LEAST-SQUARES FINITE ELEMENT SOLUTION OF EULER EQUATIONS WITH ADAPTIVE MESH REFINEMENT

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

HAYRİ YİĞİT AKARGÜN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
MECHANICAL ENGINEERING

FEBRUARY 2012

Approval of the thesis

LEAST-SQUARES FINITE ELEMENT SOLUTION OF EULER EQUATIONS WITH ADAPTIVE MESH REFINEMENT

Submitted by **HAYRİ YİĞİT AKARGÜN** in partial fulfillment of the requirements for the degree of **Master of Science in Mechanical Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan ÖZGEN ________________
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Suha ORAL ________________
Head of Department, **Mechanical Engineering**

Assist. Prof. Dr. Cüneyt SERT ________________
Supervisor, **Mechanical Engineering Dept., METU**

Examining Committee Members:

Prof. Dr. Mehmet Haluk AKSEL ________________
Mechanical Engineering Dept., METU

Assist. Prof. Dr. Cüneyt SERT ________________
Mechanical Engineering Dept., METU

Inst. Dr. Tahsin ÇETİNKAYA ________________
Mechanical Engineering Dept., METU

Assist. Prof. Dr. Mehmet Metin YAVUZ ________________
Mechanical Engineering Dept., METU

Ali AKGÜL, M.Sc. ________________
Chief Engineer, ROKETSAN

Date: ________________

I hereby declare that all the information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name :

Signature :

ABSTRACT

LEAST-SQUARES FINITE ELEMENT SOLUTION OF EULER EQUATIONS WITH ADAPTIVE MESH REFINEMENT

Akargün, Hayri Yiğit
M.Sc., Department of Mechanical Engineering
Supervisor : Assist. Prof. Dr. Cüneyt SERT

February 2012, 63 Pages

Least-squares finite element method (LSFEM) is employed to simulate 2-D and axisymmetric flows governed by the compressible Euler equations. Least-squares formulation brings many advantages over classical Galerkin finite element methods. For non-self-adjoint systems, LSFEM result in symmetric positive-definite matrices which can be solved efficiently by iterative methods. Additionally, with a unified formulation it can work in all flight regimes from subsonic to supersonic. Another advantage is that, the method does not require artificial viscosity since it is naturally diffusive which also appears as a difficulty for sharply resolving high gradients in the flow field such as shock waves. This problem is dealt by employing adaptive mesh refinement (AMR) on triangular meshes. LSFEM with AMR technique is numerically tested with various flow problems and good agreement with the available data in literature is seen.

Keywords: Least-Squares Finite Element Method, Computational Fluid Dynamics, Compressible Flow, Inviscid Flow, Euler Equations, Adaptive Mesh Refinement.

ÖZ

EN KÜÇÜK KARELER SONLU ELEMANLAR METODU İLE EULER DENKLEMLERİNİN UYARLANMIŞ AĞLARDA ÇÖZÜMÜ

Akargün, Hayri Yiğit
Yüksek Lisans, Makina Mühendisliği Bölümü
Tez Yöneticisi : Yrd. Doç. Dr. Cüneyt Sert

Şubat 2012, 63 Sayfa

Sıkıştırılabilir Euler denklemlerinin modellediği iki boyutlu düzlemsel ve eksenel simetrik akışların en küçük kareler sonlu elemanlar metodu kullanılarak benzetimi yapılmıştır. En küçük kareler formülasyonunun geleneksel Galerkin sonlu elemanlar metoduna göre birçok avantajı vardır. En küçük kareler formülasyonu kendine eş olmayan denklemler için simetrik kesin artı matris oluşturur ve böylece sonuç sistemi döngülü yöntemler ile verimli bir şekilde çözülebilir. Ayrıca, ses-altı hızlardan ses-üstü hızlara kadar bütün akış rejimleri tek bir formülasyon ile çözülebilmektedir. Bu metod kendiliğinden dağılımlı olduğu için yapay viskozite kullanımına gerek duyulmamaktadır. Ancak metodun bu özelliği aynı zamanda akıştaki keskin değişimlerin yüksek hassasiyet ile çözümlenmesini zorlaştırmaktadır. Bu sebepten dolayı, akış çözücüsüne üçgen ağlar için çözüm ağı uyarlama algoritması eklenmiştir. Geliştirilen çözücü farklı akış problemleri için sayısal olarak test edilmiştir. Elde edilen sonuçlar literatürde ulaşılan sonuçlar ile uyumludur.

Anahtar Kelimeler: En Küçük Kareler Sonlu Elemanlar Metodu, Hesaplamalı Akışkanlar Dinamiği, Sıkıştırılabilir Akışlar, Ağdasız Akışlar, Euler Denklemleri, Çözüm Ağı Uyarlaması

To My Family

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to my supervisor Dr. Cüneyt Sert for his endless contributions and guidance throughout my thesis. In every stage of this work, he has supported me with patience and been very friendly.

I also appreciate the suggestions, criticism and interpretations done on my work by the thesis examining committee members.

I would especially want to thank my wife, Müge, for her understanding and support throughout my study. I also want to express my gratitude to my parents for their support.

I would also like to thank my chief engineer Ali Akgül for his advices and criticism which have been very valuable to me. Additionally, I want to thank my colleagues in Aerodynamics Division of Roketsan for their help and encouragement.

The last but not the least, I want to thank my friends and colleagues Mehmet Akgül and Mustafa Uğur Göktolga for their support and friendship during this study.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURES

CHAPTER 1

INTRODUCTION

In the past few decades, the progress in Computational Fluid Dynamics (CFD) has been enormous. Due to increasing computational power and development of various robust solution techniques, nowadays, many complex fluid dynamics problems can be numerically simulated with acceptable accuracy. But there is always need for further development in order to be able to solve large 3-D problems with limited computational resources

In CFD, the most extensive equation of motion that describes a wide range of flow conditions is the Navier-Stokes (N-S) equations. Considering today's computational technology, direct numerical simulation of these equations is computationally expensive and not practical for many real life problems of engineering interest. For this reason, to simulate flow fields efficiently and accurately, some assumptions need to be done, which depend on the problem to be solved. For example, when the viscous forces are small compared to the inertial forces, such as high Reynolds number flows, inviscid flow assumption is usually used. For such flows, away from the boundary layer, compressible Euler equations can simulate the flow field in essentially the same manner as the Navier-Stokes equations. In the current study 2-D, compressible flow fields modeled by Euler equations are simulated.

Governing equations of flow problems can be solved numerically with different methods such as finite volume method, finite difference method finite element method, spectral method and boundary element method. In the current study finite element method is used and its basic principles are explained in the next section.

1.1 Finite Element Method (FEM)

FEM is a numerical technique used to solve partial differential, integral and variational equations. The first step in the FEM is discretizing the continuum into a finite number of simple-shaped elements. The most commonly used elements in 2-D are triangular and quadrilateral, shown in Figure 1-1. Polynomial order of shape functions that are used to describe the variation of unknowns over these elements is usually taken to be linear or quadratic in practice, but in theory it can be any desired order. In FEM, governing equations of the problem are written over each element of the solution domain one by one and individual elemental systems written in terms of elemental nodal unknowns are obtained by variational or weighted residual approaches. After obtaining elemental systems, they are assembled using element connectivity information to form a global system of linear algebraic equations. Finally, the global system is solved using an appropriate linear system solver.

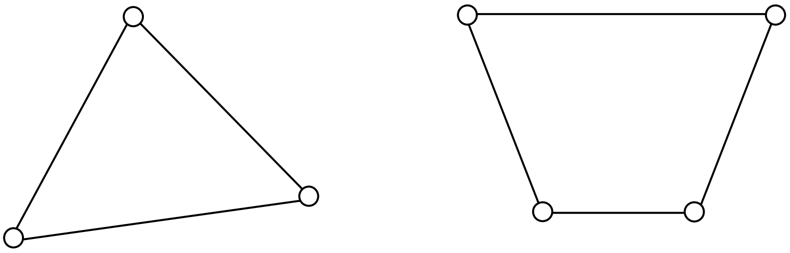

Figure 1-1: Linear triangular and quadrilateral elements

The first use of FEM, in engineering sense, goes back to 1941 when Hrennikoff developed "framework method" by dividing an elastic continuum structure into beam elements [1]. In 1960, the term "finite element method" is first used by Clough in an article [2]. Zienkiewicz published the first book about FEM in 1967 [3]. Since then, FEM has started to be used in a wide variety of engineering fields including fluid dynamics.

In FEM literature, many studies have been done for the solution of compressible Euler equations. Many different versions of FEM and mesh refinement techniques have been developed throughout the years. Hughes and Tezduyar presented Petrov-Galerkin finite element formulation for the solution of one and two dimensional compressible flows involving shocks [4]. Oden et al. used Taylor-Galerkin FEM for solving high speed compressible flows with an adaptive mesh algorithm [5]. In their study, artificial dissipation and flux-corrected transport scheme (FCT) is employed in order to avoid oscillations in the flow field. Loth, Baum and Löhner also used this method with FCT and artificial viscosity to solve high-speed axisymmetric nozzle flows [6].

A popular version of FEM employed by Beau and Tezduyar is based on Streamline-Upwind/Petrov-Galerkin (SUPG) formulation, which modifies the classical Galerkin formulation by adding numerical stability [7]. Based on this method, Tezduyar et al. introduced the deformable-spatial-domain/stabilized-space-time formulation which involves solution of moving boundary problems [8]. Peraire et al. modified SUPG formulation for solving compressible Euler equations at very low Mach numbers where standard SUPG algorithm fails to provide accurate results [9]. Peraire et al. also presented a higher order accurate FEM solution based on SUPG formulation for compressible Euler and N-S equations and compared the numerical solutions obtained by using linear and quadratic elements [10].

Bassi and Rebay used a higher-order discontinuous Galerkin FEM (DGFEM) to solve 2-D Euler equations [11]. This method combines the features of finite element and finite volume methods to develop a robust and accurate solver for flow problems involving shocks. Hartmann and Houston used the same method with an adaptive mesh refinement algorithm for compressible Euler equations and tested it with transonic nozzle flow and supersonic flow over airfoil problems [12]. Tu and

Aliabadi used DGFEM with a slope limiting procedure in order to remove oscillations at shock regions [13]. Authors claimed that added slope limiting procedure results in a robust DG solver for compressible Euler equations. Another DGFEM study presented by Feistauer and Kucera aims to solve compressible Euler equations in a wide range of Mach numbers [14]. Solution of compressible flows with DGFEM based methods is still an active research area [15-18].

1.2 Least-Squares Finite Element Method (LSFEM)

Another version of FEM, which is also used in the current study, is LSFEM. The basic principle of this method is to minimize the residuals of a first-order differential equation system in a least-squares sense [19]. As stated by Jiang the theoretical basis of the least-squares method is the bounded inverse theorem of linear operators. Based on this theorem, Jiang proved that LSFEM always leads to a convergent solution for a well-posed first order partial differential equation system without need to apply any special treatments such as upwinding or artificial dissipation.

There are significant advantages of least-squares finite element formulation compared to the classical Galerkin finite element formulation [19]. Maybe the most important one is the efficiency of the method. In LSFEM, non-self-adjoint equation systems, one of which is the governing equations of fluid dynamics, result in symmetric positive-definite global system matrices which can be solved efficiently by iterative linear solvers with relatively less memory requirement and CPU time compared to Galerkin FEM solutions [19]. This makes LSFEM especially more suitable for large-scale problems without need of high computational resources. Additionally, the theoretical basis of the least-squares minimization makes LSFEM a robust technique that does not need any special treatments and does not have problem dependent parameters to tune [20]. It can work in all flow regimes; subsonic, transonic, supersonic and hypersonic with a unified formulation. This advantage also extends to the field of structural mechanics, making LSFEM suitable for fluid structure interaction problems. Recent increase of interest in LSFEM also confirms these advantages. In the next section studies found in the literature to solve compressible Euler equations with LSFEM are summarized.

For the solution of the compressible Euler equations Fletcher used the least-squares method in primitive variables form [21]. Jiang and Carey used this method for the solution of incompressible flows with adaptive mesh refinement (AMR) in which least-squares functional is used as an error-indicator for AMR [22]. The resulting system is solved by the element-by-element conjugate gradient method. In another study, same authors extend their work for solving compressible Euler equations without mesh adaptation [23]. They used two different schemes; one minimizes the L_2 norm of the residual and the other minimizes the weighted H^1 norm of the residual with the backward-Euler time-discretization. In the H^1 norm based scheme, numerical dissipation is further increased at the discontinuities by introducing an additional term to the least-squares functional which is controlled by a user tuned parameter to overcome the oscillations near the shock regions. Both methods are tested with various problems and it is seen that L_2 scheme is a robust method but has relatively poor shock resolution, whereas H^1 method has better shock resolution but it needs user defined shock control parameter. Due to the numerical dissipation inherent to LSFEM, shock waves cannot be resolved sharply on coarse grids. To obtain high shock resolution, Habashi et al. used an r-type grid adaptation algorithm [24]. Supersonic and transonic shock reflection and flow over a bump problems are solved and flow features are shown to be captured accurately. Guaily and Megahed also used LSFEM with r-type grid adaptation for 2-D planar and axisymmetric compressible flows [25]. In addition to flow over a bump and shock-reflection problems, they also solved nozzle flow with shocks and axisymmetric jet flow. Moussaoui used a unified formulation based on LSFEM to calculate both very low Mach number flows and compressible flows [26]. Moussaoui's approach is basically to apply a new scaling technique for the flow variables which makes it numerically appropriate for very low Mach numbers. Reddy et al. presented least-squares finite element models for both high-speed inviscid and low-speed viscous compressible flows [27]. For the solution of the compressible Euler equations, Gerritsma et al. employed least-squares formulation with higher order spectral methods [28].

1.3 Adaptive Mesh Refinement (AMR)

In computational fluid dynamics, sometimes the initially created mesh can not accurately resolve the flow features such as variations and discontinuities of the variables especially for flows involving shock and expansion waves. In order to simulate the variations in a flow field accurately, the initial mesh can be adapted/modified until the desired accuracy is achieved. There are mainly three methods used for grid adaptation which are r-refinement, h-refinement and p-refinement. These methods can also be combined for hybrid adaptation purposes.

In r-refinement, mesh is modified by moving nodes without changing grid connectivity and number of nodes/elements. Grid movement algorithm is usually based on the spring analogy which basically considers each element edge as a spring with a spring constant being a function of a user selected error indicator. Figure 1-2 taken from Habeshi et. al [24] demonstrates r-refinement for supersonic flow over a bump problem. As seen from Figure 1-2, grids are moved to the shock regions to accurately capture discontinuities in the flow field. But this method is highly dependent on the initial mesh because the extend of refinement is limited to the available nodes in the domain.

In p-refinement, higher order polynomial interpolations are used on the elements to be refined without changing the number of elements. Even though the number of elements is not changed, resultant mesh has non-conformal elements with hanging nodes. An example to p-refinement of a 2-D quadrilateral element is given in Figure 1-3, in which the order of upper left element is increased from 1 to 2.

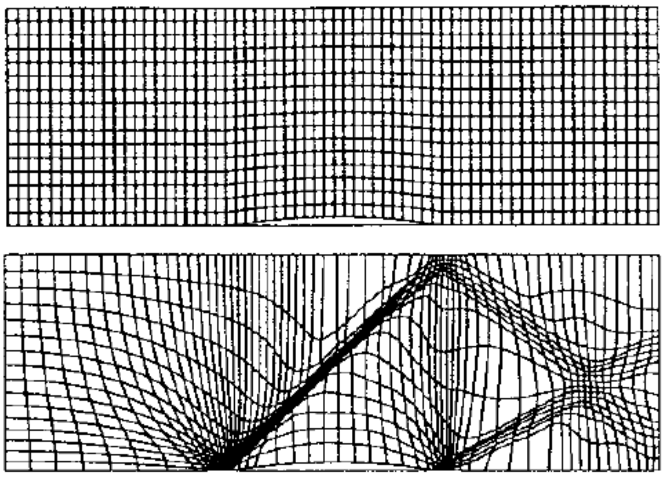

Figure 1-2: Initial and refined (using r-refinement) meshes for supersonic bump flow [24]

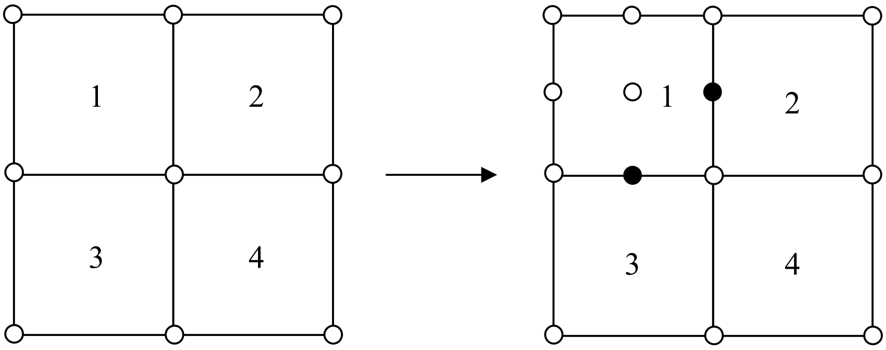

Figure 1-3: p-refinement of a quadrilateral element with hanging nodes shown in black

In h-refinement, elements are divided into smaller elements at desired regions to increase accuracy. Grid connectivity and number of nodes/elements are changed by this method, which brings complications to code development. Many different element division techniques have been developed so far. An example to h-refinement of a two-dimensional mesh of quadrilateral elements is given in Figure 1-4 , in which element 1 is divided into four elements and others remain the same, which results in non-conformal elements (element 2 and 3) with hanging nodes. It is also possible to refine quadrilateral elements without generating hanging nodes.

On the other hand it is easier to divide triangular elements without creating any hanging nodes as seen in Figure 1-5. The element division shown in Figure 1-5 is not a preferred method because dividing an element several times results in very skew elements in the region where high accuracy is needed. Another method introduced by Rivara considers the longest edge of the element to be refined [29]. In this algorithm, elements selected for refinement and their neighboring elements are bisected by their longest edges until the mesh is conforming.. This is an effective method to refine mesh without causing mesh degeneracy. But in this method mesh outside the desired refinement region is also refined to satisfy the longest edge rule. Suarez et al. pointed out this undesired mesh propagation problem [30], an extreme case of which is illustrated in Figure 1-6.

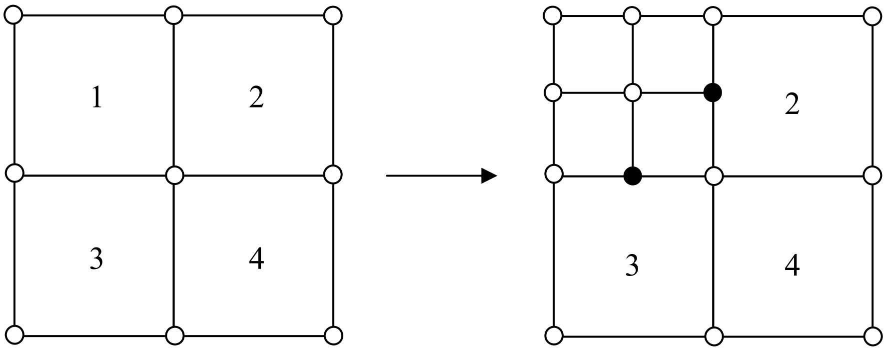

Figure 1-4: h-refinement of a quadrilateral element with hanging nodes shown in black

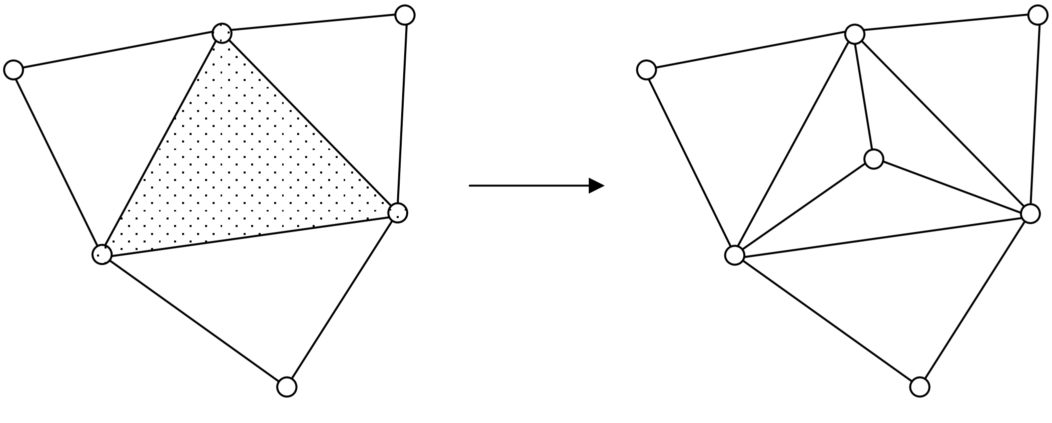

Figure 1-5: h-refinement of a triangular element without hanging nodes

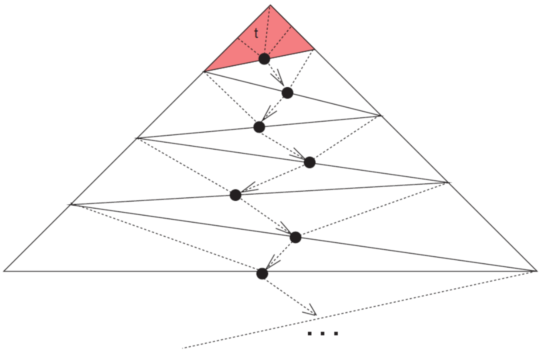

Figure 1-6: Demonstration of undesired mesh propagation problem of the longest edge refinement algorithm [30]

1.4 Current Study

In the current study, a solver based on LSFEM with an h-type AMR technique is developed to numerically simulate 2-D planar and axisymmetric flows governed by the compressible Euler equations. The developed solver works with both quadrilateral and triangular elements of linear and quadratic shape functions and it is tested with transonic and supersonic flow problems. Numerical results are in good agreement with the literature for both element types and orders. The developed code solves unsteady versions of the governing equations and marches in time to obtain a steady state solution, if one exists. In LSFEM, numerically introduced artificial viscosity can be decreased by using smaller time steps in order to sharply resolve shock waves. But using a small time step on a relatively coarse mesh causes oscillations near the shock region. In order to achieve sharp shock resolution without oscillations, AMR technique is employed for meshes with triangular elements.

It is also worth to mention that the current solver is developed starting from an initial code which was capable of solving 2-D planar, steady, incompressible Navier-Stokes equations in the $u - p - \omega$ formulation on linear quadrilateral mesh. The code is developed in C++ language.

CHAPTER 2

GOVERNING EQUATIONS & LEAST SQUARES FEM FORMULATION

In this chapter, first the governing equations for 2-D, compressible, inviscid flow will be recalled. Time-discretization of these equations will be done and the resultant equations will be linearized. Finally least-squares finite element formulation of the resultant equation set will be provided.

2.1 Governing Equations

Fundamental equations of fluid mechanics are the conservation of mass, momentum and energy. For an inviscid compressible flow, governing equations are the following

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \left(\rho \vec{V}\right) = 0 \tag{2.1}$$

$$\frac{\partial \vec{V}}{\partial t} + \vec{V} \cdot \nabla \vec{V} + \frac{1}{\rho} \nabla p = 0 \tag{2.2}$$

$$\frac{\partial p}{\partial t} + \vec{V} \cdot \nabla p + \gamma p \left(\nabla \cdot \vec{V}\right) = 0 \tag{2.3}$$

where ρ is the fluid density, $\vec{V}$ is the velocity vector, p is the pressure and the γ is the constant specific heat ratio. Conservation of energy is written in terms of pressure by using the perfect gas relation. In this thesis, compressible Euler equations are solved for two spatial dimensions, both planar and axisymmetric. Conservation equations for a planar flow in the xy Cartesian plane are given below

$$\frac{\partial \rho}{\partial t}+\rho\frac{\partial u}{\partial x}+u\frac{\partial \rho}{\partial x}+\rho\frac{\partial v}{\partial y}+v\frac{\partial \rho}{\partial y}=0 \tag{2.4}$$

$$\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+\frac{1}{\rho}\frac{\partial p}{\partial x}=0 \tag{2.5}$$

$$\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}+\frac{1}{\rho}\frac{\partial p}{\partial y}=0 \tag{2.6}$$

$$\frac{\partial p}{\partial t}+\gamma p\frac{\partial u}{\partial x}+u\frac{\partial p}{\partial x}+\gamma p\frac{\partial v}{\partial y}+v\frac{\partial p}{\partial y}=0 \tag{2.7}$$

where u and v are the velocity components in x and y directions.

For an axisymmetric flow, variables are represented in cylindrical coordinates; r, θ and z with the following assumptions; unknowns (ρ, v_z, v_r, p) are independent of the angular variable θ and the velocity in the θ direction is zero. Conservation equations for an axisymmetric flow in the rz plane are given below

$$\frac{\partial \rho}{\partial t}+\rho\frac{\partial v_z}{\partial z}+v_z\frac{\partial \rho}{\partial z}+\rho\frac{\partial v_r}{\partial r}+v_r\frac{\partial \rho}{\partial r}+\frac{\rho v_r}{r}=0 \tag{2.8}$$

$$\frac{\partial v_z}{\partial t}+v_z\frac{\partial v_z}{\partial z}+v_r\frac{\partial v_z}{\partial r}+\frac{1}{\rho}\frac{\partial p}{\partial z}=0 \tag{2.9}$$

$$\frac{\partial v_r}{\partial t}+v_z\frac{\partial v_r}{\partial z}+v_r\frac{\partial v_r}{\partial r}+\frac{1}{\rho}\frac{\partial p}{\partial r}=0 \tag{2.10}$$

$$\frac{\partial p}{\partial t}+\gamma p\frac{\partial v_z}{\partial z}+v_z\frac{\partial p}{\partial z}+\gamma p\frac{\partial v_r}{\partial r}+v_r\frac{\partial p}{\partial r}+\gamma\frac{p v_r}{r}=0 \tag{2.11}$$

where v_z and v_r are the velocity components in z and r directions, respectively.

Conservation equations for planar flows (2.4-2.7) can be written in matrix form in terms of primitive variables (in non-conservative form) as follows

$$\frac{\partial U}{\partial t}+A_1\frac{\partial U}{\partial x}+A_2\frac{\partial U}{\partial y}=0 \tag{2.12}$$

where $U^T = (\rho, u, v, p)$ is the unknown vector and the entries of the A_1 and A_2 matrices are given below

$$A_1 = \begin{bmatrix} u & \rho & 0 & 0 \\ 0 & u & 0 & 1/\rho \\ 0 & 0 & u & 0 \\ 0 & \gamma p & 0 & u \end{bmatrix} \qquad A_2 = \begin{bmatrix} v & 0 & \rho & 0 \\ 0 & v & 0 & 0 \\ 0 & 0 & v & 1/\rho \\ 0 & 0 & \gamma p & v \end{bmatrix} \tag{2.13}$$

Time discretization of the equation set (2.12) is done by the backward Euler method, during which a new unknown vector is introduced as $\Delta U = U^{n+1} - U^n$, where the superscripts denote discrete time levels. This relation can also be expressed as follows

$$\underset{\substack{\nearrow \\ \text{new time-step} \\ \text{values}}}{\begin{bmatrix} \rho \\ u \\ v \\ p \end{bmatrix}^{n+1}} = \underset{\substack{\uparrow \\ \text{previous time-step} \\ \text{values (known)}}}{\begin{bmatrix} \rho \\ u \\ v \\ p \end{bmatrix}^{n}} + \underset{\substack{\nwarrow \\ \text{new introduced} \\ \text{unknown vector}}}{\begin{bmatrix} \Delta\rho \\ \Delta u \\ \Delta v \\ \Delta p \end{bmatrix}} \tag{2.14}$$

Continuity equation (2.4), written for the current time level $n + 1$ can be expressed in terms of the new Δ unknowns and known values from the previous time level n as follows

$$\begin{aligned} &\frac{\Delta\rho}{\Delta t} + (\Delta\rho + \rho^n)\frac{\partial(\Delta u + u^n)}{\partial x} + (\Delta u + u^n)\frac{\partial(\Delta\rho + \rho^n)}{\partial x} \\ &+ (\Delta\rho + \rho^n)\frac{\partial(\Delta v + v^n)}{\partial y} + (\Delta v + v^n)\frac{\partial(\Delta\rho + \rho^n)}{\partial y} = 0 \end{aligned} \tag{2.15}$$

Neglecting the following higher order terms in the above equation

$$\Delta u\frac{\partial\Delta\rho}{\partial x}, \Delta\rho\frac{\partial\Delta u}{\partial x}, \Delta v\frac{\partial\Delta\rho}{\partial y}, \Delta\rho\frac{\partial\Delta v}{\partial y}$$

and re-arranging it such that known values are on the right-hand side, following continuity equation is obtained

$$\begin{aligned}\frac{\Delta\rho}{\Delta t} + \Delta\rho\frac{\partial u^n}{\partial x} + \rho^n\frac{\partial\Delta u}{\partial x} + \Delta u\frac{\partial\rho^n}{\partial x} + u^n\frac{\partial\Delta\rho}{\partial x} + \Delta\rho\frac{\partial v^n}{\partial y} + \rho^n\frac{\partial\Delta v}{\partial y} \\ + \Delta v\frac{\partial\rho^n}{\partial y} + v^n\frac{\partial\Delta\rho}{\partial y} = -\left(\rho^n\frac{\partial u^n}{\partial x} + u^n\frac{\partial\rho^n}{\partial x} + \rho^n\frac{\partial v^n}{\partial y} + v^n\frac{\partial\rho^n}{\partial y}\right)\end{aligned} \tag{2.16}$$

Similar to the continuity equation, momentum equation in the x-direction (2.5) written at time level $n+1$ can be expressed in terms of the new Δ unknowns as follows

$$\begin{aligned}\frac{\Delta u}{\Delta t} + (\Delta u + u^n)\frac{\partial(\Delta u + u^n)}{\partial x} + (\Delta v + v^n)\frac{\partial(\Delta u + u^n)}{\partial y} \\ + \frac{1}{\Delta\rho + \rho^n}\frac{\partial(\Delta p + p^n)}{\partial x} = 0\end{aligned} \tag{2.17}$$

Pressure term can be simplified by multiplying its numerator and denominator by $(\rho^n - \Delta\rho)$. Neglecting higher order terms and re-arranging equation (2.17), x-momentum equation becomes

$$\begin{aligned}\frac{\Delta u}{\Delta t} + \Delta u\frac{\partial u^n}{\partial x} + u^n\frac{\partial\Delta u}{\partial x} + \Delta v\frac{\partial u^n}{\partial y} + v^n\frac{\partial\Delta u}{\partial y} - \Delta\rho\frac{1}{(\rho^n)^2}\frac{\partial p^n}{\partial x} \\ + \frac{1}{\rho^n}\frac{\partial\Delta p}{\partial x} = -\left(u^n\frac{\partial u^n}{\partial x} + v^n\frac{\partial u^n}{\partial y} + \frac{1}{\rho^n}\frac{\partial p^n}{\partial x}\right)\end{aligned} \tag{2.18}$$

Similarly, y-momentum equation can be obtained as

$$\begin{aligned}\frac{\Delta v}{\Delta t} + \Delta u\frac{\partial v^n}{\partial x} + u^n\frac{\partial\Delta v}{\partial x} + \Delta v\frac{\partial v^n}{\partial y} + v^n\frac{\partial\Delta v}{\partial y} - \Delta\rho\frac{1}{(\rho^n)^2}\frac{\partial p^n}{\partial y} + \frac{1}{\rho^n}\frac{\partial\Delta p}{\partial y} \\ = -\left(u^n\frac{\partial v^n}{\partial x} + v^n\frac{\partial v^n}{\partial y} + \frac{1}{\rho^n}\frac{\partial p^n}{\partial y}\right)\end{aligned} \tag{2.19}$$

Following a similar procedure energy equation (2.7) in terms of the new Δ unknowns can be written as

$$\begin{aligned}\frac{\Delta p}{\Delta x}+\gamma\Delta p\frac{\partial u^n}{\partial x}+\gamma p^n\frac{\partial \Delta u}{\partial x}+\Delta u\frac{\partial p^n}{\partial x}+u^n\frac{\partial \Delta p}{\partial x}+\gamma\Delta p\frac{\partial v^n}{\partial y}+\gamma p^n\frac{\partial \Delta v}{\partial y}\\ +\Delta v\frac{\partial p^n}{\partial y}+v^n\frac{\partial \Delta p}{\partial y}=-\left(\gamma p^n\frac{\partial u^n}{\partial x}+u^n\frac{\partial p^n}{\partial x}+\gamma p^n\frac{\partial v^n}{\partial y}+v^n\frac{\partial p^n}{\partial y}\right)\end{aligned}\tag{2.20}$$

To summarize, equation set (2.4-2.7) is now discretized in time and linearized, yielding the following set of equations, for which the unknown vector is $\Delta U^T=(\Delta\rho,\Delta u,\Delta v,\Delta p)$.

$$\begin{aligned}\frac{\Delta \rho}{\Delta t}+\Delta\rho\frac{\partial u^n}{\partial x}+\rho^n\frac{\partial \Delta u}{\partial x}+\Delta u\frac{\partial \rho^n}{\partial x}+u^n\frac{\partial \Delta \rho}{\partial x}+\Delta\rho\frac{\partial v^n}{\partial y}+\rho^n\frac{\partial \Delta v}{\partial y}\\ +\Delta v\frac{\partial \rho^n}{\partial y}+v^n\frac{\partial \Delta \rho}{\partial y}=-\left(\rho^n\frac{\partial u^n}{\partial x}+u^n\frac{\partial \rho^n}{\partial x}+\rho^n\frac{\partial v^n}{\partial y}+v^n\frac{\partial \rho^n}{\partial y}\right)\end{aligned}\tag{2.21}$$

$$\begin{aligned}\frac{\Delta u}{\Delta t}+\Delta u\frac{\partial u^n}{\partial x}+u^n\frac{\partial \Delta u}{\partial x}+\Delta v\frac{\partial u^n}{\partial y}+v^n\frac{\partial \Delta u}{\partial y}-\Delta\rho\frac{1}{(\rho^n)^2}\frac{\partial p^n}{\partial x}+\frac{1}{\rho^n}\frac{\partial \Delta p}{\partial x}\\ =-\left(u^n\frac{\partial u^n}{\partial x}+v^n\frac{\partial u^n}{\partial y}+\frac{1}{\rho^n}\frac{\partial p^n}{\partial x}\right)\end{aligned}\tag{2.22}$$

$$\begin{aligned}\frac{\Delta v}{\Delta t}+\Delta u\frac{\partial v^n}{\partial x}+u^n\frac{\partial \Delta v}{\partial x}+\Delta v\frac{\partial v^n}{\partial y}+v^n\frac{\partial \Delta v}{\partial y}-\Delta\rho\frac{1}{(\rho^n)^2}\frac{\partial p^n}{\partial y}+\frac{1}{\rho^n}\frac{\partial \Delta p}{\partial y}\\ =-\left(u^n\frac{\partial v^n}{\partial x}+v^n\frac{\partial v^n}{\partial y}+\frac{1}{\rho^n}\frac{\partial p^n}{\partial y}\right)\end{aligned}\tag{2.23}$$

$$\begin{aligned}\frac{\Delta p}{\Delta x}+\gamma\Delta p\frac{\partial u^n}{\partial x}+\gamma p^n\frac{\partial \Delta u}{\partial x}+\Delta u\frac{\partial p^n}{\partial x}+u^n\frac{\partial \Delta p}{\partial x}+\gamma\Delta p\frac{\partial v^n}{\partial y}+\gamma p^n\frac{\partial \Delta v}{\partial y}\\ +\Delta v\frac{\partial p^n}{\partial y}+v^n\frac{\partial \Delta p}{\partial y}=-\left(\gamma p^n\frac{\partial u^n}{\partial x}+u^n\frac{\partial p^n}{\partial x}+\gamma p^n\frac{\partial v^n}{\partial y}+v^n\frac{\partial p^n}{\partial y}\right)\end{aligned}\tag{2.24}$$

Using the matrix notation previously used in equation (2.12), new set of equations can be written as

$$A\Delta U={A_1}^n\frac{\partial \Delta U}{\partial x}+{A_2}^n\frac{\partial \Delta U}{\partial y}+{A_0}^n\Delta U=f\tag{2.25}$$

where A_1 and A_2 matrices are the same as given before in equation (2.13) and the newly introduced A_0 matrix and f vector are given as follows (from now on, the superscript "n" which denotes the previous time-step in the unknowns will be omitted for clarity)

$$A_0 = \begin{bmatrix} \frac{1}{\Delta t} + \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} & \frac{\partial \rho}{\partial x} & \frac{\partial \rho}{\partial y} & 0 \\ -\frac{1}{\rho^2}\frac{\partial p}{\partial x} & \frac{1}{\Delta t} + \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} & 0 \\ -\frac{1}{\rho^2}\frac{\partial p}{\partial y} & \frac{\partial v}{\partial y} & \frac{1}{\Delta t} + \frac{\partial v}{\partial y} & 0 \\ 0 & \frac{\partial p}{\partial x} & \frac{\partial p}{\partial y} & \frac{1}{\Delta t} + \gamma\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) \end{bmatrix} \tag{2.26}$$

$$f = -\begin{bmatrix} \rho\frac{\partial u}{\partial x} + u\frac{\partial \rho}{\partial x} + \rho\frac{\partial v}{\partial y} + v\frac{\partial \rho}{\partial x} \\ u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + \frac{1}{\rho}\frac{\partial p}{\partial x} \\ u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + \frac{1}{\rho}\frac{\partial p}{\partial y} \\ \gamma p\frac{\partial u}{\partial x} + u\frac{\partial p}{\partial x} + \gamma p\frac{\partial v}{\partial y} + v\frac{\partial p}{\partial y} \end{bmatrix} \tag{2.27}$$

Following the same procedure with the planar case, for axisymmetric problems it can be shown that only the A_0 matrix and the f vector are modified as follows

$$A_{0_{axi}} = A_0 + \begin{bmatrix} \frac{v_r}{r} & 0 & \frac{\rho}{r} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{\gamma p}{r} & \frac{\gamma v_r}{r} \end{bmatrix} \quad , \quad f_{axi} = f - \begin{bmatrix} \frac{\rho v_r}{r} \\ 0 \\ 0 \\ \frac{\gamma p v_r}{r} \end{bmatrix} \tag{2.28}$$

and terms u, v, x and y in A_1, A_2, A_0 and f are expressed as v_z, v_r, z and r, respectively.

Calculations in the code are performed with the non-dimensional variables. Non-dimensionalization of these variables are done as follows

$$\rho^* = \frac{\rho}{\rho_\infty} \qquad u^* = \frac{u}{a_\infty} \qquad v^* = \frac{v}{a_\infty} \qquad p^* = \frac{p}{\rho_\infty a_\infty^2}$$

where the subscript "∞" denotes the free-stream values and "a" is the speed of sound.

2.2 Least-Squares Finite Element Formulation

Residual of equation set (2.21-2.24) is defined as follows

$$R = A\Delta U - f \tag{2.29}$$

LSFEM is based on the minimization of this residual function over the problem domain, resulting in the minimization of the following functional [19]

$$I(\Delta U) = \frac{1}{2}\int_\Omega R^T R d\Omega \tag{2.30}$$

In FEM, the domain is divided into finite number of non-overlapping subdomains, which are called elements and the variation of the unknowns over these elements are approximated by the use of nodal values of the unknowns and polynomial shape functions as given below [19]

$$\Delta U \approx \sum_{j=1}^{NEN} S_j \Delta U_j \tag{2.31}$$

where NEN is the number of elemental nodes, ΔU_j are the nodal unknown values that need to be calculated and S_j are known shape functions of space coordinates. More details about the shape functions used in this study will be given in the next section. Substituting the above approximate representation of the unknowns into equation (2.30) written over a single element, and minimizing the functional in a least squares sense it is possible to get the following elemental linear algebraic equation set

$$K^e \Delta U^e = F^e \tag{2.32}$$

where ΔU^e is the unknown variable vector, K^e and F^e are the elemental stiffness matrix and force vector of an arbitrary element e given as follows

$$K_{ij}^e = \int_{\Omega^e} (AS_i)^T (AS_j) d\Omega \tag{2.33}$$

$$F_i^e = \int_{\Omega^e} (AS_i)^T f d\Omega \tag{2.34}$$

where the differential operator A is

$$A = A_1 \frac{\partial}{\partial x} + A_2 \frac{\partial}{\partial y} + A_0 \tag{2.35}$$

In the developed code, elemental stiffness matrix K^e and force vector F^e are numerically calculated using Gauss-Legendre quadrature [31]. After the calculation of all elemental systems, they are assembled into a global set of equations. As an important property of LSFEM, global stiffness matrix turns out to be symmetric positive-definite, which is solved using an efficient element-by-element Jacobi preconditioned Conjugate Gradient method [31].

Although the formulation presented above involves unsteady terms, in this study only steady problems are solved through a time marching procedure. Upon convergence of a solution to steady-state unknown vector ΔU is expected to approach zero. Based on this, a convergence criterion is defined as the node number averaged L_2 norm of the unknowns as shown below, which is calculated for $(\Delta\rho, \Delta u, \Delta v, \Delta p)$ separately

$$\sqrt{\sum_{n=1}^{\mathrm{NN}} (\Delta U_i^n)^2 \Big/ \mathrm{NN}} \qquad i = 1{,}4 \tag{2.36}$$

where NN is the total number of nodes in the mesh.

2.3 Lagrange Shape Functions

In this thesis, 2-D triangular and quadrilateral elements in linear and quadratic forms are used to approximate the solution. Linear (4-node) and quadratic (9-node) quadrilateral master elements are shown in Figure 2-1.

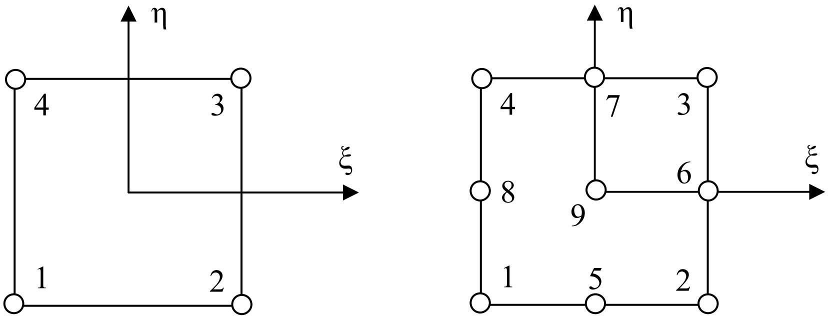

Figure 2-1: 4-node linear and 9-node quadratic quadrilateral master elements

Shape functions of the linear quadrilateral element are

$$\begin{aligned} S_1 &= \frac{1}{4}(1-\xi)(1-\eta) & S_2 &= \frac{1}{4}(1+\xi)(1-\eta) \\ S_3 &= \frac{1}{4}(1+\xi)(1+\eta) & S_4 &= \frac{1}{4}(1-\xi)(1+\eta) \end{aligned} \tag{2.37}$$

and the ones for the 9-node quadrilateral are

$$
\begin{aligned}
S_1 &= \frac{1}{4}(\xi^2 - \xi)(\eta^2 - \eta) & S_2 &= \frac{1}{4}(\xi^2 + \xi)(\eta^2 - \eta) \\
S_3 &= \frac{1}{4}(\xi^2 + \xi)(\eta^2 + \eta) & S_4 &= \frac{1}{4}(\xi^2 - \xi)(\eta^2 + \eta) \\
S_5 &= \frac{1}{2}(1 - \xi^2)(\eta^2 - \eta) & S_6 &= \frac{1}{2}(\xi^2 + \xi)(1 - \eta^2) \\
S_7 &= \frac{1}{2}(1 - \xi^2)(\eta^2 + \eta) & S_8 &= \frac{1}{2}(\xi^2 - \xi)(1 - \eta^2) \\
& & S_9 &= (1 - \xi^2)(1 - \eta^2)
\end{aligned}
\tag{2.38}
$$

Linear (3-node) and quadratic (6-node) triangular master elements are shown in Figure 2-2.

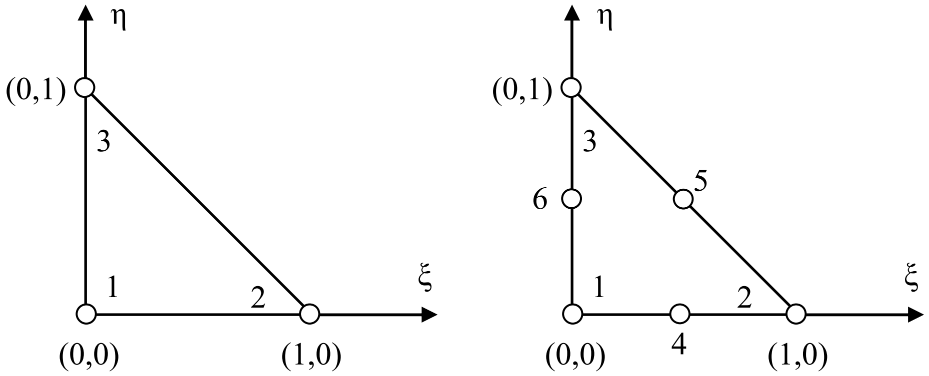

Figure 2-2: 3-node linear and 6-node quadratic triangular master elements

Shape functions for the 3-node triangular elements are

$$S_1 = 1 - \xi - \eta \qquad S_2 = \xi \qquad S_3 = \eta \tag{2.39}$$

and the ones for the 6-node triangular element are given as follows

$$
\begin{aligned}
S_1 &= 2(1-\xi-\eta)\left(\frac{1}{2}-\xi-\eta\right) & S_2 &= 2\xi\left(\xi-\frac{1}{2}\right) \\
S_3 &= 2\eta\left(\eta-\frac{1}{2}\right) & S_4 &= 4(1-\xi-\eta)\xi \\
S_5 &= 4\xi\eta & S_6 &= 4(1-\xi-\eta)\eta
\end{aligned}
\tag{2.40}
$$

When the shape functions are defined over master elements as given above, integrals of the elemental stiffness matrices and force vectors require a change of variables from (x, y) to (ξ, η), for which the standard Jacobian transformation is used [31].

2.4 Boundary Conditions

In order to have a well-posed problem, boundary conditions, number of which depends on the problem physics, have to be specified. For the equation system solved in this study, boundary conditions are easily imposed as Dirichlet type except for the wall boundary conditions. For inviscid flows, flow at the wall boundaries is tangent to the wall. Flow tangency boundary condition is usually imposed by using the penalty method or the coordinate rotation method. Both methods are implemented in the developed code, and details about them are presented below.

2.4.1 Penalty Method

In the penalty method, flow-tangency boundary condition is imposed by adding the following functional to equation (2.30)

$$
\frac{\alpha}{2}\int_{\Gamma_{wall}} \left(\vec{V}\cdot\vec{n}\right)^2 d\Gamma = \frac{\alpha}{2}\int_{\Gamma_{wall}} u_n^2 \, d\Gamma \tag{2.41}
$$

where $\vec{V}$ is the velocity vector , $\vec{n}$ is the unit vector normal to the wall boundary and α is the penalty parameter. Velocity components at the wall boundary are shown in

Figure 2-3. Penalty parameter determines how strongly the flow tangency boundary condition is imposed. Selecting a too large penalty parameter results in slower convergence, whereas a too small penalty parameter value decreases the accuracy at the wall boundaries. In this study, penalty parameter is selected in the range 0.05 and 0.2.

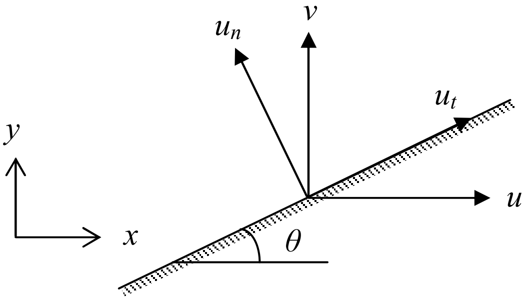

Figure 2-3: Velocity components at a solid wall

The angle θ at Figure 2-3 is calculated at the nodes and more than one element may share the same wall boundary node as seen in Figure 2-4

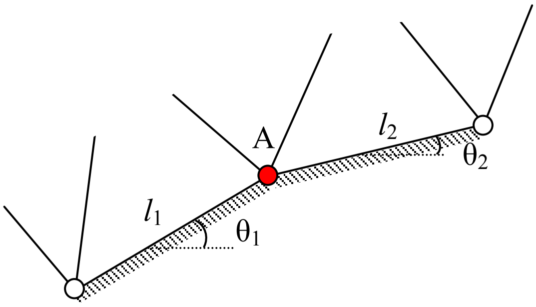

Figure 2-4: Wall boundary node A

For such cases the angle at the wall boundary node A can be calculated as follows, by using the angles and lengths of the adjacent edges [24]

$$\theta_A = \frac{l_1\theta_2 + l_2\theta_1}{l_1 + l_2} \tag{2.42}$$

Knowing these, velocity normal to the wall boundary can be written as

$$u_n = -\sin(\theta)\,u + \cos(\theta)\,v \tag{2.43}$$

where θ is the angle between wall boundary and the x-coordinate. Considering the velocity components in equation (2.43) to be written for time level $n+1$ and expanding them in terms of velocities at the previous time level n and the velocity differences between two time levels yields

$$u_n = -\sin(\theta)\,\Delta u + \cos(\theta)\Delta v - \sin(\theta)\,u^n + \cos(\theta)v^n \tag{2.44}$$

The last two terms in the above equation are known values from the previous time step. Introducing equation (2.44) into the functional (2.41), elemental stiffness matrix K^e and force vector F^e are modified at the wall boundary nodes by adding following matrix and vector respectively

$$K^e{}_{penalty} = K^e + \alpha \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \sin^2(\theta) & -\sin(\theta)\cos(\theta) & 0 \\ 0 & -\sin(\theta)\cos(\theta) & \cos^2(\theta) & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{2.45}$$

$$F^e{}_{penalty} = F^e + \alpha \begin{bmatrix} 0 \\ (\sin(\theta)\,u - \cos(\theta)\,v)\sin(\theta) \\ (\sin(\theta)\,u - \cos(\theta)\,v)\cos(\theta) \\ 0 \end{bmatrix} \tag{2.46}$$

Note that elemental stiffness matrix and force vector are modified at the wall boundary elements' wall boundary nodes only. It is also worth to mention that this modification do not destroy the important symmetric positive definite property of the discretized equation system.

2.4.2 Coordinate Rotation Method

Coordinate rotation method is a direct approach to impose flow tangency at wall boundaries. In this method, wall boundary nodes are rotated to a local coordinate system in which the velocity components are tangent and normal to the wall as seen in Figure 2-3. After this rotation, boundary condition can be implemented as a Dirichlet boundary condition by setting the normal velocity component to zero. The relation between tangent and normal velocity components and the original velocity components is given below

$$\begin{bmatrix} u_t \\ u_n \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \tag{2.47}$$

Considering that this coordinate transformation only affects the velocity components but not density or pressure, the following transformation matrix between the unknowns of the original and rotated local coordinates can be written

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & \sin(\theta) & 0 \\ 0 & -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.48}$$

In order to set the normal velocity components at the wall boundaries to zero elemental system for the wall boundary elements are modified using the transformation matrix as follows

$$(T^T K^e T)\Delta U = (T^T F^e) \tag{2.49}$$

Similar to the penalty approach, coordinate rotation also preserves the symmetric positive-definite property of the final equation set.

2.5 Adaptive Mesh Refinement (AMR)

In this study AMR is mainly used for efficient and accurate shock capturing and it is used only with triangular meshes. For this purpose first an error indicator is calculated for each element and the elements with an error exceeding the user defined tolerance value are divided into smaller ones. Details of these two steps are given below.

2.5.1 Error Indicator

The first step of grid adaptation is assigning error values for each element to determine the elements that need to be refined. These errors can be estimated by many different approaches. In this thesis the gradient approach, which is very suitable for supersonic inviscid flows, is used [38]. It uses the following area weighted L_2 norm of the pressure gradient as an error indicator

$$err = A_{cell}|\nabla p| \tag{2.50}$$

where A_{cell} is the area of the element. In determining the element division, error indicators are scaled with the average error indicator over the problem domain, which makes the algorithm more robust and problem independent. Note that although pressure gradient is used in error calculation in this study, one can also adopt other variables for this purpose.

2.5.2 Element Division

After determining the elements that need to be refined, element division is done by paying attention to preserving the conformity of the mesh. Element division algorithm used in this study is based on the bisection of the element by its longest edge [29]. In this algorithm, elements which are flagged for division are divided by their longest edges and then the non-conforming neighboring elements are bisected across their non-conforming edges to get rid of the undesired hanging nodes. Note that after the first division non-conforming neighbors are not recursively divided by their longest edges until the mesh is conforming as in Rivara's method [29].

A simple illustration of element division is shown in Figure 2-7. If the top element is flagged to be refined, first it is divided by its longest edge and then the neighbor element is divided to make the mesh conformal. As seen in the figure the neighboring element is not necessarily divided through its longest edge.

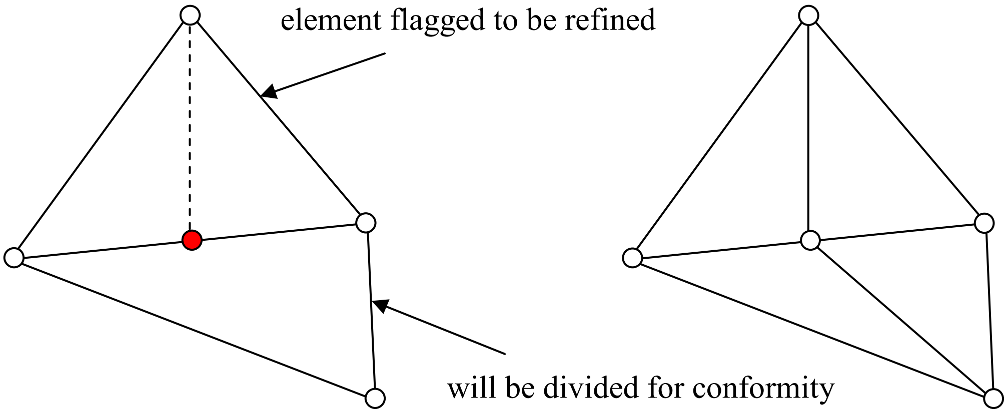

Figure 2-5: Element division

After the element division process is done, mesh connectivity is updated and new coordinates, flow-field variables and boundary condition information are assigned to the newly created nodes. This algorithm is not the best alternative if we strictly consider avoiding grid degeneracy, but it is simple and FEM is known to tolerate grid skewness to a certain degree. In addition, possible propagation problem demonstrated in Figure 1-6, which may occur in the recursively longest-edge refinement algorithms is avoided.

CHAPTER 3

RESULTS

In this chapter, results obtained with the developed code for various problems are given. First, supersonic and transonic flow over a circular bump is simulated and the results are compared with the ones found in literature. For this first problem both linear and quadratic types of quadrilateral and triangular elements are used. For the triangular elements, transonic and supersonic cases are also solved with AMR. The second test problem is chosen to be the supersonic flow over a wedge, which has a known analytical solution. In this problem AMR algorithm is challenged by starting the solution from a very coarse initial mesh. Wedge problem is also used to study the effect of time step on numerical dissipation and shock capturing. Then flow over an axisymmetric body is simulated and results are compared with available data in literature. The final test case is the supersonic flow in a scramjet inlet.

3.1 Supersonic Flow Over a Circular Bump

In this problem supersonic flow inside a channel with 4 % blockage by a circular bump as seen in the Figure 3-1 is considered. Channel length and height is 3.0 and 1.0 units respectively. Circular bump is positioned at the lower wall with a height of 0.04 units. In this classical benchmark problem it is known that as the supersonic flow with the inlet Mach number of 1.65 interacts with the bump an oblique shock emanates from the leading edge of the bump and reflects from the upper wall, which further interacts with the oblique shock that originates from the trailing edge of the bump.

The inlet boundary condition of the problem is given in the Figure 3-1. The specific heat ratio of the flowing gas is taken as 1.4. The exit boundary is left free (nothing special is done for it in the code) and on the upper and lower walls, flow tangency boundary condition is imposed by the penalty method.

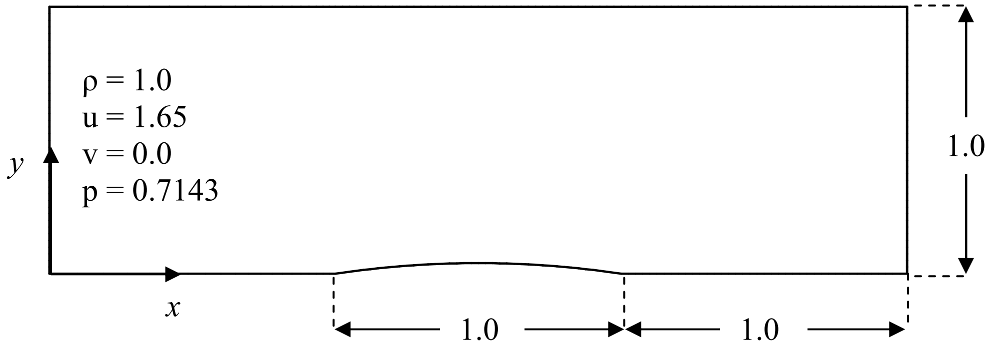

Figure 3-1: Sketch of the supersonic flow over a circular bump problem

For this first test case calculations are done for both quadrilateral and triangular meshes with linear and quadratic interpolating functions and results will be compared with the data found in the literature [33]. Quadrilateral and triangular meshes should be similar for a proper comparison of the solutions obtained with them. For this purpose, first the mesh of quadrilateral elements shown in Figure 3-2 is created and the triangular mesh, also shown in Figure 3-2 is obtained by dividing each quadrilateral element into two, ending up with the same number of nodes and unknowns for each mesh. For both meshes, on the bottom wall there are 30 uniformly distributed elements on the bump and after the bump and there are 20 elements before the bump clustered to the bump. On the inlet and exit there are 30 uniformly distributed elements.

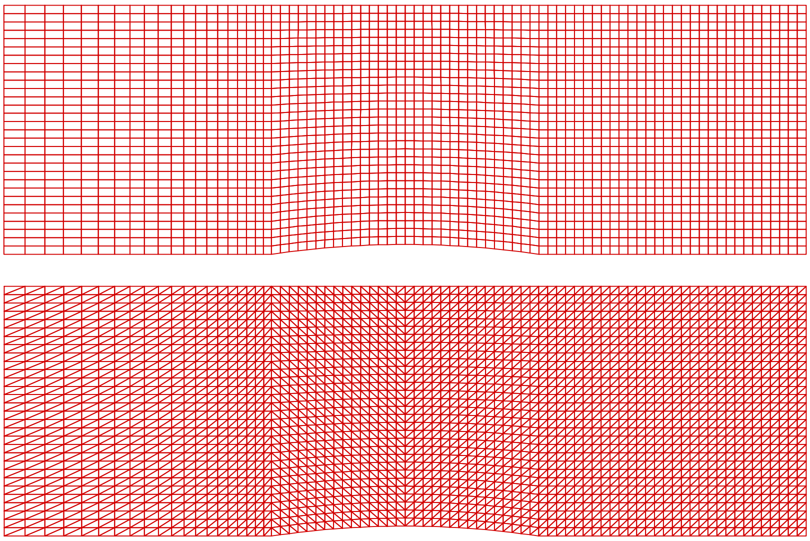

Figure 3-2: Quadrilateral and triangular computational meshes used for supersonic bump problem (2511 nodes)

As clearly stated by Jiang [19] selection of the number of Gauss quadrature points is important when least-squares formulation is used. Number of quadrature points (NGP) is selected as 4 for quadratic and linear quadrilateral elements and for linear triangular elements. However, due to stability issues NGP is selected as 3 for quadratic triangular elements.

Time step is set to 0.1 for the solutions of the supersonic bump problem. The numerical viscosity produced by the least-squares formulation is a function of time step, of which details are given by Jiang et al. [23]. Selecting a smaller time-step decreases numerical viscosity and as a result shock resolution increases, but it may also cause oscillations close to the discontinuities. The effect of time step will be inspected in detail for the flow over wedge problem.

Pressure contours of the flow field for both meshes and both element orders are given in Figure 3-3. Interactions of shock waves stated in the problem description are clearly seen in the figure. For quadrilateral and triangular linear elements (Figure 3-3 a, c) shock waves are relatively dispersed. Shock resolution improves when second order elements are used, which is expected because the number of nodes almost doubles as compared to the use of linear elements.

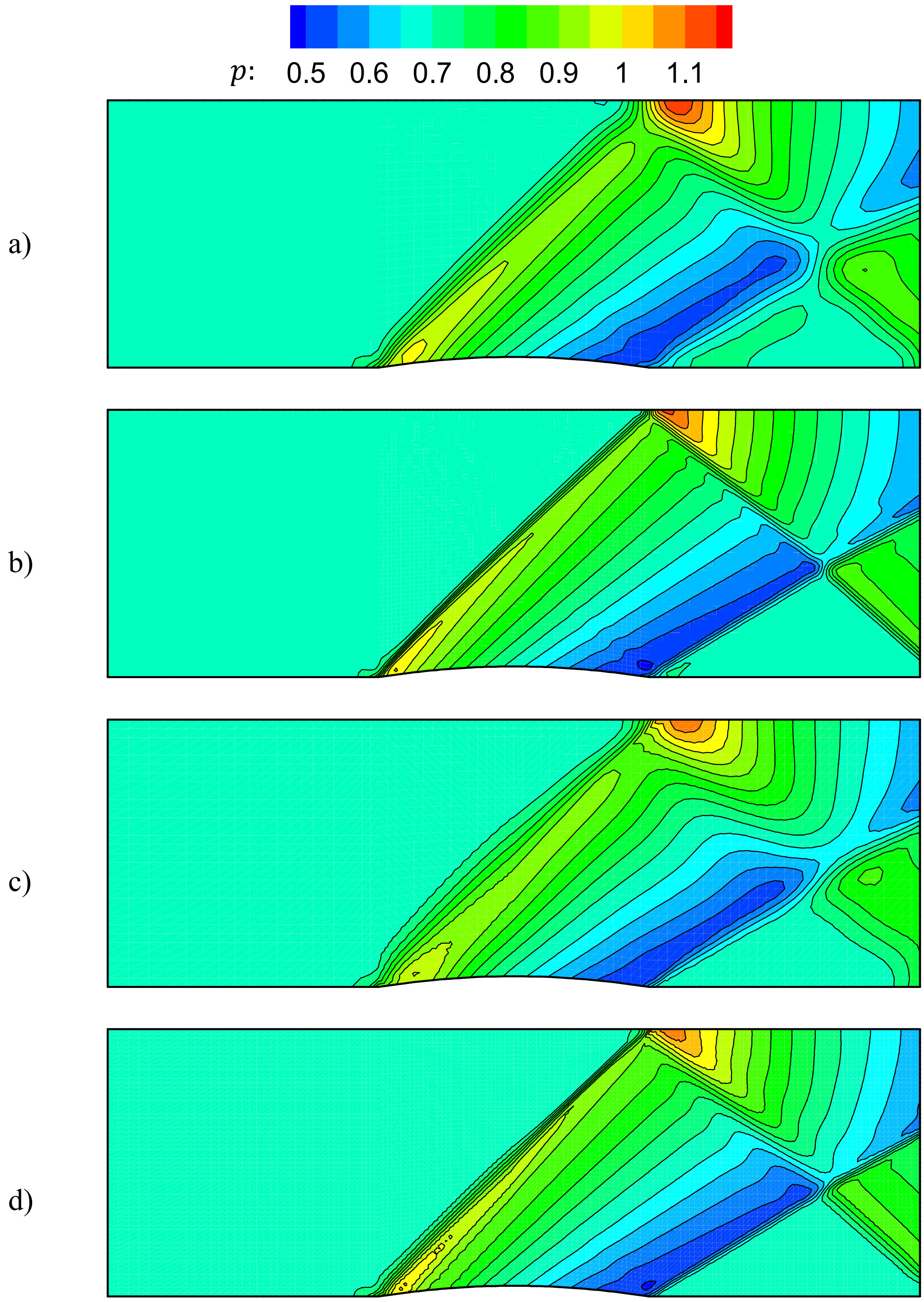

Figure 3-3: Pressure contours for supersonic bump problem with $M_{inlet} = 1.65$ and $\Delta t = 0.1$: (a) linear quadrilateral elements (b) quadratic quadrilateral elements (c) linear triangular elements (d) quadratic triangular elements

Before comparing these results in further detail, let us apply the AMR technique to the mesh of linear triangular elements. The mesh after 6 adaptation cycles can be seen in Figure 3-4. For this adaptive solution first the original triangular mesh of linear elements is used without mesh adaptation for 40 time steps. Then at every 10 iteration mesh is adapted with a pre-defined maximum adaptation level and minimum element size. After mesh is adapted 6 times, iterations continue until the convergence criterion is satisfied. It is clearly seen in Figure 3-4 how AMR increases the mesh resolution at the shock regions. A closer look at the adapted mesh around the leading edge shock is given in Figure 3-5, for which black and red colors indicate initial and final meshes, respectively.

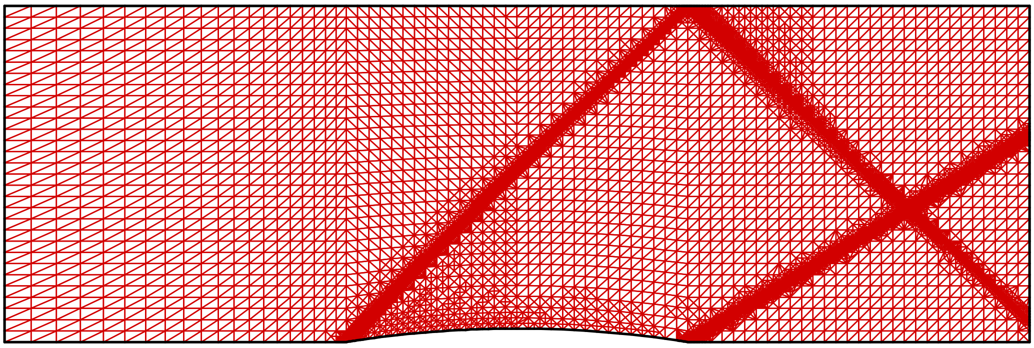

Figure 3-4: Final adapted mesh for supersonic bump problem

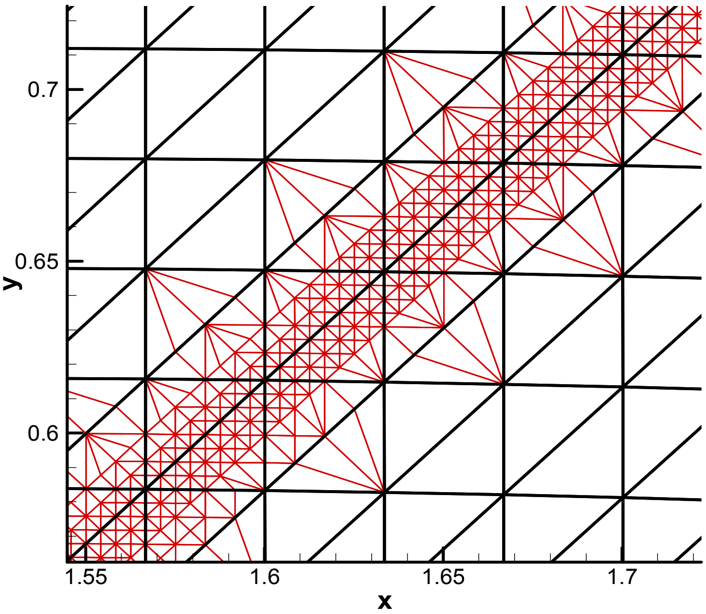

Figure 3-5: Zoomed view of the adapted mesh around the leading edge shock of supersonic bump problem

Pressure contours obtained with the final adapted mesh are given in Figure 3-6.

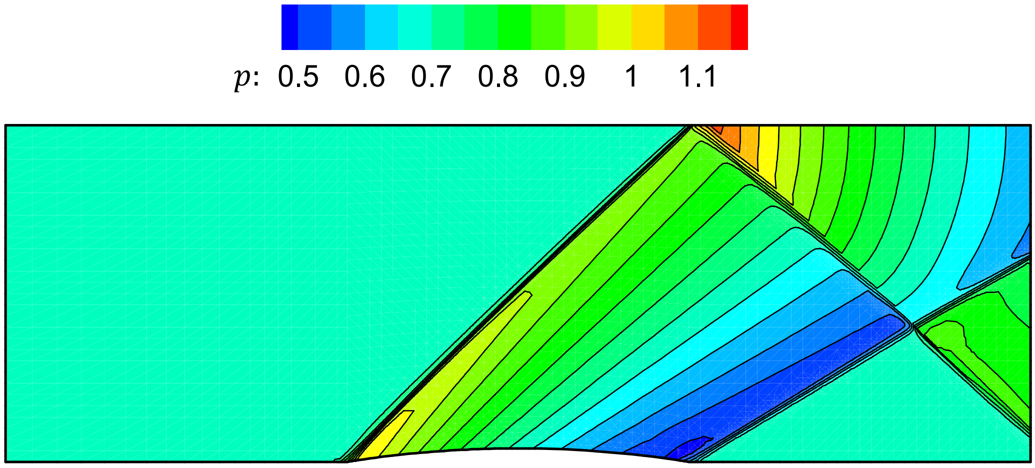

Figure 3-6: Pressure contours of the final adapted solution of the supersonic bump problem

As seen in Figure 3-6, compared to previous non-adaptive solutions shocks are captured sharply without any oscillations in the flow field. A 3-D view of the pressure contour plot is also presented in Figure 3-7. In order to compare shock resolutions better, slices are taken at $y = 0.5$ from quadrilateral and triangular meshes with order $p = 1$ (linear) and $p = 2$ (quadratic) and also from adapted triangular mesh with $p = 1$ (see Figure 3-7) and 5 different solutions are compared in Figure 3-8.

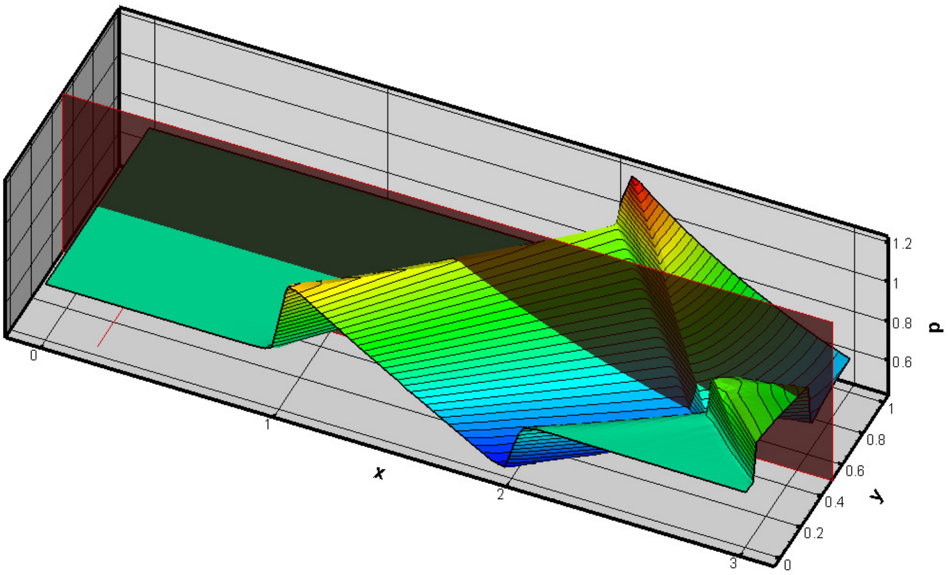

Figure 3-7: Pressure data extraction from y=0.5 for the supersonic bump problem

As seen in Figure 3-7, the slice at the $y = 0.5$ intersects the leading edge shock wave and its reflection from the upper wall, as well as the trailing edge shock wave after its interaction with the reflected shock. All of these flow features can also be seen in Figure 3-8.

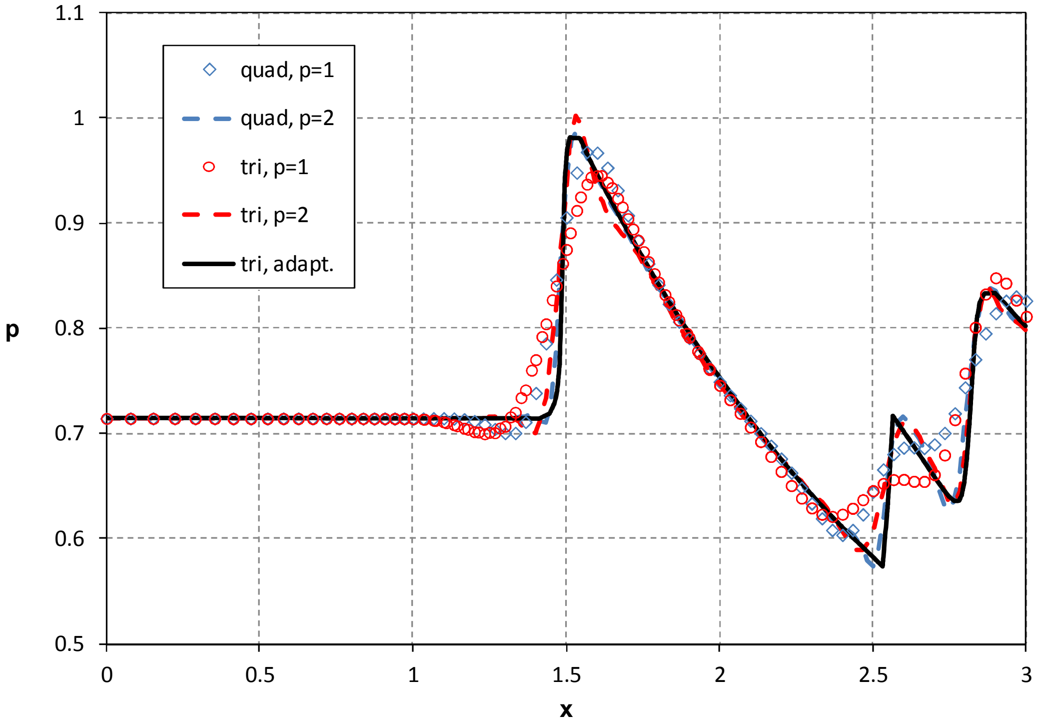

Figure 3-8: Comparison of pressure distribution at y=0.5 for 5 different solutions of the supersonic bump problem

Figure 3-8 illustrates the improvement in the shock resolution with AMR. With linear elements, for both quadrilateral and triangular meshes, shock resolution is poor. Increasing element order, improves the shock resolution to a certain degree, but cannot capture discontinuities as good as the adapted solution.

For further comparison, Mach number distribution at the lower wall is presented in Figure 3-9. The reason of the deviations can be found by analyzing the pressure contours given in Figures 3-3 and 3-6. In these contour plots, it is seen that the shock wave reflecting from the upper wall leaves from the supersonic exit. When the shock resolution is poor, shock wave is smeared and this affects the lower wall near the supersonic exit. Also, there is a shift in the Mach number distribution at the right side of the bump, which is only seen in the neighboring elements to the wall boundary. This effect is smaller at higher order elements and it is not seen in the adapted solution.

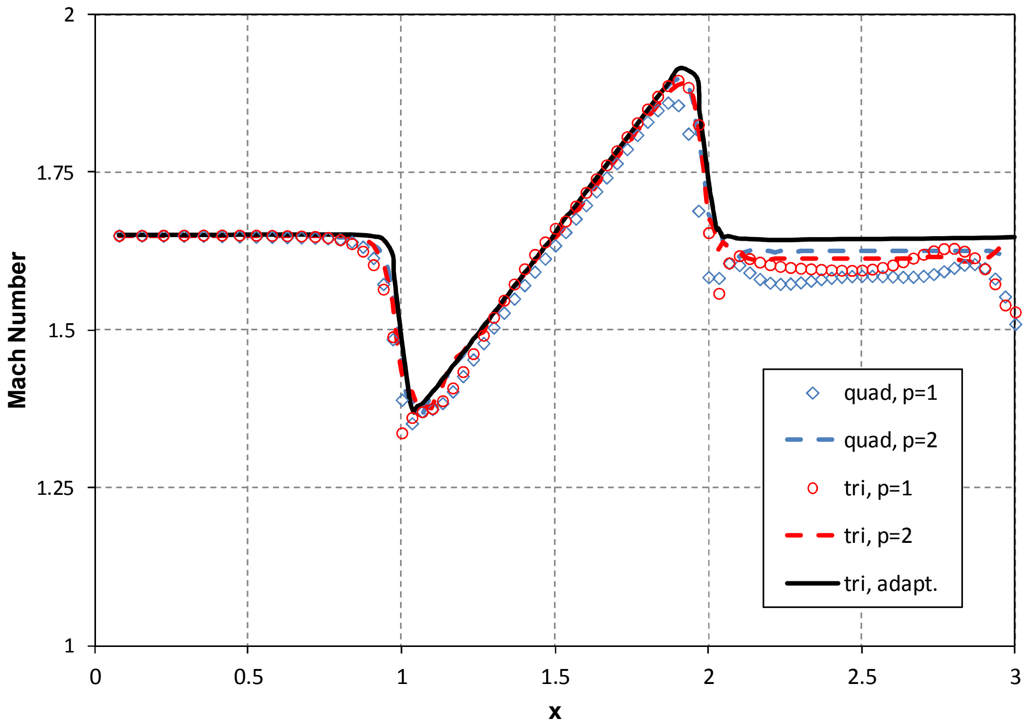

Figure 3-9: Mach number distribution at the lower wall of the supersonic bump problem

In Figure 3-10 Mach number distribution at the upper and lower wall are compared with the solution of Eidelman et al. [33] and Taghaddosi et al. [24]. At the upper wall, results are in good agreement with each other. It is seen that the reflected shock wave at the upper wall is more sharply captured in the current solution and Taghaddosi's solution since both results are obtained with AMR. At the lower wall, there is also good agreement between the results except the deviation after the bump. The Mach number at the lower wall after and before the bump is expected to be the same for an inviscid flow, which is captured only by the current solution.

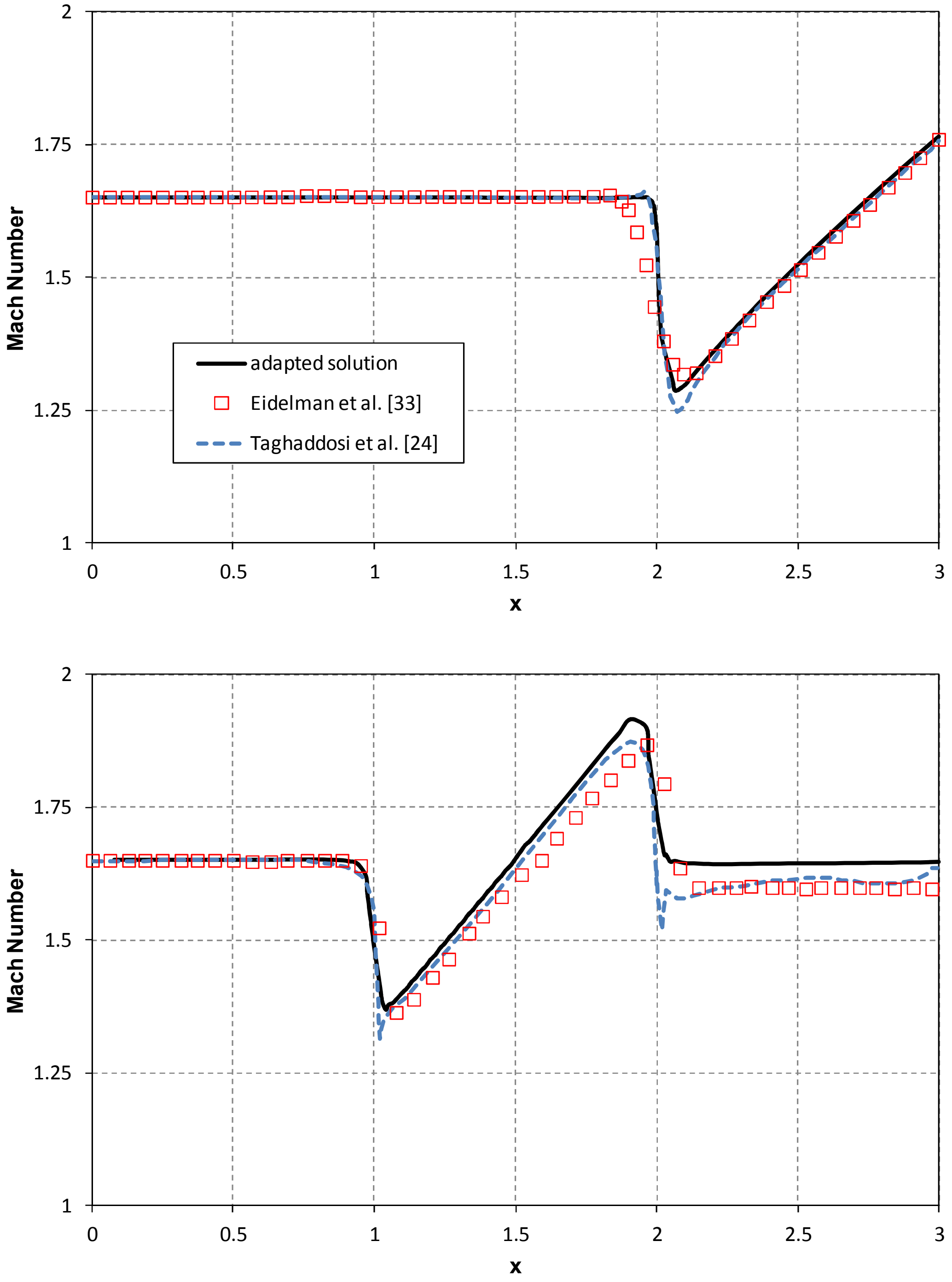

Figure 3-10: Mach number distribution at the upper (top) and lower wall (bottom) for the supersonic bump problem

Convergence histories of all the performed solutions are shown in Figure 3-11. As seen the convergence rate of the solution with quadratic triangular element after 60 iterations is significantly slower compared to the other solutions. In the adapted solution, after 40 time iterations after which the flow features become apparent, mesh adaptation takes place. At every 10 time iterations mesh adaptation is performed without waiting for the residual to drop further and after the final adaptation 100 time iterations was enough for the solution to converge.

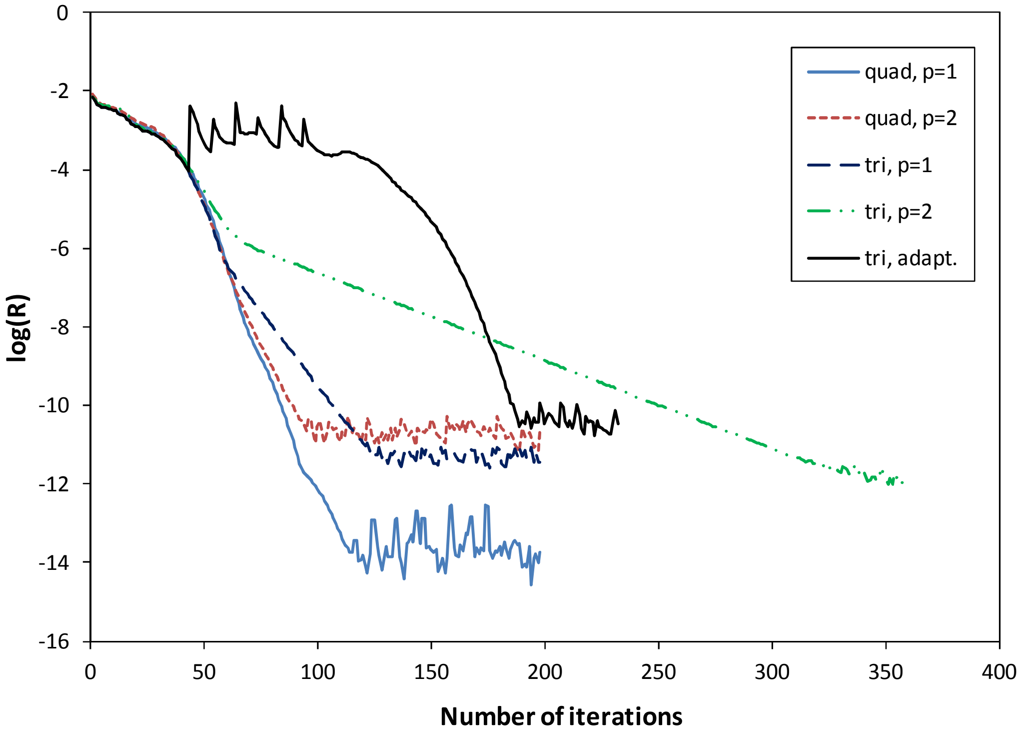

Figure 3-11: Convergence histories of different solutions for the supersonic bump problem

3.2 Transonic Flow Over a Circular Bump

This time, transonic flow with an inlet Mach number of 0.675 passes through a channel with a 10 % blockage by a circular bump. Channel dimensions and the position of the bump are the same as the previously studied problem. Flow enters the channel at a subsonic speed and reaches to supersonic speeds on the bump, which results in the formation of a shock wave.

The inlet boundary condition of the problem is given in the Figure 3-12. The specific heat ratio, γ, is taken as 1.4. At the exit boundary, pressure is set to inlet pressure value of 0.7143. On the upper and lower walls, flow tangency boundary condition is imposed. For this problem, calculations are be done with both quadrilateral and triangular meshes with linear and quadratic interpolating functions, similar to the previous problem, and the results are compared with the data found in the literature [24, 33].

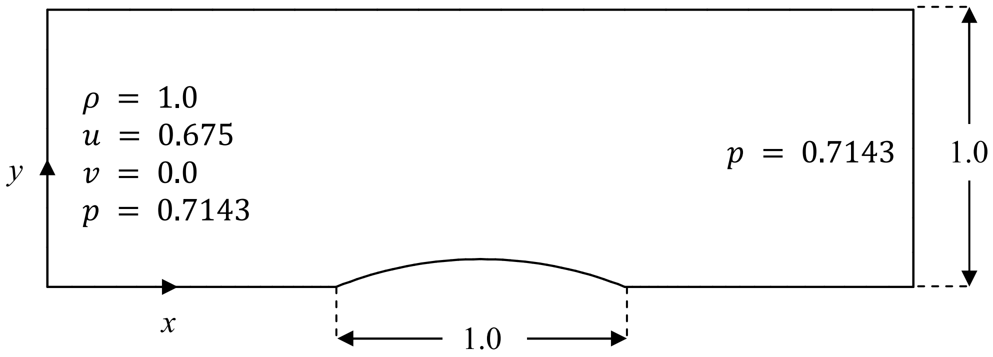

Figure 3-12: Sketch of the transonic flow over a circular bump problem

Similar to the previous problem first the quadrilateral mesh seen in Figure 3-13 is prepared and the triangular mesh is obtained by dividing each quadrilateral element into 2. For each mesh, the bottom wall has 40 uniformly distributed elements on the bump and 20 elements on the both sides of the bump clustered to the bump. There are 30 uniformly distributed elements across the channel height. Both meshes have

2511 nodes for the linear elements. Number of Gauss quadrature points and time step selections are the same as the previous supersonic bump problem.

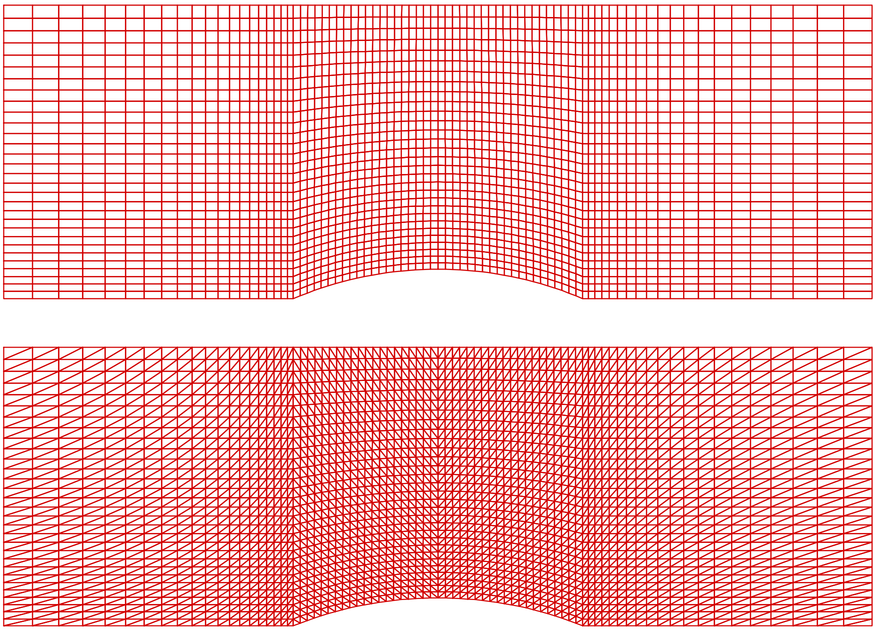

Figure 3-13: Computational meshes of (top) rectangular and (bottom) triangular elements, both having 2511 nodes

Mach number contours of the flow-field for both meshes and both element orders are given in Figure 3-14. It is seen from the figure that there is a viscous effect after the shock wave which is also mentioned in Taghaddosi et al. [24]. This effect is greatest for the use of quadrilateral linear elements. Also for second order elements, oscillatory behavior can be seen in the flow field, especially after the shock wave. In all of the solutions, shock wave is captured sharply and the shock location is approximately at $x = 1.72$ on the bump, which is in good agreement with the literature data.

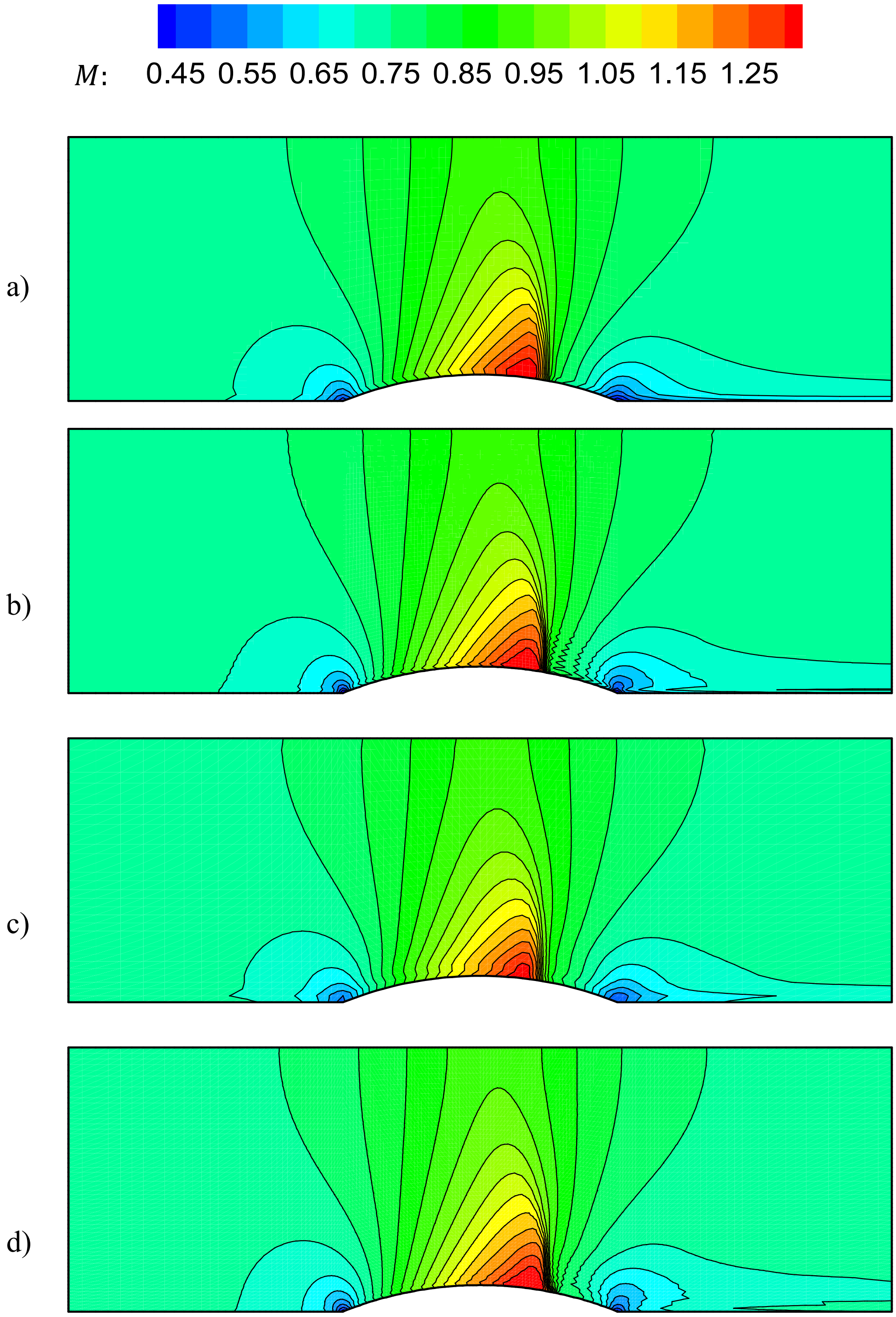

Figure 3-14: Mach number contours for transonic bump problem with $M_{inlet} = 0.675$ (a) linear quadrilateral elements (b) quadratic quadrilateral elements (c) linear triangular elements (d) quadratic triangular elements

Mach number distribution at the lower wall is given in Figure 3-15. Results are in good agreement with the literature data except small shifts in the Mach number after the bump, which is related to the viscous effect mentioned in the previous paragraph. This effect is higher for quadrilateral elements and lower for triangular elements of which results agree well with Ni [34]. Taghaddosi et al. [24] mentioned in their paper about this effect and compared their results with Eidelman et al. [33] and Ni [34]. In this comparison they observed a similar deviation of Mach number, which amplifies by the use of larger time steps [24].

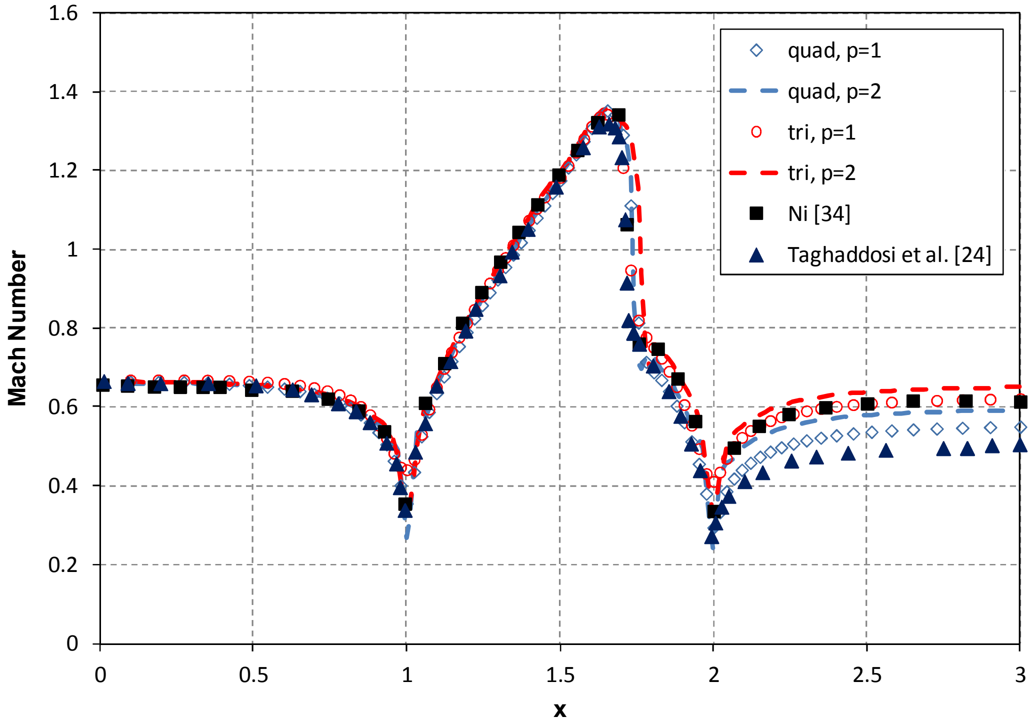

Figure 3-15: Mach number distribution at the lower wall for different element types and orders

Before investigating the results further, solution with AMR will be presented. Unlike the previous problem, AMR is done with a coarser mesh. The generated mesh seen in figure 3-16 consists of 797 nodes which is approximately three times less than the previous mesh. Adaptive computations with this coarse mesh are done with a time step of 0.2. Adaptation took place at every 20 time iterations until reaching 8 levels of adaptation. Zoomed view of initial and final adapted mesh at the shock position on the bump is given in Figure 3-17, which illustrates the improvement in the grid resolution at the shock wave. The modified grid and pressure contours near bump at different adaptation levels are given in Figure 3-18 which shows the improvement in shock resolution.

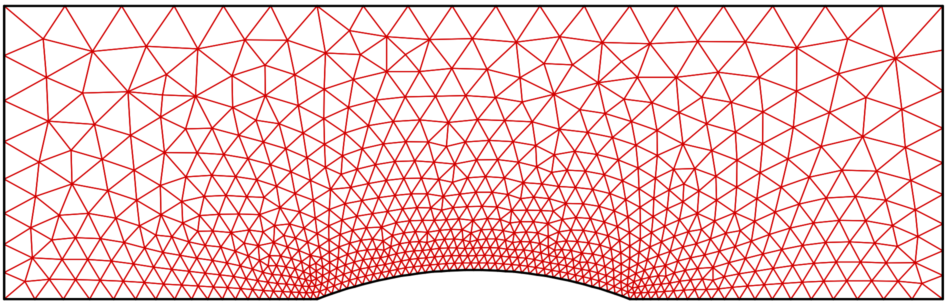

Figure 3-16: Initial coarse mesh for AMR

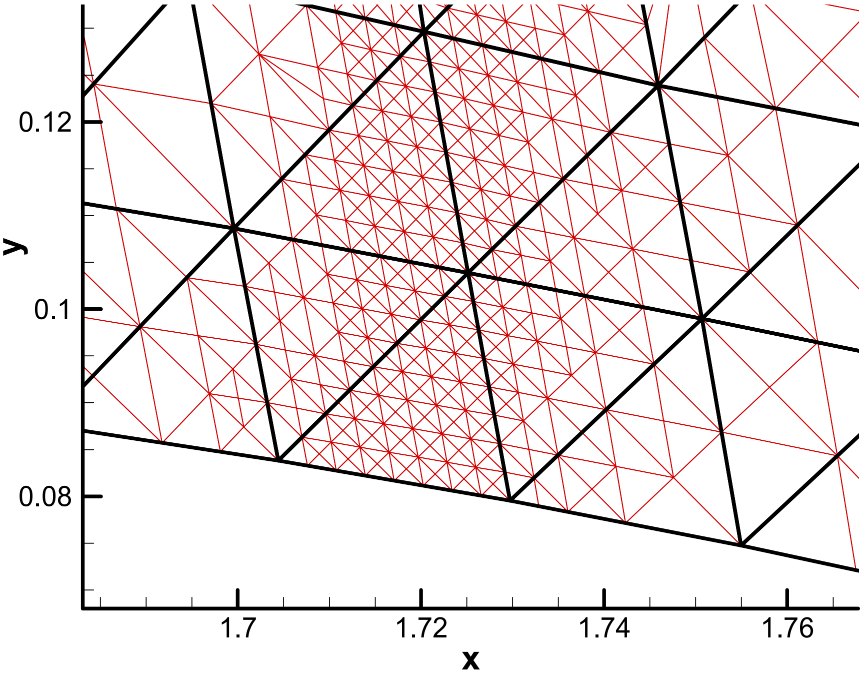

Figure 3-17: Initial (black) and adapted (red) mesh for the transonic bump problem

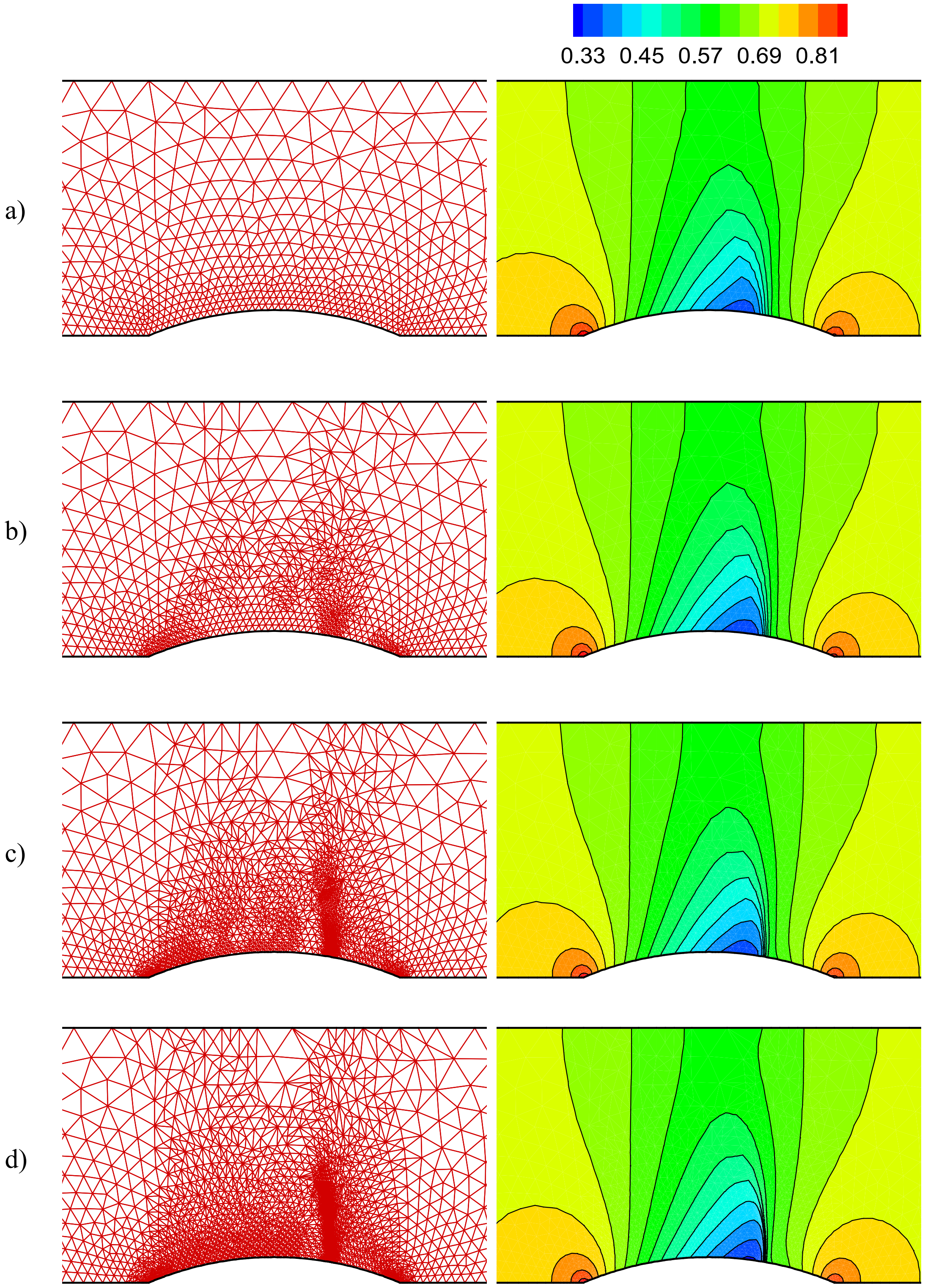

Figure 3-18: Initial and adapted meshes and pressure contours near bump, $\Delta t = 0.2$: (a) Initial mesh, (b) 2nd adaptation, (c) 4th adaptation, (d) 8th adaptation

In Figure 3-19, Mach number distribution over the lower wall obtained with the adapted solution is compared with the literature data. Adapted solution results agree very well with the results of Eidelman et al. [33] and Ni [34] throughout the channel wall and with Taghaddosi et al. [24] before the trailing edge. After the trailing edge, results of Taghaddosi et al. under predict the Mach number as the authors stated. They used a coarse quadrilateral initial mesh and employed r-type refinement and after the adaptation grid became even coarser at the downstream of the bump, which apparently increased the viscous effect. Similar behavior is also observed in the current study, especially with non-adaptive solutions as discussed before. In order to see this effect more clearly, Mach number distribution at the exit plane is given in Figure 3-20. It is seen that underestimation of Mach number at the lower wall due to viscous effect is limited to the elements right above the wall boundary only.

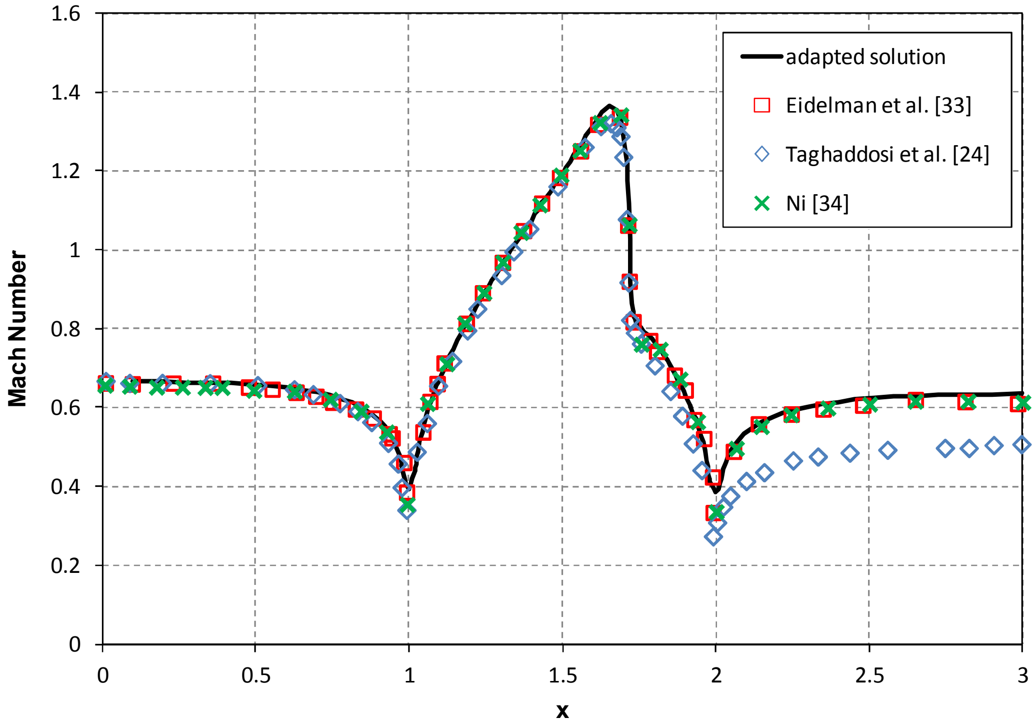

Figure 3-19: Mach number distribution at the lower wall of adapted solution of the transonic bump problem

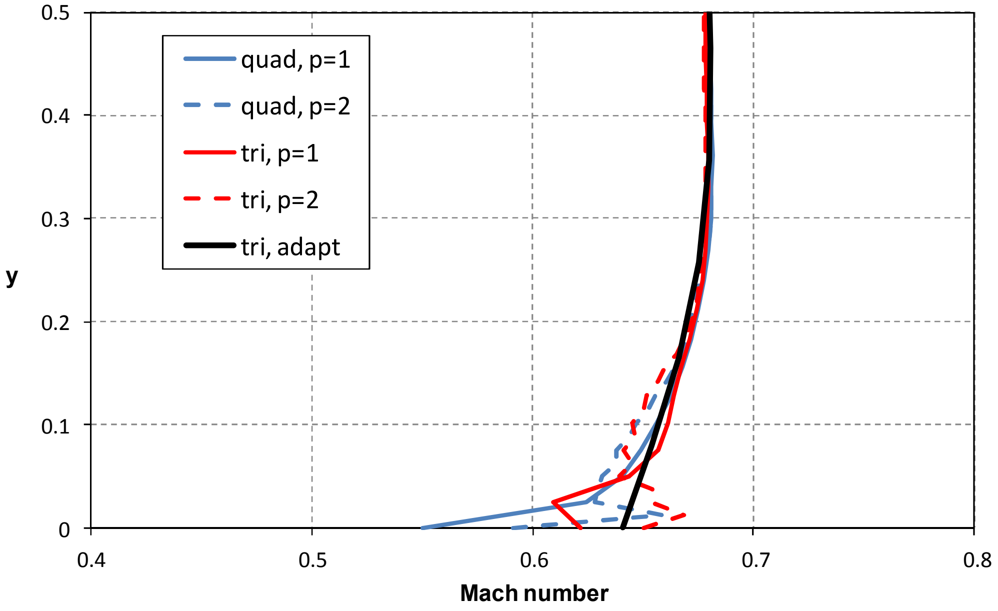

Figure 3-20: Mach number distribution at the exit of the transonic bump problem

Convergence history of the solutions are given in the below graph.

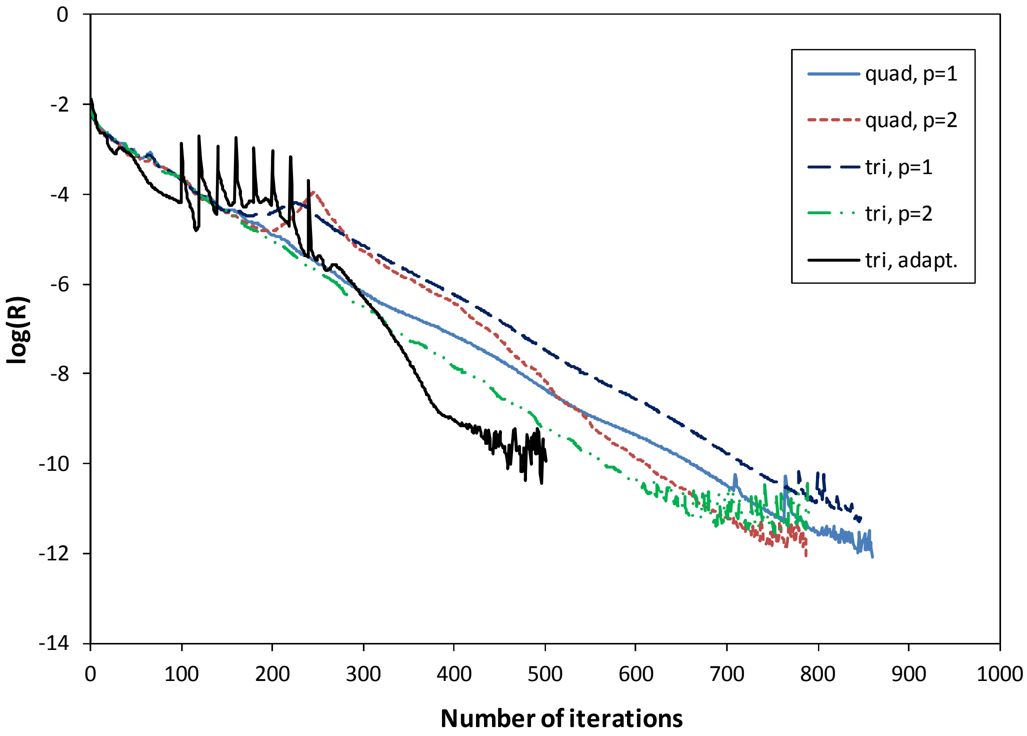

Figure 3-21: Convergence histories of different solutions for the supersonic bump problem

3.3 Supersonic Ramp Flow

In this test case, supersonic flow with an inlet Mach number of 2.0 flows over a wedge which makes a 5° angle with the free stream flow as seen in Figure 3-22. Inlet boundary conditions are given in the figure, flow tangency boundary condition is imposed at the lower and upper walls and the right boundary is left free. Gas enters the domain through the inlet at the left and an oblique shock wave emanates from the leading edge of the ramp with an angle of 34.3°. Flow properties after the shock wave can be calculated analytically. In this problem, first the effect of time step in LSFEM formulation will be inspected and then LSFEM with AMR technique will be tested by solving the problem with a very coarse initial mesh.

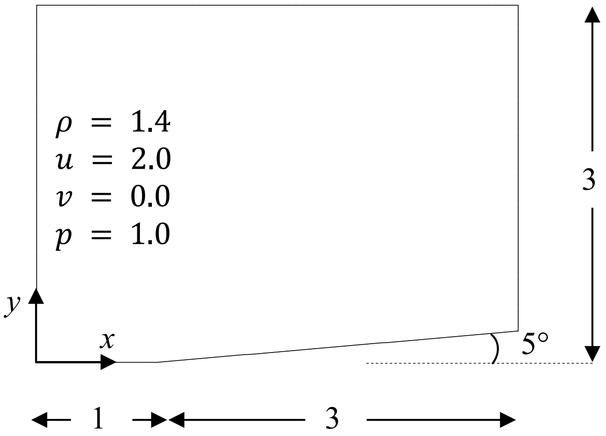

Figure 3-22: Sketch of the supersonic ramp flow

Numerical viscosity produced by the least-squares formulation is known to be a function of time step [23]. Using a smaller time step decreases numerical viscosity and as a result shock resolution increases. However, it may also cause oscillations in the solution. In order to see the effect of time step on numerical viscosity and shock resolution, triangular mesh given in Figure 3-23 is generated.

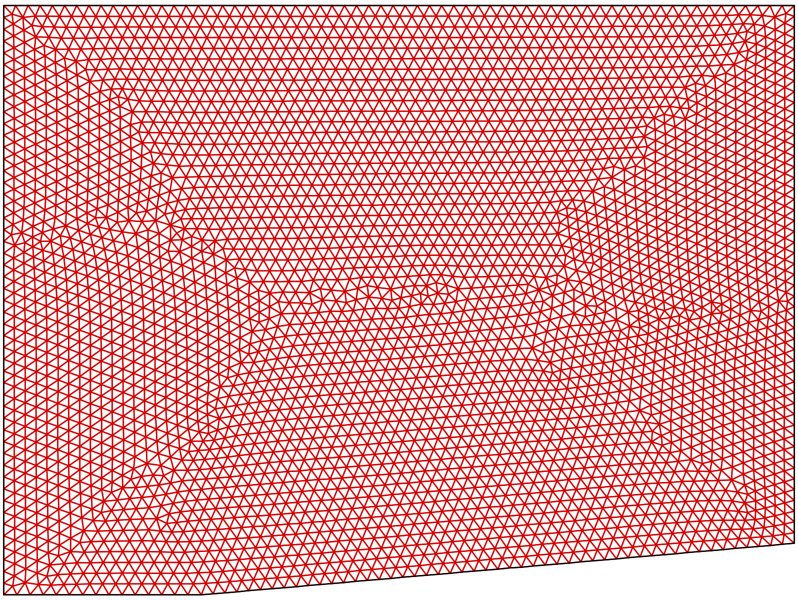

Figure 3-23: Uniform mesh for the supersonic ramp flow (7512 elements, 3872 nodes)

Pressure contours for time step values of 0.3, 0.1, 0.05 and 0.02 are given in Figure 3-24. The effect of time step on shock resolution can be seen in this figure. As expected, as the time step decreases shock wave becomes thinner. However, by decreasing the time step we also observe oscillations near the shock region. In order to see this behavior clearly, pressure data at $x = 2.8$ is presented in Figure 3-25. Shock wave intersects the $x = 2.8$ line approximately at $y = 1.226$.

As seen in Figure 3-25 as the time step gets smaller shock resolution gets better, but overshoots and undershoots before and after the shock are amplified, clearly showing the link between the time step and numerical dissipation. It is also observed that decreasing time step results in a small shift of the shock position. The possible reason of this is the oscillations before the shock wave. Another possible reason is stated in the literature as using non-conservative form of the governing equations [28, 35]. Further discussion of this issue will be done after solving the problem with AMR.

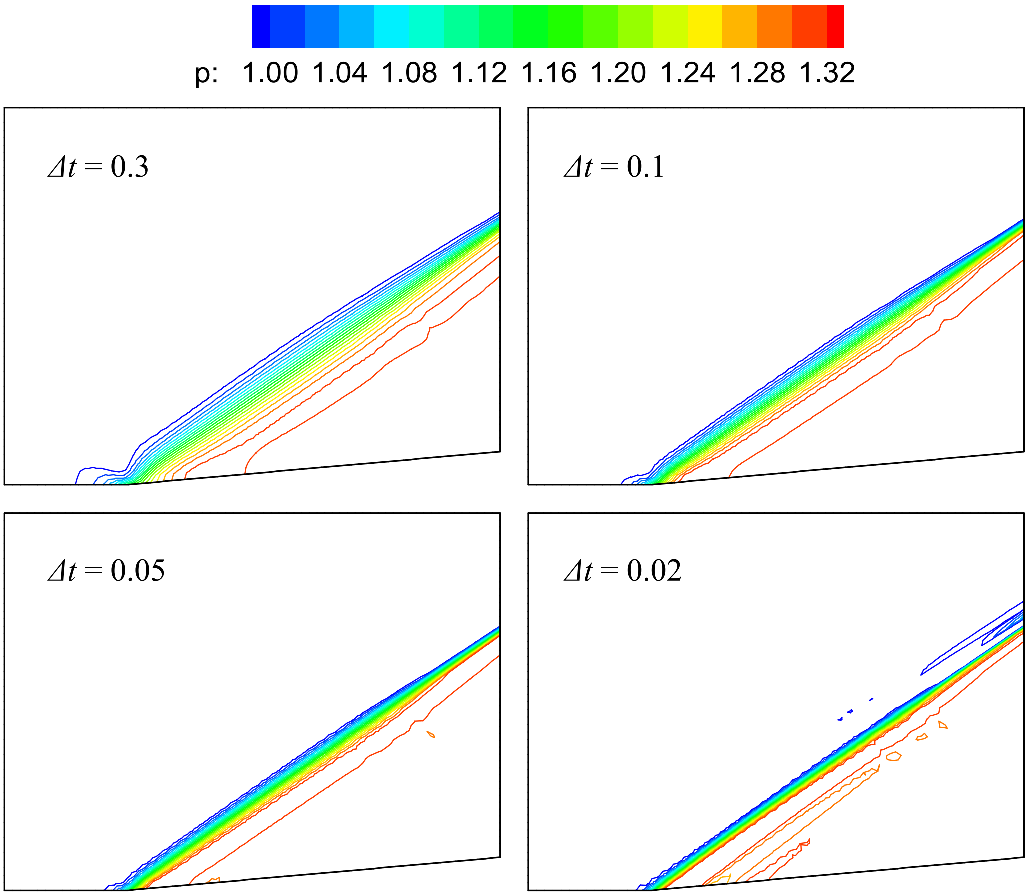

Figure 3-24: Pressure contours of supersonic ramp flow obtained with different Δt

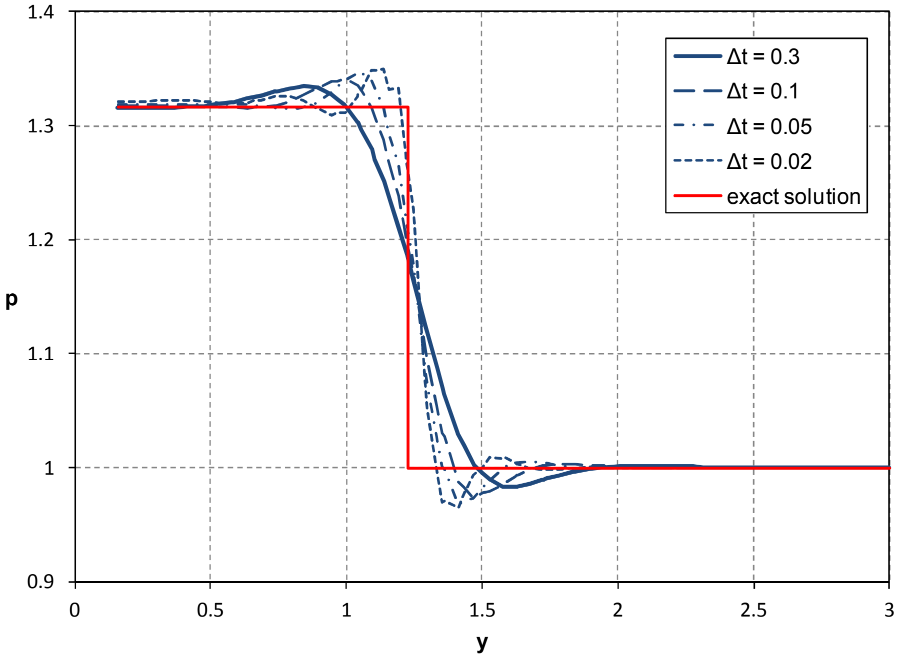

Figure 3-25: Effect of time step on pressure distribution at x=2.8 for supersonic ramp flow

It is seen that, the uniform grid used for this problem (Figure 3-23) is not fine enough to capture the shock wave accurately without oscillations. AMR can be used for better shock capturing. To show the real power of AMR, a very coarse initial mesh of 113 nodes and 194 elements (approximately 40 times coarser than the previously used uniform grid) is generated (Figure 3-26).

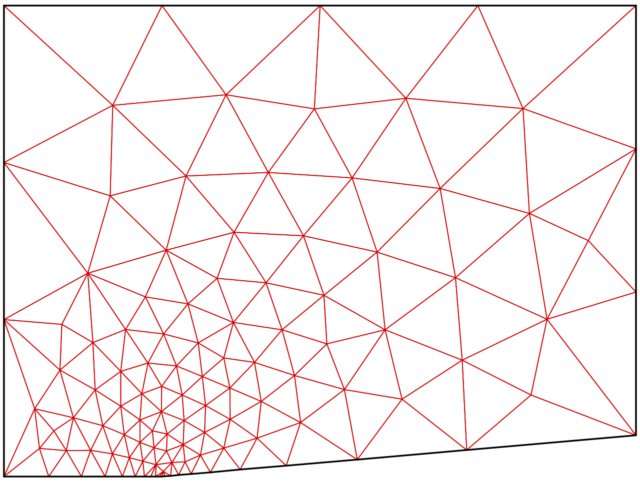

Figure 3-26: Coarse initial mesh for solving the supersonic ramp problem with AMR (113 nodes and 194 elements)

Adaptive solution of the problem is started with a time step of 0.2 and after 7 adaptation cycles, time step is reduced to 0.1 and 2 more mesh adaptation is done before finalizing the solution. Figure 3-27 shows the modified grid and improvement in shock resolution at different adaptation levels. After the final adaptation, number of elements and nodes become 11303 and 5684, respectively. It can be seen in Figure 3-29 that only the elements at the shock location are refined and undesired mesh propagation is avoided.

Because the solution started from a very coarse initial mesh, the final mesh contains very skew elements near the shock region, as seen in Figure 3-29. In order to see the effect of mesh skewness on the solution, a new simulation is done with a modified AMR algorithm, in which skew elements are also refined to make them more orthogonal. However, it is seen that this brings no improvement to the solution, but it only increases computational work and solution time. So it can be concluded that LSFEM can work satisfactorily with the skew elements seen in Figure 3-29.

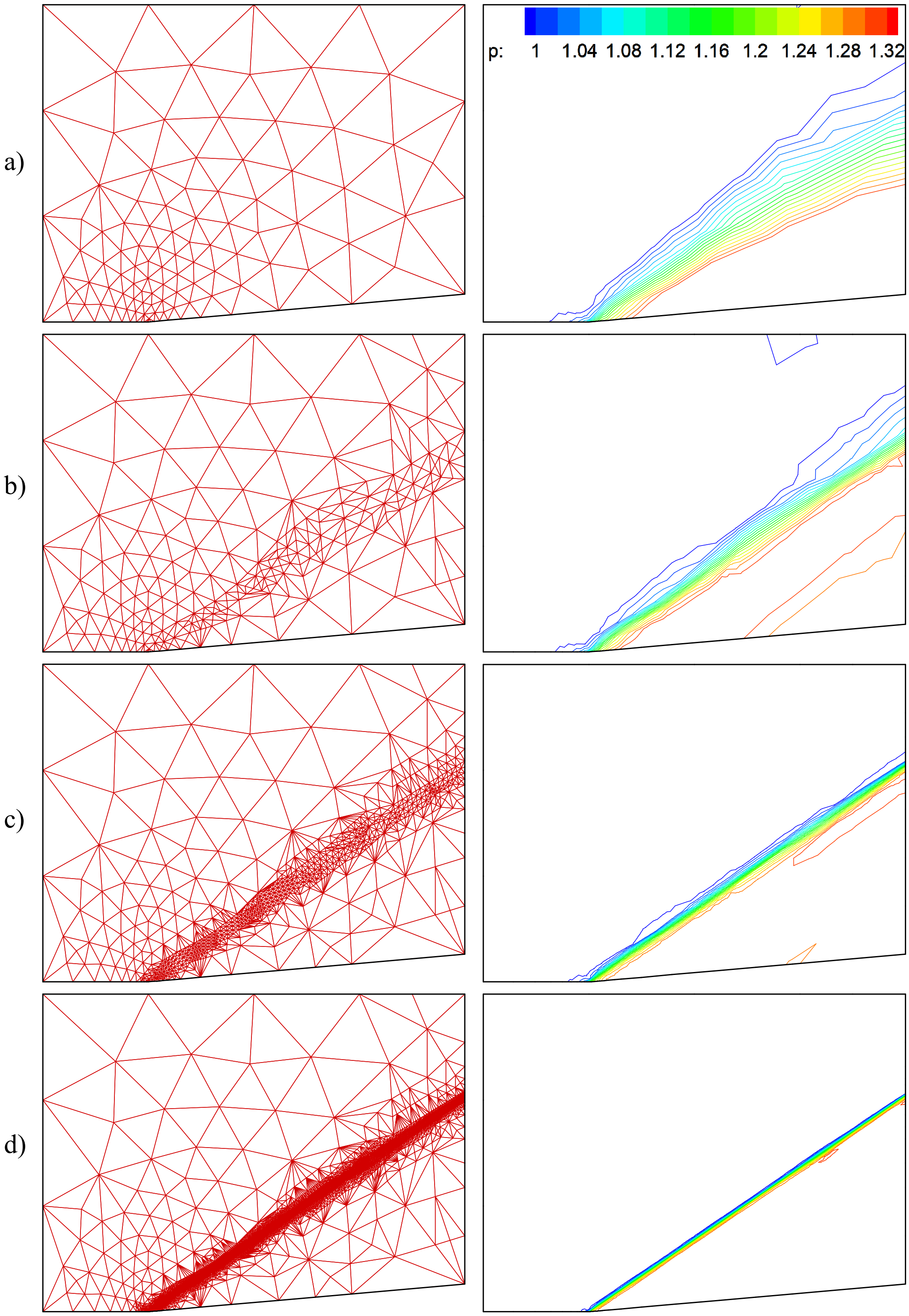

Figure 3-27: The grids and pressure contours for initial and adapted solutions of the supersonic ramp problem: (a) Initial mesh, (b) 3rd adaptation, (c) 6th adaptation, (d) 9th adaptation

The initial and final adapted mesh is shown together in Figure 3-28 to clearly illustrate the refinement process. Zoomed view to the refined elements at shock position is also presented in Figure 3-29.

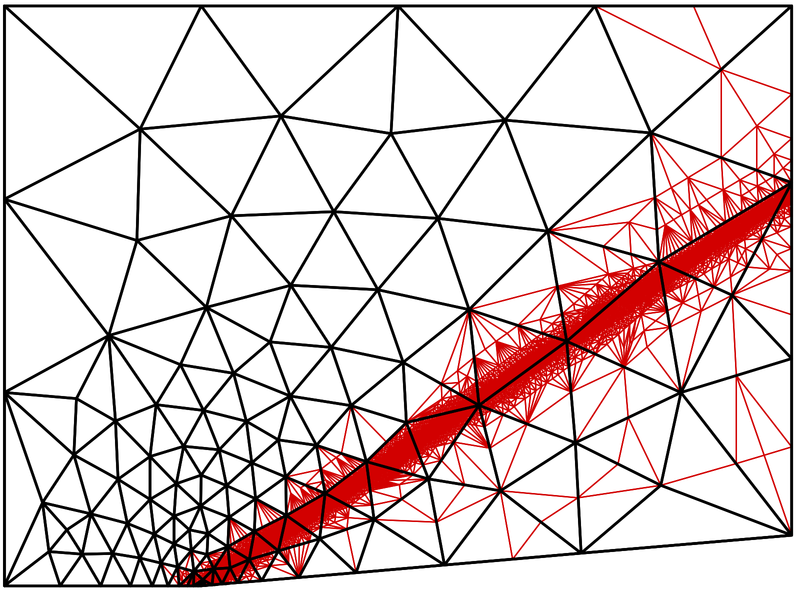

Figure 3-28: Initial and final adapted mesh used for the supersonic ramp problem

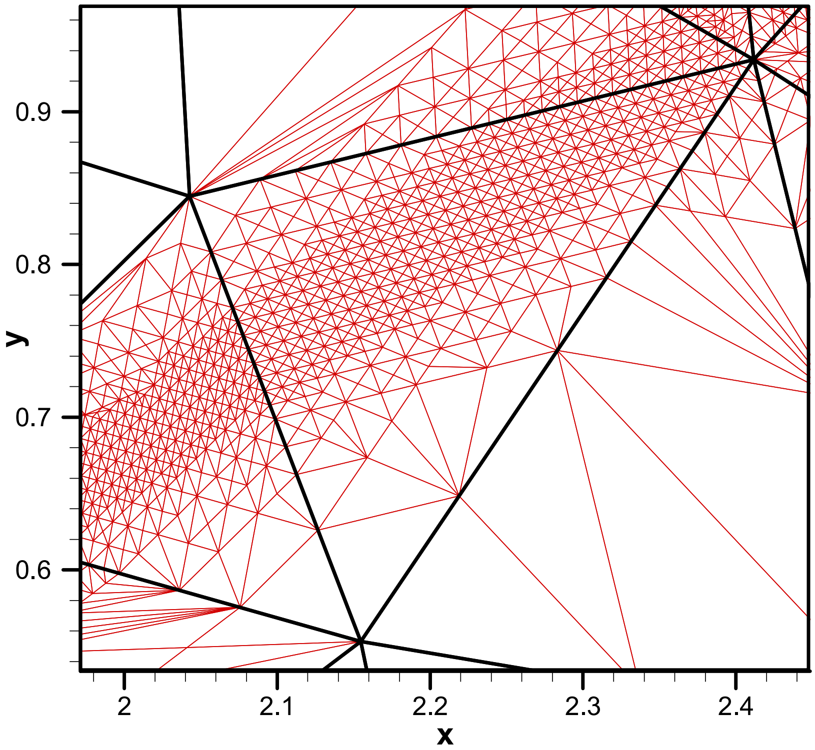

Figure 3-29: Zoomed view of the initial and adapted mesh at shock location

Figure 3-30 shows the pressure distribution of the adapted solution at $x = 2.8$. Adapted LSFEM solution predicts the shock position correctly and provides high shock resolution without oscillations. Comparing this result with the non-adaptive ones (Figure 3-25), improvement in the shock resolution is quite significant. It is also important to mention that, unlike previous observations, no deviation from the exact shock location is observed by using the non-conservative formulation with AMR algorithm.

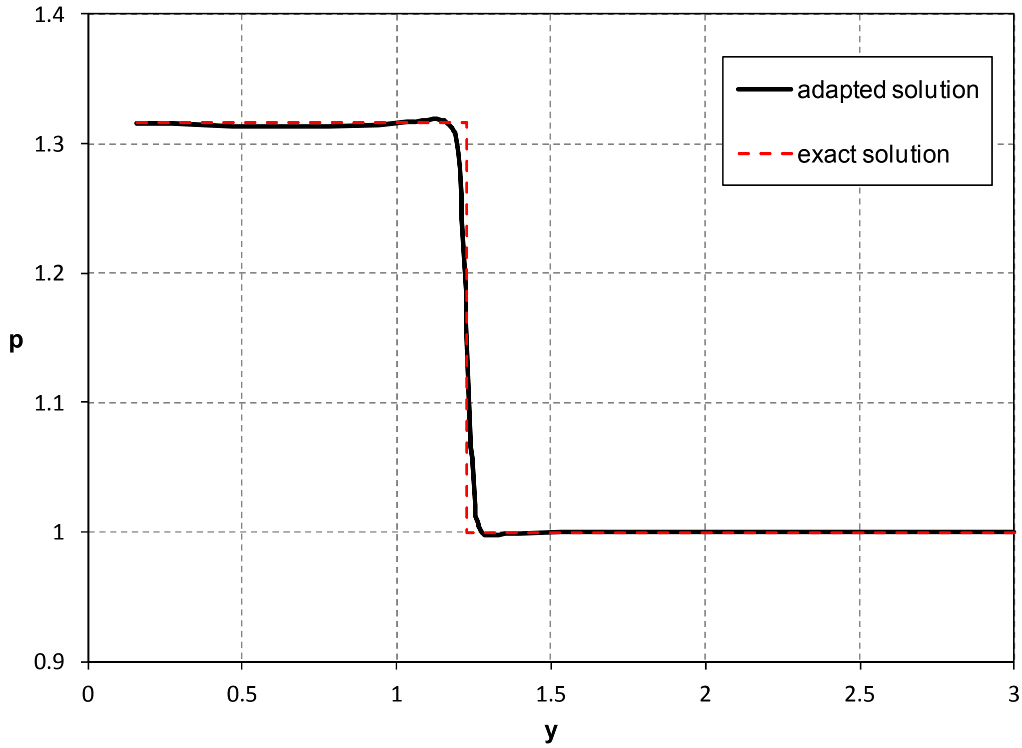

Figure 3-30: Pressure distribution for adapted and exact solution at $x = 2.8$ for the supersonic ramp problem

Finally, convergence history of the adapted solution is given in Figure 3-31. In the solution of the problem, adaptations are successively performed without waiting for the residual to drop further up to the final adaptation step after which iterations continued until the solution reaches the steady state. After 7 adaptations, time step is reduced from 0.2 to 0.1 which is followed by 2 more adaptations.

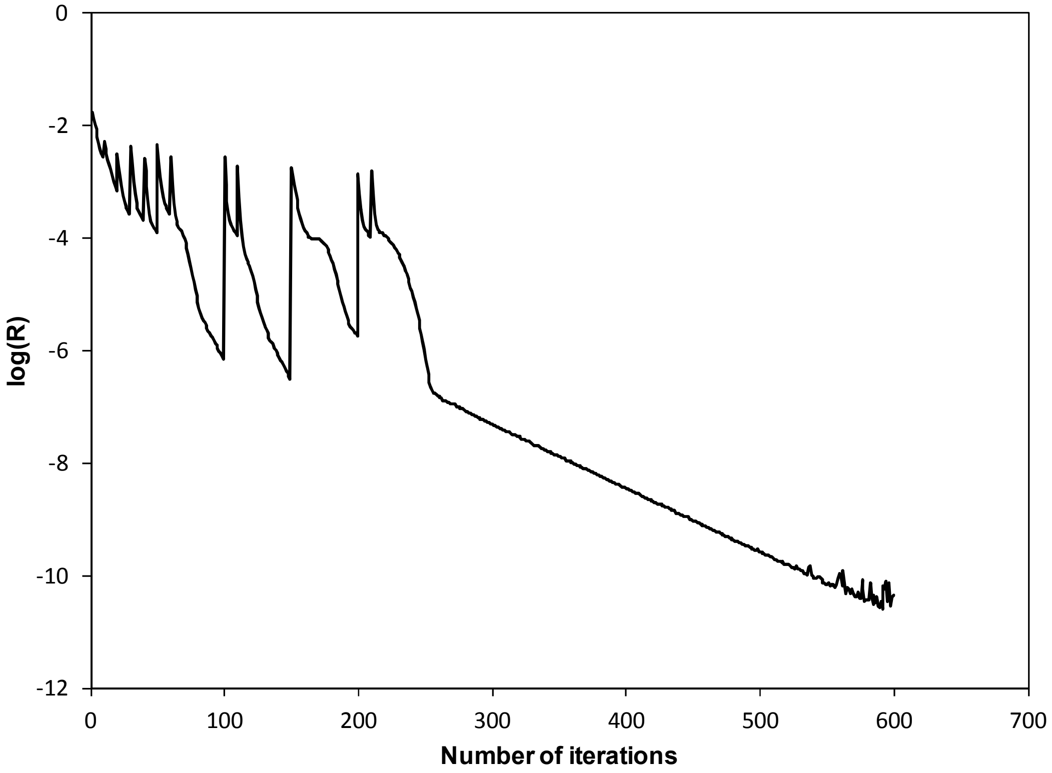

Figure 3-31: Convergence history of the adapted solution of the supersonic ramp problem

3.4 Flow Over an Axisymmetric Body

In this problem, supersonic flow of Mach number 3.0 over an axisymmetric body is solved and results are compared with available data in literature [36]. The axisymmetric body geometry is six diameters in length which has a three diameter length secant-ogive nose with a radius of 18.88 diameters.

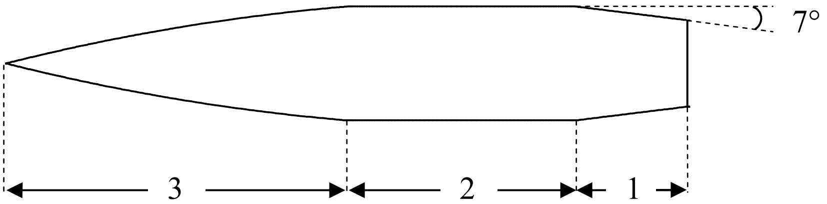

Figure 3-32: Axisymmetric body geometry (dimensions are in body diameters)

The boundary conditions for the freestream Mach number of 3.0 are $\rho = 1.0$, $u = 3.0$, $v = 0.0$ and $p = 0.7143$. The computational domain does not include the base of the axisymmetric body. Flow tangency boundary condition imposed at the axisymmetric body and top and right boundaries are left free. Initial mesh around the body is given in the Figure 3-33. Time step is set to 0.05 for the solution of the problem. The mesh is adapted to get the final results. Adapted mesh and pressure contours can be seen in the Figure 3-34.

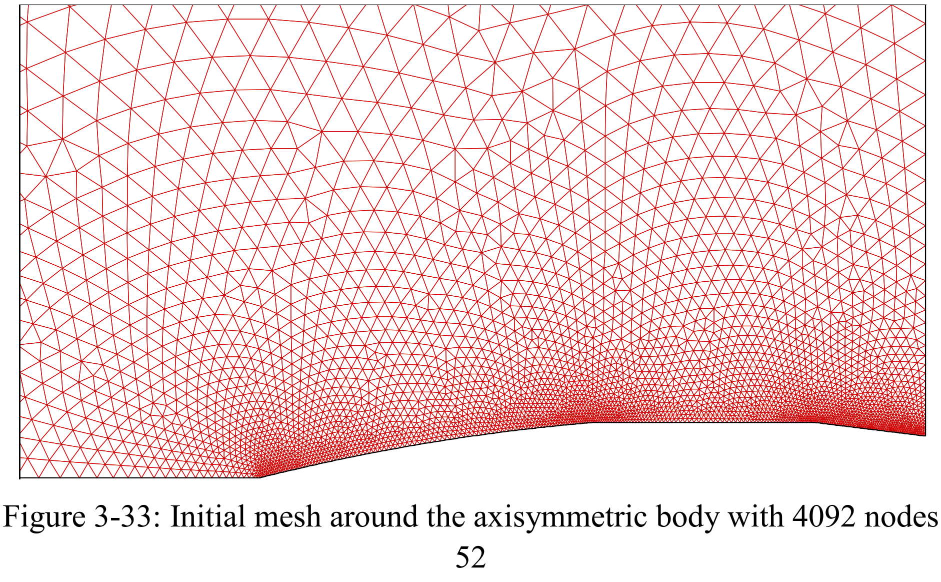

Figure 3-33: Initial mesh around the axisymmetric body with 4092 nodes

As seen in 3-34, oblique shock wave attached to the nose of the body is captured well. Expansion waves at the end of the nose and at the boattail can also be seen.

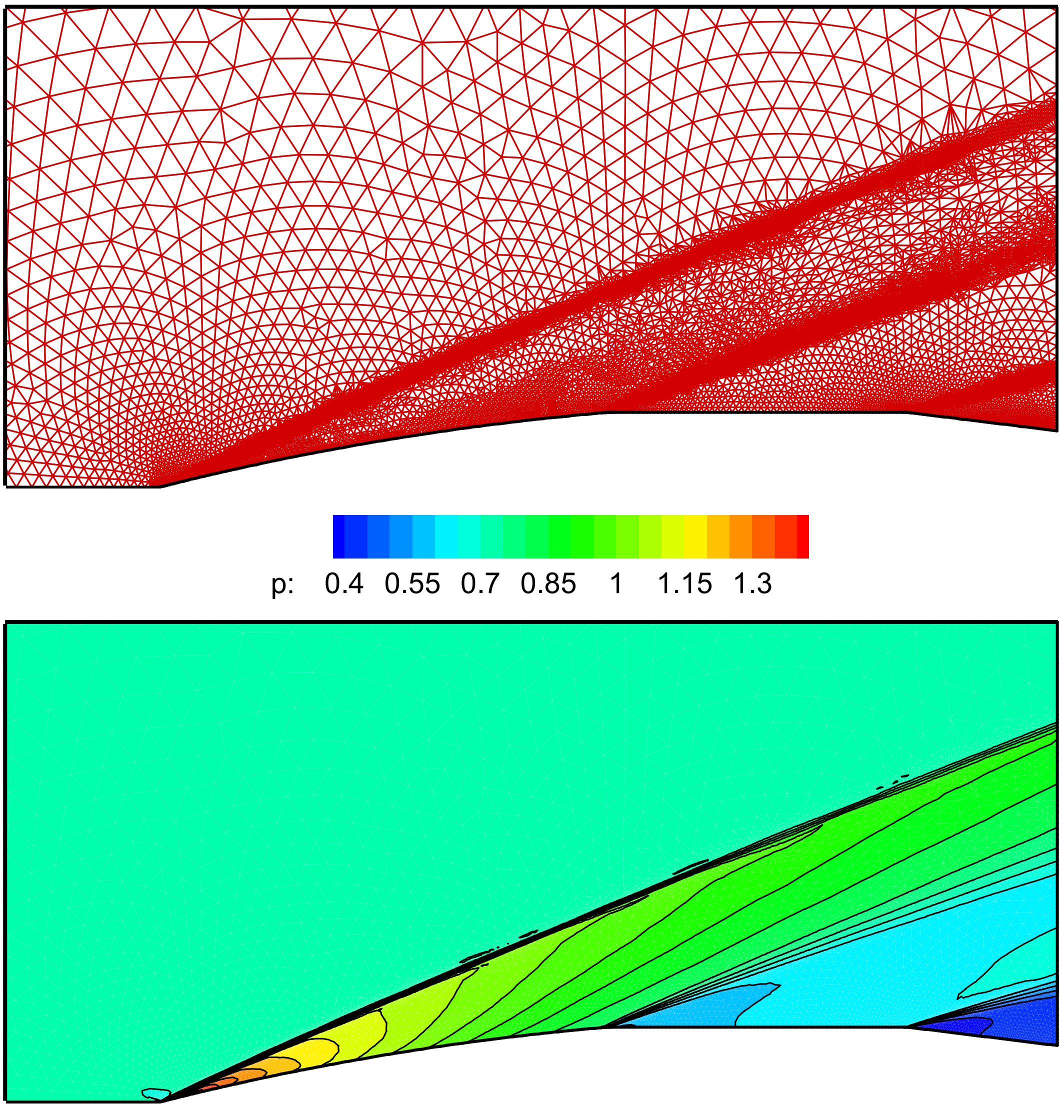

Figure 3-34: Adapted mesh and pressure contours of the axisymmetric flow problem

In Figure 3-35, surface pressure ratio is compared with the one found in literature and good agreement is observed.

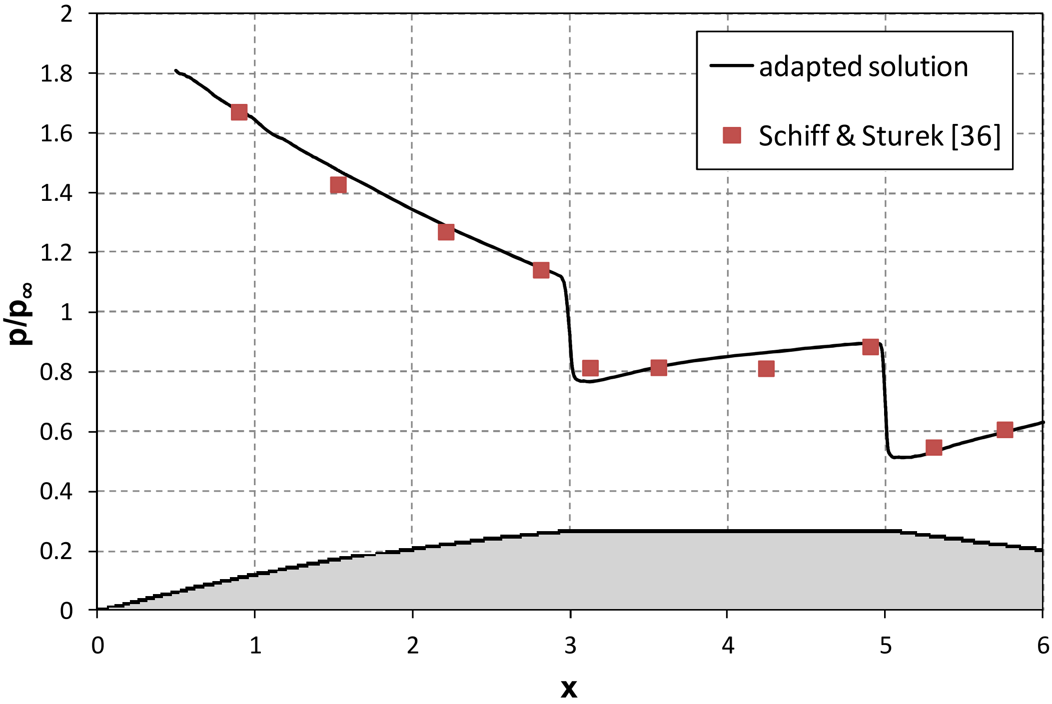

Figure 3-35: Pressure distribution on the axisymmetric body

3.5 Flow Through Scramjet Inlet

In this final test case, supersonic flow of Mach number 2.0 enters the scramjet inlet and compressed through a 10° ramp as seen in Figure 3-36. Inlet boundary conditions are also given in figure. Flow tangency boundary condition is imposed at the solid wall of the inlet by penalty method and only the upper half of the flow domain is studied by setting the velocity in the vertical direction to zero at the symmetry line. Supersonic exit is left free.

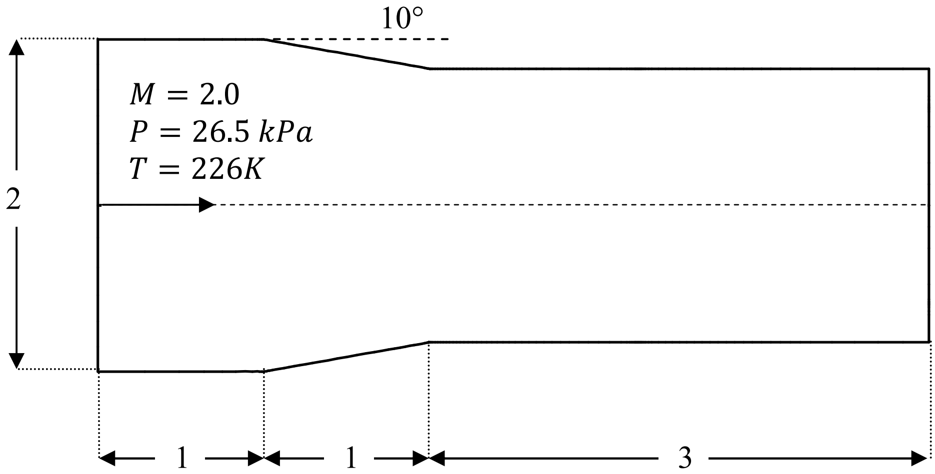

Figure 3-36: Sketch of the inlet problem (all dimensions are in meters)

Initial grid consists of 1422 triangular elements and 775 nodes and can be seen in Figure 3-37. Solution started with a time step of 0.1 and after 8 adaptations, numerical viscosity reduced by setting the time step to 0.05 followed by 2 more adaptations which can also be seen in Figure 3-40 from the jumps of the residuals.

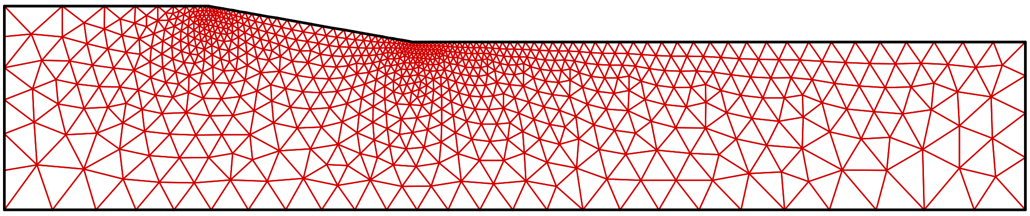

Figure 3-37: Initial mesh for the scramjet inlet problem with 1422 elements and 775 nodes

In Figure 3-38 adapted grid and pressure contours of the solution are given. Final adapted mesh consists of 20790 elements and 10512 nodes. It is seen that, shock waves and expansion fans are sharply resolved by the adapted grid and as a result discontinuities are captured with a high level of accuracy.

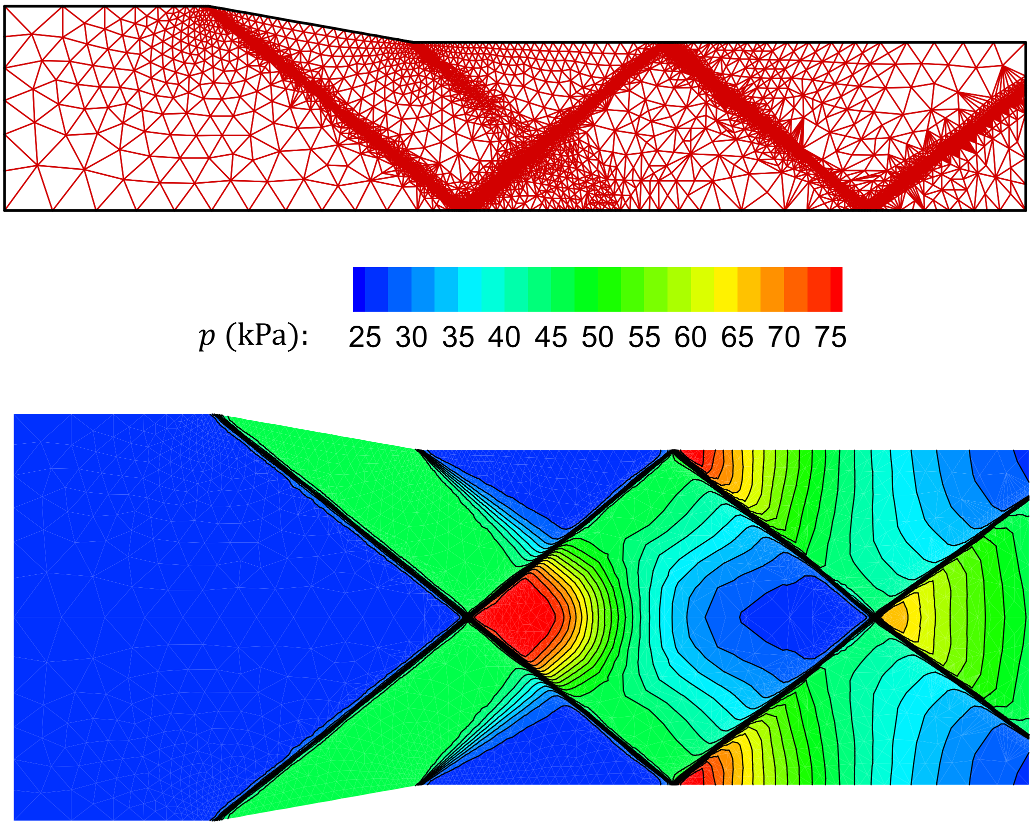

Figure 3-38: Adapted mesh and pressure contours of the scramjet inlet problem

Pressure coefficient along the wall, compared with the results of Hosseini et al. [37] is given in Figure 3-39. Adaptive LSFEM results are in good agreement with the literature data. Discontinuities are sharply captured without oscillatory behavior by LSFEM with AMR algorithm.

Convergence history of the adapted solution is given in Figure 3-40. The jump at iteration 300 corresponds to the change in the time step and others occurred due to mesh refinement.

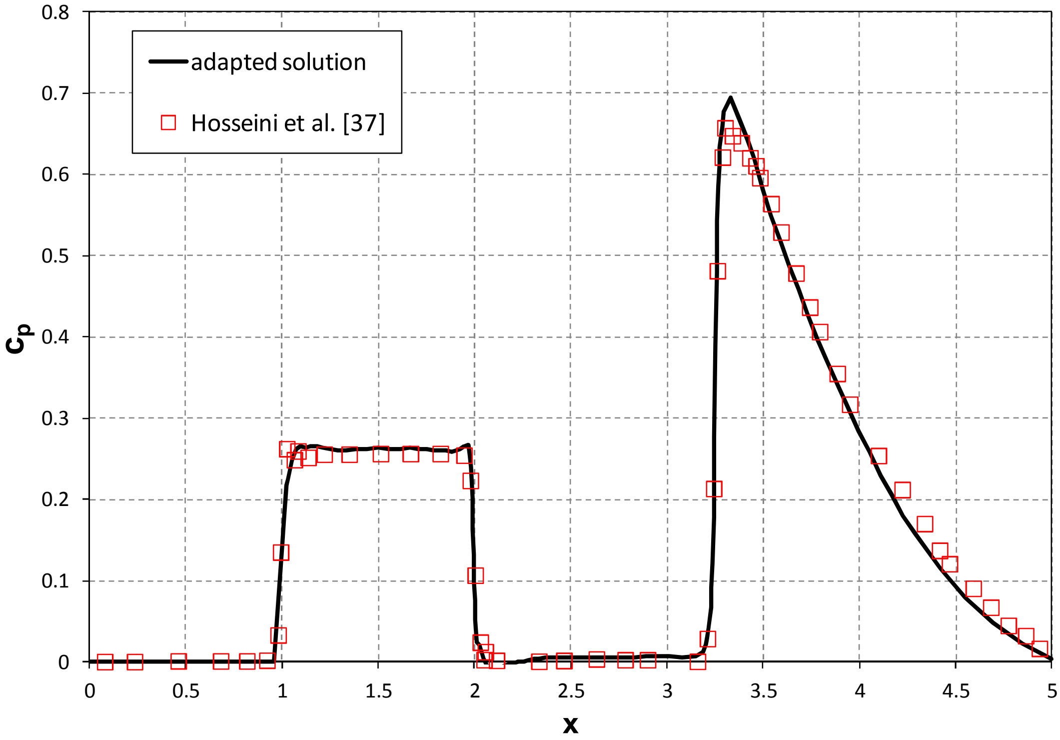

Figure 3-39: Pressure coefficient along the wall of the scramjet inlet

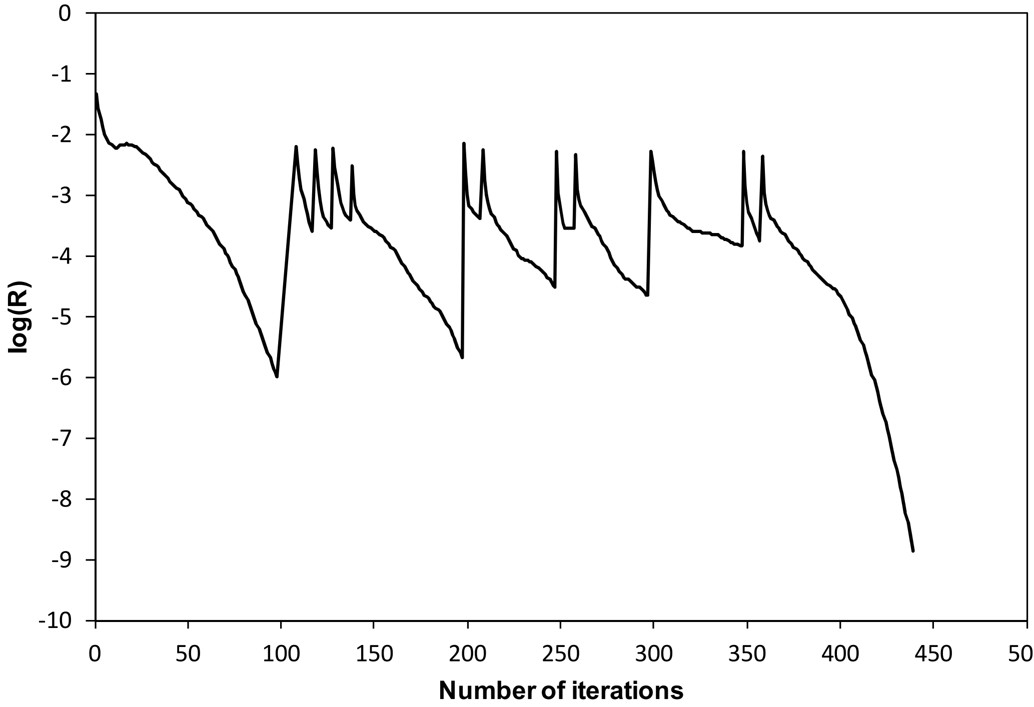

Figure 3-40: Convergence history of the adapted solution for the scramjet inlet problem

CHAPTER 4

CONCLUSION & FUTURE WORK

4.1 Conclusion

In this thesis, least-squares finite element method (LSFEM) is employed to numerically simulate 2-D planar and axisymmetric flows governed by the compressible Euler equations. There are many advantages of LSFEM over traditional FEMs. The governing equations of fluid dynamics are non-self-adjoint equations. In LSFEM, discretization of these equations yield to symmetric positive definite matrices which can be solved efficiently by iterative matrix solvers using less memory and CPU load compared to classical Galerkin FEMs. Another advantage is the robustness of the least-squares formulation; it works for all flow regimes from subsonic to supersonic with a unified formulation without need of special treatments and free parameters to tune.

In order to avoid the use of artificial viscosity for wiggle free shock capturing, h-type adaptive mesh refinement (AMR) is employed. The developed code can work with both quadrilateral and triangular elements of linear and quadratic order. The effect of element type and order on the solution accuracy is inspected in detail. The results indicate that, both element types resolve the flow features correctly and also resolution of the discontinuities increased by using quadratic interpolating functions as expected. Further increase in shock resolution is achieved by employing AMR. It is seen that very satisfactory results can be obtained by starting with a relatively coarse mesh and applying AMR. Upon inspecting the solution of flow over a bump problem, underestimation of the Mach number is seen after the trailing edge of the bump which is due to numerical diffusion. This effect is more apparently seen for the quadrilateral elements and it is lowest for the adaptive solution.

The effect of time step on numerical diffusion is studied with the supersonic ramp problem. It is observed that the use of smaller time step decreased the numerical diffusion and as a result shock waves become thinner, but also oscillatory behavior is seen in the shock regions. By the use of AMR with a very coarse initial mesh, good shock resolution without any oscillations can be obtained. It is showed that AMR refines only the elements at the shock locations and undesired mesh propagation is avoided which also introduced skew elements in the domain. But it is also seen that solution is not affected by these skew elements, which indicates the robustness of the LSFEM with AMR.

4.2 Future Work

In the current study, only mesh refinement was done. But it is planned to implement a mesh coarsening algorithm for performance considerations. It may be more efficient to start with a uniform medium density mesh and perform mesh refinement and coarsening at the same time.

Another fact concerning AMR is that, the h-type refinement is effective especially for the discontinuities in the flow field. But in the smooth flow regions, using p-type mesh refinement may give better results compared to h-type refinement. Therefore, increasing element number and order for the smooth flow regions should be investigated thoroughly in order to determine the need of combining h-type and p-type refinement.

The developed code is capable of solving 2-D planar and axisymmetric problems, and it be extended to 3-D problems. The difficulties of achieving this goal seemed to be related mostly to the modification of AMR.

Another useful study, especially for 3-D problems, will be the parallelization of the element-by-element Jacobi preconditioned Conjugate Gradient solver. This solution technique is known to have a very high parallelization potential and there are many studies in literature that can be consulted [39, 40].

REFERENCES

[1] Hrenikoff, A. "Solution of Problems in Elasticity by the Framework Method." *J. Appl. Mech.*, Vol. 8, pp. 169-175, 1941.

[2] Clough, R. W. "The Finite Element Method in Plane Stress Analysis." *Proceedings of 2nd ASCE Conference on Electronic Computation*, Pittsburgh, PA, September 8–9, 1960.

[3] Zienkiewicz, O. C. *The Finite Element Method.* McGraw-Hill, 1967.

[4] Hughes, T. J. R., Tezduyar, T. E. "Finite element methods for first-order hyperbolic systems with particular emphasis on the compressible Euler equations", *Comput. Methods Appl. Mech. Engrg*, vol. 45, pp 217-284, 1984.

[5] Oden, J. T., Strouboulis, T., Devloo, P. H. "Adaptive Finite Element Methods for high-Speed Compressible Flows." *Int. J. Numer. Meth. Fluids*, vol. 7, pp 1211-1228, 1987.

[6] Loth, E., Baum, J., Löhner, R. "Formation of Shocks Within Axisymmetric Nozzles." *AIAA J.*, vol. 30, no. 1, pp. 268-270, 1991.

[7] Le Beau, G. J., Tezduyar, T. E. "Finite Element Computation of Compressible Flows with SUPG Formulation." *Advances in Finite Element Analysis in Fluid Dynamics*, vol.123, 1991.

[8] Aliabadi, S. K., Tezduyar, T. E. "Space-time finite element computation of compressible flows involving moving boundaries and interfaces." *Comput. Methods Appl. Mech. Engrg*, vol. 107, pp. 209-223, 1993.

[9] Wong, J. S., Darmofal, D. L., Peraire, J. "The solution of the compressible Euler equations at low Mach numbers using a stabilized finite element algorithm." *Comput. Methods Appl. Mech. Engrg*, vol. 190, pp. 5719-5737, 2001.

[10] Wong, J. S., Darmofal, D. L., Peraire, J. "High Order Finite Element Discretization of the Compressible Euler and Navier-Stokes Equations." Aerospace Computational Design Laboratory, Dept. of Aeronautics & Astronautics, Massachusetts Institute of Technology, ACDL Technical Reports;FDRL TR-01-1, 2001.

[11] Bassi, F., Rebay, S. "A High-Order Accurate Discontinuous Finite Element Method for the Numerical Solution of the Compressible Navier-Stokes Equations." *J. Comput. Phys.*, vol.131, pp. 267-279, 1997.

[12] Hartmann, R., Houston, P. "Adaptive Discontinuous Galerkin Finite Element MEthods for the Compressible Euler Equations." *J. Comput. Phys.*, vol. 183, pp. 508-532, 2002.

[13] Tu, S., Aliabadi, S. "A Slope Limiting Procedure in Discontinuous Galerkin Finite Element Method for Gasdynamics Applications." *International Journal of Numerical Analysis and Modeling*, vol. 2, no. 2, pp 163-178, 2005.

[14] Feistauer, M., Kucera, V. "On a robust discontinuous Galerkin technique for the solution of compressible flow." *J. Comput. Phys.*, vol. 224, pp. 208-221, 2007.

[15] Pesch, L., van der Vegt, J. J. W. "A discontinuous Galerkin finite element discretization of the Euler equations for compressible and incompressible fluids." *J. Comput. Phys.*, vol. 227, pp. 5426-5446, 2008.

[16] Barter, G. E. *Shock Capturing with PDE-Based Artificial Viscosity for an Adaptive, Higher-Order Discontinuous Galerkin Finite Element Method.* PhD thesis, Massachusetts Institute of Technology, 2008.

[17] Feistauer, M., Kucera, V., Prokopova, J. "Discontinuous Galerkin solution of compressible flow in time-dependent domains." *Mathematics and Computers in Simulation*, vol. 80, pp. 1612-1623, 2010.

[18] Toulopoulos, I., Ekaterinaris, J. A. "Artificial boundary conditions for the numerical solution of the Euler equations by the discontinuous Galerkin method." *J. Comput. Phys.*, vol.230, pp. 5974-5995, 2011.

[19] Jiang, B. N. *The Least-Squares Finite Element Method*. Springer-Verlag, 1998.

[20] Jiang, B. N. "On the least-squares method." *Comput. Methods Appl. Mech. Engrg.*, vol. 152, pp. 239-257, 1998.

[21] Fletcher, A. J. "A primitive variable finite element formulation for inviscid, compressible flow." *J. Comput. Phys.*, vol. 33, pp. 301-312, 1979.

[22] Jiang, B. N., Carey, F. "Adaptive refinement for least-squares finite elements with element-by-element conjugate gradient solution." *Int. J. Numer. Meth. Engng.*, vol. 24, pp. 569-580, 1987.

[23] Jiang, B. N., Carey, F. "Least-squares finite element methods for compressible Euler equations." *Int. J. Numer. Meth. Fluids*, vol. 10, pp. 557-568, 1990.

[24] Habashi, W. G., Taghaddosi, F., Guevremont, G., Ait-Ali-Yahia. "An adaptive least-squares method for the compressible Euler equations." *Int. J. Numer. Meth. Fluids*, vol. 31, pp. 1121-1139, 1999.

[25] Guaily, A., Megahed, A. "An adaptive finite element method for planar and axisymmetric compressible flows." *Finite Element in Analysis and Design*, vol. 46, pp. 613-624, 2010.

[26] Moussaoui, F. "A unified approach for inviscid compressible and nearly incompressible flow by least-squares finite element method." *Applied Numerical Mathematics*, vol. 44, pp. 183-199, 2003.

[27] Reddy, J. N., Pontaza, J. P., Diao, X., Surana, K. S. "Least-squares finite element models of two-dimensional compressible flows." *Finite Element in Analysis and Design*, vol. 40, pp. 629-644, 2004.

[28] Gerritsma, M., Bas, R., Maerschalck, B., Koren, B., Deconinck, H. "Least-squares spectral element method applied to the Euler equations." *Int. J. Numer. Meth. Fluids*, vol. 57, pp. 1371-1395, 2008.

[29] Rivara, M. C. "Algorithms for refining triangular grids for adaptive and multigrid techniques." *Int. J. Numer. Meth. Engng.*, vol. 20, pp. 745-756, 1984.

[30] Suarez, J. P., Plaza, A., Carey, G. F. "The propagation problem in longest-edge refinement." *Finite Element in Analysis and Design*, vol. 42, pp. 130-151, 2005.

[31] Reddy, J. N. *An Introduction to the Finite Element Method.* 3rd ed., McGraw-Hill, 2006.

[32] Cummings, R. M., Yang, H. T., Oh, Y. H. "Supersonic, turbulent flow computation and drag optimization for axisymmetric afterbodies." *Comput. Fluids*, vol. 24, pp. 487-507, 1995

[33] Eidelman, S., Colella, P., Shreeve, R. P. "Application of the Godunov method and its second-order extension to cascade flow modeling." *AIAA J.*, vol. 22, pp. 1609–1615, 1984.

[34] Ni, R. H. "A multiple grid scheme for solving the Euler equations." *AIAA J.*, vol. 20, pp. 1565-1571, 1982.

[35] Tannehill, J. C., Anderson, D. A., Pletcher, R. H. *Computational fluid mechanics and heat transfer.* Taylor&Francis, 1997.

[36] Schiff, L. B., Sturek, W. B. "Numerical simulation of steady supersonic flow over an ogive cylinder boattail body." U.S. Army Ballistic Research Laboratory, Report 2363, 1981.

[37] Hosseini, R., Rahimian, M. H., Mirzaei, M. "Performance of High-Accuracy Schemes in Inviscid Fluxes Calculation." *The Eleventh Annual Conference of the CFD Society Canada*, Vancouver, BC, pp. 552-564, 2003.

[38] *ANSYS FLUENT Theory Guide.* 2009.

[39] Duarte, C. A. "Object Oriented Parallel Implementation of the Element by Element Conjugate Gradient Method." TICAM - Texas Inst. for Comput. and Appl. Mathematics, Austin Texas, 2009.

[40] Carey, G. F., Barragy, E., Sharma, M. "Element-by-Element Vector and Parallel Computations." *Communications In Applied Numerical Methods*, vol. 4, pp. 299-307, 1988.