Spring 1-1-2013

Anisotropic Mesh Adaptivity for Turbulent Flows with Boundary Layers

Kedar C. Chitale
University of Colorado at Boulder, kedar.chitale@colorado.edu

Anisotropic Mesh Adaptivity for Turbulent Flows with Boundary Layers

by

Kedar C. Chitale

B.Tech., College of Engineering, Pune, 2007

M.S., Rensselaer Polytechnic Institute, 2010

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Department of Aerospace Engineering Sciences

2013

This thesis entitled:
Anisotropic Mesh Adaptivity for Turbulent Flows with Boundary Layers
written by Kedar C. Chitale
has been approved for the Department of Aerospace Engineering Sciences

Kenneth Jansen

Prof. Kurt Maute

Date

The final copy of this thesis has been examined by the signatories, and we find that both the content and the form meet acceptable presentation standards of scholarly work in the above mentioned discipline.

Chitale, Kedar C. (Ph.D., Aerospace Engineering)

Anisotropic Mesh Adaptivity for Turbulent Flows with Boundary Layers

Thesis directed by Prof Kenneth Jansen

Turbulent flows are found everywhere in nature and are studied, analyzed and simulated using various experimental and numerical tools. For computational analysis, a variety of turbulence models are available and the accuracy of these models in capturing the phenomenon depends largely on the mesh spacings, especially near the walls, in the boundary layer region. Special semi-structured meshes called *mesh boundary layers* are widely used in the CFD community in simulations of turbulent flows, because of their graded and orthogonal layered structure. They provide an efficient way to achieve very fine and highly anisotropic mesh spacings without introducing poorly shaped elements. Since usually the required mesh spacings to accurately resolve the flow are not known *a priori* to the simulations, an adaptive approach based on *a posteriori* error indicators is used to achieve an appropriate mesh.

In this study, we apply the adaptive meshing techniques to turbulent flows with a focus on boundary layers. We construct a framework to calculate the critical wall normal mesh spacings inside the boundary layers based on the flow physics and the knowledge of the turbulence model. This approach is combined with numerical error indicators to adapt the entire flow region. We illustrate the effectiveness of this hybrid approach by applying it to three aerodynamic flows and studying their superior performance in capturing the flow structures in detail.

We also demonstrate the capabilities of the current developments in parallel boundary layer mesh adaptation by applying them to two internal flow problems. We also study the application of adaptive boundary layer meshes to complex geometries like multi element wings. We highlight the advantage of using such techniques for superior wake and tip region resolution by showcasing flow results. We also outline the future direction for the adaptive meshing techniques to be useful to the large scale flow computations.

Dedication

I dedicate this dissertation to my late mother, who still lives on in our hearts.

Acknowledgements

First and foremost, I would like to express my deep gratitude towards Prof. Kenneth Jansen for supporting and encouraging me throughout my entire time working in the PHASTA research group. It has been a tremendous honor to work as one of his graduate students. I was able to cultivate a good research appetite, critical thinking and problem solving skills while working with him. I would also like to thank him for providing all the necessary resources which were imperative for the research.

I would also like to express my thanks to Prof. Mark Shephard and Dr. Onkar Sahni who were close collaborators in this work. Their guidance and critical assessment was crucial in taking this research to the next level. They also made sure that I had all the resources from SCOREC available in timely manner to complete this work. Mark Beall, Saurabh Tendulkar and Rocco Nastasia of Simmetrix Inc. were very supportive in delivering their software to us when required. I would also like to thank NASA for supporting this work for past three years.

This work would not have been possible without the help and contributions from my other colleagues Dr. Alex Ovcharenko, Dr. Michel Rasquin, Dr. Seegyoung Seol, Dr. Yi Chen, Dr. Min Zhou, Cameron Smith, Jeff Martin and Benjamin Matthews. All my friends in U.S. and back home in India also played an important part in motivating me when it was really required.

Last but not the least, I express my deepest love and thanks to my wife Dr. Janhavi Modak and my father Prof. Chandrashekhar Chitale and my other family members, whose unconditional love and support kept me moving towards my goal.

Contents

Chapter

Tables

Table

Figures

Figure

Chapter 1

Introduction

1.1 Background

From the vortices in teacups to the whirlpools in sinks; from the motion of turbines to the flight of aircrafts; from the winds of hurricanes and tornadoes to the cosmic star formations from the clouds of dust; fluid flows are ubiquitous and are critical at all levels of our interaction with the universe. Most of these flows are unsteady, random, chaotic and highly irregular in nature, or simply put, turbulent. It is said that Leonardo Da Vinci was the first one to study turbulence, sometime in the 15th century, by observing water motion. Since then, six centuries have passed and turbulence still remains an unsolved problem in classical physics and a favorite topic of researchers. Different analytical, experimental and numerical methods have been invented and used to understand the basic physics behind this complex phenomenon.

In the 18th century, many scientists like Bernoulli, Euler, d'Alembert attempted to describe the fluid flows using mathematical equations. In 1840s, Navier-Stokes equations were formed with the addition of viscous effects' treatment to above attempts, and became the foundation stone of today's numerical methods in fluid dynamics. In 1858, Helmholtz established the three laws of vortex motion, which can still be found in any fluid mechanics textbook. In 1904, Ludwig Prandtl's experiments led to the discovery of boundary layers formed near the walls, by action of viscosity, which was a crucial development in fluid dynamics. In most of the 19th century and in the early part of the 20th century, the research in fluid dynamics was focussed on experimental and analytical studies as well as important theoretical developments.

With the advent of computers in the late 20th century, it became possible to solve the Navier-Stokes equations numerically, to study the physics of fluid flows. This approach is called computational fluid dynamics (CFD), which relies on the use of computers to numerically solve a system of partial differential equations (PDEs). To achieve this, the flow domain is discretized into smaller pieces. A *mesh* or a *grid* is a spatial discretization of the physical domain, comprising of finite number of mesh elements or grid cells. Different techniques like finite differences, finite volumes and finite elements were postulated to discretize and solve the PDEs. Structured grids were at first popular because of their ease of implementation and speed of calculations. But with their difficulties to adjust to complex geometries, unstructured grids started gaining popularity as computational power and reach of CFD extended. Distributed systems like supercomputers and clusters became available and computations could now be performed on many processors instead of one. This opened new doors for CFD, as much faster calculations than before were now possible. In the last two decades, the use of computational fluid dynamics has increased tremendously and has helped in improving designs, and optimizing performance, mainly in automotive, aerospace, environmental, and energy industries.

One thing to note is that with these computational methods, only approximate solutions can be obtained. The accuracy depends on various factors such as discretization scheme used, order of the numerical method etc. To get a realistic solution, it is highly desirable that inherent discretization errors due to the spacial approximations are minimized. In most complex flow problems, it is difficult to determine the adequate mesh resolution *a priori* to the simulations. In such cases, an initial mesh is used to get a first estimate of the flow solution, and the mesh is adapted using *a posteriori* error indicators.

To improve the spatial discretization, increasing the resolution in a uniform global fashion to a level required for acceptable accuracy would introduce excessive demands on the computational resources. In order to expedite the performance of numerical computations, resolution within the discretization needs to be applied in a local fashion. For this, mesh adaptation is done by locally modifying the mesh elements according to some size field information, provided by suitable

error estimators or error indicators. Error estimators provide a measure of the solution error in a certain norm. For the Navier-Stokes equations, no reliable *a priori* error estimators exist to predict the error in the solution. Because of this, *a posteriori* error indicators are used, which provide relative error information. In essence, error indicators are qualitative error measures derived from the flow solution. Traditionally, the size field is based on some scalar error information, that allows for isotropic adaptivity resulting in equilateral elements. However, real world fluid problems exhibit highly anisotropic flow features, such as boundary layers near no-slip walls, or shock waves in transonic and supersonic flows. Such solution features are most efficiently resolved by using elements which are oriented in a certain manner with significant anisotropy.

This thesis focuses on anisotropic adaptivity of turbulent flows, with a focus on boundary layers. One goal is to explore new adaptivity indicators derived from the flow physics, to decide the mesh resolution in the boundary layers, coupling them with numerical error indicators to get a hybrid strategy. Another goal is to guide the parallel anisotropic adaptivity of boundary layer meshes. The third goal is to study applications of these procedures to complex turbulent flow problems.

1.2 State of the Art and Contribution of This Thesis

Boundary layers have been studied extensively both experimentally and computationally and the near wall resolution requirements are well understood for different modeling approaches used to simulate turbulent flows [36], [37], [38], [69], [72]. For example, Reynolds-averaged Navier-Stokes (RANS) simulations have well understood near wall spacing requirements, based on the models being used (e.g. k-epsilon vs. Spalart Allmaras), and the wall treatment (wall modeled vs. wall resolved). For large eddy simulation (LES), not only the normal spacing but also the streamwise and spanwise spacings are well understood. These spacings are usually defined in plus units (y^+), which are a non-dimensional length scale derived from the molecular viscosity (μ) and a velocity scale ($u_\tau = \sqrt{\tau/\rho}$). Simple flows that include the effects of curvature and streamwise pressure gradient provide reliable guidance to what each model class requires in terms of these length scales

for attached boundary layers.

In case of viscous flows, better results can be obtained in regions of the boundary layers, with the help of a semi-structured mesh near the no-slip walls, called as *mesh boundary layers* [27]. These meshes are widely used with prior knowledge of boundary layers [19], [30], [35], [39], [42], [49], [50]. The turbulence modeling requirements are more readily met with a boundary layer structured grid near the walls, as opposed to an unstructured mesh. Highly anisotropic grids (with anisotropy of 10,000 or more) can be constructed with the semi-structured mesh, as opposed to unstructured meshes, without introducing poorly shaped elements with large dihedral angles. This stack of elements grows up to a specified distance from the wall and rest of the flow region is meshed with unstructured elements. A common method to construct these meshes is *advancing layers method* [58]. Favorable attributes of such meshes are high-aspect ratio, orthogonal and graded elements near the walls. Since this layered nature of elements is highly desirable, it needs to be preserved during adaptation. Figure 1.1 shows an example of mesh boundary layers for a pipe geometry. In 3D, a boundary layer mesh consist of prism and pyramid elements, whereas the unstructured region of the mesh consists of tetrahedral elements.

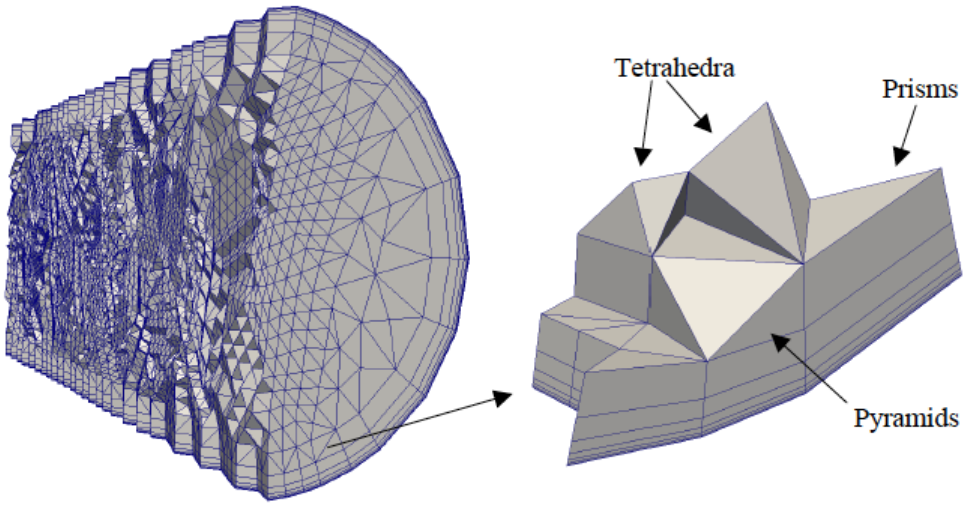


Figure 1.1: Cut of a typical finite element mesh for a pipe geometry showing a mesh boundary layer

However, a major drawback of the boundary layer structured meshing techniques is that a

prior specification of the surface mesh resolution and the point distribution along the wall normals (e.g. first layer height, total thickness of the layers, growth rate) can require a great deal of user intervention [60]. This requires expertise and knowledge of the flow solution, which is not available prior to solving the flow. For example, in transonic and supersonic cases, flow often encounters shock waves, and their positions are not known *a priori* to the flow solve. Since these are regions of discontinuities and large changes in the solution, the mesh required in their vicinity has to be very fine. Such flows either require very fine initial meshes over large portions of the domain, or many cycles of meshing to "learn" the location of the solution features and specify mesh sizes *a priori* accordingly, or adaptive grids.

Adaptive grid techniques aim at reducing the dependence on the expertise of the user as well as reducing the manual labour cost of regenerating the mesh to get better results, after visualizing the solution. Practically, one would have to undergo several cycles of meshing and re-meshing until they get to a satisfactory mesh. Adaptive grids, however, can avoid this manual repetitive cycle by automating some of the process.

Baker [8] gives an overview of different adaptation strategies for fluid dynamics. There are different adaptive methods including h (spatial adaptation), p (order of the polynomial), and r (mesh movement). Out of these, h adaptive methods are the most suitable and popular adaptation techniques for finite elements in fluid dynamics, especially when shocks or boundary layers are present. The main goal of these methods is to reduce the discretization errors by increasing the resolution in the required areas of the mesh. The basic principle behind these methods is equidistribution of the error, such that the error is equal in every element of the mesh. Babuska and Rheinboldt [7] showed that finite element meshes produce optimal results when the error is equidistributed for all elements. Under simplified assumptions, this approach has been shown to minimize the global error.

The h adaptive methods can again be classified in two broad categories, isotropic and anisotropic adaptive methods. Isotropic adaptive methods use a scalar error density and result in equilateral elements in each direction. The choices for the error estimator/indicator can vary

from the solution residual [68], [75], the adjoint solution [55], [77], the truncation error [11], [12] and the solution gradients [9]. However, with problems exhibiting high anisotropy, error representation based on a scalar quantity might not produce an optimal mesh [6]. For this, anisotropic adaptive methods can be used to further optimize the shape of the elements.

Anisotropic adaptivity of unstructured grids based on a mesh metric field is well understood and has been applied to real world flow problems [3], [14], [15], [16], [26], [48], [54], [57]. These methods have been extended to semi-structured grids near the walls by decomposing the mesh boundary layers in terms of an in-plane and a normal directions [62]. Boundary layer adaptivity with unstructured grids has also been carried out on a single processor, changing the in-plane spacing [61], for viscous flows. However, previous studies have been limited to pipe and cardiovascular flows, with an emphasis only on resolving the flow quantities near the walls, like the wall shear stress. However, for turbulent flows, the boundary layer meshes are critical not only for resolving the wall quantities, but for overall behavior of the physical boundary layer and the accuracy of the turbulence models, and have a big impact on the overall structure of the flow. Therefore, studying the application of adaptive boundary layer meshes to turbulent flows is required, and is the primary focus of this thesis.

Since the turbulence models have specific requirements for the near wall mesh spacings, it is natural to think about using the flow physics information to drive the boundary layer mesh adaptation. Flow physics can also give information about the local height of the physical boundary layer. This combined information can be an effective indicator for thickness adaptation of boundary layers and can be used to automatically adjust the boundary layer meshes. However, this approach is limited to the boundary layers and needs to be combined with other numerical error indicators to drive adaptivity of the entire flow domain. In this thesis, we present such a hybrid strategy and demonstrate it's effectiveness by applying it to three difference aerodynamic flows.

So far, computational power has increased as per Moore's law's prediction since 1950s, doubling nearly every two years. In recent times, this has largely been possible due to the parallel processing paradigm. Recently, supercomputer systems have been developed with hundreds of

thousands of cores for computation, clocking petaflops (10^{15}) speeds. As new and more complex problems are undertaken for computational studies, it is desirable to move the simulations to a parallel framework, to make the most of these available resources. This has two advantages, one is handling larger problem sizes than before, and second is achieving the results faster than before. With efforts going on to achieve exa-scale computation, the way ahead for CFD is undoubtedly with parallel processing.

For flow simulations, we use a finite element formulation which is described in references [79] for incompressible flows and [80] for compressible flows. Our implementation of this formulation already has full capability to run in parallel and has been shown to scale very well on different systems up to 288k cores for 1.2 billion tetrahedra element mesh [63]. Efforts are being carried out to test the flow solver on 3 million MPI processes with a 92 billion tetrahedra element mesh for an aerodynamic rudder geometry. Clearly, if adaptive techniques are to be applied to these large meshes, it has to be done in parallel. In addition to developing error indictors that drive the anisotropic size field determination, a secondary goal of this thesis is to guide the development of parallel boundary layer adaptivity and demonstrate the capabilities as they become fully operational with additional adaptive features. Our codes are designed such that all aspects can work on parallel systems. We showcase the advances in parallel boundary layer adaptivity by applying it to two internal flow problems.

1.3 Outline of the Thesis

The organization of this thesis is as follows. In the second chapter, we introduce Hessian based anisotropic adaptation methods and size field calculation. We also explain how these methods are extended to boundary layer meshes. In the third chapter, we introduce different attributes of the boundary layer meshes and illustrate different strategies for their calculations using the flow physics. In the fourth chapter, we apply our thickness adaptation techniques from Chapter 3 to three aerodynamic flow problems, showing their effectiveness and efficiency. In the fifth chapter, we study the application of serial adaptive boundary layer meshes to complex multi element wings,

highlighting their superior flow capturing. In the sixth chapter, we showcase two internal flow applications of the parallel in-plane adaptive boundary layer techniques with results. In the seventh and the last chapter we provide concluding remarks and discuss the future path for adaptive techniques.

Chapter 2

Anisotropic Adaptation and Extension to Boundary Layers

The accuracy of the flow simulations depends strongly on the mesh quality and resolution. To get the best quality results, the resolution of the mesh can be increased globally, in a uniform fashion. However, the available computational resources put a constraint on such a global refinement, especially for complex problems. A smart way to deal with this is to adapt the mesh only in the regions which require higher resolution. This can be done by anticipating which regions in the flow require higher resolution *a priori*, e.g. corners and blends. However, for more complex problems, this approach largely depends on the expertise of the user and therefore can be problematic. This underlines the need for an automated adaptive procedure, which aims at reducing the error in the solution, and produces high quality meshes.

Turbulent fluid flows posses highly anisotropic flow features such as boundary layers near the no-slip walls or shock waves in transonic and supersonic flows. Applying mesh adaptation in an isotropic fashion in such a case would mean wastage of computational resources. To remedy this, mesh adaptation needs to make use of the anisotropic information embedded in the flow solution variables, and increase the efficiency of the simulations. As already discussed, *a priori* error estimators do not exist for Navier-Stokes equations, so reliable *a posteriori* error indicators need to be constructed that provide qualitative measure of the error.

This chapter describes interpolation based error indicators and their combinations with other numerical error estimators. We then discuss how this formulation can be extended to boundary layer meshes.

2.1 Hessian Based Adaptation

The finite element formulation of our flow solver described in the reference [79], makes use of piecewise linear basis functions. This means that the main interpolation error term is of second order. Consequently, the second derivatives of the solution variables form the major portion of the discretization error [18]. The error analysis in [25] and [31] has shown that the rate of convergence of the total error is same as that of the interpolation estimates in their respective cases, with simplified assumptions.

In 3D flow problems, directional error indicators can be obtained by using second derivatives of a solution field in the form of a Hessian:

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 u}{\partial x^2} & \frac{\partial^2 u}{\partial x \partial y} & \frac{\partial^2 u}{\partial x \partial z} \\ \frac{\partial^2 u}{\partial x \partial y} & \frac{\partial^2 u}{\partial y^2} & \frac{\partial^2 u}{\partial y \partial z} \\ \frac{\partial^2 u}{\partial x \partial z} & \frac{\partial^2 u}{\partial y \partial z} & \frac{\partial^2 u}{\partial z^2} \end{bmatrix} \tag{2.1}$$

The Hessian is a symmetric matrix and can be computed from any scalar solution field. The popular choices are speed and density, but some combinations of two or more variables can also be chosen. It is possible to obtain local estimates of the interpolation error in different norms, based on the Hessian [5], [24], [43].

The Hessian matrix is decomposed as $H = R\Lambda R^T$, where R is the eigenvector matrix and Λ is the diagonal matrix of eigenvalues. The directions associated with the eigenvectors are referred to as the principal directions and the eigenvalues are equivalent to the second derivatives along the local principal directions. High eigenvalues are associated with high error in the corresponding principal direction. Similarly, a low eigenvalue means lower error in the corresponding direction [62]. Mesh sizes (mesh edge lengths required in a particular direction) can be calculated from this at each vertex of the mesh to drive the mesh adaptation procedures.

Ideally, the Hessian can be constructed using any solution quantity. However, it is only natural that the nature of the problem should guide this choice. Traditionally, only velocity or density Hessians have been used for adaptation. But in some fluid flows, like impinging jets, it

might make sense to use the total pressure or some other combination of velocity and pressure to compute the Hessian. Also, specific weights can be assigned to each quantity before calculating the Hessian. In another approach, Hessians of different flow quantities can be constructed and the resulting mesh metric fields can be intersected to give a final metric. Experience indicates that the former approach results is a better match to the solution anisotropy that is often lost by the intersection of different metric fields and thus it is generally preferred.

2.1.1 Concept of a Mesh Metric Field

The Hessian provides qualitative information about where the local interpolation error is higher and where it is lower. From this information, actual mesh sizes need to be calculated for mesh adaptation. For finite elements, mesh sizes can be defined in terms of the distances between two adjacent mesh vertices (edge lengths). Traditionally, the mesh size field has been specified as a scalar quantity, but for anisotropic adaptation it is defined with a second order tensor. To specify an anisotropic size distribution, a mesh metric field is defined with the help of the Hessian and is called a *mesh metric tensor*. This tensor prescribes the desired anisotropy at a point [13]. It is defined as a symmetric positive definite matrix M, whose associated quadratic form $< X, MX >$, defines a mapping of an ellipsoid in the physical space into a sphere in the metric space, meaning that any element in the physical space maps to a similar element in the metric space with equilateral sides. In this sense it can be seen as a transformation matrix. Figure 2.1 shows the idea behind the transformation.

The goal of the adaptive meshing algorithm is to output a mesh with regular elements in the metric space, where each mesh edge e must satisfy:

$$< e, Me >= 1 \tag{2.2}$$

In practice, this criteria is relaxed by the mesh adaptation software to ensure that a mesh can be adapted.

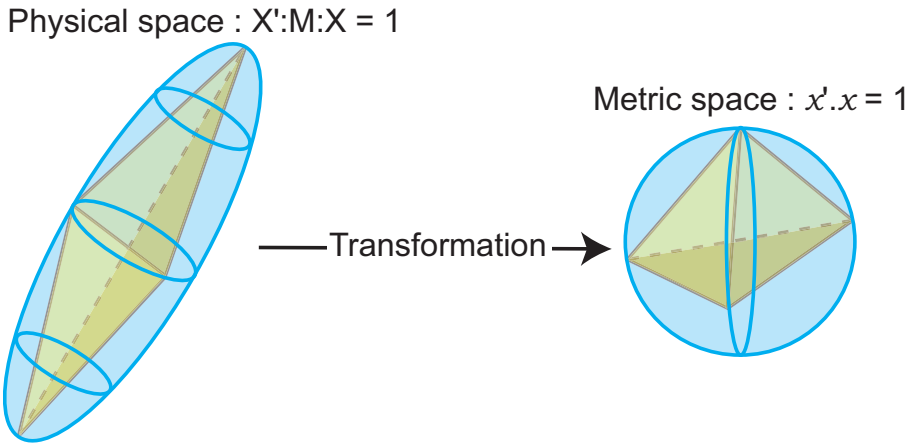


Figure 2.1: Transformation associated with a mesh metric tensor [60]

The mathematical form of this transformation matrix is given by Eq. 2.3:

$$\mathbf{T(x,y,z)} = \begin{bmatrix} 1/h_1 & 0 & 0 \\ 0 & 1/h_2 & 0 \\ 0 & 0 & 1/h_3 \end{bmatrix} \begin{bmatrix} \vec{e_1} \\ \vec{e_2} \\ \vec{e_3} \end{bmatrix} \tag{2.3}$$

where $\vec{e_1}, \vec{e_2}, \vec{e_3}$ are the unit vectors in the three principal directions and h_1, h_2, h_3 are the desired mesh edge lengths (mesh sizes) in the corresponding directions.

2.1.2 Combining the Hessian with Other Error Estimators

Though the Hessian is very useful in extracting the directional information from the flow solution, it is not always accurate. For example, an extrapolation technique is used near the walls to project the interior values of the gradients on to the nodes that lie on the domain boundary. When using the Hessian of speed, it is seen that the near wall resolution requests are often very tight and can introduce excessive refinement on the walls.

To remedy these problems, some *a posteriori* error estimators/indicators can be used in combination with the Hessian to drive the adaptation. Vendetti et al. [78] developed an approach, wherein the Mach number Hessian sets the anisotropy of mesh elements and the adjoint adaptation parameter dictates the tightest mesh spacing. For finite elements, calculating the adjoint adaptation

parameter requires solving another system of PDEs, to complete the dual problem. This is very expensive in terms of computational resources.

Since we solve Navier-Stokes equations which are PDEs, we can naturally get the PDE residuals from the solution process for the equations being solved. Root mean squares of the solution fluctuations can also be readily obtained from the solver. These quantities have been traditionally used as the scalar error estimators/indicators to drive isotropic adaptivity, which gives equilateral elements. We have extended the approach such that the Hessians provide the relative scales and the directional information for adaptation, but the smallest mesh size is calculated from the traditional isotropic error indicators. These error indicators are assigned relative weights and are combined to give the final scalar indicator, which drives the smallest mesh size during adaptivity.

The above approach has a distinct advantage in aerodynamic flows. Usually, Hessians of speed or pressure are not very strong in the wake. With velocity Hessians, most of the adaptivity is concentrated on the walls and with pressure Hessians, stagnation points get a lot of refinement. Due to this, the wake region usually remains under resolved when adapted with Hessians. PDE residuals are usually strong in the wake region and can give a better estimate of the error in this region. Using a combination of the Hessians and the scalar error indicators ensures that none of the anisotropy is lost and yet important weaker solution features in aerodynamic flows such as the wake and the tip region, get their due refinement.

2.1.3 Extension to Boundary Layers

The methodology described above is applicable for unstructured elements. However, boundary layer meshes are structured and graded layers of elements near the no slip walls. During adaptation, it is highly desirable to maintain this advantageous structure. If the anisotropic adaptation procedure described above is applied naively, it can destroy the structure of the boundary layer mesh. Sahni et al. [61] extend the anisotropic adaptation technique to boundary layers, which combines the strengths of both pre-defined boundary layer meshing and fully unstructured adaptation.

In reference [61], the boundary layer mesh definition is decomposed into a layer surface (2D) and a thickness (1D) shown in Figure 2.2. The triangular surfaces at top of each boundary layer stack are called as *layer surfaces*, while the lines which are roughly orthogonal to the wall are referred to as the *growth curves*. The mesh sizes on the layer surfaces are referred to as the *in-plane* sizes and the ones on the growth curves are referred to as the *normal* spacings. Several layered elements form a boundary layer stack and the top of the boundary layer stack interfaces with the unstructured part of the mesh.

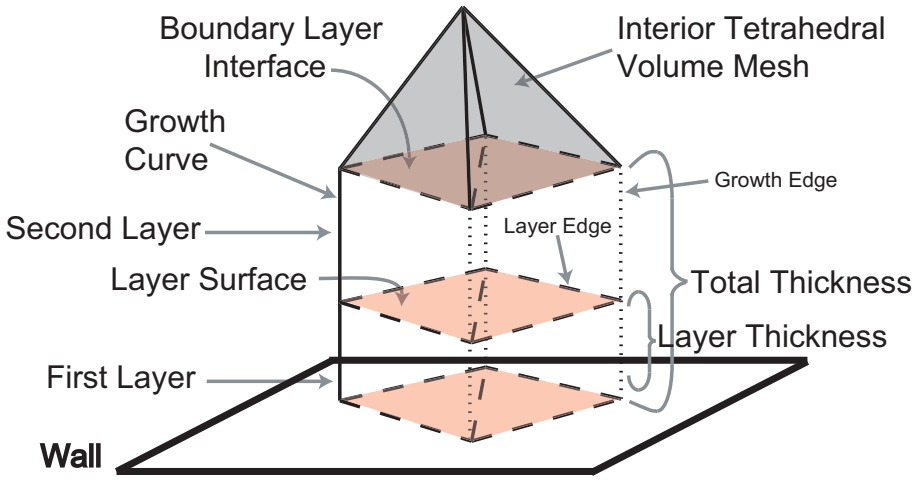


Figure 2.2: Conceptual decomposition of a boundary layer mesh [61]

Following this decomposition, the ellipsoid in Figure 2.1 is also decomposed as an ellipse that lies on the layer surfaces and a normal component aligned with the thickness of boundary layers as seen in Figure 2.3.

To preserve the layered structure of the boundary layer elements, mesh modifications are divided in two steps, in-plane adaptation and thickness adjustment. At any point in the boundary layer, the mesh metric ellipsoid is viewed as a combination of an ellipse residing on the layer surface and a normal component aligned with the growth curves. In-plane adaptation affects the mesh resolution on the layer surfaces and thickness adjustment takes care of the resolution in the normal direction.

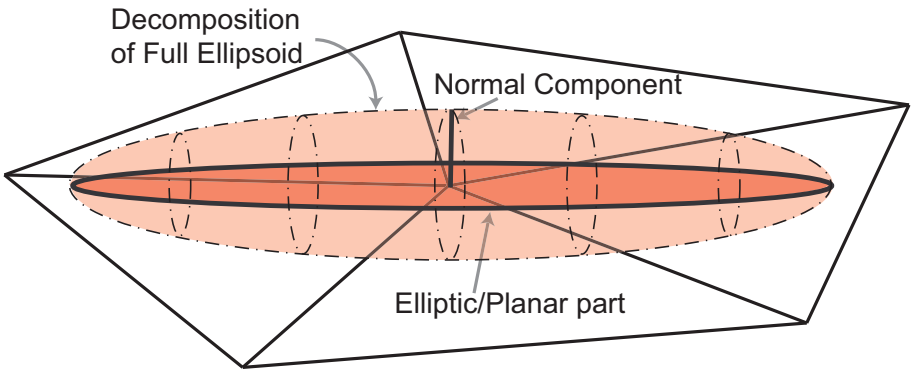


Figure 2.3: Conceptual decomposition of the ellipsoid associated with the mesh metric tensor [61]

2.2 Size Field Smoothing

Many times, coarse solutions develop non-smooth size field requests, especially in the boundary layer. A potentially unstable situation can occur if the mesh adaptation accurately adapts to such requests, since many high fidelity solvers produce oscillatory results in such cases. This issue, already challenging for isotropic size fields, can be even more problematic for anisotropic size fields owing to the tensor representation of the field which may oscillate not only in value but also in direction.. The common solution of smoothing the size field must be accomplished with greater care for anisotropic size fields. If applied naively, the desired anisotropy can be easily destroyed, or reduced much more than desired.

The mesh sizes calculated near the walls from the Hessians are usually a magnitude lower than that in the unstructured region of the mesh. As a result, the boundary layer stack is refined much more than the adjacent unstructured mesh, leading to so called "spider elements" and poorly shaped elements, locally. To remedy this, we apply separate smoothing techniques to eigenvalues (which provide the mesh spacings), and the eigenvectors (three principal directions) of the mesh metric.

There are two challenges in smoothing the principal directions; the orthogonality between the vectors needs to be maintained, and the direction with smallest mesh size for the wall vertices needs to be perpendicular to the wall. Applying the general smoothing directly to each compo-

nent of the directions do not guarantee these criteria automatically. Algorithm 1 describes an alternate smoothing strategy for the mesh size field developed in this work that better maintains perpendicular wall vectors.

Algorithm 1: Algorithm for smoothing the mesh size field

1. Decompose the mesh size field in mesh sizes and corresponding principal directions.
2. For each owned vertex on each part of the mesh, collect mesh sizes from the neighboring vertices to form a sum for each mesh size (3 mesh sizes in total).
3. Communicate between parts to combine the sums on the part boundaries to form a complete sum for each vertex on the part boundaries.
4. Form an average by dividing the sum by number of vertices contributing to the average to get the final smoothed mesh sizes.
5. Smooth each component of the principal directions to form an average using neighboring vertices with communications on part boundaries similar to the mesh sizes.
6. For all the wall nodes, check if the direction for the smallest size is perpendicular to the wall, if not, orient the vector in wall normal direction.
7. Check if all the vectors are orthogonal to each other, if not, fix the vector for the smallest size as base and orient the other two to be orthogonal to it and to each other.
8. Normalize the vectors to get unit vectors for the final smoothed principal directions.
9. Combine the mesh sizes and the principal directions to form the mesh size field.

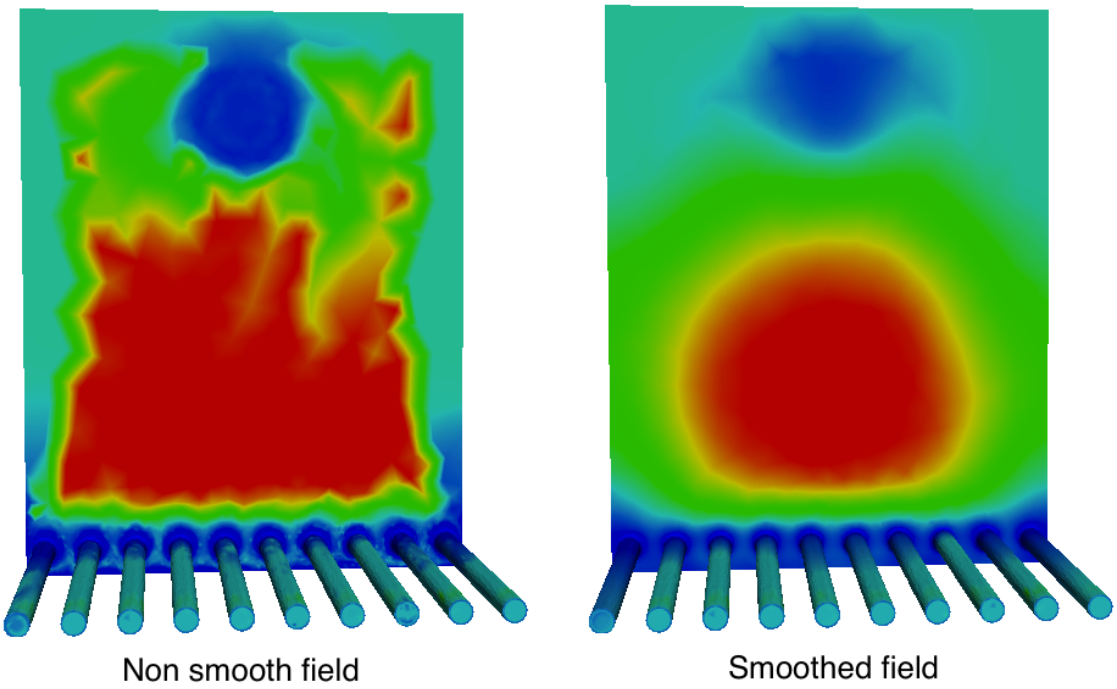


Figure 2.4: Effect of smoothing on mesh sizes of a manifold case

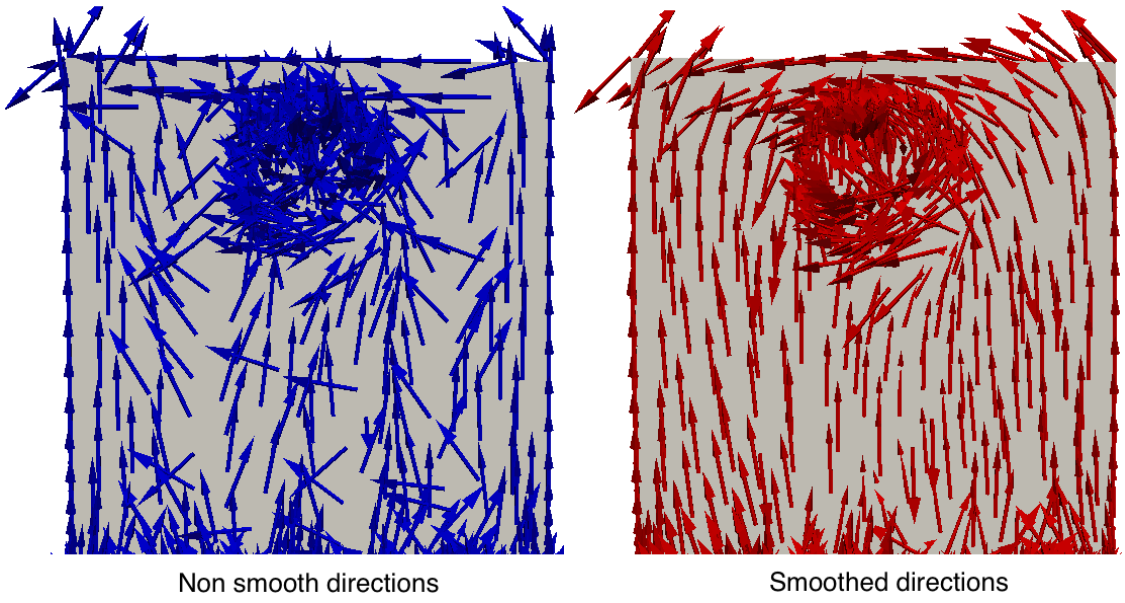


Figure 2.5: Effect of smoothing on principal directions of adaptivity

Figure 2.4 shows mesh sizes for a heat transfer manifold case (case details in Chapter 6). The smoothing of the mesh sizes can be easily seen from the pictures. The mesh adaptation software has lesser difficulty in adapting to smoothed size fields than non-smooth fields and this results in better meshes in terms of smooth transitions between elements. Smoothing of the principal directions can be seen in Figure 2.5. Before smoothing, the directions are not oriented in a well defined manner. The changes in the directions for adjacent elements is considerable and can produce elements with abruptly changing mesh sizes ("patchy adaptivity"). After smoothing, the eigenvectors become more regularly oriented in similar directions, which is simpler for the adaptation procedures to handle and gets rid of the "patchiness".

Chapter 3

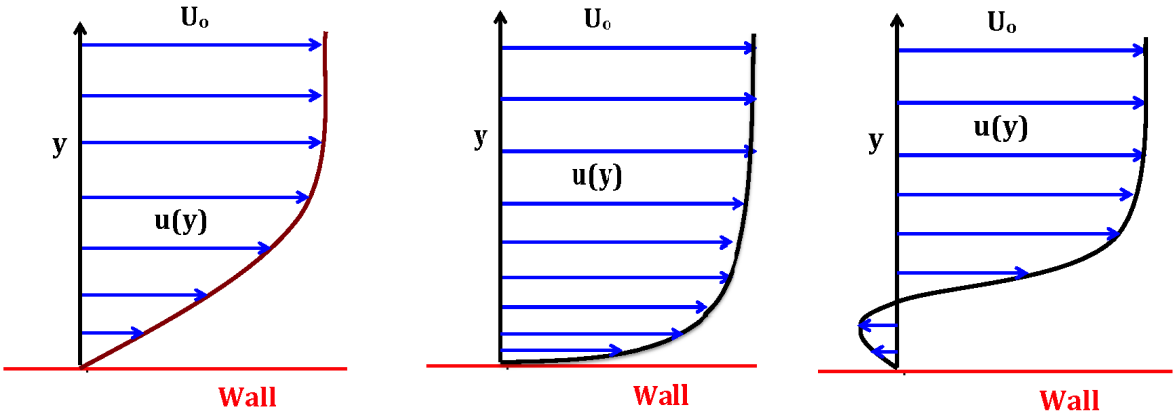
Boundary Layer Thickness Adaptation

This chapter discusses a new hybrid strategy to control the important attributes of the boundary layer meshes, in the thickness direction. The new hybrid strategy uses physical information to guide the calculations of normal spacings of the boundary layer whereas the Hessian bases error indicators drive the in-plane adaptivity [17].

3.1 Types of Boundary Layers

In 1904, Prandtl showed that the flow past a body can be divided into two regions: a very thin layer close to the body where viscosity is important, and the remaining region outside this layer where viscosity's effects are negligible [64]. This effect of viscosity inside the boundary layer accounts for much of the friction drag on the body and, if the flow separates, that separation location can strongly influence the form drag as well. The edge of the velocity boundary layer is defined by the point where the velocity is 99% of the free-stream velocity (δ_{99}) [64].

At this point it is important to note that the control of boundary layer mesh must be different for laminar and turbulent boundary layers. This thesis focuses on turbulent boundary layers as they are the most prevalent type for high Reynolds number flows and are more complex in nature. Also, the mesh spacing requirements to resolve the turbulent boundary layers are much tighter than that for the laminar boundary layers, due to much larger velocity gradients near the wall (Figure 3.1). Thus our treatment for turbulent boundary layers is sufficient for resolving laminar boundary layers, although it may not be optimal for this case.



(a) Laminar BL (b) Turbulent BL (c) Separated BL

Figure 3.1: Types of boundary layer profiles

The adaptive control of boundary layers must also differentiate between attached and separated boundary layers. In many flows, due to adverse pressure gradients and/or sharp turns and corners, boundary layers separate from the wall and form a free shear layer. The treatment of these boundary layers needs special care as the flow physics in this region is entirely different than that of the attached boundary layers. Figure 3.1 shows the two types of attached boundary layer profiles and a typical separated boundary layer profile.

3.2 Turbulent Boundary Layers

Figure 3.2 shows a typical turbulence boundary layer profile plotted as $u^+ = u/u_\tau$ vs. $y^+ = yu_\tau/\nu$ on a semi-log scale, where u_τ is the friction velocity. The turbulent boundary layer profile consists of 2 distinct regions; a *viscous sub layer* very close to the wall, where u^+ roughly equal y^+ and extends till about $y^+ = 5$ and a *log region*, away from the wall typically beyond $y^+ > 30$, where u^+ follows a log law. The region in between is called *buffer region*, which blends the two profiles smoothly.

Turbulent flows are usually simulated using one of the popular turbulence models for high Reynolds numbers and the mesh resolution requirements to resolve the turbulent boundary layers are well known for each of these turbulence modeling approaches. The Reynolds averaged Navier-

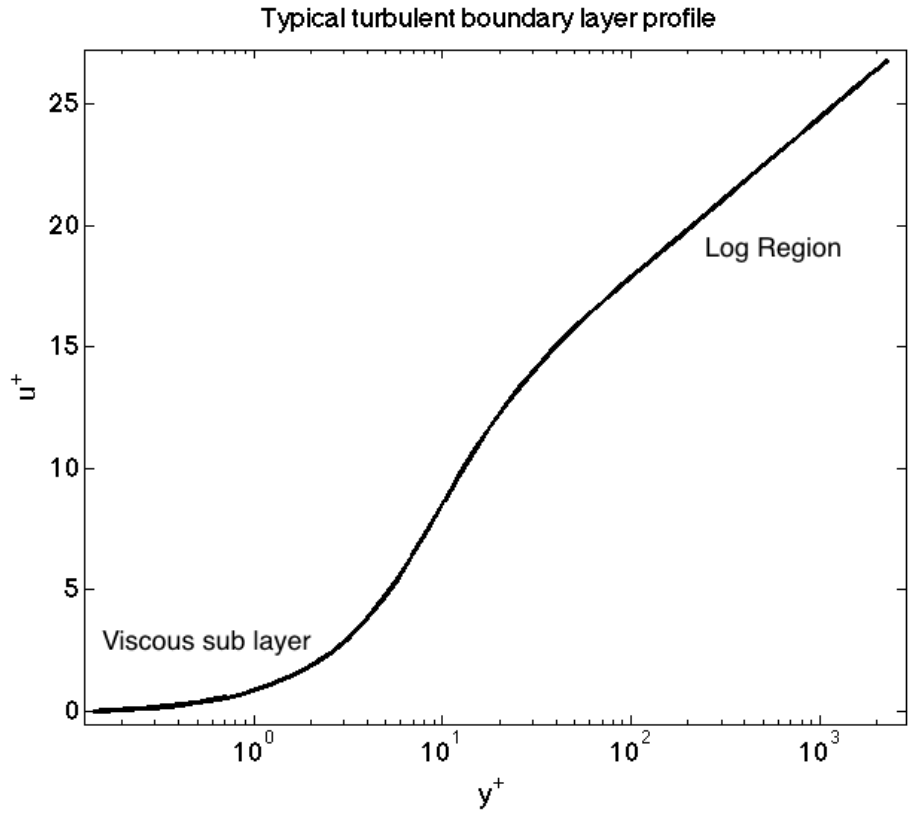


Figure 3.2: A typical turbulent boundary layer profile

Stokes (RANS) models are very popular and are widely used in industry and academia. Different turbulence models associated with RANS were developed in last three decades; like the Spalart Allmaras (SA) one equation model [71] and SST Menter two equation model [51]. Other popular modeling approaches include large eddy simulation (LES) [32], [67], [76] and detached eddy simulation (DES) [72], [73]. Recently, with advances in parallel computing and petaflop computing power, direct numerical simulation (DNS) [52] is being more regularly applied, however, it is still constrained to lower Reynolds number flows compared to other modeling approaches.

For turbulent boundary layers, the mesh resolution near the no-slip walls is crucial for correctly capturing the boundary layer profile. Different modeling approaches mentioned above have varying requirements for this spacing. These spacings are defined with a non-dimensional parameter called y^+, e.g., RANS-SA model with wall resolved treatment requires that the first cell height in the boundary layer should lie between y^+ of 1 to 5. If wall models are used with the turbulence

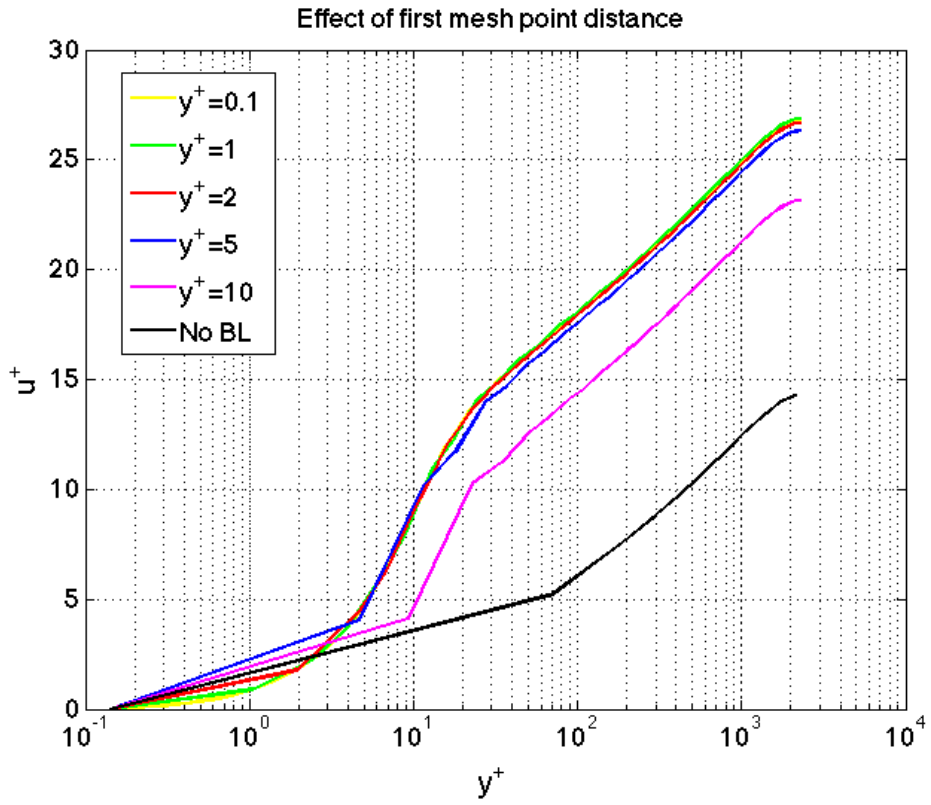


Figure 3.3: Effect of the first cell height on turbulent boundary layer profile (RANS-SA turbulence model)

model, y^+ can be increased to 30-50. If these requirements are not satisfied, then large numerical errors are introduced in the solution.

Figure 3.3 shows the variation in the turbulent boundary layer profile of a turbulent pipe flow with changing first cell height of the mesh boundary layer, for RANS-SA wall resolved approach. For all the first cell heights below y^+ of 5, the boundary layer profiles are well predicted and agree with each other. As y^+ is increased beyond a value of 5, the boundary layer profile starts diverging from other results, showing a wrong behavior, with the worst profile being obtained with no boundary layer at all. This signifies the need for the adaptive control of boundary layers in the thickness direction to avoid getting bad results by controlling the attributes of the mesh boundary layers.

3.3 Important Attributes of the Mesh Boundary Layers

The geometric description of the mesh within a boundary layer can be defined by following parameters:

1. First cell height of the mesh boundary layer (t_o)
2. Total height of the mesh boundary layer (T)
3. Total number of layered elements in the boundary layer (n)
4. Growth factor (r)

The interdependence of these parameters is given by following equation:

$$T = t_o \sum_{i=1}^{i=n} r^{(i-1)} \tag{3.1}$$

which indicates that the height of each succeeding layer in the boundary layer mesh, away from the wall, increases by the growth factor times the height of the preceding layer.

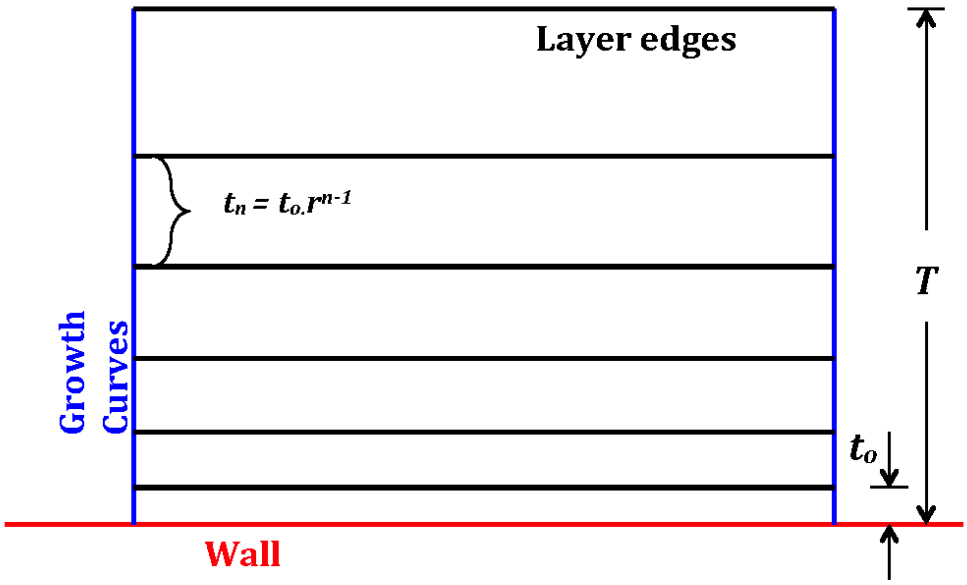


Figure 3.4: Attributes of the mesh boundary layers

It follows that three of these parameters can be chosen independently, and fourth one is set by Eq. 3.1. The total height of the boundary layer (T) and the first cell height (t_o) are typically

viewed as the most important parameters in terms of the flow physics and are usually controlled by the user. The third independent parameter usually should be the growth factor (r), but keeping in mind the total number of elements (n) these choices would create.

Usually, these parameters are chosen on an *ad hoc* basis, by the user, or with some knowledge of the initial/free-stream conditions and the Reynolds number. The mesh is generated and a solution is obtained and analyzed. If the flow solution is not satisfactory, then mesh is regenerated using a different set of values. This obviously is very cumbersome and time consuming. Moreover, the values of boundary layer parameters are usually chosen as constants throughout the domain. However, the flow solution is not constant, and the variables which dictate these parameter values (like wall shear stress), vary throughout the domain. Therefore, an automated procedure for selection of these parameters and adapting the mesh boundary layer throughout the domain, based on the flow solution, is highly desirable.

Next, we will describe the calculations of key physical quantities such as wall shear stress which is used to determine the thickness attributes of the boundary layers.

3.4 Calculation of the Wall Shear Stress

The near wall mesh spacing requirements for turbulence models depend on y^+ and needs the knowledge of the friction velocity u_τ, which can be calculated from the wall shear stress τ_w as $u_\tau = \sqrt{\frac{\tau_w}{\rho}}$. Many solvers readily provide wall shear stress as a field after post processing. However, we use alternative methods to calculate this field in a fast and an efficient manner.

Since most of the boundary layers in high Reynolds number flows are turbulent, one method to calculate τ_w for attached boundary layers is by using Spalding's law [74] which gives y^+ as a function of u^+, written in a special form given by:

$$y^+ = f(u^+) = u^+ + A[e^{(\kappa u^+)} - 1 - (\kappa u^+)^2/2 - (\kappa u^+)^3/6 - (\kappa u^+)^4/24] \tag{3.2}$$

where $A = 0.1108$ and $\kappa = 0.4$ are dimensionless constants. This law is valid through the entire turbulent boundary layer profile when it is attached and in equilibrium. u^+ is the dimensionless

velocity at points along the boundary layer, normalized by the friction velocity (u/u_τ).

Using Eq. 3.2, u^+ can be calculated at various points along the boundary layer, with following iterative approach.

(1) Calculate the distance of the point on the boundary layer from the wall: Δy.

(2) Retrieve the velocity at this point from the flow solution: u.

(3) Guess an initial value of u_τ^1 and calculate initial $u^+ = u/u_\tau^1$ and $y^+ = \Delta y u_\tau^1$.

(4) Use Newton's method to iteratively solve Eq. 3.2 till convergence and update value of u_τ at each iteration.

(5) Use the final value of u_τ to calculate final value of u^+ (u/u_τ).

This process can be repeated at any number of points along the boundary layer. In the end, a simple average over the number of points used gives the final friction velocity u_τ that is less prone to solution error, particularly on the initial, coarse grids where the *a priori* guessed resolution is far from correct. Note that Eq. 3.2 is often used as a "wall model" and thus, in addition to smoothing out the shear stress, we are effectively using a wall model to determine the resolved wall resolution. We use 3 to 5 points on an average along the growth curves to calculate u_τ. The wall shear stress can then be calculated as $\tau_w = u_\tau^2 \rho$.

Another method to calculate an approximate estimate of the wall shear stress, is using a finite difference approach near the wall. Using the first vertex from the wall and known u and Δy values at that point, τ_w can be calculated using $\tau_w = \mu \frac{du}{dy} = \mu \frac{u}{\Delta y}$. Here du equals u because the velocity is zero at the walls. This alternative method could be used everywhere but we have found it less robust than Spalding's law based procedure and we therefore use it only for flow regions where the boundary layer is not attached to the wall, where Spalding's law is not valid.

Figure 3.5 shows the wall shear stress calculation using the method described above for a flat plate turbulent boundary layer. The calculations give similar behavior to the analytical results obtained with $1/7^{\text{th}}$ law. The percentage error between the analytical and the computed results is

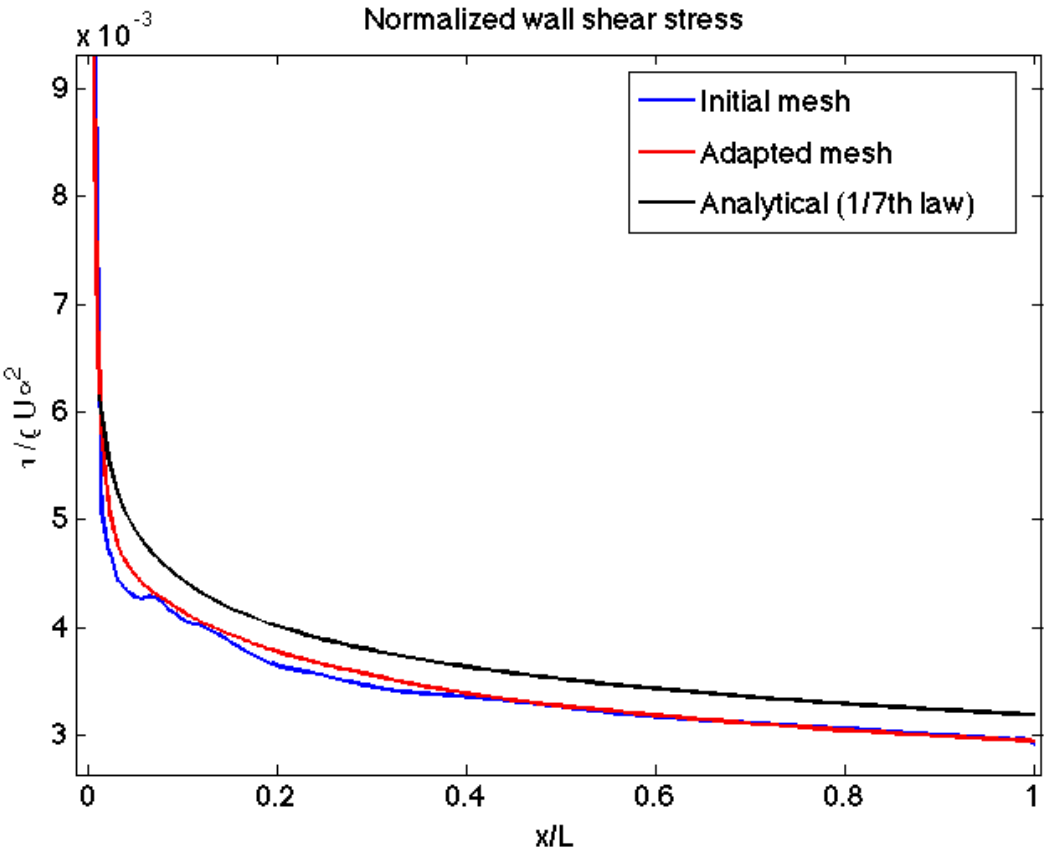


Figure 3.5: Wall shear stress calculations for a flat plate using Spalding's law

less in terms of the first cell height estimates given by both these methods. Note that when using the wall shear stress to calculate the first cell height, we have a margin of y^+ of 1 to 5 for RANS-SA wall resolved approach and similar margins for other turbulence modeling approaches.

3.5 Thickness Adaptation

The boundary layer adaptation procedure used in this thesis uses the decomposed Hessian-based method for the in-plane adaptation and the method given in this section for the thickness adaptation. Since the flow physics plays a vital role in determining thickness attributes of the boundary layer, we base our approach on the physical information that can be derived from the flow solution which is described next. All these attributes are defined on the wall vertices which are used by the adaptation software to adjust the thickness of the mesh boundary layers growing

from the wall vertices.

3.5.1 First Cell Height (t_o)

As already pointed out, different turbulence modeling approaches have different mesh spacing requirements close to the walls. Even in the same family of turbulence models such as RANS, different approaches require varying mesh spacings depending on if the boundary layer is integrated to the wall (wall resolved approach) or if wall functions are used (wall modeling approach). The wall resolved approach makes a low Reynolds number assumption near the walls and requires the first cell height inside the viscous sub layer of the boundary layer ($y^+ < 5$). The wall modeling approach makes suitable assumptions for near wall behavior of the boundary layer and requires that the first cell height is beyond the viscous sub layer and into the log layer ($y^+ > 30$). If these requirements are not met for either of these modeling classes, then large numerical errors are possible in turbulence calculations predicting erroneous behavior, as seen in Figure 3.3. However, the friction velocity, which is required to calculate y^+, is not known *a priori*. This makes adaptive control of the first cell height, based on the flow solution, critical.

Let us assume that the turbulence model requires the first cell height to be equal to y_1^+. If we have an initial coarse mesh with a solution, using the wall shear stress (τ_w), t_o can be calculated, by the following algorithm:

(1) Get the kinematic viscosity (ν) and desired y_1^+ according to the turbulence model being used. (Suggested values are $y_1^+ = 1$ for wall resolved RANS-SA, $y_1^+ = 30 - 50$ for wall modeled $k - \epsilon$, 0.5 for wall resolved $k - \epsilon$ etc.)

(2) Calculate the friction velocity u_τ as discussed in the previous section.

(3) Calculate the first layer height of the boundary layer by $t_o = \nu y_1^+ / u_\tau$.

Figure 3.6 shows calculations of the first cell height of the mesh boundary layer, for a turbulent flat plate. The first cell height is requested at $y^+ = 1$. The analytical approach is based on the wall

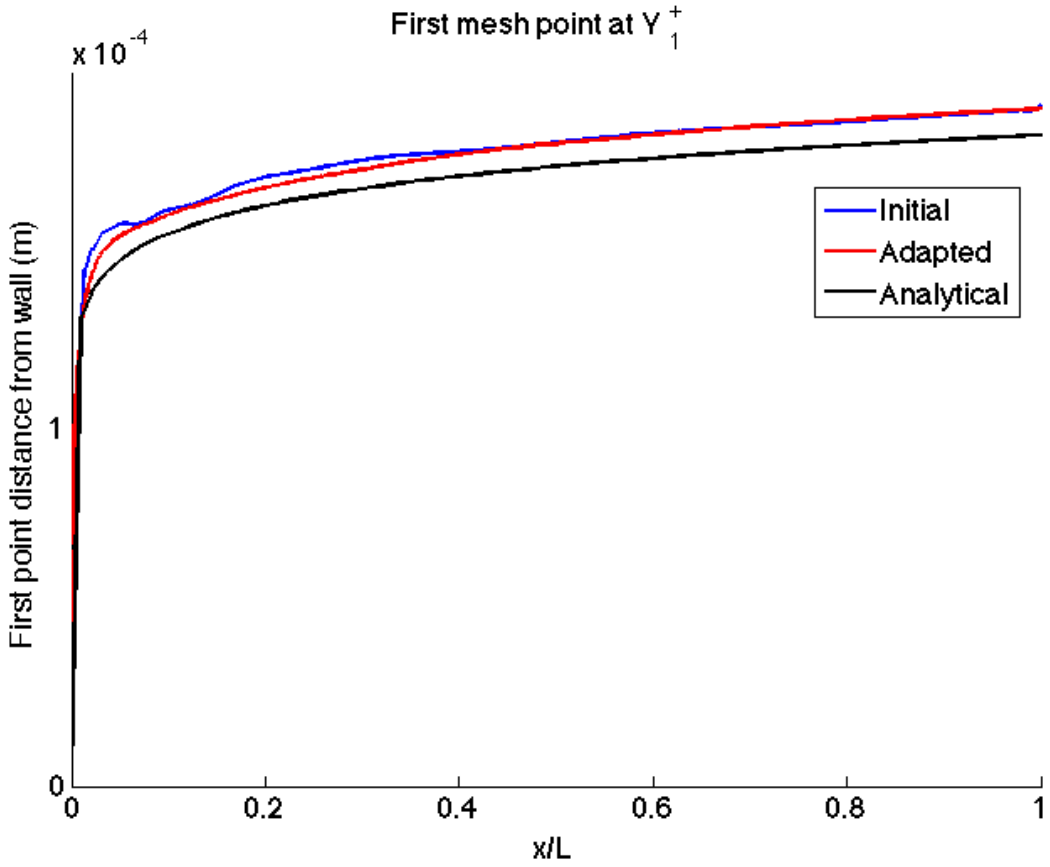


Figure 3.6: Estimates of the first cell height of the mesh boundary layer for a flat plate turbulent boundary layer

shear stress from Figure 3.5. The calculated values for initial and adapted meshes show similar behavior as the analytical values. The maximum difference between the two is 5e-6 m, which is less than a 5% difference and is well within the permissible limits of our margin of $y^+ = 1 - 5$.

3.5.2 Total Height of the Boundary Layer (T)

It is desirable to have the total height of the mesh boundary layers equal to or greater than the velocity boundary layer height given by δ_{99}. δ_{99} is the distance from the wall at which the velocity becomes 99% of the free stream velocity, and is an accepted measure of the boundary layer thickness. It is usually tricky to calculate δ_{99} directly as it requires knowledge of a reference velocity. For simple problems like a flat plate, the reference velocity is usually the free stream velocity, but it can have different local values for more complex problems, where the flow as a whole undergoes

acceleration or deceleration. This presents a difficulty in directly calculating the boundary layer height.

To calculate T, we base our approach on the observation that vorticity outside of an attached boundary layer is negligible. Since the boundary layers have the largest velocity gradients very close to the wall, vorticity here is the highest and decreases as one moves farther away from the wall. As boundary layer growth curves are essentially perpendicular to the wall, one can walk along these edges starting from the wall, and determine the point at which the vorticity drops below a threshold value. This threshold value depends on the local maximum value of vorticity for attached boundary layers, which is most often encountered at the wall. Through our analysis, we have found that a good value for the threshold is 0.02% of the wall vorticity magnitude.

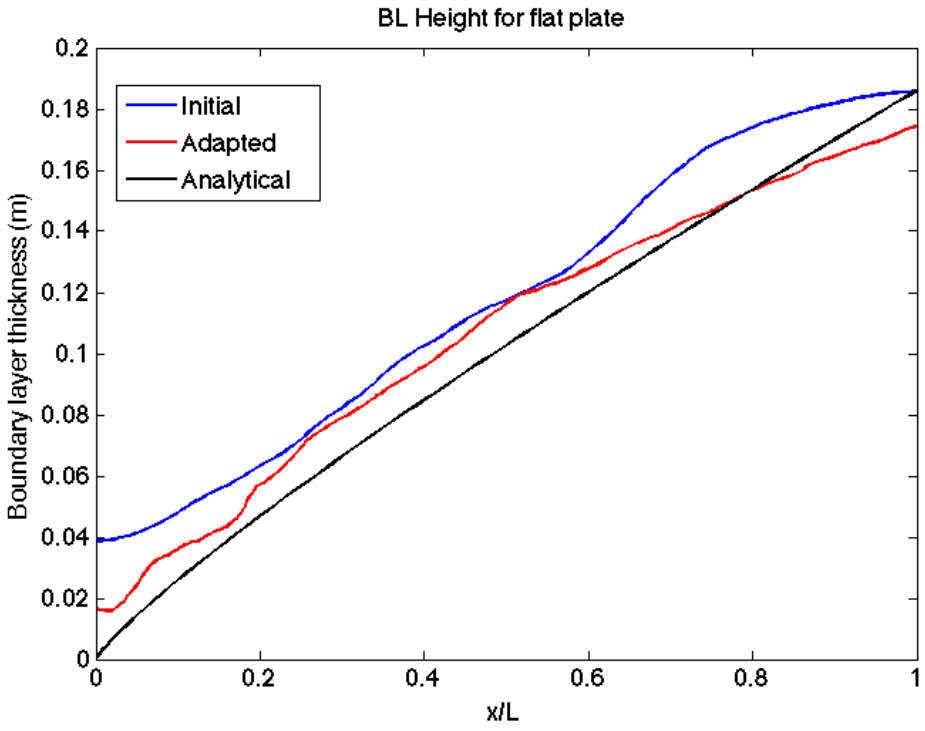


Figure 3.7: Total Height of the velocity boundary layer for a turbulent flat plate

Figure 3.7 shows calculations of the boundary layer height using vorticity method for a turbulent flat plate. The values calculated for the initial and the adapted meshes are close to those predicted by the analytical law which states that δ_{99} varies as $x^{4/5}$. The adapted mesh predicts a

closer agreement with analytical values, indicating that it might take a couple of adaptation loops to get to the right behavior.

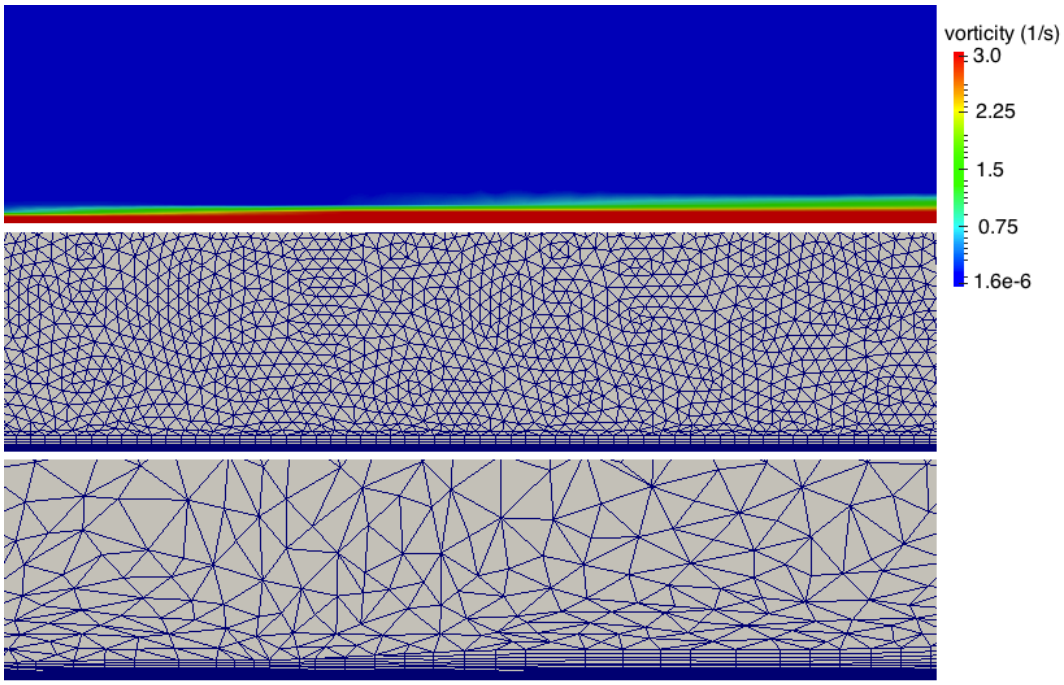


Figure 3.8: Vorticity magnitude, initial and adapted meshes for a flat plate

Figure 3.8 shows the vorticity magnitude, the initial mesh and the adapted mesh for a turbulent flat plate. Thickness adaptation of the mesh can be clearly seen as the total height of the mesh boundary layers increases in the streamwise direction, closely following the vorticity magnitude.

To show the importance of adaptivity with our technique, Figure 3.9 shows the boundary layer profiles for a coarse initial mesh (with first cell height at $y^+ = 20$), an adapted mesh (with first cell height at $y^+ = 1$) and experimental data by Andersen [4] . The figure illustrates that we are able to adapt mesh boundary layer thickness of coarse meshes to get better boundary layer profiles that agree well with experiments, which is one of the aims of this thesis.

3.5.3 Growth Factor (r) and Total Number of Layers (n)

To increase the boundary layer elements' height away from the wall, a growth factor (also called as the stretching factor) greater than 1 is used. This is because the tightest mesh spacing

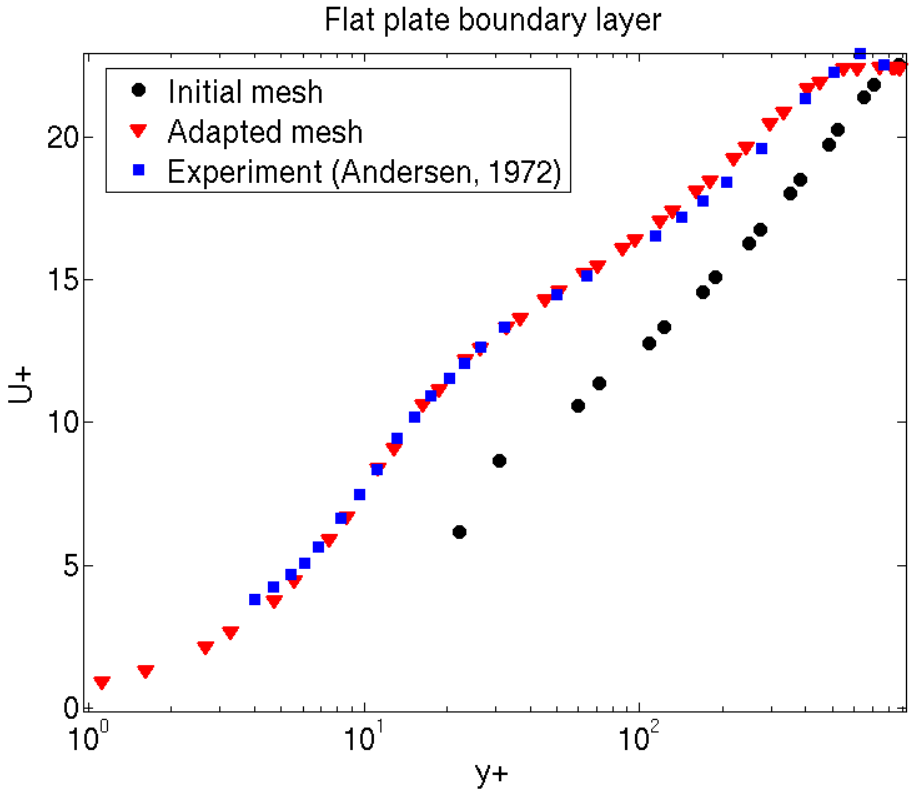


Figure 3.9: Turbulent flat plate boundary layer profile at Re_x of 1 million

is required very close to the wall, but this requirement is not as strict farther away from the wall. An ideal scenario would be to achieve height of the last layer equal to the unstructured sizes of the mesh and get a smooth transition. The mesh adaptation process that we use provides options like boundary layer gradation factor, which control the transition of boundary layer into the unstructured part of the mesh, smoothly [47].

There are general guidelines for what the ideal growth factor should be, from the perspective of turbulence modeling. Spalart [69, 70] states that the growth factor should be close to 1.25 to accurately capture the log layer. Generally a growth factor beyond the value of 1.4 is deemed too large for accurately capturing the boundary layers. Many meshing tools are based on setting t_o, T and n, and the growth factor is automatically calculated, internally. The accuracy then in turn hinges on the knowledge and prior calculations to make sure that the growth factor being calculated is acceptable.

The adaptation tool gives the ability to set the growth factor at each wall vertex. We set r in the range of 1.2 - 1.25 to be within the acceptable limits. Selecting a growth factor less than 1.2 has the disadvantage of creating more elements in the boundary layer than needed. The number of layers are then calculated using Eq. 3.1.

3.6 Adaptive Control for Separated Boundary Layers

The techniques described above for calculating the different aspects of the boundary layer meshes work well for attached boundary layers. However, separated boundary layers need extra care and special detection strategies due to different flow physics that must be captured.

3.6.1 Detection of Separated Boundary Layers

To treat separated boundary layers properly, they must first be detected. As it can be seen from Figure 3.1, they have a unique profile characterized by flow reversal. We again make use of the wall normal growth curves and walk along the growth edges to detect a change in the flow direction. If a change (usually more than 120^o) is detected across the profile, then the vertex on the wall in that layer is marked as *separated*; otherwise the boundary layer is treated as an attached boundary layer.

This method requires that the total height of the mesh boundary layers in this region at least exceeds the height at which the flow direction is reversed. This means that typically initial boundary layer meshes for such regions should be tall enough and the mesh close to the wall be fine enough to capture the flow reversal. Currently, we make sure that this criteria is satisfied through initial meshing, but an iterative adaptive procedure like the one we use eventually leads to suitable meshes which are able to capture this effect.

3.6.2 Calculation of the Wall Shear Stress

The method of using Spalding's law to calculate the wall shear stress is not appropriate for separated boundary layers, since the typical turbulent boundary layer profile is absent. In locations

where separation is detected the wall shear stress is calculated using the finite difference method explained earlier. The accuracy of such calculations is not as good as other methods (Spalding's law), but it gives a reasonable estimate. Also, for separated boundary layers, the first cell height of the mesh boundary layer is not as crucial as for the attached boundary layers, hence such an approximate approach is justifiable.

3.6.3 Total Height of the Boundary Layer (T)

For separated boundary layers, the free shear layer might be separated from the wall to a fair distance, in which case it might not be prudent to increase the boundary layer height. For example, for an airfoil beyond stall, the flow separates near the leading edge on the suction side and forms a large recirculation region. It would be impractical to increase the mesh boundary layer height beyond this region as it would create poorly shaped elements near the interface. Even though it would be a good feature to separate the mesh boundary layer from the wall to resolve the free shear layer, this capacity is still under development with the current adaptive tools that we use.

The techniques explained above for attached boundary layers predict that the boundary layer's height should be increased to the height of the complete shear layer. However, this is not always practical for separated boundary layers as this height might introduce excessive stretching of the elements near the interface.

In a more practical approach, the boundary layer height is increased beyond the height at which the flow reversal is detected so that the boundary layer mesh is tall enough to "sense" the change in the flow direction and the rest of the boundary layer mesh is destroyed and the free shear layer region is meshed with unstructured elements, with specific care to resolve the free shear layer. This approach is used in this work. For anisotropic adaptivity, velocity Hessians give good resolution in these layers since the anisotropy of the top of the shear layer is not very high.

3.7 Shock Wave - Boundary Layer Interactions

In transonic and supersonic flows, the main region of interest is where the shock waves and the boundary layers interact. Shock waves are sharp discontinuities in the flow and can cause thickening of the boundary layer and even separation. Due to their complex nature, these regions require higher resolution. Since the in-plane mesh sizes for the shock wave - boundary layer interactions are smaller, care needs to be taken to trim away the boundary layer elements that develop anisotropy in the wrong direction as the in-plane resolution becomes finer than the normal direction resolution. One way to control this is by limiting the mesh boundary layer height.

When the flow encounters a shock, the solution changes sharply, giving high gradients in this region. We implemented a shock wave - boundary layer detection algorithm which detects a shock-boundary layer interaction by looking at the pressure gradients on the wall. If such a region is detected, then the wall vertices are marked to not increase the boundary layer height beyond a certain limit to avoid poorly shaped boundary layer elements. The boundary layer height in the shock region is limited by a factor of the upstream boundary layer height, so that it does not change abruptly with the shock. Such an abrupt increase in the boundary layer height can cause problems for the mesh adaptation procedures.

3.8 Smoothing of the Wall Fields

The quantities like wall shear stress which form the basis of our physics based adaptivity strategy, can be non-smooth. If the mesh adaptation is done for these non-smooth fields then undesirable output meshes can be obtained. To avoid this we have extended the smoothing strategy explained in Chapter 2 to wall quantities, such that only the vertices which lie on the walls, contribute to the average. This smoothing technique is applied to the wall shear stress (which controls the first cell height) and the total height of the boundary layer, which are the main parameters being controlled through thickness adaptivity. Algorithm 2 gives a brief outline of the procedure followed for the smoothing.

Algorithm 2: Algorithm for smoothing the wall fields

1. For each owned wall vertex on a part, gather the wall fields from the neighboring vertices that lie on the walls to create a sum.
3. Communicate between parts to combine the sums on the part boundaries to form a complete sum for each vertex on the part boundaries.
4. Form an average by dividing the sum by number of vertices contributing to the average to get the final smoothed wall field.

3.9 Roadmap for the Next Chapters

In the next chapters, we present and discuss results of adaptivity. The adaptivity is carried in two stages, one is *in-plane adaptation* in which the resolution on the layer surfaces is changed and other is *thickness adaptation* which changes the normal distribution of points of the boundary layers. The main contribution of this thesis is not the development of mesh adaptation procedures but the novel error indicators in boundary layers and their application to turbulent flows. Ideally, what we want is fully parallel adaptivity of mesh boundary layers in all directions (thickness and in-plane). However, this is currently limited by the available mesh adaptation software. Due to this, we group the results in different chapters according to the nature of adaptivity being carried out in each case.

In Chapter 4, we display the application results of the developments described in this chapter for thickness adjustment of the mesh boundary layers, along with in-plane adaptivity. These results were obtained on a single processor. Though the geometries in these studies are relatively simpler, the flow is still complex. In Chapter 5, we present serial in-plane only adaptivity (no thickness adaptation) results for very complex geometries like multi element wings. Even though our thickness calculation procedures work with these complex geometries, the adaptation softwares have not been able to handle these geometries. For the geometries like multi element wings, in-plane boundary layer adaptivity itself is fairly complex and has not been carried out previously, so these results are state of the art. For both of these chapters, the results were obtained in serial with the software provided by Simmetrix Inc. [34], which is a proprietary mesh adaptation software.

However, one thing to note is that the extent of boundary layer adaptivity that can be

carried out in serial is limited. As we start to create meshes with hundreds of million elements, for adaptivity, we want to work in parallel. In Chapter 6, we show results for parallel in-plane only adaptivity of boundary layers (no thickness adaptation) using an open source mesh adaptation software based on mesh adaptation procedures described in [21], [53], [61] developed at Rensselaer Polytechnic Institute in the Scientific Computation Research Center (SCOREC). Parallel thickness adaptation is still under development and will not be discussed in this thesis.

For all the application results in every chapter, the initial meshes were generated using the meshing software made available by Simmetrix Inc. [34].

Chapter 4

Application Results: Serial Thickness Adaptation

In previous chapters, we have outlined our overall strategies of mesh adaptation for boundary layer meshes. In this chapter we present application results of our newly developed physics based indicators discussed in Chapter 3 to drive thickness adaptation and their combination with other numerical error indicators. The mesh boundary layers are adapted in the in-plane direction as well as in the thickness direction, in serial. One incompressible and two transonic cases are studied and presented.

4.1 NACA 0012

The first application is a NACA 0012 airfoil, which is a 2D airfoil geometry. Reynolds number based on the chord is 6 million. The Mach number is 0.15 and the flow is modeled with an incompressible flow solver. Three different angle of attacks were studied and the initial mesh was adapted in each case with two adaptation loops. The thickness of the boundary layers was calculated with the methodology given in Chapter 3. A combined strategy of Hessians and scalar error indicators was used to drive adaptivity which is explained in Chapter 2. This choice was used because of the need to refine the wake of the flow which is characterized by high PDE residuals.

4.1.1 Angle of Attack: 0°

Figure 4.1 shows the comparison of our calculations for the boundary layer height with experiments conducted by Becker [10]. Our calculations show good agreement with the experiments

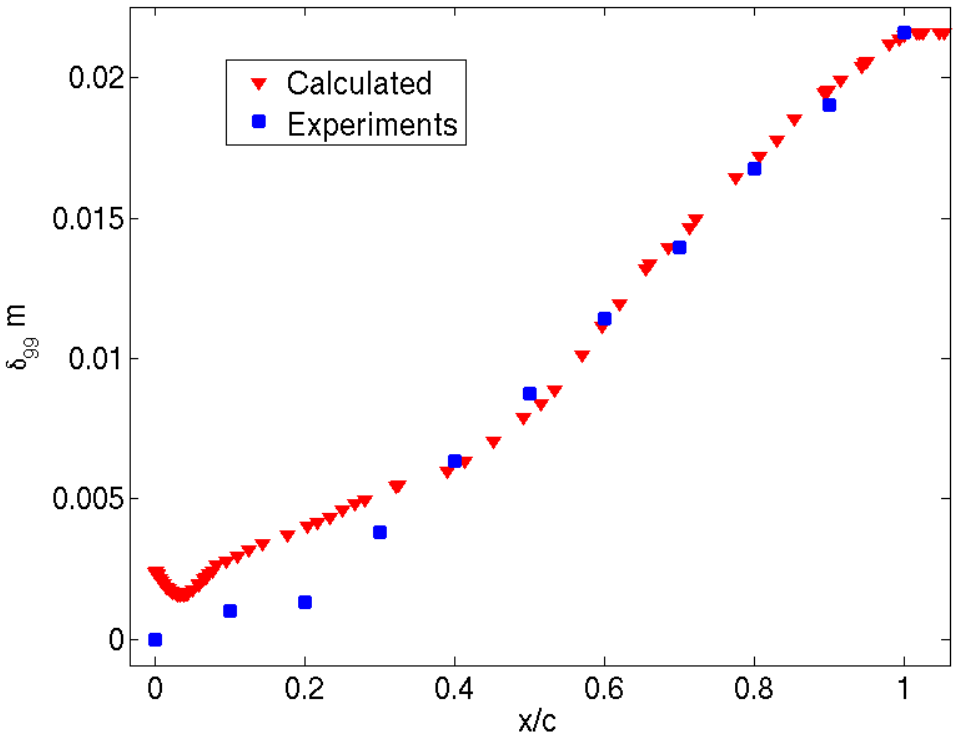


Figure 4.1: Comparison of boundary layer height calculations with experiments for NACA 0012 0° angle of attack

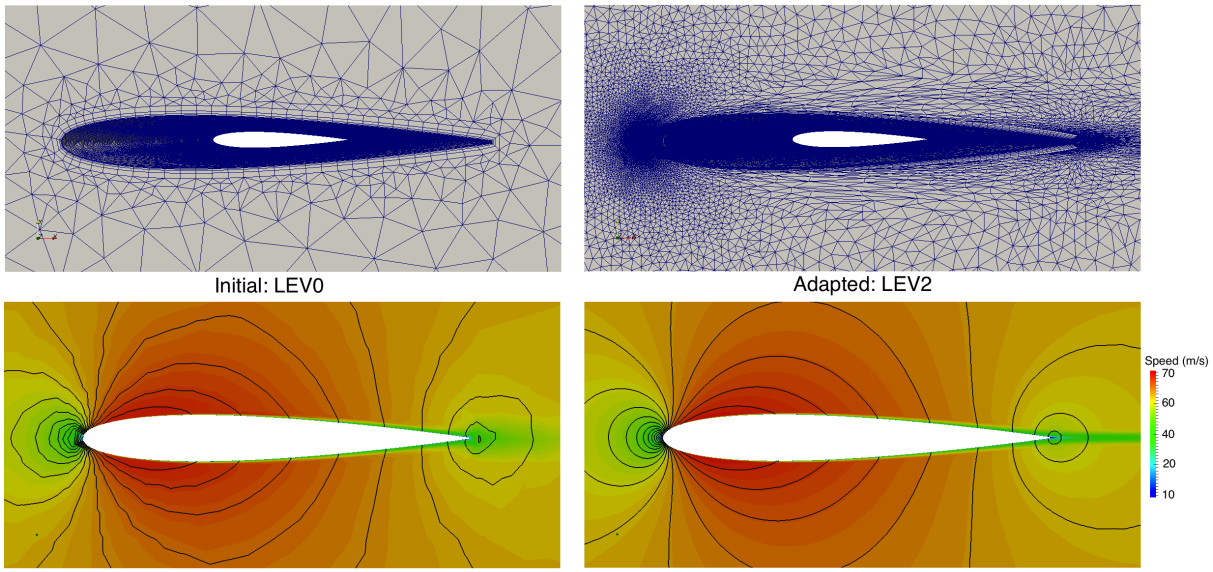


Figure 4.2: Cuts of the initial LEV0 and adapted LEV2 meshes, showing anisotropic adaptivity and speed distributions with pressure contours (black lines) for 0° angle of attack on NACA 0012

except at the start of the airfoil where the boundary layer is laminar. The calculations over predict the boundary layer height in the laminar region, which is not a bad thing because usually it is desirable to have the mesh boundary layer height greater than the physical boundary layer height.

Figure 4.2 shows the initial and the adapted meshes for the 0° angle of attack for NACA 0012. The pictures also show the speed distribution giving some indication of the height of the boundary layer and the pressure contours (black lines). The pressure contours become smoother with adaptivity. The adapted heights of the boundary layers on the upper and the lower surfaces of the wing are equal, as it should be for 0° angle of attack. The stagnation point and the wake receive more refinement than other areas. Anisotropic elements oriented with longer edges in the streamwise direction can be seen easily.

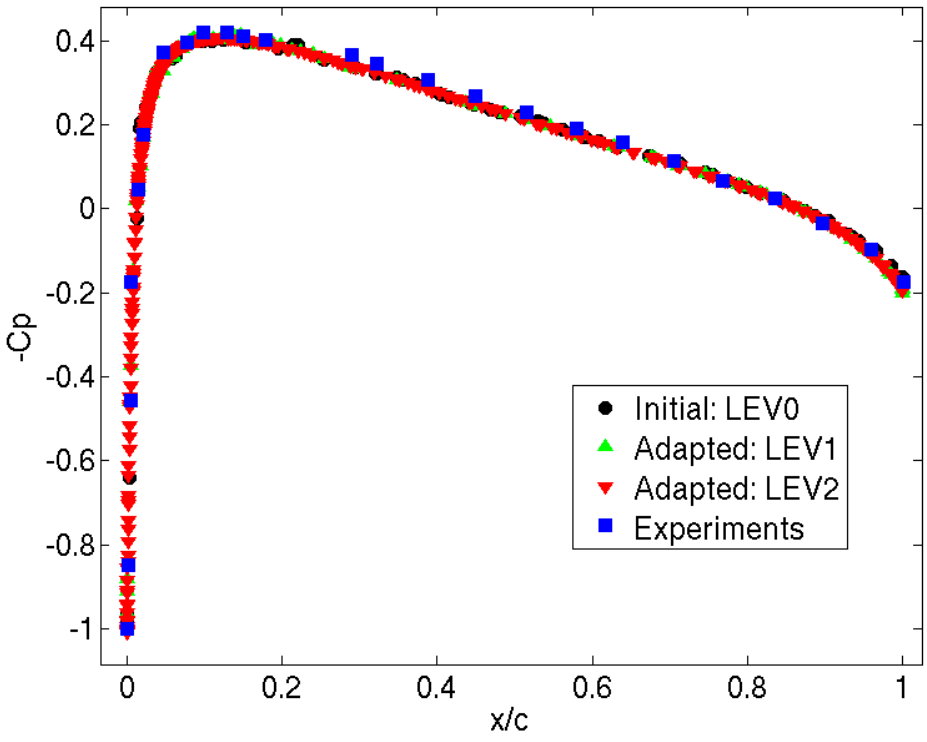


Figure 4.3: Coefficient of pressure plots for 0° angle of attack on NACA 0012

Figure 4.3 shows the coefficients of pressure on the airfoil. All meshes give good agreement with the experiments.

4.1.2 Angle of Attack: 5°

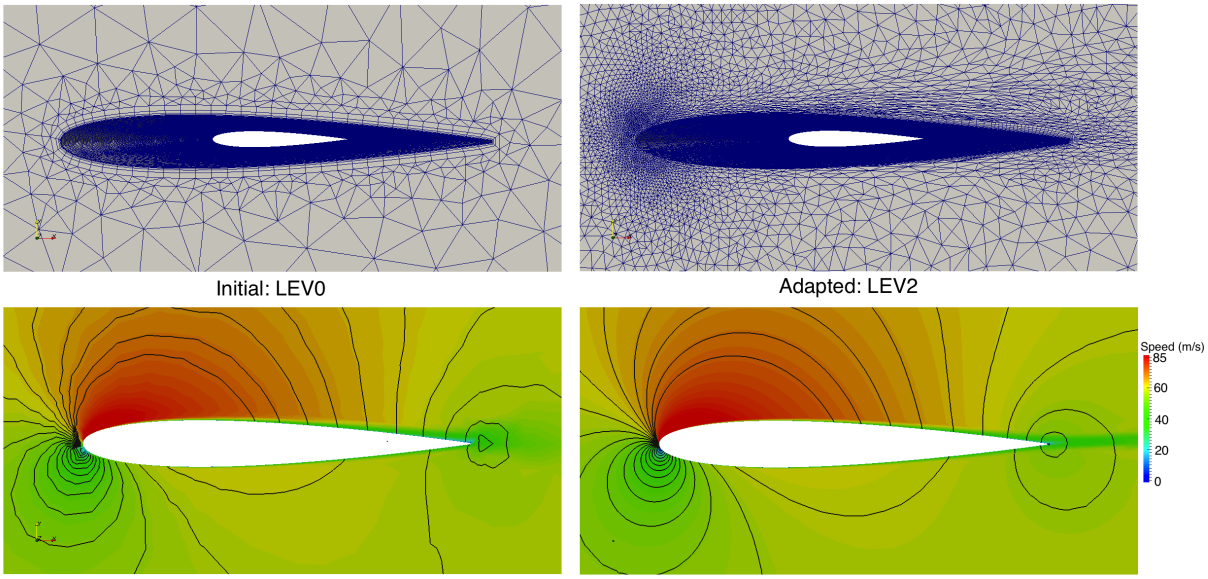


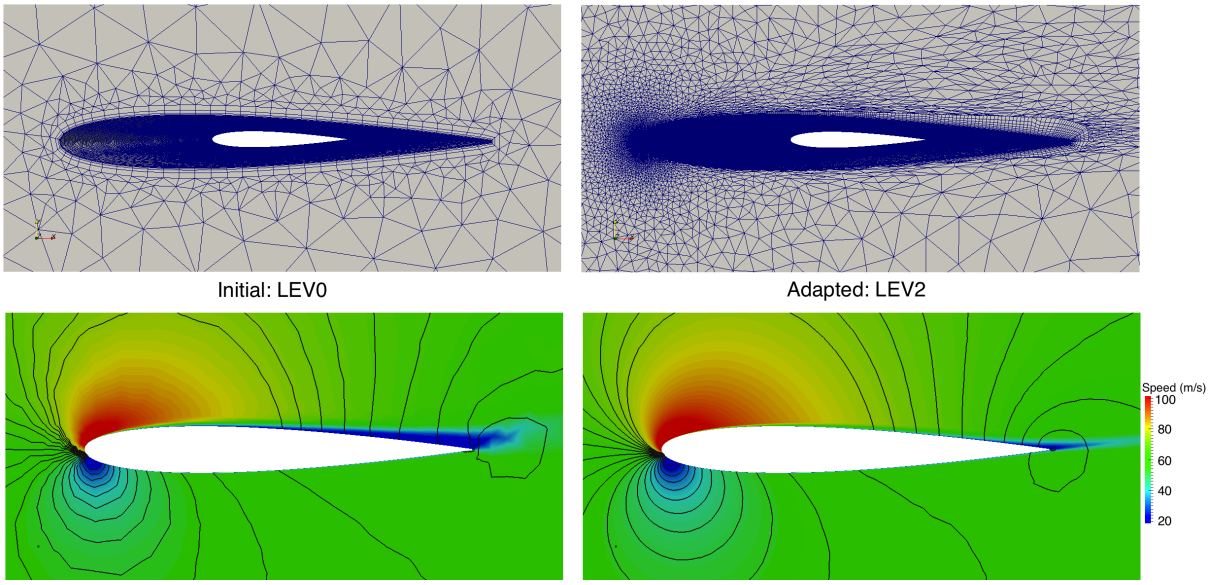
Figure 4.4: Cuts of the initial LEV0 and adapted LEV2 meshes, showing anisotropic adaptivity and speed distributions with pressure contours (black lines) for 5° angle of attack on NACA 0012

As the angle of attack is increased, the physical boundary layer on the suction surface has to face an adverse pressure gradient and starts getting thicker than the pressure surface. This behavior is correctly captured by our thickness adaptation strategies and the mesh boundary layer on the suction surface is adapted to be greater in height. The boundary layer on the pressure surface is comparatively smaller. This can be seen in Figure 4.4. The adapted mesh develops anisotropy and captures the contours better than the initial mesh.

4.1.3 Angle of Attack: 10°

As the angle of attack is further increased, the boundary layer on the suction surface thickens much more than the one on the pressure surface. This behavior is captured accordingly by our tools and is shown in Figure 4.5. The speed and pressure distribution is also enhanced with adaptivity.

Coefficients of pressure on the suction and pressure surfaces of the airfoil are plotted in Figure 4.6. The pressure data on the upper surface is taken from [29] and that on the lower surface is from [45]. The suction pressure peak near the leading edge on the upper surface is better captured



s

Figure 4.5: Cuts of the initial LEV0 and adapted LEV2 meshes, showing anisotropic adaptivity and speed distributions with pressure contours (black lines) for 10° angle of attack on NACA 0012

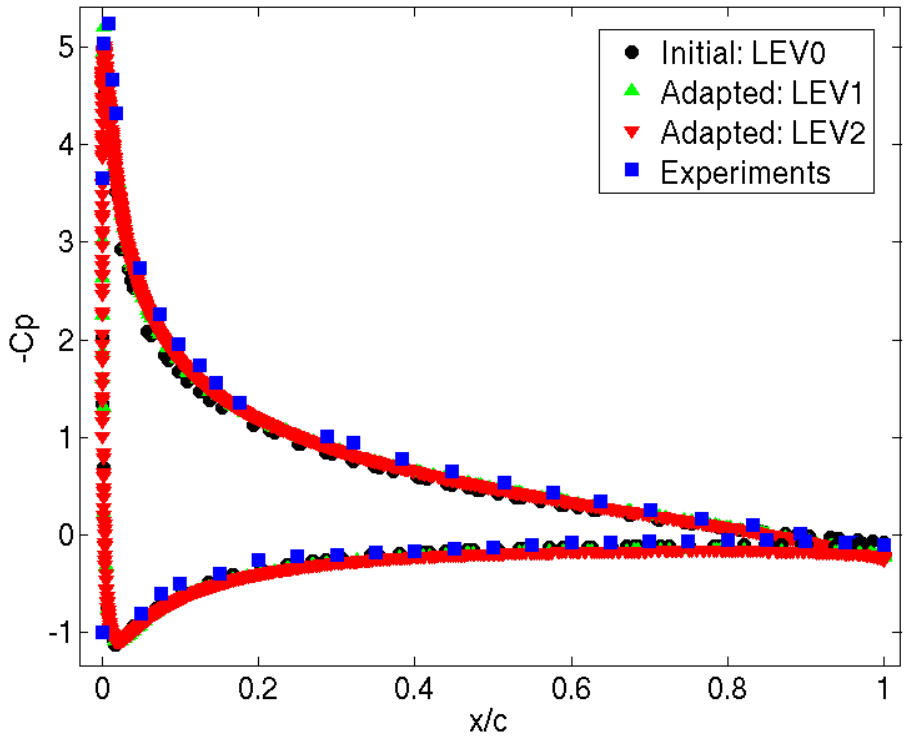


Figure 4.6: Coefficient of pressure plots for 10° angle of attack on NACA 0012

with the adapted meshes. The adapted meshes predict lesser separation than the initial mesh which is also evident from Figure 4.5 near the trailing edge, and is in accordance with experimental results.

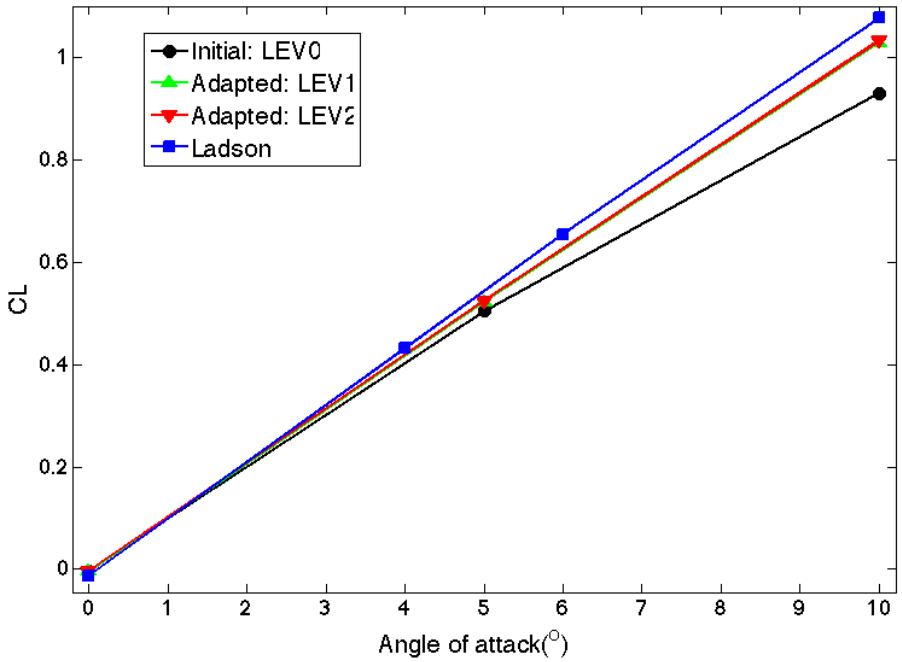


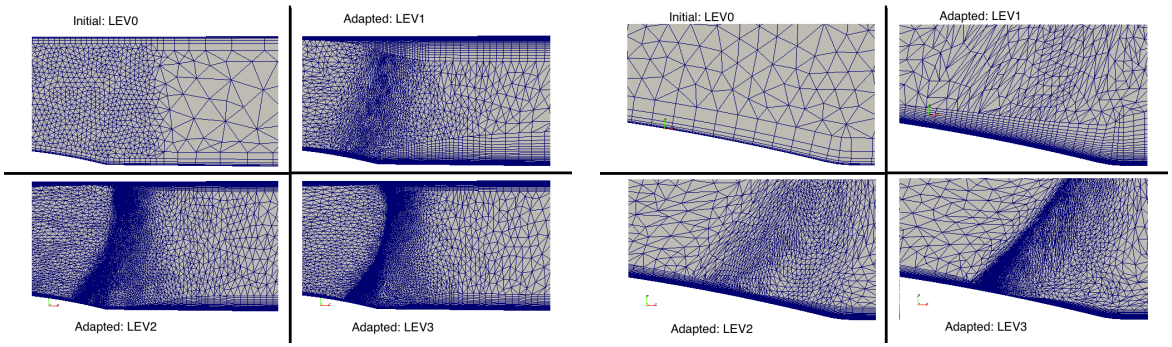
Figure 4.7: Coefficient of lift for NACA 0012 for different meshes in comparison to the experiment

The coefficient of lift curve is plotted in Figure 4.7. The experimental data is from Ladson [44]. At 0° angle of attack, all meshes show good agreement with the experiments. The initial mesh shows deteriorating C_L values at higher angle of attacks and adapted meshes show closer results to the experiments. The initial mesh fails to predict the correct C_L at 10° angle of attack but the adapted mesh shows reasonable agreement. This wrong behavior of the initial mesh can be attributed to the false separation predicted on the suction surface, near the trailing edge. As the boundary layer thickness is adjusted in the adapted mesh, this behavior is eliminated. The curves for the adapted LEV1 and LEV2 meshes lie on top of each other, indicating grid convergence.

4.2 Delery Bump

The second case that we used for thickness adaptation analysis is the Delery bump, which is a 2D bump with steady transonic flow. This case is often used to evaluate the performance of turbulence models. Air enters the nozzle from a reservoir at $p_t = 96000\,Pa$ and $T_t = 300\,K$, and accelerates over the bump reaching supersonic speeds. The outflow is maintained at 61500 Pa. Since the flow can not adjust to the back pressure, a shock appears on the leeward surface of the bump. The experiments for this case were first carried out by Delery [22].

The finite element formulation described in reference [80] was used for the compressible simulations. We used RANS Spalart-Allmaras one equation turbulence model [71] in this study. The adaptivity used combined pressure and velocity Hessians as error indicators [53]. The first cell height is requested at $y^+ = 1$. Because the flow separates after the shock, the separation detection algorithm detects this region and the boundary layer attributes are set accordingly.



(a) Initial and adapted meshes

(b) Zooms of the shock wave - boundary layer interaction zone

Figure 4.8: Initial and adapted meshes for Delery bump

Figure 4.8(a) shows zooms of the initial and the adapted meshes over the bump. Note that the initial mesh is very coarse and is not a practical choice as a stand alone mesh to resolve the shock. Here we want to show the effectiveness of our adaptation approach by showing that even by starting with such a poor mesh, we are able to adaptively reach a good mesh which can resolve the important structures like shocks to a satisfactory degree.

As seen from the figure, the boundary layer thickens after the shock as expected. Before the

shock, the boundary layer thickness remains low on both bottom and top walls. In the separated region, the boundary layer height is maintained until flow reversal is detected in the boundary layer profile. The LEV1 mesh shows some overshoot in the boundary layer thickness, but subsequent adapted meshes show converged behavior in terms of the boundary layer height. Adaptive refinement follows the curved shock such that the direction normal to the shock is refined much more than along the shock. This behavior is more evident from Figure 4.8(b), which displays the meshes in the shock wave - boundary layer interaction region. The adapted LEV2 and LEV3 meshes knock down the boundary layer height in the vicinity of the shock which is in accordance to the shock detection algorithm. One thing to notice is that the initial mesh has too few points near the wall to properly detect the flow reversal within the boundary layer profile, which occurs immediately after the shock and the boundary layer height is maintained according to the attached boundary layer algorithm. However the adapted LEV1 mesh becomes fine enough to detect this profile change and separation is detected and the boundary layer is destroyed immediately after the shock and is maintained only until the height at which flow reversal is detected.

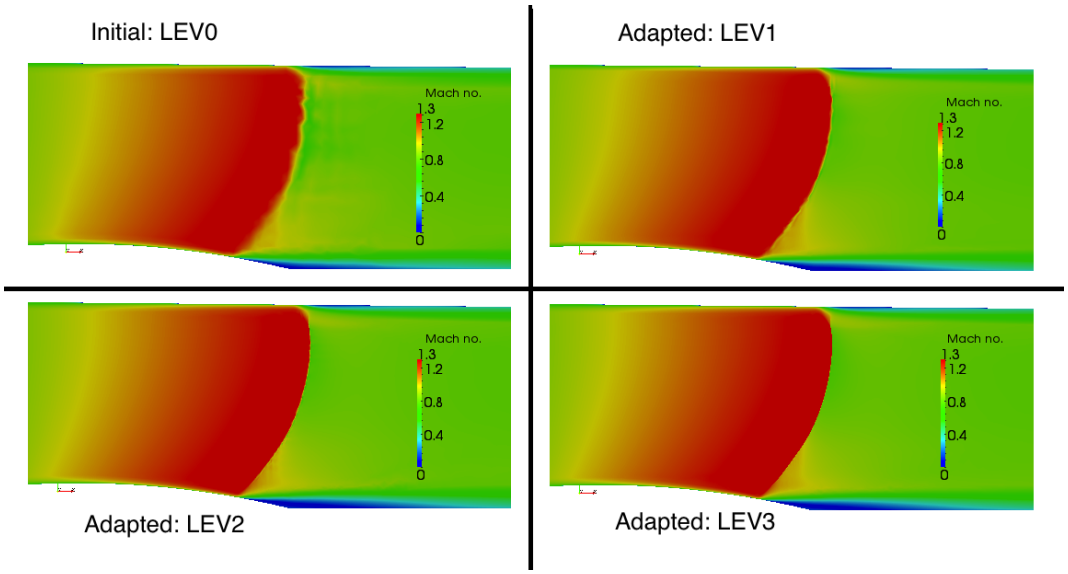


Figure 4.9: Mach number for a slice along the symmetry plane

Figure 4.9 shows slices of Mach number and Figure 4.10 shows Mach number contours for the initial and the adapted meshes. The contours show progressive sharpness in the shock region,

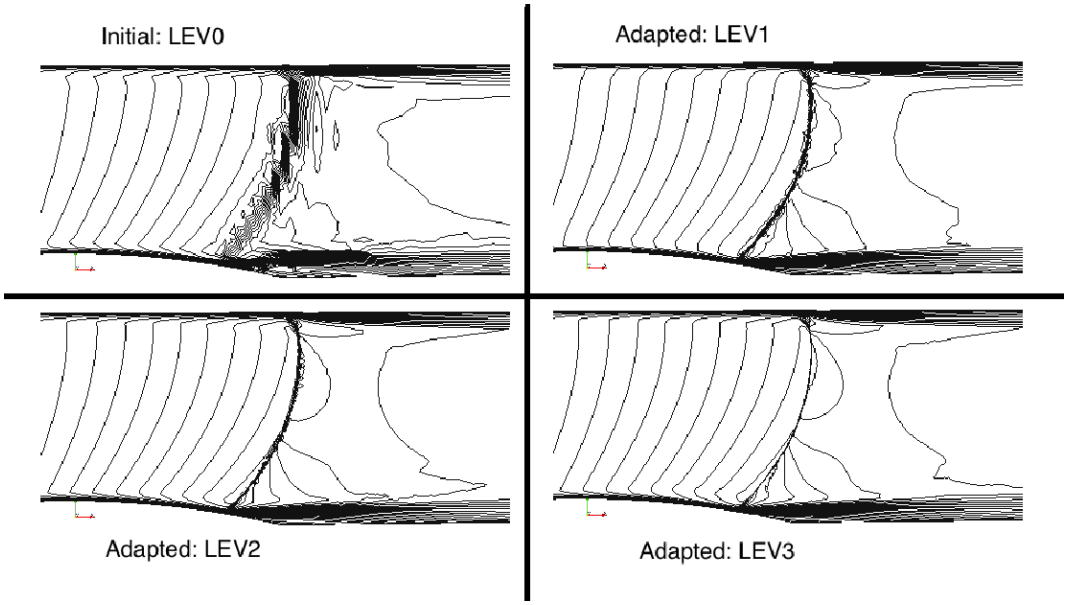


Figure 4.10: Mach number contours for initial and adapted meshes

with adaptivity. The LEV3 mesh gives the sharpest resolution of Mach number contours. A weak vertical shock is seen in Figure 4.9, originating from the intersection of the curved and the straight section of the strong shock.

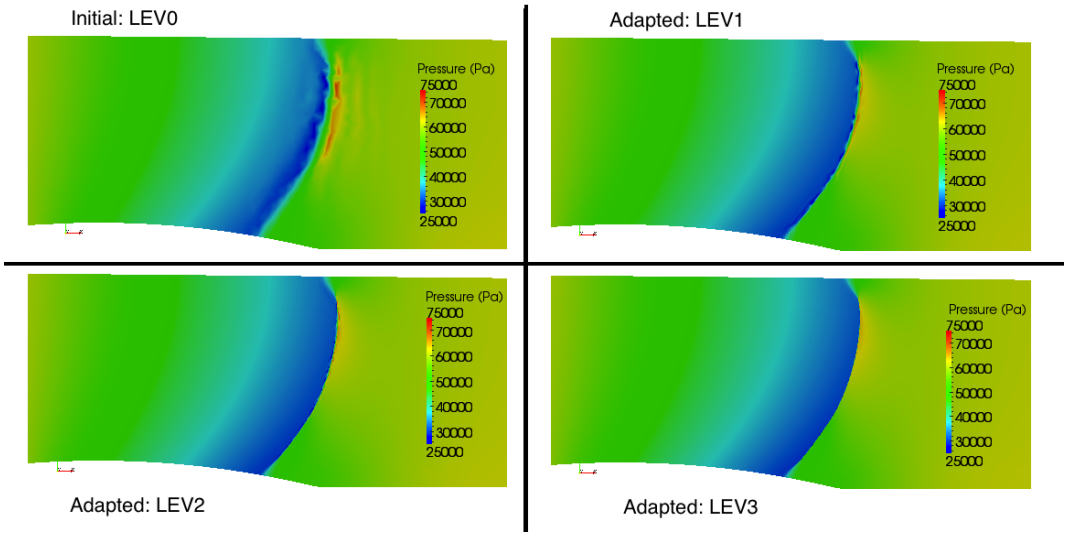


Figure 4.11: Pressure slice for initial and adapted meshes

Figure 4.11 shows pressure slice for the shock region over the bump. The adapted meshes are able to resolve the shock sharply as compared to the initial mesh, the best resolution is given

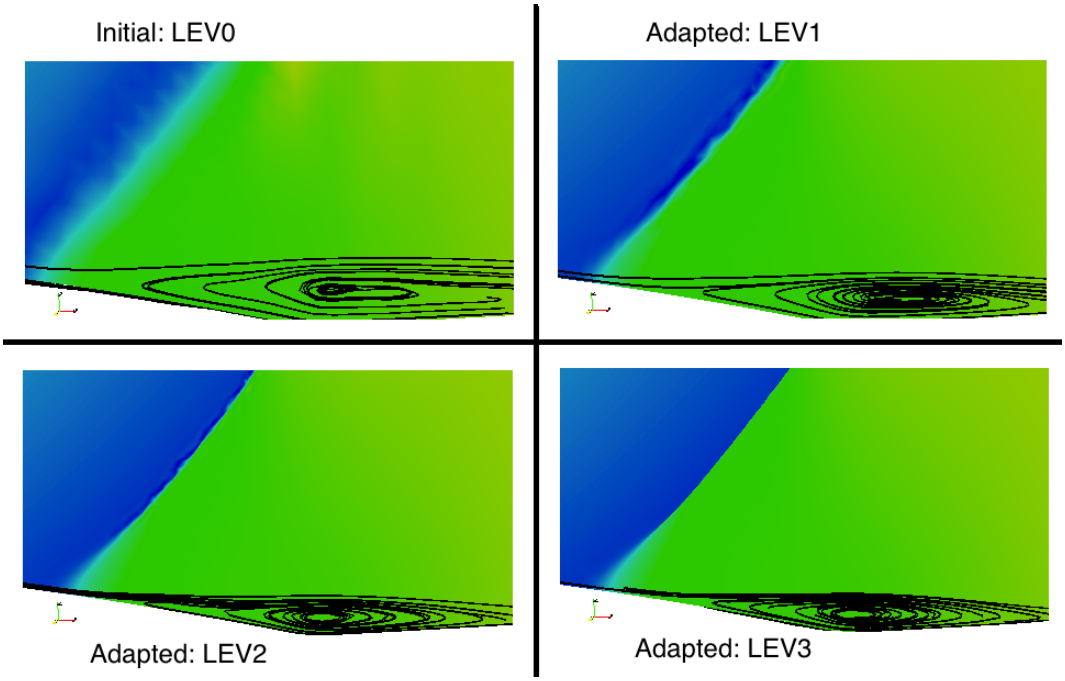


Figure 4.12: Streamlines in the separation zone

by the LEV3 mesh.

Figure 4.12 shows a zoom of the shock wave - boundary layer interaction zone and streamlines in the separation zone after the shock. The location of the interaction zone is better predicted in the adapted meshes and moves a bit to the right with respect to the initial mesh. The initial mesh over predicts the size of the the separation bubble, which gets corrected in the adapted meshes. The LEV2 and LEV3 meshes show similar behavior of the streamlines, indicating converged behavior for the separation bubble.

	Separation point (x/h)	Reattachment point (x/h)
Initial mesh (LEV0)	22.33	26.60
Adapted mesh (LEV1)	22.08	26.66
Adapted mesh (LEV2)	22.15	27.08
Adapted mesh (LEV3)	22.14	27.16
Experiments (Delery [22])	21.84	27.04

Table 4.1: Separation and reattachment locations for Delery bump, normalized by the bump height

Table 4.1 lists the separation and reattachment points for different meshes. The distance

values are normalized by the height of the bump (h), which is 12mm. Initial mesh predicts delayed separation and earlier reattachment than the experimental values. The adapted meshes, especially LEV2 and LEV3, give a significantly better agreement with the experiments.

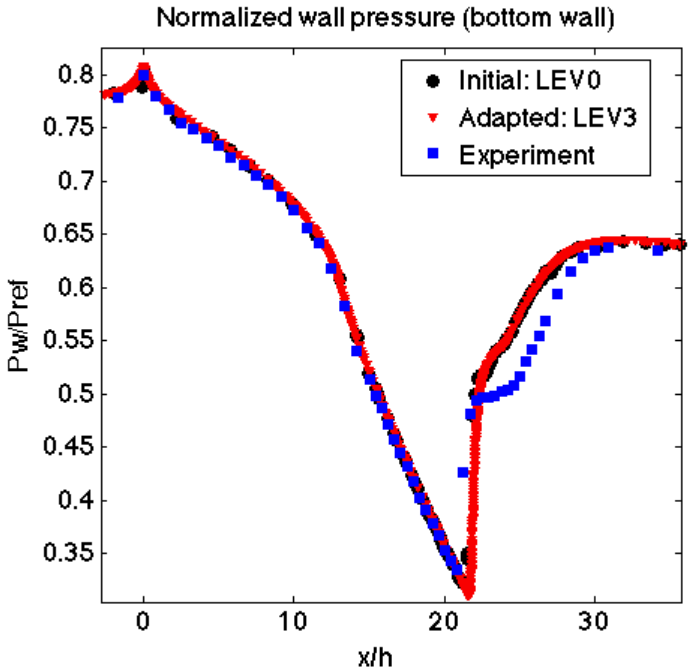
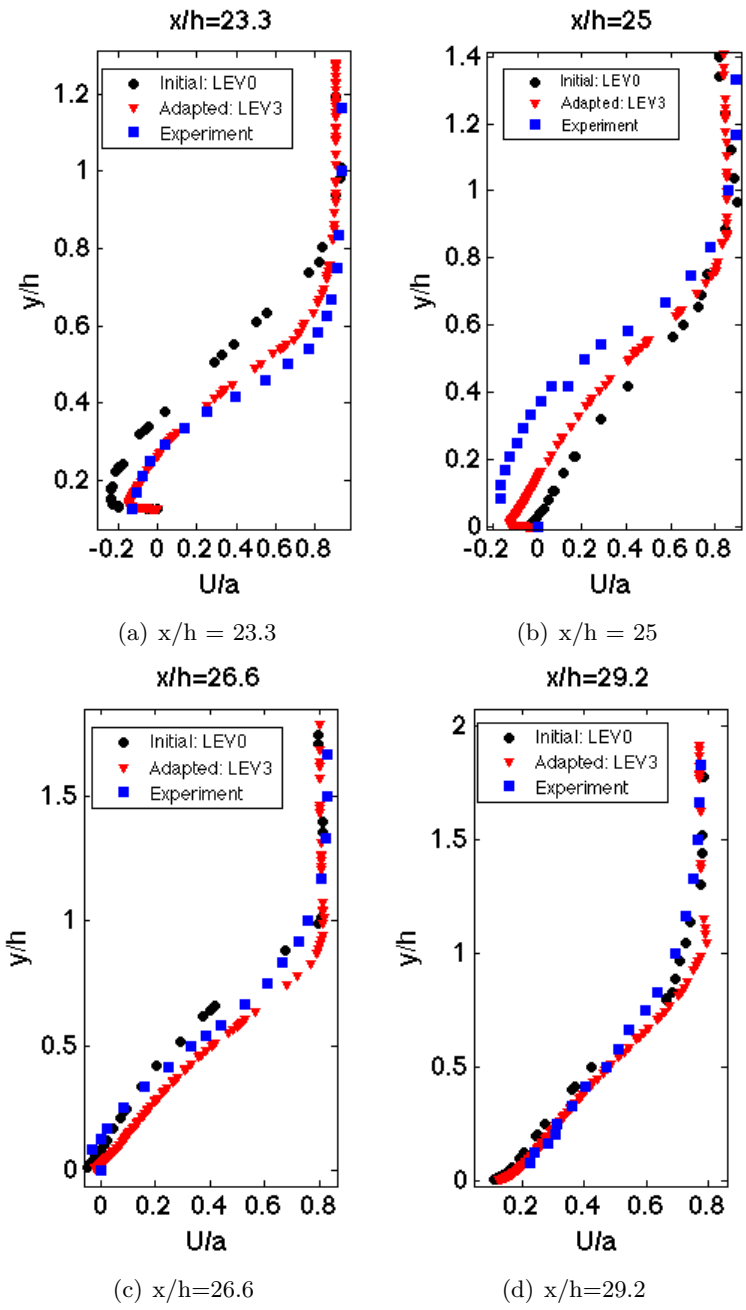


Figure 4.13: Normalized bottom wall pressure for Delery bump

The normalized bottom wall pressure is plotted in Figure 4.13 and indicates reasonable agreement between the experimental and the simulation values for the initial and the adapted meshes. The flow solver over predicts the wall pressure after the shock, in the separation zone. This behavior has been seen before by Emory et al. [23] and Lien et al. [46]. These differences can be attributed to the turbulence model's limitations in capturing the effect of separation, and so it is not a concern of this paper.

Boundary layer profiles are plotted in Figure 4.14 along various streamwise locations. Figure 4.14(a) and 4.14(b) show boundary layer profiles in the separation zone after the shock. The initial mesh is not able to capture the correct behavior due to its coarse resolution in this region. The adapted LEV3 mesh shows much better agreement with the experimental values in the separation zone. This is particularly important because in the separation zone, we are limiting the height of mesh boundary layers. For the boundary layer profiles after reattachment shown in



(a) x/h = 23.3

(b) x/h = 25

(c) x/h=26.6

(d) x/h=29.2

Figure 4.14: Boundary layer profiles at various streamwise sections

Figure 4.14(c) and 4.14(d), the adapted and the initial meshes both give good approximations.

To make sure grid convergence was reached, we uniformly refined the LEV3 mesh. This means that each mesh edge was split into two to get a new LEV4 mesh. This LEV4 mesh showed no significant difference in the solution when compared to the LEV3 mesh, which makes a strong

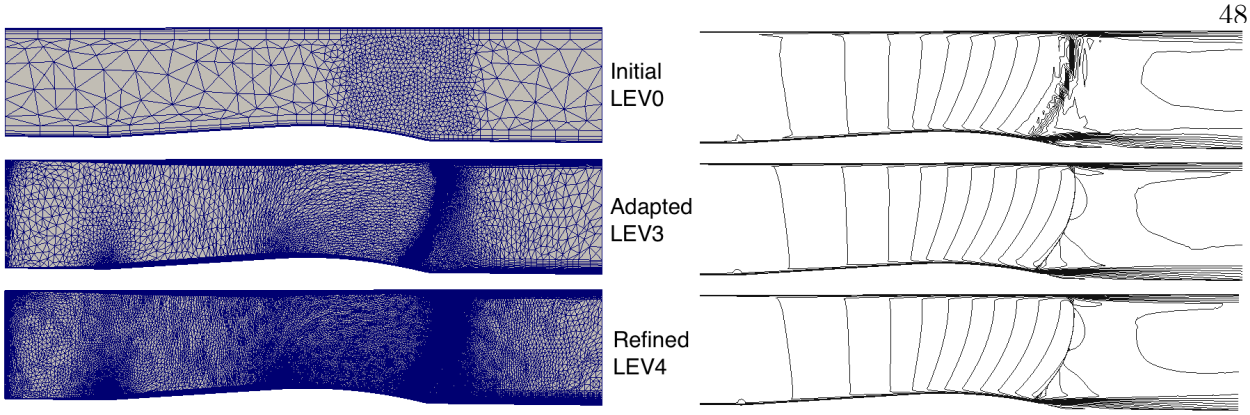


Figure 4.15: Initial LEV0, adapted LEV3 and uniformly refined LEV4 meshes and corresponding Mach number contours for the Delery bump case

case for verification of the results. Figure 4.15 shows the LEV4 mesh generated with uniform refinement in comparison to the initial LEV0 and the adapted LEV3 mesh. The mach contours on the right show no significant difference in their behavior between LEV3 and LEV4, displaying that grid independence is reached. This means that any further mesh refinement beyond the LEV3 mesh produces very similar results and is unnecessary. It further suggests that the remaining disagreements with the experiment can be attributed to limitations in the turbulence model and/or differences between the experiment and the problem defined here (e.g., 3D side walls). This strongly suggests that our adaptive procedure is verified.

4.2.1 Comparison with Nested Refinement Approach

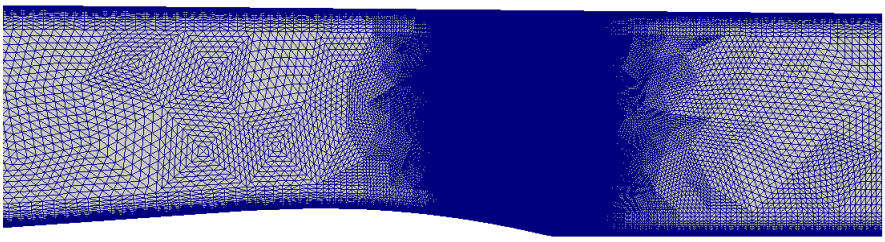
To display the effectiveness of adaptive refinement approach, we compared it with nested refinement of the Delery bump case. In nested refinement, each mesh edge is split in two. The mesh boundary layers are not refined in the direction of the growth curves, meaning that the normal spacing is kept constant. The first two refinement passes are done with this approach. However, sometimes there is also a need to change the normal spacing of the boundary layers to check if that has any effect on the flow solution. With the adaptive tools that we use, this can be achieved by converting the prisms and pyramids in the boundary layers to tetrahedra. In the next refinement pass, the normal spacing is also changed using tetrahedra elements in the boundary layers.

The coarse mesh that we used in the adaptive approach is not a practical mesh which can be used in any real analysis and is inadequate to correctly capture the shock. While it was useful in showing that we could start with such a bad mesh and still arrive at a good mesh using adaptivity, for the nested refinement approach it makes sense to use a more practical mesh. For this, the area near the leeward side of the bump is refined preemptively using box refinement.

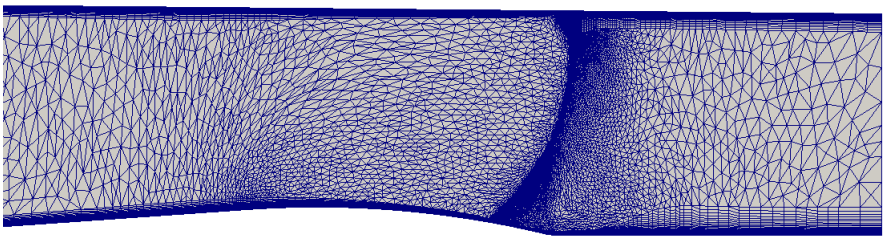
The comparison of the adapted LEV3 mesh and the refined LEV3 mesh is given in Table 4.2. The refined LEV3 mesh has 85.16M elements which is far greater than the 1.78M elements of the adapted LEV3 mesh.

Mesh	# elements	# vertices
Refined mesh: LEV3	85.16M	16.72M
Adapted mesh: LEV3	1.78M	330k

Table 4.2: Computational comparison of meshes



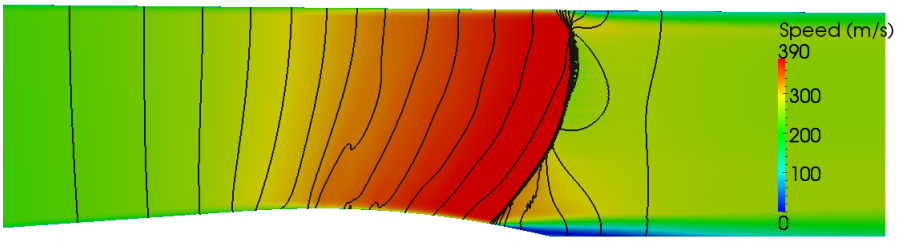
(a) Refined: LEV3



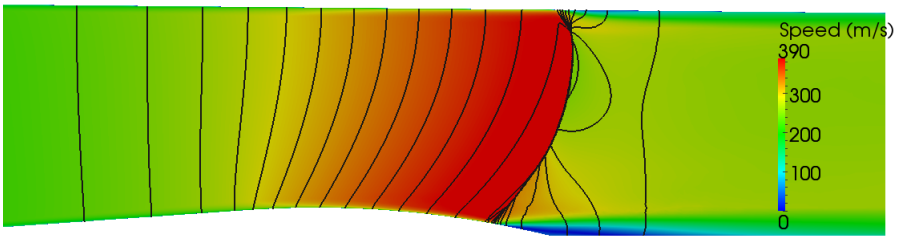
(b) Adapted: LEV3

Figure 4.16: Refined LEV3 and adapted LEV3 meshes for the Delery bump

Figure 4.16 shows the refined LEV3 and the adapted LEV3 meshes over the bump. Note that the mesh boundary layers in the refined LEV3 mesh are converted to all tetrahedra elements to refine the wall normal direction. Though the adapted LEV3 mesh shows prism boundary layers, for



(a) Refined: LEV3



(b) Adapted: LEV3

Figure 4.17: Speed and pressure contours for the refined LEV3 and adapted LEV3 meshes for the Delery bump

fair comparison, that mesh was also converted to all tetrahedra and those statistics have been used in Table 4.2. From the mesh pictures it is clear that the reason of efficiency given by the adapted mesh is due to the selective refinement of the shock region and the anisotropy which develops as a result. The other flow regions remain comparatively coarse for the adapted LEV3 mesh, which contribute to the computational savings, whereas for the nested refinement all regions are uniformly refined.

Figure 4.17 shows the speed contours (in color) and pressure contours (black lines) for the refined LEV3 and the adapted LEV3 meshes. The location and the shape of the shock is same for both the meshes, at this level of visualization. The sizes of the separation bubble predicted by both meshes match with each other. This indicates that both these LEV3 meshes give very similar results. However, computationally these two meshes are vastly different and using the adapted mesh leads to significant savings in terms of the computational resources. This underlines the efficiency and the effectiveness of the adaptive technology.

4.3 ONERA M6 Wing

The ONERA M6 wing is a classic CFD validation case. Air enters the wind tunnel at transonic speed and is accelerated over the wing to supersonic speeds causing a shock to appear on the upper surface of the wing. The free stream Mach number is 0.84, and the angle of attack is 3.06^o. The free stream pressure and temperature are 42.89 *psi* and 255.5 K. The Reynolds number is 11.72 million based on the mean aerodynamic chord the. This flow marks a strong need for adaptive grids due to its unknown shock location *a priori* to the flow solve and complex nature of the lambda shock. The reference experimental data is from Schmitt and Charpin in 1979 [65].

We used Spalart Allmaras one equation turbulence model for this case [71]. Pressure Hessians were used as error indicators to resolve the shock on the wing surface. The first cell height was requested at $y^+ = 1$.

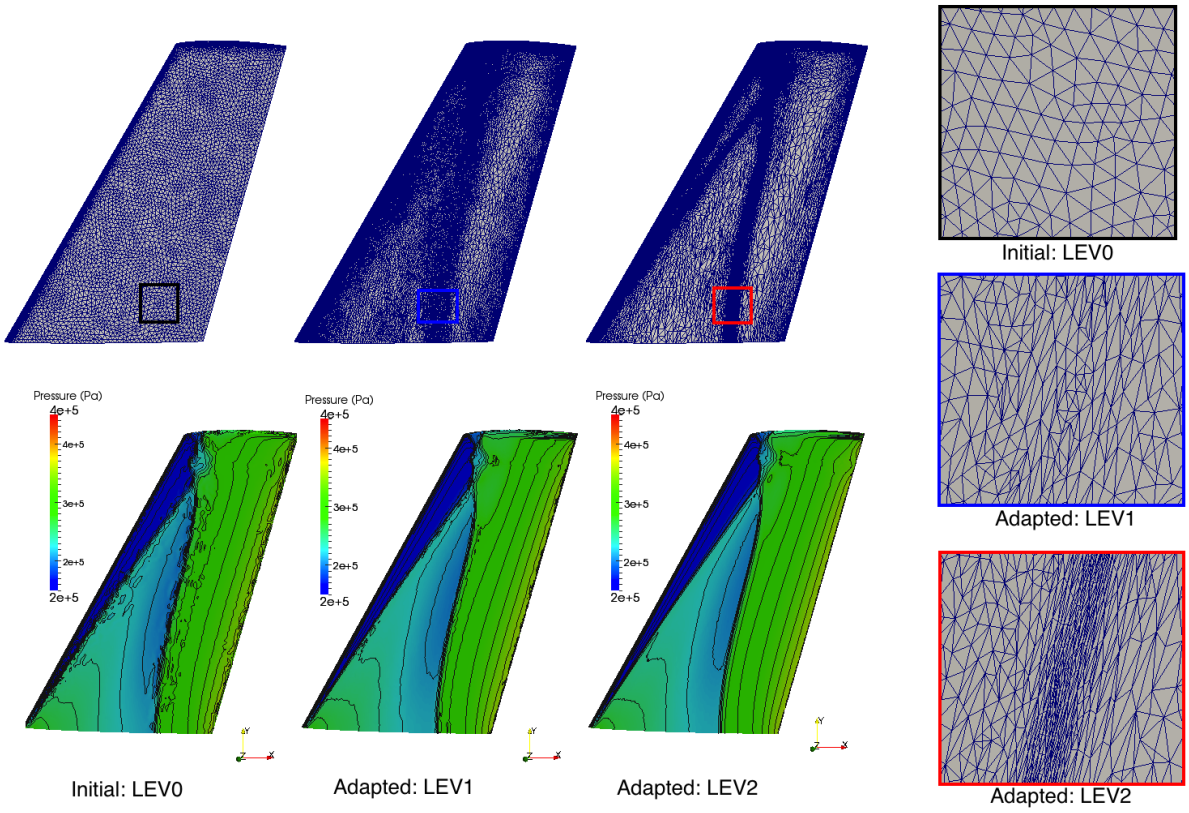


Figure 4.18: Meshes and corresponding pressure plot and contour

Figure 4.18 shows the initial and the adapted meshes for the wing and corresponding pressure contours, on the left side. The mesh gets refined in the shock region and the lambda shape of the shock is clearly replicated in the mesh. The mesh after the shock is coarsened, due to low values of pressure gradients in this region. The pressure contours become sharper and more regular in shape with adaptivity. On the right hand side of Figure 4.18, the pictures show the zooms of the shock region on the wing, displaying the anisotropy developed in the surface elements along the shock as compared to the direction normal to the shock. One thing to notice is that the elements start aligning themselves with the shock in the adapted LEV1 mesh, but need one more adaptation loop to completely show this behavior in a satisfactory manner.

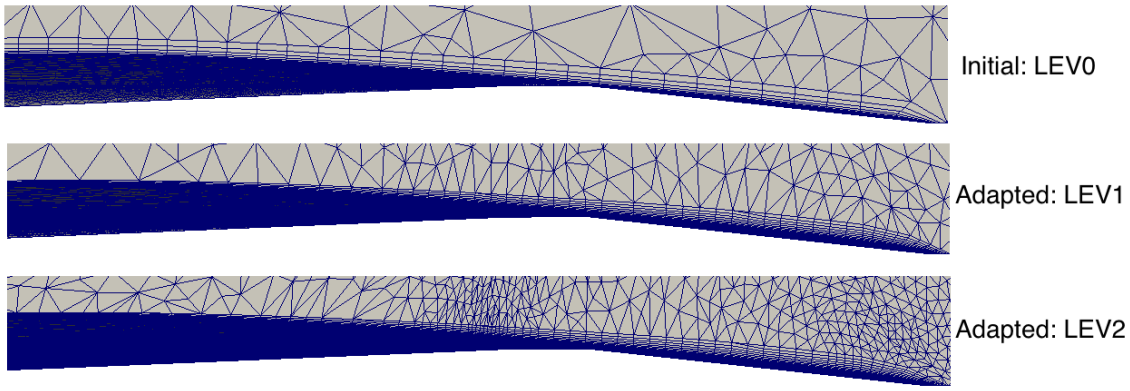
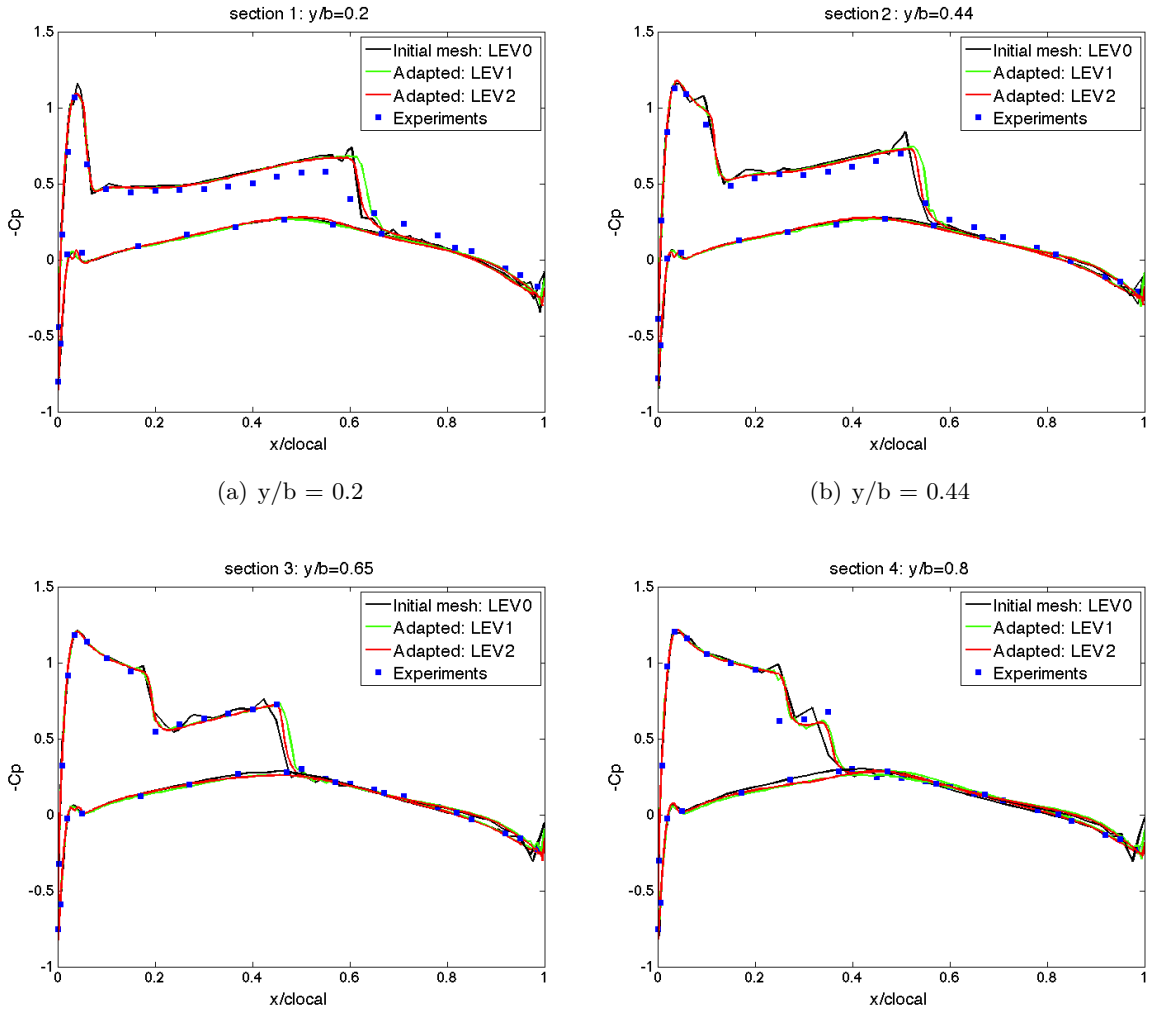


Figure 4.19: Change in the boundary layer height on upper surface of the wing

To illustrate the changing attributes of the mesh boundary layer with adaptivity, Figure 4.3 shows the boundary layer on the upper surfaces of the wing for initial and adapted meshes. Clearly, the boundary layer prior to the shock remains relatively low in total height. After the shock the boundary layer thickens as expected. The zone where shock wave meets the boundary layer can also be seen and the elements oriented with finer resolution normal to the shock and longer mesh along the shock are clearly visible.

Figure 4.20 shows coefficient of pressure plots for different spanwise sections of the wing, taken at different Y sections where b is the total span of the wing. The initial mesh shows some overshoots near the shock which is expected due to its coarse nature. Adaptivity removes this problem and



(a) y/b = 0.2

(b) y/b = 0.44

(c) y/b = 0.65

(d) y/b = 0.8

Figure 4.20: Coefficient of pressure on the wing

gives C_p values which match the experimental data. The initial mesh also over predicts the suction pressure peak for $y/b = 0.2$ section which gets eliminated with adaptivity. The plots for the LEV1 and LEV2 meshes almost lie on top of each other except near the shocks where some differences can be seen.

One particular area of interest is the shock on section 4 ($y/b = 0.8$), which is in fact a double shock. Our flow solver based on a finite element formulation as mentioned before is able to capture this double discontinuity and adaptivity improves the agreement with the experiments in this region.

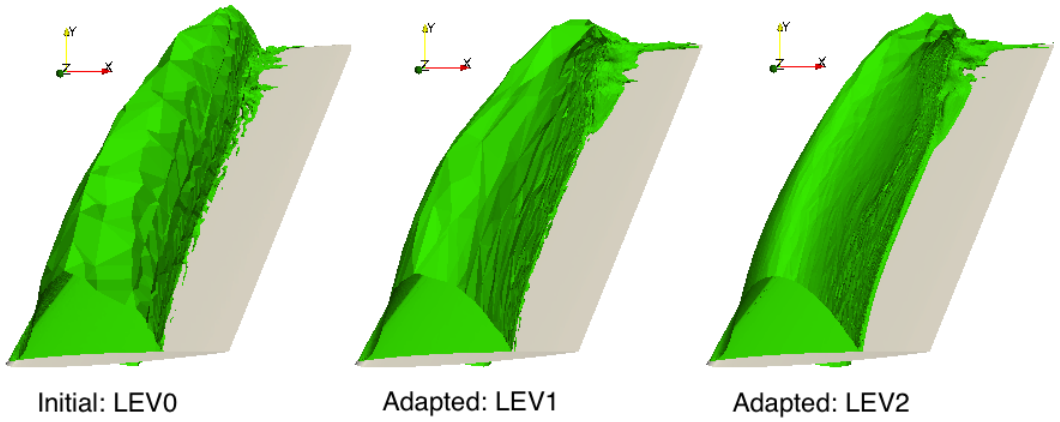


Figure 4.21: Iso-surface of Mach 1 for initial and adapted meshes

Figure 4.21 shows iso-surfaces of Mach number equal to unity on the upper surface of the wing. The anisotropy developed in the spanwise direction is replicated on the iso-surfaces. In general, the adapted meshes show smoother curves indicating higher degree of resolution as compared to the initial mesh.

Overall, for ONERA M6 wing case we see that the adapted meshes were able to capture the shock with higher accuracy than the initial mesh. The C_p values agree well with the experiments and improve with adaptivity.

In this chapter, we have successfully showcased thickness adaptation of mesh boundary layers and displayed advantages of adaptivity by comparing the results with experiments and in some cases, uniform refinement. The results in this chapter reinforce the idea of adaptive techniques being effective and efficient to get better flow results.

Chapter 5

Application Results: Serial In-plane Only Adaptation for Multi Element Wings

We present application of our in-plane boundary layer adaptive methods to complex aerodynamic wings. We have chosen popular multi element wings for this study which were used in the 1st and 2nd high lift prediction workshops organized by NASA [1, 2]. Boundary layer adaptivity for these cases is challenging due to their complex geometric shape and the complex flow field with interacting and intersecting boundary layers. Previous adaptive studies for these cases with unstructured grids have been limited to frozen boundary layers, i.e. only adapting the mesh outside the viscous region near the walls [41], [56]. So even though we display only in-plane adaptivity results in serial, the boundary layer adaptivity carried out in this chapter is state of the art. These cases were adapted in serial using the mesh adapt software made available by Simmetrix Inc. [34].

5.1 NASA Trap Wing

The first case for the multi element wing in-plane adaptation is a trapezoidal wing geometry. This multi-element wing has slats and flaps with the main wing, and was used as the analysis case in the 1st high lift prediction workshop organized by NASA in 2010 [1]. The configuration used is a landing configuration with the slat at 30° and the flap at 25° angle with respect to the chord of the main wing. The experiments for NASA trap wing were performed at NASA Langley in a 14x22-foot wind tunnel [40].

Figure 5.1 shows geometry of NASA trap wing. The element preceding the main wing is the slat and the one beyond its trailing edge is the flap. There are no slat and flap brackets for this

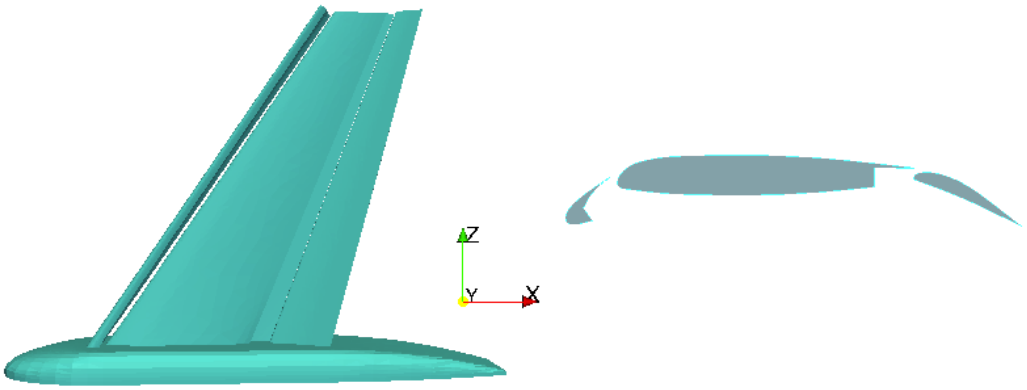


Figure 5.1: Geometry of NASA trap wing showing side view and a cut view of the section

geometry. A coarse mesh was generated with mixed element boundary layers and was used to get an initial solution. Two adaptivity loops were carried out and a combined approach of Hessians and scalar error indicators was used to drive adaptivity as explained in Chapter 2. This results in better wake refinement than just using the Hessians since momentum equation residuals are usually high in the wake.

The flow was modeled as an incompressible turbulent flow with Spalart Allmaras turbulence model [71]. Other relevant case setup information is given in Table 5.1.

Mach number	Mean aerodynamic chord (MAC)	Re_{MAC}	Angle of attack
0.2	1.0 m	4.3 million	13°

Table 5.1: Problem definition for NASA trap wing

The initial mesh for NASA trap wing has 3.3M elements, the adapted LEV1 mesh has 7.8M elements and the adapted LEV2 mesh has 12.3M elements. The seemingly small increase in the number of elements per adapt cycle can be attributed to the creation of anisotropic elements. Figure 5.2 shows cuts of the initial LEV0 and the adapted LEV2 meshes and corresponding speed distribution (colored) and pressure contours (black lines). Superior capturing of the wake with adaptivity is evident from the anisotropic mesh in this region and the speed pictures. The pressure contours which are a bit jagged for the initial mesh, become more regular and smooth in the adapted

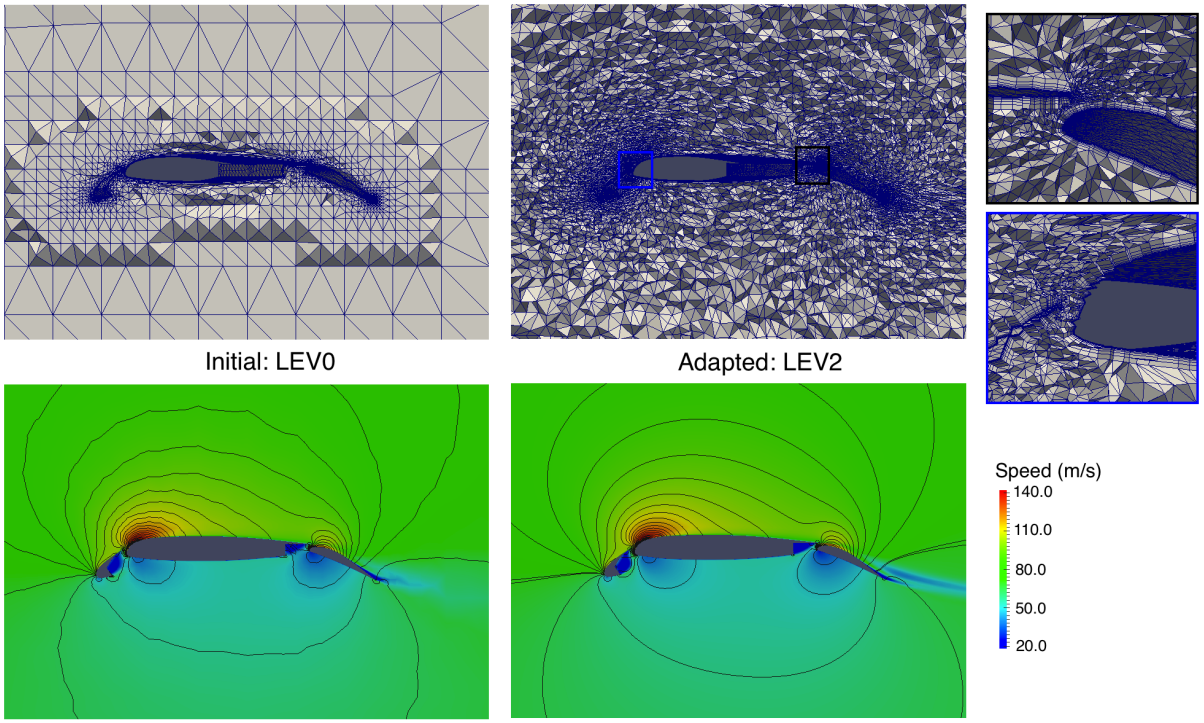
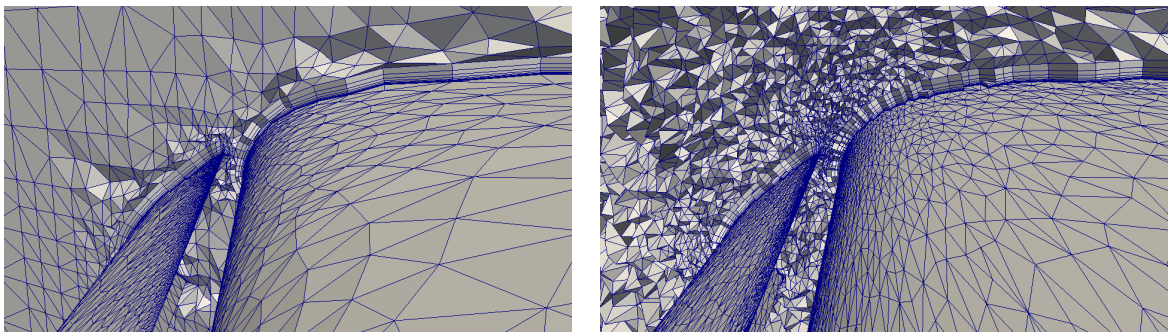


Figure 5.2: Cuts of the initial LEV0 and adapted LEV2 meshes, showing anisotropic adaptivity and speed distribution with pressure contours

mesh. The zooms on the right show the adapted mesh near the leading edges of the main wing and the flap element showing the refinement in these regions, also displaying refinement near the trailing edges of the slat and the main wing.

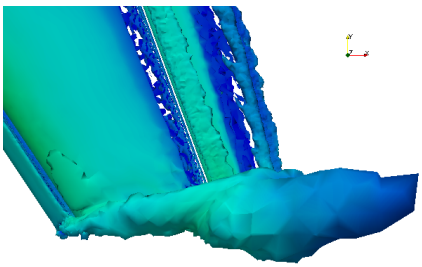


(a) Initial mesh: LEV0 (b) Adapted mesh: LEV2

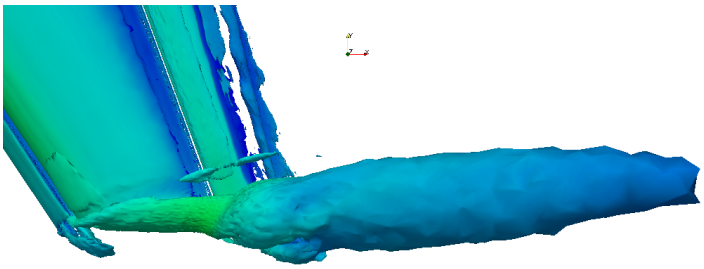
Figure 5.3: Cut view of the initial and the adapted meshes near the leading edge of the main wing, showing anisotropic elements

Figure 5.3 displays the mesh near the leading edge of the main wing for the initial LEV0 and

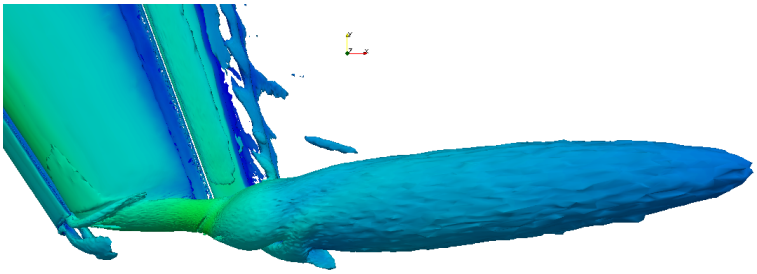
the adapted LEV2 meshes to illustrate the anisotropy developed in the spanwise direction. The leading edge is refined such that the elements have smaller edge lengths in the streamwise direction as compared to the spanwise direction. This behavior is expected because the flow changes in the streamwise direction are very large near the leading edge of the main wing. This figure also shows the in-plane boundary layer adaptivity as the stacks of boundary layers get split in the layer direction.



(a) Initial mesh: LEV0



(b) Adapted mesh: LEV1



(c) Adapted mesh: LEV2

Figure 5.4: Iso-surfaces of Q criterion for NASA trap wing showing the tip vortex

Figure 5.4 shows the iso-surfaces of the Q criterion which captures the elongated tip vortex. Q

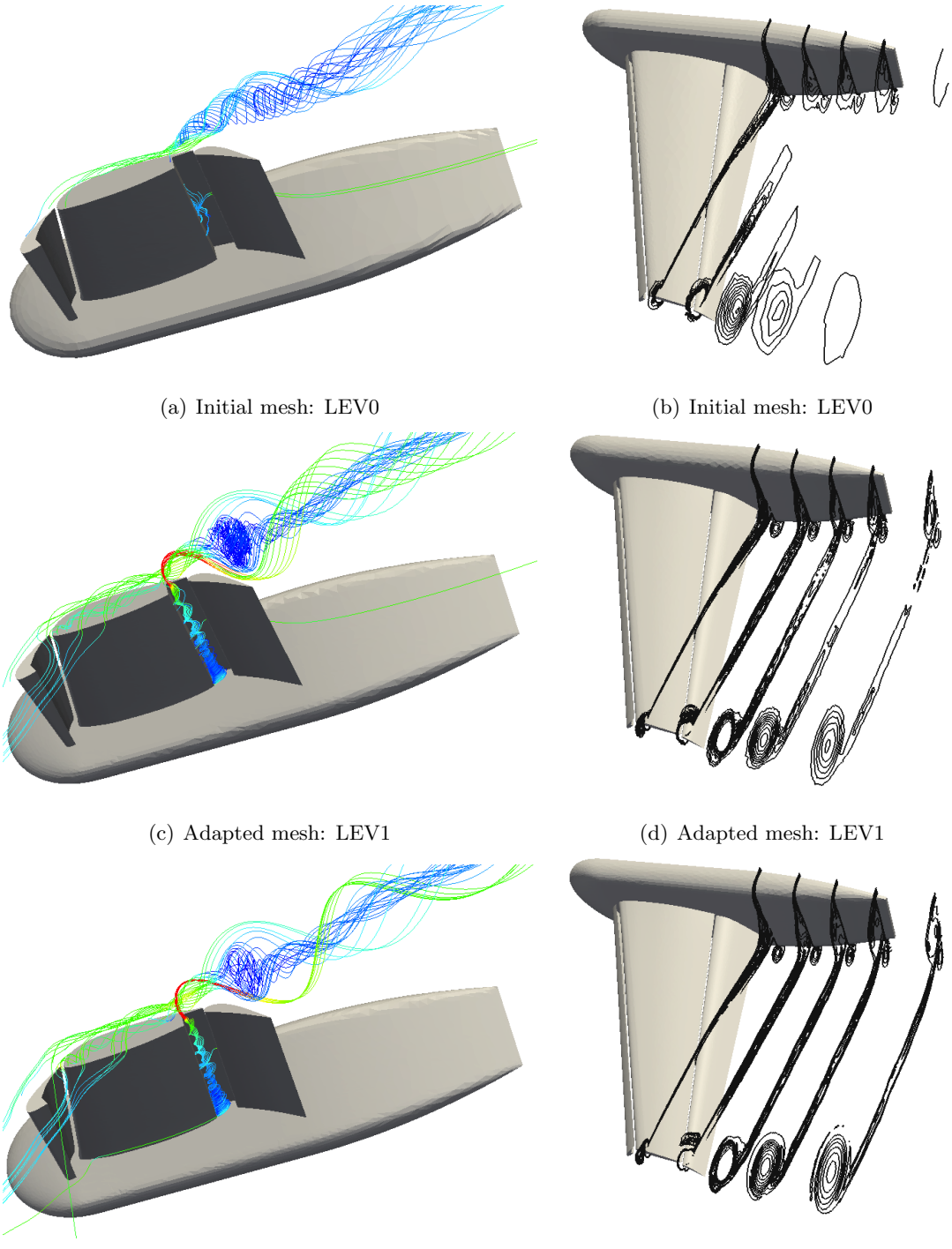
criterion is calculated as $0.5(\Omega_{ij}\Omega_{ij} - S_{ij}S_{ij})$ where Ω is the asymmetric part and S is the symmetric part of the velocity gradient tensor. Since vorticity usually captures high normal gradients in boundary layer flows, the vortical structures get overshadowed. The Q criterion eliminates this problem by signifying the regions where rotation dominates the shear and strain, which is often the case for the tip vortex for positive Q values.

The iso-surfaces indicate that the tip vortex dissipates quickly for the initial mesh due to its coarseness. However, its real structure is captured much better in the adapted meshes. This is important because with our error indicators we get refinement in the tip area and the wake and so the flow structures are captured much better in this region than the coarse initial mesh. The effect of anisotropic elements in the wake can also be seen on the structure of the tip vortex.

Some other results of adaptivity are shown in Figure 5.5 which displays the streamlines near the tip on the left hand side and vorticity magnitude contours on the right hand side for different meshes. The initial coarse mesh gives a lot of separation and a complete free shear layer near the tip, but the adapted meshes predict more turbulent activity in this region. The streamlines inside the flap slot of the main wing are not clearly seen in the initial mesh but can be seen in the adapted meshes. The vorticity magnitude contours of the initial mesh indicate that the tip vortex gets dissipated quickly due to the coarseness of the mesh. The initial mesh is not able to completely resolve the side-of-body vortices as well, which are resolved much better in both of the adapted meshes. Overall, adaptivity captures the tip and side-of-body vortical features much better due to increased refinement.

The coefficients of pressure for the slat, the main wing and the flap elements are plotted in Figure 5.6 and Figure 5.7. Note that the Y axis is plotted for $-C_p$. For the slat element, the agreement with experiments is good for all the meshes especially for the 17% and 50% sections, however, the adapted meshes show less scatter of the C_p values for the 17% section. Near the tip (Figure 5.7(a)), the adapted meshes show overall slightly better agreement with the experiments compared to the initial mesh.

For the main wing, the agreement is good for all the three meshes with the experimental



(a) Initial mesh: LEV0

(b) Initial mesh: LEV0

(c) Adapted mesh: LEV1

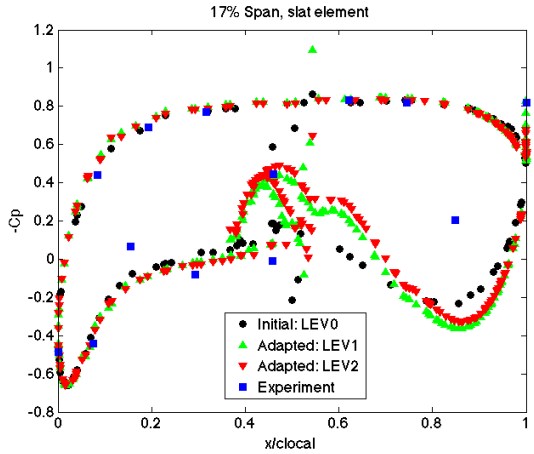
(d) Adapted mesh: LEV1

(e) Adapted mesh: LEV2

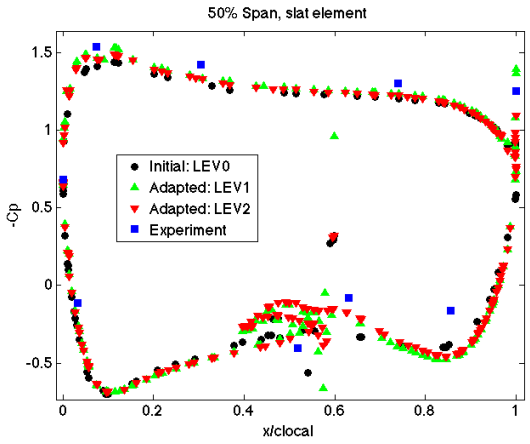
(f) Adapted mesh: LEV2

Figure 5.5: On left: streamlines near the tip, on right: contours of vortex magnitude on various slices with normals in X direction

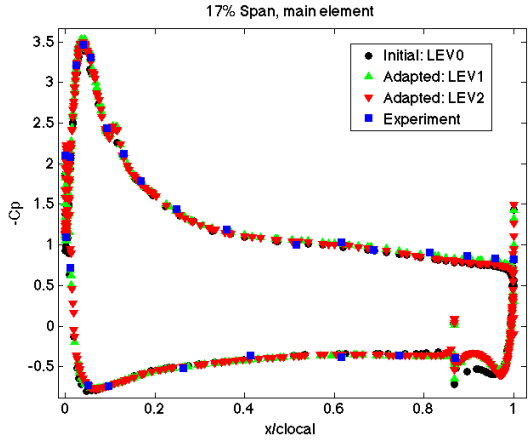
data for 17% ad 50% sections. Some minute differences are seen near the trailing edge. The initial



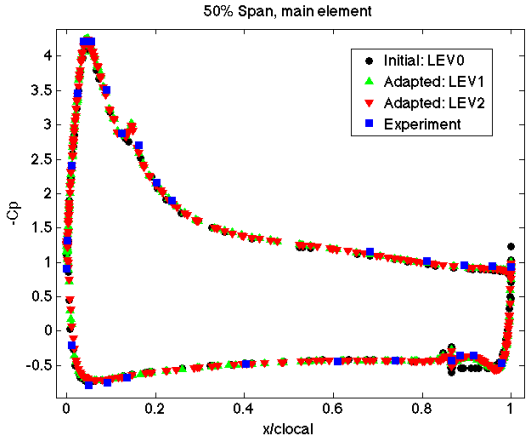
(a) Slat element: 17% span



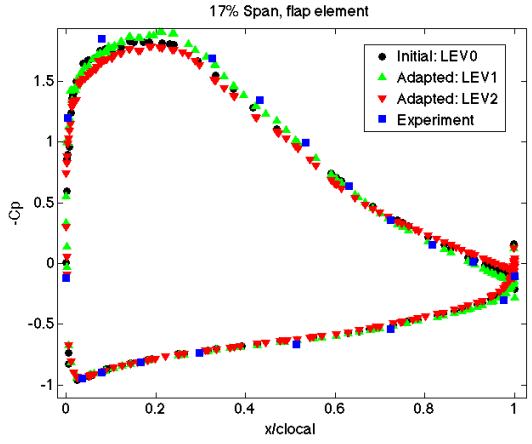
(b) Slat element: 50% span



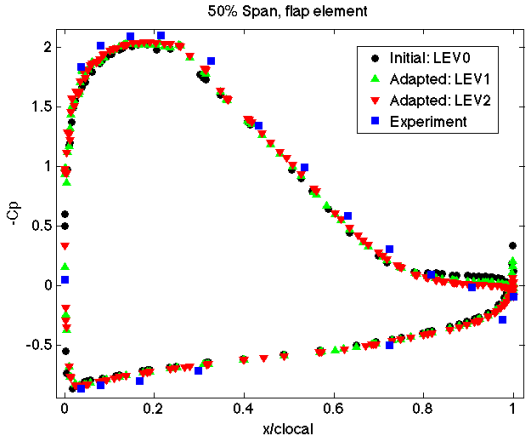
(c) Main wing: 17% span



(d) Main wing: 50% span

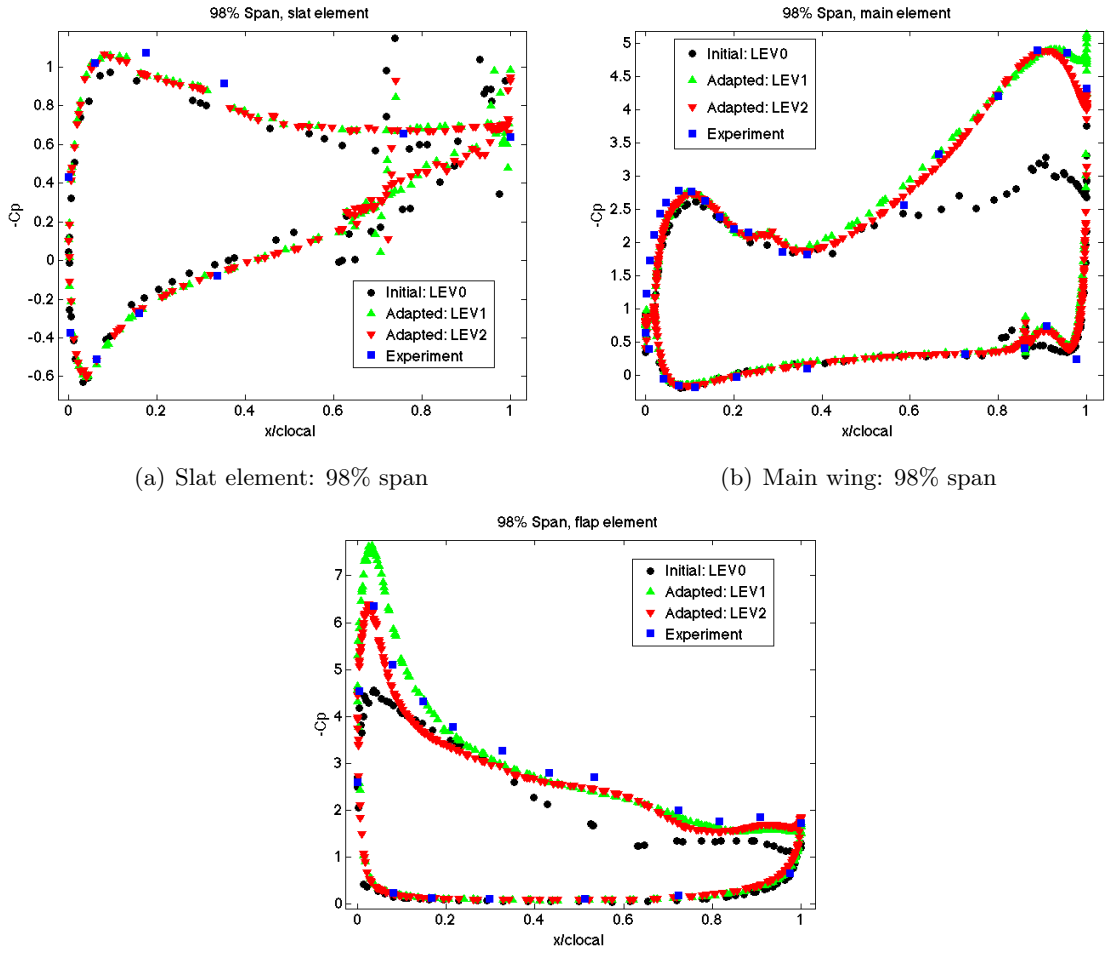


(e) Flap element: 17% span



(f) Flap element: 50% span

Figure 5.6: Coefficient of pressure for 17% and 50% span



(a) Slat element: 98% span

(b) Main wing: 98% span

(c) Flap element: 98% span

Figure 5.7: Coefficient of pressure for 98% span

mesh does not have enough resolution on the pressure side near the trailing edge and thus fails to capture the effect of reduction in the thickness of the wing in that region, which is captured by the adapted meshes.

Again near the tip (Figure 5.7(b)), the initial mesh under predicts separation which leads to a flatter C_p curve for the suction side of the wing, far off from the experimental values, with the worst behavior seen near the trailing edge. With adaptivity, this region receives enough refinement and is captured with a greater accuracy. The C_p values for the adapted meshes are in good agreement with the experiments near the tip of the main wing for both adapted meshes. For the pressure side also, the adapted meshes perform better where the thickness of the wing drops down.

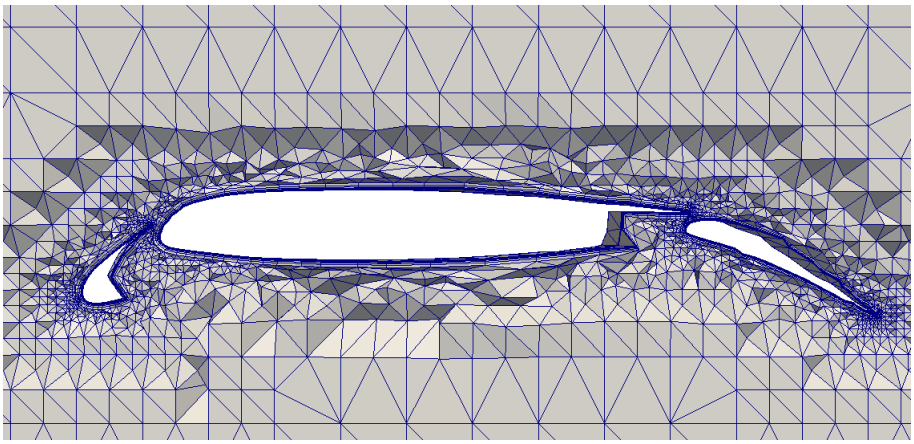
For the flap element, the C_p values on 17% ad 50% sections show good agreement with the experimental values. This indicates that the initial mesh is adequate to accurately predict the flow phenomenon away from the tip. This is not surprising since flow is attached near the in-board and the mid-board sections. However, near the tip (Figure 5.7(c)) for the 98% span section, the initial mesh under predicts the suction peak. Interestingly, the adapted LEV1 mesh shows an overshoot in the suction pressure peak near the nose of the flap, but the adapted LEV2 mesh shows good agreement with the experiments. This indicates that usually at least a couple of adaptivity passes could be required to arrive at a good mesh. For the rest of the section, the adapted meshes show better agreement with the experimental data, also capturing the wavy nature near the mid-section area. The initial mesh under predicts the separation near the trailing edge which is again predicted better by the adapted meshes.

5.1.1 Comparison with Nested Refinement Approach

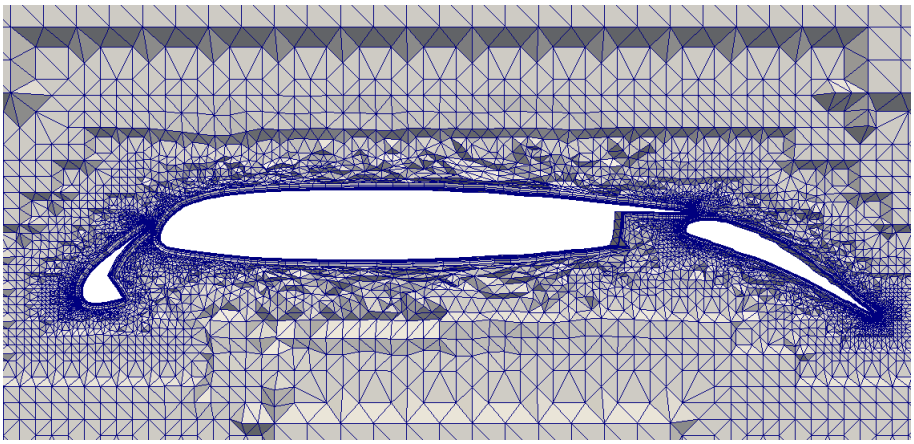
To display the effectiveness of adaptive refinement approach, we compare it with nested refinement of NASA trap wing. In the first high lift prediction workshop [1], participants were required to create a coarse, a medium and a fine mesh and their results were compared. The mesh sizes (mesh edge lengths) in each subsequent grids were 1.5 times smaller than the previous ones. We use a similar approach but instead we perform a nested refinement operation to reach the next level of the mesh. We then compare results from these meshes with our adapted mesh results.

In nested refinement (also called uniform refinement), each mesh edge is split in two. This means that each tetrahedron element in the mesh gets refined into 8 new tetrahedra elements. The mixed element mesh boundary layers are not refined in the direction of the growth curves, meaning that the normal spacing is kept constant. This results in refinement of each prism element into 3 new prism elements and each pyramid element is refined into 2 new pyramid elements.

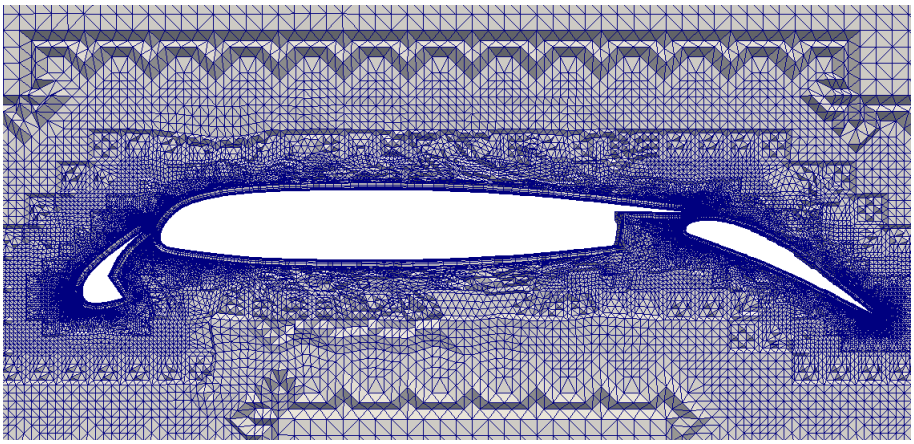
Figure 5.8 shows the cut views of the meshes at each refinement level created with nested refinement. The refinement of prisms and tetrahedra as explained before can be easily seen in the pictures.



(a) Initial Mesh: LEV0



(b) Uniformly refined mesh: LEV1



(c) Uniformly refined mesh: LEV2

Figure 5.8: Cut views of the initial and the uniformly refined meshes for NASA trap wing

Table 5.2 shows the computational comparison of the uniformly refined meshes with the ini-

Mesh	# elements	# vertices
Initial mesh: LEV0	3.39M	1.13M
Refined mesh: LEV1	20.65M	6.65M
Refined mesh: LEV2	139.33M	37.18M
Adapted mesh: LEV2	12.85M	3.46M

Table 5.2: Computational comparison of meshes

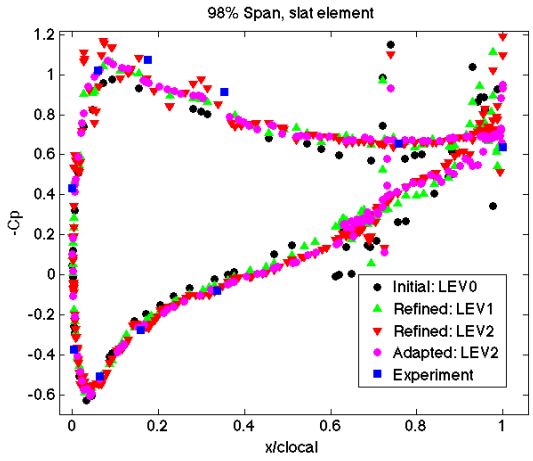
tial and the adapted LEV2 mesh. Between the LEV0 to the refined LEV2 meshes, the number of elements roughly increases by 6-7 times at each level, whereas the adapted LEV2 mesh has only increased by 3 times after 2 adaptivity passes. This indicates significant savings in the computational resources if the results of these meshes are similar, which we will investigate next.

Since the C_p values predicted by the initial mesh agree well with the experiments for 17% and 50% spanwise sections, we only plot the C_p values for the 98% span section in Figure 5.9 for the meshes created with nested refinement. As the mesh is uniformly refined, The C_p values gradually start showing better agreement with the experiments with the refined LEV2 mesh showing the best results. The adapted LEV2 and the refined LEV2 mesh both give comparable C_p values and are in reasonable agreement with the experiments for all the three elements.

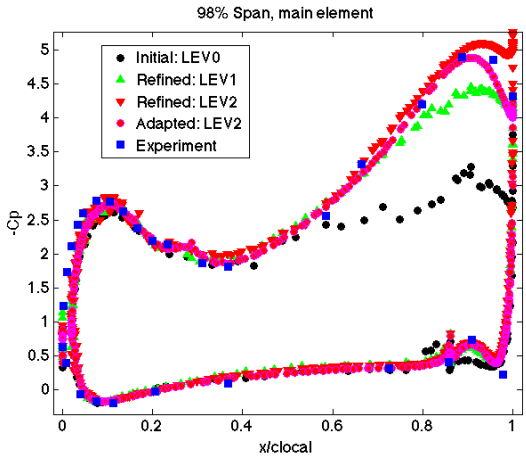
Figure 5.10 shows iso-surfaces for the Q criterion for the refined LEV2 and the adapted LEV2 meshes. The shape of the tip vortex is similar for both of the meshes. The size of the vortex given by the refined LEV2 mesh is smaller than that of the adapted LEV2 mesh, indicating that the adaptation gives a little bit better resolution in this area over the refined mesh and one more refinement cycle might be required to arrive at the same result.

From the vorticity contours plotted in Figure 5.11, it can be seen that both of the adapted and the refined LEV2 meshes give similar resolution in the tip area. The off body vortices are captured to a satisfactory degree by both of the meshes.

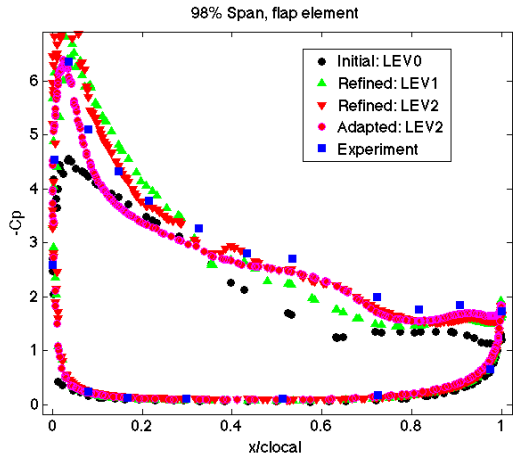
Overall, above results show that the adaptive boundary layer meshing technique provides an easy and an efficient way to generate satisfactory meshes for capturing the required flow features for complex geometries which would otherwise require careful selection of mesh generation parameters. The savings in computational resources are significant in terms of the number of elements in the



(a) Slat element: 98% span



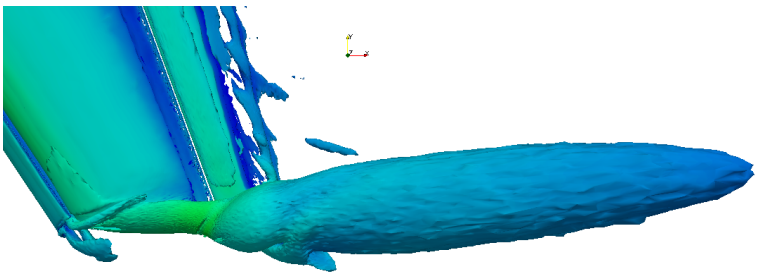
(b) Main wing: 98% span



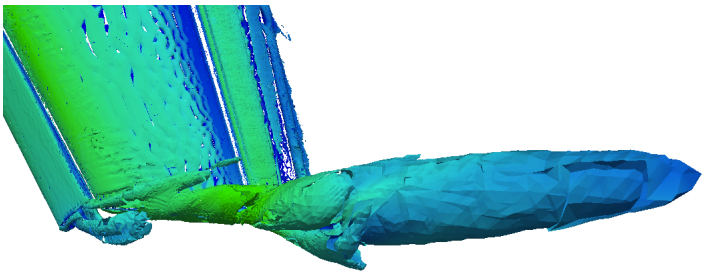
(c) Flap element: 98% span

Figure 5.9: Coefficient of pressure for 98% span

mesh when compared with the nested refinement approach to get similar results. We saw that the adapted LEV2 mesh with 12M elements gave similar results to the refined LEV2 mesh with 140M. This reduction in the number of elements by a factor of more than 10 can be attributed to the anisotropy in the adapted mesh due to stretching of the elements in particular directions and to selective refinement based on error indicators. This is a good marker that adaptive approach is better in getting a good mesh capable of generating better results efficiently.

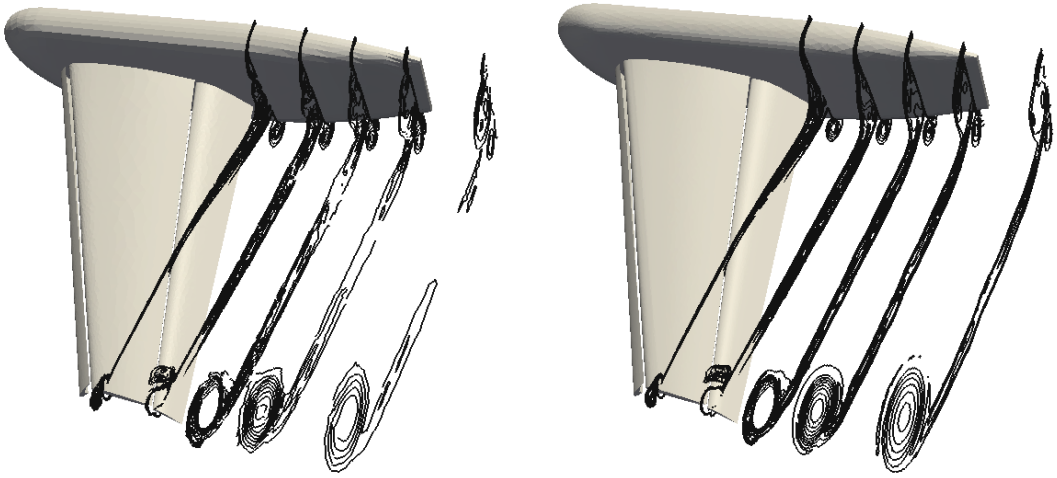


(a) Initial mesh: LEV0



(b) Adapted mesh: LEV2

Figure 5.10: Iso-surfaces of Q criterion for NASA trap wing showing the tip vortex



(a) Refined mesh: LEV2

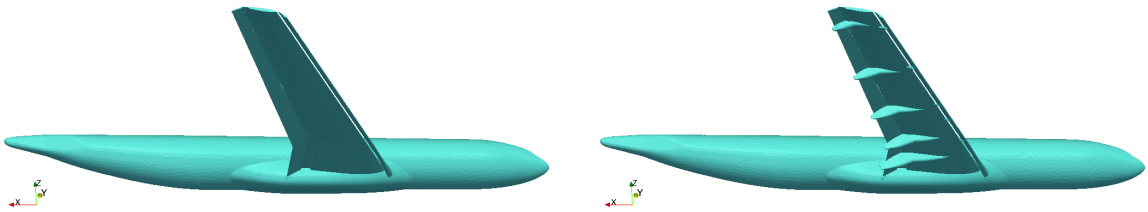
(b) Adapted mesh: LEV2

Figure 5.11: Iso-surfaces of Q criterion for NASA trap wing showing the tip vortex

5.2 EUROLIFT DLR-F11 High Lift Configuration

The second test case for in-plane adaptivity of multi element wings is the DLR-F11 wing, which is used for analysis in the 2nd high lift prediction workshop [2]. The wing is a multi element

wing similar to NASA trap wing, but more complex and realistic in nature. This geometry has two configurations, one with flap fairings and slat tracks (config 4) and other without them (config 2). The experimental results can be found in reference [59].



(a) Config 2 (b) Config 4

Figure 5.12: Geometry of the DLR-F11 wing

Figure 5.12 displays the geometries for the config 2 and config 4 of the DLR-F11 wing, clearly showing the fuselage, slat, flap, and flap fairings for config 4. The problem definition is given in Table 5.3. We use the incompressible flow solver since the Mach number is fairly low. RANS Spalart-Allmaras [71] turbulence model is used similar to other test cases in this thesis. The simulations were run on BG/Q (Mira) supercomputer at Argonne and on Janus supercomputer at UCB, with as many processors as 64k.

Mach number	Mean aerodynamic chord (MAC)	Re_{MAC}	Angle of attack
0.175	0.347 m	15.1 million	7°

Table 5.3: Problem definition for DLR-F11 wing

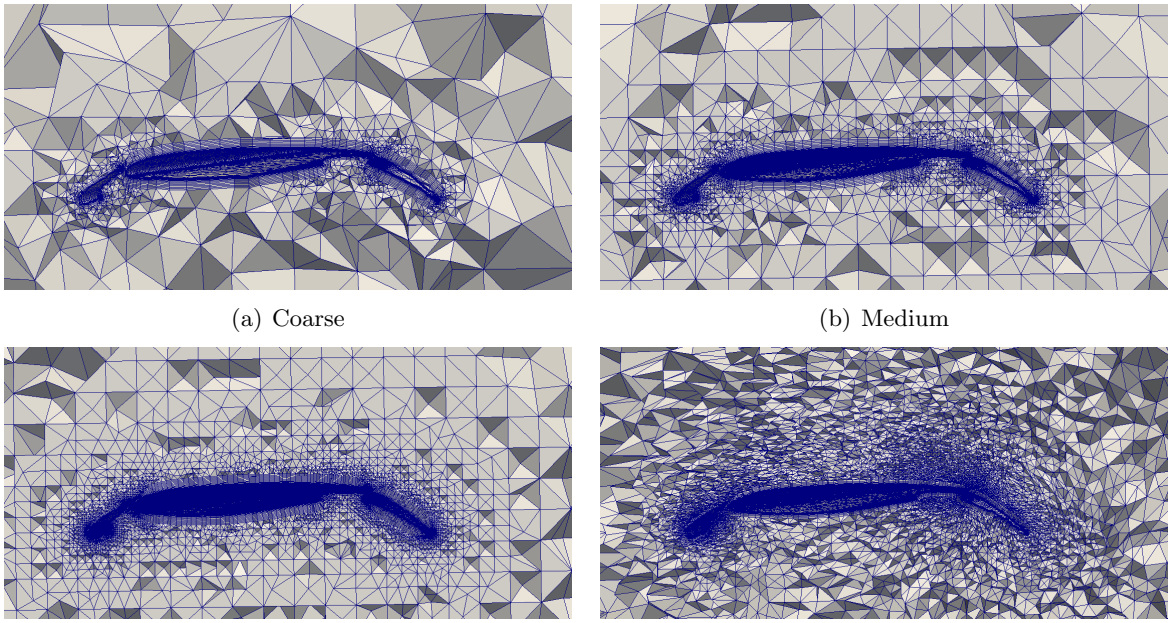
Mesh	# of elements	# of vertices	First cell height (m)
Coarse mesh	32.28M	13.55M	5.5e-7
Medium mesh	91.55M	37.34M	3.7e-7
Fine mesh	287.89M	112.96M	2.4e-7
Adapted mesh	40.69M	14.11	5.5e-7

Table 5.4: Computational comparison of meshes

We ran in-plane adaptivity on an extra coarse mesh for the config 2 geometry. The adapt

strategy was the combined approach similar to the one used for NASA trap wing. Another set of meshes were created, namely coarse, medium and fine to match the resolutions required by the 2nd high lift prediction workshop [2].

Table 5.4 shows the comparison of the different meshes used for analysis. The number of elements increase roughly by 3 times from coarse to medium to fine meshes. The first cell height is also reduced in each subsequent mesh. The number of elements and vertices in the adapted mesh are on the same order as of the coarse mesh.



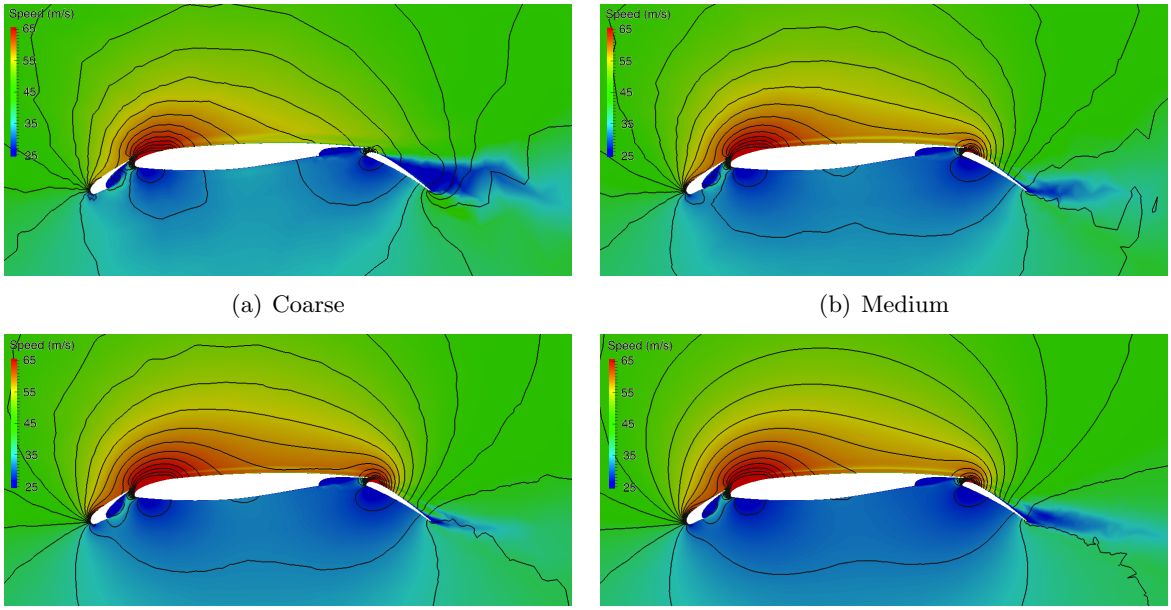
(a) Coarse (b) Medium

(c) Fine (d) Adapted

Figure 5.13: Cut views of different meshes for the DLR-F11 wing

Figure 5.13 and Figure 5.14 show cut view pictures of the meshes at mid-span section and their corresponding speed contours (in color) and pressure contours (black lines). The coarse mesh over predicts the separation over the flap but the medium, adapted and fine meshes show similar behavior. The pressure contours are also jagged and irregular for the coarse mesh, but become smoother for other meshes.

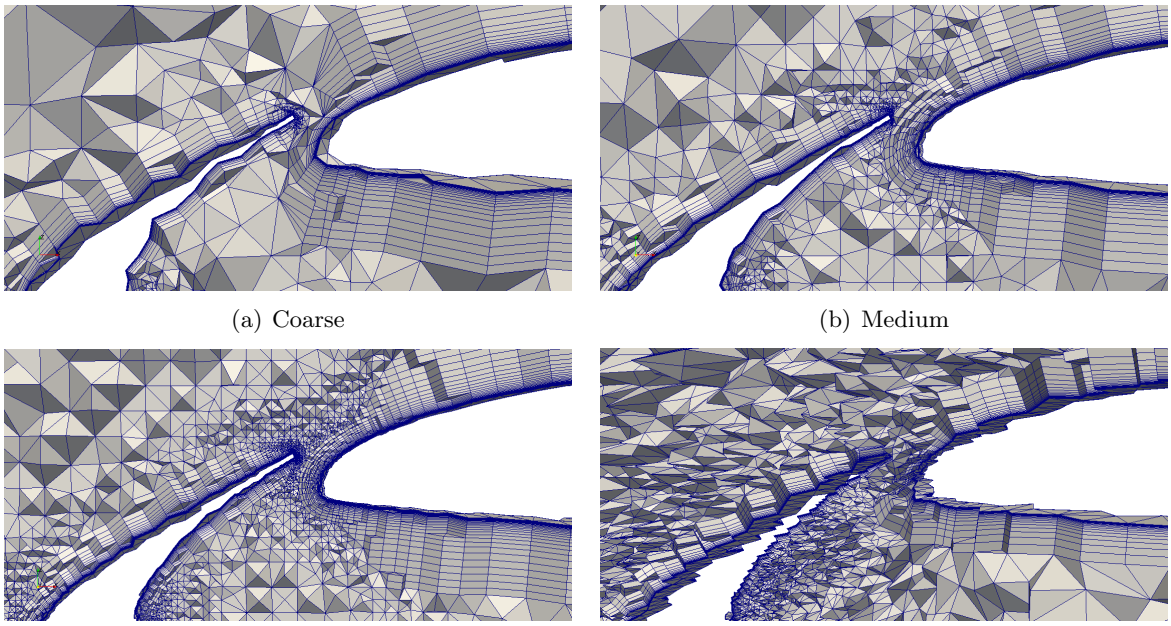
Figure 5.15 shows cut views of the meshes near the nose of the main wing and the trailing edge of the slat. Increasing curvature refinement and refinement in the trailing edge region can be



(a) Coarse (b) Medium

(c) Fine (d) Adapted

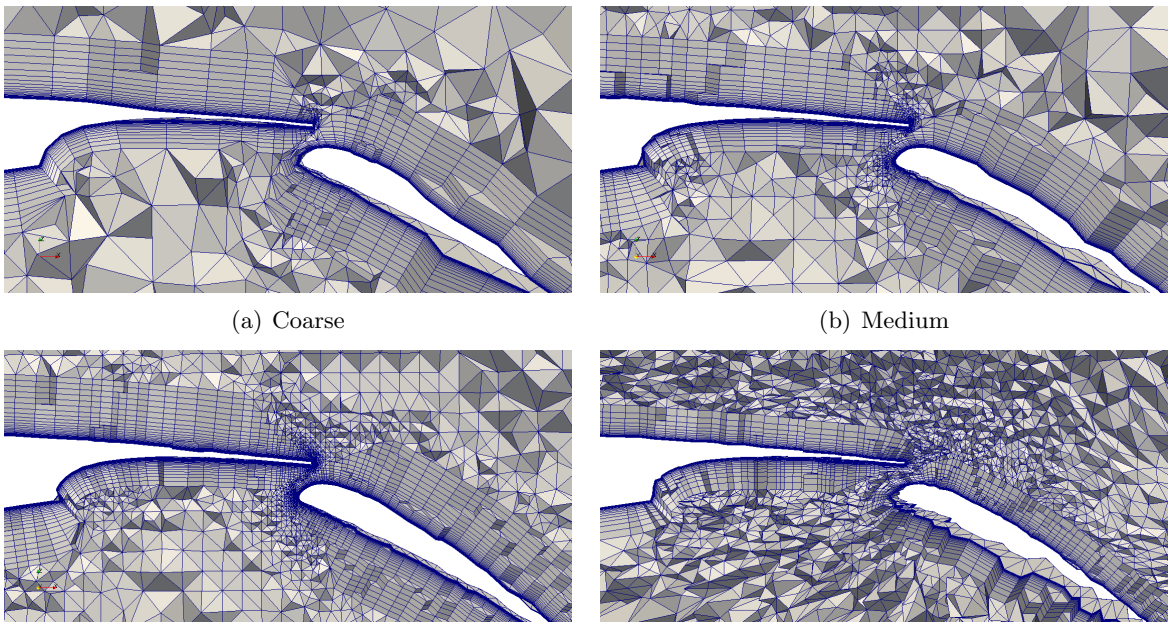
Figure 5.14: Speed contours (colored) and pressure contours (black lines) for different meshes of the DLR-F11 wing



(a) Coarse (b) Medium

(c) Fine (d) Adapted

Figure 5.15: Cut views of the region near slat trailing edge and the nose of the main wing for different meshes of the DLR-F11 wing



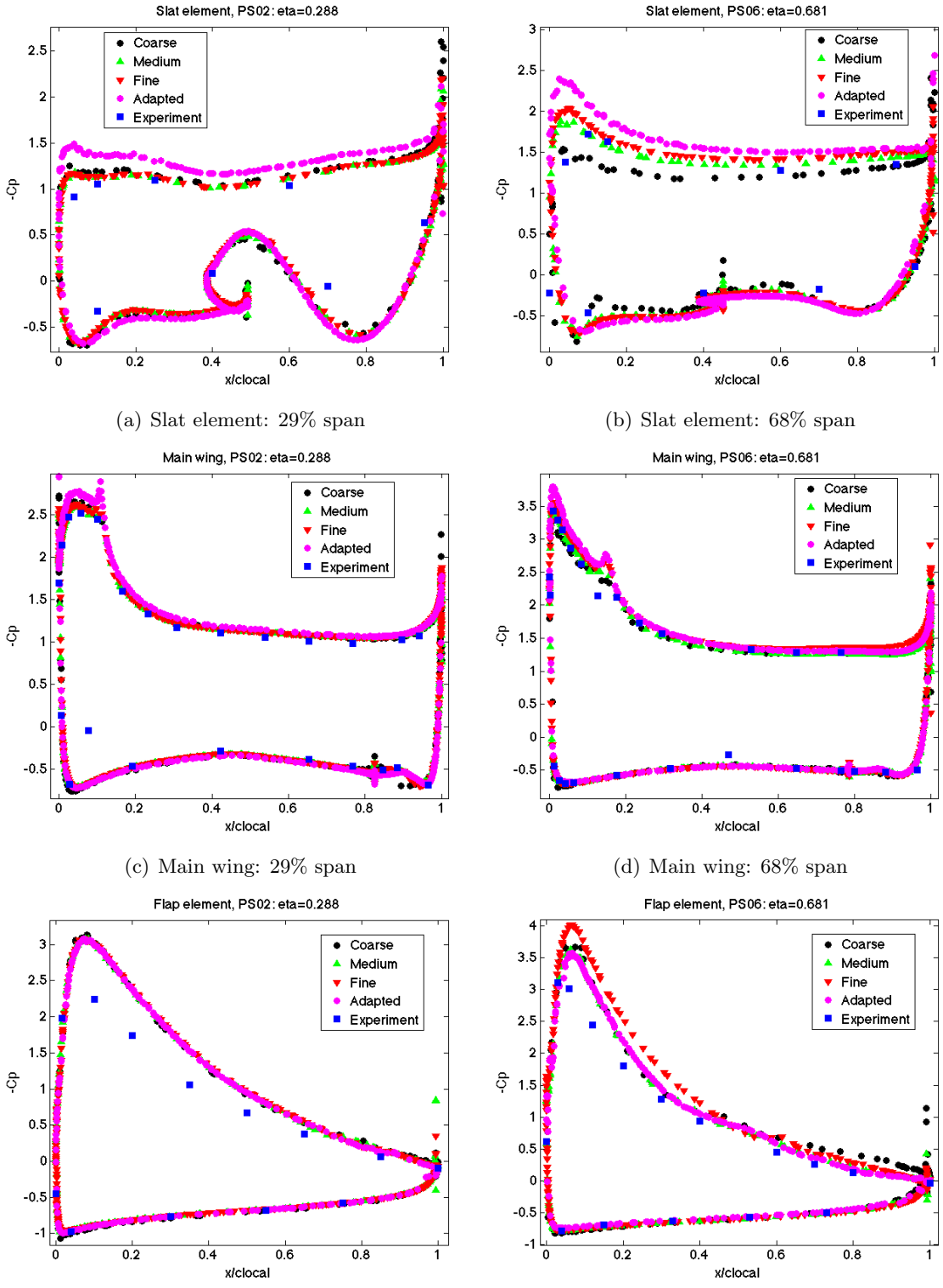
(a) Coarse (b) Medium

(c) Fine (d) Adapted

Figure 5.16: Cut views of the region near flap trailing edge and the nose of the flap wing for different meshes of the DLR-F11 wing

clearly seen from the coarse to the medium to the fine mesh. The anisotropy in the elements can be seen for the adapted mesh. Figure 5.16 shows cut views of the meshes near the trailing edge of the main wing and the nose of the flap to display progressive refinement in the boundary layer region. The adapted mesh shows anisotropy in the streamwise and the spanwise direction getting developed.

Figure 5.17 plots the coefficient of pressure at the 29% and 68% spanwise sections on the slat, the main wing and the flap. Note that $-C_p$ is plotted on the Y axis. For the slat, the adapted mesh over predicts the suction pressure relative to other meshes but gives a better agreement than the coarse mesh on the pressure side. For 29%, all curves lie pretty much on top of each other except for the adapted results. For 68%, the coarse mesh under predicts the pressure on the suction side and the adapted mesh over predicts it. The medium and the fine mesh are in a good agreement with the experiments. However, the adapted mesh gives better agreement than the coarse mesh on the pressure surface.



(a) Slat element: 29% span

(b) Slat element: 68% span

(c) Main wing: 29% span

(d) Main wing: 68% span

(e) Flap element: 29% span

(f) Flap element: 68% span

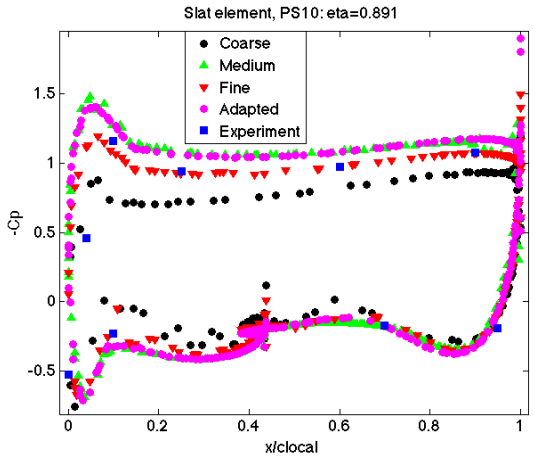
Figure 5.17: Coefficient of pressure for 29% and 68% span for the DLR-F11 wing for 7° angle of attack

For the main wing, all meshes give good agreement with the experimental results for the 29% and 68% sections. The adapted mesh over predicts the suction peak by a small amount. A good thing to note is that all the meshes are able to capture the effect of separation on the pressure, near the trailing edge.

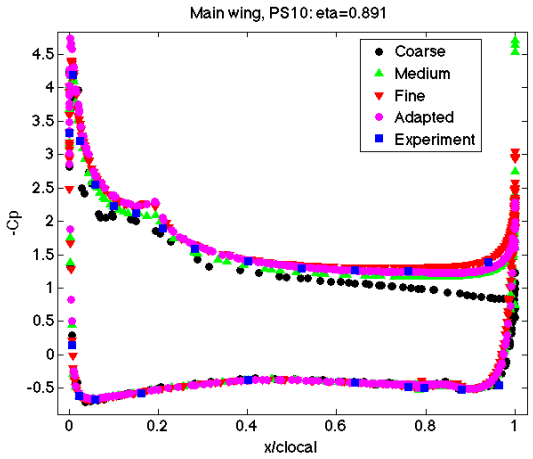
All meshes give deviating results from the experiments for the flap 29% section, but agree well with each other. This indicates that we have mesh convergence but the turbulence model is unable to capture the effect of separation near the root of the wing, leading to the conclusion that we have verified results for this particular turbulence model. For the 68% section of the flap, the fine mesh overshoots the pressure peak on the suction side as compared to other meshes, but other meshes agree with each other. Near the trailing edge, the coarse mesh predicts more separation compared to the experiments and other meshes.

The outboard coefficients of pressure at 89% span section are plotted in Figure 5.18. The coarse mesh shows degrading performance and does not give matching values to the experiments for all the three elements. For the slat, it under predicts the pressure on the suction surface for all of the section. Interestingly, the fine mesh gives a good agreement with the experimental data but the medium and the adapted mesh both over predict the pressure on the suction side by a small amount. On the pressure surface of the slat as well, adapted mesh performs better than the coarse mesh.

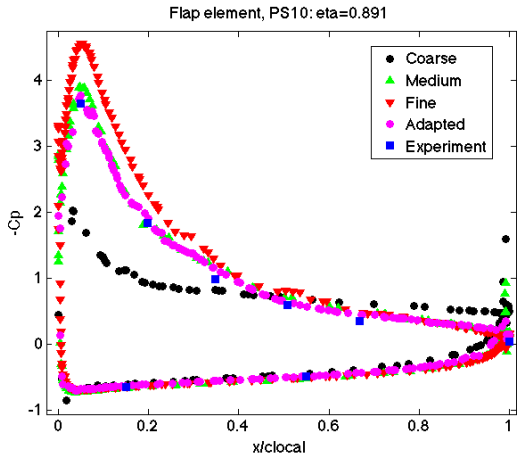
For the main wing section (Figure 5.18(b)), the coarse mesh gives continuous increasing pressure (sustained adverse pressure gradient) on the suction side near the trailing edge indicating attached behavior which does not agree with the experiments. The medium, fine and the adapted mesh all are in good agreement with the experiments and capture the effect of separation near the trailing edge to a good degree. For the flap section, the coarse mesh does a poor job at capturing the suction pressure peak and gives a flatter curve which does not match the experiments. The medium and the adapted meshes give values that lie on top of each other and agree well with the experimental values. The fine mesh, however, overshoots the pressure peak slightly, indicating that these meshes bound the experimental values. Near the trailing edge, except for the coarse mesh,



(a) Slat element: 89% span



(b) Main wing: 89% span



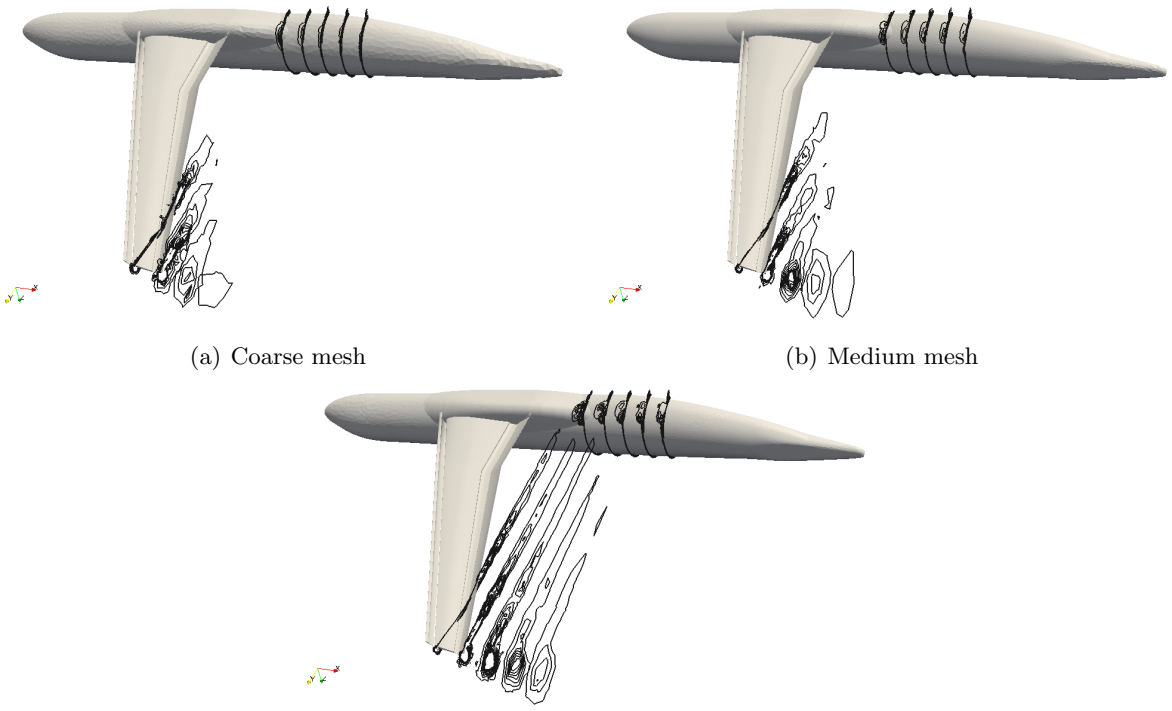
(c) Flap element: 89% span

Figure 5.18: Coefficient of pressure for 89% span

other meshes show attached behavior like the experiments.

The vorticity contours on planes with X normals are plotted in Figure 5.19 for the coarse, medium and the adapted meshes. The adapted mesh undoubtedly gives better vorticity contours in the tip region compared to others. Adaptive refinement in this region is high and the increased resolution helps in capturing the flow features. The coarse mesh is unable to resolve the off body vortices, but they can be seen in the medium and the adapted mesh.

Overall, above results show that the adapted mesh which has comparable mesh size (in terms of number of elements) to the coarse mesh performs better in almost all aspects. It also performs



(a) Coarse mesh

(b) Medium mesh

(c) Adapted mesh

Figure 5.19: Vorticity contours on planes with normals in X direction for the DRL-F11 wing

as good as the medium mesh or even better in some aspects. The medium mesh is around 3 times larger than the adapted mesh, so this shows significant savings in computational resources. This also underlines the effectiveness and efficiency of the adaptivity approach to reach a desired mesh for complex geometries. Some results indicate that to arrive at completely verified results, another adaptivity pass needs to be performed.

In this chapter, through adaptivity results of NASA trap wing and DLR-F11 wing, we have showcased the efficiency of our adaptivity approach. This is a significant development for boundary layer adaptivity in terms of handling complex geometrical domains as well as working with complex turbulent flows, which are both true in the cases of these multi element wings.

Chapter 6

Application Results: In-plane Only Adaptation in Parallel

As a part of this thesis, in-plane adaptivity of complex turbulent flow problems was carried out in parallel. The work helped in extending the adaptivity approach to parallel systems and scaling studies were also undertaken. This chapter illustrates two internal flow application results of in-plane adaptivity for turbulent flows using parallel processing. The boundary layers are not adapted in the thickness direction, as parallelization of the thickness adaptation procedures is still under development.

All the parallel in-plane adaptations in this chapter were performed with an open source software based on the mesh adaptation procedures outlined in [21], [53], [61] developed at Rensselaer Polytechnic Institute in the Scientific Computation Research Center (SCOREC) [66].

6.1 Heat Transfer Manifold

The heat transfer manifold test case consists of a large diameter cylindrical pipe as the inlet, a relatively thin and flat manifold section, and twenty small outlet pipes. The water flow through this geometry was modeled as steady, incompressible with RANS Spalart-Allmaras turbulence model [71]. A turbulent velocity profile was used at the inflow, with Reynolds number of 1 million, based on the inflow diameter. No-slip boundary conditions were assumed at walls and weakly applied zero pressure at all outlets. The solution parameter used in the Hessian calculation is the static pressure combined with scaled dynamic pressure, defined as $P + 0.5\alpha\rho u^2$, where the scaling factor $\alpha = 0.2$ was chosen to achieve an appropriate balance of the static and dynamic pressure in terms of adaptive

refinement. Eigenvalues were computed from the Hessian matrix and were scaled appropriately to get the mesh sizes along with the three orthogonal directions.

The adaptive loop which consists of a flow solve and a mesh adaptation within each cycle was carried out twice, and at each cycle, the flow solve was started from previous loop's solution. This speeds up the convergence and eliminates the need to start from zeroth time step. Each cycle was divided into 1000 time steps with a constant time step size of $0.1s$. The computations use a pre-defined boundary layer mesh, with layers of constant thickness and isotropic elements in interior volume, consisting of 472k elements in total. The adapted boundary layer mesh consists of 16.3M elements. The mesh adaptation procedure was applied one time for each of the two analysis cycles.

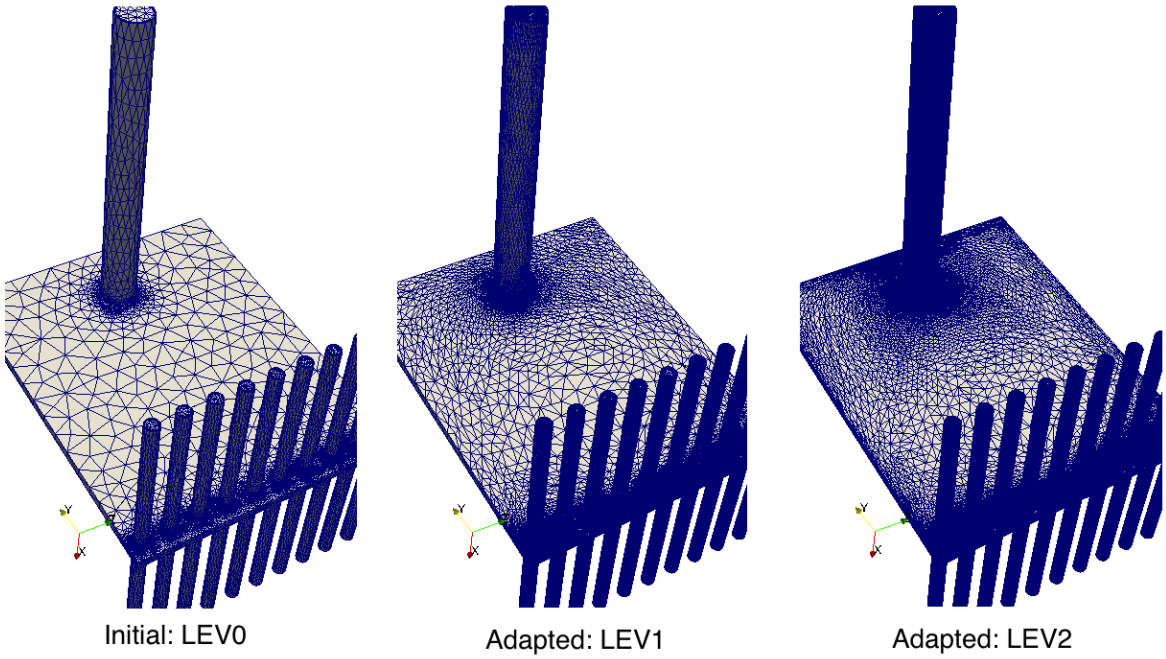


Figure 6.1: Initial and adapted meshes for the heat transfer manifold

Figure 6.1 shows the initial LEV0 and the adapted LEV1 and LEV2 meshes for the heat transfer manifold. Figure 6.2 shows the view of the meshes from the bottom and corresponding pressure distributions. The adaptive refinement in the stagnation region of the inlet pipe and the manifold can be seen, clearly. The pressure distributions show improvement in the solution because

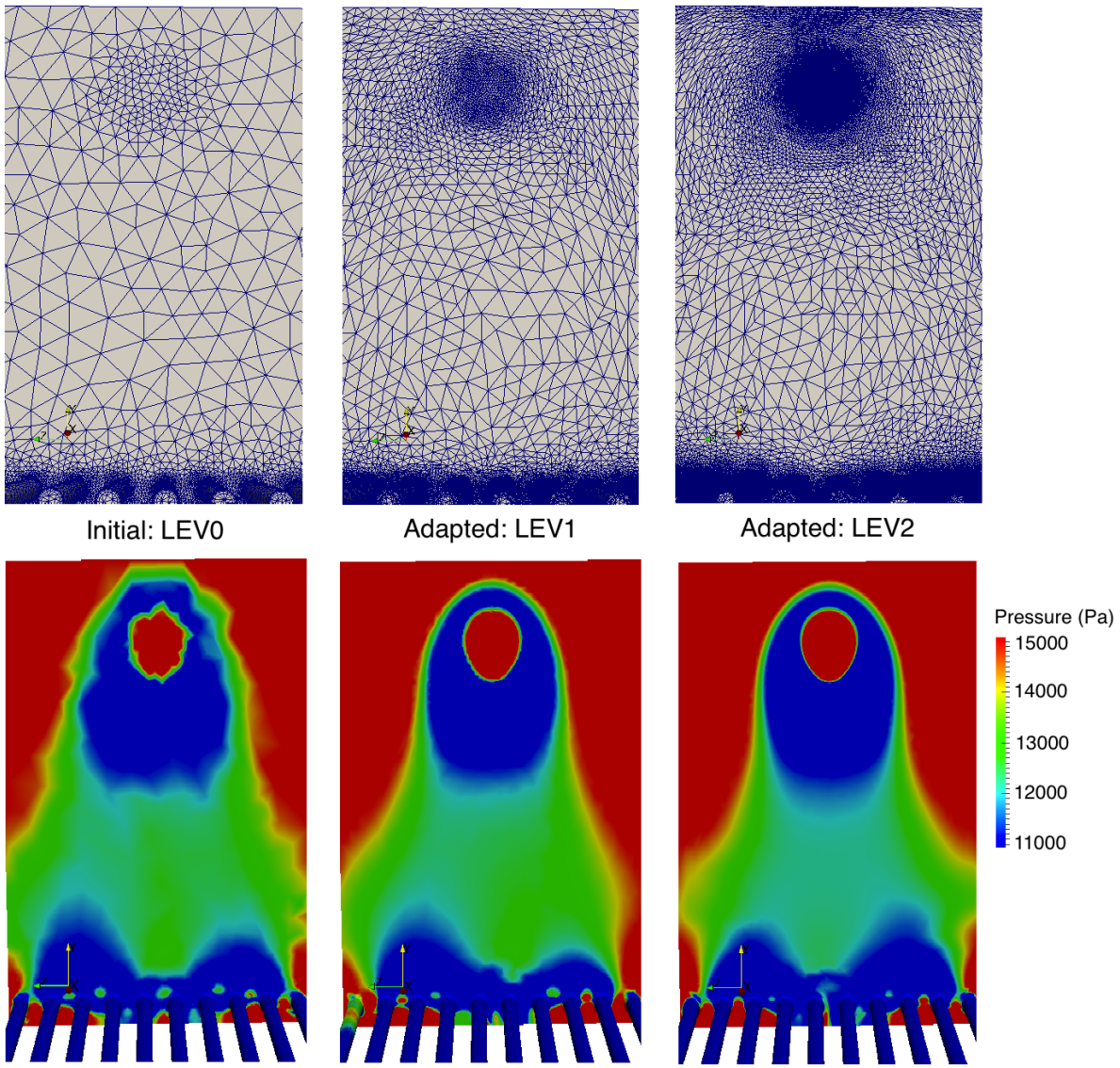


Figure 6.2: Meshes and pressure distribution (view from bottom) for the heat transfer manifold

of refinement.

Figure 6.3 shows cut views of the meshes and corresponding pressure slices near the intersection of the inlet pipe and the manifold. Similarly, Figure 6.4 shows the cut views of the meshes and corresponding pressure slices near the intersection of one of the outlet pipes and the manifold. The initial mesh is too coarse and these figures demonstrate its inability to accurately capture the flow phenomena. Critical flow regions including stagnation point and turns around the fillets of the

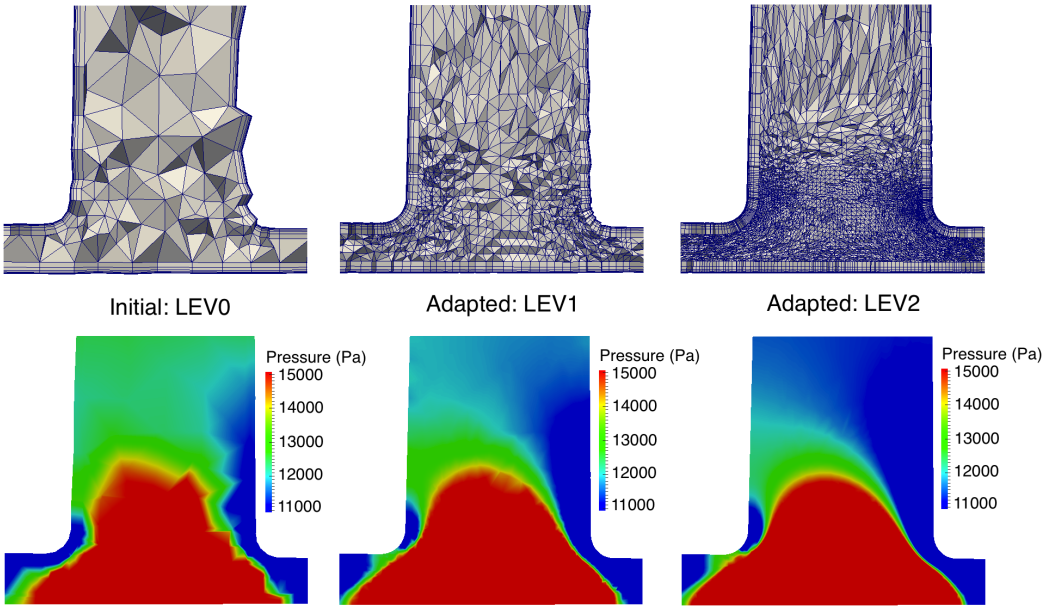


Figure 6.3: Cut view of the inlet pipe and the manifold

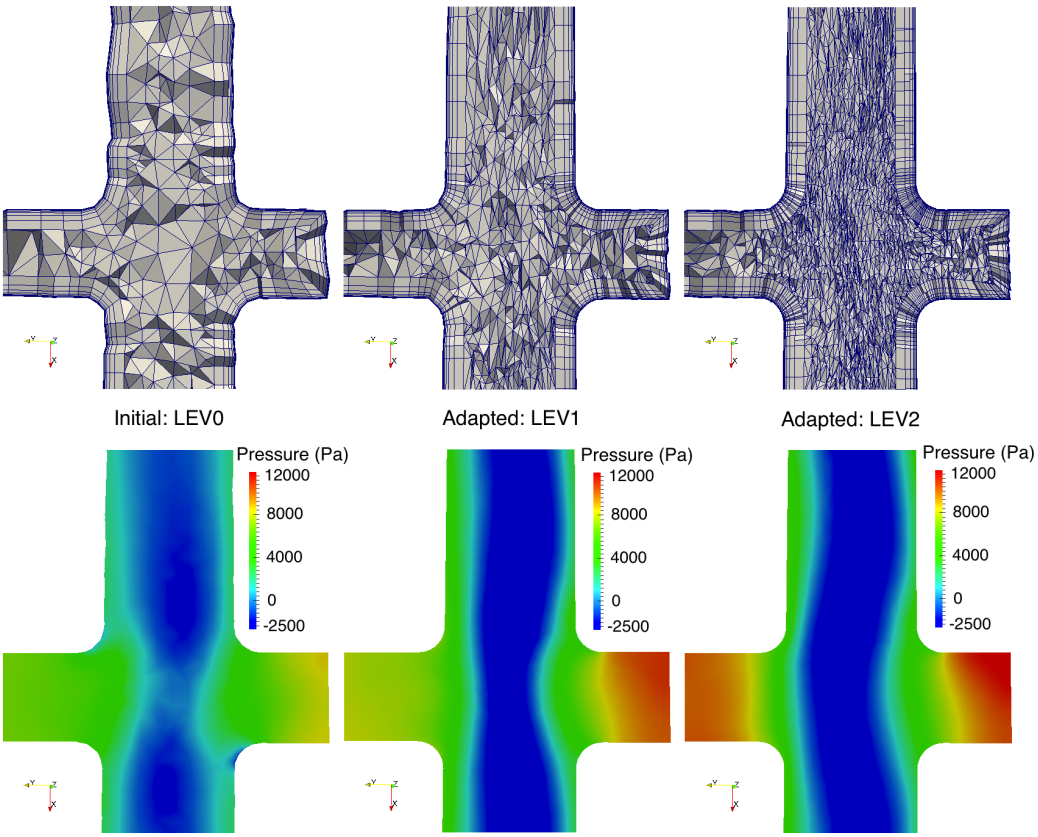


Figure 6.4: Cut view of one of the the outlet pipe and the manifold

pipes get significantly refined and that is reflected by smoother and more accurate solution results. The walls of the manifold, especially the wall closest to the inflow pipe, gets refined to a higher degree. The fillets of outflow pipes also get more refinement. The central part of the manifold relatively gets lesser refinement because of relatively lesser gradation in solution fields. Moreover, away from flow regions with stagnation and turns, flow anisotropy in the streamwise direction is present that is accounted with highly anisotropic mesh elements, which results in significant computational savings.

6.1.1 Comparisons with *a priori* Mesh Generation

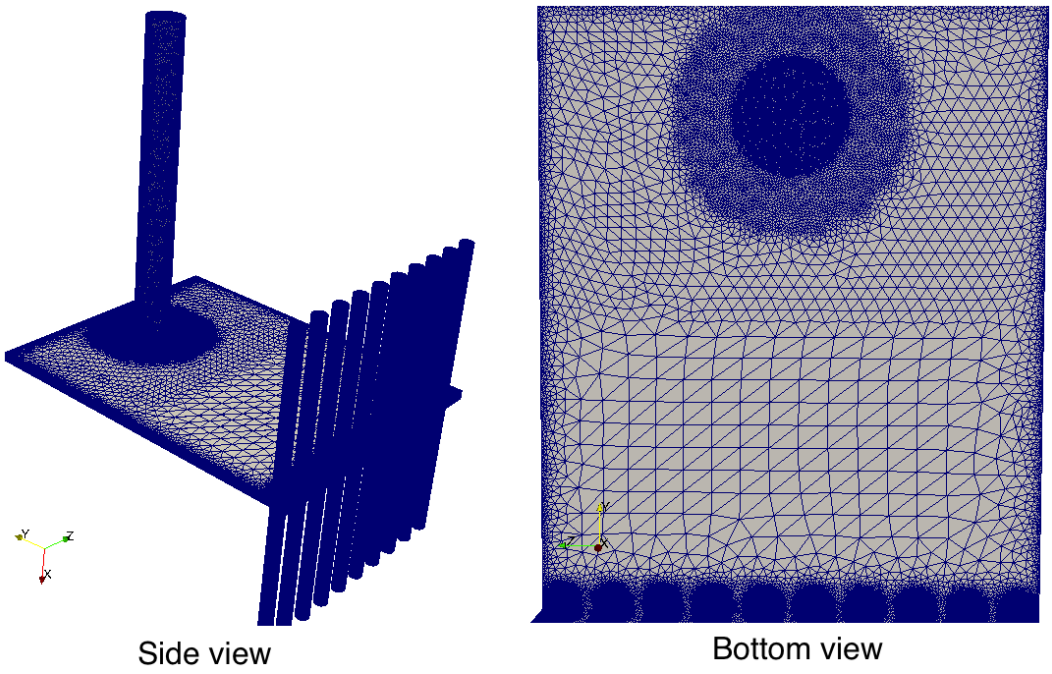


Figure 6.5: *a priori* fine mesh generated with some knowledge of adaptive refinement

To display the effectiveness of anisotropic adaptivity, a fine mesh was generated by anticipating where the mesh would need more resolution (like stagnation points and fillets). Refinement zones were created in the regions where adaptive strategy showed refinement, to get the required fine mesh sizes, and this mesh is called *a priori* fine mesh. Figure 6.5 shows the side view and the view from the bottom for this mesh, showing the preemptive refinement of the stagnation region

and the outflow pipes along with corners and blends.

The flow computations were performed with similar settings as for the adapted and the initial cases. Mesh convergence analysis was performed by plotting the pressure drop between the inflow and the outflow faces of the pipes.

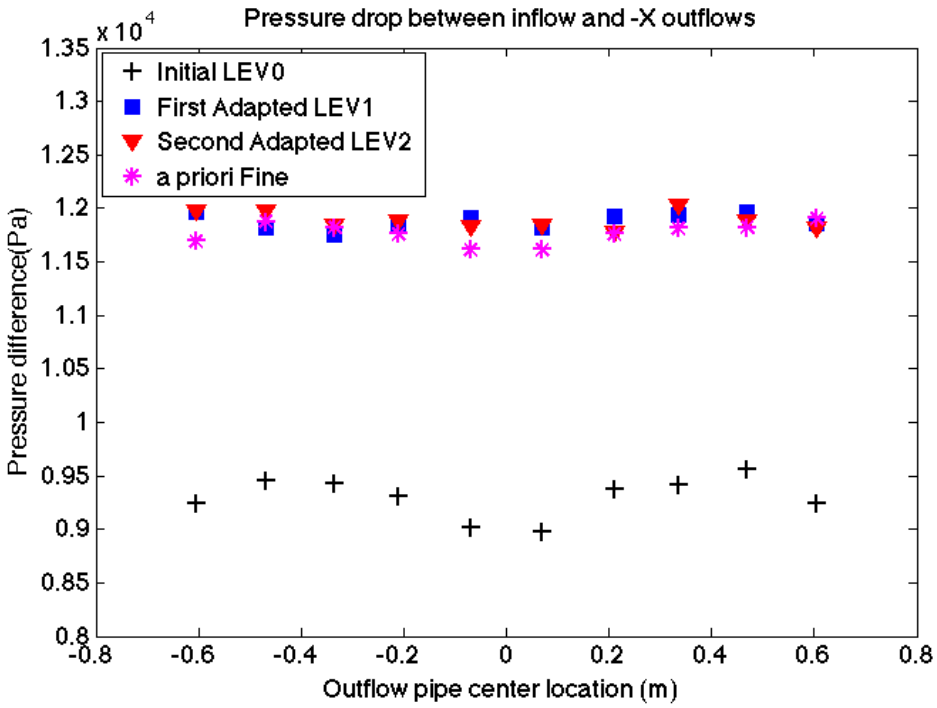


Figure 6.6: Pressure drop across inlet and outlet pipes for bottom outlet pipes with +X normals

Figure 6.6 and Figure 6.7 show the pressure drops for bottom and top outlet pipes with respect to the inflow, respectively. The initial mesh predicts completely wrong values but with adaptivity the values seem to converge to similar values. The adapted meshes predict close values to the *a priori* mesh's prediction. The maximum percentage difference between the values given by the adapted LEV2 and the *a priori* mesh is less than 2%.

Table 6.1 gives a computational comparison of the various meshes used. It can be clearly seen that with adaptivity, momentum equation's residual decreases, which points towards better convergence. The number of elements in the adapted LEV2 mesh is nearly half that of the *a priori* fine mesh, but both give comparable residuals. This difference in the mesh size can be attributed

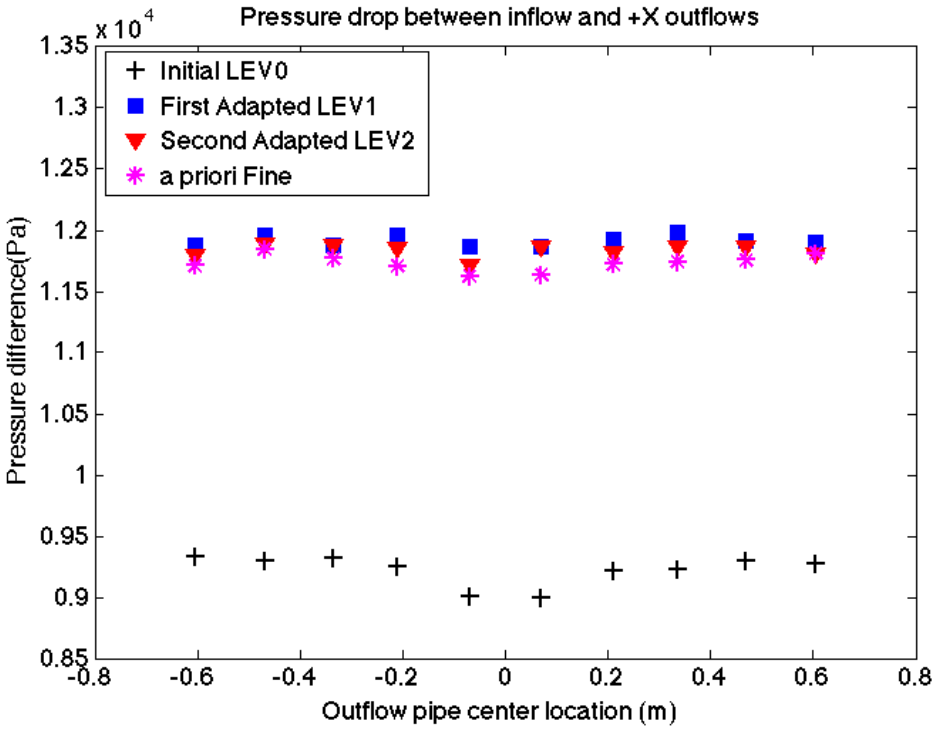


Figure 6.7: Pressure drop across inlet and outlet pipes for top outlet pipes with -X normals

	# elements	# processors	Momentum equation residual	Time for 100 time steps (s)
Initial mesh: LEV0	472k	16	1e-5	4032
Adapted mesh: LEV1	2.8M	128	3e-6	9242
Adapted mesh: LEV2	16.3M	256	9e-7	15130
a priori fine mesh	33M	256	8e-7	17620

Table 6.1: Computational comparison of meshes

to the anisotropic adaptation which results in significant savings in computational resources, due to orienting and stretching the elements in particular directions.

6.1.2 Scaling Studies

To evaluate the parallel performance of the boundary layer mesh adaptivity, a strong scaling study was conducted, where mesh adaptation in the second cycle of adaptive loop was executed on a

range of processors: 256 to 4,096 cores. The scaling is based on the execution time on 256 processors and is defined as $(n_{proc-base} * time_{base})/(n_{proc-test} * time_{test})$, where $n_{proc-base}$ is the number of base processors, $time_{base}$ is the execution time on the base number of processors, $n_{proc-test}$ is the number of test processors and $time_{test}$ is the execution time on that many number of processors. All available cores per node were requested during the adaptation runs. Table 6.2 gives the scaling of second cycle mesh adaptation run times with the initial mesh of 16M elements, and the adapted one consisting of 81M elements.

# procs	256	512	1024	2048
Time (s)	1194.34	785.44	514.45	421.09
Scaling	1	0.76	0.58	0.36

Table 6.2: Strong scaling study for the heat transfer manifold conducted on NERSC Hopper [20] and Blue Gene/P [33]

As indicated in Table 6.2, the mesh adaptation times decrease with the increased number of cores. As the given mesh is distributed to more processors, there is little computation performed during mesh modification operations relative to the substantial increase in communications, the scaling decreases on high core counts (note that a strong scaling study is performed and therefore, the problem size is fixed). However, the analysis has been shown to scale strongly with the similar amount of workload. The actual time required for mesh adaptation is not even 1% of the flow solver time. This difference increases further for unsteady simulations. Because of this, even though the strong scaling of mesh adaptation software is not good, the ability to adapt at higher number of processes is very desirable. This is also because at such massively parallel scales, it is impossible to bring the mesh back on a serial processor just for adaptation. So the ability to adapt on parallel systems is more important than its scaling, to aid the flow simulations.

# procs	256	1536	9216
# elements / proc	62.5k	62k	65k
Time (s)	1194.34	1381.55	1716.23
Scaling	1	0.97	0.86

Table 6.3: Weak scaling study for the heat transfer manifold conducted on NERSC Hopper [20] and Blue Gene/P [33]

Another important scaling study is that of the weak scaling. Most of the time during adaptation, mesh is refined far more than it is coarsened, resulting in an increase in the size of the mesh. This mesh is further partitioned to higher number of processors to run further simulations and adaptivity. Since we are increasing the problem size at each adaptation loop, weak scaling is actually more important than the strong scaling. Table 6.3 shows the results of the weak scaling, where number of elements per processor are kept nearly constant. A scaling of 0.86 is seen at 9216 processors which is considered very good.

6.2 3D Shallow Cavity

The second case study for parallel in-plane adaptation was carried out for a 3D shallow cavity flow. The flow is modeled as an unsteady incompressible air flow. Air enters through the long inlet pipe and enters the shallow cavity which has a length to depth ratio of 6. This causes the boundary layer from the inlet pipe to form a free shear layer in the cavity which shows vortex shedding as the shear layer starts wobbling. The flow exits through a long outlet pipe.

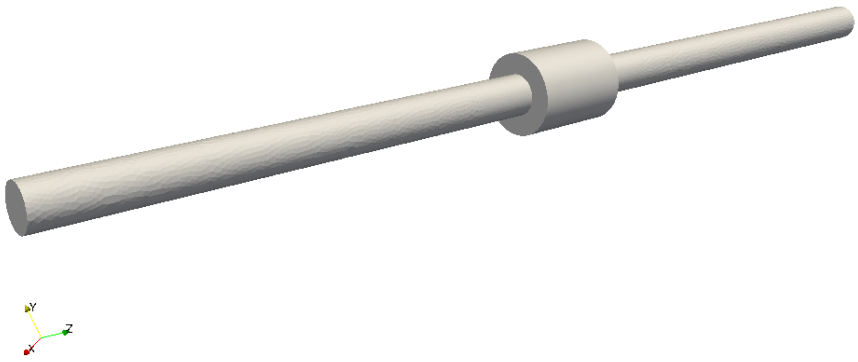


Figure 6.8: 3D shallow cavity geometry setup

To resolve the unsteady turbulent scales, DDES (delayed detached eddy simulation) [72]

turbulence model was used. This model is a hybrid of RANS and LES which behaves like RANS in the regions near the walls and in regions with insufficient resolution and behaves like LES elsewhere where the mesh resolution is fine enough. As mesh is refined in different areas, LES turns on automatically in these regions and the scales are captured much more effectively than RANS.

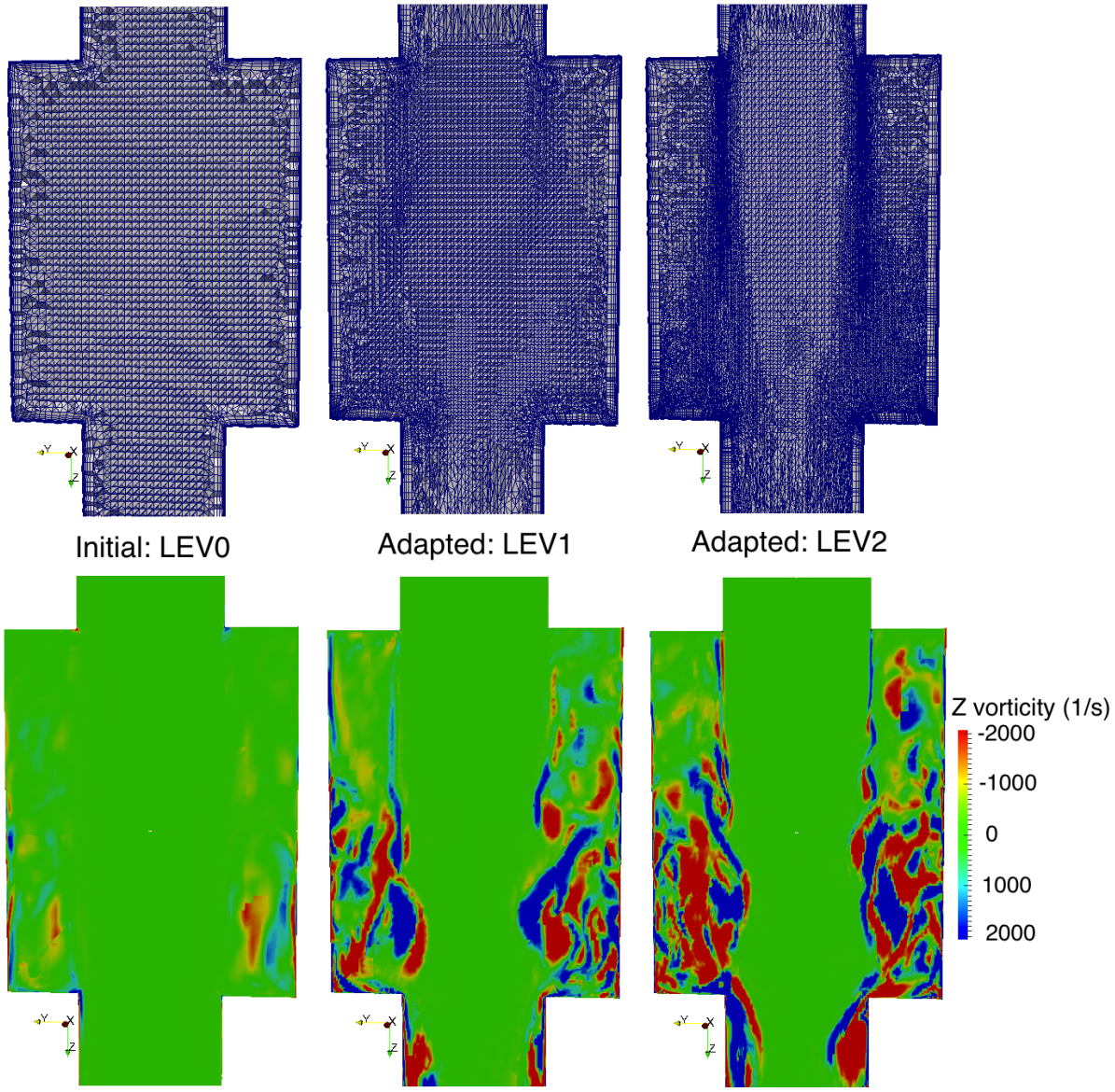


Figure 6.9: LEV0 (left), LEV1(middle) and LEV2(right) meshes and a slice showing corresponding stream wise vorticity

For adaptivity, Hessians of time averaged velocity were used to calculate the anisotropic

sizes because the main region of interest is the free shear layer, where velocity gradients are high. Since this is an unsteady flow, the number of time steps required to give a good time average for adaptivity was carefully chosen. Even though the real aim of adaptivity is to capture the unsteady scales more effectively, the averaging must be done in such a manner that the mesh should not really reflect the unsteady structures. We chose a time interval of 5 times the time required for the flow to traverse the length of the cavity for averaging, however, different time intervals may give similar or better results. The main point to note is to not have this interval very small which might reflect in bad adaptivity and an unsteady mesh.

Figure 6.9 shows the cut views of the initial and the adapted meshes in the cavity zone and corresponding vorticity in the streamwise direction. Note that the vorticity is for a particular time step and not time averaged. The refinement is mainly seen in the region of the free shear layer. The smallest edge lengths are encountered very close to the start of the cavity from where the boundary layer separates. The mesh is also refined near the end of the cavity and spans a larger portion of the cavity since the free shear layer has evolved into vortical structures interacting with the wall. The shear layer does roll over till roughly half length of the cavity and this region is also refined. The cavity away from the free shear layer near the start of the cavity does not get a lot of refinement.

The initial mesh does not show a lot of activity of Z vorticity (Z is the streamwise direction), indicating that the initial mesh is inadequate to resolve the shear layer instability and instead provides an artificially steady flow. We do not see a lot of unsteady activity and most of the flow regions are resolved as RANS due to lack of resolution. The error indicators do however recognize that the unsteady features are missed and, with adaptivity, the LEV1 mesh starts showing some signs of unsteady flow and the region near the end of the cavity starts getting resolved by LES. With one more adaptivity pass, we reach good enough resolution such that most of the cavity is modeled with LES (except for the boundary layers) and vortical structures start becoming more prominent in the flow. The breaking of shear layer in vortices can be clearly seen in the adapted LEV2 mesh. Thus, with adaptivity we are able to refine the mesh such that the unsteady structures

in the flow are resolved.

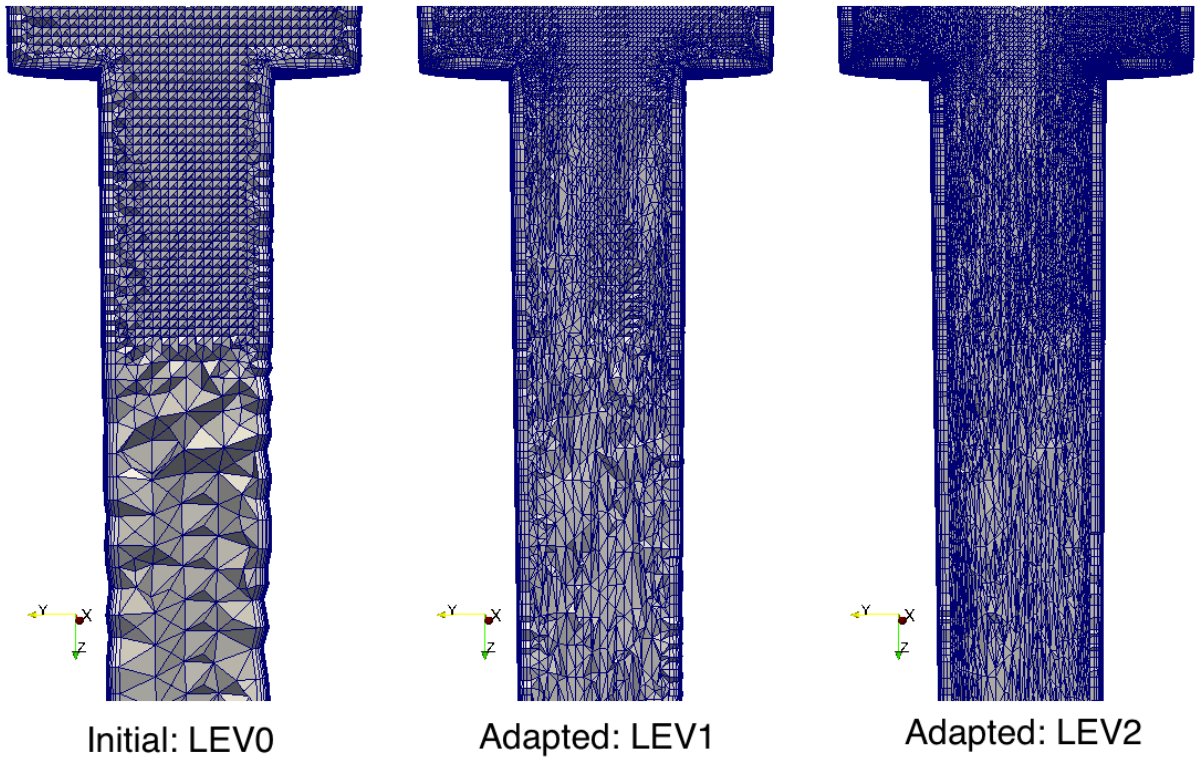


Figure 6.10: Cut view of LEV0 (left), LEV1(middle) and LEV2(right) near the outflow of the shallow cavity

Figure 6.10 shows cut views of the initial and the adapted meshes near the cavity outlet and some portion of the outlet pipe. The in-plane boundary layer adaptation can be clearly seen with anisotropy getting developed in the streamwise direction in the outlet pipe.

Figure 6.11 and Figure 6.12 show the power spectral density plots for pressure and velocity near the end of the cavity. The computations were performed by gathering the time accurate signal at the same point in the domain for several time steps and then applying a *pwelch* window in MATLAB. The spectra show that the power density remains fairly low for the initial mesh. Since the power spectra is calculated by removing the mean signal from each time step, this means that the fluctuations remain fairly low, indicating that the instability is substantially suppressed but not completely eliminated. With adaptivity, the power density increases by over a magnitude. The power spectra for the adapted LEV1 and LEV2 meshes are pretty much coincident indicating

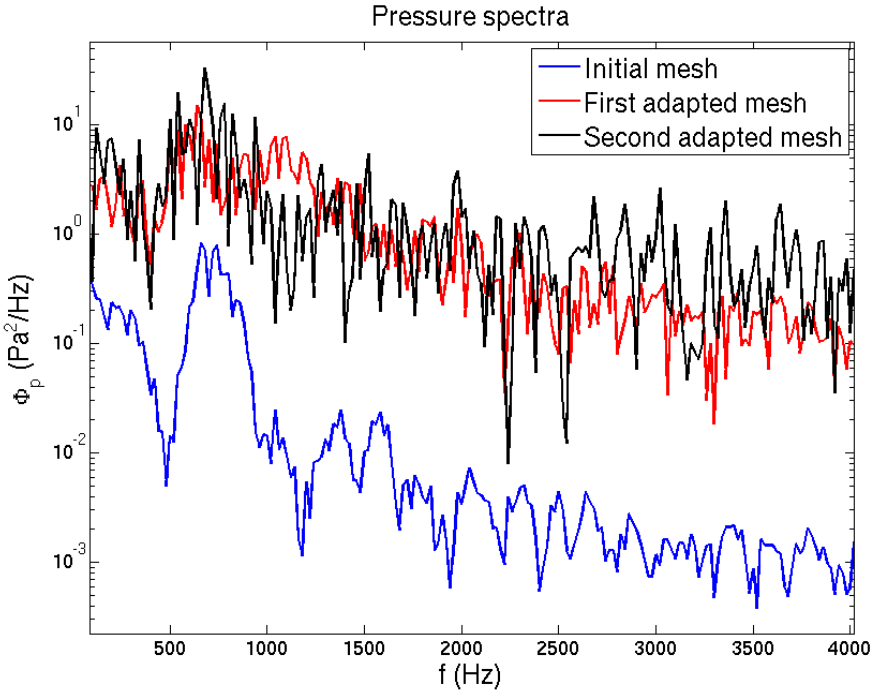


Figure 6.11: Pressure spectra near the cavity exit for the shallow cavity

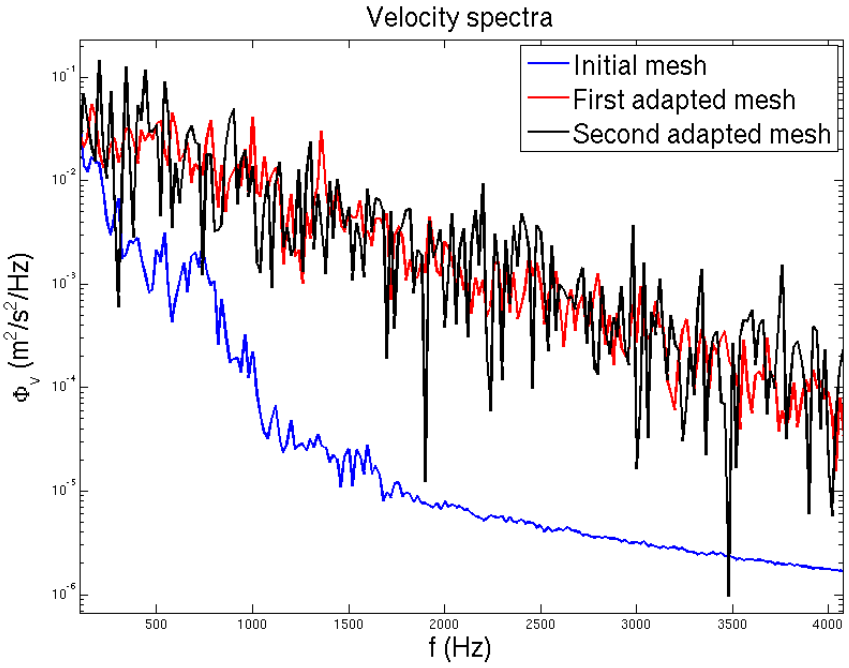


Figure 6.12: Velocity spectra near the cavity exit for the shallow cavity

good mesh convergence. The LEV2 mesh also shows the highest peak in the pressure spectra at a frequency of $680Hz$ which is fairly close the the experimental value of $640Hz$ [28].

Chapter 7

Conclusion and Future Work

7.1 Conclusion

In this thesis, we have extended and applied the adaptive boundary layer techniques to turbulent flows. We developed new techniques to calculate the thickness specifications for the mesh boundary layers, based on the flow solution and combined this approach with the numerical error indicators to drive adaptivity in the entire flow domain. The work has shown promise in correctly calculating the first cell height and the total height of the attached boundary layers. Special considerations were made for separated boundary layers and shock wave - boundary layer interactions where calculating the mesh sizes is complicated. The three applications to aerodynamic flows have shown that this combined strategy can be very effective in capturing the flow physics accurately for both incompressible and compressible flows. The case of NACA 0012 airfoil, was a good example of how well the boundary layer height is calculated with our tools. The ONERA M6 wing and the Delery bump cases proved that these methods can be effectively applied to complex transonic flows where separation takes place.

We also demonstrated the ability to adapt the boundary layers in parallel for two internal turbulent flow applications. Superior flow feature resolution was displayed through the results and efficient performance of anisotropic meshes was underlined in comparison with an *a priori* generated mesh for the heat transfer manifold application. The use of these techniques for unsteady flow calculations was demonstrated in the case of 3D shallow cavity, where adaptation of the free shear layer led to capturing of more unsteady vortical scales with each successive adaptation.

In this work we showed that the boundary layer adaptivity techniques can be applied to real world complex turbulent flow problems like multi element wings which are usually very hard to mesh with satisfactory resolution in required areas. The adaptive boundary layer techniques showed that superior wake and tip region resolution can be easily achieved with the help of numerical error indicators. The strategy that we developed to combine Hessians and scalar error indicators created desired meshes which achieved better agreement with the experiments than the initial mesh. The results showed that our adaptive techniques are a promising way to mesh such complex geometries in an efficient manner.

7.2 Future Work

- Undoubtedly, the future of mesh adaptation lies with the parallel paradigm. As flow simulations are now done at extreme computational scales, mesh adaptation needs to keep up with these advantages to support these simulations. One bottleneck here is the scaling of mesh adaptation procedures. The strong scaling deteriorates very quickly as the number of elements per process is reduced. The flow solver can scale to as few as 5k - 15k elements per MPI process (depending on the architecture) [63]. However, the ideal range for mesh adaptation is somewhere 50k - 80k elements per process [53]. Hence the performance of the mesh adaptation procedures at this level is not satisfactory and this gap in the behavior of mesh adaptation and flow solver needs to be addressed. Efforts need to be undertaken to increase the scaling of the mesh adaptation procedures.

- Though thickness adaptation is a strong tool, its use in this thesis has been limited to relatively simpler geometries (even though complex flows). In the future, attempts need to be made to adapt complex geometries like the multi element wings with working thickness adaptation. This requires some effort on the mesh adaptation procedures' side to provide a more robust tool, as well as on our side to regulate our mesh size requests so that adaptation is easier. Currently, we use simple smoothing strategies to ensure that we pass smooth fields

to the mesh adaptation software, but for complex geometries with complex intersections of boundary layers, more development is needed.

- Current implementation of calculations for boundary layer height and detection of separated flow make use of the initial mesh boundary layer for its convenient data structures. However, situations might arise where initial boundary layer height is too low to capture these effects correctly and reaching the required height iteratively might take a few adaptation loops. To remedy this, growth edges can be extended by using virtual lines to enter the unstructured region of the mesh. These virtual lines can then be used to gather the required vorticity values and velocity profiles from inside the unstructured region, normal to the wall and would improve current calculations. This approach would be much more expensive than using the standard mesh boundary layer data structures, but would help immensely for poorly constructed initial meshes.

- In this work, we have mainly used RANS based models for turbulence modeling. The behavior of the strategies we have developed should ideally be universal, but needs to be studied in detail. For example, if the turbulence model used is $k - \epsilon$, the first cell height is not as low as for RANS wall resolved methods. Due to this, after adaptation, if first cell height is increased accordingly for the $k - \epsilon$ model, the separation detection algorithm may encounter problems due to lack of resolution near the wall. Issues like this for other turbulent models where the boundary layer is not integrated to the wall need to be identified and resolved.

- For superior resolution of the separated boundary layers, a feature to detach the mesh boundary layers from the wall to follow the free shear layers would be ideal. This requires some efforts on the mesh adaptation side.

Bibliography

[1] 1st high lift prediction workshop. http://hiliftpw.larc.nasa.gov/index-workshop1.html, June 2010.

[2] 2nd high lift prediction workshop. http://hiliftpw.larc.nasa.gov/index-workshop2.html, June 2013.

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

[4] P. S. Andersen, W. M. Kays, and R. J. Moffat. The turbulent boundary layer on a porous plate: An experimental study of the fluid mechanics for adverse free-stream pressure gradients. In Report No. HMT-15. Dept. Mech Eng., Stanford University CA, 1972.

[5] T. Apel and M. Dobrowolski. Anisotropic interpolation with applications to the finite element method. Computing, 47:277–293, 1992.

[6] E. F. D' Azvedo and R. B. Simpson. On optimal interpolation triangle incidences. SIAM Journal on Scientific and Statistical Computing, 12:755–786, 1991.

[7] I. Babuska and W. C. Rheinboldt. Error estimates for adaptive finite element computation. SIAM Journal on Numerical Analysis, 18:736–754, 1978.

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

[9] K. Bathe and H. Zhang. A mesh adaptivity procedure for CFD and fluid-structure interactions. Computers and Structures, 87:604–617, 2009.

[10] J. V. Becker. Boundary layer transition on the N.A.C.A. 0012 and 23012 airfoils in the 8-foot high-speed wind tunnel. In NACA-SR-137. National Advisory Committee for Aeronautics, January 1940.

[11] M. J. Berger and A. Jameson. Automatic adaptive grid refinement for euler equations. AIAA Journal, 23:561–568, 1985.

[12] M. J. Berger and J. Oliger. Adaptive mesh refinement for hyperbolic partial differential equations. Journal of Computational Physics, 53:484–512, 1984.

[13] H. Borouchaki, P. L. George, and B. Mohammadi. Delaunay mesh generation governed by metric specifications. part II. Applications. Finite Elements in Analysis and Design, 25:85–109, 1997.

[14] 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.

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

[16] A. J. Chen and Y. Kallinderis. Adaptive hybrid (prismatic-tetrahedral) grids for incompressible flows. International Journal for Numerical Methods in Fluids, 26:1085–1105, 1998.

[17] K.C. Chitale, O. Sahni, S. Tendulkar, R. Nastasia, K. E. Jansen, and M. S. Shephard. Boundary layer adaptation for transonic turbulent flows. In 21st AIAA Computational Fluid Dynamics Conference, San Diego, CA, June 2013. AIAA Paper 2013–2445.

[18] P. G. Ciarlet. The finite element method for elliptic problems. North-Holland Amsterdam, 1978.

[19] S. D. Connell and M. E. Braaten. Semi-structured mesh generation for three-dimensional Navier-Stokes calculations. AIAA Journal, 33:1017–1024, 1995.

[20] Cray. Cray XE6. `http://www.cray.com/Products/XE/CrayXE6System.aspx`.

[21] H. L. de Cougny and M. S. Shephard. Parallel mesh adaptation by local mesh modification. Technical Report SCOREC #21-1995, Scientific Computation Research Center, RPI, Troy, NY, 1995.

[22] J. M. Delery. Investigation of strong shock turbulent boundary layer interaction in 2-D transonic flows with emphasis on turbulence phenomena. In Proceedings of 14th Fluid and Plasma Dynamics conference. AIAA, 1981.

[23] M. Emory, R. Pecnik, and G. Laccarino. Modeling structural uncertainties in Reynolds-averaged computations of shock/boundary layer interactions. In Proceedings of 49th Aerospace Science Meeting. AIAA Paper 2011–0479, 2011.

[24] L. Formaggia and S. Perotto. New anisotropic a priori error estimates. Numerische Mathematik, 89:641–667, 2001.

[25] L. P. Franca and S. Frey. Stabilized finite element methods: II. The incompressible Navier-Stokes equations. Computer Methods in Applied Mechanics and Engineering, 99:209–233, 1992.

[26] P. J. Frey and F. Alauzet. Anisotropic mesh adaptation for CFD computations. Computer Methods in Applied Mechanics and Engineering, 194:5068–5082, 2005.

[27] R. V. Garimella and M. S. Shephard. Boundary layer mesh generation for viscous flow simulations. International Journal for Numerical Methods in Engineering, 49:193–218, 2000.

[28] M. Geveci, P. Oshkai, D. Rockwell, J.-C. Lin, and M. Pollack. Imaging of the self-excited oscillation of flow past a cavity during generation of a flow tone. Journal of Fluids and Structures, 18:665–694, 2003.

[29] N. Gregory and C. L. O'Reilly. Low-speed aerodynamic characteristics of NACA 0012 aerofoil section, including the effects of upper-surface roughness simulating hoar frost. In R&M 3726. National Physical Laboratory, January 1970.

[30] O. Hassan, K. Morgan, E. J. Probert, and J. Peraire. Unstructured tetrahedral mesh generation for three-dimensional viscous flows. International Journal for Numerical Methods in Engineering, 39:549–567, 1996.

[31] T. J. R. Hughes, L. P. Franca, and G. M. Hulbert. A new finite element formulation for fluid dynamics: VIII. the Galerkin/least-squares method for advective-diffusive equations. Computer Methods in Applied Mechanics and Engineering, 73:173–189, 1989.

[32] T. J. R. Hughes, L. Mazzei, and K. E. Jansen. Large-eddy simulation and the variational multiscale method. Computing and Visualization in Science, 3:47–59, 2000.

[33] IBM. IBM guide to using the Blue Gene/P. `http://www.redbooks.ibm.com/abstracts/sg247287.html`.

[34] Simmetrix Inc. `www.simmetrix.com`.

[35] Y. Ito and K. Nakahashi. Unstructured mesh generation for viscous flow computations. In Proceedings of the 11th International Meshing Roundtable. IMR, 2002.

[36] K. E. Jansen. Unstructured grid large eddy simulations of wall bounded flows. In Annual Research Briefs,, pages 151–156, NASA Ames/Stanford University, 1993. Center for Turbulence Research.

[37] K. E. Jansen. Unstructured grid large eddy simulations of flow over an airfoil. In Annual Research Briefs, pages 161–173, NASA Ames/Stanford University, 1994. Center for Turbulence Research.

[38] K. E. Jansen, Z. Johan, and T. R. Hughes. Implementation of one-equation turbulence model within a stabilized finite element formulation of a symmetric advective-diffusive system. Computer Methods in Applied Mechanics and Engineering, 105:405–433, 1993.

[39] K. E. Jansen, M. S. Shephard, and M. W. Beall. On anisotropic mesh generation and quality control in complex flow problems. In Proceedings of 10th International Meshing Roundtable. IMR, 2001.

[40] P. L. Johnson, K. M. Jones, and M. D. Madson. Experimental investigation of a simplified 3D high lift configuration in support of CFD validation. In 18th AIAA Applied Aerodynamics Conference, Denver, CO. AIAA Paper 2000–4217, August 2000.

[41] Ravindra K., N. V. Shende, S. K. Patel, and N. Balakrishnan. Unstructured adaptive mesh calculations for NASA TRAP WING using the code HiFUN. In 30th AIAA Applied Aerodynamics Conference, New Orleans, Louisiana, June 2012. AIAA Paper 2012–2849.

[42] A. Khawaja and Y. Kallinderis. Hybrid grid generation for turbomachinery and aerospace applications. International Journal for Numerical Methods in Engineering, 49:145–166, 2000.

[43] G. Kunert. Toward anisotropic mesh construction and error estimation in the finite element solver. Numerical Methods for Partial Differential Equations, 18:625–648, 2002.

[44] C. L. Ladson. Effects of independent variation of mach and Reynolds numbers on the low-speed aerodynamic characteristics of the NACA 0012 airfoil section. In NASA TM 4074. National Aeronautics and Space Administration, October 1988.

[45] C. L. Ladson, A. S. Hill, and W. G. Johnson. Pressure distribution from high reynolds number transonic tests of an NACA 0012 airfoil in the Langley 0.3-meter transonic cryogenic tunnel. In NASA TM 10052. National Aeronautics and Space Administration, December 1987.

[46] F. S. Len and G. Kalitzin. Computations of transonic flow with $v^2 - f$ turbulence model. International Journal of Heat and Fluid Flow, 22:53–61, 2001.

[47] X. Li, J. F. Remacle, N. Chevaugeon, and M. S. Shephard. Anisotropic mesh gradation control. In Proceedings of the 13th International Meshing Roundtable, pages 401–412, 2004.

[48] X. Li, M. S. Shephard, and M. W. Beall. 3D anisotropic mesh adaptation by mesh modifications. Computer Methods in Applied Mechanics and Engineering, 194:4915–4950, 2005.

[49] M. J. Marchant and N. P. Weatherill. Unstructured grid generation for viscous flow simulations. In Proceedings of 4th International Conference on Numerical Grid Generation in Computational Field Simulations and Related Fields., Pineridge, Swansea, 1994.

[50] M. J. Marcum. Generation of unstructured grids for viscous flow applications. In 33rd AIAA Aerospace Sciences Meeting, Reno, NV, January 1995. AIAA Paper 95–0212.

[51] F. R. Menter. Two-equation eddy-viscosity turbulence models for engineering applications. AIAA Journal, 32(8):1598–1605, August 1994.

[52] P. Moin and K. Mahesh. Direct numerical simulation: A tool in turbulence research. Annual Review of Fluid Mechanics, 30:539–578, 1998.

[53] A. Ovcharenko, K. Chitale, O. Sahni, K. E. Jansen, and M. S. Shephard. Parallel anisotropic mesh adaptation with boundary layers. In Proceedings of the 21st International Meshing Roundtable, 82, pages 437–455. IMR, 2012.

[54] C. C. Pain, A. P. Umpleby, C. R. E. de Oliveira, and A. J. H. Goddard. Tetrahedral mesh optimization and adaptivity for steady-state and transient finite element calculations. Computer Methods in Applied Mechanics and Engineering, 190:3771–3796, 2001.

[55] M. A. Park. Three-dimensional turbulent RANS adjoint-based error correction. AIAA Paper 2003–3849, 2003.

[56] M. A. Park, E. M. Lee-Rausch, and C. L. Rumsey. FUN3D and CFL3D computations for the first high lift prediction workshop. In 49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, Orlando, Florida, January 2011. AIAA Paper 2011–936.

[57] J. Peraire, M. Vahdati, K. Morgan, and O. C. Zienkiewics. Adaptive remeshing for compressible flow computations. Journal of Computational Physics, 72:449–466, 1987.

[58] S. Pirzadeh. Unstructured viscous grid generation by advancing-layers method. AIAA Journal, 32:1735–1737, 1994.

[59] R. Rudnik, K. Huber, and S. Melber-Wilkending. EUROLIFT test case description for the 2nd high lift prediction workshop. In AIAA Paper 2012–2924, June 2012.

[60] O. Sahni. Automated adaptive viscous flow simulations. PhD thesis, Rensselaer Polytechnic Institute, 2006.

[61] O. Sahni, K. E. Jansen, M. S. Shephard, C. A. Taylor, and M. W. Beall. Adaptive boundary layer meshing for viscous flow simulations. Engineering with Computers, 24(3):267–285, 2008.

[62] O. Sahni, J. Muller, K. E. Jansen, M. S. Shephard, and C. A. Taylor. Efficient anisotropic adaptive discretization of the cardiovascular system. Computer Methods in Applied Mechanics and Engineering, 195:5634–5655, 2006.

[63] O. Sahni, M. Zhou, M. S. Shephard, and K. E. Jansen. Scalable implicit finite element solver for massively parallel processing with demonstration to 160K cores. In Proceedings of the Conference on High Performance Computing, Networking, Storage and Analysis. Super Computing, 2009.

[64] H. Schlichting and K. Gersten. Boundary layer theory. McGraw Hill, 1965.

[65] V. Schmitt and F. Charpin. Pressure distribution on the ONERA-M6-Wing at transonic Mach numbers. In Report of the Fluid Dynamics Panel Working Group 04, volume 138. AGARD, May 1979.

[66] SCOREC. Scorec open source mesh adaptation software. `http://http://scorec.rpi.edu/software.php`.

[67] J. Smagorinsky. General circulation experiments with the primitive equations, I. The basic experiment. Monthly Weather Review, 91:99–152, 1963.

[68] T. Sonar. Strong and weak norm refinement indicators based on the finite element residual for compressible flow computations. Impact of Computing Science and Engineering, 5:111–127, 1993.

[69] P. R. Spalart. Young-person's guide to detached-eddy simulation grids. In NASA/CR-2001-211032. National Aeronautics and Space Administration, 2001.

[70] P. R. Spalart. Trends in turbulence treatments. AIAA Paper 2000–2306, June 2000.

[71] P. R. Spalart and S. R. Allmaras. A one-equation turbulence model for aerodynamic flows. In AIAA Paper 92–439, 1992.

[72] P. R. Spalart, S. Deck, M. L. Shur, K. D. Squires, M. K. Strelets, and A. Travin. A new version of detached-eddy simulation, resistant to ambiguous grid densities. Theoretical and Computational Fluid Dynamics, 20(3):181–195, 2006.

[73] P.R. Spalart, W.H. Jou, M. Stretlets, and S.R. Allmaras. Comments on the feasibility of LES for wings and on a hybrid RANS/LES approach. In Advances in DNS/LES, pages 137–147, Columbus, 1997. Greyden Press.

[74] D. B. Spalding. A single formula for the law of the wall. Journal of Applied Mechanics, 28:455–458, 1961.

[75] I. Thomas and T. Sonar. On a second order residual estimator for numerical schemes for nonlinear hyperbolic conservation laws. Journal of Computational Physics, 171(1):227–242, 2001.

[76] O. V. Vasilyev and T. S. Lund. A general theory of discrete filtering for LES in complex geometry. In Annual Research Briefs, pages 67–82, NASA Ames / Stanford University, 1997. Center for Turbulence Research.

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

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

[79] C. H. Whiting and K. E. Jansen. A stabilized finite element method for the incompressible Navier-Stokes equations using a hierarchical basis. International Journal of Numerical Methods in Fluids, 35:93–116, 2001.

[80] C. H. Whiting, K. E. Jansen, and S. Dey. Hierarchical basis in stabilized finite element methods for compressible flows. Computer Methods in Applied Mechanics and Engineering, 192:5167–5185, 2003.