1 \otimes \mathbf{r}_2 + \mathbf{r}_2 \otimes \mathbf{r}_1 + \frac{1}{6}\mathbf{t} \otimes \mathbf{t}) - \\ & \frac{\alpha d_f A_f}{4|\mathbf{s}|}(\nabla\nabla p_1 + \nabla\nabla p_2) : (\mathbf{r}_1 \otimes \mathbf{r}_2 - \mathbf{r}_2 \otimes \mathbf{r}_1) \end{aligned} \tag{C.10}$$

Subtracting Equation (A.9), which represents the second-order discretization of the volumetric flow rate, from the above equation results in an estimation of the face volumetric flow error.

$$\begin{aligned} \Delta J_{\text{vol}} = & \frac{A_f}{8}\left[\nabla\nabla(\mathbf{v}_1 \cdot \hat{\mathbf{n}}) + \nabla\nabla(\mathbf{v}_2 \cdot \hat{\mathbf{n}})\right] : (\mathbf{r}_1 \otimes \mathbf{r}_2 + \mathbf{r}_2 \otimes \mathbf{r}_1 + \frac{1}{6}\mathbf{t} \otimes \mathbf{t}) - \\ & \frac{\alpha d_f A_f}{4|\mathbf{s}|}(\nabla\nabla p_1 + \nabla\nabla p_2) : (\mathbf{r}_1 \otimes \mathbf{r}_2 - \mathbf{r}_2 \otimes \mathbf{r}_1) \end{aligned} \tag{C.11}$$

As seen, the evaluation of the face flow error involves reconstructing the second derivatives, called the Hessian tensor, of velocity and pressure. The procedure for reconstructing the solution Hessian is presented in Appendix F.

C.2 Error in Discretization of Momentum Flow

The integral form of the momentum equation for a control volume is expressed in Equation (A.14),

$$\sum_{\text{faces}} \left(\int_{\text{face}} (\rho \mathbf{v}\mathbf{v}) \cdot \hat{\mathbf{n}}\, dA + \int_{\text{face}} p\hat{\mathbf{n}}\, dA - \int_{\text{face}} \mu(\nabla \mathbf{v} + \nabla \mathbf{v}^T) \cdot \hat{\mathbf{n}}\, dA \right) = 0$$

where the integrals are evaluated on the faces of a control volume. In this section, we derive the third-order discretized from of the above equation. Then we compare the result to the second-order discretized form, Equations (A.16) and (A.18), and then determine the flow error and the residual. Note that the above equation is a vector equation with two components in x and y directions. Therefore we have to discretize each component separately.

C.2.1 The x-Component of Momentum Flow

The x component of the momentum flow across an interior face, $J^{\text{int}}_{x-\text{mom}}$, can be written in the following form:

$$J_{x-\text{mom}} = \underbrace{\int_{\text{face}} (\rho \tilde{V}_n) u \, dA}_{J_{\text{adv}}} + \underbrace{\int_{\text{face}} p n_x \, dA}_{F_{\text{pres}}} - \underbrace{\int_{\text{face}} \mu \left(\nabla u + \frac{\partial \mathbf{v}}{\partial x} \right) \cdot \hat{\mathbf{n}} \, dA}_{F_{\text{visc}}} \tag{C.12}$$

where $n_x = \hat{\mathbf{n}} \cdot \hat{\mathbf{i}}$. In the above equation J_{adv} is the advected momentum flow, F_{pres} is the pressure force, and F_{visc} is the viscous force on the face. Let us discretize these three term separately.

Advection Term in x-Momentum

The formula for the advection term in x-Momentum is:

$$J_{\text{adv}} = \int_{\text{face}} \rho \tilde{V}_n u \, dA \tag{C.13}$$

where $\tilde{V}_n$ is the face advecting velocity, Equation (C.8), and u is the face advected velocity extrapolated from the node of the upwind control volume.

$$u = u_{\text{up}} + \nabla u_{\text{up}} \cdot \mathbf{r'}_{\text{up}} + \frac{1}{2} \nabla \nabla u_{\text{up}} : (\mathbf{r'}_{\text{up}} \otimes \mathbf{r'}_{\text{up}})$$

Substituting the latter and Equation (C.8) into the integral of Equation (C.13) and evaluating the integral results in:

$$\begin{aligned} J_{\text{adv}} = \rho J_{\text{vol}} \left[u_{\text{up}} + \nabla u_{\text{up}} \cdot \mathbf{r}_{\text{up}} + \frac{1}{2} \nabla \nabla u_{\text{up}} : \left(\mathbf{r}_{\text{up}} \otimes \mathbf{r}_{\text{up}} + \frac{1}{12} \mathbf{t} \otimes \mathbf{t} \right) \right] + \\ \frac{\rho A_f}{24} \left[(\nabla \mathbf{v}_1 + \nabla \mathbf{v}_2) : (\hat{\mathbf{n}} \otimes \mathbf{t}) \right] (\nabla u_{\text{up}} \cdot \mathbf{t}) \end{aligned} \tag{C.14}$$

Pressure Force Term in x-Momentum

The pressure force in the x-momentum equation is:

$$F_{\text{pres}} = \int_{\text{face}} p n_x \, dA \tag{C.15}$$

where p is the pressure at the face, given by:

$$p = \frac{1}{2}(p_1 + p_2) + \frac{1}{4}(\nabla p_1 + \nabla p_2) \cdot (\mathbf{r'}_1 + \mathbf{r'}_2) + \frac{1}{8}(\nabla\nabla p_1 + \nabla\nabla p_2) : (\mathbf{r'}_1 \otimes \mathbf{r'}_2 + \mathbf{r'}_2 \otimes \mathbf{r'}_1)$$

Substituting the latter into Equation (C.15) and evaluating the integral yields:

$$F_{\text{pres}} = n_x A_f \left[\frac{1}{2}(p_1 + p_2) + \frac{1}{4}(\nabla p_1 + \nabla p_2) \cdot (\mathbf{r}_1 + \mathbf{r}_2) + \frac{1}{8}(\nabla\nabla p_1 + \nabla\nabla p_2) : \left(\mathbf{r}_1 \otimes \mathbf{r}_2 + \mathbf{r}_2 \otimes \mathbf{r}_1 + \frac{1}{6}\mathbf{t} \otimes \mathbf{t}\right)\right] \tag{C.16}$$

Viscous Force Term in x-Momentum

The viscous force term in the x-momentum equation is:

$$\begin{aligned} F_{\text{visc}} &= \int_{\text{face}} \mu \left(\nabla u + \frac{\partial \mathbf{v}}{\partial x}\right) \cdot \hat{\mathbf{n}} dA \\ &= \int_{\text{face}} \mu (\nabla u \cdot \hat{\mathbf{n}}) dA + \int_{\text{face}} \mu \left(\frac{\partial \mathbf{v}}{\partial x} \cdot \hat{\mathbf{n}}\right) dA \end{aligned} \tag{C.17}$$

The discretized form of the first term in the above equation is:

$$\nabla u \cdot \hat{\mathbf{n}} = \alpha \left(\frac{u_2 - u_1}{|\mathbf{s}|}\right) + \frac{1}{2}(\nabla u_1 + \nabla u_2)^k \cdot (\hat{\mathbf{n}} - \alpha\hat{\mathbf{s}}) + \frac{1}{4}(\nabla\nabla u_1 + \nabla\nabla u_2)^k : \left[\frac{\alpha}{|\mathbf{s}|}(\mathbf{r}'_1 \otimes \mathbf{r}'_2 - \mathbf{r}'_2 \otimes \mathbf{r}'_1) + (\mathbf{r}'_1 + \mathbf{r}'_2) \otimes \hat{\mathbf{n}}\right]$$

which results in:

$$\begin{aligned} \int_{\text{face}} \mu(\nabla u \cdot \hat{\mathbf{n}})\, dA = \; \mu A_f \Bigg\{ & \alpha \left(\frac{u_2 - u_1}{|\mathbf{s}|}\right) + \frac{1}{2}(\nabla u_1 + \nabla u_2) \cdot (\hat{\mathbf{n}} - \alpha\hat{\mathbf{s}}) + \\ & \frac{\alpha}{4|\mathbf{s}|}(\nabla\nabla u_1 + \nabla\nabla u_2) : (\mathbf{r}_1 \otimes \mathbf{r}_2 - \mathbf{r}_2 \otimes \mathbf{r}_1) + \\ & \frac{1}{4}(\nabla\nabla u_1 + \nabla\nabla u_2) : [(\mathbf{r}_1 + \mathbf{r}_2) \otimes \hat{\mathbf{n}})] \Bigg\} \end{aligned} \tag{C.18}$$

The discretized form of the second term in Equation (C.17) is:

$$\frac{\partial \mathbf{v}}{\partial x} = \left[\frac{1}{2}(\nabla \mathbf{v}_1 + \nabla \mathbf{v}_2) + \frac{1}{4}(\nabla\nabla \mathbf{v}_1 + \nabla\nabla \mathbf{v}_2) \cdot (\mathbf{r'}_1 + \mathbf{r'}_2)\right] \cdot \hat{\mathbf{i}}$$

which results in:

$$\int_{\text{face}} \mu \left(\frac{\partial \mathbf{v}}{\partial x} \cdot \hat{\mathbf{n}} \right) dA = \frac{1}{2}(\nabla \mathbf{v}_1 + \nabla \mathbf{v}_2) : (\hat{\mathbf{n}} \otimes \hat{\mathbf{i}}) + \left[\frac{1}{4}(\nabla\nabla \mathbf{v}_1 + \nabla\nabla \mathbf{v}_2) \cdot (\mathbf{r}_1 + \mathbf{r}_2) \right] : (\hat{\mathbf{n}} \otimes \hat{\mathbf{i}}) \tag{C.19}$$

Therefore the total viscous force becomes,

$$\begin{aligned} F_{\text{visc}} = \mu A_f \Bigg\{ & \alpha \left(\frac{u_2 - u_1}{|\mathbf{s}|} \right) + \frac{1}{2}(\nabla u_1 + \nabla u_2) \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) + \\ & \frac{\alpha}{4|\mathbf{s}|}(\nabla\nabla u_1 + \nabla\nabla u_2) : (\mathbf{r}_1 \otimes \mathbf{r}_2 - \mathbf{r}_2 \otimes \mathbf{r}_1) + \\ & \frac{1}{4}(\nabla\nabla u_1 + \nabla\nabla u_2) : [(\mathbf{r}_1 + \mathbf{r}_2) \otimes \hat{\mathbf{n}})] + \\ & \frac{1}{2}(\nabla \mathbf{v}_1 + \nabla \mathbf{v}_2) : (\hat{\mathbf{n}} \otimes \hat{\mathbf{i}}) + \\ & \left[\frac{1}{4}(\nabla\nabla \mathbf{v}_1 + \nabla\nabla \mathbf{v}_2) \cdot (\mathbf{r}_1 + \mathbf{r}_2) \right] : (\hat{\mathbf{n}} \otimes \hat{\mathbf{i}}) \Bigg\} \end{aligned} \tag{C.20}$$

Substituting Equations (C.14), (C.16), and (C.20) into Equation (C.12) results in the discretized form of the x component of the momentum flow across an interior face, $J^{\text{int}}_{x-\text{mom}}$.

$$
\begin{aligned}
J^{\text{int}}_{x-\text{mom}} = \rho J_{\text{vol}} \Bigg[& u_{\text{up}} + \nabla u_{\text{up}} \cdot \mathbf{r}_{\text{up}} + \frac{1}{2} \nabla\nabla u_{\text{up}} : \left(\mathbf{r}_{\text{up}} \otimes \mathbf{r}_{\text{up}} + \frac{1}{12} \mathbf{t} \otimes \mathbf{t} \right) \Bigg] + \\
& \frac{\rho A_f}{24} \left[(\nabla \mathbf{v}_1 + \nabla \mathbf{v}_2) : (\hat{\mathbf{n}} \otimes \mathbf{t}) \right] (\nabla u_{\text{up}} \cdot \mathbf{t}) + \\
& n_x A_f \Bigg[\frac{1}{2}(p_1 + p_2) + \frac{1}{4}(\nabla p_1 + \nabla p_2) \cdot (\mathbf{r}_1 + \mathbf{r}_2) + \\
& \qquad \frac{1}{8}(\nabla\nabla p_1 + \nabla\nabla p_2) : \left(\mathbf{r}_1 \otimes \mathbf{r}_2 + \mathbf{r}_2 \otimes \mathbf{r}_1 + \frac{1}{6} \mathbf{t} \otimes \mathbf{t} \right) \Bigg] - \\
& \mu A_f \Bigg\{ \alpha \left(\frac{u_2 - u_1}{|\mathbf{s}|} \right) + \frac{1}{2}(\nabla u_1 + \nabla u_2) \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) + \\
& \qquad \frac{\alpha}{4|\mathbf{s}|}(\nabla\nabla u_1 + \nabla\nabla u_2) : (\mathbf{r}_1 \otimes \mathbf{r}_2 - \mathbf{r}_2 \otimes \mathbf{r}_1) + \\
& \qquad \frac{1}{4}(\nabla\nabla u_1 + \nabla\nabla u_2) : [(\mathbf{r}_1 + \mathbf{r}_2) \otimes \hat{\mathbf{n}})] + \\
& \qquad \frac{1}{2}(\nabla \mathbf{v}_1 + \nabla \mathbf{v}_2) : (\hat{\mathbf{n}} \otimes \hat{\mathbf{i}}) + \\
& \qquad \left[\frac{1}{4}(\nabla\nabla \mathbf{v}_1 + \nabla\nabla \mathbf{v}_2) \cdot (\mathbf{r}_1 + \mathbf{r}_2) \right] : (\hat{\mathbf{n}} \otimes \hat{\mathbf{i}}) \Bigg\}
\end{aligned}
\tag{C.21}
$$

Comparing the above third-order equation and its second-order counterpart, Equation (A.20), provides an estimate of the x-momentum flow error across an interior face.

$$\begin{aligned}\Delta J^{\text{int}}_{x-\text{mom}} = &\frac{\rho u_{\text{up}} A_f}{8}\left[\nabla\nabla(\mathbf{v}_1\cdot\hat{\mathbf{n}})+\nabla\nabla(\mathbf{v}_2\cdot\hat{\mathbf{n}})\right]:\left(\mathbf{r}_1\otimes\mathbf{r}_2+\mathbf{r}_2\otimes\mathbf{r}_1+\frac{1}{6}\mathbf{t}\otimes\mathbf{t}\right)-\\
&\frac{\alpha d_f \rho u_{\text{up}} A_f}{4|\mathbf{s}|}(\nabla\nabla p_1+\nabla\nabla p_2):(\mathbf{r}_1\otimes\mathbf{r}_2-\mathbf{r}_2\otimes\mathbf{r}_1)+\\
&\frac{\rho A_f}{4}(\mathbf{v}_1\cdot\hat{\mathbf{n}}+\mathbf{v}_2\cdot\hat{\mathbf{n}})\left[\nabla\nabla u_{\text{up}}:\left(\mathbf{r}_{\text{up}}\otimes\mathbf{r}_{\text{up}}+\frac{1}{12}\mathbf{t}\otimes\mathbf{t}\right)\right]+\\
&\frac{\rho A_f}{24}\left[(\nabla\mathbf{v}_1+\nabla\mathbf{v}_2):(\hat{\mathbf{n}}\otimes\mathbf{t})\right](\nabla u_{\text{up}}\cdot\mathbf{t})+\\
&\frac{n_x A_f}{8}(\nabla\nabla p_1+\nabla\nabla p_2):\left(\mathbf{r}_1\otimes\mathbf{r}_2+\mathbf{r}_2\otimes\mathbf{r}_1+\frac{1}{6}\mathbf{t}\otimes\mathbf{t}\right)-\\
&\mu A_f\left\{\frac{\alpha}{4|\mathbf{s}|}(\nabla\nabla u_1+\nabla\nabla u_2):(\mathbf{r}_1\otimes\mathbf{r}_2-\mathbf{r}_2\otimes\mathbf{r}_1)+\right.\\
&\quad\frac{1}{4}(\nabla\nabla u_1+\nabla\nabla u_2):[(\mathbf{r}_1+\mathbf{r}_2)\otimes\hat{\mathbf{n}})]+\\
&\quad\left.\left[\frac{1}{4}(\nabla\nabla\mathbf{v}_1+\nabla\nabla\mathbf{v}_2)\cdot(\mathbf{r}_1+\mathbf{r}_2)\right]:(\hat{\mathbf{n}}\otimes\hat{\mathbf{i}})\right\}\end{aligned} \tag{C.22}$$

C.2.2 The y-Component of Momentum Flow

The y component of the momentum flow across an interior face, $J^{\text{int}}_{y-\text{mom}}$, is:

$$J_{y-\text{mom}} = \underbrace{\int_{\text{face}}(\rho\tilde{V}_n)v\,dA}_{J_{\text{adv}}} + \underbrace{\int_{\text{face}} pn_y\,dA}_{F_{\text{pres}}} - \underbrace{\int_{\text{face}}\mu\left(\nabla v+\frac{\partial\mathbf{v}}{\partial y}\right)\cdot\hat{\mathbf{n}}\,dA}_{F_{\text{visc}}} \tag{C.23}$$

where J_{adv} is the momentum advection across the face, F_{pres} is the pressure force on the face, and F_{visc} is the viscous force on the face. In the following paragraphs we discretize these terms.

Advection Term in y-Momentum

The formula for the advection term in the y-momentum flow is:

$$J_{\text{adv}} = \int_{\text{face}}\rho\tilde{V}_n v\,dA \tag{C.24}$$

where $\tilde{V}_n$ is the face advecting velocity, given in Equation (C.8), and u is the face advected velocity extrapolated from the upwind control volume.

$$v = v_{\text{up}} + \nabla v_{\text{up}} \cdot \mathbf{r'}_{\text{up}} + \frac{1}{2}\nabla\nabla v_{\text{up}} : (\mathbf{r'}_{\text{up}} \otimes \mathbf{r'}_{\text{up}})$$

Substituting the latter and Equation (C.8) into the integral of Equation (C.24) and evaluating the integral yields:

$$\begin{aligned} J_{\text{adv}} = \rho J_{\text{vol}} \left[v_{\text{up}} + \nabla v_{\text{up}} \cdot \mathbf{r}_{\text{up}} + \frac{1}{2}\nabla\nabla v_{\text{up}} : \left(\mathbf{r}_{\text{up}} \otimes \mathbf{r}_{\text{up}} + \frac{1}{12}\mathbf{t} \otimes \mathbf{t} \right) \right] + \\ \frac{\rho A_f}{24} \left[(\nabla \mathbf{v}_1 + \nabla \mathbf{v}_2) : (\hat{\mathbf{n}} \otimes \hat{\mathbf{t}}) \right] (\nabla v_{\text{up}} \cdot \mathbf{t}) \end{aligned} \tag{C.25}$$

Pressure Force Term in y-Momentum

The pressure force in the y-momentum equation is:

$$F_{\text{pres}} = \int_{\text{face}} p n_y \, dA \tag{C.26}$$

where p is the pressure at the face, given by:

$$\begin{aligned} p = \frac{1}{2}(p_1 + p_2) + \frac{1}{4}(\nabla p_1 + \nabla p_2) \cdot (\mathbf{r'}_1 + \mathbf{r'}_2) + \\ \frac{1}{8}(\nabla\nabla p_1 + \nabla\nabla p_2) : (\mathbf{r'}_1 \otimes \mathbf{r'}_2 + \mathbf{r'}_2 \otimes \mathbf{r'}_1) \end{aligned}$$

Substituting the latter into Equation (C.26) and evaluating the integral yields:

$$\begin{aligned} F_{\text{pres}} = n_y A_f \left[\frac{1}{2}(p_1 + p_2) + \frac{1}{4}(\nabla p_1 + \nabla p_2) \cdot (\mathbf{r}_1 + \mathbf{r}_2) + \right. \\ \left. \frac{1}{8}(\nabla\nabla p_1 + \nabla\nabla p_2) : \left(\mathbf{r}_1 \otimes \mathbf{r}_2 + \mathbf{r}_2 \otimes \mathbf{r}_1 + \frac{1}{6}\mathbf{t} \otimes \mathbf{t} \right) \right] \end{aligned} \tag{C.27}$$

Viscous Force Term in y-Momentum

The viscous force term in the y-momentum equation is:

$$\begin{aligned} F_{\text{visc}} &= \int_{\text{face}} \mu \left(\nabla v + \frac{\partial \mathbf{v}}{\partial y} \right) \cdot \hat{\mathbf{n}} dA \\ &= \int_{\text{face}} \mu (\nabla v \cdot \hat{\mathbf{n}}) dA + \int_{\text{face}} \mu \left(\frac{\partial \mathbf{v}}{\partial y} \cdot \hat{\mathbf{n}} \right) dA \end{aligned} \tag{C.28}$$

The discretized form of the first term of the above equation is:

$$\nabla v \cdot \hat{\mathbf{n}} = \alpha \left(\frac{v_2 - v_1}{|\mathbf{s}|} \right) + \frac{1}{2}(\nabla v_1 + \nabla v_2)^k \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) + \frac{1}{4}(\nabla\nabla v_1 + \nabla\nabla v_2)^k : \left[\frac{\alpha}{|\mathbf{s}|}(\mathbf{r}'_1 \otimes \mathbf{r}'_2 - \mathbf{r}'_2 \otimes \mathbf{r}'_1) + (\mathbf{r}'_1 + \mathbf{r}'_2) \otimes \hat{\mathbf{n}} \right]$$

Therefore,

$$\begin{aligned} \int_{\text{face}} \mu(\nabla v \cdot \hat{\mathbf{n}})\, dA = \ \mu A_f \Bigg\{ & \alpha \left(\frac{v_2 - v_1}{|\mathbf{s}|} \right) + \frac{1}{2}(\nabla v_1 + \nabla v_2) \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) + \\ & \frac{\alpha}{4|\mathbf{s}|}(\nabla\nabla v_1 + \nabla\nabla v_2) : (\mathbf{r}_1 \otimes \mathbf{r}_2 - \mathbf{r}_2 \otimes \mathbf{r}_1) + \\ & \frac{1}{4}(\nabla\nabla v_1 + \nabla\nabla v_2) : [(\mathbf{r}_1 + \mathbf{r}_2) \otimes \hat{\mathbf{n}})] \Bigg\} \end{aligned} \tag{C.29}$$

The discretized form of the second term in Equation (C.28) is:

$$\frac{\partial \mathbf{v}}{\partial y} = \left[\frac{1}{2}(\nabla \mathbf{v}_1 + \nabla \mathbf{v}_2) + \frac{1}{4}(\nabla\nabla \mathbf{v}_1 + \nabla\nabla \mathbf{v}_2) \cdot (\mathbf{r'}_1 + \mathbf{r'}_2) \right] \cdot \hat{\mathbf{j}}$$

Therefore,

$$\begin{aligned} \int_{\text{face}} \mu \left(\frac{\partial \mathbf{v}}{\partial y} \cdot \hat{\mathbf{n}} \right) dA = & \frac{1}{2}(\nabla \mathbf{v}_1 + \nabla \mathbf{v}_2) : (\hat{\mathbf{n}} \otimes \hat{\mathbf{j}}) + \\ & \left[\frac{1}{4}(\nabla\nabla \mathbf{v}_1 + \nabla\nabla \mathbf{v}_2) \cdot (\mathbf{r}_1 + \mathbf{r}_2) \right] : (\hat{\mathbf{n}} \otimes \hat{\mathbf{j}}) \end{aligned} \tag{C.30}$$

And the viscous force becomes,

$$\begin{aligned} F_{\text{visc}} = \mu A_f \Bigg\{ & \alpha \left(\frac{v_2 - v_1}{|\mathbf{s}|} \right) + \frac{1}{2}(\nabla v_1 + \nabla v_2) \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) + \\ & \frac{\alpha}{4|\mathbf{s}|}(\nabla\nabla v_1 + \nabla\nabla v_2) : (\mathbf{r}_1 \otimes \mathbf{r}_2 - \mathbf{r}_2 \otimes \mathbf{r}_1) + \\ & \frac{1}{4}(\nabla\nabla v_1 + \nabla\nabla v_2) : [(\mathbf{r}_1 + \mathbf{r}_2) \otimes \hat{\mathbf{n}})] + \\ & \frac{1}{2}(\nabla \mathbf{v}_1 + \nabla \mathbf{v}_2) : (\hat{\mathbf{n}} \otimes \hat{\mathbf{j}}) + \\ & \left[\frac{1}{4}(\nabla\nabla \mathbf{v}_1 + \nabla\nabla \mathbf{v}_2) \cdot (\mathbf{r}_1 + \mathbf{r}_2) \right] : (\hat{\mathbf{n}} \otimes \hat{\mathbf{j}}) \Bigg\} \end{aligned} \tag{C.31}$$

Substituting Equations (C.25), (C.27), and (C.31) into Equation (C.23) results in the y component of the momentum flow across a face, $J_{y-\text{mom}}$.

$$
\begin{aligned}
J_{y-\text{mom}} = \rho J_{\text{vol}} \Bigg[& v_{\text{up}} + \nabla v_{\text{up}} \cdot \mathbf{r}_{\text{up}} + \frac{1}{2} \nabla\nabla v_{\text{up}} : \left(\mathbf{r}_{\text{up}} \otimes \mathbf{r}_{\text{up}} + \frac{1}{12} \mathbf{t} \otimes \mathbf{t} \right) \Bigg] + \\
& \frac{\rho A_f}{24} \left[(\nabla \mathbf{v}_1 + \nabla \mathbf{v}_2) : (\hat{\mathbf{n}} \otimes \hat{\mathbf{t}}) \right] (\nabla v_{\text{up}} \cdot \mathbf{t}) + \\
& n_y A_f \Bigg[\frac{1}{2}(p_1 + p_2) + \frac{1}{4}(\nabla p_1 + \nabla p_2) \cdot (\mathbf{r}_1 + \mathbf{r}_2) + \\
& \quad \frac{1}{8}(\nabla\nabla p_1 + \nabla\nabla p_2) : \left(\mathbf{r}_1 \otimes \mathbf{r}_2 + \mathbf{r}_2 \otimes \mathbf{r}_1 + \frac{1}{6} \mathbf{t} \otimes \mathbf{t} \right) \Bigg] - \\
& \mu A_f \Bigg\{ \alpha \left(\frac{v_2 - v_1}{|\mathbf{s}|} \right) + \frac{1}{2}(\nabla v_1 + \nabla v_2) \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) + \\
& \quad \frac{\alpha}{4|\mathbf{s}|}(\nabla\nabla v_1 + \nabla\nabla v_2) : (\mathbf{r}_1 \otimes \mathbf{r}_2 - \mathbf{r}_2 \otimes \mathbf{r}_1) + \\
& \quad \frac{1}{4}(\nabla\nabla v_1 + \nabla\nabla v_2) : [(\mathbf{r}_1 + \mathbf{r}_2) \otimes \hat{\mathbf{n}})] + \\
& \quad \frac{1}{2}(\nabla \mathbf{v}_1 + \nabla \mathbf{v}_2) : (\hat{\mathbf{n}} \otimes \hat{\mathbf{j}}) + \\
& \quad \left[\frac{1}{4}(\nabla\nabla \mathbf{v}_1 + \nabla\nabla \mathbf{v}_2) \cdot (\mathbf{r}_1 + \mathbf{r}_2) \right] : (\hat{\mathbf{n}} \otimes \hat{\mathbf{j}}) \Bigg\}
\end{aligned}
\tag{C.32}
$$

Comparing the above equation, which is third-order accurate, and Equation (A.22), which is second-order accurate, we can estimate the error of the y-momentum flow across a face.

$$\begin{aligned}
\Delta J_{y-\text{mom}} = & \frac{\rho v_{\text{up}} A_f}{8} \left[\nabla\nabla(\mathbf{v}_1 \cdot \hat{\mathbf{n}}) + \nabla\nabla(\mathbf{v}_2 \cdot \hat{\mathbf{n}})\right] : (\mathbf{r}_1 \otimes \mathbf{r}_2 + \mathbf{r}_2 \otimes \mathbf{r}_1 + \frac{1}{6}\mathbf{t} \otimes \mathbf{t}) - \\
& \frac{\alpha d_f \rho v_{\text{up}} A_f}{4|\mathbf{s}|} (\nabla\nabla p_1 + \nabla\nabla p_2) : (\mathbf{r}_1 \otimes \mathbf{r}_2 - \mathbf{r}_2 \otimes \mathbf{r}_1) + \\
& \frac{\rho A_f}{4} (\mathbf{v}_1 \cdot \hat{\mathbf{n}} + \mathbf{v}_2 \cdot \hat{\mathbf{n}}) \left[\nabla\nabla v_{\text{up}} : \left(\mathbf{r}_{\text{up}} \otimes \mathbf{r}_{\text{up}} + \frac{1}{12}\mathbf{t} \otimes \mathbf{t}\right)\right] + \\
& \frac{\rho A_f}{24} \left[(\nabla\mathbf{v}_1 + \nabla\mathbf{v}_2) : (\hat{\mathbf{n}} \otimes \hat{\mathbf{t}})\right] (\nabla v_{\text{up}} \cdot \mathbf{t}) + \\
& \frac{n_y A_f}{8} (\nabla\nabla p_1 + \nabla\nabla p_2) : \left(\mathbf{r}_1 \otimes \mathbf{r}_2 + \mathbf{r}_2 \otimes \mathbf{r}_1 + \frac{1}{6}\mathbf{t} \otimes \mathbf{t}\right) - \\
& \mu A_f \Bigg\{ \frac{\alpha}{4|\mathbf{s}|} (\nabla\nabla v_1 + \nabla\nabla v_2) : (\mathbf{r}_1 \otimes \mathbf{r}_2 - \mathbf{r}_2 \otimes \mathbf{r}_1) + \\
& \quad \frac{1}{4} (\nabla\nabla v_1 + \nabla\nabla v_2) : [(\mathbf{r}_1 + \mathbf{r}_2) \otimes \hat{\mathbf{n}})] + \\
& \quad \left[\frac{1}{4} (\nabla\nabla \mathbf{v}_1 + \nabla\nabla \mathbf{v}_2) \cdot (\mathbf{r}_1 + \mathbf{r}_2)\right] : (\hat{\mathbf{n}} \otimes \hat{\mathbf{j}}) \Bigg\}
\end{aligned} \tag{C.33}$$

Appendix D

Analysis of Face Interpolation and Extrapolation

In this work, we use cell-centred finite volume method for discretizing the set of conservation equations. In the finite volume method, we try to balance the net flow across the faces of each control volume. However, we only store the variables at the nodes of the control volumes. Therefore we have to estimate the face values of variables based on their node values. In this appendix, we derive the second- and third-order interpolation formulae for this purpose.

D.1 Interpolation to an Interior Face

The first case that we examine is the interpolation to an interior face. Figure D.1 shows the schematic of such a case. The points 1 and 2 are the nodes of the neighbouring control volumes. The objective of this appendix is to determine the value of ϕ and its derivatives at some arbitrary point, f, at the face based on its values and derivatives at the nodes 1 and 2.

Let us expand the value of ϕ at the nodes 1 and 2 about the point f using third order

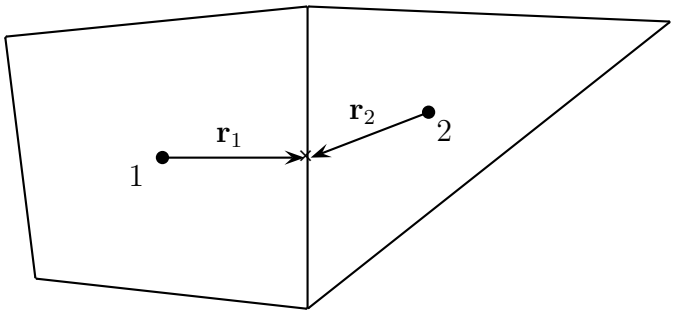


Figure D.1: Schematic of an interior face and its neighbouring control volumes

accurate Taylor series.

$$\phi_1 = \phi_f - \nabla\phi_f \cdot \mathbf{r}_1 + \frac{1}{2}(\nabla\nabla\phi_f \cdot \mathbf{r}_1) \cdot \mathbf{r}_1 + \ldots \tag{D.1}$$

$$\phi_2 = \phi_f - \nabla\phi_f \cdot \mathbf{r}_2 + \frac{1}{2}(\nabla\nabla\phi_f \cdot \mathbf{r}_2) \cdot \mathbf{r}_2 + \ldots \tag{D.2}$$

Adding Equations (D.1) and (D.2) and rearranging the result yields,

$$\phi_f \approx \frac{1}{2}(\phi_1 + \phi_2) + \frac{1}{2}\nabla\phi_f \cdot (\mathbf{r}_1 + \mathbf{r}_2) - \frac{1}{4}\nabla\nabla\phi_f : (\mathbf{r}_1 \otimes \mathbf{r}_1 + \mathbf{r}_2 \otimes \mathbf{r}_2) \tag{D.3}$$

Now, let us take the gradient and Hessian of the latter.

$$\begin{aligned} \nabla\phi_f \approx\ & \frac{1}{2}(\nabla\phi_1 + \nabla\phi_2) + \frac{1}{2}\nabla\nabla\phi_f \cdot (\mathbf{r}_1 + \mathbf{r}_2) - \\ & \nabla\left[\frac{1}{4}\nabla\nabla\phi_f : (\mathbf{r}_1 \otimes \mathbf{r}_1 + \mathbf{r}_2 \otimes \mathbf{r}_2)\right] \end{aligned} \tag{D.4}$$

$$\begin{aligned} \nabla\nabla\phi_f \approx\ & \frac{1}{2}(\nabla\nabla\phi_1 + \nabla\nabla\phi_2) + \nabla\left[\frac{1}{2}\nabla\nabla\phi_f \cdot (\mathbf{r}_1 + \mathbf{r}_2)\right] - \\ & \nabla\nabla\left[\frac{1}{4}\nabla\nabla\phi_f : (\mathbf{r}_1 \otimes \mathbf{r}_1 + \mathbf{r}_2 \otimes \mathbf{r}_2)\right] \end{aligned} \tag{D.5}$$

Substituting Equation (D.4) into Equation (D.3) and neglecting higher-order terms, results in:

$$\begin{aligned} \phi_f \approx\ & \frac{1}{2}(\phi_1 + \phi_2) + \frac{1}{2}\left[\frac{1}{2}(\nabla\phi_1 + \nabla\phi_2) + \frac{1}{2}\nabla\nabla\phi_f \cdot (\mathbf{r}_1 + \mathbf{r}_2)\right] \cdot (\mathbf{r}_1 + \mathbf{r}_2) - \\ & \frac{1}{4}\nabla\nabla\phi_f : (\mathbf{r}_1 \otimes \mathbf{r}_1 + \mathbf{r}_2 \otimes \mathbf{r}_2) \end{aligned} \tag{D.6}$$

This relation can be simplified even further by substituting Equation (D.5) in it.

$$\phi_f \approx \frac{1}{2}(\phi_1+\phi_2)+\frac{1}{4}(\nabla\phi_1+\nabla\phi_2)\cdot(\mathbf{r}_1+\mathbf{r}_2)+\frac{1}{8}(\nabla\nabla\phi_1+\nabla\nabla\phi_2):(\mathbf{r}_1\otimes\mathbf{r}_2+\mathbf{r}_2\otimes\mathbf{r}_1) \quad \text{(D.7)}$$

The latter is the third-order accurate approximation of the value of ϕ at the face. If we only need a second-order accurate formula, we may also neglect the last term in Equation (D.7).

$$\nabla\phi_f \approx \frac{1}{2}(\nabla\phi_1+\nabla\phi_2)+\frac{1}{4}(\nabla\nabla\phi_1+\nabla\nabla\phi_2)\cdot(\mathbf{r}_1+\mathbf{r}_2) \quad \text{(D.8)}$$

D.2 Extrapolation to a Boundary Face

The second case that we examine is the extrapolation to a boundary face. Figure D.2 shows the schematic of such a face. The point 1 is the node of the neighbouring control volume and the point f is the face integration point. The objective of this section is to determine a variable, ϕ, at the point f. Note that we use this method only in the case that the value of ϕ is not prescribed by the boundary conditions; otherwise we use the boundary condition.

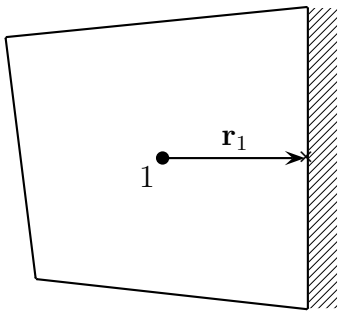


Figure D.2: Schematic of a boundary face and the neighbouring control volume

The derivation of the extrapolation formula is very similar to that of the interpolation formula for an interior face. The first step is to expand the value of ϕ at the face point, f, about the node 1 using a third order accurate Taylor series.

$$\phi_f = \phi_1 + \nabla\phi_1\cdot\mathbf{r}_1+\frac{1}{2}(\nabla\nabla\phi_1\cdot\mathbf{r}_1)\cdot\mathbf{r}_1+\ldots \quad \text{(D.9)}$$

Keeping the terms up to third order yields:

$$\phi_f \approx \phi_1 + \nabla\phi_1 \cdot \mathbf{r}_1 + \frac{1}{2}\nabla\nabla\phi_1 : (\mathbf{r}_1 \otimes \mathbf{r}_1) \tag{D.10}$$

The above formula is the third-order accurate extrapolation to the face. In the case of a second-order accurate discretization, we neglect the last term.

$$\phi_f \approx \phi_1 + \nabla\phi_1 \cdot \mathbf{r}_1 \tag{D.11}$$

Appendix E

Analysis of Face-Normal Gradient

In a second-order finite volume method, not only the values of variables but also their gradients are stored at the nodes of control volumes. The common approach for calculating these gradients is to use a gradient reconstruction method like the one explained in Appendix F. Unfortunately, gradient reconstruction methods might introduce instability in the iterative solution since the gradients are lagged and appear as source terms in the discretized equations. One remedy for this problem is to discretize the gradients based on the active terms in order to stabilize the iterative solver. In this appendix we explain how to discretize the face gradients based on the active terms at the control volume nodes.

E.1 Face-normal Gradient at an Interior Face

Appendix F explains how to calculate gradients at the node of a control volume. Having the gradients at a node, we can use Equation (D.8) determines the gradients at the face, shown in Figure E.1.

$$\nabla\phi|_{\text{face}} \approx \frac{1}{2}(\nabla\phi_1 + \nabla\phi_2) + \frac{1}{4}(\nabla\nabla\phi_1 + \nabla\nabla\phi_2)\cdot(\mathbf{r}_1 + \mathbf{r}_2) \tag{E.1}$$

Note that the above equation can only be evaluated based on the lagged solution at the nodes 1 and 2. However, we can discretize the directional derivative along the line joining the nodes 1 and 2 based on the active solution. Therefore we can resolve the gradient at the face into two components; an active component and a lagged component.

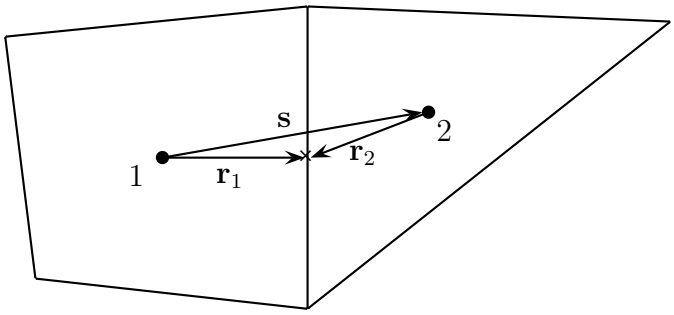


Figure E.1: Schematic of an interior face and the neighbouring control volumes

Let us expand the value of ϕ at the nodes 1 and 2 about the face integration point, f, using a third order accurate Taylor series.

$$\phi_1 = \phi_f - \nabla\phi_f \cdot \mathbf{r}_1 + \frac{1}{2}(\nabla\nabla\phi_f \cdot \mathbf{r}_1) \cdot \mathbf{r}_1 + \dots \tag{E.2}$$

$$\phi_2 = \phi_f - \nabla\phi_f \cdot \mathbf{r}_2 + \frac{1}{2}(\nabla\nabla\phi_f \cdot \mathbf{r}_2) \cdot \mathbf{r}_2 + \dots \tag{E.3}$$

Subtracting Equation (E.2) from Equation (E.3) yields the following formula:

$$\phi_2 - \phi_1 = -\nabla\phi_f \cdot \underbrace{(\mathbf{r}_2 - \mathbf{r}_1)}_{-\mathbf{s}} + \frac{1}{2}\nabla\nabla\phi_f : (\mathbf{r}_2 \otimes \mathbf{r}_2 - \mathbf{r}_1 \otimes \mathbf{r}_1) \tag{E.4}$$

Rearranging the latter results in

$$\left.\frac{\partial\phi}{\partial s}\right|^{k+1} = \left(\frac{\phi_2 - \phi_1}{|\mathbf{s}|}\right)^{k+1} - \frac{1}{4|\mathbf{s}|}(\nabla\nabla\phi_1 + \nabla\nabla\phi_2)^k : (\mathbf{r}_2 \otimes \mathbf{r}_2 - \mathbf{r}_1 \otimes \mathbf{r}_1) \tag{E.5}$$

As seen, the above formula expresses the gradient component in the $\hat{\mathbf{s}}$ direction based on the active solution. Therefore we can write the gradient at the face partially based on the active solution using the following formula.

$$\nabla\phi^{k+1} = \nabla\phi^k + \left(\left.\frac{\partial\phi}{\partial s}\right|^{k+1} - \left.\frac{\partial\phi}{\partial s}\right|^{k}\right)\hat{\mathbf{s}} \tag{E.6}$$

The dot product of the above equation by the face-normal unit vector, $\hat{\mathbf{n}}$, is:

$$\left.\frac{\partial\phi}{\partial n}\right|^{k+1} = \nabla\phi^k \cdot \hat{\mathbf{n}} + \left(\left.\frac{\partial\phi}{\partial s}\right|^{k+1} - \nabla\phi^k \cdot \hat{\mathbf{s}}\right)\underbrace{(\hat{\mathbf{s}} \cdot \hat{\mathbf{n}})}_{\alpha} \tag{E.7}$$

where $\alpha = \hat{\mathbf{s}} \cdot \hat{\mathbf{n}}$ is called the nonorthogonality parameter. Therefore,

$$\left.\frac{\partial \phi}{\partial n}\right|^{k+1} = \alpha \left.\frac{\partial \phi}{\partial s}\right|^{k+1} + \nabla \phi^k \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) \tag{E.8}$$

Substituting Equations (E.1) and (E.5) into the latter results in the final formula:

$$\begin{aligned} \left.\frac{\partial \phi}{\partial n}\right|^{k+1} = \alpha \left(\frac{\phi_2 - \phi_1}{|\mathbf{s}|}\right)^{k+1} + \frac{1}{2}(\nabla \phi_1 + \nabla \phi_2)^k \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) + \\ \frac{1}{4}(\nabla\nabla \phi_1 + \nabla\nabla \phi_2)^k : \left[\frac{\alpha}{|\mathbf{s}|}(\mathbf{r}_1 \otimes \mathbf{r}_2 - \mathbf{r}_2 \otimes \mathbf{r}_1) + (\mathbf{r}_1 + \mathbf{r}_2) \otimes \hat{\mathbf{n}}\right] \end{aligned} \tag{E.9}$$

The above formula is the third-order accurate discretization of the face-normal gradient at the face with partially active terms. If one neglects the third term on the right hand side, which involves the Hessian tensor, the second-order accurate formula is obtained.

$$\left.\frac{\partial \phi}{\partial n}\right|^{k+1} = \alpha \left(\frac{\phi_2 - \phi_1}{|\mathbf{s}|}\right)^{k+1} + \frac{1}{2}(\nabla \phi_1 + \nabla \phi_2)^k \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) \tag{E.10}$$

E.2 Face-normal Gradient at a Boundary Face

The discretization of the face-normal component of $\nabla \phi$ at a boundary face is very similar to that of an interior face, presented in the previous section. However in the case of a boundary face, Equations (E.1) and (E.5) must be modified as follows.

$$\nabla \phi \approx \nabla \phi_1 + \nabla\nabla \phi_1 \cdot \mathbf{r}_1 \tag{E.11}$$

$$\left.\frac{\partial \phi}{\partial s}\right|^{k+1} = \frac{\phi_b - \phi_1^{k+1}}{|\mathbf{s}|} - \frac{1}{2|\mathbf{s}|}\nabla\nabla \phi_1^k : (\mathbf{r}_1 \otimes \mathbf{r}_1) \tag{E.12}$$

Note that Equation (E.8) is valid for a boundary face. Therefore substituting equations (E.11) and (E.12) into Equation (E.8) results in the final discretized formula.

$$\begin{aligned} \left.\frac{\partial \phi}{\partial n}\right|^{k+1} = \alpha \left(\frac{\phi_b - \phi_1^{k+1}}{|\mathbf{s}|}\right) + \nabla \phi_1^k \cdot (\hat{\mathbf{n}} - \alpha \hat{\mathbf{s}}) + \\ \nabla\nabla \phi_1^k : \left[-\frac{\alpha}{2|\mathbf{s}|}(\mathbf{r}_1 \otimes \mathbf{r}_1) + \mathbf{r}_1 \otimes \hat{\mathbf{n}}\right] \end{aligned} \tag{E.13}$$

Note that the above formula is third-order accurate. Therefore in the case of a second-order discretization we neglect the last term on the right hand side.

Appendix F

Gradient and Hessian Reconstruction Scheme

The recovery of solution derivatives, which is usually referred to as the *solution reconstruction*, is crucial in second-order and higher-order finite volume methods. In this appendix, we describe a solution reconstruction scheme for the first and second derivatives in a discrete field. Then, we apply the scheme to the cell-centred finite volume method on an unstructured mesh.

The conventional approach for calculating gradients in the finite volume method, proposed by Barth and Jespersen [15] and Barth [14], is based on Green's Theorem or least-mean-square principle. Barth's scheme is not the only solution reconstruction scheme and Jawahar and Kamath [54] provide a comprehensive review of the other works on the subject.

The problem with the Barth's scheme is related to sharp gradients and oscillations in the vicinity of discontinuities like shock waves. Nevertheless, our emphasis in this study is on incompressible laminar flows. Therefore we may use Barth's scheme with slight modifications as proposed by Baserinia and Stubley [17]. In the following sections, we describe the scheme.

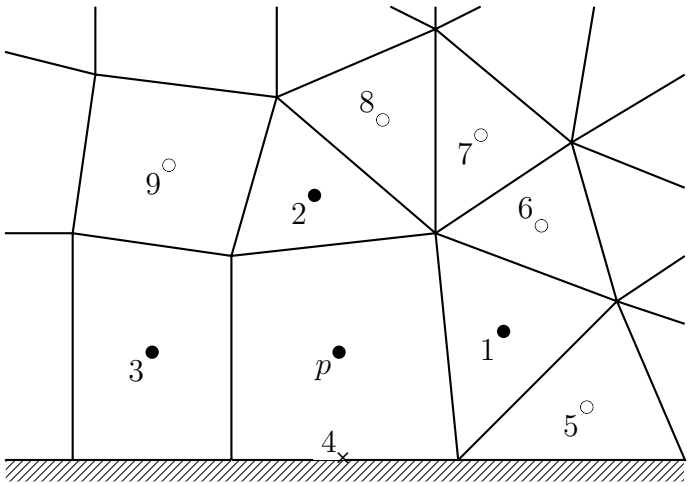


Figure F.1: Schematic of computational molecule for gradient and Hessian Reconstruction; Bold dots (•) are face neighbours, hollow dots (∘) are vertex neighbours, and cross (×) is the boundary node.

F.1 One-Pass Least-Mean-Square Approach

In this section, we show how to use the least-mean-square approach to calculate the solution derivatives. Figure F.1 indicates the schematic of a typical mixed triangular-quadrilateral mesh. Suppose the value of a scalar variable, ϕ, is given at the nodes of the control volumes and at the boundary. The objective is to find the gradient, $\nabla\phi$, and the Hessian, $\nabla\nabla\phi$, at the node p in Figure F.1.

Let us write the second order Taylor series expansion of ϕ about the node p as follows:

$$\phi(\mathbf{r}) = \phi_p + \nabla\phi_p \cdot (\mathbf{r} - \mathbf{r}_p) + [\nabla\nabla\phi_p \cdot (\mathbf{r} - \mathbf{r}_p)] \cdot (\mathbf{r} - \mathbf{r}_p) \tag{F.1}$$

Where $\mathbf{r}$ is the position vector and the subscript $[\,]_p$ denotes the value at the node p. The gradient and Hessian at the node p are unknown. However, we can find them through a least-mean-square approach. Let us define the mean-square parameter, E, as follows:

$$E \equiv \sum_{i \in \text{nbr}} [\phi(\mathbf{r}_i) - \phi_i]^2 \tag{F.2}$$

where i is the index of the neighbouring nodes, ϕ_i is the value of ϕ at the i-th node, and $\phi(\mathbf{r}_i)$ is the value ϕ at the i-th node based on Equation (F.2). The goal is to find the

gradient and Hessian so that E becomes minimized. For the sake of brevity, let us define Δx and Δy as,

$$\begin{aligned}\Delta x &\equiv (\mathbf{r}-\mathbf{r}_p)\cdot\hat{\mathbf{i}} = x - x_p \\ \Delta y &\equiv (\mathbf{r}-\mathbf{r}_p)\cdot\hat{\mathbf{j}} = y - y_p\end{aligned}$$

and expand Equation (F.1) in terms of Δx and Δy as follows:

$$\phi(x,y) = \phi_p + \phi_{x_p}\Delta x + \phi_{y_p}\Delta y + \phi_{xx_p}\Delta x^2 + 2\phi_{xy_p}\Delta x\Delta y + \phi_{yy_p}\Delta y^2 \qquad \text{(F.3)}$$

As seen, the gradient and Hessian components are the coefficients of the above polynomial. Therefore we can find them through a least square approach by using the following equations.

$$\frac{\partial E}{\phi_{x_p}} = \frac{\partial E}{\phi_{y_p}} = \frac{\partial E}{\phi_{xx_p}} = \frac{\partial E}{\phi_{xy_p}} = \frac{\partial E}{\phi_{yy_p}} = 0$$

The result is a 5×5 linear system of equations [69].

$$\begin{pmatrix} \sum\Delta x^2 & \sum\Delta x\Delta y & \frac{1}{2}\sum\Delta x^3 & \sum\Delta x^2\Delta y & \frac{1}{2}\sum\Delta x\Delta y^2 \\ \sum\Delta x\Delta y & \sum\Delta y^2 & \frac{1}{2}\sum\Delta x^2\Delta y & \sum\Delta x\Delta y^2 & \frac{1}{2}\sum\Delta y^3 \\ \sum\Delta x^3 & \sum\Delta x^2\Delta y & \frac{1}{2}\sum\Delta x^4 & \sum\Delta x^3\Delta y & \frac{1}{2}\sum\Delta x^2\Delta y^2 \\ \sum\Delta x^2\Delta y & \sum\Delta x\Delta y^2 & \frac{1}{2}\sum\Delta x^3\Delta y & \sum\Delta x^2\Delta y^2 & \frac{1}{2}\sum\Delta x\Delta y^3 \\ \sum\Delta x\Delta y^2 & \sum\Delta y^3 & \frac{1}{2}\sum\Delta x^2\Delta y^2 & \sum\Delta x\Delta y^3 & \frac{1}{2}\sum\Delta y^4 \end{pmatrix} \begin{pmatrix} \phi_{x_p} \\ \phi_{y_p} \\ \phi_{xx_p} \\ \phi_{xy_p} \\ \phi_{yy_p} \end{pmatrix} = \begin{pmatrix} \sum\Delta\phi\Delta x \\ \sum\Delta\phi\Delta y \\ \sum\Delta\phi\Delta x^2 \\ \sum\Delta\phi\Delta x\Delta y \\ \sum\Delta\phi\Delta y^2 \end{pmatrix}$$

Solving the above system of equations involves the information of at least five neighbouring points. In practice, a control volume in a triangular or quadrilateral mesh has only three or four neighbouring control volumes, respectively. Therefore one must use a bigger computational molecule and take the vertex neighbours into account, shown in Figure F.1. A bigger computational molecule in turn infers more complex data structures and more intensive computations. Therefore we may prefer to find an alternative approach.

F.2 Two-Pass Least-Mean-Square Approach

The two-pass approach is similar to the one-pass approach in the sense that we calculate the gradient and Hessian using a least-mean-square principle. The only difference is that we calculate the gradient and Hessian in a segregated way. In other words, we calculate the gradient first and then we use the gradient to calculate the Hessian. In the following subsections, we explain the approach.

F.2.1 Pass One: Gradient Reconstruction

Let us write Equation F.3 in the following form:

$$\phi(\mathbf{r}) = \phi_p + \phi_{x_p}\Delta x + \phi_{y_p}\Delta y + H(\Delta x, \Delta y) \tag{F.4}$$

where $H(\Delta x, \Delta y)$ consists of the quadratic terms in Equation (F.3).

$$H(\Delta x, \Delta y) = \phi_{xx_p}\Delta x^2 + 2\phi_{xy_p}\Delta x\Delta y + \phi_{yy_p}\Delta y^2 \tag{F.5}$$

We can find the values of $\nabla\phi_p$ using a least-mean-square fit of the above equation to the solution at the node p and its neighbouring nodes. The mean-square parameter is:

$$E_{\text{grad}} \equiv \sum_i [\phi(\mathbf{r}_i) - \phi_i - H(\mathbf{r}_i)]^2 \tag{F.6}$$

We need to minimize E_{grad} in the above relation. Therefore

$$\frac{\partial E_{\text{grad}}}{\phi_{x_p}} = \frac{\partial E_{\text{grad}}}{\phi_{y_p}} = 0$$

which results in the following system of equations:

$$\begin{pmatrix} \sum \Delta x_i^2 & \sum \Delta x_i \Delta y_i \\ \sum \Delta x_i \Delta y_i & \sum \Delta y_i^2 \end{pmatrix} \begin{pmatrix} \phi_{x_p} \\ \phi_{y_p} \end{pmatrix}^k = \begin{pmatrix} \sum(\Delta\phi_i - H_i)\Delta x_i \\ \sum(\Delta\phi_i - H_i)\Delta y_i \end{pmatrix}^k \tag{F.7}$$

where $H_i = H(\mathbf{r}_i) = H(x_i, y_i)$. In the latter, we can solve the system for ϕ_{x_p} and ϕ_{y_p}. However, the value of function H at the nodes, H_i, is unknown. In the next sub-section, we show how to calculate it.

F.2.2 Pass Two: Hessian Reconstruction

The first-order Taylor series expansion of $\nabla\phi$ about the node p is:

$$\nabla\phi(\mathbf{r}) = \nabla\phi_p + \nabla\nabla\phi_p \cdot (\mathbf{r} - \mathbf{r}_p) \tag{F.8}$$

Note that the latter is a vector equation that is equivalent to the following system:

$$\begin{aligned} \phi_x(\mathbf{r}) &= \phi_{x_p} + \phi_{xx_p}\Delta x + \phi_{xy_p}\Delta y \\ \phi_y(\mathbf{r}) &= \phi_{y_p} + \phi_{yx_p}\Delta x + \phi_{yy_p}\Delta y \end{aligned}$$

We can find the components of $\nabla\nabla\phi_p$ using a least-mean-square fit of the above equation to the solution gradients at the node p and its neighbouring nodes. The mean-square parameter is:

$$E_{\text{Hess}} \equiv \sum_i |\nabla\phi(\mathbf{r}_i) - \nabla\phi_i|^2 = \sum_i \left\{[\phi_x(\mathbf{r}_i) - \phi_{x_i}]^2 + [\phi_y(\mathbf{r}_i) - \phi_{y_i}]^2\right\} \tag{F.9}$$

We need to minimize E_{Hess} in the above relation. Therefore

$$\frac{\partial E_{\text{Hess}}}{\phi_{xx_p}} = \frac{\partial E_{\text{Hess}}}{\phi_{xy_p}} = \frac{\partial E_{\text{Hess}}}{\phi_{yy_p}} = 0$$

which results in the following system of equations:

$$\begin{pmatrix} \sum \Delta x^2 & \sum \Delta x \Delta y & 0 \\ \sum \Delta x \Delta y & \sum \Delta x^2 + \Delta y^2 & \sum \Delta x \Delta y \\ 0 & \sum \Delta x \Delta y & \sum \Delta x^2 \end{pmatrix} \begin{pmatrix} \phi_{xx_p} \\ \phi_{xy_p} \\ \phi_{yy_p} \end{pmatrix}^{k+1} = \begin{pmatrix} \sum \Delta\phi_x \Delta x \\ \sum \Delta\phi_x \Delta y + \Delta\phi_y \Delta x \\ \sum \Delta\phi_y \Delta y \end{pmatrix}^{k} \tag{F.10}$$

Let us summarize the entire algorithm in the following subsection.

F.2.3 Gradient and Hessian Reconstruction Algorithm

The entire algorithm of gradient and Hessian reconstruction is presented in Algorithm 1. The procedure starts with initializing the gradient and Hessian components to zero. Then, we use Equation (F.7) to calculate the gradient. However, at the first step the value of H is unknown and we ignore it. After calculating the Hessian tensor using Equation (F.10) we update the value of H using Equation (F.5) and substitute the result in Equation (F.7). Therefore we obtain a better estimate for the gradient. We iterate this procedure until convergence.

Algorithm 1 Gradient and Hessian Reconstruction

Require: ϕ_i at $\mathbf{r}_i$ for all control volumes and ε

Ensure: $||\nabla\phi^{\text{new}} - \nabla\phi^{\text{old}}|| < \varepsilon$

1: $\nabla\phi^{\text{new}} \leftarrow 0$
2: $\nabla\nabla\phi \leftarrow 0$
3: **repeat**
4: $\quad \nabla\phi^{\text{old}} \leftarrow \nabla\phi_p^{\text{new}}$
5: $\quad$ Calculate H using Equation (F.5)
6: $\quad$ Calculate $\nabla\phi^{\text{new}}$ using Equation (F.7)
7: $\quad$ Calculate $\nabla\nabla\phi$ using Equation (F.10)
8: **until** $||\nabla\phi^{\text{new}} - \nabla\phi^{\text{old}}|| < \varepsilon$
9: **return** $\nabla\phi$ and $\nabla\nabla\phi$

Appendix G

Residual Estimator for Isotropic Triangular Meshes

In Appendix C, we presented a general method for estimating the residual of a numerical solution. In this appendix, we present a simplified version of the same residual estimation method that is suitable for isotropic triangular meshes.

The general approach is based on the assumption that the second derivatives of physical variables are locally constant in each control volume. In addition, we assume that all of control volumes are nearly equilateral triangles. The question is how to simplify the proposed residual estimator for such a situation.

G.1 Local Face Geometry

Figure G.1 shows an example of an interior face in a uniform isotropic triangular mesh. As seen, the control volumes are equilateral triangles with the characteristic size h. We also consider the unit vectors $\hat{\mathbf{t}}$ and $\hat{\mathbf{n}}$ along the face-tangential and face-normal directions.

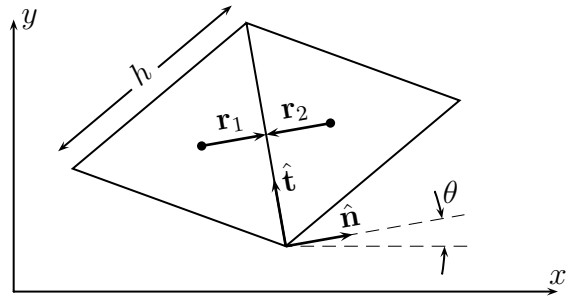


Figure G.1: Example of an interior face between two control volumes in an isotropic triangular mesh

G.2 Error in Discretization of Mass Flow

In Appendix C, we showed that the discretization of the mass flow across an interior face is based on the following formula:

$$\begin{aligned}\Delta J_{\text{vol}} =& \frac{A_f}{8}\Big[n_x(\nabla\nabla u_1 + \nabla\nabla u_2) : (\mathbf{r}_1 \otimes \mathbf{r}_2 + \mathbf{r}_2 \otimes \mathbf{r}_1 + \frac{1}{6}\mathbf{t} \otimes \mathbf{t}) - \\ & n_y(\nabla\nabla v_1 + \nabla\nabla v_2) : (\mathbf{r}_1 \otimes \mathbf{r}_2 + \mathbf{r}_2 \otimes \mathbf{r}_1 + \frac{1}{6}\mathbf{t} \otimes \mathbf{t})\Big] - \\ & \frac{\alpha d_f A_f}{4|\mathbf{s}|}(\nabla\nabla p_1 + \nabla\nabla p_2) : (\mathbf{r}_1 \otimes \mathbf{r}_2 - \mathbf{r}_2 \otimes \mathbf{r}_1)\end{aligned} \tag{G.1}$$

Let us assume that all the second derivatives, the Hessian tensors, are locally constant in the vicinity of the control volume 1, which implies:

$$\begin{aligned}\nabla\nabla p_1 &= \nabla\nabla p_2 = \nabla\nabla p \\ \nabla\nabla u_1 &= \nabla\nabla u_2 = \nabla\nabla u \\ \nabla\nabla v_1 &= \nabla\nabla v_2 = \nabla\nabla v\end{aligned} \tag{G.2}$$

In addition, a close examination of Figure G.1 indicates:

$$\begin{aligned}\mathbf{r}_1 = -\mathbf{r}_2 &= \frac{\sqrt{3}h}{6}\hat{\mathbf{n}} \\ \mathbf{t} &= h\hat{\mathbf{t}} \\ A_f &= h\end{aligned} \tag{G.3}$$

Substituting these relations back into Equation (G.1) yields:

$$\Delta J_{\text{vol}} = \frac{h^3}{24}(n_x \nabla\nabla u + n_y \nabla\nabla v) : (\hat{\mathbf{t}} \otimes \hat{\mathbf{t}} - \hat{\mathbf{n}} \otimes \hat{\mathbf{n}}) \tag{G.4}$$

For further simplification we can write $\hat{\mathbf{t}}$ and $\hat{\mathbf{n}}$ in terms of the face-normal orientation, θ.

$$\begin{aligned} \hat{\mathbf{n}} &= \cos\theta\,\hat{\mathbf{i}} + \sin\theta\,\hat{\mathbf{j}} \\ \hat{\mathbf{t}} &= -\sin\theta\,\hat{\mathbf{i}} + \cos\theta\,\hat{\mathbf{j}} \end{aligned} \tag{G.5}$$

Therefore,

$$\hat{\mathbf{t}} \otimes \hat{\mathbf{t}} - \hat{\mathbf{n}} \otimes \hat{\mathbf{n}} = \begin{pmatrix} -\cos 2\theta & -\sin 2\theta \\ -\sin 2\theta & \cos 2\theta \end{pmatrix}$$

Substituting the latter back into Equation (G.4), results in:

$$\begin{aligned} \Delta J_{\text{vol}} = \frac{h^3}{24} \Bigg[&\cos\theta \left(-\frac{\partial^2 u}{\partial x^2}\cos 2\theta - 2\frac{\partial^2 u}{\partial x \partial y}\sin 2\theta + \frac{\partial^2 u}{\partial y^2}\cos 2\theta \right) + \\ &\sin\theta \left(-\frac{\partial^2 v}{\partial x^2}\cos 2\theta - 2\frac{\partial^2 v}{\partial x \partial y}\sin 2\theta + \frac{\partial^2 v}{\partial y^2}\cos 2\theta \right) \Bigg] \end{aligned} \tag{G.6}$$

which can be written in a more convenient form:

$$\begin{aligned} \Delta J_{\text{vol}} = \frac{h^3}{24} \Bigg[&\left(-\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right)\cos^3\theta + \left(-4\frac{\partial^2 u}{\partial x \partial y} - \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right)\cos^2\theta \sin\theta + \\ &\left(\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} - 4\frac{\partial^2 v}{\partial x \partial y} \right)\cos\theta \sin^2\theta + \left(\frac{\partial^2 v}{\partial x^2} - \frac{\partial^2 v}{\partial y^2} \right)\sin^3\theta \Bigg] \end{aligned} \tag{G.7}$$

We obtained the mass flow error across an interior face as a function of θ. However, as seen in Figure G.2, the total flow error across the faces of a control volume, which we call residual, is equal to:

$$\delta_{\text{vol}} = \Delta J_{\text{vol}}(\theta) + \Delta J_{\text{vol}}(\theta + \frac{2\pi}{3}) + \Delta J_{\text{vol}}(\theta + \frac{4\pi}{3}) \tag{G.8}$$

Where δ_{vol} is the residual of the incompressible mass equation for the control volume of Figure G.2. Unfortunately, the residual is a function of the control volume orientation. However, if we take the derivative of δ_{vol} with respect to θ, we can find the orientation that maximizes the residual.

$$\frac{d\delta_{\text{vol}}}{d\theta} = 0$$

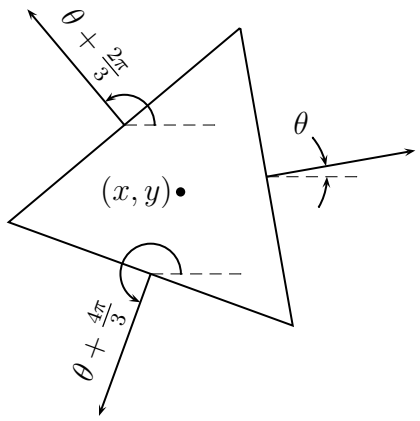


Figure G.2: Flows Across Faces of a Isotropic Triangular Control Volume

Taking the above derivative and solving the obtained equation for θ results in the following equation for the residual:

$$\delta_{\text{vol}}^{\max} = \frac{h^3}{32}\left[\left(-\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + 4\frac{\partial^2 u}{\partial x \partial y} + \frac{\partial^2 v}{\partial x^2} - \frac{\partial^2 v}{\partial y^2}\right)^2 + \left(-\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + 4\frac{\partial^2 v}{\partial x \partial y} + \frac{\partial^2 v}{\partial x^2} - \frac{\partial^2 v}{\partial y^2}\right)^2\right]^{\frac{1}{2}} \quad \text{(G.9)}$$

The latter is the residual of volumetric flow rate for an isotropic triangular control volume. Should we interested in the residual of mass flow rate, we multiply the above equation by the fluid density, ρ.

G.3 Error in Discretization of x-Momentum Flow

In Appendix C, we showed that the discretization of the x component of the momentum flow across an interior face is based on the following formula.

$$\begin{aligned}
\Delta J_{x-\text{mom}} = {} & \frac{\rho u_{\text{up}} A_f}{8}\left[\nabla\nabla(\mathbf{v}_1\cdot\hat{\mathbf{n}})+\nabla\nabla(\mathbf{v}_2\cdot\hat{\mathbf{n}})\right]:(\mathbf{r}_1\otimes\mathbf{r}_2+\mathbf{r}_2\otimes\mathbf{r}_1+\frac{1}{6}\mathbf{t}\otimes\mathbf{t})- \\
& \frac{\alpha d_f \rho u_{\text{up}} A_f}{4|\mathbf{s}|}(\nabla\nabla p_1+\nabla\nabla p_2):(\mathbf{r}_1\otimes\mathbf{r}_2-\mathbf{r}_2\otimes\mathbf{r}_1)+ \\
& \frac{\rho A_f}{4}(\mathbf{v}_1\cdot\hat{\mathbf{n}}+\mathbf{v}_2\cdot\hat{\mathbf{n}})\left[\nabla\nabla u_{\text{up}}:\left(\mathbf{r}_{\text{up}}\otimes\mathbf{r}_{\text{up}}+\frac{1}{12}\mathbf{t}\otimes\mathbf{t}\right)\right]+ \\
& \frac{\rho A_f}{24}\left[(\nabla\mathbf{v}_1+\nabla\mathbf{v}_2):(\hat{\mathbf{n}}\otimes\hat{\mathbf{t}})\right](\nabla u_{\text{up}}\cdot\mathbf{t})+ \\
& \frac{n_x A_f}{8}(\nabla\nabla p_1+\nabla\nabla p_2):\left(\mathbf{r}_1\otimes\mathbf{r}_2+\mathbf{r}_2\otimes\mathbf{r}_1+\frac{1}{6}\mathbf{t}\otimes\mathbf{t}\right)- \\
& \mu A_f\left\{\frac{\alpha}{4|\mathbf{s}|}(\nabla\nabla u_1+\nabla\nabla u_2):(\mathbf{r}_1\otimes\mathbf{r}_2-\mathbf{r}_2\otimes\mathbf{r}_1)+\right. \\
& \quad \frac{1}{4}(\nabla\nabla u_1+\nabla\nabla u_2):[(\mathbf{r}_1+\mathbf{r}_2)\otimes\hat{\mathbf{n}})]+ \\
& \quad \left.\left[\frac{1}{4}(\nabla\nabla\mathbf{v}_1+\nabla\nabla\mathbf{v}_2)\cdot(\mathbf{r}_1+\mathbf{r}_2)\right]:(\hat{\mathbf{n}}\otimes\hat{\mathbf{i}})\right\}
\end{aligned} \tag{G.10}$$

Using Equations (G.2) and (G.3) we can simplify the above equation. In addition, we can assume without loss of generality that the control volume with the subscript 1 is the upwind control volume.

$$\mathbf{r}_{\text{up}} = \mathbf{r}_1$$

$$u_{\text{up}} = u_1$$

Therefore, Equation (G.10) reduces to the following equation:

$$\begin{aligned}
\Delta J_{x-\text{mom}} = {} & \frac{h^3}{24}\rho u(n_x\nabla\nabla u+n_y\nabla\nabla v):(\hat{\mathbf{t}}\otimes\hat{\mathbf{t}}-\hat{\mathbf{n}}\otimes\hat{\mathbf{n}})+ \\
& \frac{h^3}{24}\rho(n_x u+n_y v)\nabla\nabla u:(\hat{\mathbf{t}}\otimes\hat{\mathbf{t}}+\hat{\mathbf{n}}\otimes\hat{\mathbf{n}})+ \\
& \frac{h^3}{12}\rho[(n_x\nabla u+n_y\nabla v)\cdot\hat{\mathbf{t}}](\nabla u\cdot\hat{\mathbf{t}})+ \\
& \frac{h^3}{24}n_x\nabla\nabla p:(\hat{\mathbf{t}}\otimes\hat{\mathbf{t}}-\hat{\mathbf{n}}\otimes\hat{\mathbf{n}})
\end{aligned} \tag{G.11}$$

Using the identities presented in Equations (G.5) we have:

$$\hat{\mathbf{t}} \otimes \hat{\mathbf{t}} - \hat{\mathbf{n}} \otimes \hat{\mathbf{n}} = \begin{pmatrix} -\cos 2\theta & -\sin 2\theta \\ -\sin 2\theta & \cos 2\theta \end{pmatrix} \qquad \hat{\mathbf{t}} \otimes \hat{\mathbf{t}} + \hat{\mathbf{n}} \otimes \hat{\mathbf{n}} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Substituting the above relations into Equation (G.11) results in the x-momentum flow error across the face.

$$\begin{aligned}
\Delta J_{x-\text{mom}} = \\
&\frac{h^3}{12}\Bigg\{ \left[\rho\left(u\frac{\partial^2 u}{\partial y^2} - u\frac{\partial^2 u}{\partial x^2} + \frac{\partial u}{\partial y}^2 - \frac{\partial u}{\partial x}^2 + \frac{\partial u}{\partial y}\frac{\partial v}{\partial x} + \frac{\partial u}{\partial x}\frac{\partial v}{\partial y} + 2u\frac{\partial^2 v}{\partial x \partial y}\right) + \left(\frac{\partial^2 p}{\partial y^2} - \frac{\partial^2 p}{\partial x^2}\right)\right]\cos^3\theta + \\
&\left[\rho\left(u\frac{\partial^2 v}{\partial x^2} - u\frac{\partial^2 v}{\partial y^2} + \frac{\partial u}{\partial x}\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\frac{\partial v}{\partial y} + 2u\frac{\partial^2 u}{\partial x \partial y} + 2\frac{\partial u}{\partial x}\frac{\partial u}{\partial y}\right) + 2\frac{\partial^2 p}{\partial x \partial y}\right]\sin^3\theta + \\
&\left[\rho\left(u\frac{\partial^2 u}{\partial x^2} - 2u\frac{\partial^2 v}{\partial x \partial y} - \frac{\partial u}{\partial y}\frac{\partial v}{\partial x} - \frac{\partial u}{\partial x}\frac{\partial v}{\partial y} + \frac{\partial u}{\partial x}^2\right) + \frac{1}{2}\left(\frac{\partial^2 p}{\partial x^2} - \frac{\partial^2 p}{\partial y^2}\right)\right]\cos\theta + \\
&\left[\rho\left(\frac{v}{2}\frac{\partial^2 u}{\partial x^2} + \frac{v}{2}\frac{\partial^2 u}{\partial y^2} - \frac{u}{2}\frac{\partial^2 v}{\partial x^2} + \frac{u}{2}\frac{\partial^2 v}{\partial y^2} - 2u\frac{\partial^2 u}{\partial x \partial y} + \frac{\partial u}{\partial y}\frac{\partial v}{\partial y} - 2\frac{\partial u}{\partial x}\frac{\partial u}{\partial y}\right) - 2\frac{\partial^2 p}{\partial x \partial y}\right]\sin\theta\Bigg\}
\end{aligned} \tag{G.12}$$

The above formula is the x-momentum flow error across an interior face as a function of θ. However, as discussed in the previous section, the residual is equal to:

$$\delta_{x-\text{mom}} = \Delta J_{x-\text{mom}}(\theta) + \Delta J_{x-\text{mom}}(\theta + \frac{2\pi}{3}) + \Delta J_{x-\text{mom}}(\theta + \frac{4\pi}{3})$$

Substituting Equation (G.12) into the latter and taking the derivative of the resulting equation with respect to θ results in the maximum value of the control volume residual.

$$\begin{aligned}
\Delta J^{\max}_{x-\text{mom}} = \frac{h^3}{16}\Bigg\{ &\left[\rho\left(u\frac{\partial^2 u}{\partial y^2} - u\frac{\partial^2 u}{\partial x^2} + \frac{\partial u}{\partial y}^2 - \frac{\partial u}{\partial x}^2 + \frac{\partial u}{\partial y}\frac{\partial v}{\partial x} + \frac{\partial u}{\partial x}\frac{\partial v}{\partial y} + 2u\frac{\partial^2 v}{\partial x \partial y}\right) + \left(\frac{\partial^2 p}{\partial y^2} - \frac{\partial^2 p}{\partial x^2}\right)\right]^2 + \\
&\left[\rho\left(u\frac{\partial^2 v}{\partial x^2} - u\frac{\partial^2 v}{\partial y^2} + \frac{\partial u}{\partial x}\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\frac{\partial v}{\partial y} + 2u\frac{\partial^2 u}{\partial x \partial y} + 2\frac{\partial u}{\partial x}\frac{\partial u}{\partial y}\right) + 2\frac{\partial^2 p}{\partial x \partial y}\right]^2\Bigg\}^{\frac{1}{2}}
\end{aligned} \tag{G.13}$$

G.4 Residual of y-Momentum Flow

The derivation of the residual formula for the y-Momentum equation is similar to that of the x-Momentum, presented in the previous section. We start from the discretization

error of the y component of the momentum flow across an interior face.

$$\begin{aligned}\Delta J_{y-\text{mom}} = &\frac{\rho v_{\text{up}} A_f}{8}\left[\nabla\nabla(\mathbf{v}_1\cdot\hat{\mathbf{n}})+\nabla\nabla(\mathbf{v}_2\cdot\hat{\mathbf{n}})\right]:(\mathbf{r}_1\otimes\mathbf{r}_2+\mathbf{r}_2\otimes\mathbf{r}_1+\frac{1}{6}\mathbf{t}\otimes\mathbf{t})-\\ &\frac{\alpha d_f \rho v_{\text{up}} A_f}{4|\mathbf{s}|}(\nabla\nabla p_1+\nabla\nabla p_2):(\mathbf{r}_1\otimes\mathbf{r}_2-\mathbf{r}_2\otimes\mathbf{r}_1)+\\ &\frac{\rho A_f}{4}(\mathbf{v}_1\cdot\hat{\mathbf{n}}+\mathbf{v}_2\cdot\hat{\mathbf{n}})\left[\nabla\nabla v_{\text{up}}:\left(\mathbf{r}_{\text{up}}\otimes\mathbf{r}_{\text{up}}+\frac{1}{12}\mathbf{t}\otimes\mathbf{t}\right)\right]+\\ &\frac{\rho A_f}{24}\left[(\nabla\mathbf{v}_1+\nabla\mathbf{v}_2):(\hat{\mathbf{n}}\otimes\hat{\mathbf{t}})\right](\nabla v_{\text{up}}\cdot\mathbf{t})+\\ &\frac{n_y A_f}{8}(\nabla\nabla p_1+\nabla\nabla p_2):\left(\mathbf{r}_1\otimes\mathbf{r}_2+\mathbf{r}_2\otimes\mathbf{r}_1+\frac{1}{6}\mathbf{t}\otimes\mathbf{t}\right)-\\ &\mu A_f\Bigg\{\frac{\alpha}{4|\mathbf{s}|}(\nabla\nabla v_1+\nabla\nabla v_2):(\mathbf{r}_1\otimes\mathbf{r}_2-\mathbf{r}_2\otimes\mathbf{r}_1)+\\ &\quad\frac{1}{4}(\nabla\nabla v_1+\nabla\nabla v_2):[(\mathbf{r}_1+\mathbf{r}_2)\otimes\hat{\mathbf{n}}]+\\ &\quad\left[\frac{1}{4}(\nabla\nabla\mathbf{v}_1+\nabla\nabla\mathbf{v}_2)\cdot(\mathbf{r}_1+\mathbf{r}_2)\right]:(\hat{\mathbf{n}}\otimes\hat{\mathbf{j}})\Bigg\}\end{aligned} \tag{G.14}$$

However, using Equations (G.2) and (G.3) we can reduce the above equation to the following one:

$$\begin{aligned}\Delta J_{y-\text{mom}} = &\frac{h^3}{24}\rho v(n_x\nabla\nabla u+n_y\nabla\nabla v):(\hat{\mathbf{t}}\otimes\hat{\mathbf{t}}-\hat{\mathbf{n}}\otimes\hat{\mathbf{n}})+\\ &\frac{h^3}{24}\rho(n_x u+n_y v)\nabla\nabla v:(\hat{\mathbf{t}}\otimes\hat{\mathbf{t}}+\hat{\mathbf{n}}\otimes\hat{\mathbf{n}})+\\ &\frac{h^3}{12}\rho[(n_x\nabla u+n_y\nabla v)\cdot\hat{\mathbf{t}}](\nabla v\cdot\hat{\mathbf{t}})+\\ &\frac{h^3}{24}n_y\nabla\nabla p:(\hat{\mathbf{t}}\otimes\hat{\mathbf{t}}-\hat{\mathbf{n}}\otimes\hat{\mathbf{n}})\end{aligned} \tag{G.15}$$

Expanding the above equation and writing the unit vectors $\hat{\mathbf{n}}$ and $\hat{\mathbf{t}}$ in terms of θ results in the following equation:

$$\begin{aligned}
\Delta J_{y-\text{mom}} = \\
&\frac{h^3}{12}\Bigg\{ \left[\rho\left(v\frac{\partial^2 u}{\partial y^2} - v\frac{\partial^2 u}{\partial x^2} + \frac{\partial u}{\partial y}\frac{\partial v}{\partial y} - \frac{\partial u}{\partial x}\frac{\partial v}{\partial x} + 2v\frac{\partial^2 v}{\partial x \partial y} + 2\frac{\partial v}{\partial x}\frac{\partial v}{\partial y}\right) + 2\frac{\partial^2 p}{\partial x \partial y}\right]\cos^3\theta + \\
&\left[\rho\left(v\frac{\partial^2 v}{\partial x^2} - v\frac{\partial^2 v}{\partial y^2} + \frac{\partial v}{\partial x}^2 - \frac{\partial v}{\partial y}^2 + \frac{\partial u}{\partial x}\frac{\partial v}{\partial y} + \frac{\partial v}{\partial x}\frac{\partial u}{\partial y} + 2v\frac{\partial^2 u}{\partial x \partial y}\right) + \left(\frac{\partial^2 p}{\partial x^2} - \frac{\partial^2 p}{\partial y^2}\right)\right]\sin^3\theta + \\
&\left[\rho\left(\frac{v}{2}\frac{\partial^2 u}{\partial x^2} - \frac{v}{2}\frac{\partial^2 u}{\partial y^2} + \frac{u}{2}\frac{\partial^2 v}{\partial x^2} + \frac{u}{2}\frac{\partial^2 v}{\partial y^2} - 2v\frac{\partial^2 v}{\partial x \partial y} + \frac{\partial u}{\partial x}\frac{\partial v}{\partial x} - 2\frac{\partial v}{\partial x}\frac{\partial v}{\partial y}\right) - 2\frac{\partial^2 p}{\partial x \partial y}\right]\cos\theta + \\
&\left[\rho\left(v\frac{\partial^2 v}{\partial y^2} - 2v\frac{\partial^2 u}{\partial x \partial y} - \frac{\partial u}{\partial y}\frac{\partial v}{\partial x} - \frac{\partial u}{\partial x}\frac{\partial v}{\partial y} + \frac{\partial v}{\partial y}^2\right) + \frac{1}{2}\left(\frac{\partial^2 p}{\partial y^2} - \frac{\partial^2 p}{\partial x^2}\right)\right]\sin\theta\Bigg\}
\end{aligned} \tag{G.16}$$

The above formula is the y-momentum flow error across an interior face as a function of θ. However, as discussed in the previous section, the residual is equal to:

$$\delta_{y-\text{mom}} = \Delta J_{y-\text{mom}}(\theta) + \Delta J_{y-\text{mom}}(\theta + \frac{2\pi}{3}) + \Delta J_{y-\text{mom}}(\theta + \frac{4\pi}{3})$$

Substituting Equation (G.16) into the latter and taking the derivative of the resulting equation with respect to θ results in the maximum value of the control volume residual.

$$\begin{aligned}
\Delta J^{\max}_{y-\text{mom}} = \frac{h^3}{16}\Bigg\{ &\left[\rho\left(v\frac{\partial^2 u}{\partial y^2} - v\frac{\partial^2 u}{\partial x^2} + \frac{\partial u}{\partial y}\frac{\partial v}{\partial y} - \frac{\partial u}{\partial x}\frac{\partial v}{\partial x} + 2v\frac{\partial^2 v}{\partial x \partial y} + 2\frac{\partial v}{\partial x}\frac{\partial v}{\partial y}\right) + 2\frac{\partial^2 p}{\partial x \partial y}\right]^2 + \\
&\left[\rho\left(v\frac{\partial^2 v}{\partial x^2} - v\frac{\partial^2 v}{\partial y^2} + \frac{\partial v}{\partial x}^2 - \frac{\partial v}{\partial y}^2 + \frac{\partial u}{\partial x}\frac{\partial v}{\partial y} + \frac{\partial v}{\partial x}\frac{\partial u}{\partial y} + 2v\frac{\partial^2 u}{\partial x \partial y}\right) + \left(\frac{\partial^2 p}{\partial x^2} - \frac{\partial^2 p}{\partial y^2}\right)\right]^2 \Bigg\}^{\frac{1}{2}}
\end{aligned} \tag{G.17}$$

Appendix H

Error Indicator for Anisotropic Quadrilateral Meshes

In Appendix C, we presented a general method for estimating the residual of a numerical solution. This residual estimation method was based on flow errors across the faces of each control volume. In this appendix, we use the same principle in order to derive an error indicator based on face flow errors that is applicable to anisotropic quadrilateral unstructured meshes. For this purpose, we assume that the local distribution of solution variables is quadratic and the local geometry of the mesh is uniform and rectangular. These assumptions greatly simplify the equations for the face flow errors, presented in Appendix C. Therefore we can obtain an error indicator for anisotropic meshes.

H.1 Local Face Geometry

Figure H.1 shows an example of an interior face in a uniform anisotropic quadrilateral mesh. As seen, the control volumes are rectangles with two different characteristic sizes h_1 and h_2. We also consider the unit vectors $\hat{\mathbf{t}}$ and $\hat{\mathbf{n}}$ along the face-tangential and face-normal directions.

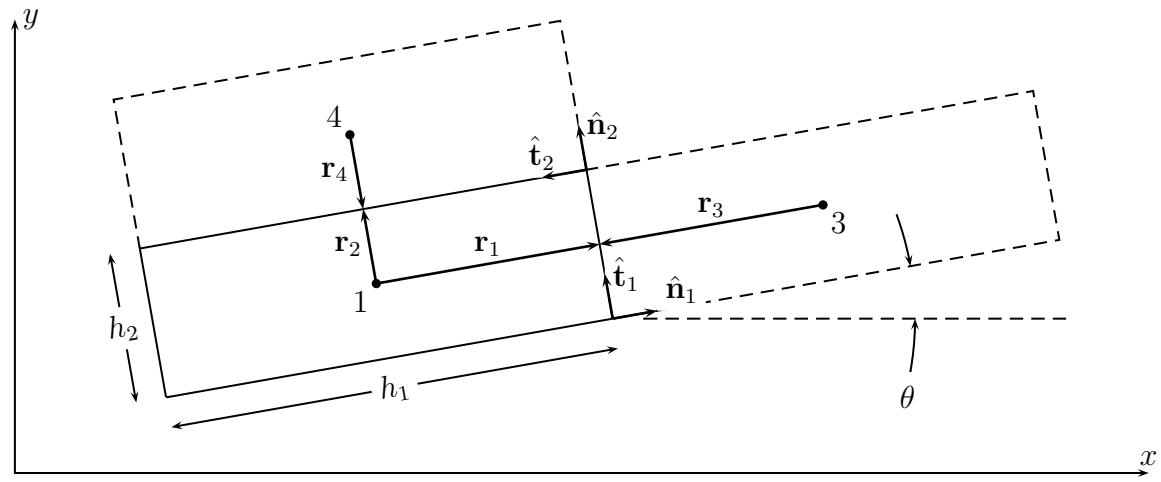


Figure H.1: Example of interior faces in an anisotropic quadrilateral mesh

H.2 Error Indicator for Mass Equation

In Appendix C, we showed that the discretization of the mass flow across an interior face reads the following formula:

$$\begin{aligned}\Delta J_{\text{vol}} =& \frac{A_f}{8}\Big[n_x(\nabla\nabla u_1 + \nabla\nabla u_2) : (\mathbf{r}_1 \otimes \mathbf{r}_2 + \mathbf{r}_2 \otimes \mathbf{r}_1 + \frac{1}{6}\mathbf{t} \otimes \mathbf{t}) - \\ & n_y(\nabla\nabla v_1 + \nabla\nabla v_2) : (\mathbf{r}_1 \otimes \mathbf{r}_2 + \mathbf{r}_2 \otimes \mathbf{r}_1 + \frac{1}{6}\mathbf{t} \otimes \mathbf{t})\Big] - \\ & \frac{\alpha d_f A_f}{4|\mathbf{s}|}(\nabla\nabla p_1 + \nabla\nabla p_2) : (\mathbf{r}_1 \otimes \mathbf{r}_2 - \mathbf{r}_2 \otimes \mathbf{r}_1)\end{aligned} \tag{H.1}$$

Let us assume that the distribution of solution variables is locally quadratic in each control volume and its neighbouring control volumes. In other words, we assume that the second order derivatives of solution variables are locally constant.

$$\begin{aligned}\nabla\nabla p_1 &= \nabla\nabla p_3 = \nabla\nabla p_4 = \nabla\nabla p \\ \nabla\nabla u_1 &= \nabla\nabla u_3 = \nabla\nabla u_4 = \nabla\nabla u \\ \nabla\nabla v_1 &= \nabla\nabla v_3 = \nabla\nabla v_4 = \nabla\nabla v\end{aligned} \tag{H.2}$$

In addition, an examination of Figure H.1 indicates:

$$\begin{aligned}
\mathbf{r}_1 &= -\mathbf{r}_3 = \frac{1}{2}h_1\hat{\mathbf{n}}_1 \\
\mathbf{r}_2 &= -\mathbf{r}_4 = \frac{1}{2}h_2\hat{\mathbf{n}}_2 \\
\hat{\mathbf{n}}_2 &= \hat{\mathbf{t}}_1 \\
\hat{\mathbf{t}}_2 &= -\hat{\mathbf{n}}_1 \\
A_{f_1} &= h_2 \\
A_{f_2} &= h_1
\end{aligned} \tag{H.3}$$

where A_{f_1} is the face area between control volumes 1 and 3 and A_{f_2} is the face area between control volumes 1 and 4. Substituting these relations back into Equation (H.1) yields:

$$\begin{aligned}
\Delta J_{\mathrm{vol}_{13}} &= \frac{h_2}{8}(n_{x_1}\nabla\nabla u + n_{y_1}\nabla\nabla v) : (-h_1^2\hat{\mathbf{n}}_1 \otimes \hat{\mathbf{n}}_1 + \frac{1}{3}h_2^2\hat{\mathbf{t}}_1 \otimes \hat{\mathbf{t}}_1) \\
\Delta J_{\mathrm{vol}_{14}} &= \frac{h_1}{8}(n_{x_2}\nabla\nabla u + n_{y_2}\nabla\nabla v) : (-h_2^2\hat{\mathbf{n}}_2 \otimes \hat{\mathbf{n}}_2 + \frac{1}{3}h_1^2\hat{\mathbf{t}}_2 \otimes \hat{\mathbf{t}}_2)
\end{aligned} \tag{H.4}$$

The latter equations show the face flow errors across the short and long faces of an anisotropic quadrilateral control volume. If we use the above formulas for estimating the overall residual of a control volume the result would be zero. The reason is that the face flow errors of opposite faces in an anisotropic quadrilateral control volume are equal in magnitude but opposite in sign. Therefore the face flow errors across the faces sums to zero. To address this problem we use the L_2-norm of face flow errors as the error indicator:

$$\begin{aligned}
\varepsilon_{\mathrm{vol}}^2 = \sum \Delta J_{\mathrm{vol}}^2 = \frac{1}{32}\Bigg\{ &\left[h_2(n_{x_1}\nabla\nabla u + n_{y_1}\nabla\nabla v) : (-h_1^2\hat{\mathbf{n}}_1 \otimes \hat{\mathbf{n}}_1 + \frac{1}{3}h_2^2\hat{\mathbf{t}}_1 \otimes \hat{\mathbf{t}}_1)\right]^2 + \\
&\left[h_1(n_{x_2}\nabla\nabla u + n_{y_2}\nabla\nabla v) : (-h_2^2\hat{\mathbf{n}}_2 \otimes \hat{\mathbf{n}}_2 + \frac{1}{3}h_1^2\hat{\mathbf{t}}_2\hat{\mathbf{t}}_2)\otimes\right]^2 \Bigg\}
\end{aligned} \tag{H.5}$$

We use the above formula as the error indicator for the mass equation. Note that ε in the above equation is not the residual, but a measure for the overall flow errors across

the faces of each control volume. Nevertheless, this measure is related to the solution residual.

For further simplification of Equation (H.5), we can write $\hat{\mathbf{t}}$ and $\hat{\mathbf{n}}$ in terms of face normal orientation, θ.

$$\begin{aligned} \hat{\mathbf{n}}_1 = -\hat{\mathbf{t}}_2 = & \cos\theta\,\hat{\mathbf{i}} + \sin\theta\,\hat{\mathbf{j}} \\ \hat{\mathbf{t}}_1 = \hat{\mathbf{n}}_2 = & -\sin\theta\,\hat{\mathbf{i}} + \cos\theta\,\hat{\mathbf{j}} \end{aligned} \tag{H.6}$$

Therefore,

$$\hat{\mathbf{n}}_1 \otimes \hat{\mathbf{n}}_1 = \hat{\mathbf{t}}_2 \otimes \hat{\mathbf{t}}_2 = \begin{pmatrix} \cos^2\theta & \cos\theta\sin\theta \\ \cos\theta\sin\theta & \sin^2\theta \end{pmatrix} \tag{H.7}$$

$$\hat{\mathbf{n}}_2 \otimes \hat{\mathbf{n}}_2 = \hat{\mathbf{t}}_1 \otimes \hat{\mathbf{t}}_1 = \begin{pmatrix} \sin^2\theta & -\cos\theta\sin\theta \\ -\cos\theta\sin\theta & \cos^2\theta \end{pmatrix} \tag{H.8}$$

Substituting the latter back into Equation (H.5) and expanding the Hessian terms results in:

$$\begin{aligned} \varepsilon_{\text{vol}}^2 = \; & \frac{h^6}{32}\Bigg\{ \left[A\left(-\frac{\partial^2 u}{\partial x^2}\right) + \frac{1}{3A^3}\left(\frac{\partial^2 u}{\partial y^2}\right)\right]\cos^3\theta + \\ & \left[A\left(-2\frac{\partial^2 u}{\partial x\partial y} - \frac{\partial^2 v}{\partial x^2}\right) + \frac{1}{3A^3}\left(-2\frac{\partial^2 u}{\partial x\partial y} + \frac{\partial^2 v}{\partial y^2}\right)\right]\cos^2\theta\sin\theta + \\ & \left[A\left(-\frac{\partial^2 u}{\partial y^2} - 2\frac{\partial^2 v}{\partial x\partial y}\right) + \frac{1}{3A^3}\left(\frac{\partial^2 u}{\partial x^2} - 2\frac{\partial^2 v}{\partial x\partial y}\right)\right]\cos\theta\sin^2\theta + \\ & \left[A\left(-\frac{\partial^2 v}{\partial y^2}\right) + \frac{1}{3A^3}\left(\frac{\partial^2 v}{\partial x^2}\right)\right]\sin^3\theta\Bigg\}^2 + \\ & \frac{h^6}{32}\Bigg\{ \left[\frac{1}{A}\left(-\frac{\partial^2 v}{\partial y^2}\right) + \frac{A^3}{3}\left(\frac{\partial^2 v}{\partial x^2}\right)\right]\cos^3\theta + \\ & \left[\frac{1}{A}\left(\frac{\partial^2 u}{\partial y^2} + 2\frac{\partial^2 v}{\partial x\partial y}\right) + \frac{A^3}{3}\left(-\frac{\partial^2 u}{\partial x^2} + 2\frac{\partial^2 v}{\partial x\partial y}\right)\right]\cos^2\theta\sin\theta + \\ & \left[\frac{1}{A}\left(-2\frac{\partial^2 u}{\partial x\partial y} - \frac{\partial^2 v}{\partial x^2}\right) + \frac{A^3}{3}\left(-2\frac{\partial^2 u}{\partial x\partial y} + \frac{\partial^2 v}{\partial y^2}\right)\right]\cos\theta\sin^2\theta + \\ & \left[\frac{1}{A}\left(\frac{\partial^2 u}{\partial x^2}\right) + \frac{A^3}{3}\left(-\frac{\partial^2 u}{\partial y^2}\right)\right]\sin^3\theta\Bigg\}^2 \end{aligned} \tag{H.9}$$

Where A is the square root of the control volume aspect ratio, h is the control volume characteristic size, and θ is the control volume orientation.

$$A = \sqrt{\frac{h_1}{h_2}}$$
$$h = \sqrt{h_1 h_2}$$

Equation (H.9) shows that the error indicator for the mass equation reads the following form:

$$\varepsilon_{\text{vol}} = h^3 f_{\text{vol}}(A, \theta, \text{physical solution})$$

To obtain the optimal value of h for mesh adaptation purposes, we need to minimize ε_{vol} in the above equation. For this purpose, we have to minimize f_{vol} first. Unfortunately, the minimization of f_{vol} can only be performed numerically since the function is very complicated. For this purpose, we need to substitute the values of all physical variables and their derivatives in Equation (H.9). The we can determine the values of control volume aspect ratio, A^2, and orientation, θ, so that the function f_{vol} becomes minimized. As a result we obtain the following formula:

$$\varepsilon_{\text{vol}} = h^3 f_{\text{vol}}^{\min}(\text{physical solution})$$

which may be used for adaptation as in Chapter 5.

H.3 Error Indicator for x-Momentum Equation

In Appendix C, we showed that the discretization of the x component of the momentum flow across an interior face is based on the following formula:

$$\begin{aligned}\Delta J_{x-\text{mom}} = &\frac{\rho u_{\text{up}} A_f}{8}\left[\nabla\nabla(\mathbf{v}_1\cdot\hat{\mathbf{n}})+\nabla\nabla(\mathbf{v}_2\cdot\hat{\mathbf{n}})\right]:(\mathbf{r}_1\otimes\mathbf{r}_2+\mathbf{r}_2\otimes\mathbf{r}_1+\frac{1}{6}\mathbf{t}\otimes\mathbf{t})-\\ &\frac{\alpha d_f \rho u_{\text{up}} A_f}{4|\mathbf{s}|}(\nabla\nabla p_1+\nabla\nabla p_2):(\mathbf{r}_1\otimes\mathbf{r}_2-\mathbf{r}_2\otimes\mathbf{r}_1)+\\ &\frac{\rho A_f}{4}(\mathbf{v}_1\cdot\hat{\mathbf{n}}+\mathbf{v}_2\cdot\hat{\mathbf{n}})\left[\nabla\nabla u_{\text{up}}:\left(\mathbf{r}_{\text{up}}\otimes\mathbf{r}_{\text{up}}+\frac{1}{12}\mathbf{t}\otimes\mathbf{t}\right)\right]+\\ &\frac{\rho A_f}{24}\left[(\nabla\mathbf{v}_1+\nabla\mathbf{v}_2):(\hat{\mathbf{n}}\otimes\hat{\mathbf{t}})\right](\nabla u_{\text{up}}\cdot\mathbf{t})+\\ &\frac{n_x A_f}{8}(\nabla\nabla p_1+\nabla\nabla p_2):\left(\mathbf{r}_1\otimes\mathbf{r}_2+\mathbf{r}_2\otimes\mathbf{r}_1+\frac{1}{6}\mathbf{t}\otimes\mathbf{t}\right)-\\ &\mu A_f\left\{\frac{\alpha}{4|\mathbf{s}|}(\nabla\nabla u_1+\nabla\nabla u_2):(\mathbf{r}_1\otimes\mathbf{r}_2-\mathbf{r}_2\otimes\mathbf{r}_1)+\right.\\ &\qquad\frac{1}{4}(\nabla\nabla u_1+\nabla\nabla u_2):[(\mathbf{r}_1+\mathbf{r}_2)\otimes\hat{\mathbf{n}})]+\\ &\qquad\left.\left[\frac{1}{4}(\nabla\nabla\mathbf{v}_1+\nabla\nabla\mathbf{v}_2)\cdot(\mathbf{r}_1+\mathbf{r}_2)\right]:(\hat{\mathbf{n}}\otimes\hat{\mathbf{i}})\right\}\end{aligned} \tag{H.10}$$

Using Equations (H.2) and (H.3) we can simplify the above equation. In addition, we can assume without loss of generality that the control volume with the subscript 1 is the upwind control volume.

$$\mathbf{r}_{\text{up}} = \mathbf{r}_1$$

$$u_{\text{up}} = u_1$$

Therefore Equation (H.10) reduces to the following equations:

$$\begin{aligned}\Delta J_{x-\text{mom}_{13}} = &\frac{1}{8}\rho u h_1(n_{x_1}\nabla\nabla u+n_{y_1}\nabla\nabla v):\left(-h_2^2\hat{\mathbf{n}}_1\otimes\hat{\mathbf{n}_1}+\frac{h_1^2}{3}\hat{\mathbf{t}}_1\otimes\hat{\mathbf{t}}_1\right)+\\ &\frac{1}{8}\rho h_1(n_{x_1}u+n_{y_1}v)\left[\nabla\nabla u:\left(h_2^2\hat{\mathbf{n}}_1\otimes\hat{\mathbf{n}}_1+\frac{h_1^2}{3}\hat{\mathbf{t}}_1\otimes\hat{\mathbf{t}}_1\right)\right]+\\ &\frac{1}{12}\rho h_1^3[(n_{x_1}\nabla u+n_{y_1}\nabla v)\cdot\hat{\mathbf{t}}_1](\nabla u\cdot\hat{\mathbf{t}}_1)+\\ &\frac{1}{8}n_{x_1}h_1\nabla\nabla p:\left(-h_2^2\hat{\mathbf{n}}_1\otimes\hat{\mathbf{n}}_1+\frac{h_1^2}{3}\hat{\mathbf{t}}_1\otimes\hat{\mathbf{t}}_1\right)\end{aligned} \tag{H.11}$$

$$\begin{aligned}\Delta J_{x-\mathrm{mom}_{14}} = &\frac{1}{8}\rho u h_2 (n_{x_2}\nabla\nabla u + n_{y_2}\nabla\nabla v) : \left(-h_1^2 \hat{\mathbf{n}}_2 \otimes \hat{\mathbf{n}}_{\mathbf{2}} + \frac{h_2^2}{3}\hat{\mathbf{t}}_2 \otimes \hat{\mathbf{t}}_2\right) + \\ &\frac{1}{8}\rho h_2 (n_{x_2} u + n_{y_2} v)\left[\nabla\nabla u : \left(h_1^2 \hat{\mathbf{n}}_2 \otimes \hat{\mathbf{n}}_2 + \frac{h_2^2}{3}\hat{\mathbf{t}}_2 \otimes \hat{\mathbf{t}}_2\right)\right] + \\ &\frac{1}{12}\rho h_2^3 [(n_{x_2}\nabla u + n_{y_2}\nabla v)\cdot \hat{\mathbf{t}}_2](\nabla u \cdot \hat{\mathbf{t}}_2) + \\ &\frac{1}{8} n_{x_2} h_2 \nabla\nabla p : \left(-h_1^2 \hat{\mathbf{n}}_2 \otimes \hat{\mathbf{n}}_2 + \frac{h_1^1}{3}\hat{\mathbf{t}}_2 \otimes \hat{\mathbf{t}}_2\right)\end{aligned} \tag{H.12}$$

The above equations represent the x-momentum flow errors across the short and long faces of an anisotropic quadrilateral control volume, respectively. If we use the above formulae for estimating the overall residual of a control volume the result would be zero since the contribution from opposite faces cancel out each other. Therefore we define the error indicator based on the L_2-norm of face flow errors to alleviate this problem.

$$\varepsilon^2_{x-\mathrm{mom}} = \Delta J^2_{x-\mathrm{mom}_{13}} + \Delta J^2_{x-\mathrm{mom}_{14}}$$

Using the above equation and the identities of Equations (H.7) and (H.8), we can reduce the error indicator, $\varepsilon_{x-\mathrm{mom}}$, to the following form:

$$
\begin{aligned}
\varepsilon^2_{x-\mathrm{mom}} = \\
\frac{h^6}{32}\Bigg\{ & \left[\; A\left(-\frac{\partial^2 p}{\partial x^2}\right) + \frac{1}{3A^3}\left(2\rho u\frac{\partial u}{\partial y} + 2\rho\frac{\partial u}{\partial y}^2 + \frac{\partial^2 p}{\partial y^2}\right)\right]\cos^3\theta + \\
& \left[\; A\left(-\rho u\frac{\partial^2 v}{\partial x^2} + \rho v\frac{\partial^2 u}{\partial x^2} - 2\frac{\partial^2 p}{\partial x\partial y}\right) + \frac{1}{3A^3}\Bigg(- 4\rho u\frac{\partial^2 u}{\partial x\partial y} + \rho u\frac{\partial^2 v}{\partial y^2} + \right. \\
& \qquad \left. \rho v\frac{\partial^2 u}{\partial y^2} - 4\rho\frac{\partial u}{\partial x}\frac{\partial u}{\partial y} + 2\rho\frac{\partial u}{\partial y}\frac{\partial v}{\partial y} - 2\frac{\partial^2 p}{\partial x\partial y}\Bigg)\right]\cos^2\theta\sin\theta + \\
& \left[\; A\left(-2\rho u\frac{\partial^2 v}{\partial x\partial y} + 2\rho v\frac{\partial^2 u}{\partial x\partial y} - \frac{\partial^2 p}{\partial y^2}\right) + \frac{1}{3A^3}\Bigg(2\rho u\frac{\partial^2 u}{\partial x^2} - 2\rho u\frac{\partial^2 v}{\partial x\partial y} - \right. \\
& \qquad \left. 2\rho v\frac{\partial^2 u}{\partial x\partial y} - 2\rho\frac{\partial u}{\partial x}\frac{\partial v}{\partial y} + 2\rho\frac{\partial u}{\partial x}^2 - 2\rho\frac{\partial u}{\partial y}\frac{\partial v}{\partial x} + \frac{\partial^2 p}{\partial x^2}\Bigg)\right]\cos\theta\sin^2\theta + \\
& \left[\; A\left(-\rho u\frac{\partial^2 v}{\partial y^2} + \rho v\frac{\partial^2 u}{\partial y^2}\right) + \frac{1}{3A^3}\left(\rho u\frac{\partial^2 v}{\partial x^2} + \rho v\frac{\partial^2 u}{\partial x^2} + 2\rho\frac{\partial u}{\partial x}\frac{\partial v}{\partial x}\right)\right]\sin^3\theta\Bigg\}^2 + \\
\frac{h^6}{32}\Bigg\{ & \left[\; \frac{1}{A}\left(-\rho u\frac{\partial^2 v}{\partial y^2} + \rho v\frac{\partial^2 u}{\partial y^2}\right) + \frac{A^3}{3}\left(\rho u\frac{\partial^2 v}{\partial x^2} + \rho v\frac{\partial^2 u}{\partial x^2} + 2\rho\frac{\partial u}{\partial x}\frac{\partial v}{\partial x}\right)\right]\cos^3\theta + \\
& \left[\; \frac{1}{A}\left(2\rho u\frac{\partial^2 v}{\partial x\partial y} - 2\rho v\frac{\partial^2 u}{\partial x\partial y} + \frac{\partial^2 p}{\partial y^2}\right) + \frac{A^3}{3}\Bigg(- 2\rho u\frac{\partial^2 u}{\partial x^2} + 2\rho u\frac{\partial^2 v}{\partial x\partial y} + \right. \\
& \qquad \left. 2\rho v\frac{\partial^2 u}{\partial x\partial y} + 2\rho\frac{\partial u}{\partial x}\frac{\partial v}{\partial y} - 2\rho\frac{\partial u}{\partial x}^2 + 2\rho\frac{\partial u}{\partial y}\frac{\partial v}{\partial x} - \frac{\partial^2 p}{\partial x^2}\Bigg)\right]\cos^2\theta\sin\theta + \\
& \left[\; \frac{1}{A}\left(-\rho u\frac{\partial^2 v}{\partial x^2} + \rho v\frac{\partial^2 u}{\partial x^2} - 2\frac{\partial^2 p}{\partial x\partial y}\right) + \frac{A^3}{3}\Bigg(- 4\rho u\frac{\partial^2 u}{\partial x\partial y} + \rho u\frac{\partial^2 v}{\partial y^2} + \right. \\
& \qquad \left. \rho v\frac{\partial^2 u}{\partial y^2} - 4\rho\frac{\partial u}{\partial x}\frac{\partial u}{\partial y} + 2\rho\frac{\partial u}{\partial y}\frac{\partial v}{\partial y} - 2\frac{\partial^2 p}{\partial x\partial y}\Bigg)\right]\cos\theta\sin^2\theta + \\
& \left[\; \frac{1}{A}\left(\frac{\partial^2 p}{\partial x^2}\right) + \frac{A^3}{3}\left(-2\rho u\frac{\partial^2 u}{\partial y^2} - 2\rho\frac{\partial u}{\partial y}^2 - \frac{\partial^2 p}{\partial y^2}\right)\right]\sin^3\theta\Bigg\}^2
\end{aligned}
\tag{H.13}
$$

The above formula is the error indicator for the x-momentum equation as a function of the mesh characteristic size, h, aspect ratio, A^2, and orientation, θ. As seen, the error indicator is of the following form:

$$\varepsilon_{x-\mathrm{mom}} = h^3 f_{x-\mathrm{mom}}(A, \theta, \text{physical solution})$$

To obtain the optimal value of h for mesh adaptation purposes, we need to minimize $\varepsilon_{x-\text{mom}}$ in the above equation. The procedure is the same as that of the mass equation, discussed in the previous section. We need to minimize $f_{x-\text{mom}}$ with respect to A and θ in order to obtain the optimal cell aspect ratio and orientation. Therefore,

$$\varepsilon_{x-\text{mom}} = h^3 f^{\min}_{x-\text{mom}}(\text{physical solution})$$

which is used in Chapter 5 for anisotropic mesh adaptation.

H.4 Error Indicator for y-Momentum Equation

The derivation of the error indicator for the y-Momentum equation is similar to that of the x-Momentum, presented in the previous section. We start from the discretization error of the y component of the momentum flow across an interior face.

$$\begin{aligned}
\Delta J_{y-\text{mom}} = {} & \frac{\rho v_{\text{up}} A_f}{8}\left[\nabla\nabla(\mathbf{v}_1\cdot\hat{\mathbf{n}}) + \nabla\nabla(\mathbf{v}_2\cdot\hat{\mathbf{n}})\right] : (\mathbf{r}_1\otimes\mathbf{r}_2 + \mathbf{r}_2\otimes\mathbf{r}_1 + \frac{1}{6}\mathbf{t}\otimes\mathbf{t}) - \\
& \frac{\alpha d_f \rho v_{\text{up}} A_f}{4|\mathbf{s}|}(\nabla\nabla p_1 + \nabla\nabla p_2) : (\mathbf{r}_1\otimes\mathbf{r}_2 - \mathbf{r}_2\otimes\mathbf{r}_1) + \\
& \frac{\rho A_f}{4}(\mathbf{v}_1\cdot\hat{\mathbf{n}} + \mathbf{v}_2\cdot\hat{\mathbf{n}})\left[\nabla\nabla v_{\text{up}} : \left(\mathbf{r}_{\text{up}}\otimes\mathbf{r}_{\text{up}} + \frac{1}{12}\mathbf{t}\otimes\mathbf{t}\right)\right] + \\
& \frac{\rho A_f}{24}\left[(\nabla\mathbf{v}_1 + \nabla\mathbf{v}_2) : (\hat{\mathbf{n}}\otimes\hat{\mathbf{t}})\right](\nabla v_{\text{up}}\cdot\mathbf{t}) + \\
& \frac{n_y A_f}{8}(\nabla\nabla p_1 + \nabla\nabla p_2) : \left(\mathbf{r}_1\otimes\mathbf{r}_2 + \mathbf{r}_2\otimes\mathbf{r}_1 + \frac{1}{6}\mathbf{t}\otimes\mathbf{t}\right) - \\
& \mu A_f\left\{\frac{\alpha}{4|\mathbf{s}|}(\nabla\nabla v_1 + \nabla\nabla v_2) : (\mathbf{r}_1\otimes\mathbf{r}_2 - \mathbf{r}_2\otimes\mathbf{r}_1) + \right. \\
& \qquad \frac{1}{4}(\nabla\nabla v_1 + \nabla\nabla v_2) : [(\mathbf{r}_1 + \mathbf{r}_2)\otimes\hat{\mathbf{n}})] + \\
& \qquad \left.\left[\frac{1}{4}(\nabla\nabla\mathbf{v}_1 + \nabla\nabla\mathbf{v}_2)\cdot(\mathbf{r}_1 + \mathbf{r}_2)\right] : (\hat{\mathbf{n}}\otimes\hat{\mathbf{j}})\right\}
\end{aligned} \tag{H.14}$$

Using Equations (H.2) and (H.3) we can simplify the above equation. In addition, we can assume without loss of generality that the control volume with the subscript 1 is the

upwind control volume.

$$\mathbf{r}_{\text{up}} = \mathbf{r}_1$$
$$u_{\text{up}} = u_1$$

Therefore Equation (H.14) reduces to the following equations:

$$\begin{aligned}
\Delta J_{x-\text{mom}_{13}} = &\frac{1}{8}\rho v h_1 (n_{x_1}\nabla\nabla u + n_{y_1}\nabla\nabla v) : \left(-h_2^2 \hat{\mathbf{n}}_1 \otimes \hat{\mathbf{n}_1} + \frac{h_1^2}{3}\hat{\mathbf{t}}_1 \otimes \hat{\mathbf{t}}_1\right) + \\
&\frac{1}{8}\rho h_1 (n_{x_1} u + n_{y_1} v)\left[\nabla\nabla v : \left(h_2^2 \hat{\mathbf{n}}_1 \otimes \hat{\mathbf{n}}_1 + \frac{h_1^2}{3}\hat{\mathbf{t}}_1 \otimes \hat{\mathbf{t}}_1\right)\right] + \\
&\frac{1}{12}\rho h_1^3 [(n_{x_1}\nabla u + n_{y_1}\nabla v)\cdot\hat{\mathbf{t}}_1](\nabla v \cdot \hat{\mathbf{t}}_1) + \\
&\frac{1}{8} n_{y_1} h_1 \nabla\nabla p : \left(-h_2^2 \hat{\mathbf{n}}_1 \otimes \hat{\mathbf{n}}_1 + \frac{h_1^2}{3}\hat{\mathbf{t}}_1 \otimes \hat{\mathbf{t}}_1\right)
\end{aligned} \tag{H.15}$$

$$\begin{aligned}
\Delta J_{x-\text{mom}_{14}} = &\frac{1}{8}\rho v h_2 (n_{x_2}\nabla\nabla u + n_{y_2}\nabla\nabla v) : \left(-h_1^2 \hat{\mathbf{n}}_2 \otimes \hat{\mathbf{n}_2} + \frac{h_2^2}{3}\hat{\mathbf{t}}_2 \otimes \hat{\mathbf{t}}_2\right) + \\
&\frac{1}{8}\rho h_2 (n_{x_2} u + n_{y_2} v)\left[\nabla\nabla v : \left(h_1^2 \hat{\mathbf{n}}_2 \otimes \hat{\mathbf{n}}_2 + \frac{h_2^2}{3}\hat{\mathbf{t}}_2 \otimes \hat{\mathbf{t}}_2\right)\right] + \\
&\frac{1}{12}\rho h_2^3 [(n_{x_2}\nabla u + n_{y_2}\nabla v)\cdot\hat{\mathbf{t}}_2](\nabla v \cdot \hat{\mathbf{t}}_2) + \\
&\frac{1}{8} n_{y_2} h_2 \nabla\nabla p : \left(-h_1^2 \hat{\mathbf{n}}_2 \otimes \hat{\mathbf{n}}_2 + \frac{h_1^1}{3}\hat{\mathbf{t}}_2 \otimes \hat{\mathbf{t}}_2\right)
\end{aligned} \tag{H.16}$$

The above equations represent the y-momentum flow errors across the short and long faces of an anisotropic quadrilateral control volume, respectively. If we use the above formulae for estimating the overall residual of a control volume the result would be zero since the contribution from opposite faces cancel out each other. Therefore we define the error indicator based on the L_2-norm of face flow errors to alleviate this problem.

$$\varepsilon^2_{y-\text{mom}} = \Delta J^2_{y-\text{mom}_{13}} + \Delta J^2_{y-\text{mom}_{14}}$$

Using the above equation and the identities of Equations (H.7) and (H.8), we can reduce the error indicator, $\varepsilon_{y-\mathrm{mom}}$, to the following form:

$$
\begin{aligned}
\varepsilon_{y-\mathrm{mom}}^2 = \\
\frac{h^6}{32}\Bigg\{ & \left[A\left(-\rho v\frac{\partial^2 u}{\partial x^2}+\rho u\frac{\partial^2 v}{\partial x^2}\right)+\frac{1}{3A^3}\left(\rho v\frac{\partial^2 u}{\partial y^2}+\rho u\frac{\partial^2 v}{\partial y^2}+2\rho\frac{\partial u}{\partial y}\frac{\partial v}{\partial y}\right)\right]\cos^3\theta + \\
& \left[A\left(2\rho u\frac{\partial^2 v}{\partial x\partial y}-2\rho v\frac{\partial^2 u}{\partial x\partial y}-\frac{\partial^2 p}{\partial x^2}\right)+\frac{1}{3A^3}\Bigg(-2\rho v\frac{\partial^2 u}{\partial x\partial y}+2\rho v\frac{\partial^2 v}{\partial y^2}- \right. \\
& \qquad \left. 2\rho u\frac{\partial^2 v}{\partial x\partial y}-2\rho\frac{\partial u}{\partial x}\frac{\partial v}{\partial y}+2\rho\frac{\partial v}{\partial y}^2-2\rho\frac{\partial u}{\partial y}\frac{\partial v}{\partial x}+\frac{\partial^2 p}{\partial y^2}\Bigg)\right]\cos^2\theta\sin\theta + \\
& \left[A\left(-\rho v\frac{\partial^2 u}{\partial y^2}+\rho u\frac{\partial^2 v}{\partial y^2}-2\frac{\partial^2 p}{\partial x\partial y}\right)+\frac{1}{3A^3}\Bigg(-4\rho v\frac{\partial^2 v}{\partial x\partial y}+\rho v\frac{\partial^2 u}{\partial x^2}+ \right. \\
& \qquad \left. \rho u\frac{\partial^2 v}{\partial x^2}-4\rho\frac{\partial v}{\partial x}\frac{\partial v}{\partial y}+2\rho\frac{\partial u}{\partial x}\frac{\partial v}{\partial x}-2\frac{\partial^2 p}{\partial x\partial y}\Bigg)\right]\cos\theta\sin^2\theta + \\
& \left[A\left(-\frac{\partial^2 p}{\partial y^2}\right)+\frac{1}{3A^3}\left(2\rho v\frac{\partial^2 v}{\partial x^2}+2\rho\frac{\partial v}{\partial x}^2+\frac{\partial^2 p}{\partial x^2}\right)\right]\sin^3\theta\Bigg\}^2 + \\
\frac{h^6}{32}\Bigg\{ & \left[\frac{1}{A}\left(-\frac{\partial^2 p}{\partial y^2}\right)+\frac{A^3}{3}\left(2\rho v\frac{\partial^2 v}{\partial x^2}+2\rho\frac{\partial v}{\partial x}^2+\frac{\partial^2 p}{\partial x^2}\right)\right]\cos^3\theta + \\
& \left[\frac{1}{A}\left(\rho v\frac{\partial^2 u}{\partial y^2}-\rho u\frac{\partial^2 v}{\partial y^2}+2\frac{\partial^2 p}{\partial x\partial y}\right)+\frac{A^3}{3}\Bigg(4\rho v\frac{\partial^2 v}{\partial x\partial y}-\rho v\frac{\partial^2 u}{\partial x^2}- \right. \\
& \qquad \left. \rho u\frac{\partial^2 v}{\partial x^2}+4\rho\frac{\partial v}{\partial x}\frac{\partial v}{\partial y}-2\rho\frac{\partial u}{\partial x}\frac{\partial v}{\partial x}+2\frac{\partial^2 p}{\partial x\partial y}\Bigg)\right]\cos^2\theta\sin\theta + \\
& \left[\frac{1}{A}\left(2\rho u\frac{\partial^2 v}{\partial x\partial y}-2\rho v\frac{\partial^2 u}{\partial x\partial y}-\frac{\partial^2 p}{\partial x^2}\right)+\frac{A^3}{3}\Bigg(-2\rho v\frac{\partial^2 u}{\partial x\partial y}+2\rho v\frac{\partial^2 v}{\partial y^2}- \right. \\
& \qquad \left. 2\rho u\frac{\partial^2 v}{\partial x\partial y}-2\rho\frac{\partial u}{\partial x}\frac{\partial v}{\partial y}+2\rho\frac{\partial v}{\partial y}^2-2\rho\frac{\partial u}{\partial y}\frac{\partial v}{\partial x}+\frac{\partial^2 p}{\partial y^2}\Bigg)\right]\cos\theta\sin^2\theta + \\
& \left[\frac{1}{A}\left(\rho v\frac{\partial^2 u}{\partial x^2}-\rho u\frac{\partial^2 v}{\partial x^2}\right)+ \right. \\
& \qquad \left. \frac{A^3}{3}\left(-\rho v\frac{\partial^2 u}{\partial y^2}-\rho u\frac{\partial^2 v}{\partial y^2}-2\rho\frac{\partial u}{\partial y}\frac{\partial v}{\partial y}\right)\right]\sin^3\theta\Bigg\}^2
\end{aligned}
\tag{H.17}
$$

The above formula is the error indicator for the y-momentum equation as a function of mesh characteristic size, h, aspect ratio, A^2, and orientation, θ. As seen, the error indicator is of the following form:

$$\varepsilon_{y-\text{mom}} = h^3 f_{y-\text{mom}}(A, \theta, \text{physical solution})$$

To obtain the optimal value of h for mesh adaptation purposes, we need to minimize $\varepsilon_{y-\text{mom}}$ in the above equation. The procedure is the same as that of the mass equation, discussed in Section H.1. We need to minimize $f_{y-\text{mom}}$ with respect to A and θ in order to obtain the optimal cell aspect ratio and orientation. Therefore,

$$\varepsilon_{y-\text{mom}} = h^3 f^{\min}_{y-\text{mom}}(\text{physical solution})$$

which is used in Chapter 5 for anisotropic mesh adaptation.

Appendix I

Intersection of Metric Tensors

In Chapter 5, we explained that for combining a few anisotropic metric tensors, we have to find their intersection. In this work, we use a slightly modified version of the algorithm proposed by Borouchaki *et al.* [19] and Castro-Diaz *et al.* [26]. In this algorithm, we calculate the eigenvalues and eigenvectors of each metric tensor and then estimate their intersection. The intersection metric calculated by this algorithm is approximate. However, its accuracy is sufficient for the most practical applications.

Algorithm 2 Intersection of two metric tensors

Require: Metric tensors: $\mathbf{M}_1$ and $\mathbf{M}_2$

Ensure: $\mathbf{H}$ is the intersection of $\mathbf{M}_1$ and $\mathbf{M}_2$

1: Find eigenvalues of $\mathbf{M}_1$, λ_{11} and λ_{12}
2: Find eigenvectors of $\mathbf{M}_1$, $\mathbf{v}_{11}$ and $\mathbf{v}_{12}$
3: $l_{11} \leftarrow \max(\lambda_{11}, \mathbf{v}_{11}^T \mathbf{M}_2 \mathbf{v}_{11})$
4: $l_{12} \leftarrow \max(\lambda_{12}, \mathbf{v}_{12}^T \mathbf{M}_2 \mathbf{v}_{12})$
5: Build tensor $\mathbf{H}_1$ with eigenvalues l_{11} and l_{12} and eigenvectors $\mathbf{v}_{11}$ and $\mathbf{v}_{12}$
6: $k_1 = \max(\lambda_{11}, \lambda_{12}) / \min(\lambda_{11}, \lambda_{12})$
7: Find eigenvalues of $\mathbf{M}_2$, λ_{21} and λ_{22}
8: Find eigenvectors of $\mathbf{M}_2$, $\mathbf{v}_{21}$ and $\mathbf{v}_{22}$
9: $l_{21} \leftarrow \max(\lambda_{21}, \mathbf{v}_{21}^T \mathbf{M}_1 \mathbf{v}_{21})$
10: $l_{22} \leftarrow \max(\lambda_{22}, \mathbf{v}_{22}^T \mathbf{M}_1 \mathbf{v}_{22})$
11: Build tensor $\mathbf{H}_2$ with eigenvalues l_{21} and l_{22} and eigenvectors $\mathbf{v}_{21}$ and $\mathbf{v}_{22}$
12: $k_2 = \max(\lambda_{21}, \lambda_{22}) / \min(\lambda_{21}, \lambda_{22})$
13: **if** $k_1 > k_2$ **then**
14: $\quad \mathbf{H} \leftarrow \mathbf{H}_1$
15: **else**
16: $\quad \mathbf{H} \leftarrow \mathbf{H}_2$
17: **end if**