An H -adaptive finite element compressible flow solver applied to light -gas gun design

Timothy Todd de Bues
University of Nevada, Las Vegas

AN H-ADAPTIVE FINITE ELEMENT

COMPRESSIBLE FLOW SOLVER

APPLIED TO LIGHT-GAS

GUN DESIGN

by

Timothy Todd de Bues

Bachelor of Science, Aerospace Engineering
University of Southern California
1991

Master of Science, Aerospace Engineering
California State University, Long Beach
1994

A dissertation submitted in partial fulfillment
of the requirements for the

Doctor of Philosophy Degree in Mechanical Engineering
Department of Mechanical Engineering
Howard R. Hughes College of Engineering

Graduate College
University of Nevada, Las Vegas
May 2003

UMI®

UNLV

Dissertation Approval

The Graduate College

University of Nevada, Las Vegas

_____April 15_____, 20_03_

The Dissertation prepared by

Timothy Todd deBues
_____

Entitled

An h-Adaptive Finite Element Compressible Flow Solver Applied to _____

Light Gas Gun Design _____

_____

is approved in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Mechanical Engineering _____

_____
Examination Committee Chair

_____
Examination Committee Member

_____
Examination Committee Member

_____
Examination Committee Member

_____
Graduate College Faculty Representative

_____
Dean of the Graduate College

1017-52

ii

ABSTRACT

**An H-Adaptive Finite Element
Compressible Flow Solver
Applied to Light-Gas
Gun Design**

by

Timothy Todd de Bues

Dr. Darrell W. Pepper, Examination Committee Chair
Director, Nevada Center for Advanced Computational Methods
Professor of Mechanical Engineering
University of Nevada, Las Vegas

The Joint Actinide Shock Physics Experimental Research (JASPER) facility utilizes a

two-stage light gas gun to conduct equation of state experiments. The gun has a launch

tube bore diameter of 28 mm, and is capable of launching projectiles at a velocity of 7.4

km/s using compressed hydrogen as a propellant. A numerical study is conducted to

determine what effects, if any, launch tube exit geometry changes have on attitude of the

projectile in flight. A comparison of two launch tube exit geometries is considered. The

first case is standard muzzle geometry where the wall of the bore and the outer surface of

the launch tube form a 90 degree angle. The second case includes a 26.6 degree bevel

transition from the wall of the bore to the outer surface of the launch tube. The finite

element method is employed to model the Euler equations and the compressible Navier-

Stokes equations. The numerical method incorporates the use of trilinear, hexahedral,

isoparametric elements, as well as the use of Petrov-Galerkin weighting applied to the

advection terms. Mass lumping allows an explicit Euler scheme to be used in conjunction with a second-order Runge-Kutta approximation to advance the discretized equations in time. An h-adaptive mesh refinement scheme based on elemental flow feature gradients is utilized for greater solution accuracy. For both cases, solutions are calculated for several positions downstream of the launch tube exit. Numerical solutions obtained indicate that both cases will have an adverse effect on flight attitude of the projectile, with the beveled muzzle geometry performing worse than the standard case.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

I wish to express my appreciation to all the people who have supported me in this endeavor. First, I would like to thank my advisor, Dr. Darrell Pepper for his sage advise, encouragement, and friendship. I would also like to thank Dr. Robert Boehm, Dr. William Culbreth, Dr. Samir Moujaes, and Dr. George Miel for taking the time to serve on my committee. I would also like to thank Dr. Yitung Chen and all the people at the Nevada Center for Advanced Computational Methods for their assistance and encouragement. Finally, I wish to thank my family for their love, patience and support, without which none of this would be possible.

CHAPTER 1

INTRODUCTION

This dissertation deals with the development of a three-dimensional finite element compressible flow model, and how it applies to the muzzle design of a two-stage light gas gun. A basic road map of this work is presented in this chapter.

Chapter 2 focuses on the governing equations for compressible flow. Therein, the dimensionless forms of the governing equations are developed.

Many numerical techniques for modeling systems of partial differential equations are in use today. Chapter 3 centers on the finite element formulation of the governing equations. Finite element methods are commonly used today for the solution of compressible flow problems. Since the solution algorithm is decoupled from the process of mesh generation, finite element methods can utilize unstructured meshes allowing any arbitrarily shaped region to be discretized. Also, it enables local refinement of the mesh to occur independently of the solution algorithm.

The primary disadvantage of attacking complex problems in three dimensions is that the number of elements required to resolve certain flow phenomena increases substantially. Finer mesh density is especially important in accurately capturing various flow features, such as the precise locations of shocks. Rather than using a finer mesh throughout the entire solution domain, mesh adaptation is employed. Further discussion

1

on the mesh adaptation methods and strategies concerning this work will be handled in Chapter 4.

Results of several benchmark test cases will be presented in Chapter 5. Benchmarking is an important part of numerical model development. If results of well documented experimental data or theoretical data can be duplicated with a numerical model, then greater trust can be given to that model's results as it is applied to new problems.

In Chapter 6, results for the flow field around the muzzle of a light gas gun are presented. Two different muzzle configurations are considered. The first case is standard muzzle geometry where the wall of the bore and the outer surface of the launch tube form a $90°$ angle. The second case includes a $26.6°$ bevel transition from the wall of the bore to the outer surface of the launch tube. Due to the extreme nature of the problem, numerical simulation is the only feasible way of examining what effects, if any, launch tube exit geometry changes have on the attitude of the projectile in flight.

Finally, in Chapter 7, conclusions will be drawn based on the results obtained, and recommendations will be made regarding future research.

CHAPTER 2

GOVERNING EQUATIONS FOR COMPRESSIBLE FLOW

The model developed in this study is governed by the compressible Navier-Stokes Equations. Neglecting body forces, the governing equations may be written in nonconservation form as follows:

continuity:

$$\frac{D\rho}{Dt} + \rho\nabla\cdot\mathbf{u} = 0 \qquad (2.1)$$

momentum:

$$\rho\frac{D\mathbf{u}}{Dt} = \nabla\cdot\boldsymbol{\sigma} \qquad (2.2)$$

internal energy:

$$\rho\frac{De}{Dt} = -\nabla\cdot\mathbf{q} + \sigma_{ij}\frac{\partial u_i}{\partial x_j} \qquad (2.3)$$

Brueckner (1991) noted that when compared to the total energy formulation, the internal energy formulation for compressible flow results in a more stable algorithm. For this reason, the internal energy formulation is used in this study.

In equations (2.1) through (2.3), the convective derivative is given by

$$\frac{D}{Dt} = \frac{d}{dt} + \mathbf{u}\cdot\nabla \qquad (2.4)$$

3

M is the mach number, γ is the ratio of specific heats, and Fourier's Law gives the required relation between temperature and heat flux as:

$$\mathbf{q} = -k\nabla T \tag{2.5}$$

The terms of the Stokes stress tensor $\boldsymbol{\sigma}$ are given by:

$$\sigma_{ij} = \left(-p + \lambda\nabla\cdot\mathbf{u}\right)\delta_{ij} + D_{ij} \tag{2.6}$$

where

$$D_{ij} = \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right). \tag{2.7}$$

According to Stokes' hypothesis

$$\lambda = -\frac{2}{3}\mu. \tag{2.8}$$

The viscosity, μ, is temperature dependent and is determined using Sutherland's formula (Schlichting, 1979):

$$\frac{\mu}{\mu_\infty} = \left(\frac{T}{T_\infty}\right)^{3/2}\frac{T_\infty + S}{T + S}, \tag{2.9}$$

where μ_∞ is the viscosity of the fluid at the reference temperature T_∞, and the Sutherland constant, $S = 110.4\,\mathrm{K}$.

In order to close the above system of equations, a relation for pressure is needed. The equation of state for an ideal gas is used

$$p = \rho RT. \tag{2.10}$$

In addition, a calorically perfect gas is assumed, giving the relation between internal energy and temperature as:

$$e = c_v T \qquad (2.11)$$

where c_v is the specific heat at constant volume.

To highlight the important similarity parameters for viscous, compressible flow, a nondimensional form of the compressible Navier-Stokes Equations is employed. In order to obtain this nondimensional form, the following dimensionless variables are introduced (Anderson, 2000):

$$\mathbf{u}^* = \frac{\mathbf{u}}{U_\infty} \quad , \quad \mathbf{x}^* = \frac{\mathbf{x}}{L} \quad , \quad t^* = \frac{t U_\infty}{L}$$

$$\rho^* = \frac{\rho}{\rho_\infty} \quad , \quad T^* = \frac{T}{T_\infty} \quad , \quad p^* = \frac{p}{p_\infty} \qquad (2.12)$$

$$e^* = \frac{e}{c_v T_\infty} \quad , \quad k^* = \frac{k}{k_\infty} \quad , \quad \mu^* = \frac{\mu}{\mu_\infty}$$

where L, U_∞, ρ_∞, T_∞, p_∞, e_∞, k_∞, and μ_∞ are free stream reference values.

After inserting equations (2.12) into equations (2.1), (2.2), (2.3), (2.10), and (2.11) the following forms of the governing equations are obtained (dropping the * notation for convenience):

continuity:

$$\frac{D\rho}{Dt} + \rho \nabla \cdot \mathbf{u} = 0 \qquad (2.13)$$

momentum:

$$\rho \frac{D\mathbf{u}}{Dt} = \nabla \left[-\frac{1}{\gamma M^2} p + \frac{\gamma}{\mathrm{Re}} (\nabla \cdot \mathbf{u}) \right] + \frac{1}{\mathrm{Re}} (\nabla \cdot \mathbf{D}) \qquad (2.14)$$

internal energy:

$$\rho \frac{De}{Dt} = \frac{\gamma}{\mathrm{Re}\,\mathrm{Pr}} \boldsymbol{\nabla} \cdot (\mu \nabla e) +$$

$$\gamma(\gamma - 1) M_\infty^2 \left\{ -\left[\frac{1}{\gamma M^2} p + \frac{2}{3} \frac{\mu}{\mathrm{Re}} \boldsymbol{\nabla} \cdot \mathbf{u} \right] \boldsymbol{\nabla} \cdot \mathbf{u} + \frac{1}{\mathrm{Re}} D_{ij} \frac{\partial u_i}{\partial x_j} \right\} \tag{2.15}$$

The dimensionless relation between internal energy and temperature becomes

$$e = T. \tag{2.16}$$

Using equation (2.16) the dimensionless equation of state can be written as:

$$p = \rho e \tag{2.17}$$

The dimensionless similarity parameters appearing in equations (2.14) and (2.15) are now evident:

Mach number:

$$M = \frac{U_\infty}{\sqrt{\gamma\, p_\infty / \rho_\infty}} \tag{2.18}$$

Reynolds number:

$$\mathrm{Re} = \frac{\rho_\infty U_\infty L}{\mu_\infty} \tag{2.19}$$

Prandtl number:

$$\mathrm{Pr} = \frac{\mu_\infty c_p}{k_\infty} \tag{2.20}$$

Ratio of specific heats:

$$\gamma = \frac{c_p}{c_v} \tag{2.21}$$

The Sutherland viscosity formula must also be cast in terms of the dimensionless variables. Using dimensionless variables of equation (2.12), equation (2.9) can be rewritten as:

$$\mu = T^{3/2} \frac{1+\overline{S}}{T+\overline{S}}$$
(2.22)

where

$$\overline{S} = \frac{S}{T_\infty}$$
(2.23)

A closed set of dimensionless equations now exists. In the next chapter, a suitable finite element model based on these governing equations will be created.

CHAPTER 3

FINITE ELEMENT FORMULATION

Many numerical techniques for modeling systems of partial differential equations are in use today. For compressible flow problems, finite difference and finite element methods are frequently used. One of the main advantages of finite difference methods is that the algorithms are fairly easy to encode. Since finite difference methods have been in use longer than any other method, a myriad of solution schemes already exists for many types of specific problems. The major drawback of finite difference methods is that the computational domain is somewhat limited to structured meshes.

On the other hand, finite element methods can utilize unstructured meshes, since the solution algorithm and the process of mesh generation are uncoupled. This decoupling permits any arbitrarily shaped region to be discretized (Lohner, et al., 1986). Also it enables local refinement of the mesh to occur independently of the solution algorithm. For these reasons, the finite element method has been chosen to model the governing equations.

Method of Weighted Residuals

In order to examine the method of weighted residuals, it is necessary to define the following linear function spaces:

8

$$H^1(\Omega) = \left\{ f(\mathbf{x}) \,\left|\, \int_\Omega \left[f^2 + (\nabla f)^2 \right] d\Omega < \infty \right. \right\} \tag{3.1}$$

$$H_0^1(\Omega) = \left\{ f \,\left|\, f \in H^1(\Omega) \text{ and } f(\mathbf{x}) = 0 \text{ on } \Gamma_1 \right. \right\} \tag{3.2}$$

$$\bar{H}^1(\Omega) = \left\{ f \,\left|\, f \in H^1(\Omega) \text{ and } f(\mathbf{x}) = g_1 \text{ on } \Gamma_1 \right. \right\} \tag{3.3}$$

Consider the case of linear one-dimensional heat conduction given by the following governing equation:

$$-\frac{d}{dx}\left(k \frac{d\phi}{dx} \right) = f \qquad \Omega = \left\{ x \mid a < x < b \right\} \tag{3.4}$$

For simplicity, this equation can be interpreted as a linear operator, L, acting on functions $u(\mathbf{x})$, over a domain Ω, bounded by Γ. Equation (3.4) can be rewritten as:

$$Lu = f \qquad \mathbf{x} \in \Omega \tag{3.5}$$

with boundary conditions:

$$u = g_1(x) \qquad \mathbf{x} \in \Gamma_1 \tag{3.6}$$

$$\mathbf{n} \cdot (k \cdot \nabla u) = g_2(x) \qquad \mathbf{x} \in \Gamma_2 \tag{3.7}$$

where g_1 and g_2 are given functions, $\mathbf{n}$ is the outward unit normal to Γ, and $\Gamma_1 \cup \Gamma_2 = \Gamma$.

The residual function based on equation (3.5) can be defined as:

$$R(u, \mathbf{x}) \equiv Lu(\mathbf{x}) - f(x) \tag{3.8}$$

We seek to minimize the residual $R(u, \mathbf{x})$ over the computational domain. The residual measures the error in the satisfaction of the solution. In other words, if $\hat{u}$ is the exact solution of equation (3.5), then

$$R(\hat{u}, \mathbf{x}) = 0 \tag{3.9}$$

Defining a weighting function $w = w(\mathbf{x}) \in H_0^1$, the weighted residual form of equation (3.8) is obtained:

$$\int_\Omega w(\mathbf{x}) R(u, \mathbf{x}) d\Omega = 0 \tag{3.10}$$

or

$$\int_\Omega w \left\{ -\frac{d}{dx}\left(k\frac{d\phi}{dx}\right) - f \right\} dx = 0 \tag{3.11}$$

To weaken the second derivative term, Green's Theorem is applied. The problem may now be stated as: Find a function $\phi(\mathbf{x}) \in \bar{H}^1(\Omega)$ such that

$$\int_\Omega \left(k\frac{dw}{dx}\frac{d\phi}{dx} - wf \right) d\Omega - \left[w\left(-k\frac{d\phi}{dx}\right) \right]_{x=a}^{x=b} = 0 \qquad \forall \, w \in H_0^1(\Omega) \tag{3.12}$$

The functions ϕ are called the trial functions and the functions w are called the test or weighting functions. The particular weighted residual method sought depends on the choice of the weighting functions.

Galerkin Finite Element Method

The main difficulty with the solution of equation (3.12) is that the Sobolev spaces $H^1(\Omega)$ and $H_0^1(\Omega)$ are infinite-dimensional (Heinrich and Pepper, 1999). The Galerkin method remedies this by doing two things. First, define the following n-dimensional subspaces:

$$H^1(n, \Omega) \subset H^1(\Omega) \tag{3.13}$$

$$H_0^1(n,\Omega) \subset H_0^1(\Omega) \tag{3.14}$$

Second, choose a set of shape functions $N_i(\mathbf{x})$ in $H_0^1(\Omega)$ that forms the basis for $H^1(n,\Omega)$:

$$\phi(\mathbf{x}) \approx \phi^h(\mathbf{x}) = \sum_{i=1}^{n} N_i(\mathbf{x})c_i \tag{3.15}$$

and set the weighting functions, w_i, equal to $N_i(\mathbf{x})$. When applied to equation (3.12), this results in n algebraic equations for the n coefficients c_i.

Transformation to Natural Coordinates

The finite element method involves the discretization of the solution domain into a finite number of elements. Each element is bounded by a number of nodes. The elements may not overlap, but may share nodes with adjacent elements. The wide variety of element shapes across the solution domain makes determination of the shape functions and numerical integration tedious. To simplify these processes, a transformation from the Cartesian coordinate system to the so called "natural" coordinate system is employed (see Figure 3-1). In the natural coordinate system, all physical dimensions (ξ, η, ζ) of the isoparametric elements lie within the range of -1 to 1.

(X8,Y8,Z8)
(X5,Y5,Z5)
(X7,Y7,Z7)
(X1,Y1,Z1)
(X3,Y3,Z3)
(X2,Y2,Z2)

(-1,-1,1)
(1,-1,1)
(-1,1,1)
(-1,-1,-1)
(1,1,1)
(-1,1,-1)
(1,-1,-1)
(1,1,-1)
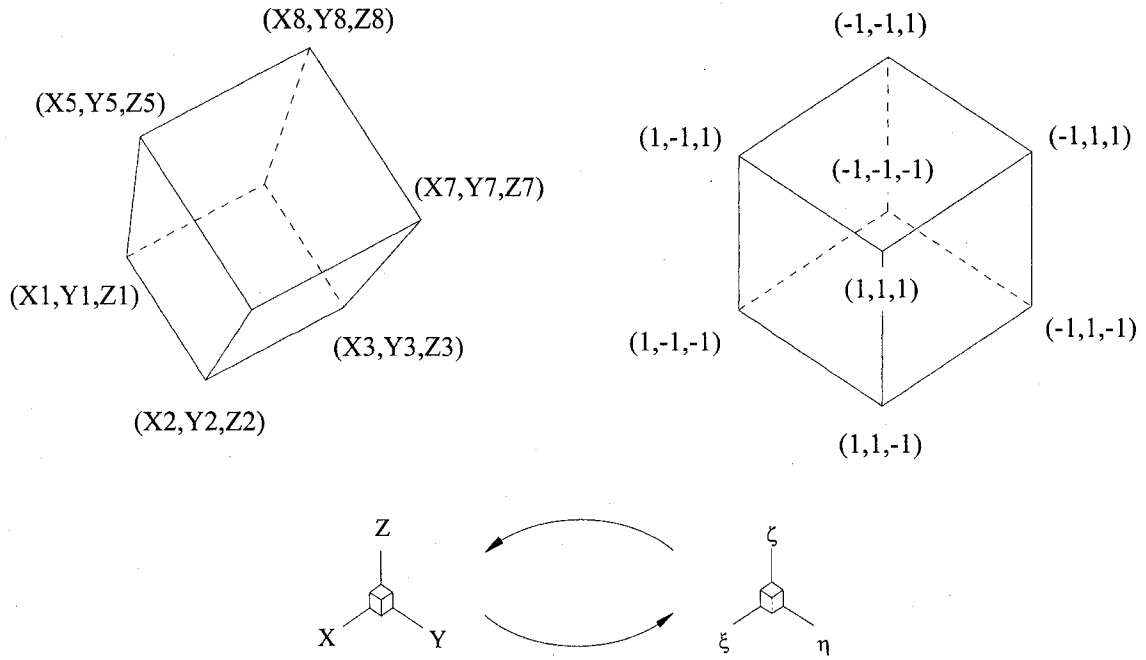
Figure 3-1. Coordinate transformation to natural coordinates

In the natural coordinate system, the shape functions can easily be written as (Pepper & Heinrich, 1992):

$$\begin{bmatrix} N_1 \\ N_2 \\ N_3 \\ N_4 \\ N_5 \\ N_6 \\ N_7 \\ N_8 \end{bmatrix} = \frac{1}{8} \begin{bmatrix} (1-\xi) & (1-\eta) & (1-\zeta) \\ (1+\xi) & (1-\eta) & (1-\zeta) \\ (1+\xi) & (1+\eta) & (1-\zeta) \\ (1-\xi) & (1+\eta) & (1-\zeta) \\ (1-\xi) & (1-\eta) & (1+\zeta) \\ (1+\xi) & (1-\eta) & (1+\zeta) \\ (1+\xi) & (1+\eta) & (1+\zeta) \\ (1-\xi) & (1+\eta) & (1+\zeta) \end{bmatrix}$$

(3.16)

The derivatives of the shape functions are obtained using the chain rule (Beer and Watson, 1992; Fletcher, 1984):

$$\frac{\partial N_i}{\partial \xi} = \frac{\partial N_i}{\partial x}\frac{\partial x}{\partial \xi} + \frac{\partial N_i}{\partial y}\frac{\partial y}{\partial \xi} + \frac{\partial N_i}{\partial z}\frac{\partial z}{\partial \xi} \tag{3.17}$$

$$\frac{\partial N_i}{\partial \eta} = \frac{\partial N_i}{\partial x}\frac{\partial x}{\partial \eta} + \frac{\partial N_i}{\partial y}\frac{\partial y}{\partial \eta} + \frac{\partial N_i}{\partial z}\frac{\partial z}{\partial \eta} \tag{3.18}$$

$$\frac{\partial N_i}{\partial \zeta} = \frac{\partial N_i}{\partial x}\frac{\partial x}{\partial \zeta} + \frac{\partial N_i}{\partial y}\frac{\partial y}{\partial \zeta} + \frac{\partial N_i}{\partial z}\frac{\partial z}{\partial \zeta} \tag{3.19}$$

Equations (3.17) through (3.19) can be rewritten in matrix form as:

$$\begin{bmatrix} \dfrac{\partial N_i}{\partial \xi} \\[2mm] \dfrac{\partial N_i}{\partial \eta} \\[2mm] \dfrac{\partial N_i}{\partial \zeta} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial x}{\partial \xi} & \dfrac{\partial y}{\partial \xi} & \dfrac{\partial z}{\partial \xi} \\[2mm] \dfrac{\partial x}{\partial \eta} & \dfrac{\partial y}{\partial \eta} & \dfrac{\partial z}{\partial \eta} \\[2mm] \dfrac{\partial x}{\partial \zeta} & \dfrac{\partial y}{\partial \zeta} & \dfrac{\partial z}{\partial \zeta} \end{bmatrix} \begin{bmatrix} \dfrac{\partial N_i}{\partial x} \\[2mm] \dfrac{\partial N_i}{\partial y} \\[2mm] \dfrac{\partial N_i}{\partial z} \end{bmatrix} \tag{3.20}$$

or

$$\begin{bmatrix} \dfrac{\partial N_i}{\partial \xi} \\[2mm] \dfrac{\partial N_i}{\partial \eta} \\[2mm] \dfrac{\partial N_i}{\partial \zeta} \end{bmatrix} = \mathbf{J} \begin{bmatrix} \dfrac{\partial N_i}{\partial x} \\[2mm] \dfrac{\partial N_i}{\partial y} \\[2mm] \dfrac{\partial N_i}{\partial z} \end{bmatrix} \tag{3.21}$$

where $\mathbf{J}$ is the Jacobian matrix. Similarly, the derivatives of any computational variable can be transformed into the natural coordinate system via the relation:

$$\begin{bmatrix} \dfrac{\partial}{\partial \xi} \\[2mm] \dfrac{\partial}{\partial \eta} \\[2mm] \dfrac{\partial}{\partial \zeta} \end{bmatrix} = \mathbf{J} \begin{bmatrix} \dfrac{\partial}{\partial x} \\[2mm] \dfrac{\partial}{\partial y} \\[2mm] \dfrac{\partial}{\partial z} \end{bmatrix} \tag{3.22}$$

When dealing with isoparametric elements, it is sometimes convenient to express the Jacobian in terms of the shape functions defined in equation (3.16) as follows:

$$\mathbf{J} = \begin{bmatrix} \sum \dfrac{\partial N_i}{\partial \xi} x_i & \sum \dfrac{\partial N_i}{\partial \xi} y_i & \sum \dfrac{\partial N_i}{\partial \xi} z_i \\[2mm] \sum \dfrac{\partial N_i}{\partial \eta} x_i & \sum \dfrac{\partial N_i}{\partial \eta} y_i & \sum \dfrac{\partial N_i}{\partial \eta} z_i \\[2mm] \sum \dfrac{\partial N_i}{\partial \zeta} x_i & \sum \dfrac{\partial N_i}{\partial \zeta} y_i & \sum \dfrac{\partial N_i}{\partial \zeta} z_i \end{bmatrix} \tag{3.23}$$

In the transfer back to Cartesian coordinates, the inverse of equation (3.22) is used.

$$\begin{bmatrix} \dfrac{\partial}{\partial x} \\[2mm] \dfrac{\partial}{\partial y} \\[2mm] \dfrac{\partial}{\partial z} \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} \dfrac{\partial}{\partial \xi} \\[2mm] \dfrac{\partial}{\partial \eta} \\[2mm] \dfrac{\partial}{\partial \zeta} \end{bmatrix} \tag{3.24}$$

In equation (3.24) the inverse of the Jacobian (Fletcher, 1997) is:

$$\mathbf{J}^{-1} = \begin{bmatrix} \dfrac{\partial x}{\partial \xi} & \dfrac{\partial x}{\partial \eta} & \dfrac{\partial x}{\partial \zeta} \\[2mm] \dfrac{\partial y}{\partial \xi} & \dfrac{\partial y}{\partial \eta} & \dfrac{\partial y}{\partial \zeta} \\[2mm] \dfrac{\partial z}{\partial \xi} & \dfrac{\partial z}{\partial \eta} & \dfrac{\partial z}{\partial \zeta} \end{bmatrix} \tag{3.25}$$

Numerical Integration

Although many numerical methods exist for evaluating integrals, Gauss-Legendre quadrature is a very appropriate algorithm to implement. Recall that the finite element method employed for this study makes use of the transformation to a normalized computational domain where all physical dimensions (ξ, η, ζ) lie within the range of

-1 to 1. Since the limits of integration used for Gauss-Legendre quadrature also range from -1 to 1, no change in limits is required.

Integrals in this study take the form of

$$\int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} f(\xi,\eta,\zeta) \; |\mathbf{J}| \; d\xi d\eta d\zeta \tag{3.26}$$

Using Gauss-Legendre quadrature to evaluate this integral yields (Pepper and Heinrich, 1992):

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n} w_i w_j w_k \, f\left(\xi_i,\eta_j,\zeta_k\right) \; |\mathbf{J}|\left(\xi_i,\eta_j,\zeta_k\right) \tag{3.27}$$

where w_i, w_j, and w_k are the Gauss weights of integration evaluated at the Gauss points ξ_i, η_j, and ζ_k. For the trilinear hexahedral elements used in this study, $2 \times 2 \times 2$ quadrature is used. Therefore, since $n = 2$ the following values are used:

$$w_i = w_j = w_k = 1.0 \tag{3.28}$$

$$\xi_i = \eta_j = \zeta_k = \pm\frac{1}{\sqrt{3}} \tag{3.29}$$

Finite Element Formulation for Compressible Flow

Repeated here for convenience, the dimensionless governing equations are:

continuity:

$$\frac{D\rho}{Dt} + \rho\nabla\cdot\mathbf{u} = 0 \tag{3.30}$$

momentum:

$$\rho\frac{D\mathbf{u}}{Dt} = \nabla\left[-\frac{1}{\gamma M^2}p+\frac{\gamma}{Re}(\nabla\cdot\mathbf{u})\right] + \frac{1}{Re}(\nabla\cdot\mathbf{D}) \tag{3.31}$$

internal energy:

$$\rho \frac{De}{Dt} = \frac{\gamma}{\mathrm{Re\,Pr}} \nabla \cdot (\mu \nabla e) +$$

$$\gamma(\gamma - 1) M_\infty^2 \left\{ -\left[\frac{1}{\gamma M^2} p + \frac{2}{3} \frac{\mu}{\mathrm{Re}} \nabla \cdot \mathbf{u} \right] \nabla \cdot \mathbf{u} + \frac{1}{\mathrm{Re}} D_{ij} \frac{\partial u_i}{\partial x_j} \right\} \tag{3.32}$$

equation of state:

$$p = \rho e \tag{3.33}$$

Since the equation of state gives the pressure strictly in terms of density and internal energy, no further manipulation of this equation is necessary.

Multiplying the continuity, momentum, and energy equations by the weighting functions and integrating over the volume Ω with boundary surface Γ yields:

continuity:

$$\int_\Omega W_i \left\{ \frac{\partial \rho}{\partial t} + (\mathbf{u} \cdot \nabla) \rho \right\} d\Omega = -\int_\Omega W_i (\rho \nabla \cdot \mathbf{u}) \, d\Omega \tag{3.34}$$

momentum:

$$\int_\Omega W_i \left\{ \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right\} d\Omega = \int_\Omega W_i \nabla \left[-\frac{1}{\gamma M^2} p \right] d\Omega$$

$$+ \int_\Omega W_i \nabla \left[\frac{\gamma}{\mathrm{Re}} (\nabla \cdot \mathbf{u}) \right] d\Omega \tag{3.35}$$

$$+ \int_\Omega W_i \frac{1}{\mathrm{Re}} (\nabla \cdot \mathbf{D}) \, d\Omega$$

internal energy:

$$\int_\Omega W_i\, \rho\left\{\frac{\partial e}{\partial t}+(\mathbf{u}\cdot\nabla)e\right\}\, d\Omega \;=\; \int_\Omega W_i\left\{\frac{\gamma}{\mathrm{Re}\,\mathrm{Pr}}\nabla\cdot(\mu\nabla e)\right\}\, d\Omega$$

$$-\int_\Omega W_i\gamma(\gamma-1)M_\infty^2\left\{\left[\frac{1}{\gamma M^2}p+\frac{2}{3}\frac{\mu}{\mathrm{Re}}\nabla\cdot\mathbf{u}\right]\nabla\cdot\mathbf{u}\right\}\, d\Omega \qquad (3.36)$$

$$+\int_\Omega W_i\gamma(\gamma-1)M_\infty^2\left\{\frac{1}{\mathrm{Re}}D_{ij}\frac{\partial u_i}{\partial x_j}\right\}\, d\Omega$$

Weak Formulation

Note that equations (3.35) and (3.36) contain second order derivatives. Due to the difficult nature of numerically modeling second order derivative terms, the momentum and energy equations are weakened by applying the Green-Gauss theorem.

Suppose φ is a scalar field represented by $\varphi = \varphi(x,y,z)$, and $\mathbf{q}$ is an arbitrary vector given by

$$\mathbf{q} = \begin{bmatrix} q_x \\ q_y \\ q_z \end{bmatrix}. \qquad (3.37)$$

The Green-Gauss theorem can be written as:

$$\int_V \varphi(\nabla\cdot\mathbf{q})\, dV \;=\; \int_S \varphi\, \mathbf{q}^{\mathrm{T}}\mathbf{n}\, dS \;-\; \int_V (\nabla\varphi)^{\mathrm{T}}\mathbf{q}\, dV \qquad (3.38)$$

where $\mathbf{n}$ is the unit vector normal to and directed outward from the boundary of volume V. Equation (3.38) takes the volume integral of the divergence of vector $\mathbf{q}$ multiplied by φ and relates it to a surface integral. In one dimension this is known simply as

integrating by parts. This process has the desired effect of eliminating second order derivative terms.

Applying equation (3.38) to the terms in equations (3.35) and (3.36) containing second order derivatives, and recognizing that the weight functions in those terms are analogous to φ in equation (3.38), the governing equations may now be rewritten in their weakened form.

continuity:

$$\int_\Omega W_i \left\{ \frac{\partial \rho}{\partial t} + (\mathbf{u} \cdot \nabla)\rho \right\} d\Omega = - \int_\Omega W_i \left(\rho \nabla \cdot \mathbf{u} \right) d\Omega \qquad (3.39)$$

momentum:

$$\int_\Omega \rho W_i \left\{ \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} \right\} d\Omega = - \int_\Omega W_i \nabla \left[\frac{1}{\gamma M^2} p \right] d\Omega$$

$$- \int_\Omega (\nabla W_i)^{\mathrm{T}} \left[\frac{\lambda}{\mathrm{Re}} (\nabla \cdot \mathbf{u}) \right] d\Omega - \int_\Omega \frac{1}{\mathrm{Re}} (\nabla W_i)^{\mathrm{T}} \cdot \mathbf{D}\, d\Omega \qquad (3.40)$$

$$+ \int_\Gamma W_i \left[\frac{\lambda}{\mathrm{Re}} (\nabla \cdot \mathbf{u}) \right]^{\mathrm{T}} \cdot \mathbf{n}\, d\Gamma + \int_\Gamma W_i \frac{1}{\mathrm{Re}} \left(\mathbf{D}^{\mathrm{T}} \cdot \mathbf{n} \right) d\Gamma$$

internal energy:

$$\int_\Omega W_i\, \rho \left\{ \frac{\partial e}{\partial t} + (\mathbf{u} \cdot \nabla)e \right\} d\Omega = - \int_\Omega \frac{\gamma}{\mathrm{Re}\,\mathrm{Pr}} \mu\, \nabla W_i\, \nabla e\, d\Omega -$$

$$- \int_\Omega W_i \gamma (\gamma - 1) M_\infty^2 \left\{ \left[\frac{1}{\gamma M^2} p + \frac{2}{3} \frac{\mu}{\mathrm{Re}} \nabla \cdot \mathbf{u} \right] \nabla \cdot \mathbf{u} \right\} d\Omega \qquad (3.41)$$

$$+ \int_\Omega W_i \gamma (\gamma - 1) M_\infty^2 \left\{ D_{ij} \frac{\partial u_i}{\partial x_j} \right\} d\Omega + \int_\Gamma \frac{\gamma}{\mathrm{Re}\,\mathrm{Pr}} \mu\, W_i (\mathbf{n} \cdot \nabla e)\, d\Gamma$$

In three dimensions, the weakened governing equations may be written as:

continuity:

$$\int_\Omega W_i \frac{\partial \rho}{\partial t}\, d\Omega = -\int_\Omega W_i \left[u\frac{\partial \rho}{\partial x} + v\frac{\partial \rho}{\partial y} + w\frac{\partial \rho}{\partial z} \right] d\Omega$$

$$-\int_\Omega W_i \rho \left[\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right] d\Omega$$

(3.42)

x-momentum:

$$\int_\Omega W_i \rho \frac{\partial u}{\partial t}\, d\Omega = -\int_\Omega W_i \rho \left[u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} \right] d\Omega$$

$$-\int_\Omega W_i \left[\frac{1}{\gamma M^2}\frac{\partial p}{\partial x} \right] d\Omega$$

$$-\int_\Omega \frac{\mu}{\mathrm{Re}} \left\{ \frac{\partial W_i}{\partial x}\left[\frac{4}{3}\frac{\partial u}{\partial x} - \frac{2}{3}\frac{\partial v}{\partial y} - \frac{2}{3}\frac{\partial w}{\partial z} \right] \right.$$

$$\left. + \frac{\partial W_i}{\partial y}\left[\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right] + \frac{\partial W_i}{\partial z}\left[\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right] \right\} d\Omega$$

$$+ \int_\Gamma W_i \frac{\mu}{\mathrm{Re}} \left\{ \left[\frac{4}{3}\frac{\partial u}{\partial x} - \frac{2}{3}\frac{\partial v}{\partial y} - \frac{2}{3}\frac{\partial w}{\partial z} \right]\mathrm{n}_x \right.$$

$$\left. + \left[\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right]\mathrm{n}_y + \left[\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right]\mathrm{n}_z \right\} d\Gamma$$

(3.43)

y-momentum:

$$\int_\Omega W_i \rho \frac{\partial v}{\partial t} d\Omega = -\int_\Omega W_i \rho \left[u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z} \right] d\Omega$$

$$- \int_\Omega W_i \left[\frac{1}{\gamma M^2}\frac{\partial p}{\partial y} \right] d\Omega$$

$$- \int_\Omega \frac{\mu}{Re} \left\{ \frac{\partial W_i}{\partial x}\left[\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right] + \frac{\partial W_i}{\partial y}\left[-\frac{2}{3}\frac{\partial u}{\partial x} + \frac{4}{3}\frac{\partial v}{\partial y} - \frac{2}{3}\frac{\partial w}{\partial z} \right] \right.$$

$$\left. + \frac{\partial W_i}{\partial z}\left[\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} \right] \right\} d\Omega$$

$$+ \int_\Gamma W_i \frac{\mu}{Re} \left\{ \left[\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right]n_x + \left[-\frac{2}{3}\frac{\partial u}{\partial x} + \frac{4}{3}\frac{\partial v}{\partial y} - \frac{2}{3}\frac{\partial w}{\partial z} \right]n_y \right.$$

$$\left. + \left[\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} \right]n_z \right\} d\Gamma \tag{3.44}$$

z-momentum:

$$\int_\Omega W_i \rho \frac{\partial w}{\partial t} d\Omega = -\int_\Omega W_i \rho \left[u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z} \right] d\Omega$$

$$- \int_\Omega W_i \left[\frac{1}{\gamma M^2}\frac{\partial p}{\partial z} \right] d\Omega$$

$$- \int_\Omega \frac{\mu}{Re} \left\{ \frac{\partial W_i}{\partial x}\left[\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right] + \frac{\partial W_i}{\partial y}\left[\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right] \right.$$

$$\left. + \frac{\partial W_i}{\partial z}\left[-\frac{2}{3}\frac{\partial u}{\partial x} - \frac{2}{3}\frac{\partial v}{\partial y} + \frac{4}{3}\frac{\partial w}{\partial z} \right] \right\} d\Omega$$

$$+ \int_\Gamma W_i \frac{\mu}{Re} \left\{ \left[\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right]n_x + \left[\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right]n_y \right.$$

$$\left. + \left[-\frac{2}{3}\frac{\partial u}{\partial x} - \frac{2}{3}\frac{\partial v}{\partial y} + \frac{4}{3}\frac{\partial w}{\partial z} \right]n_z \right\} d\Gamma \tag{3.45}$$

internal energy:

$$\int_\Omega W_i \rho \frac{\partial e}{\partial t} d\Omega = - \int_\Omega W_i \rho \left[u \frac{\partial e}{\partial x} + v \frac{\partial e}{\partial y} + w \frac{\partial e}{\partial z} \right] d\Omega$$

$$- \int_\Omega \frac{\gamma\mu}{\mathrm{Re\,Pr}} \left[\frac{\partial W_i}{\partial x}\frac{\partial e}{\partial x} + \frac{\partial W_i}{\partial y}\frac{\partial e}{\partial y} + \frac{\partial W_i}{\partial z}\frac{\partial e}{\partial z} \right] d\Omega$$

$$- \int_\Omega W_i (\gamma - 1) p \left[\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right] d\Omega$$

$$- \int_\Omega W_i \left[\gamma(\gamma-1) M_\infty^2 \right] \frac{\mu}{\mathrm{Re}} \left\{ -\frac{2}{3}\left[\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right]^2 \right.$$

$$+ 2\left(\frac{\partial u}{\partial x} \right)^2 + 2\left(\frac{\partial v}{\partial y} \right)^2 + 2\left(\frac{\partial w}{\partial z} \right)^2$$

$$\left. + \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^2 + \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right)^2 \right\} d\Omega$$

$$+ \int_\Gamma \frac{\gamma\mu}{\mathrm{Re\,Pr}} W_i \left[\frac{\partial e}{\partial x}\mathrm{n}_x + \frac{\partial e}{\partial y}\mathrm{n}_y + \frac{\partial e}{\partial z}\mathrm{n}_z \right] d\Gamma \qquad (3.46)$$

Petrov-Galerkin Method

As discussed earlier, choosing the weighting functions in equations (3.42) through (3.46) to be equal to the shape functions (i.e. $W_i = N_i$) produces a Galerkin finite element formulation. Using trilinear hexahedral elements, the density, velocity, temperature and pressure approximations for each element are:

$$\rho(\mathbf{x},t) = \sum_{i=1}^{nnode} N_i(\mathbf{x})\rho_i(t) \qquad (3.47)$$

$$\mathbf{u}(\mathbf{x},t) = \sum_{i=1}^{nnode} N_i(\mathbf{x})\mathbf{u}_i(t) \qquad (3.48)$$

$$e(\mathbf{x},t) = \sum_{i=1}^{nnode} N_i(\mathbf{x})e_i(t) \tag{3.49}$$

$$p(\mathbf{x},t) = \sum_{i=1}^{nnode} N_i(\mathbf{x})p_i(t) \tag{3.50}$$

where *nnode* is the number of nodes, and N_i denotes the appropriate shape function. Derivatives of the computational variables for each element are approximated as:

$$\frac{\partial \Phi}{\partial t}(\mathbf{x},t) = \sum_{i=1}^{nnode} N_i(\mathbf{x})\frac{\partial \Phi_i}{\partial t}(t) \tag{3.51}$$

$$\frac{\partial \Phi}{\partial x}(\mathbf{x},t) = \sum_{i=1}^{nnode} \frac{\partial N_i}{\partial x}(\mathbf{x})\Phi_i(t) \tag{3.52}$$

$$\frac{\partial \Phi}{\partial y}(\mathbf{x},t) = \sum_{i=1}^{nnode} \frac{\partial N_i}{\partial y}(\mathbf{x})\Phi_i(t) \tag{3.53}$$

$$\frac{\partial \Phi}{\partial z}(\mathbf{x},t) = \sum_{i=1}^{nnode} \frac{\partial N_i}{\partial z}(\mathbf{x})\Phi_i(t) \tag{3.54}$$

where Φ represents the computational variables ρ, $\mathbf{u}$, e, and p.

One of the major difficulties with the Galerkin method is the production of spurious oscillations throughout the computational domain (Almeida and Galeao, 1996). These spurious oscillations tend to manifest in areas where downwind hard boundary conditions exist. A widely used method of minimizing these oscillations is by the addition of some perturbation factors, $P_u^i(\mathbf{x})$, $P_v^i(\mathbf{x})$, $P_w^i(\mathbf{x})$, and $P_e^i(\mathbf{x})$ to the Galerkin weighting functions on the advection terms, causing them to be weighted in the upwind direction (Hughes and Brooks, 1979; Hughes, et al., 1982). This results in the Petrov-Galerkin method, sometimes called the "upwind Petrov-Galerkin" or "streamline upwind Petrov-Galerkin" method.

The Petrov-Galerkin weighting functions for momentum and energy are, for all nodes $i = 1, 2, \ldots, nnode$:

$$W_u^i(\mathbf{x}) = N^i(\mathbf{x}) + P_u^i(\mathbf{x}) \tag{3.55}$$

$$W_v^i(\mathbf{x}) = N^i(\mathbf{x}) + P_v^i(\mathbf{x}) \tag{3.56}$$

$$W_w^i(\mathbf{x}) = N^i(\mathbf{x}) + P_w^i(\mathbf{x}) \tag{3.57}$$

$$W_e^i(\mathbf{x}) = N^i(\mathbf{x}) + P_e^i(\mathbf{x}) \tag{3.58}$$

Substituting the perturbation functions derived by Brueckner (1991) into equations (3.55) through (3.58) yields:

$$W_u^i(\mathbf{x}) = N^i(\mathbf{x}) + \frac{\alpha_u h}{2|\mathbf{u}|}\left(\mathbf{u} \cdot \nabla N^i(\mathbf{x})\right) \tag{3.59}$$

$$W_v^i(\mathbf{x}) = N^i(\mathbf{x}) + \frac{\alpha_v h}{2|\mathbf{u}|}\left(\mathbf{u} \cdot \nabla N^i(\mathbf{x})\right) \tag{3.60}$$

$$W_w^i(\mathbf{x}) = N^i(\mathbf{x}) + \frac{\alpha_w h}{2|\mathbf{u}|}\left(\mathbf{u} \cdot \nabla N^i(\mathbf{x})\right) \tag{3.61}$$

$$W_e^i(\mathbf{x}) = N^i(\mathbf{x}) + \frac{\alpha_e h}{2|\mathbf{u}|}\left(\mathbf{u} \cdot \nabla N^i(\mathbf{x})\right) \tag{3.62}$$

where

$$\alpha_u = \coth\left(\frac{\gamma_u}{2}\right) - \frac{2}{\gamma_u} \tag{3.63}$$

$$\alpha_v = \coth\left(\frac{\gamma_v}{2}\right) - \frac{2}{\gamma_v} \tag{3.64}$$

$$\alpha_w = \coth\left(\frac{\gamma_w}{2}\right) - \frac{2}{\gamma_w} \tag{3.65}$$

$$\alpha_e = \coth\left(\frac{\gamma_e}{2}\right) - \frac{2}{\gamma_e} \tag{3.66}$$

and

$$\gamma_u = \frac{\rho \, Re |\mathbf{u}| h}{\mu \left(1 + \dfrac{u^2}{3|\mathbf{u}|^2} \right)} \tag{3.67}$$

$$\gamma_v = \frac{\rho \, Re |\mathbf{u}| h}{\mu \left(1 + \dfrac{v^2}{3|\mathbf{u}|^2} \right)} \tag{3.68}$$

$$\gamma_w = \frac{\rho \, Re |\mathbf{u}| h}{\mu \left(1 + \dfrac{w^2}{3|\mathbf{u}|^2} \right)} \tag{3.69}$$

$$\gamma_e = \frac{\rho \, Pr \, Re |\mathbf{u}| h}{\mu \gamma} \tag{3.70}$$

For the trilinear hexahedral element, the characteristic element length, h, is given by:

$$h = |h_1| + |h_2| + |h_3| \tag{3.71}$$

where

$$h_1 = \frac{\mathbf{u} \cdot \mathbf{a}}{|\mathbf{u}|} \tag{3.72}$$

$$h_2 = \frac{\mathbf{u} \cdot \mathbf{b}}{|\mathbf{u}|} \tag{3.73}$$

$$h_3 = \frac{\mathbf{u} \cdot \mathbf{c}}{|\mathbf{u}|} \tag{3.74}$$

and, as shown in Figure 3-2,

$$\mathbf{a} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \frac{1}{4} \begin{bmatrix} (x_2 - x_1) + (x_3 - x_4) + (x_6 - x_5) + (x_7 - x_8) \\ (y_2 - y_1) + (y_3 - y_4) + (y_6 - y_5) + (y_7 - y_8) \\ (z_2 - z_1) + (z_3 - z_4) + (z_6 - z_5) + (z_7 - z_8) \end{bmatrix} \tag{3.75}$$

$$\mathbf{b} = \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} = \frac{1}{4}\begin{bmatrix} (x_3 - x_2) + (x_4 - x_1) + (x_7 - x_6) + (x_8 - x_5) \\ (y_3 - y_2) + (y_4 - y_1) + (y_7 - y_6) + (y_8 - y_5) \\ (z_3 - z_2) + (z_4 - z_1) + (z_7 - z_6) + (z_8 - z_5) \end{bmatrix} \qquad (3.76)$$

$$\mathbf{c} = \begin{bmatrix} c_x \\ c_y \\ c_z \end{bmatrix} = \frac{1}{4}\begin{bmatrix} (x_5 - x_1) + (x_6 - x_2) + (x_7 - x_3) + (x_8 - x_4) \\ (y_5 - y_1) + (y_6 - y_2) + (y_7 - y_3) + (y_8 - y_4) \\ (z_5 - z_1) + (z_6 - z_2) + (z_7 - z_3) + (z_8 - z_4) \end{bmatrix} \qquad (3.77)$$
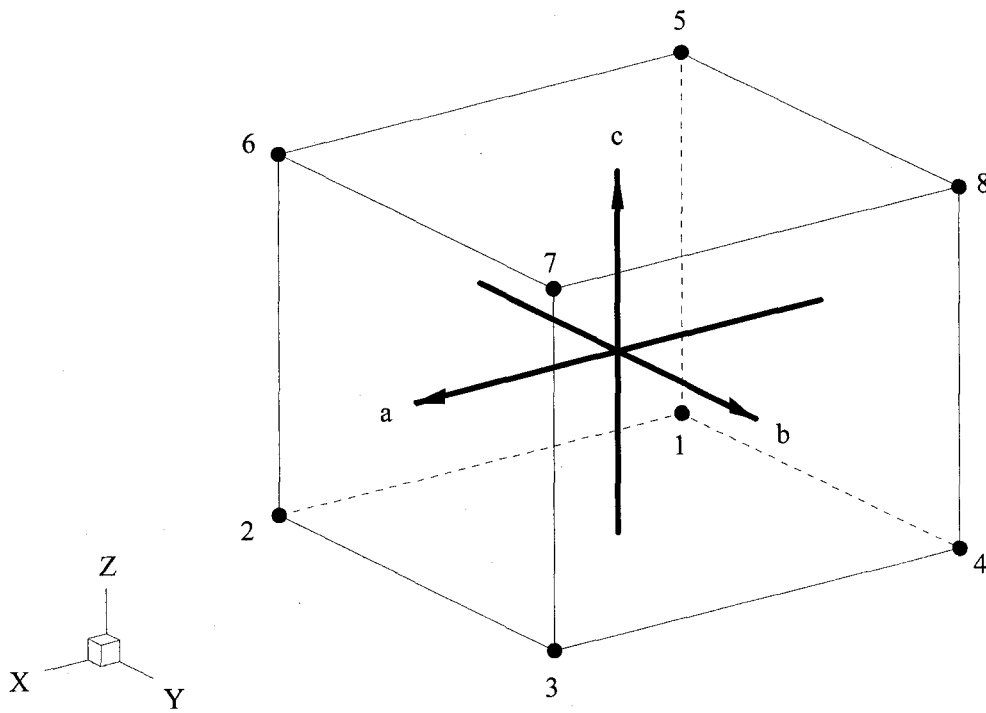

Figure 3-2. Definition of vectors **a**, **b**, and **c**.

In three dimensions, the Petrov-Galerkin form of the governing equations may now be written as follows:

continuity:

$$\int_\Omega N_i \left(N_j \frac{\partial \rho_j}{\partial t} \right) d\Omega =$$

$$- \int_\Omega N_i \left[\left(\sum N_j u_j \right) \left(\sum \frac{\partial N_j}{\partial x} \rho_j \right) + \left(\sum N_j v_j \right) \left(\sum \frac{\partial N_j}{\partial y} \rho_j \right) \right.$$

$$+ \left(\sum N_j w_j \right) \left(\sum \frac{\partial N_j}{\partial z} \rho_j \right) \right] d\Omega$$

$$- \int_\Omega N_i \left(\sum N_j \rho_j \right) \left[\left(\sum \frac{\partial N_j}{\partial x} u_j \right) + \left(\sum \frac{\partial N_j}{\partial x} v_j \right) + \left(\sum \frac{\partial N_j}{\partial x} w_j \right) \right] d\Omega$$

(3.78)

x-momentum:

$$\int_\Omega \rho N_i N_j \frac{\partial u_j}{\partial t} d\Omega =$$

$$- \int_\Omega \rho W_u \left[\left(\sum N_j u_j \right) \left(\sum \frac{\partial N_j}{\partial x} u_j \right) + \left(\sum N_j v_j \right) \left(\sum \frac{\partial N_j}{\partial y} u_j \right) \right.$$

$$\left. + \left(\sum N_j w_j \right) \left(\sum \frac{\partial N_j}{\partial z} u_j \right) \right] d\Omega$$

$$- \int_\Omega N_i \left[\frac{1}{\gamma M^2} \left(\sum \frac{\partial N_j}{\partial x} p_j \right) \right] d\Omega$$

$$- \int_\Omega \frac{\mu}{\text{Re}} \left\{ \frac{\partial N_i}{\partial x} \left[\frac{4}{3} \left(\sum \frac{\partial N_j}{\partial x} u_j \right) - \frac{2}{3} \left(\sum \frac{\partial N_j}{\partial y} v_j \right) - \frac{2}{3} \left(\sum \frac{\partial N_j}{\partial z} w_j \right) \right] \right.$$

$$+ \frac{\partial N_i}{\partial y} \left[\left(\sum \frac{\partial N_j}{\partial x} v_j \right) + \left(\sum \frac{\partial N_j}{\partial y} u_j \right) \right]$$

$$\left. + \frac{\partial N_i}{\partial z} \left[\left(\sum \frac{\partial N_j}{\partial x} w_j \right) + \left(\sum \frac{\partial N_j}{\partial z} u_j \right) \right] \right\} d\Omega$$

(3.79)

y-momentum:

$$
\int_\Omega \rho N_i N_j \frac{\partial v_j}{\partial t}\, d\Omega = -\int_\Omega \rho W_v \left[\left(\sum N_j u_j\right)\left(\sum \frac{\partial N_j}{\partial x} v_j\right) + \left(\sum N_j v_j\right)\left(\sum \frac{\partial N_j}{\partial y} v_j\right) \right.
$$

$$
\left. + \left(\sum N_j w_j\right)\left(\sum \frac{\partial N_j}{\partial z} v_j\right) \right] d\Omega
$$

$$
- \int_\Omega N_i \left[\frac{1}{\gamma M^2}\left(\sum \frac{\partial N_j}{\partial y} p_j\right) \right] d\Omega
$$

$$
- \int_\Omega \frac{\mu}{\mathrm{Re}} \left\{ \frac{\partial N_i}{\partial x}\left[\left(\sum \frac{\partial N_j}{\partial y} u_j\right) + \left(\sum \frac{\partial N_j}{\partial x} v_j\right) \right] \right.
$$

$$
+ \frac{\partial N_i}{\partial y}\left[-\frac{2}{3}\left(\sum \frac{\partial N_j}{\partial x} u_j\right) + \frac{4}{3}\left(\sum \frac{\partial N_j}{\partial y} v_j\right) - \frac{2}{3}\left(\sum \frac{\partial N_j}{\partial z} w_j\right) \right]
$$

$$
\left. + \frac{\partial N_i}{\partial z}\left[\left(\sum \frac{\partial N_j}{\partial y} w_j\right) + \left(\sum \frac{\partial N_j}{\partial z} v_j\right) \right] \right\} d\Omega \tag{3.80}
$$

z-momentum:

$$
\int_\Omega \rho N_i N_j \frac{\partial w_j}{\partial t}\, d\Omega = -\int_\Omega \rho W_w \left[\left(\sum N_j u_j\right)\left(\sum \frac{\partial N_j}{\partial x} w_j\right) + \left(\sum N_j v_j\right)\left(\sum \frac{\partial N_j}{\partial y} w_j\right) \right.
$$

$$
\left. + \left(\sum N_j w_j\right)\left(\sum \frac{\partial N_j}{\partial z} w_j\right) \right] d\Omega
$$

$$
- \int_\Omega N_i \left[\frac{1}{\gamma M^2}\left(\sum \frac{\partial N_j}{\partial z} p_j\right) \right] d\Omega
$$

$$
- \int_\Omega \frac{\mu}{\mathrm{Re}} \left\{ \frac{\partial N_i}{\partial x}\left[\left(\sum \frac{\partial N_j}{\partial z} u_j\right) + \left(\sum \frac{\partial N_j}{\partial x} w_j\right) \right] \right.
$$

$$
+ \frac{\partial N_i}{\partial y}\left[\left(\sum \frac{\partial N_j}{\partial z} v_j\right) + \left(\sum \frac{\partial N_j}{\partial y} w_j\right) \right]
$$

$$
\left. + \frac{\partial N_i}{\partial z}\left[-\frac{2}{3}\left(\sum \frac{\partial N_j}{\partial x} u_j\right) - \frac{2}{3}\left(\sum \frac{\partial N_j}{\partial y} v_j\right) + \frac{4}{3}\left(\sum \frac{\partial N_j}{\partial z} w_j\right) \right] \right\} d\Omega \tag{3.81}
$$

internal energy:

$$\int_\Omega \rho N_i N_j \frac{\partial e_j}{\partial t} \, d\Omega =$$

$$- \int_\Omega \rho W_e \left[\left(\sum N_j u_j \right)\left(\sum \frac{\partial N_j}{\partial x} e_j \right) + \left(\sum N_j v_j \right)\left(\sum \frac{\partial N_j}{\partial y} e_j \right) \right.$$

$$\left. + \left(\sum N_j w_j \right)\left(\sum \frac{\partial N_j}{\partial z} e_j \right) \right] d\Omega$$

$$- \int_\Omega \frac{\gamma \mu}{\mathrm{Re}\,\mathrm{Pr}} \left[\frac{\partial N_i}{\partial x}\left(\sum \frac{\partial N_j}{\partial x} e_j \right) + \frac{\partial N_i}{\partial y}\left(\sum \frac{\partial N_j}{\partial y} e_j \right) + \frac{\partial N_i}{\partial z}\left(\sum \frac{\partial N_j}{\partial z} e_j \right) \right] d\Omega$$

$$- \int_\Omega W_i (\gamma - 1)\left(\sum N_j p_j \right)\left[\left(\sum \frac{\partial N_j}{\partial x} u_j \right) + \left(\sum \frac{\partial N_j}{\partial y} v_j \right) + \left(\sum \frac{\partial N_j}{\partial z} w_j \right) \right] d\Omega$$

$$- \int_\Omega W_i \left[\gamma(\gamma - 1) M_\infty^2 \right] \frac{\mu}{\mathrm{Re}} \left\{ -\frac{2}{3}\left[\left(\sum \frac{\partial N_j}{\partial x} u_j \right) + \left(\sum \frac{\partial N_j}{\partial y} v_j \right) + \left(\sum \frac{\partial N_j}{\partial z} w_j \right) \right]^2 \right.$$

$$+ 2\left(\sum \frac{\partial N_j}{\partial x} u_j \right)^2 + 2\left(\sum \frac{\partial N_j}{\partial y} v_j \right)^2 + 2\left(\sum \frac{\partial N_j}{\partial z} w_j \right)^2$$

$$+ \left[\left(\sum \frac{\partial N_j}{\partial y} u_j \right) + \left(\sum \frac{\partial N_j}{\partial x} v_j \right) \right]^2 + \left[\left(\sum \frac{\partial N_j}{\partial z} u_j \right) + \left(\sum \frac{\partial N_j}{\partial x} w_j \right) \right]^2 \tag{3.82}$$

$$\left. + \left[\left(\sum \frac{\partial N_j}{\partial z} v_j \right) + \left(\sum \frac{\partial N_j}{\partial y} w_j \right) \right]^2 \right\} d\Omega$$

Temporal Integration

Equations (3.78) through (3.82) can now be expressed as a system of ordinary differential equations:

$$[\mathbf{L}]\left\{ \frac{\partial \rho_i}{\partial t} \right\} = \{ \mathbf{R}_\rho \} \tag{3.83}$$

$$[\mathbf{M}]\left\{\frac{\partial u_i}{\partial t}\right\} = \{\mathbf{R}_u\} \qquad (3.84)$$

$$[\mathbf{M}]\left\{\frac{\partial v_i}{\partial t}\right\} = \{\mathbf{R}_v\} \qquad (3.85)$$

$$[\mathbf{M}]\left\{\frac{\partial w_i}{\partial t}\right\} = \{\mathbf{R}_w\} \qquad (3.86)$$

$$[\mathbf{M}]\left\{\frac{\partial e_i}{\partial t}\right\} = \{\mathbf{R}_e\} \qquad (3.87)$$

Here, the mass matrices **L** and **M** are:

$$[\mathbf{L}] = \left[\int_\Omega N_i \left(N_j\right) d\Omega \right] \qquad (3.88)$$

$$[\mathbf{M}] = \left[\int_\Omega \rho N_i \left(N_j\right) d\Omega \right] \qquad (3.89)$$

and $\{\mathbf{R}_\rho\}$, $\{\mathbf{R}_u\}$, $\{\mathbf{R}_v\}$, $\{\mathbf{R}_w\}$, and $\{\mathbf{R}_e\}$ are the right hand side load vectors.

In order to obtain a fully explicit algorithm, mass lumping is used to diagonalize the mass matrices. The diagonal terms for the lumped mass matrices for the continuity equation, l_i, and the momentum and energy equations, m_i, are given by (Brueckner, 1991):

$$l_i = \int_\Omega N_i \left(\sum_{j=1}^n N_j\right) d\Omega \qquad (3.90)$$

$$m_i = \int_\Omega \rho N_i \left(\sum_{j=1}^n N_j\right) d\Omega \qquad (3.91)$$

Because the mass matrix is lumped along the diagonal, inversion is trivial, allowing an explicit solution to the system of equations. In this work, an explicit Euler scheme is used in conjunction with a second-order Runge-Kutta approximation to advance the discretized equations in time.

$$\mathbf{\Phi}^{n+\frac{1}{2}} = \mathbf{\Phi}^n + \frac{\Delta t}{2} \left[\mathbf{M}^{-1}\mathbf{R} \right]^n \tag{3.92}$$

$$\mathbf{\Phi}^{n+1} = \mathbf{\Phi}^{n+\frac{1}{2}} - \Delta t \left[\mathbf{M}^{-1}\mathbf{R} \right]^{n+\frac{1}{2}} \tag{3.93}$$

where $\mathbf{\Phi}$ contains the solution variables:

$$\mathbf{\Phi} = \begin{bmatrix} \rho_i \\ u_i \\ v_i \\ w_i \\ T_i \end{bmatrix} \tag{3.94}$$

Finally, at each node, the value of the pressure is calculated using the equation of state.

$$p_i = \rho_i e_i \tag{3.95}$$

Since this is an explicit algorithm, there is, of course, a limitation on the size of the time step. In order to ensure stability, the time step used is simply the CFL stability criterion for compressible flow given by (Roache, 1976):

$$\Delta t \leq \frac{h}{|\mathbf{u}| + a} \tag{3.96}$$

In equation (3.96) h is the characteristic element length given by (3.71), $|\mathbf{u}|$ is the magnitude of the velocity, and a is the local speed of sound. A global value of Δt is used, meaning that Δt is calculated for every element in the domain and the smallest value is used.

For steady state problems the equations are advanced in time until each of the dependent variables is converged to an absolute error, ε, according to the relation:

$$\left| \Phi^n - \Phi^{n+1} \right| \leq \varepsilon \qquad (3.97)$$

where $\varepsilon = 10^{-4}$.

CHAPTER 4

MESH ADAPTATION

Mesh adaptation provides a means of achieving the fine mesh density required for accurately locating shocks without unnecessarily compromising computational speed (Pepper and Stephenson, 1995). For any given system of equations, as the number of elements in the computational domain decreases, the work at each time step will decrease, but the overall error will most likely increase (Capon and Jimack, 1995). Rather than using a finer mesh throughout the entire solution domain, the mesh can be refined in areas with extreme flow feature gradients and left alone or unrefined in areas with little or no gradients, thus providing a balance in solution accuracy and computational speed.

There are three distinct adaptation schemes currently in use: *r*-adaptation, *p*-adaptation, and *h*-adaptation. In *r*-adaptive methods, the number of elements in a mesh remains constant, but relocating nodes changes the size and shape of the elements. This method is also known as adaptive remeshing. The advantage of this method is that the overall number of elements remains fixed, thus the number of computations per iteration is fixed. This method is useful for quickly locating shocks in Euler flows, because nodes can be quickly relocated to regions of high gradients. However, this method does not perform well for viscous flows due to the difficulties in resolving the mesh around both shocks and boundary layers (Ilinca, et al., 1998). Another disadvantage of r-adaptive

32

methods is that odd shaped elements can easily be produced. In finite element methods, highly skewed elements are problematic, often causing the solution process to diverge.

In *p*-adaptive methods the number of elements in a mesh remains constant, but the degree of the interpolating polynomial is increased on elements where greater solution accuracy is warranted. For example, in two dimensions, an initial mesh may be comprised of four-node quadrilateral elements with a linear interpolating polynomial. For p-adaptation to occur, the polynomial must be changed to a quadratic, cubic, etc., which requires additional nodes. This method has the advantage of high solution accuracy. The disadvantage is that *p*-adaptive methods can be troublesome to implement. The development of the required data management scheme can be difficult, especially on domains with complex geometries (Oden and Demkowicz, 1989).

In *h*-adaptive methods, the number of elements in a mesh changes through a series of refinements and recoveries. Refinement can be accomplished in one of two ways: block embedding or element division. Block embedding (see Figure 4-1) is a scheme where an element of similar shape is embedded in the original element and nodes between old and new are connected in such a way that elements of similar type are created. The distinct advantage of block embedding over element division is that block embedding creates no constrained or "virtual" nodes, which can become computationally expensive. This method of refinement works well if elements are to be refined one level. Beyond one level, it is very likely that highly skewed and odd-shaped elements will be created.

Element division creates new elements by dividing the original parent element through midpoints on all sides (see Figure 4-2). This method creates new child elements that are essentially the same shape as the original parent element. Therefore, if the parent

elements comprising the original mesh are well shaped, then all subsequent child elements will be well shaped. Even though element division creates virtual nodes, it is well suited for multiple levels of division. For these reasons, element division is the refinement scheme used in this study.
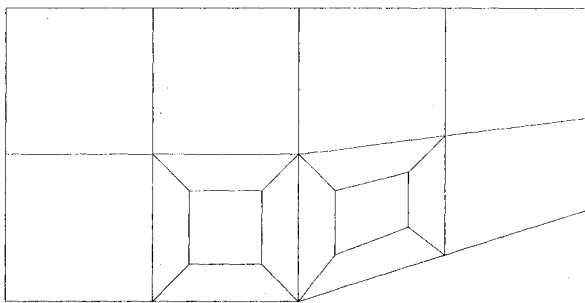

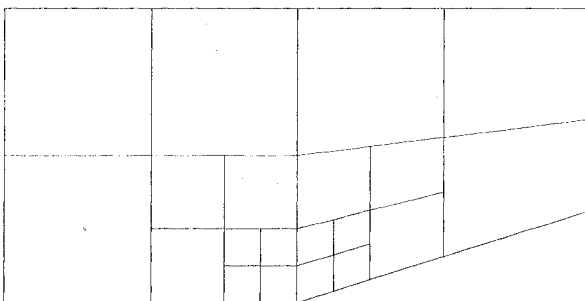Figure 4-1. Refinement using block embedding

Figure 4-2. Refinement using element division

Overall, the h-adaptive method has the advantage of increased solution accuracy. However, this increase in accuracy is achieved at the expense of computational speed. As the number of elements increases, the number of computations per iteration also increases. There has been much interest in p-adaptive and hp-adaptive schemes lately,

and it has been demonstrated that, in many cases, they perform as well or better than *h*-adaptive methods (Zienkiewicz and Taylor, 2000). However, for the purposes of this study, the *h*-adaptation process is employed, since it is less complex to program.

Adaptation Rules

The *h*-adaptation process is governed by some general rules (Heinrich and Pepper, 1999), which can be summarized as follows:

Refinement rules:

1. Refinement of a quadrilateral element results in the creation of 4 child elements and in the creation of up to five new nodes. Refinement of a hexahedral element results in the creation of 8 child elements and in the creation of up to nineteen new nodes.

2. An element may be refined only when adjacent elements are at the same refinement level or higher.

3. Duplicate nodes must not be created.

4. An embedded node that is created along a domain boundary is not a virtual node.

Recovery rules:

1. To be unrefined, a group of elements must not contain another group of elements, and each element of the group to be unrefined must not be a neighbor to an element with a higher level.

2. If a group of elements is unrefined, then the embedded virtual node along the edge common to an element that is not a member of the group will be eliminated.

3. If a group of elements is unrefined, then the node along the edge common to this group and its neighbor group will become a virtual node.

4. If a group of elements is unrefined, all embedded nodes along a domain boundary will be eliminated.

The first refinement rule is the most basic principle of the *h*-adaptive process, and is illustrated in Figure 4-3. It states that the refinement of a bilinear quadrilateral element results in the creation of 4 child elements and in the creation of up to five new nodes. It also states that the refinement of a trilinear hexahedral element results in the creation of 8 child elements and in the creation of up to nineteen new nodes.
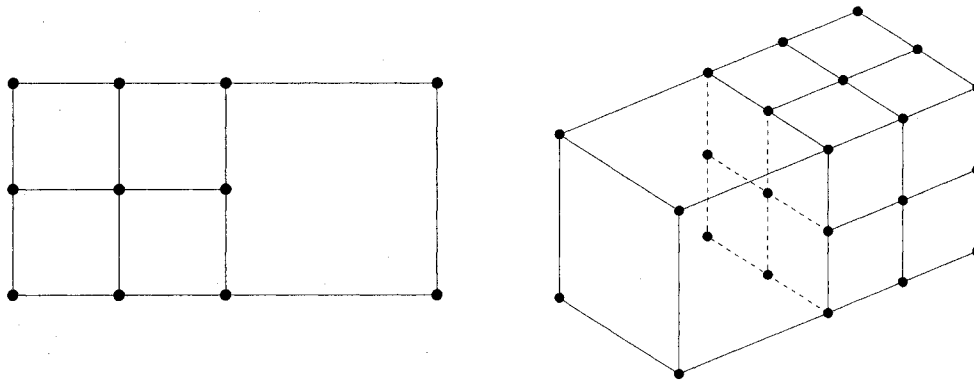

Figure 4-3. Refined and unrefined elements in two and three dimensions.

The second refinement rule states that an element may be refined only when adjacent elements are at same level or one level higher. The purpose of this rule is to limit the number of virtual nodes in the computational domain. As more elements throughout the computational domain are adapted, more virtual nodes are created. The values of the solution variables at the virtual nodes must be calculated separately and can quickly become computationally expensive.

Consider the two-dimensional case of two adjacent elements where only one undergoes the adaptation process (see Figure 4-4). The bilinear parent element that has been adapted has yielded the four bilinear child elements *A*, *B*, *C*, and *D*. Each child element has four nodes, one at each corner. The adjacent unadapted, bilinear element *E* still has one node at each of its four corners, but now also has a new virtual node formed by the children of the adjacent parent element. If elements *B* and *D* were adapted again, and the element *E* remained unchanged, the number of virtual nodes in the unadapted element would climb to three (see Figure 4-5). In this scenario, in order to comply with the second refinement rule, element *E* would have to be adapted, creating four new elements (see Figure 4-6), with elements *F* and *H* each having only one virtual node.
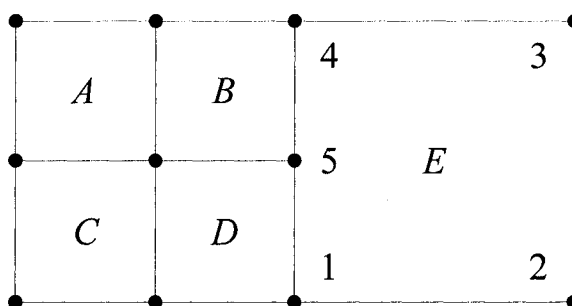

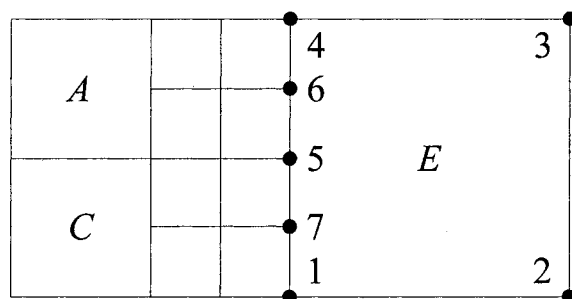Figure 4-4. Unrefined element *E* has one virtual node (node 5).

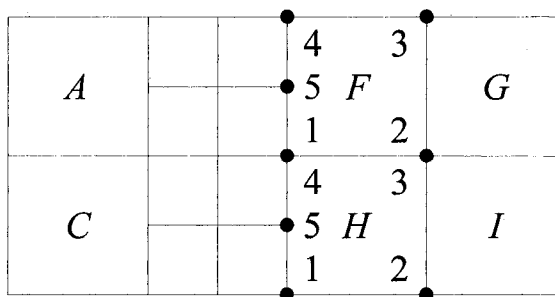Figure 4-5. Unrefined element *E* has three virtual nodes (nodes 5, 6, and 7).

Figure 4-6. Element E is adapted to prevent excess virtual nodes.

Extending this concept to three dimensions, the importance of this rule becomes even more evident. Consider two adjacent, three-dimensional, trilinear, hexahedral elements where one undergoes adaptation (see Figure 4-7). Eight new trilinear child elements are created, and the unrefined element A gains five virtual nodes on the face bordering the new children B, C, D, and E. If elements B, C, D, and E were refined again, and element A were left unchanged, the number of virtual nodes on the bordering face of element A would jump to 21.
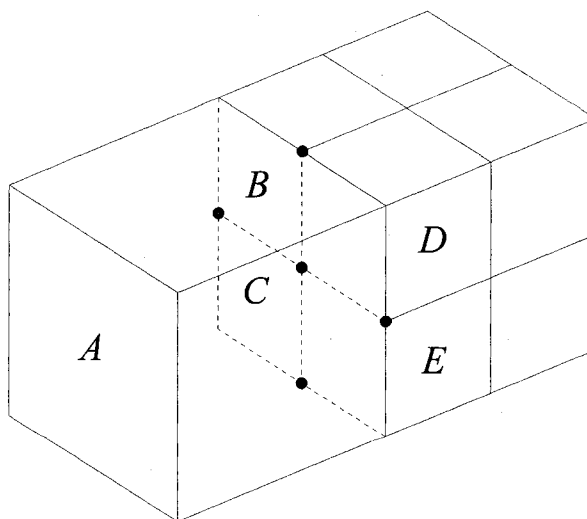

Figure 4-7. Location of the five virtual nodes on the bordering face.

Also, when working in three dimensions, the elements adjacent to the adjacent elements must be considered, due to the fact that they also can create virtual nodes. In other words, if any of the elements adjacent to elements B, C, D, or E in Figure 4-7 were adapted to a higher level than elements B, C, D, or E then additional virtual nodes would be introduced to element A (see Figure 4-8), thus violating the second refinement rule.
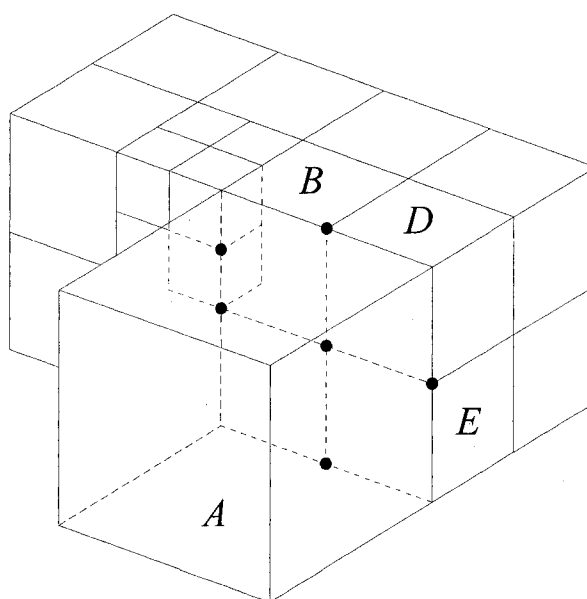

Figure 4-8. Element A now has six virtual nodes.

Refinement rule three states that duplicate nodes must not be created. This rule has critical importance due to the fact that two distinct nodes occupying the same spatial coordinates can cause the solution process to diverge. Therefore, to eliminate the possibility of creating duplicate nodes, existing nodes in adjacent child elements must be utilized when adapting an element. For example, if the element E in Figure 4-4 were adapted, node 3 of element D or node 2 of the element B would have to be incorporated into the connectivity of the new child elements. As a consequence of this action, the

virtual node (node 5) of element E is eliminated. As can be seen in figure 4-6, the previous virtual node is now replaced by a real node that can be described as node 1 of element F or node 4 of element H.

As before, the incorporation of a third dimension brings more difficulty to the situation. In three dimensions the elements adjacent to the adjacent elements, as well as the adjacent elements themselves must be examined for existing nodes, so as not to create any duplicates. As seen in figure 4-7, if element A were to be adapted, the five nodes from elements B, C, D, and E corresponding to the five virtual nodes of element A must be incorporated into the structure of the new child elements to be created. If the element to be adapted has no elements at a higher level adjacent to any of its faces, then the elements adjacent to the adjacent elements must be examined. This is due to the fact that these elements share a common edge with the parent element being adapted, on which sits a single node which must be utilized (see figure 4-9)
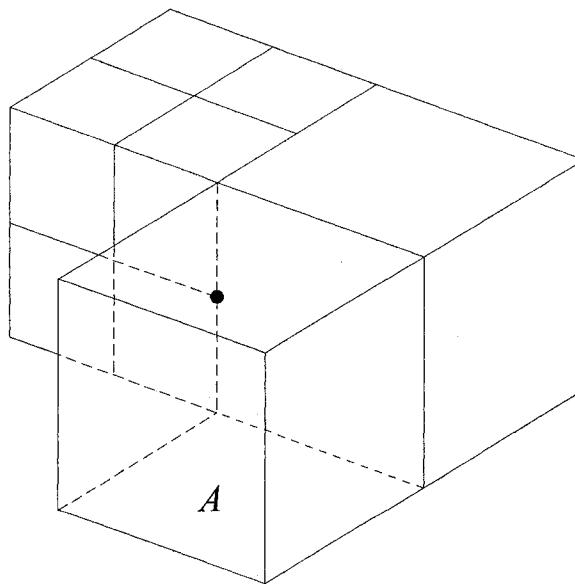

Figure 4-9. Elements adjacent to the adjacent of element A
contributes virtual node to element A.

Refinement rule four states that an embedded node that is created along a domain boundary is not a virtual node. Such nodes are given the same boundary conditions applied to the nodes on either side of it.

The recovery rules are simply the opposite of the refinement rules and subject to all the conditions described above for the refinement rules.

Adaptation Process

The h-adaptive process can be generally described in a few simple steps. First, at a specified time in the calculation, the iterative solution process is stopped and flow feature gradients are calculated across every element in the solution domain. This process is flexible in that any flow feature, such as velocity, density, temperature, pressure, etc., or a combination of flow features can be used as a trigger for mesh adaptation. In the case of compressible flows, the density gradient is a suitable criterion for element refinement (Burton and Pepper, 1993).

Second, a certain number of elements are tagged for adaptation. For each element the density gradient is calculated using the relation:

$$A_e = \max_{ij} \left| \rho_i - \rho_j \right| \tag{4.1}$$

Next the mean is calculated,

$$\overline{A}_e = \frac{1}{n} \sum_{i=1}^{nelem} A_{e_i} \tag{4.2}$$

as well as the standard deviation,

$$\sigma = \sqrt{\frac{\sum_{i=1}^{nelem} A_{e_i}^2 - n\overline{A}_e^2}{n-1}} \tag{4.3}$$

Finally, a normal distribution of the gradient is determined by subtracting the mean and dividing by the standard deviation (Taylor and Karlin, 1994).

$$A_e^{\,n} = \frac{A_e - \overline{A}_e}{\sigma} \qquad (4.4)$$

Elements where $A_e^{\,n} > \alpha$ are tagged for refinement. The value of α is flexible, but experience shows that values of $0.7 < \alpha < 0.8$ work well.

Third, tagged elements are refined according to the refinement rules cited above. The values of the computational variables are interpolated onto the new nodes from the nodes of the parent element using a piecewise linear polynomial.

Fourth, holes are eliminated (see Figure 4-10). "Holes" are elements that are nearly or entirely surrounded by elements that have been adapted to a higher level. Holes have many virtual nodes, and are undesirable due to the computational expense associated with virtual nodes. The process of eliminating holes can very often create new holes. Therefore, this step is repeated until no new holes are found.
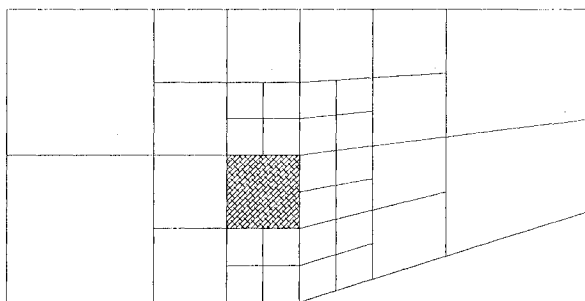

Figure 4-10. Shaded element is a hole.

Fifth, a certain number of elements are tagged for recovery. Recovery simply means that a parent element is recovered by eliminating its children. As in the refinement process, for each element the density gradient is calculated using equation (4.1), and this

quantity is then normalized by subtracting the mean and dividing by the standard deviation as in equation (4.4). Elements where $A_e^{\,n} < \beta$ are tagged for recovery. Experience shows that values of $0.2 < \beta < 0.3$ work well.

Sixth, tagged elements are recovered. Recovery occurs by simply reversing the process of refinement. As the child elements are removed, any nodes unique to the children as well as any corresponding virtual nodes are deleted.

Finally, the process is repeated at set intervals. For a rapidly changing solution, a higher interval is desired than if the solution is nearing convergence.

CHAPTER 5

VALIDATION

Results from several test cases will now be presented. All computations were performed on a personal computer with an Intel Pentium 4 processor with 1GB of RAM.

Three Dimensional Euler Flow Over 15° Compression Corner

Flow over a ramp, wedge or compression corner is a widely used test case for compressible flow models (Zienkiewicz and Taylor (b), 2000; Brueckner, 1991). In the Euler limit, the precise location of shocks can rapidly be obtained.
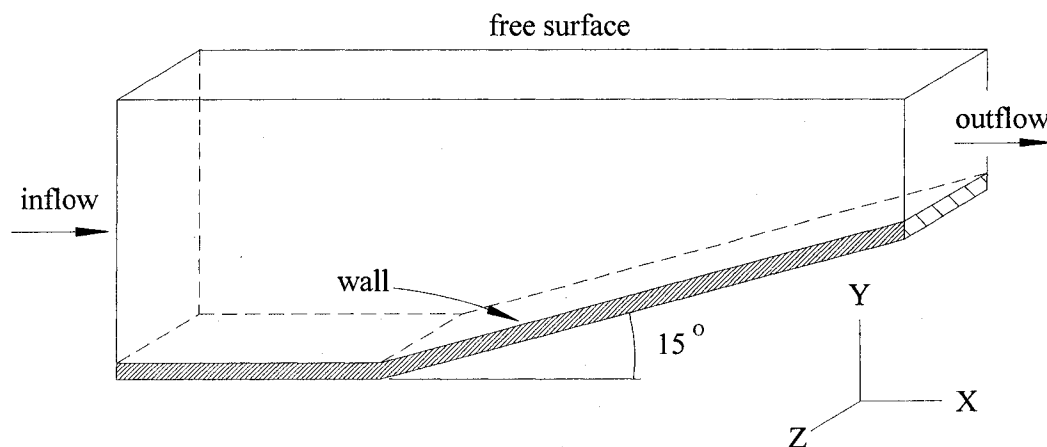

Figure 5-1. Boundary conditions for compressible flow over three-dimensional 15° compression corner.

44

The geometry and general boundary conditions are shown in Figure 5-1. The nodes at the inflow boundary are set according to the following dirichelet conditions:

$$u_{in} = U_\infty$$

$$v_{in} = w_{in} = 0$$

$$\rho_{in} = \rho_\infty$$

$$T_{in} = T_\infty$$

Since no dirichelet conditions are specified on either the free surface boundary or the outflow boundary, the finite element method described earlier defaults to a zero-flux boundary. As is common practice in Euler solutions, the wall is treated as a slip surface. In other words, only the velocity components tangential to the wall are applied.

The following fluid properties for air are used:

$$Pr = 0.72$$

$$\gamma = 1.4$$

$$Re = 10^4$$

Two cases are presented on the same geometry for free stream Mach numbers of three and five. For each case, solutions are presented with and without the use of h-adaptation. Based on the known two-dimensional oblique shock relations (Anderson, 1990), Mach 3 flow encountering a 15° wedge should produce a shock wave angled 32.3° up from the horizontal. The expected ratios for density, pressure and temperature are:

$$\frac{\rho_2}{\rho_1} = 2.03 \tag{5.1}$$

$$\frac{p_2}{p_1} = 2.82 \tag{5.2}$$

$$\frac{T_2}{T_1} = 1.39 \tag{5.3}$$

Mach 5 flow encountering a 15° wedge should produce a shock wave angled 24.3° up from the horizontal. The expected ratios for density, pressure and temperature are:

$$\frac{\rho_2}{\rho_1} = 2.75 \tag{5.4}$$

$$\frac{p_2}{p_1} = 4.80 \tag{5.5}$$

$$\frac{T_2}{T_1} = 1.74 \tag{5.6}$$

Note that in the equations above, subscript 1 indicates values in upstream of the shock, and subscript 2 indicates values downstream of the shock.

Compression Corner Results without Adaptation for $M_\infty = 3$

The best way to get a converged solution of a steady state problem using mesh adaptation is first to get a nearly converged solution on a coarse mesh. The mesh must be fine enough to allow flow features to develop, yet coarse enough to reach convergence quickly. The coarse mesh used for this case is presented in Figure 5-2. This mesh consists of 5408 elements with 6468 nodes.

Density, pressure and temperature contours along the central z-plane are presented in Figures 5-3 through 5-5. The shock wave is diffused over several element widths and is observed at a 33° angle. Irregularities on the outer edges of the shock wave indicate that the elements are too large to adequately capture the high density, pressure and temperature gradients.
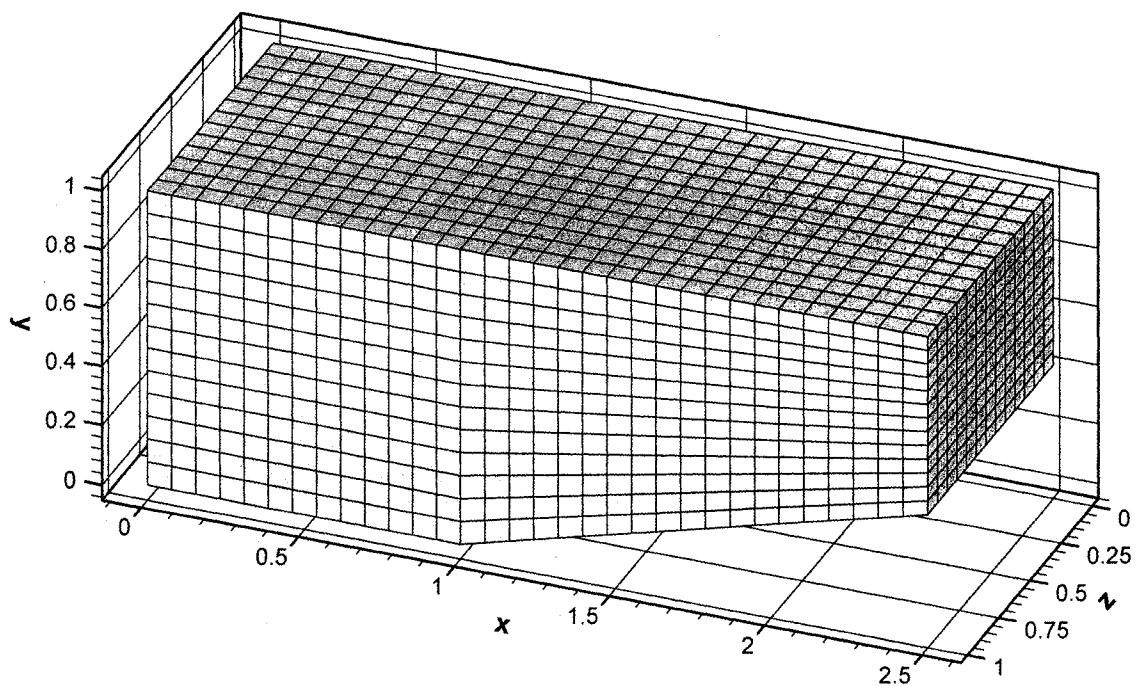
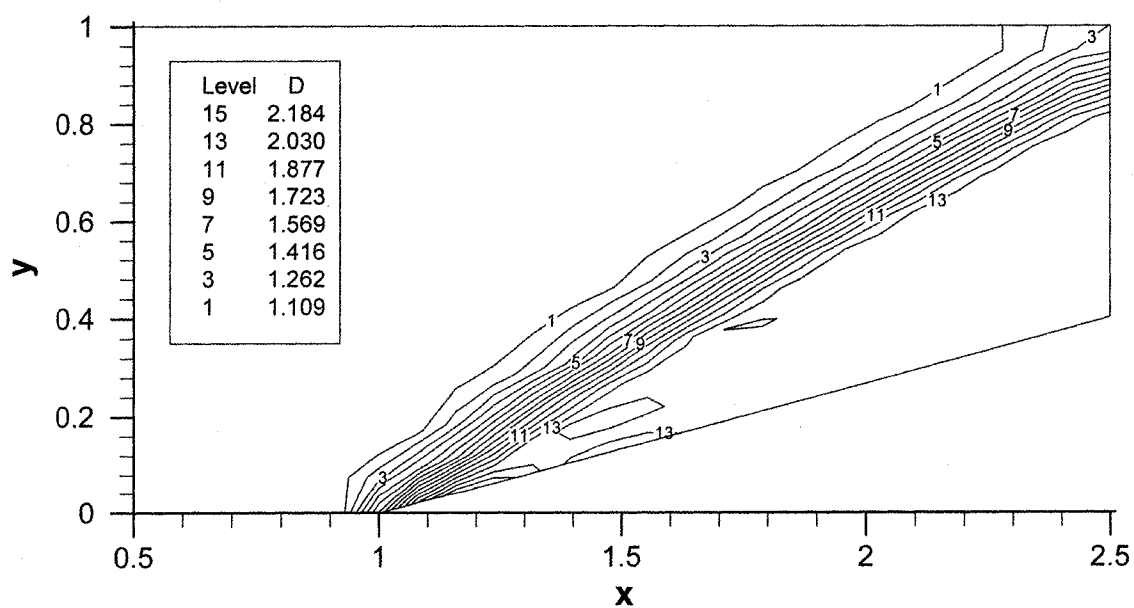Figure 5-2. Coarse mesh for three-dimensional 15° compression corner.

Figure 5-3. Density contours at $z = 0.5$ for three-dimensional compression corner at $M_\infty = 3$ without adaptation.
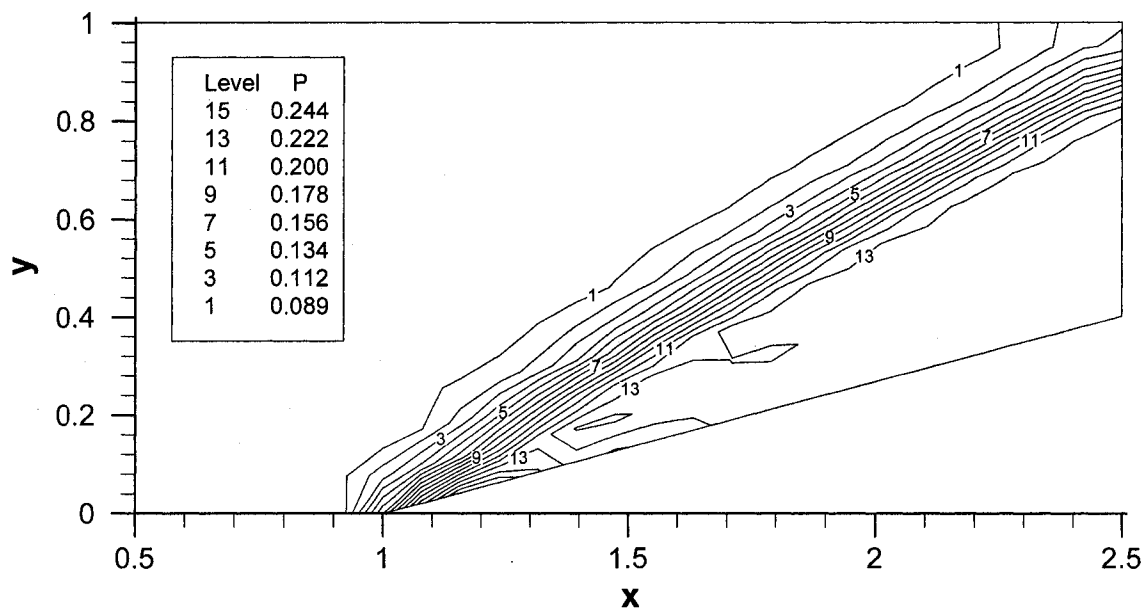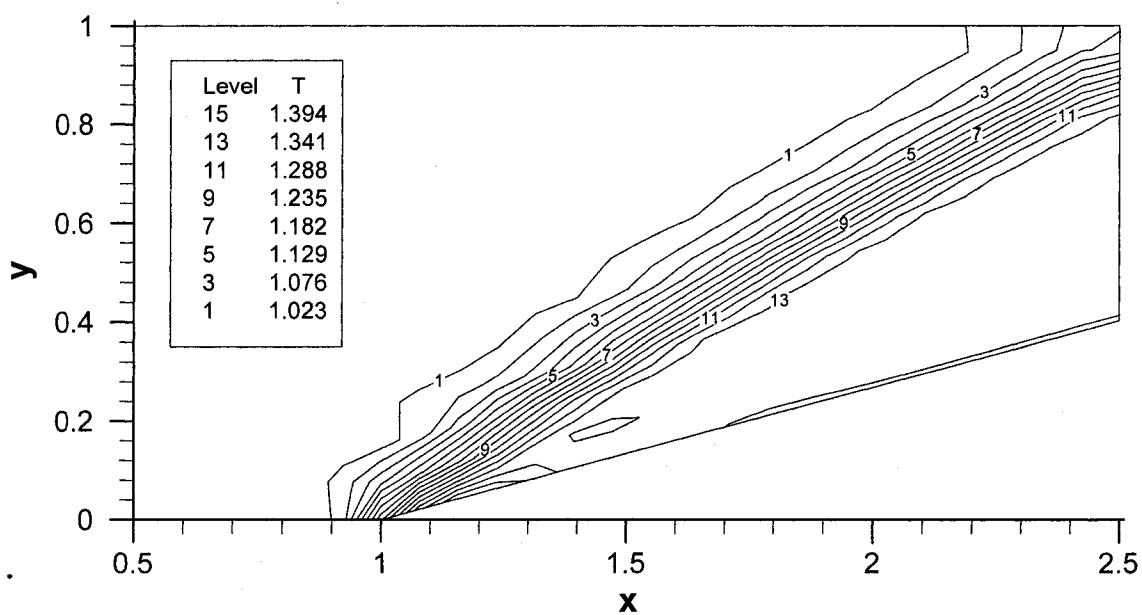
Level	P
15	0.244
13	0.222
11	0.200
9	0.178
7	0.156
5	0.134
3	0.112
1	0.089

Figure 5-4. Pressure contours at $z = 0.5$ for three-dimensional compression corner at $M_\infty = 3$ without adaptation.

Level	T
15	1.394
13	1.341
11	1.288
9	1.235
7	1.182
5	1.129
3	1.076
1	1.023

Figure 5-5. Temperature contours at $z = 0.5$ for three-dimensional compression corner at $M_\infty = 3$ without adaptation.

Compression Corner Results with Two Levels of Adaptation for $M_\infty = 3$

Once a nearly-converged solution is achieved on the coarse mesh, the h-adaptive algorithm is engaged. As described in Chapter four, as the solution progresses, elements undergo several refinements and recoveries. Once the location of the various flow features is sufficiently resolved, many elements are recovered and no further divisions are made. At this point, the mesh is said to be "converged." For this particular case, the converged mesh with two levels of adaptation is shown in Figure 5-6. It shows 57,096 elements and 59,086 nodes.

Density, pressure and temperature contours along the central z-plane are presented in Figures 5-7 through 5-9. While the shock wave is still diffused over several element widths, its exact location is more evident, and is observed at a 32.5° angle. Irregularities on the outer edges of the shock wave are minimized on the finer elements in the adapted region.
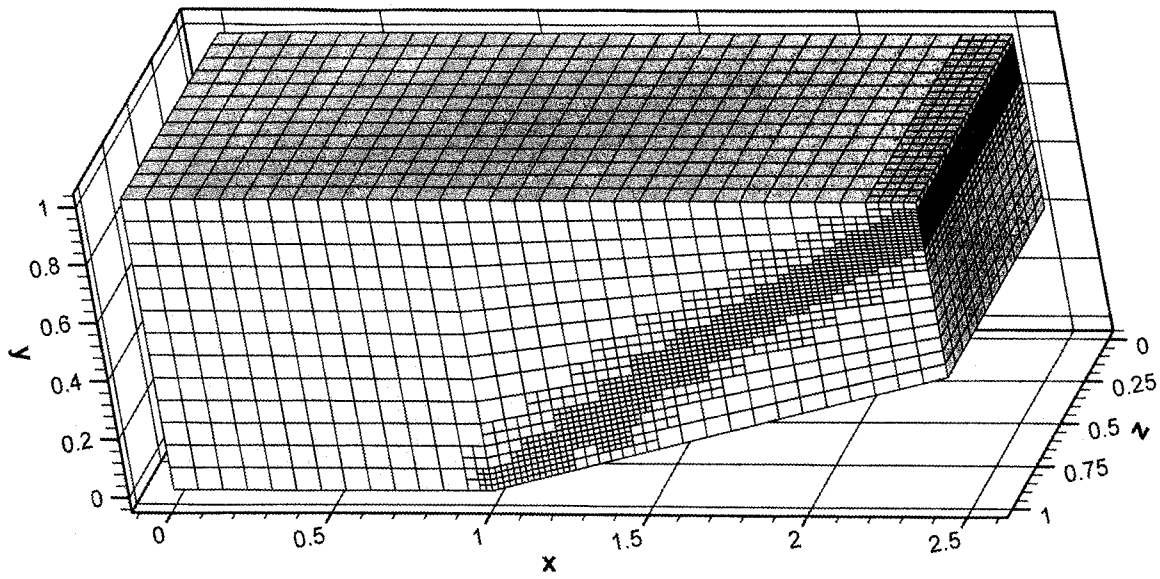
Figure 5-6. Converged mesh for three-dimensional 15° compression corner at $M_\infty = 3$ with two levels of h-adaptation.

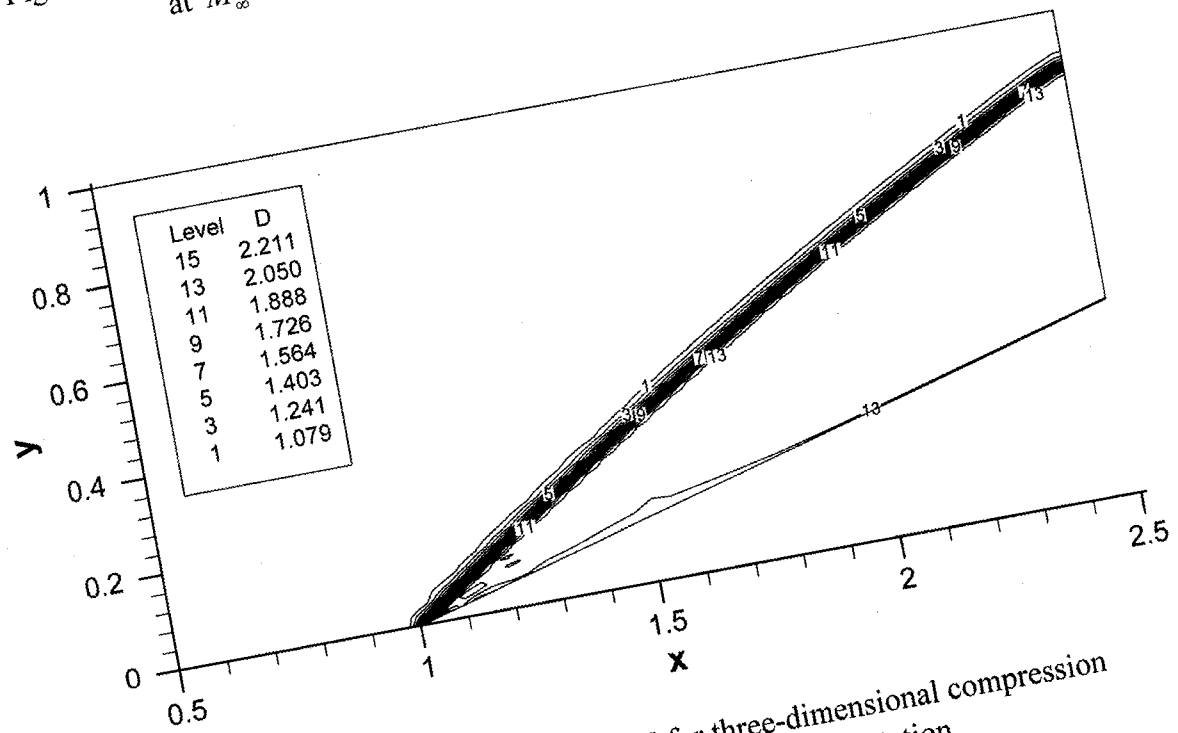| Level	D
15	2.211
13	2.050
11	1.888
9	1.726
7	1.564
5	1.403
3	1.241
1	1.079

Figure 5-7. Density contours at $z = 0.5$ for three-dimensional compression corner at $M_\infty = 3$ with two levels of h-adaptation.
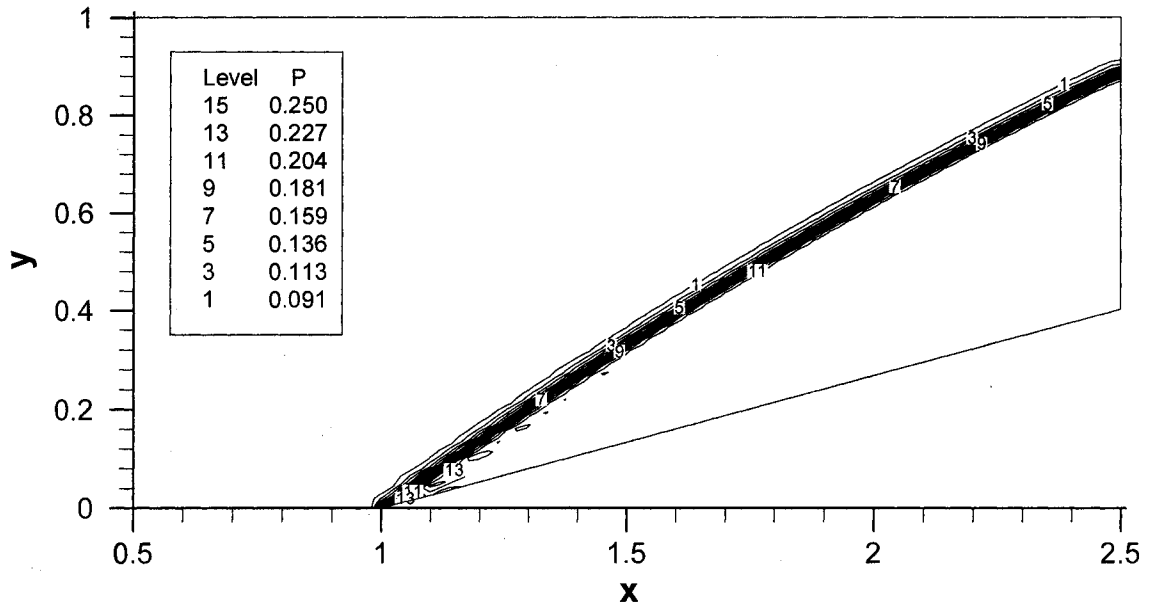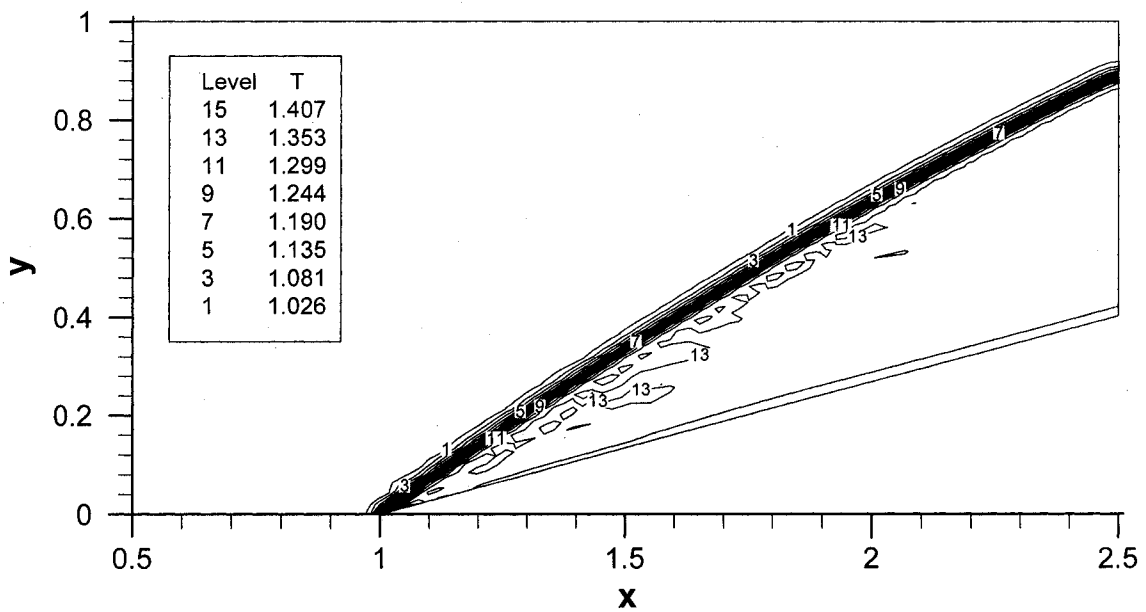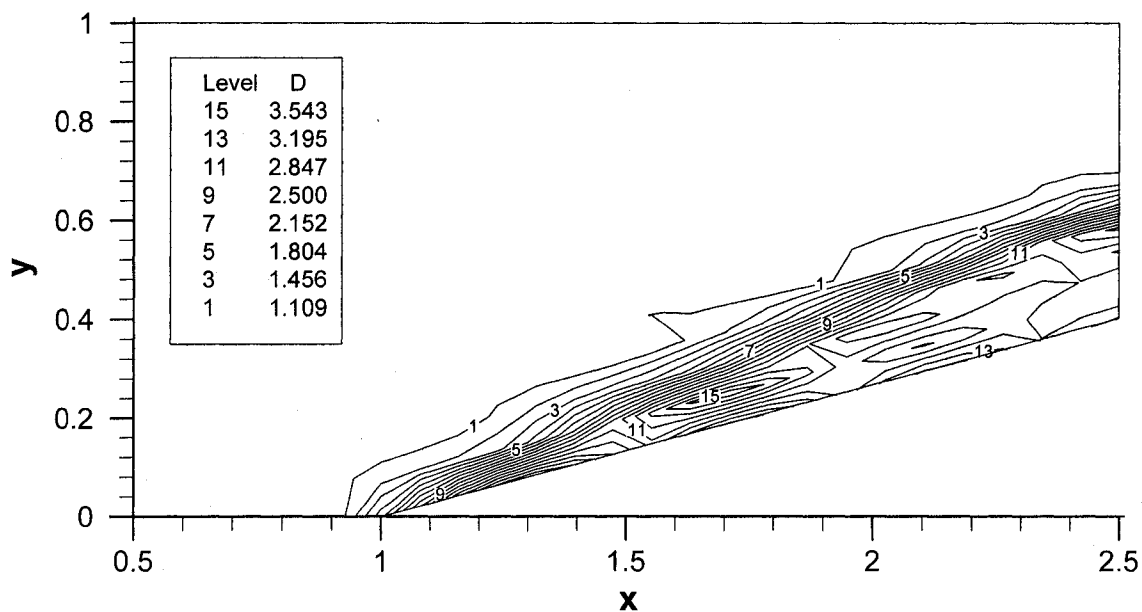
Figure 5-8. Pressure contours at $z = 0.5$ for three-dimensional compression
corner at $M_\infty = 3$ with two levels of h-adaptation.

Figure 5-9. Temperature contours at $z = 0.5$ for three-dimensional compression
corner at $M_\infty = 3$ with two levels of h-adaptation.

Compression Corner Results without Adaptation for $M_\infty = 5$

As in the previous case without adaptation the mesh used here is the same one depicted in Figure 5-2. Density, pressure and temperature contours along the central z-plane are presented in Figures 5-10 through 5-12.

The shock wave is diffused over several element widths and is observed at a $24°$ angle. Since the space behind the shock is smaller than the Mach 3 case, it is even more difficult to adequately capture the high density, pressure and temperature gradients on the coarse mesh, resulting in even greater irregularities.

Figure 5-10. Density contours at $z = 0.5$ for three-dimensional compression corner at $M_\infty = 5$ without adaptation.
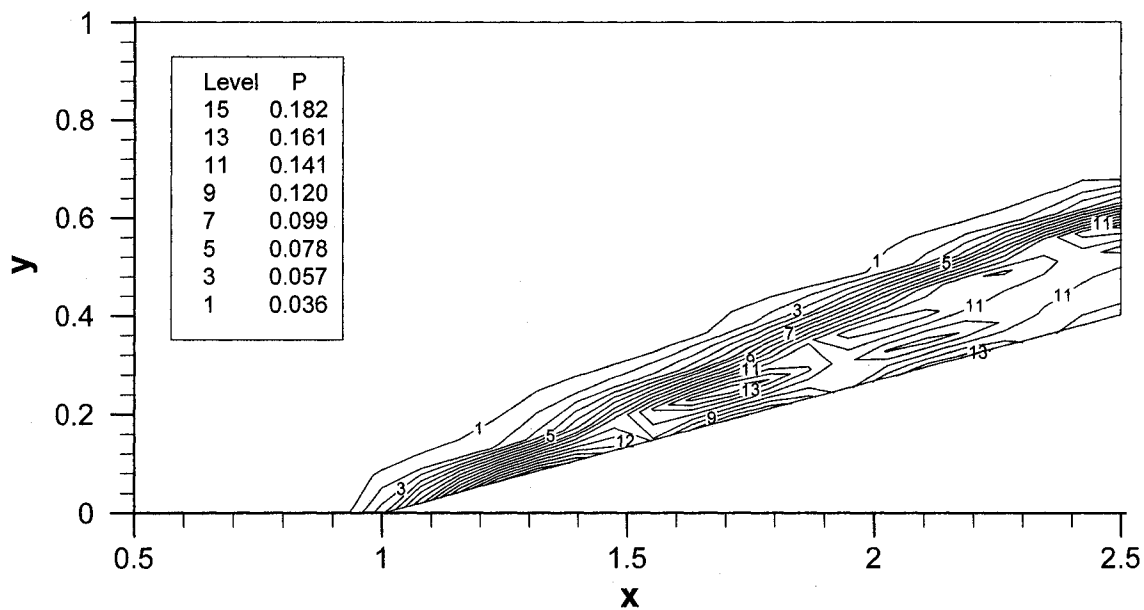
Figure 5-11. Pressure contours at $z = 0.5$ for three-dimensional compression corner at $M_\infty = 5$ without adaptation.
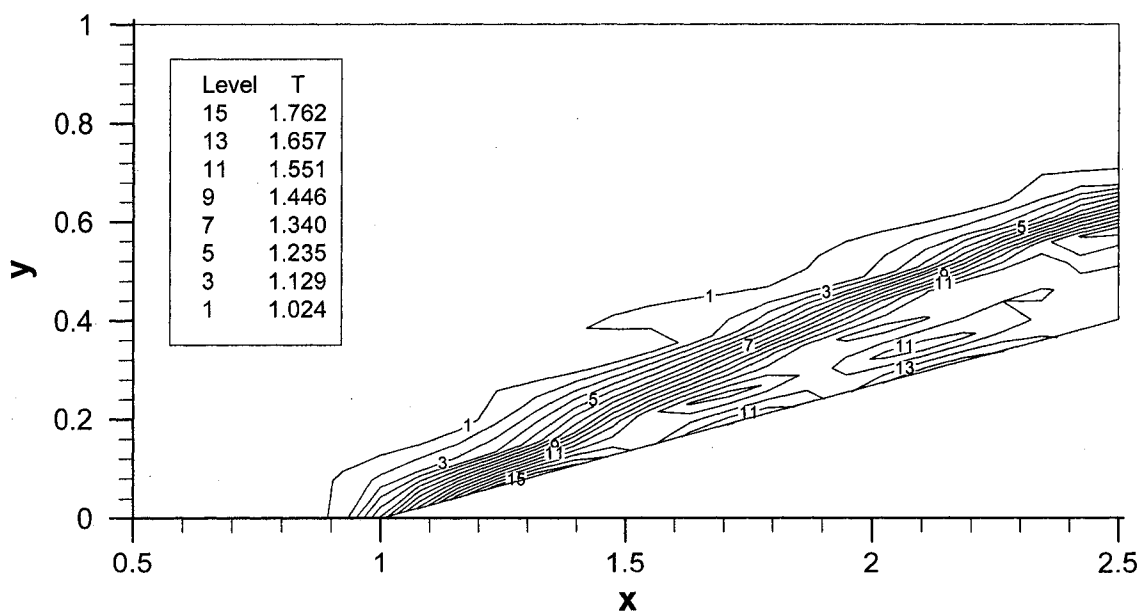

Figure 5-12. Temperature contours at $z = 0.5$ for three-dimensional compression corner at $M_\infty = 5$ without adaptation.

Compression Corner Results with Two Levels of Adaptation for $M_\infty = 5$

For this particular case, the converged mesh with two levels of adaptation is shown in Figure 5-13. It shows 41,704 elements and 44,166 nodes. Density, pressure and temperature contours along the central z-plane are presented in Figures 5-14 through 5-16. While the shock wave is still diffused over several element widths, its exact location is more evident, and is observed at a $24°$ angle. Irregularities around the shock are reduced.
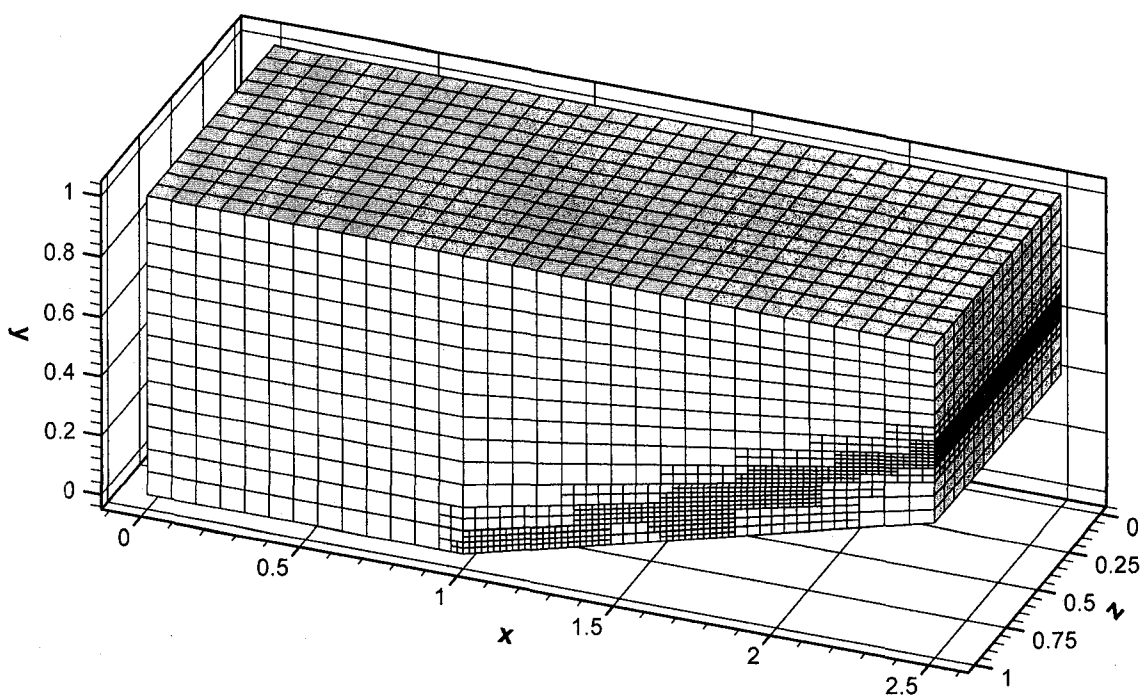

Figure 5-13. Converged mesh for three-dimensional $15°$ compression corner at $M_\infty = 5$ with two levels of h-adaptation.
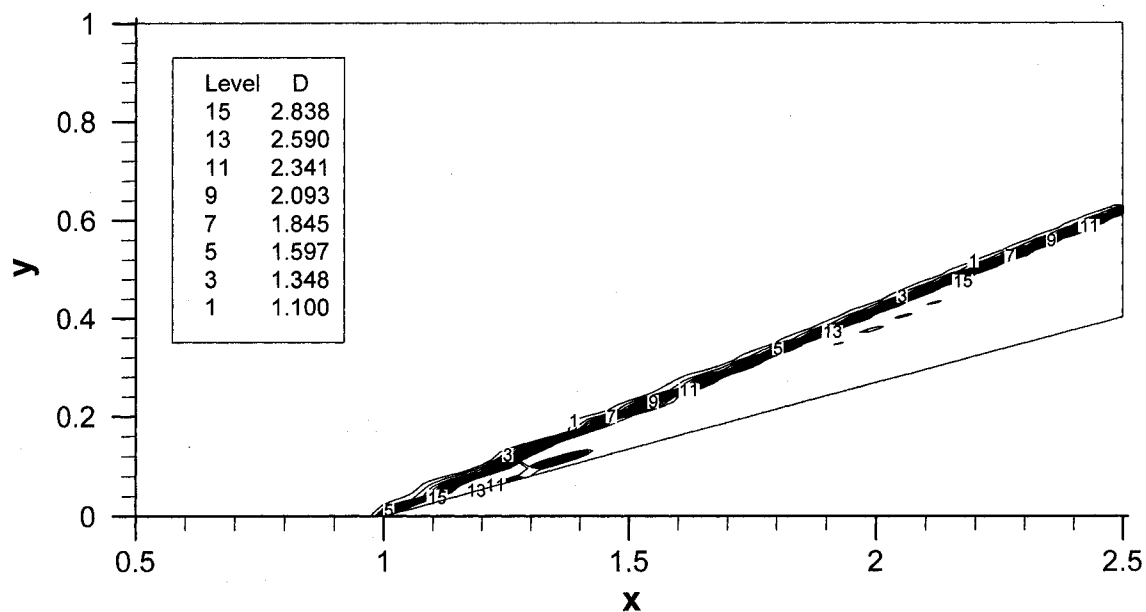
Figure 5-14. Density contours at $z = 0.5$ for three-dimensional compression
corner at $M_\infty = 5$ with two levels of h-adaptation.
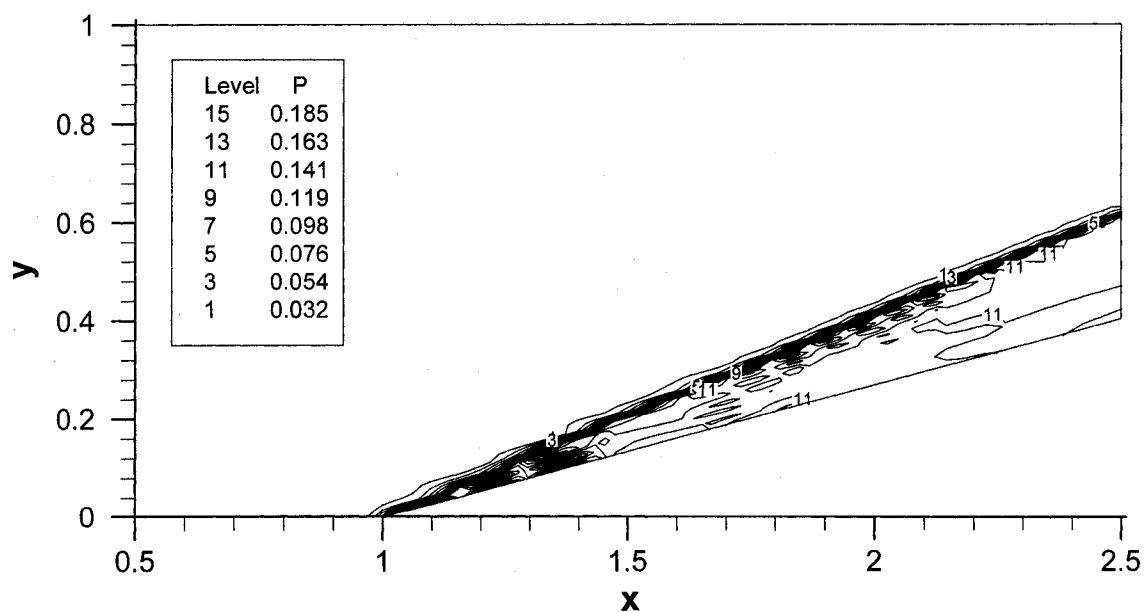

Figure 5-15. Pressure contours at $z = 0.5$ for three-dimensional compression
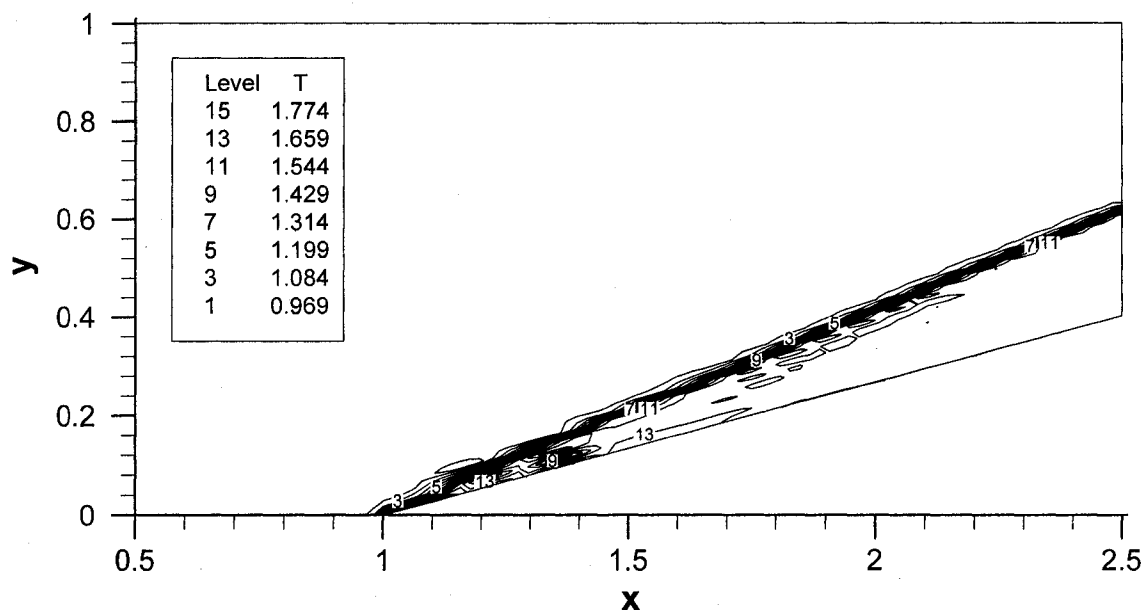corner at $M_\infty = 5$ with two levels of h-adaptation.

Figure 5-16. Temperature contours at $z = 0.5$ for three-dimensional compression
corner at $M_\infty = 5$ with two levels of h-adaptation.

Comparison of Euler Solutions with Theoretical Data

Since the geometry of the above test cases is constant along the z-axis, it is not

unreasonable to relate central z-plane data to two dimensional theoretical solutions. The

data presented above can be compared with the two-dimensional oblique shock relations

for a calorically perfect gas (Anderson, 1990). Comparisons of computed and theoretical

density ratios for all cases are presented in Table 5-1. Comparisons of computed and

theoretical pressure ratios are presented in Table 5-2. Comparisons of computed and

theoretical temperature ratios are presented in Table 5-3. The error in all cases without

adaptation ranges from approximately 1% to 16%. The error in all cases with adaptation

ranges from approximately 1% to 2%, which indicates that the Euler model with h-

adaptation performs very well.

Table 5-1. Comparison of density ratio.

M_∞	Adaptation	$\dfrac{\rho_2}{\rho_1}$ computed	$\dfrac{\rho_2}{\rho_1}$ theoretical	% error
3	None	2.04	2.03	0.49
3	2 levels	2.05	2.03	0.99
5	None	3.20	2.75	16.4
5	2 levels	2.80	2.75	1.82

Table 5-2. Comparison of pressure ratio.

M_∞	Adaptation	$\dfrac{p_2}{p_1}$ computed	$\dfrac{p_2}{p_1}$ theoretical	% error
3	None	2.73	2.82	3.19
3	2 levels	2.76	2.82	2.14
5	None	5.01	4.80	4.38
5	2 levels	4.71	4.80	1.88

Table 5-3. Comparison of temperature ratio.

M_∞	Adaptation	$\dfrac{T_2}{T_1}$ computed	$\dfrac{T_2}{T_1}$ theoretical	% error
3	None	1.36	1.39	2.16
3	2 levels	1.37	1.39	1.44
5	None	1.72	1.74	1.15
5	2 levels	1.77	1.74	1.72

Three Dimensional Viscous Flow Over a Flat Plate

Carter was the first to calculate a numerical solution to the two-dimensional flow over an isothermal flat plate (Carter, 1972). Since then many people have used Carter's solution as a benchmark for both two-dimensional and three-dimensional code validation (Brueckner and Pepper, 1995; Devloo et al., 1988).
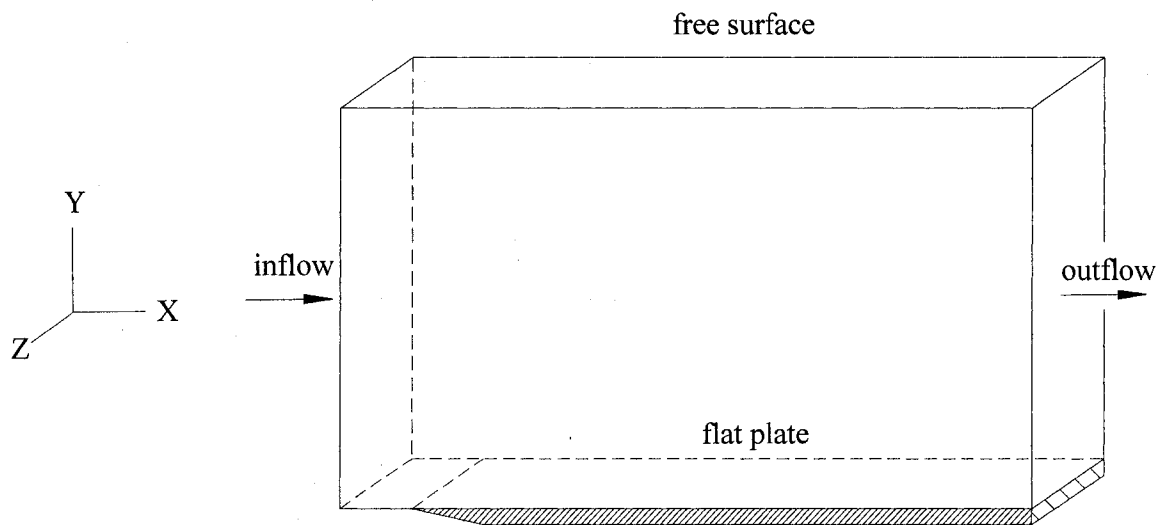

Figure 5-17. Boundary conditions for three-dimensional viscous compressible flow over a flat plate.

The geometry and general boundary conditions are shown in Figure 5-17. The leading edge of the flat plate is located at $x = 0.1$ and the outflow boundary is located at $x = 1.1$. The domain extends 1 unit in the z-direction and 0.75 units in the y-direction. The nodes at the inflow boundary are set according to the following dirichelet conditions:

$$u_{in} = U_\infty$$

$$v_{in} = w_{in} = 0$$

$$\rho_{in} = \rho_\infty$$

$$T_{in} = T_{\infty} = 205 \text{ K}$$

$$M_{in} = M_{\infty} = 3.0$$

Inflow dirichelet conditions are also imposed on the free surface boundary. However, this only works if the free surface boundary is placed far enough away, so that the leading-edge shock does not intersect it. On the surface of the flat plate:

$$u_{wall} = v_{wall} = w_{wall} = 0$$

$$T_{wall} = T_s$$

where T_s is the dimensionless stagnation temperature (Brueckner and Pepper, 1995):

$$T_s = 1 + \left[(\gamma - 1)/2 \right] M_{\infty}^2 \qquad (5.7)$$

In order to simulate an infinitely long flat plate, the velocity values at the outflow boundary nodes are set equal to the values of the interior adjacent nodes.

The following fluid properties for air are used:

$$\text{Pr} = 0.72$$

$$\gamma = 1.4$$

$$\text{Re} = 10^3$$

The mesh used for this case is presented in Figure 5-18. This mesh consists of 28,080 elements with 32,400 nodes and is biased toward the flat plate in order to accurately capture the boundary layer. Density, pressure, temperature, and mach contours along the central z-plane are presented in Figures 5-19 through 5-22. The velocity profile along the central z-plane is presented in Figure 5-23.

When the flow encounters the flat plate, both a shock wave and a boundary layer are spawned at the leading edge. Based on work by Carter, the boundary layer should evolve

such that the boundary layer edge is located between $y = 0.20$ and $y = 0.25$ at the outlet. Likewise, the leading-edge shock should be located between $y = 0.42$ and $y = 0.57$ at the outlet. The calculated contours of density, pressure, and temperature are in agreement with those obtained by Brueckner and Pepper (1995) and Devloo, et al. (1988).

Figure 5-20 does show some oscillation in pressure values near the flat plate, which is more prominent at the leading edge and weakens downstream. It appears to be caused by insufficient resolution of the extremely high pressure gradients that occur at the leading edge of the plate. Carter experienced the same oscillations in wall pressure in his study when using the continuity equation to solve for the density at the flat plate. He noted that with finer grids the oscillations are smaller and dampen out further upstream. Carter also noted that a quadratic extrapolation of the density in the in the y-direction at the flat plate significantly reduced the oscillations.
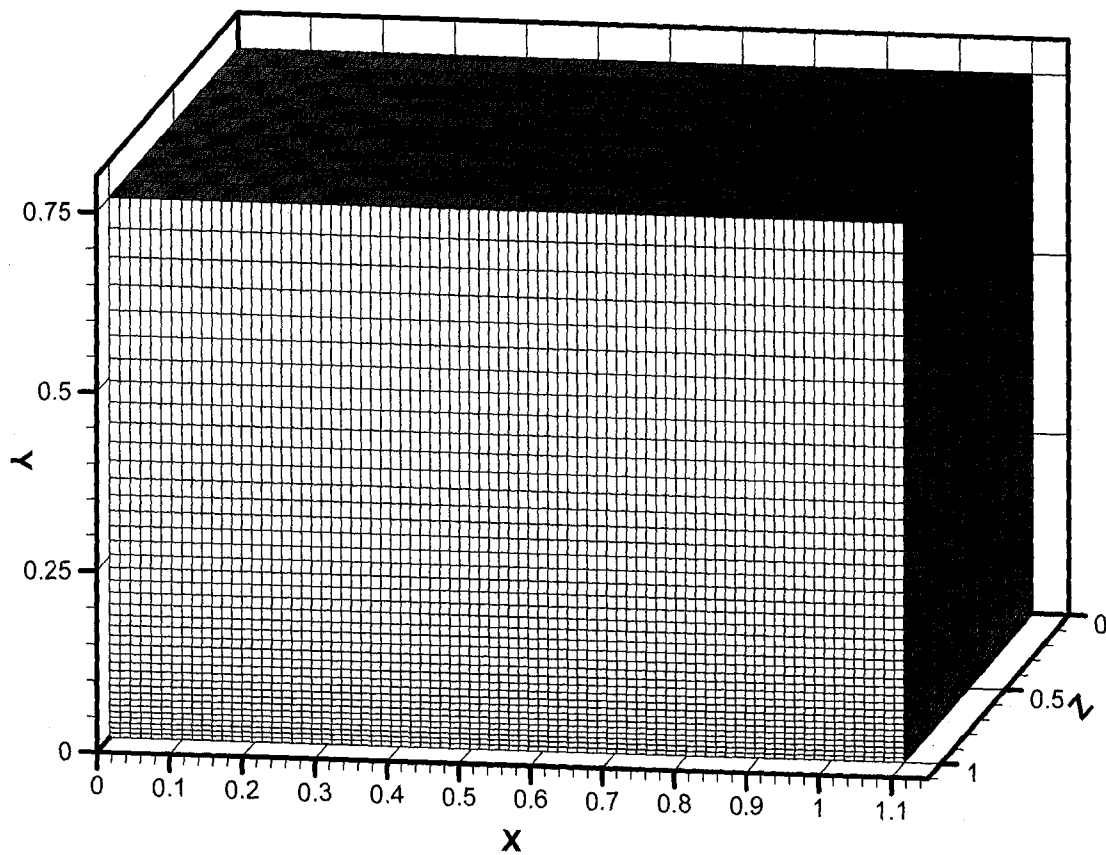
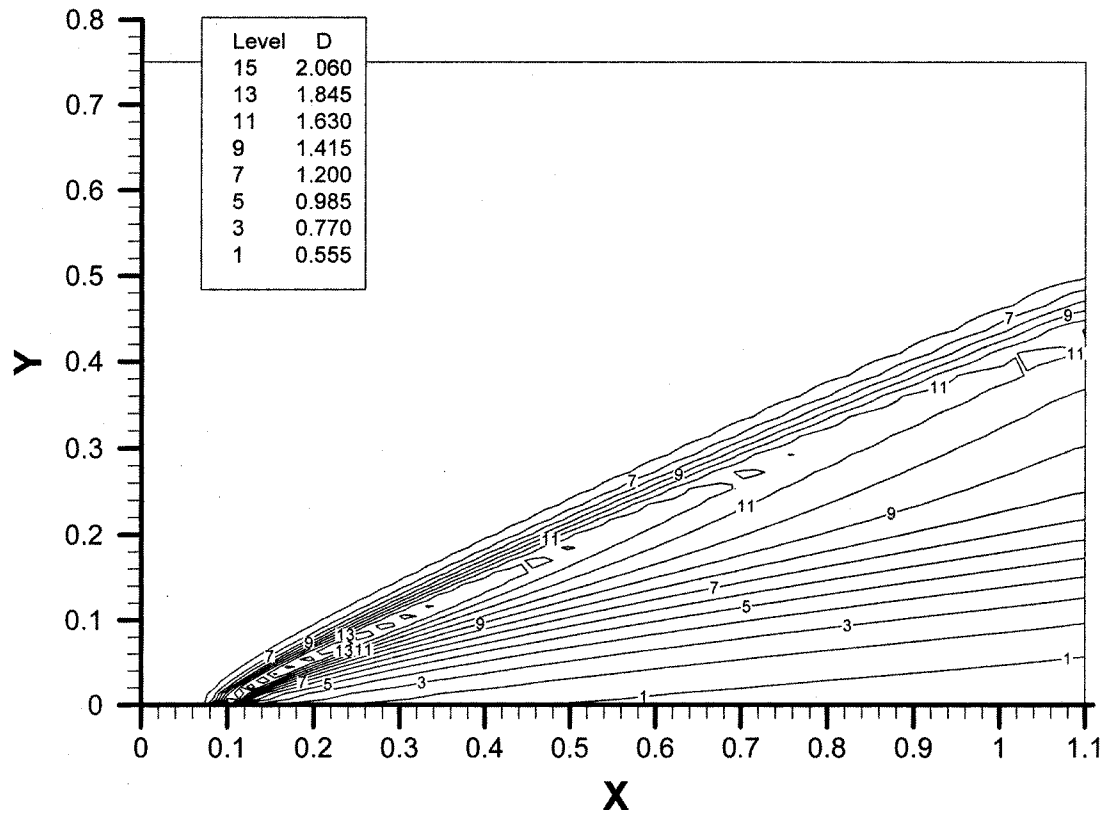Figure 5-18. Three-dimensional mesh for viscous compressible flow over a flat plate.

Figure 5-19. Density contours at $z = 0.5$ for three-dimensional viscous compressible flow over a flat plate.
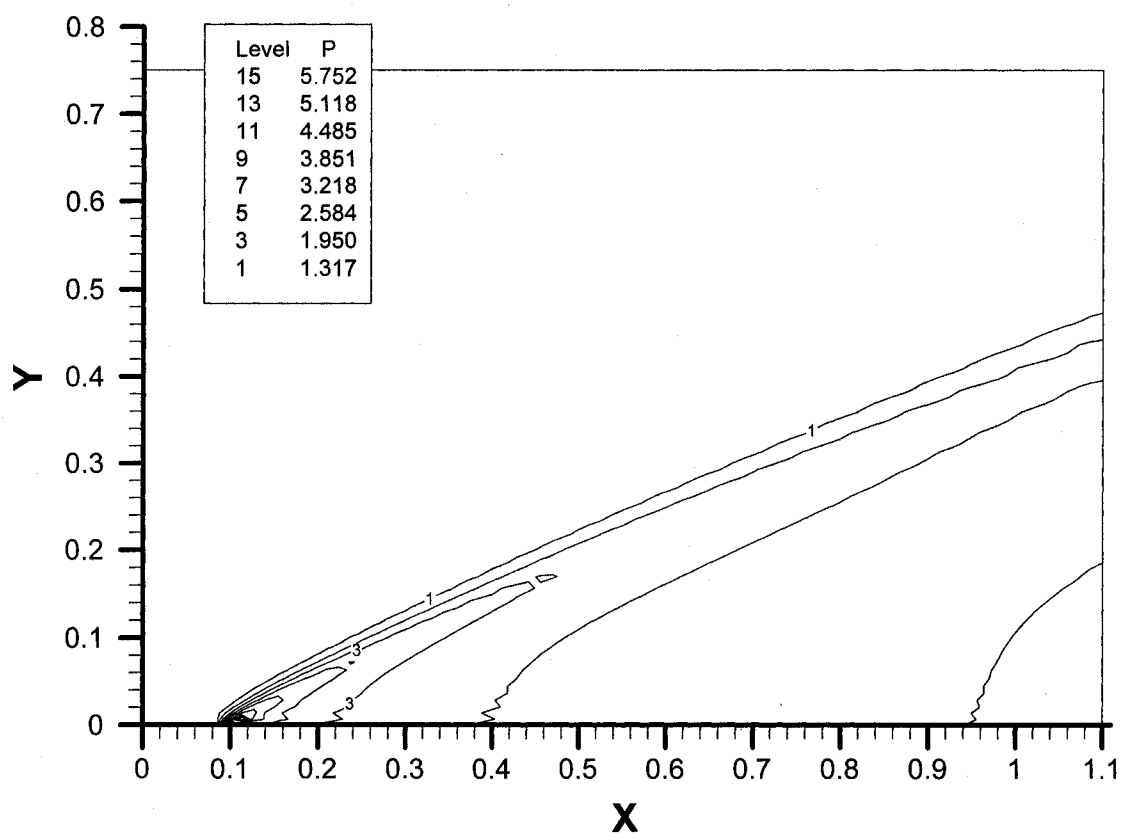
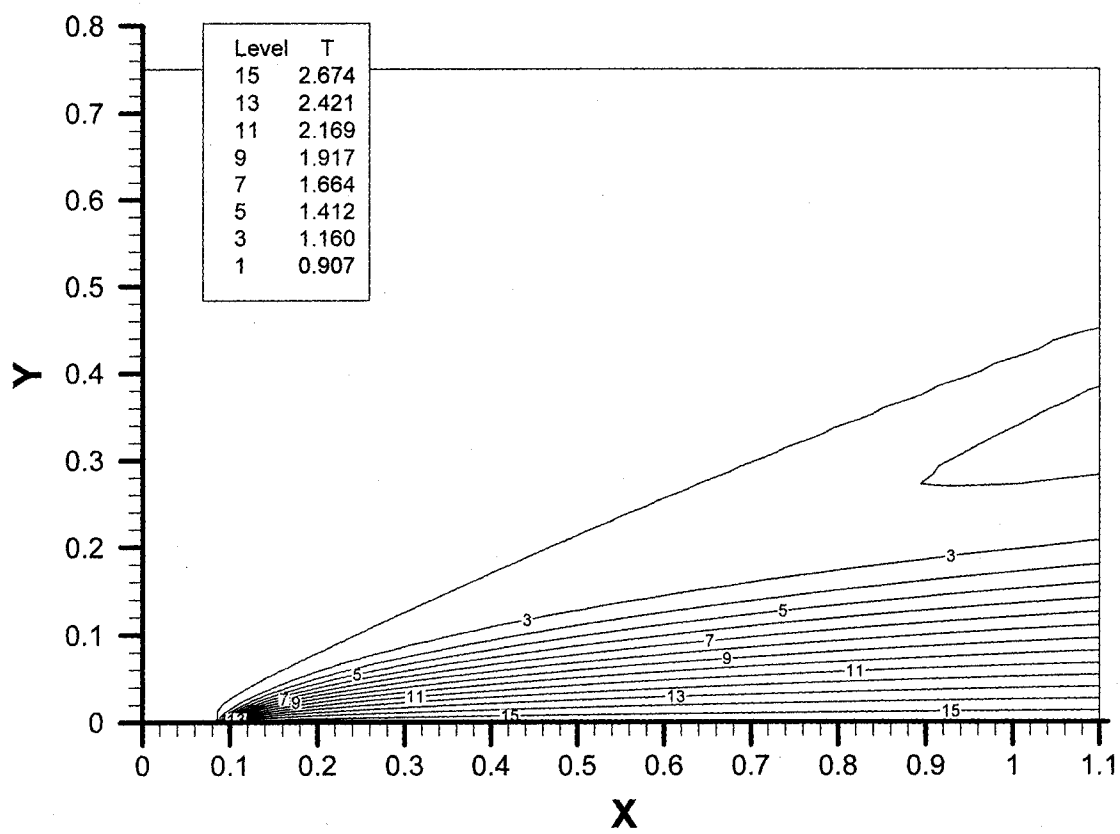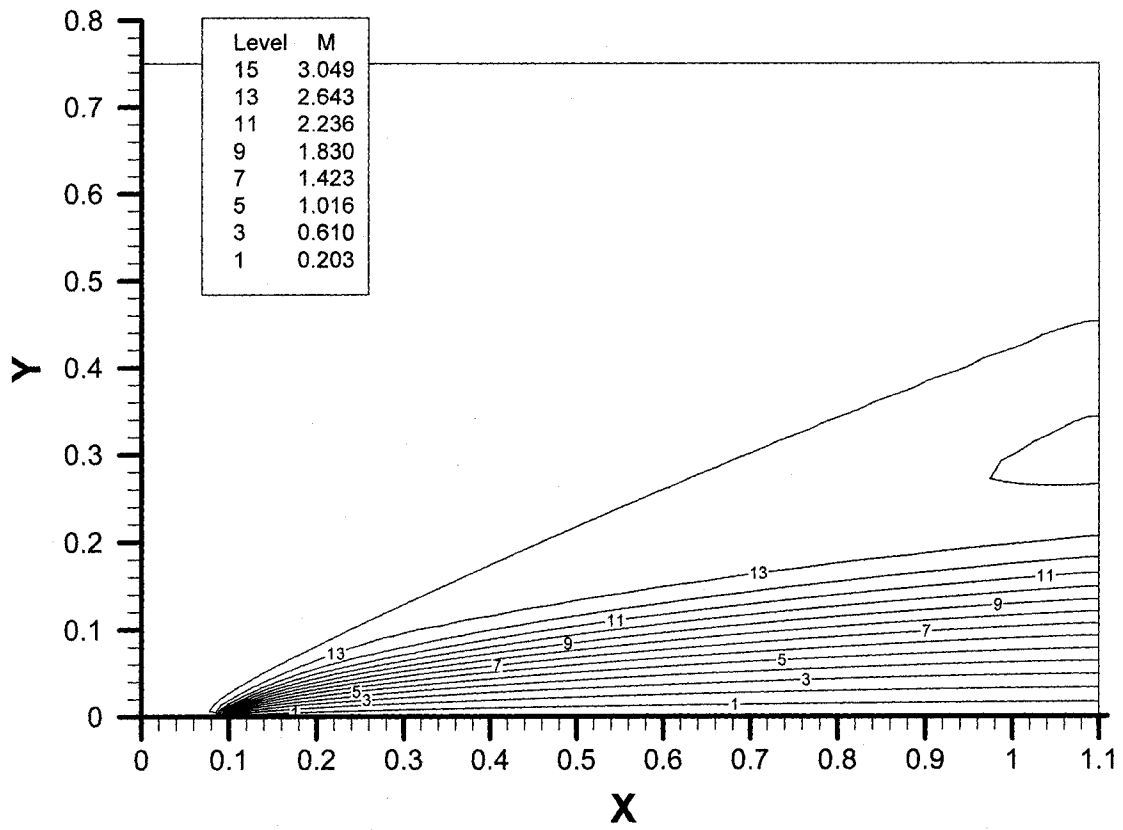Figure 5-20. Pressure contours at $z = 0.5$ for three-dimensional viscous compressible flow over a flat plate.

Level	T
15	2.674
13	2.421
11	2.169
9	1.917
7	1.664
5	1.412
3	1.160
1	0.907

Figure 5-21. Temperature contours at $z = 0.5$ for three-dimensional viscous compressible flow over a flat plate.

Figure 5-22. Mach contours at $z = 0.5$ for three-dimensional viscous compressible flow over a flat plate.
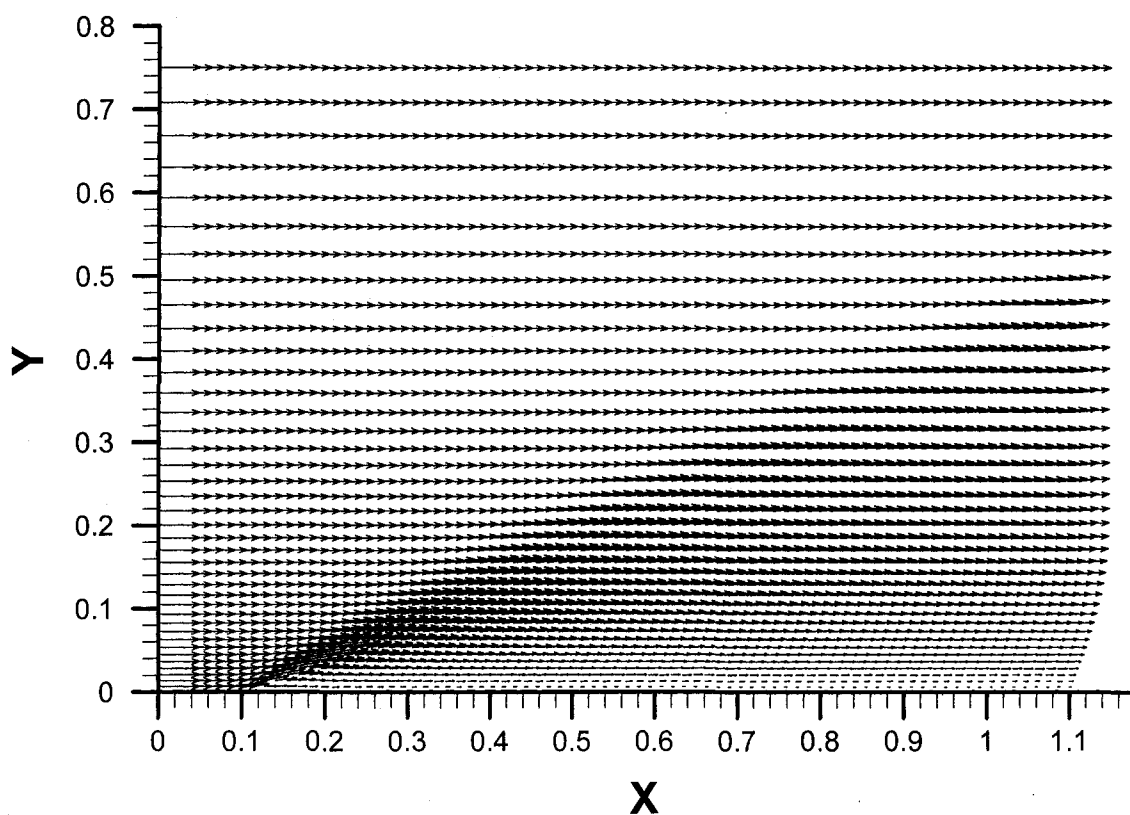
Figure 5-23. Velocity profile at $z = 0.5$ for three-dimensional viscous compressible flow over a flat plate.

Figures 5-24 through 5-27 show profiles of various flow properties at the outflow boundary calculated by the current finite element model along the central z-plane, along with a couple of two-demensional finite difference cases performed by Carter. Carter solved this problem using several different grids. The fine grid results shown were obtained by Carter using a grid spacing of 0.015 fixed in the y-direction and varying from 0.015 to 0.020 in the x-direction. The coarse grid results shown were obtained by Carter using a grid spacing of 0.05 fixed in both the x-direction and y-direction. The three-dimensional results obtained in this study were performed on a mesh with a 0.0135 fixed element length in the x-direction, a fixed element length of 0.10 in the z-direction and a variable element length in the y-direction biased toward the $y = 0$ plane, with a range of 0.01 to 0.05.

Looking at Figure 5-24, it is evident that the edge of the boundary layer is located at $y = 0.25$. Within the boundary layer, the plotted results at the outlet calculated from the current model agree more with the fine-grid results of Carter. This is to be expected, since it is in this region where the y-spacing of the three-dimensional mesh is more closely matched with the fine grid used by Carter.

Looking at Figures 5-25 and 5-26, Carter locates the leading-edge shock at $y = 0.57$ on the fine-grid, and at $y = 0.42$ on the coarse-grid. The results obtained by the current three-dimensional model locate the leading-edge shock at $y = 0.44$, indicating better agreement with Carter's coarse-grid solution. This is not surprising, since it is in this region where the y-spacing of the three-dimensional mesh is more closely matched with the coarse grid used by Carter.
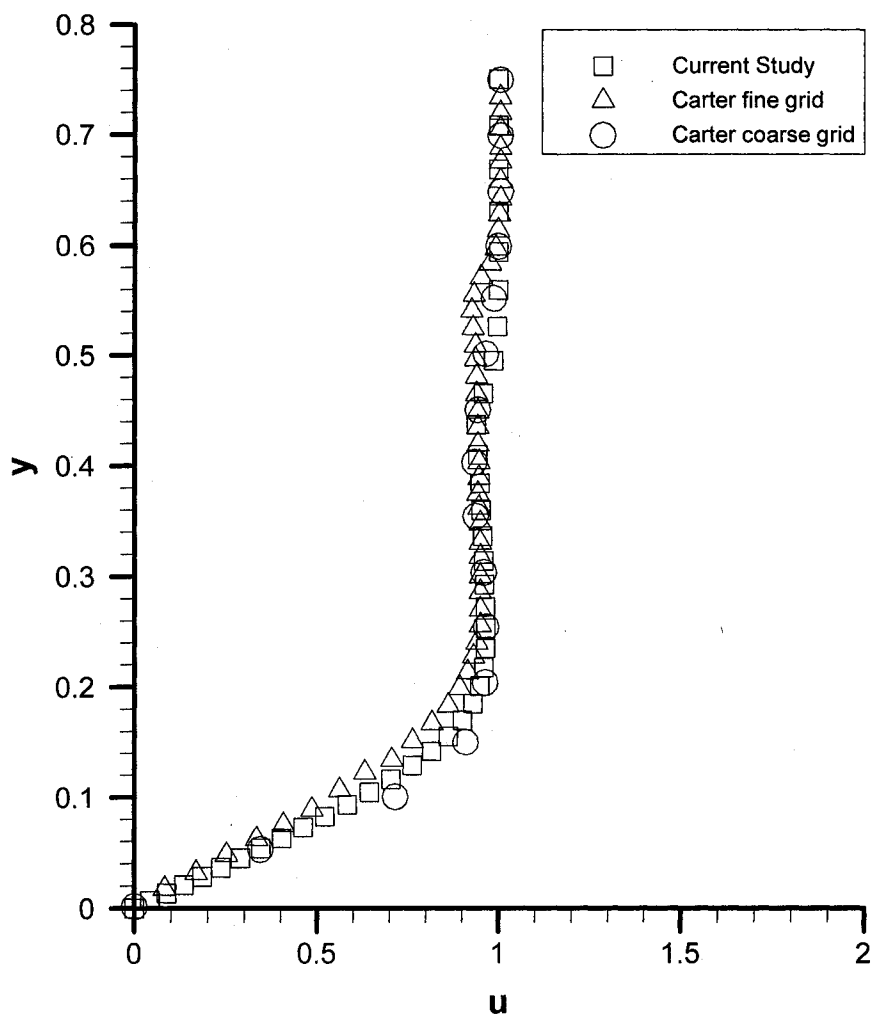
Figure 5-24. Comparison of u values at the outflow boundary between the results of Carter and current finite element results at $z = 0.5$.
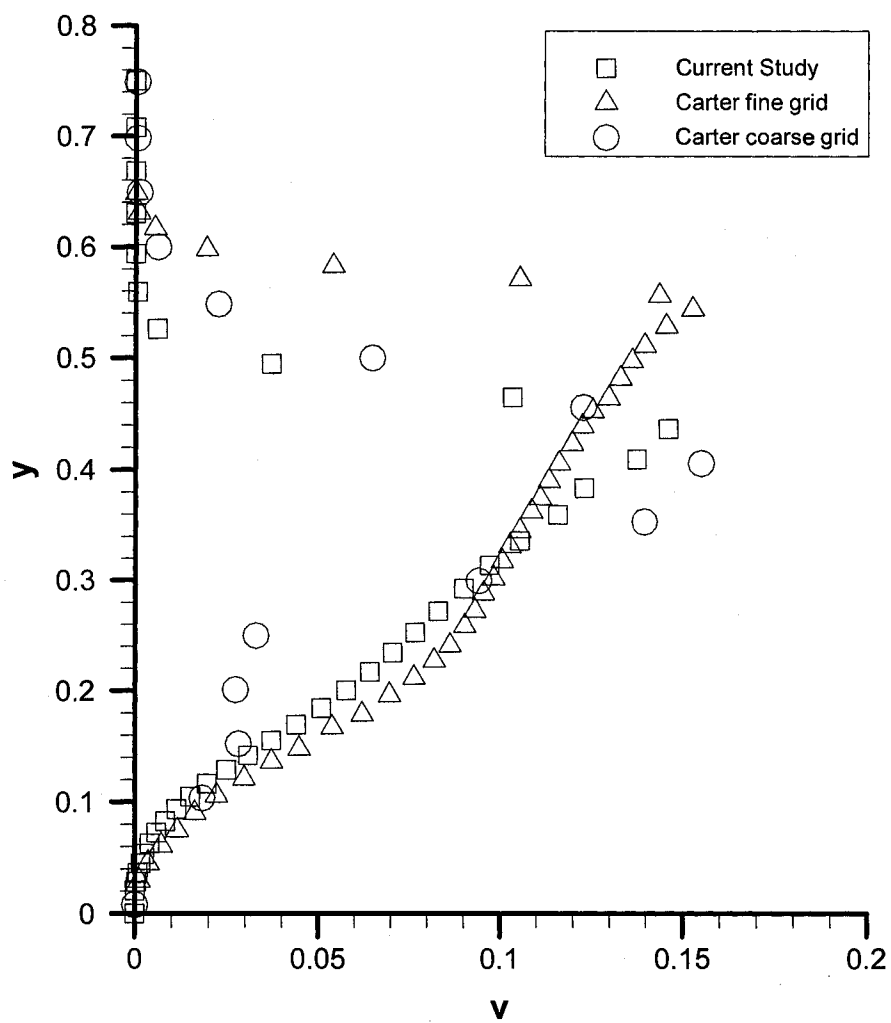
Figure 5-25. Comparison of *v* values at the outflow boundary between the results of Carter and current finite element results at *z* = 0.5.
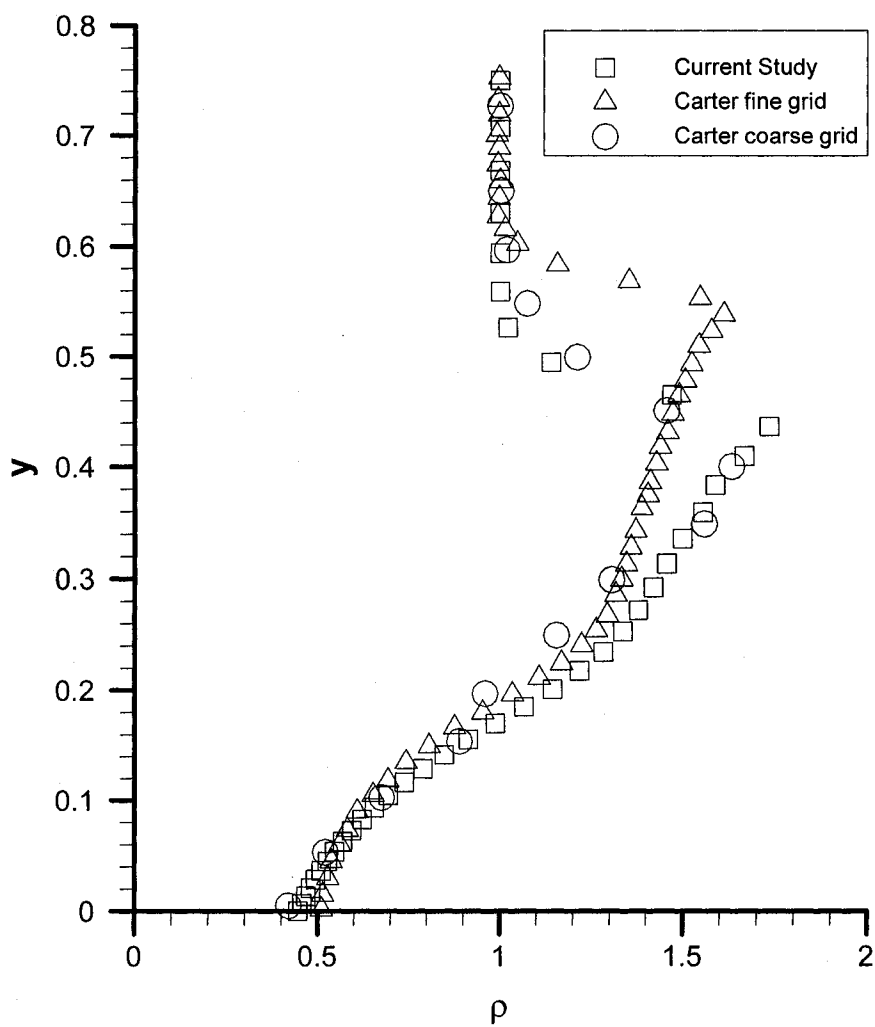
Figure 5-26. Comparison of density values at the outflow boundary between the results of Carter and current finite element results at $z = 0.5$.
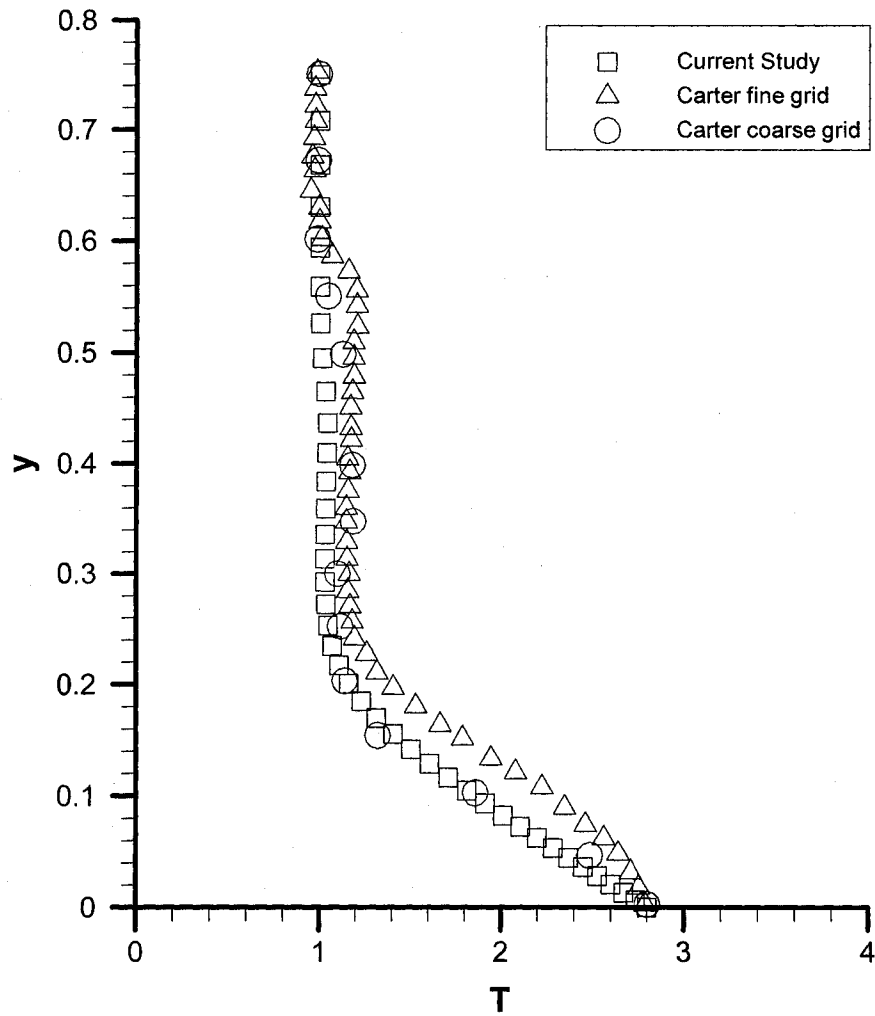
Figure 5-27. Comparison of temperature values at the outflow boundary between the results of Carter and current finite element results at $z = 0.5$.

CHAPTER 6

APPLICATION TO LIGHT GAS GUN DESIGN

The first light gas gun was developed due to the need to achieve higher projectile

velocities. It was determined that higher muzzle velocities could be achieved if the

column of conventional powder gas driving the projectile was replaced with a light-

weight gas such as hydrogen (Crozier and Hume, 1957). Since then, single-stage, two-

stage, and three-stage light gas guns have been used for hypervelocity impact studies

(Schonberg and Cooper, 1994) and equation of state experiments (Nellis, et al., 1991).

When impacted by a high-velocity projectile, strong shock waves are generated in a

target specimen. Equation of state data for the target material can then be obtained using

a method based on the Rankine-Hugoniot equations (Mitchell and Nellis, 1981).

The Joint Actinide Shock Physics Experimental Research (JASPER) facility utilizes a

two-stage light gas gun to conduct equation of state experiments (Braddy, et al., 2001).

Figure 6-1 illustrates the major components of the JASPER light gas gun. The pump tube

is 11.5 meters long with a bore diameter of 89 mm and a piston mass of 4.5 kg. The

launch tube is 8.1 meters long with a bore diameter of 28 mm. Hydrogen is used to

propel projectiles with a mass range of 16.5 g to 26.5 g to a velocity of 7.4 km/s. The

projectiles are cylindrical in shape, with a diameter of approximately 28 mm and a length

of 25.4 mm.

72
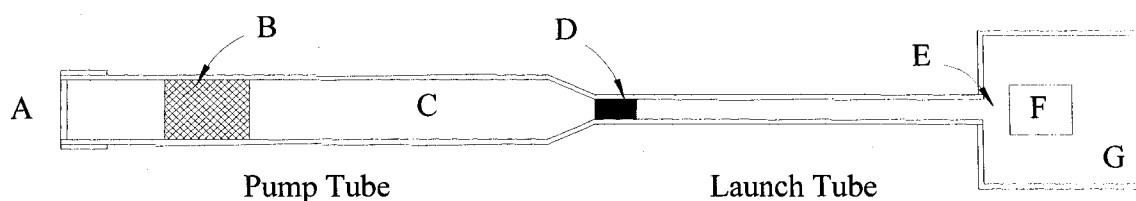
A	Breech
B	Piston
C	Hydrogen Gas
D	Projectile
E	Free Flight Zone
F	Primary Target Chamber
G	Secondary Containment Chamber

Figure 6-1. Diagram of JASPER light gas gun.

An explosive charge is loaded into the breech behind the piston. The remaining section of the pump tube in front of the piston is filled with hydrogen gas. When a shot is fired, the explosive charge is detonated, sending the piston down the pump tube, compressing the hydrogen. When the hydrogen reaches a pressure of approximately 400 bar (Mespoulet, 2001), a petal valve separating the pump tube and launch tube ruptures, allowing the compressed hydrogen to propel the projectile down the launch tube toward the target.

Due to the hazardous nature of the experiments, the target is placed within the primary target chamber. The primary target chamber is equipped with an explosively driven ultra fast closure valve to contain any debris resulting from the projectile impacting with the target. The primary target chamber is placed inside the secondary containment chamber, which is designed to contain hydrogen deflagration and provide

containment should the primary target chamber fail. Before the shot, a vacuum is pulled on the secondary containment chamber and launch tube.

From the muzzle exit to the entrance of the primary target chamber the projectile encounters a free flight zone approximately 1 meter in length. What the projectile does in this free flight zone is of particular interest in this study. Ideally, the projectile should impact the target with no tilt in the axial direction, ensuring that the shock propagates through the target as uniformly as possible. This work is focused on determining what effects, if any, launch tube exit geometry changes have on attitude of the projectile in flight.

At the muzzle exit, a muzzle protector is attached to guard against debris. Depending on the configuration of the muzzle protector, the geometry of the muzzle may be altered. Two different configurations of this muzzle protector are under consideration (see Figure 6-2). The first case is standard muzzle geometry where the wall of the bore and the outer surface of the launch tube form a 90 degree angle. The second case includes a 26.6 degree bevel transition from the wall of the bore to the outer surface of the launch tube. For both cases, solutions are calculated for several positions downstream of the launch tube exit.
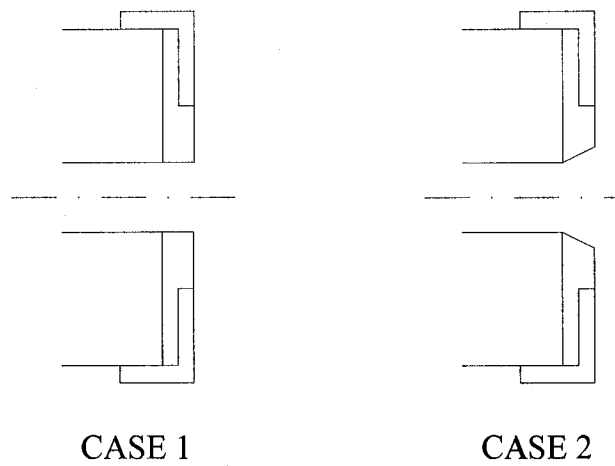
CASE 1   CASE 2

Figure 6-2. Cross-section of muzzle exit showing attached protectors.

Boundary Conditions

The computational domains for both cases are illustrated in Figure 6-3. The exit plane of the muzzle is defined as $x = 0$. The nodes at the inlet boundary are set according to the following dirichelet conditions:

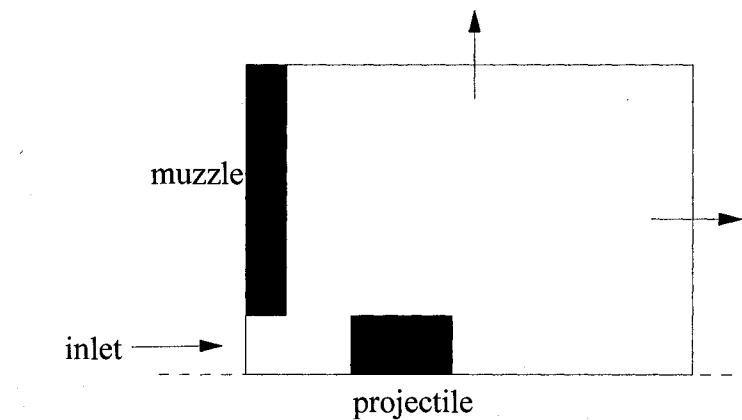$$u_{in} = U_\infty$$

$$v_{in} = 0$$
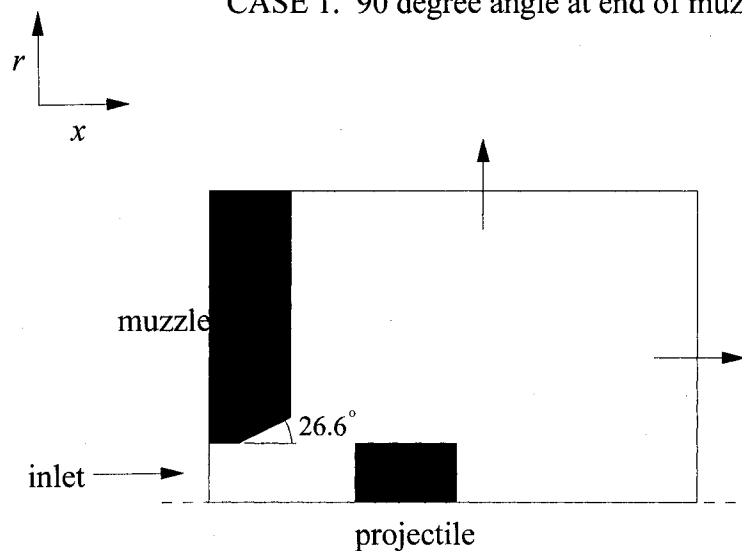
$$w_{in} = 0$$

$$T_{in} = T_\infty$$

$$M_{in} = 2.8 .$$

On the surfaces of the projectile and muzzle

$$u_{wall} = v_{wall} = w_{wall} = 0$$

The temperature ate the wall, T_{wall}, is set equal to the nondimensional stagnation temperature given by equation (5.7).

CASE 1. 90 degree angle at end of muzzle.

CASE 2. 26.6 degree angle at end of muzzle.

Figure 6-3. Axisymmetric representation of launch tube exit geometries.

It should be noted that all simulations were conducted in a quasi-steady state, meaning that at each location the projectile is held fixed while the flow field is calculated. In other words, when performing a simulation for a particular location of the projectile, results for prior locations are not taken into account. Hence, the current study is focused on indicating if changes in projectile attitude might occur, and not with quantifying the actual changes in projectile attitude.

Two-Dimensional Axisymmetric Results

The initial coarse mesh used for all cases is comprised of elements that are 1 mm square, or close to 1 mm square, depending on geometry. All calculations were conducted on the coarse mesh with 2 levels of h-adaptation (see Figures 6-4 and 6-5).

Pressure contours around the projectile located at $x = 4$, 16, 32, 48, and 64 mm are presented for both cases in Figures 6-6 through 6-15. The difference between the two cases is more evident closer to the launch tube. When comparing the two cases further away from the muzzle, there is little difference in the pressure contours behind the projectile. It would appear that the effects of the fluid expansion out of the muzzle are most prominent within approximately 35 mm of exit.

When comparing the pressure contours at each location from case to case, it is evident that the flow fields are different. However, there is no information indicating that one flow field would have more influence than the other with regard to the attitude of the projectile. Based on these two-dimensional results, a series of three-dimensional simulations were performed for both cases at projectile locations less than 35 mm.
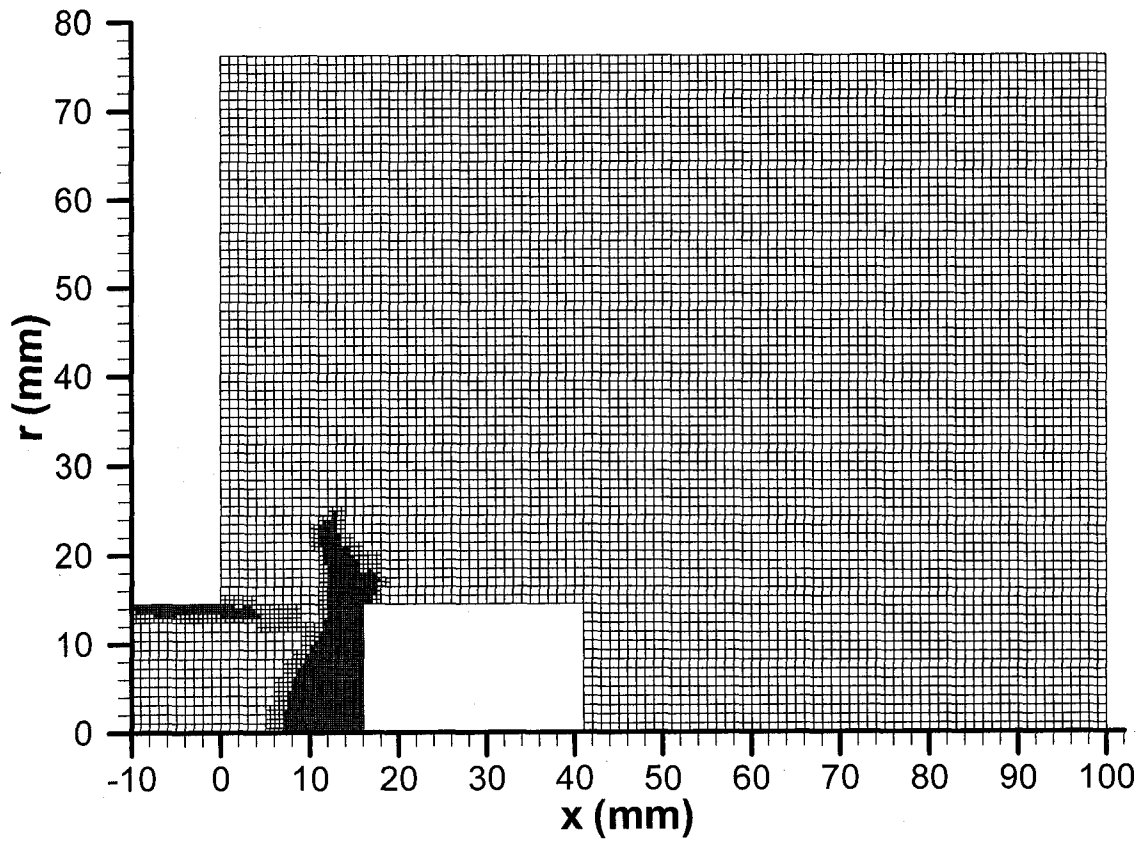
Figure 6-4. Example of axisymmetric case 1 mesh with two levels of h-adaptation.
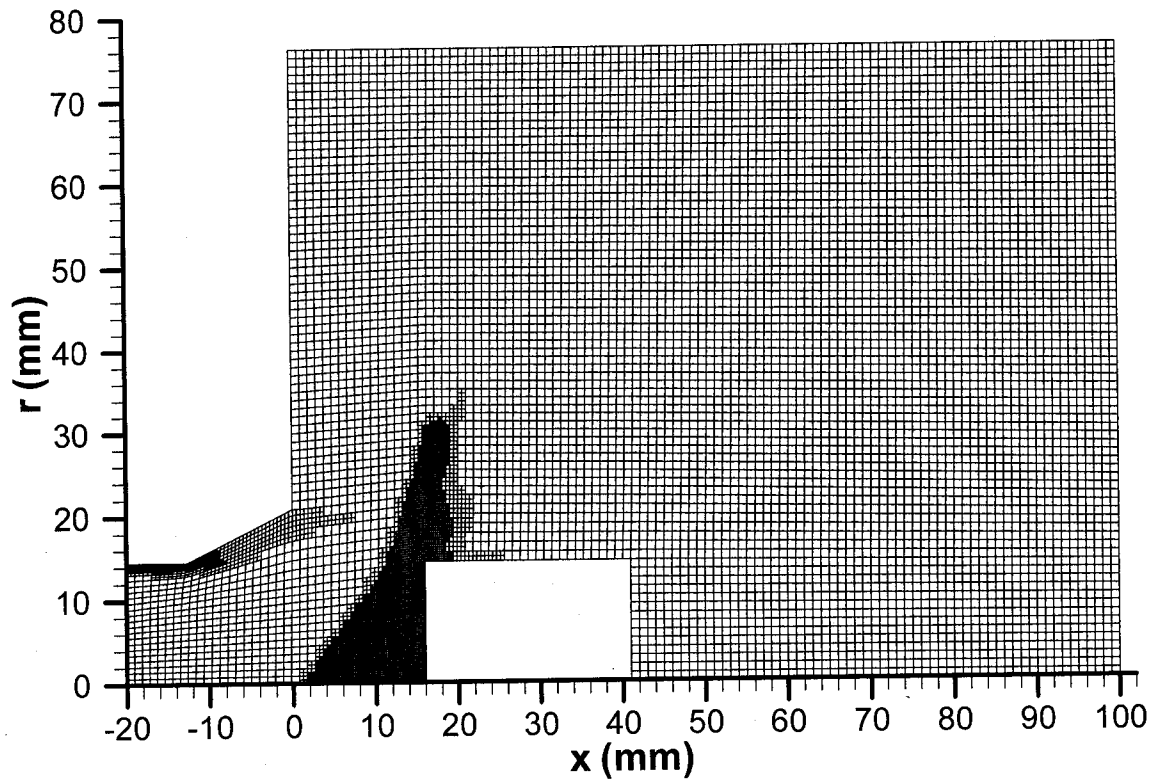
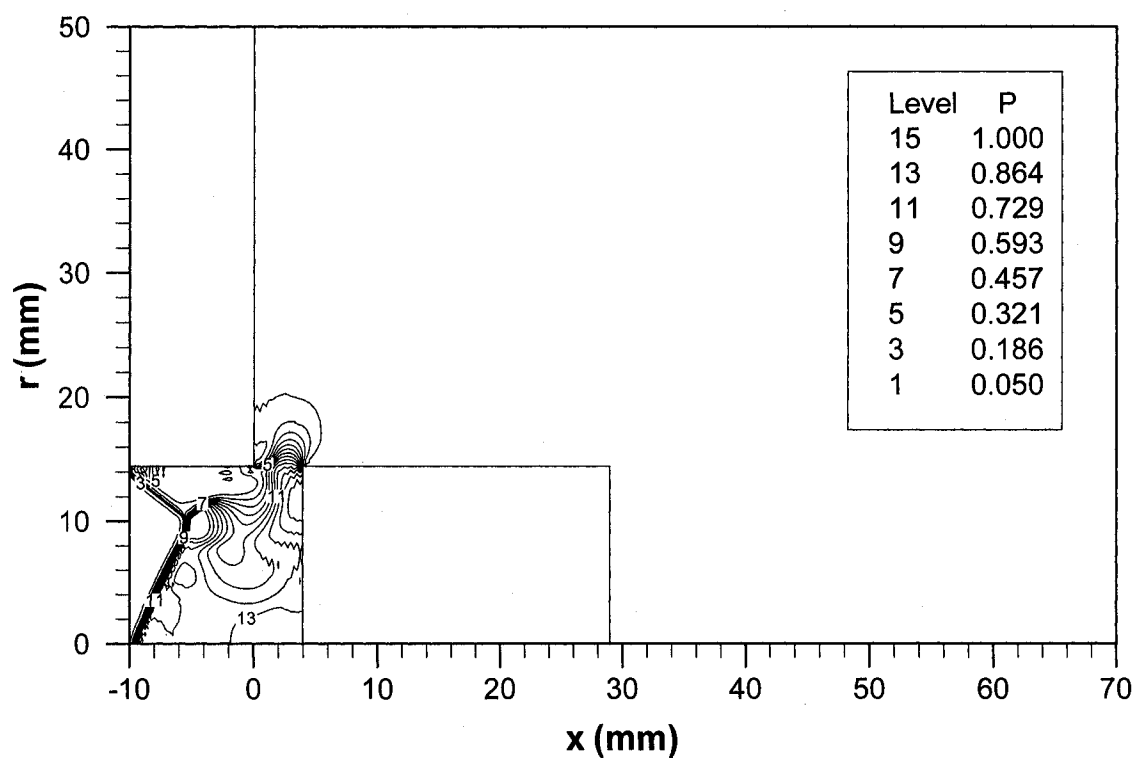Figure 6-5. Example of axisymmetric case 2 mesh with two levels of *h*-adaptation.

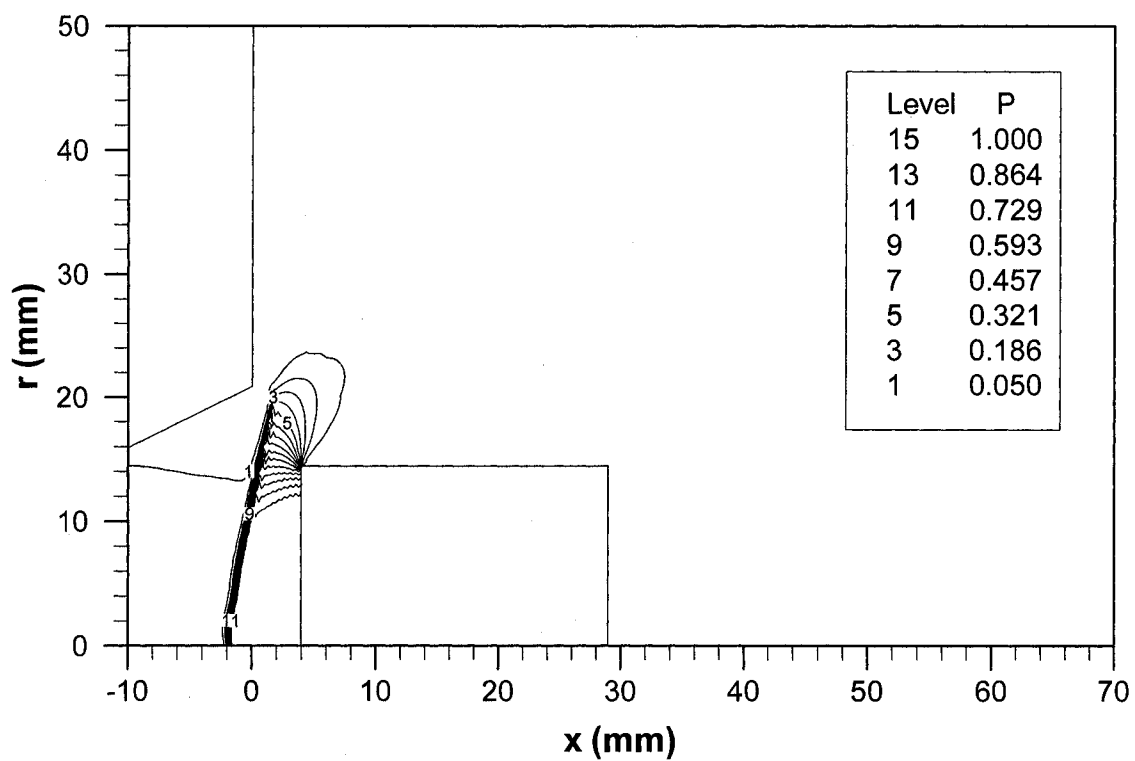Figure 6-6. Pressure contours around projectile for case 1 at $x = 4$ mm.

Figure 6-7. Pressure contours around projectile for case 2 at $x = 4$ mm.
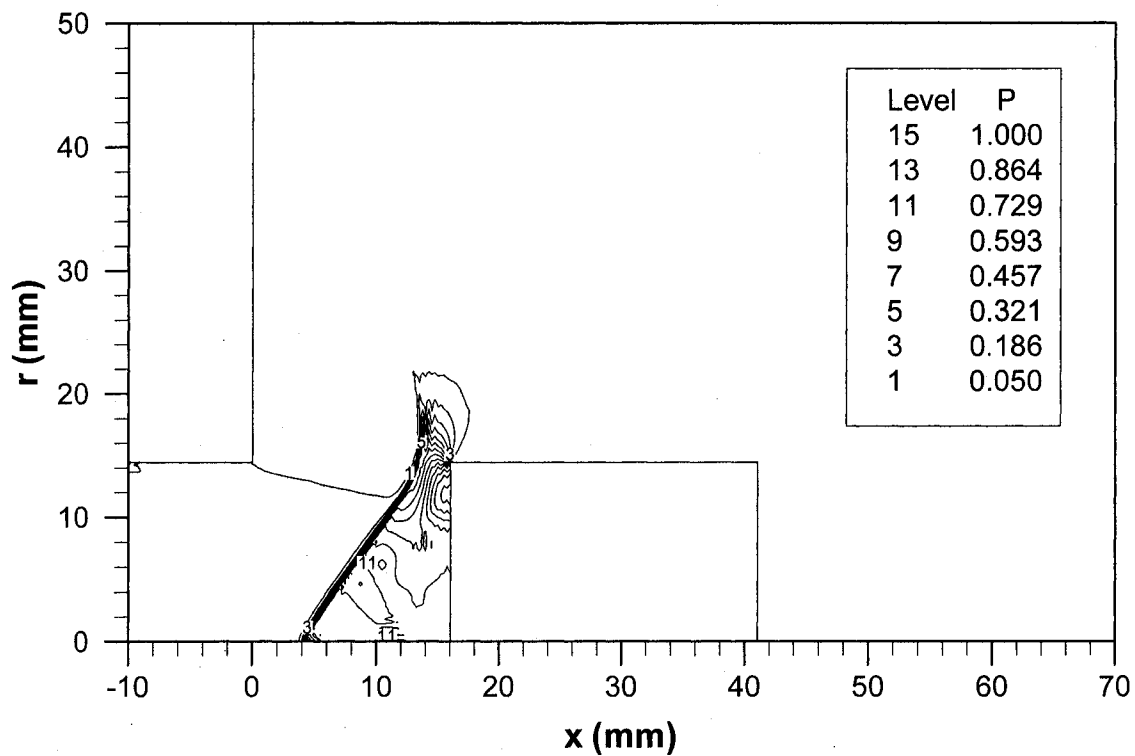
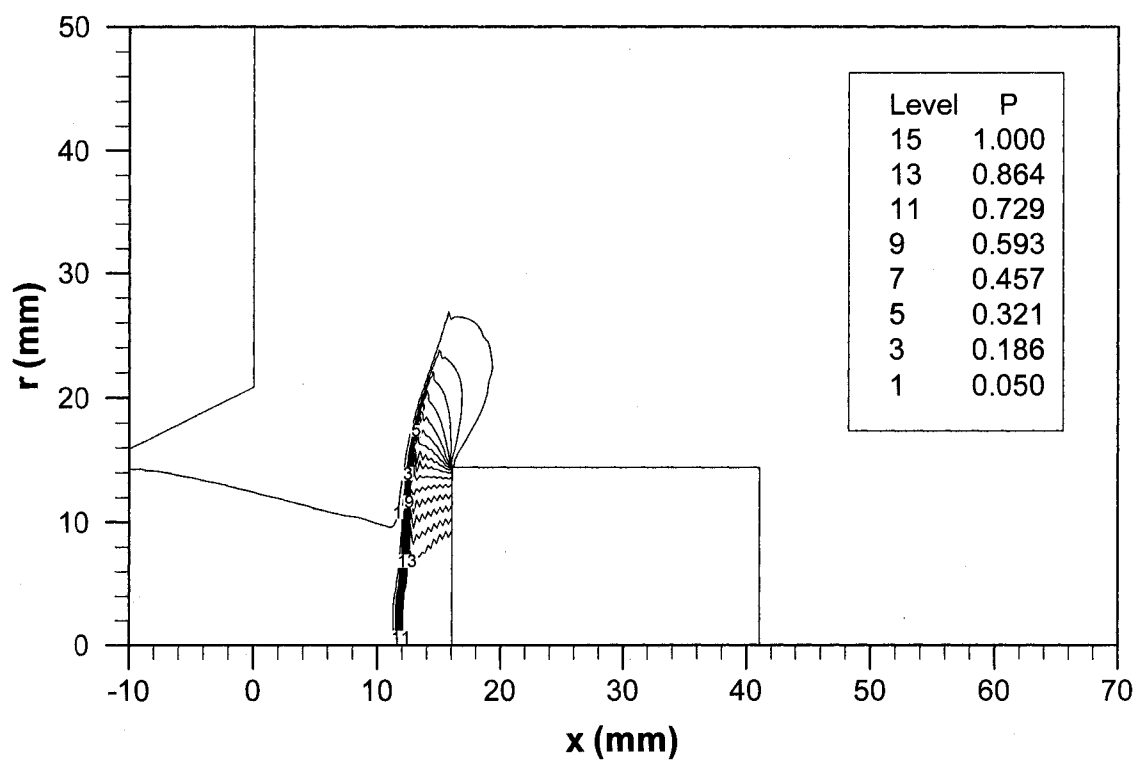Figure 6-8. Pressure contours around projectile for case 1 at $x = 16$ mm.

Figure 6-9. Pressure contours around projectile for case 2 at $x = 16$ mm.

Figure 6-10. Pressure contours around projectile for case 1 at $x = 32$ mm.

Figure 6-11. Pressure contours around projectile for case 2 at $x = 32$ mm.

Figure 6-12. Pressure contours around projectile for case 1 at $x = 48$ mm.

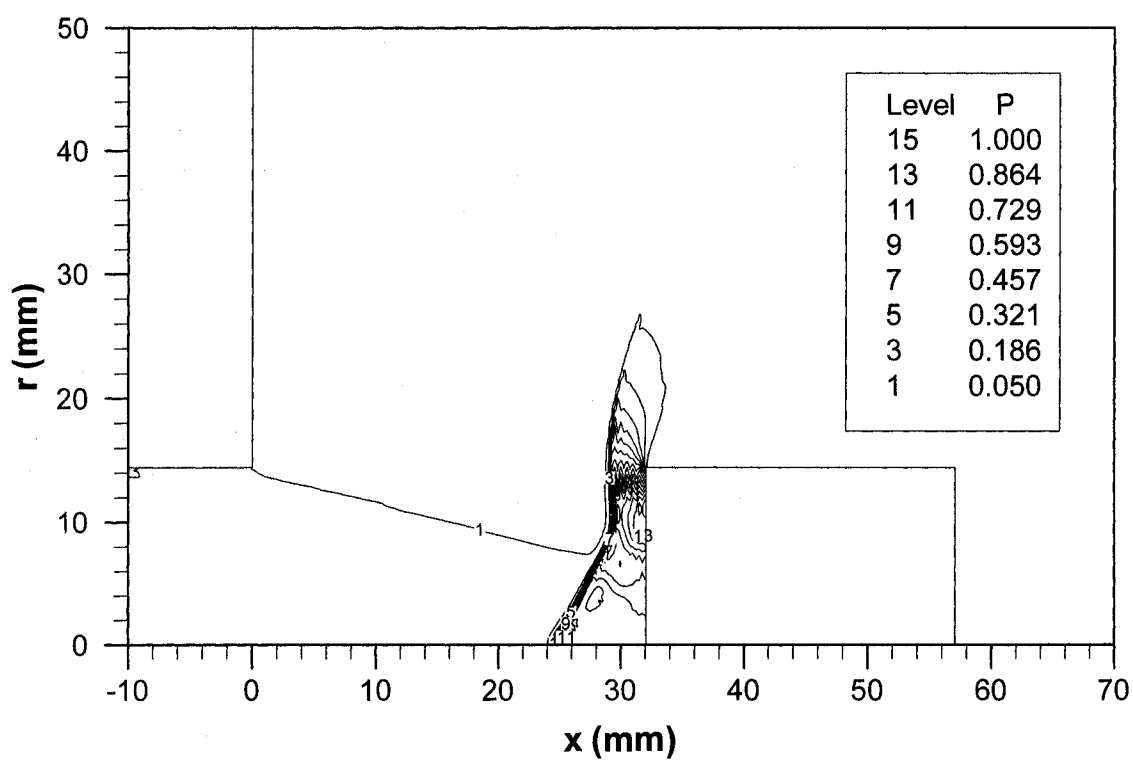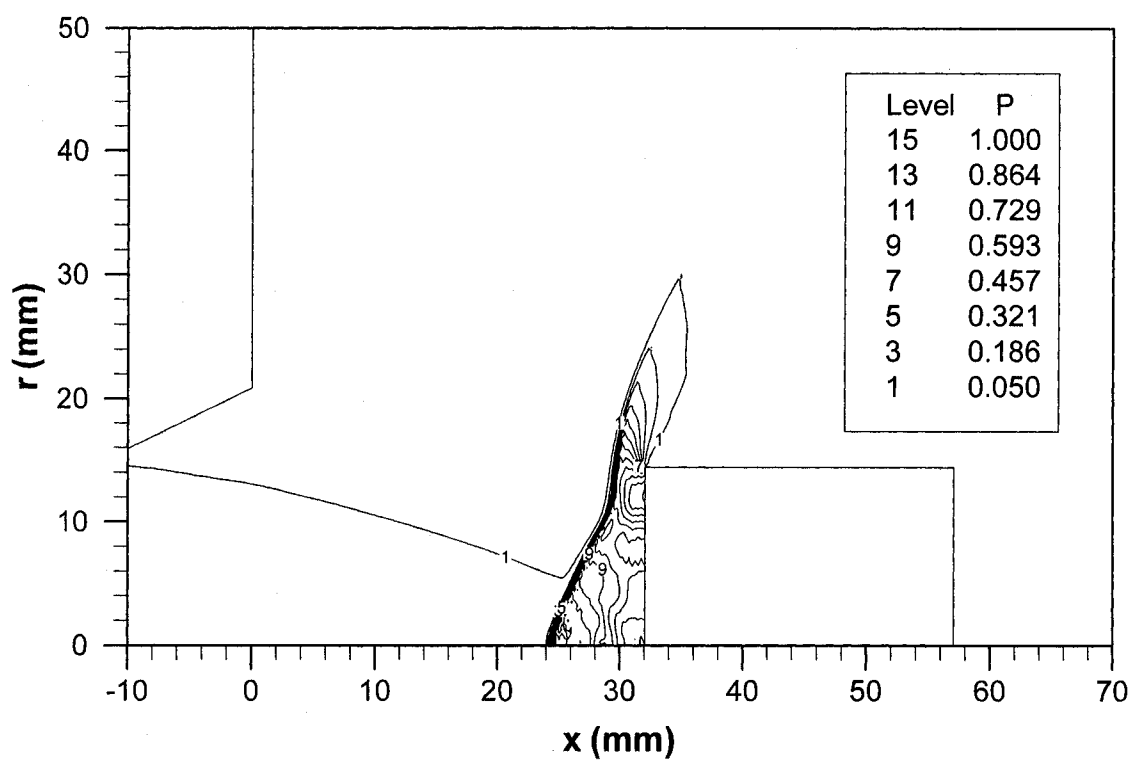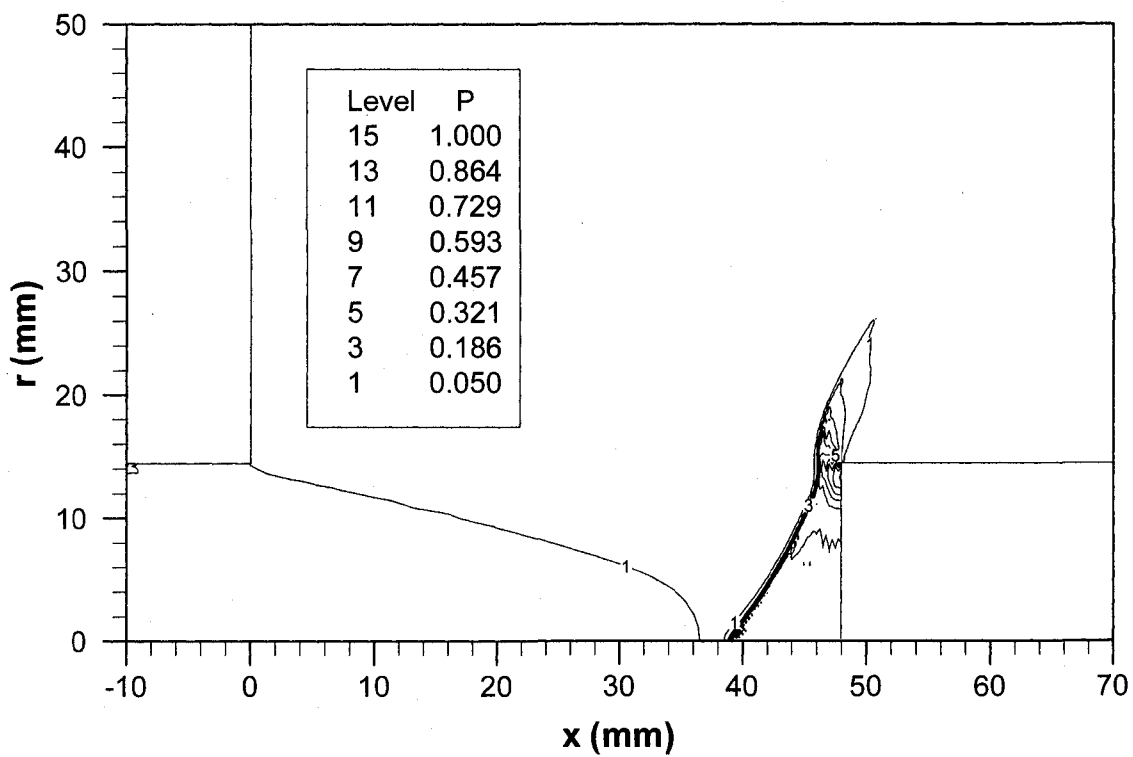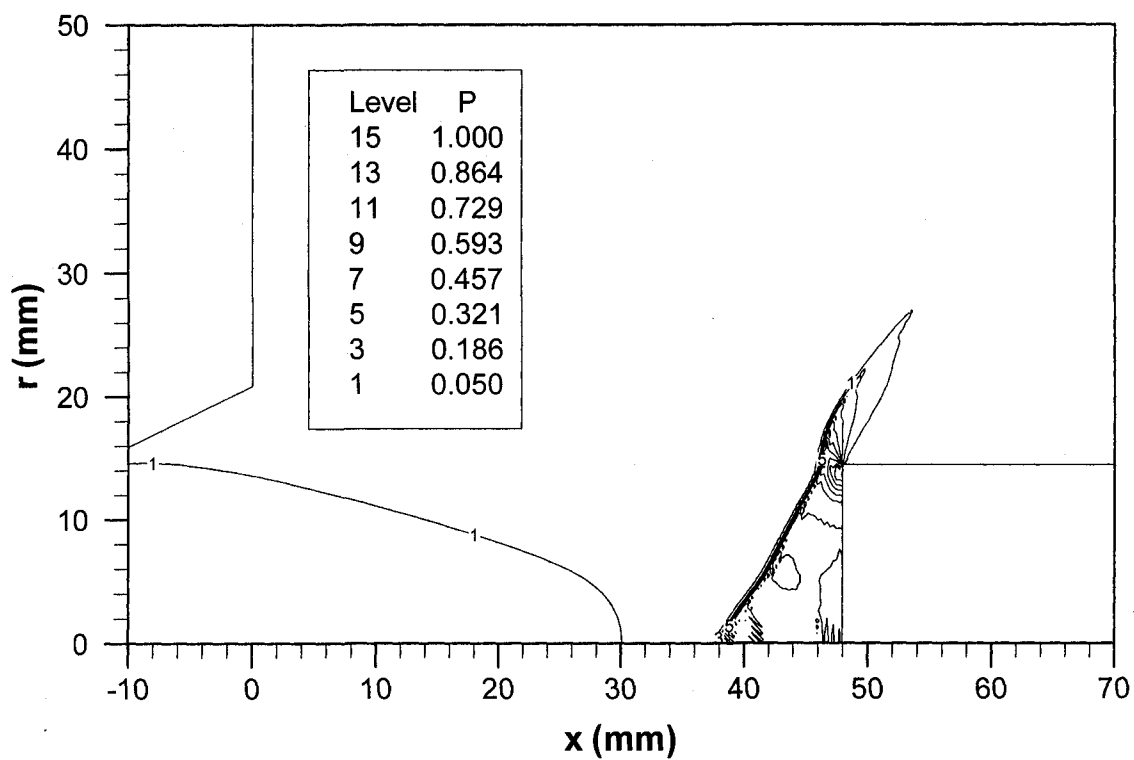Figure 6-13. Pressure contours around projectile for case 2 at x = 48 mm.
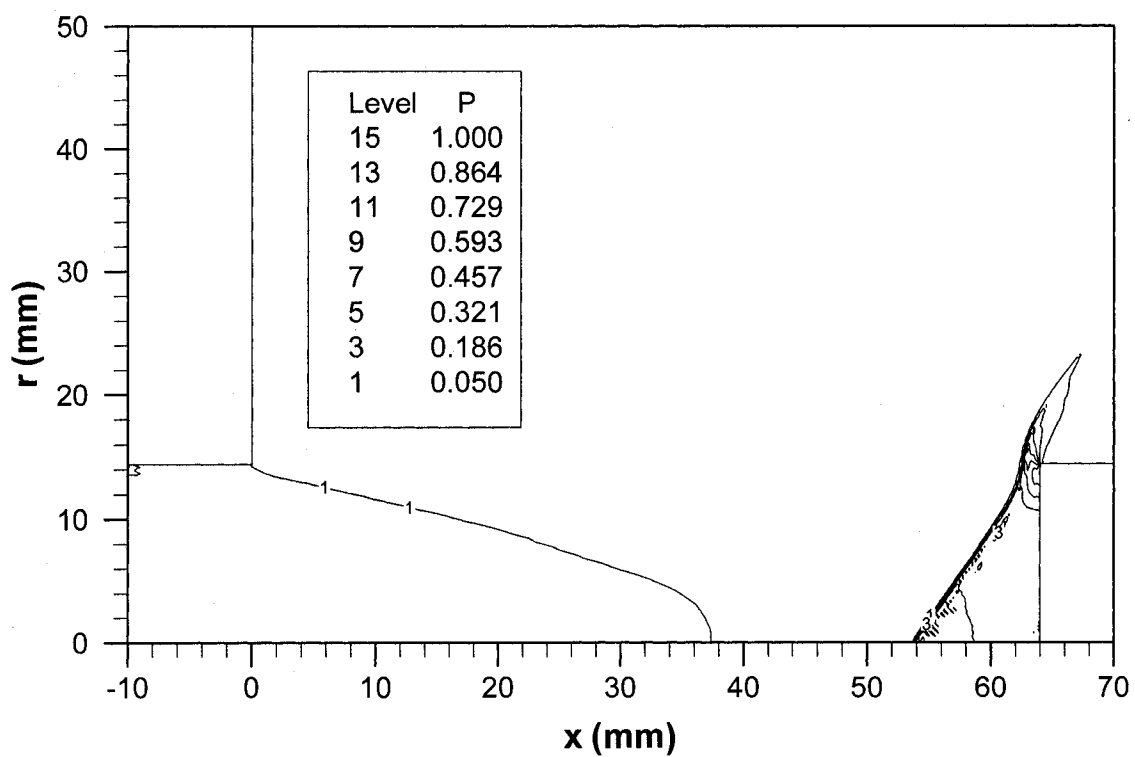
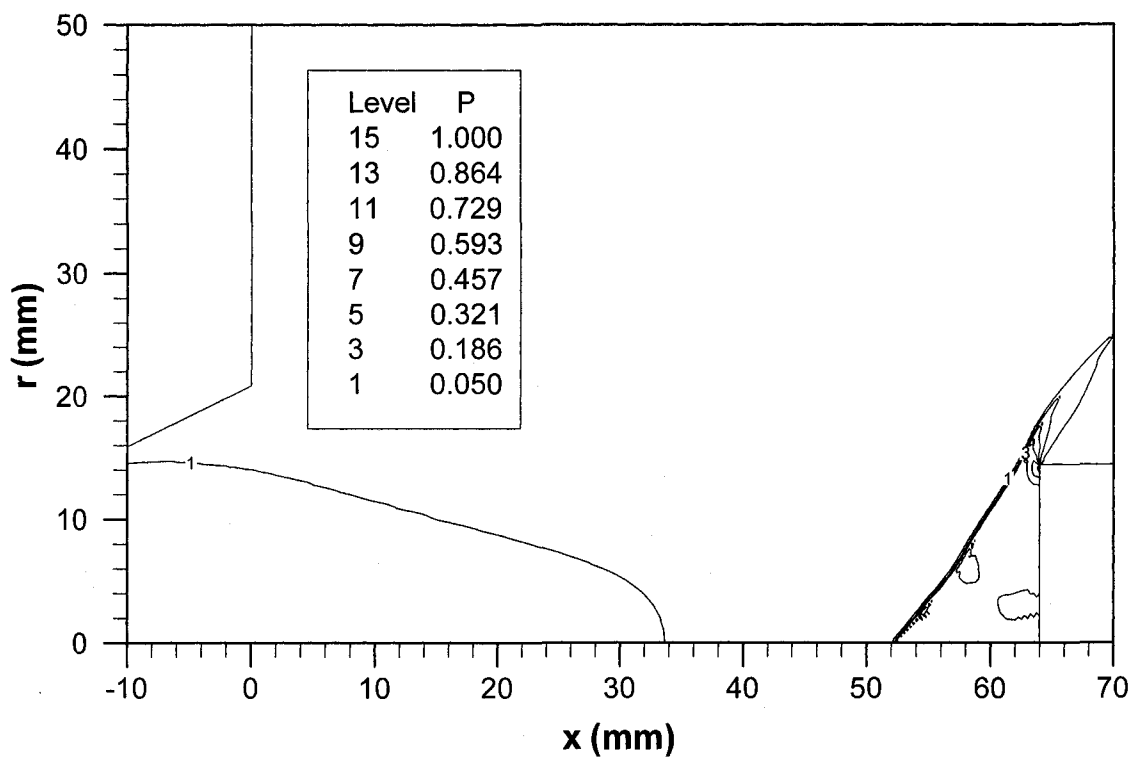Figure 6-14. Pressure contours around projectile for case 1 at $x = 64$ mm.

Figure 6-15. Pressure contours around projectile for case 2 at $x = 64$ mm.

Three Dimensional Results

An example of an initial coarse mesh used is shown in figure 6-16. The coarse mesh consists of 21,472 nodes and 18,858 elements. After adapting the mesh to two levels the resulting mesh contains 126,767 nodes and 127,570 elements (see Figure 6-17). Solutions for the projectile located at $x = 8$, 16, 24, and 32 for both cases are presented in Figures 6-18 through 6-25. These figures indicate the pressure contours at slices in the x-plane coincident with the rear face of the projectile.

For both cases the flow field is irregular close to muzzle exit. As indicated on the slices located at 8 mm, there is clearly an asymmetric pressure distribution on the rear face of the projectile (see Figures 6-18 & 6-19). It is possible that this asymmetric pressure distribution could cause a change in attitude of the projectile. When comparing Figures 6-18 and 6-19, it is interesting to note that case 2 shows more irregularity than case 1, indicating that the configuration of case 2 would actually cause more tilt in the projectile.

Further downstream, at 16 mm, 24 mm, and 32 mm (see Figures 6-20 through 6-25), the pressure contours are symmetric, which would indicate no changes in attitude would be spawned from those locations.
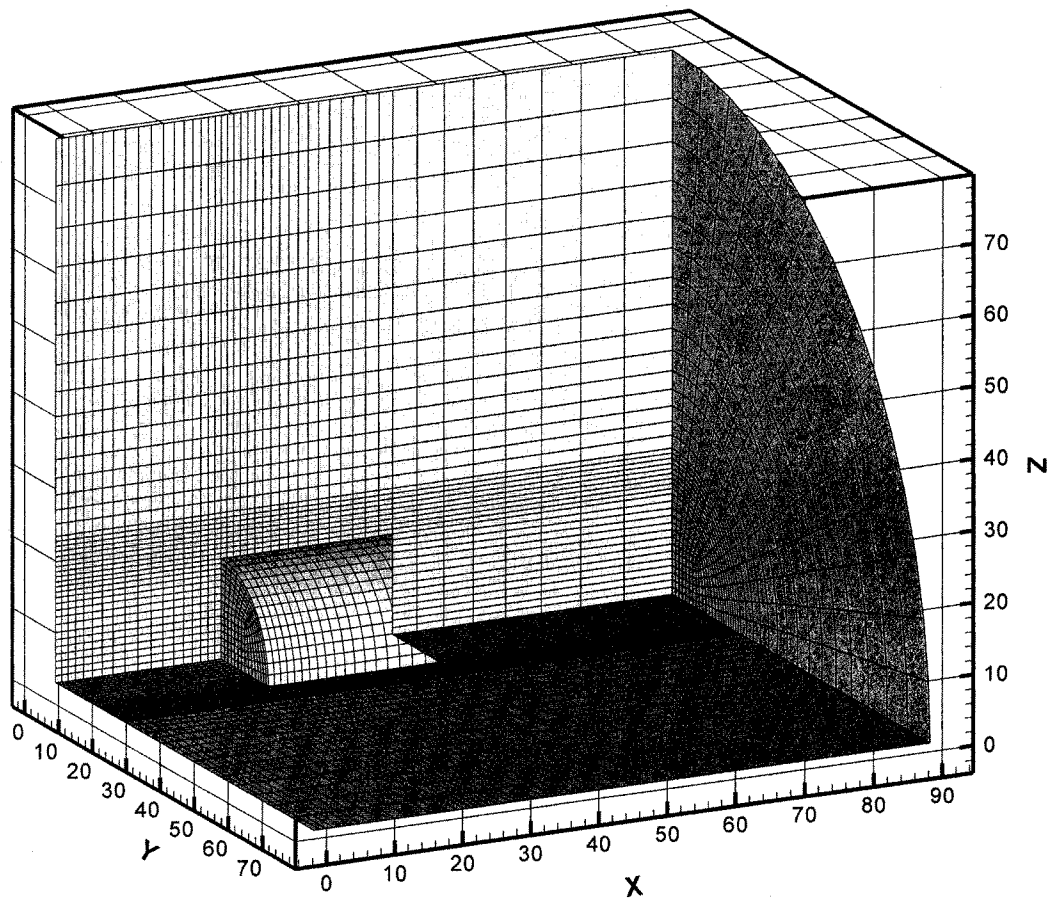
Figure 6-16. Example of initial coarse three-dimensional mesh with 21,472 nodes and 18,858 elements (all dimensions in mm).
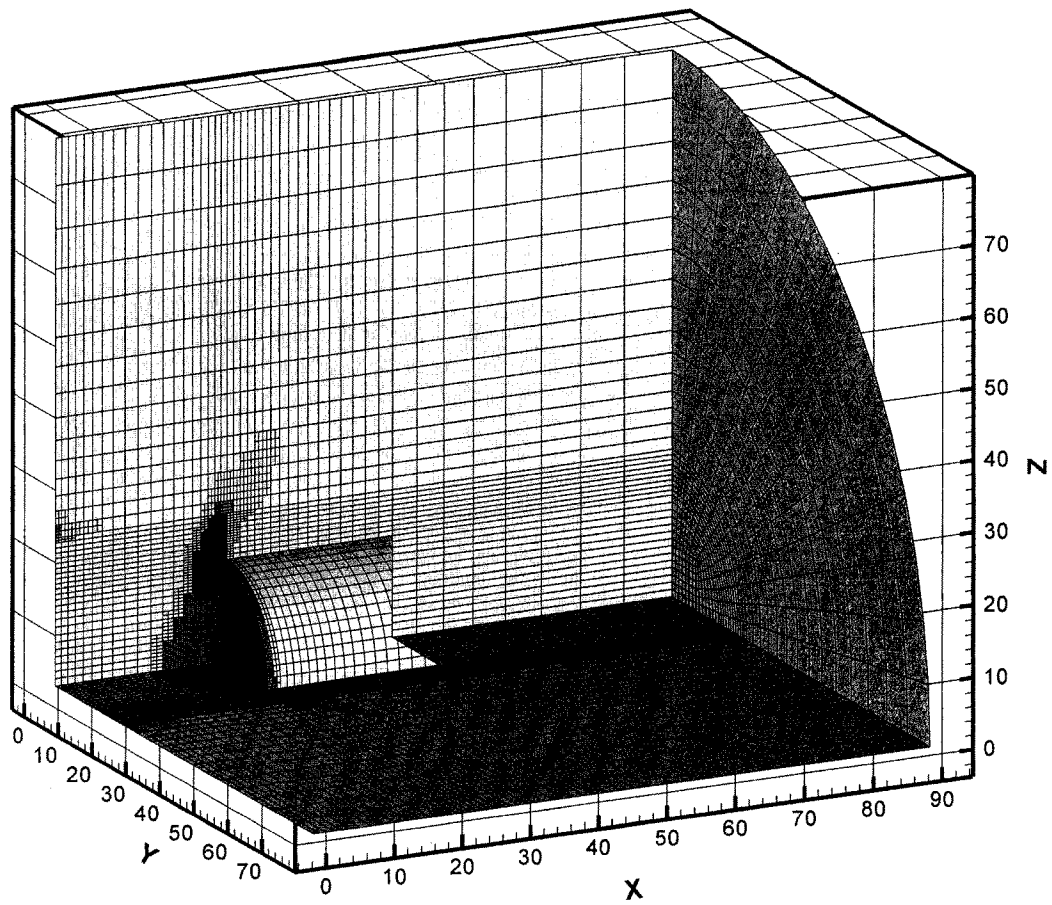
Figure 6-17. Example of mesh refined 2 levels with 126,767 nodes and 127,570 elements (all dimensions in mm).
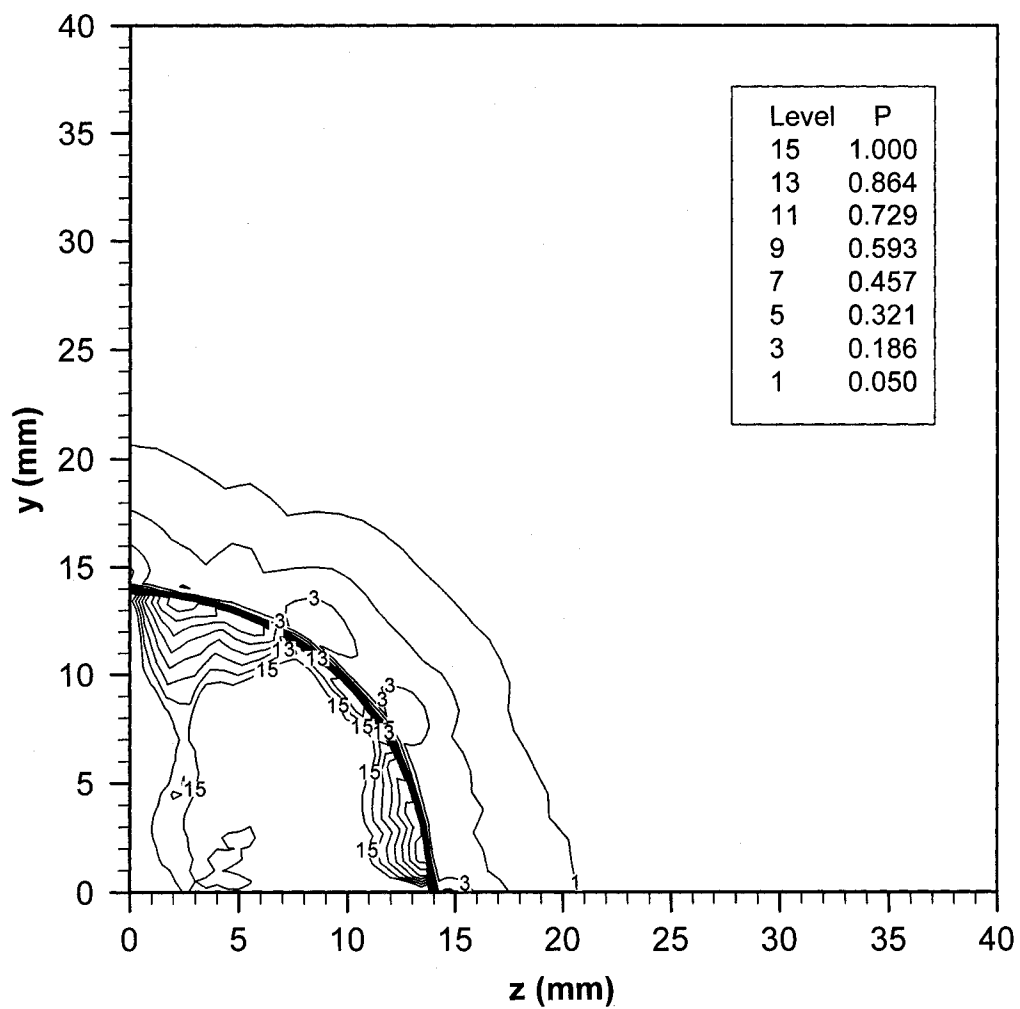
Figure 6-18. Pressure contours for case 1 on rear face of projectile at $x = 8$ mm.
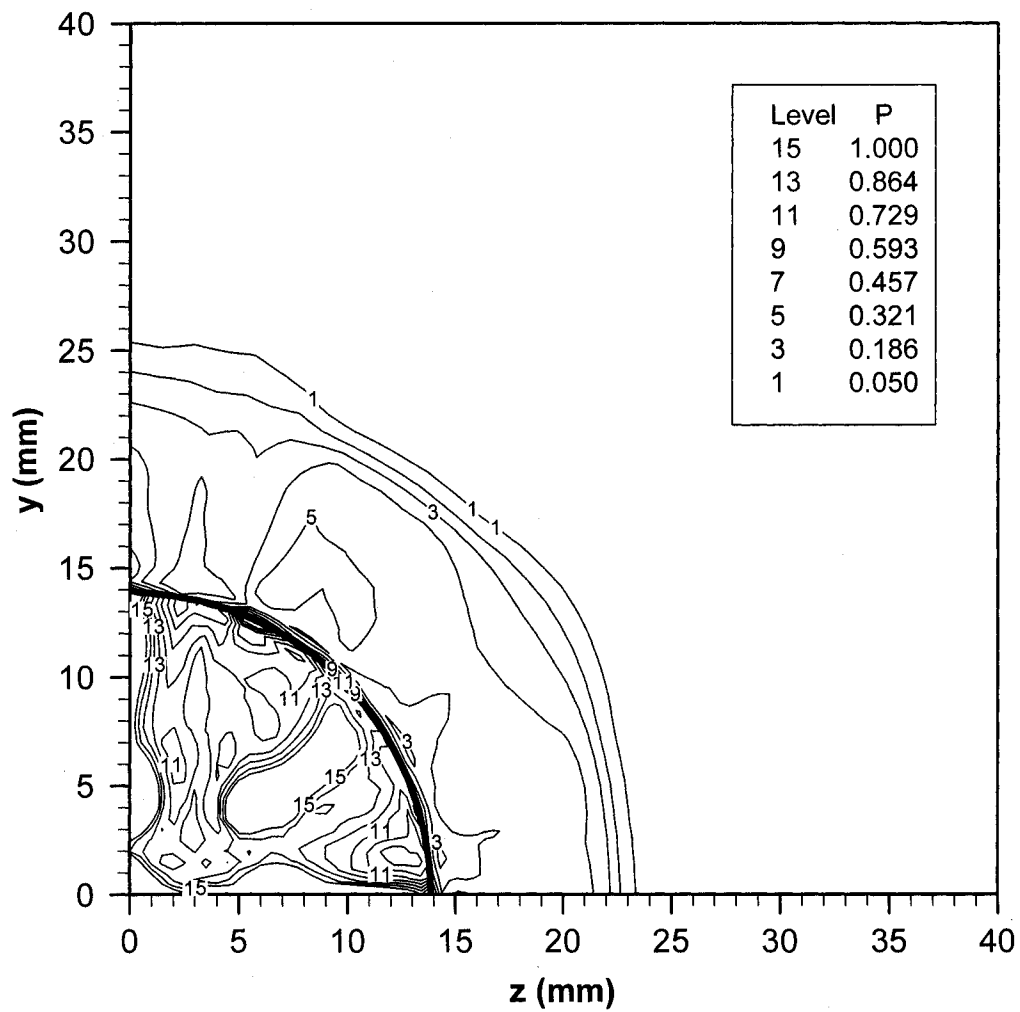
Figure 6-19. Pressure contours for case 2 on rear face of projectile at $x = 8$ mm.
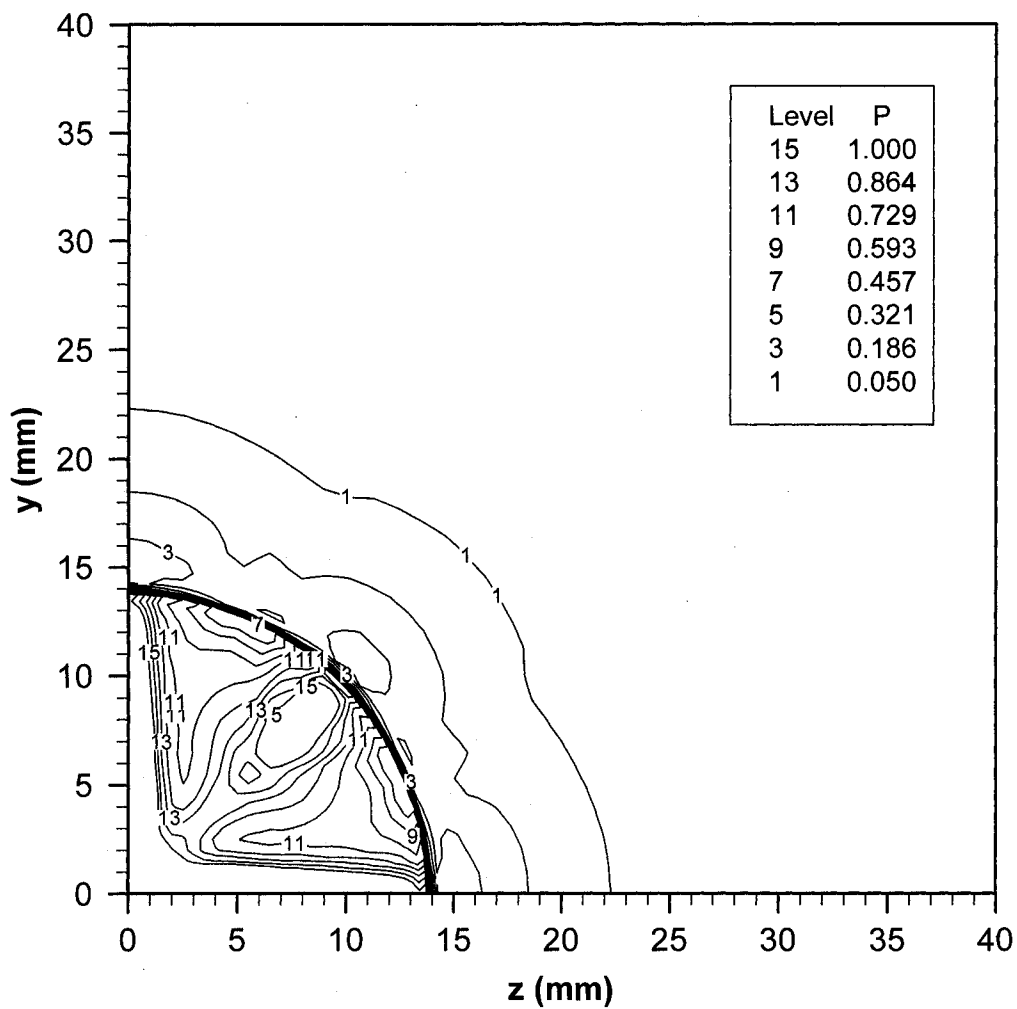
Figure 6-20. Pressure contours for case 1 on rear face of projectile at x = 16 mm.
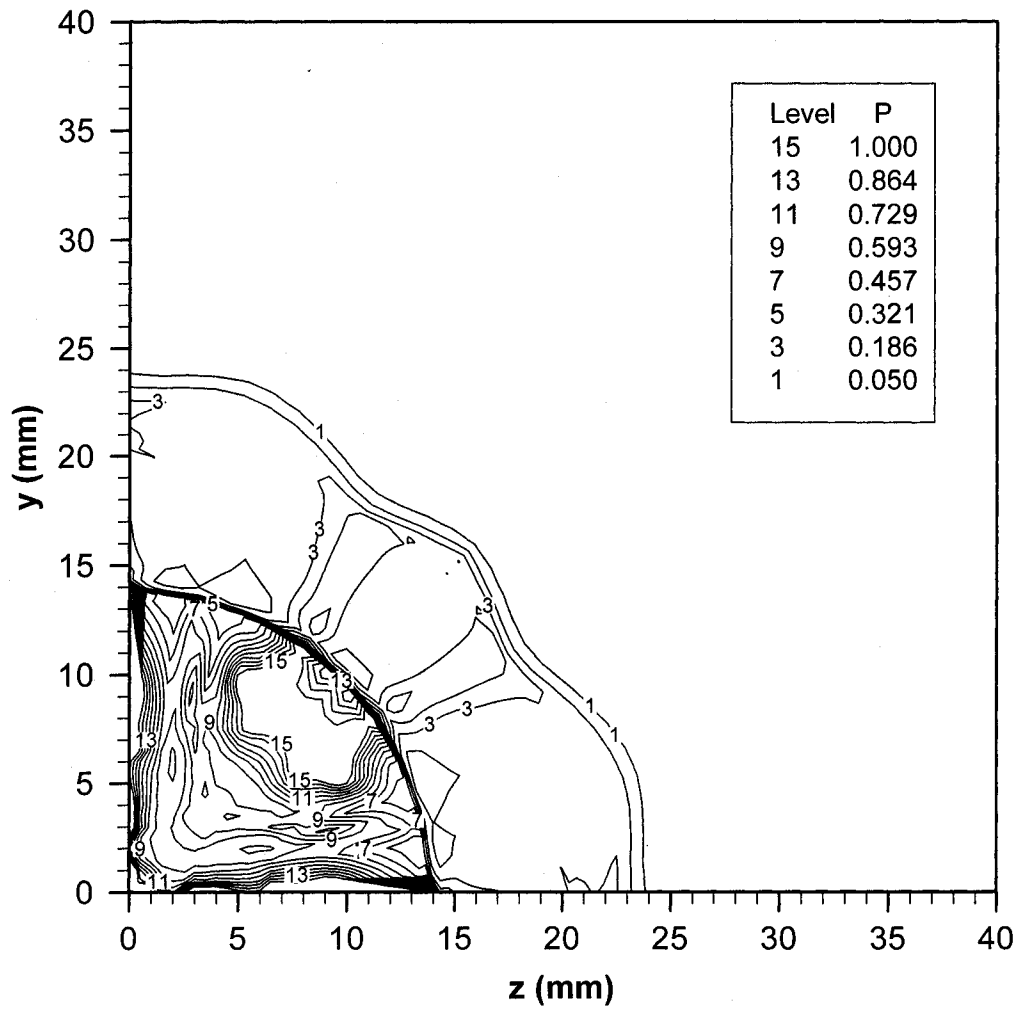
Figure 6-21. Pressure contours for case 2 on rear face of projectile at $x = 16$ mm.
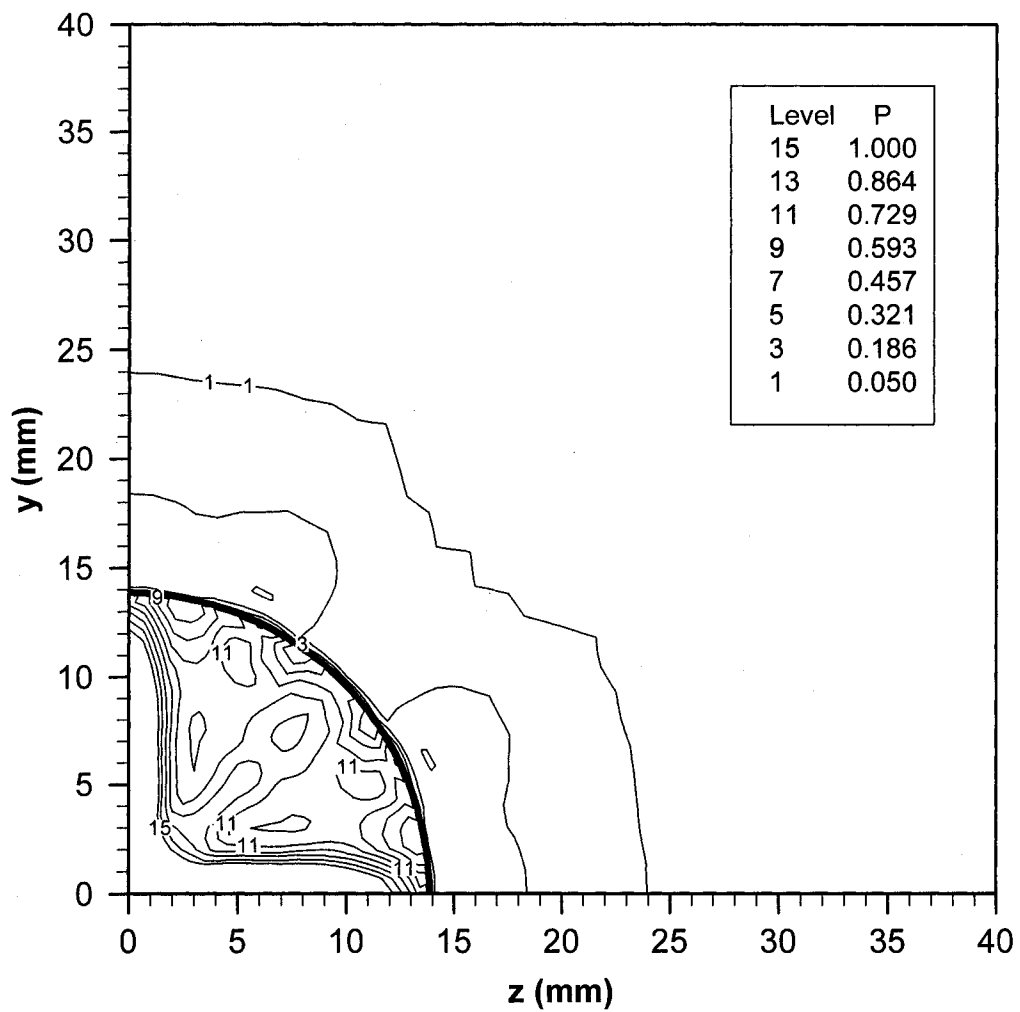
Figure 6-22. Pressure contours for case 1 on rear face of projectile at $x = 24$ mm.
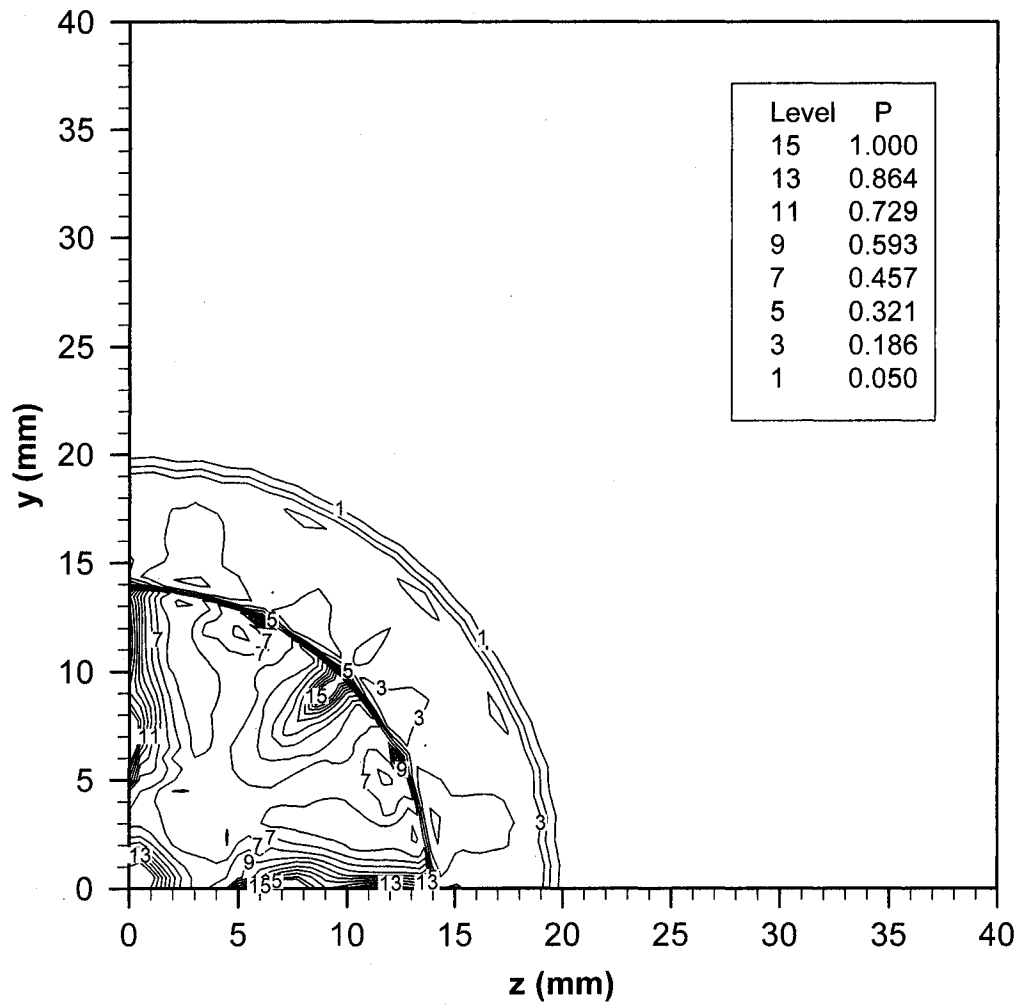
Figure 6-23. Pressure contours for case 2 on rear face of projectile at $x = 24$ mm.
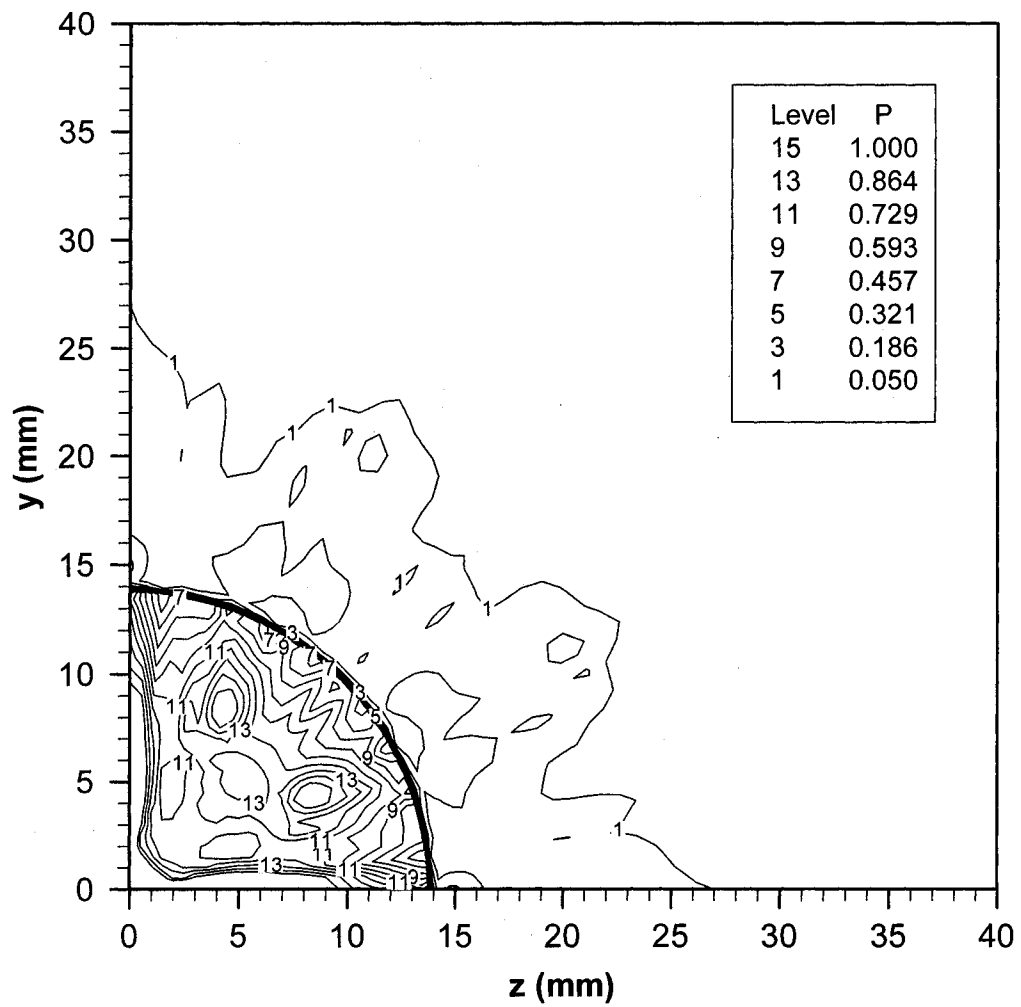
Figure 6-24. Pressure contours for case 1 on rear face of projectile at $x = 32$ mm.

Level	P
15	1.000
13	0.864
11	0.729
9	0.593
7	0.457
5	0.321
3	0.186
1	0.050
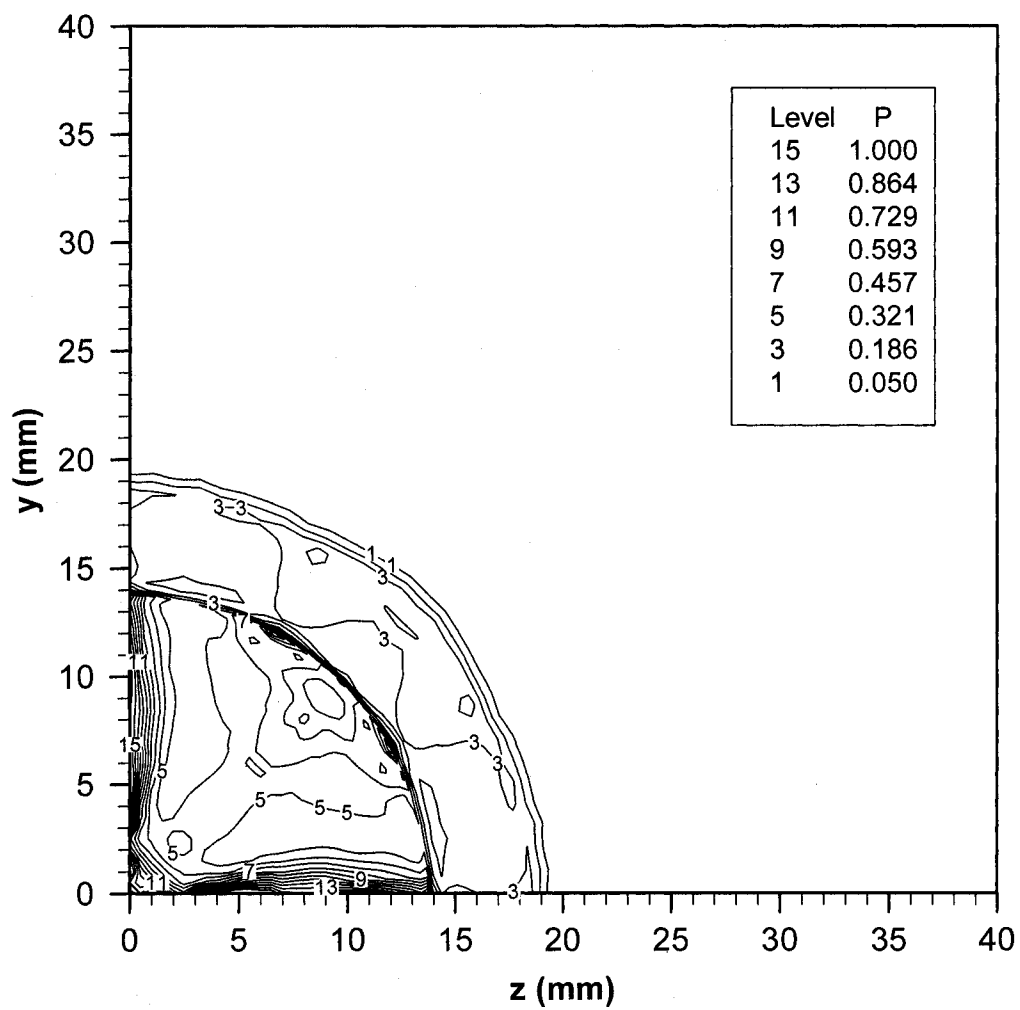
Figure 6-25. Pressure contours for case 2 on rear face of projectile at $x = 32$ mm.

CHAPTER 7

CONCLUSIONS

A compressible flow solver for viscous and inviscid flows has been developed. The finite element method employed incorporates the use of trilinear, hexahedral, isoparametric elements, as well as the use of Petrov-Galerkin weighting applied to the advection terms. An h-adaptive mesh refinement scheme based on elemental flow feature gradients was also developed and implemented.

Numerical solutions of several benchmark problems were presented, illustrating this model's ability to accurately capture shock waves and resolve viscous boundary layers. The benchmark results also illustrated the ability of the h-adaptive mesh refinement algorithm to increase solution accuracy.

The algorithm developed here was also used to investigate the flow field around a projectile as it exits the muzzle of the JASPER light-gas gun. Specifically the model was applied to investigate if a change in muzzle geometry would cause the projectile to tilt in the axial direction during free flight. A comparison between two launch tube exit geometries was made. The first case was standard muzzle geometry, where the wall of the bore and the outer surface of the launch tube form a 90 degree angle. The second case included a 26.6 degree bevel transition from the wall of the bore to the outer surface of the launch tube.

101

Results showed that for both cases the flow field is irregular close to the muzzle exit and more uniform further downstream. This would indicate that any tilting in the projectile would be spawned closer to the muzzle. Of particular interest was the fact that close to the muzzle, case 2 showed more irregularity than case 1. This would suggest that the configuration of case 2 might actually cause more tilt in the projectile, and that the case 2 configuration is not an improvement over case 1.

Based on the current study, several recommendations can be made for future research. First, greater computational speed and resources would certainly allow much larger problems to be examined. Therefore, a logical next step would be to parallelize the compressible flow solver as well as the h-adaptive algorithm.

Another logical progression would be to implement this compressible flow solver with an hp-adaptation scheme, since hp-adaptive methods are known to have better convergence rates than h-adaptive methods (Oden and Demkowicz, 1989). Much study has been devoted to applying hp-adaptive schemes to a wide variety of fluid flow problems. Hp-adaptive techniques have been used with elliptic boundary value problems for many years (Suli and Houston, 2003). While the application to hyperbolic problems is less common, some work has been done at the two dimensional level, such as that by Devloo, et al. (1988). The use of three dimensional hp-adaptation is even rarer today, although some studies are underway (Demkowicz, et al., 2002). Applying an hp-adaptation scheme to this compressible flow solver would be an interesting and challenging project.

Finally, as noted in Chapter 6, all simulations for the JASPER problem were conducted in a quasi-steady state, meaning that at each location the projectile is held

fixed while the flow field is calculated. The drawback of this approach is that at each location of the projectile, flow interactions with the projectile at prior locations are not taken into account. Therefore, this method cannot quantify the amount of tilt that the projectile might experience. It would be desirable to have the ability to track the changes in the projectile's flight path as it interacts with the flow field. This would certainly require greater computational effort, as changes in the mesh would have to be updated frequently.

APPENDIX A

NOMENCLATURE

a	speed of sound
A	flow feature gradient
c_p	specific heat at constant pressure
c_v	specific heat at constant volume
e	internal energy
h	characteristic element length
$\mathbf{J}$	Jacobian matrix
k	thermal conductivity
L	reference length; linear operator
$\mathbf{L}$	mass matrix for continuity equation
M	Mach number
$\mathbf{M}$	mass matrix for momentum and energy equations
$\mathbf{n}$	outward unit normal vector
N_i	shape function
p	pressure
P	Petrov-Galerkin perturbation factor
Pr	Prandtl number
$\mathbf{q}$	heat flux vector
R	universal gas constant; residual
$\mathbf{R}$	right hand side load vector
Re	Reynolds number
S	Sutherland constant
t	time
T	temperature
u	x-component of velocity vector
$\mathbf{u}$	velocity vector
U_∞	free stream velocity
v	y-component of velocity vector
w	z-component of velocity vector; Gauss weight
w, W_i	weighting function
x	horizontal Cartesian coordinate
$\mathbf{x}$	Cartesian space vector
y	lateral Cartesian coordinate
z	vertical Cartesian coordinate

104

α	Petrov-Galerkin optimal value; upper refinement threshold
β	lower refinement threshold
γ	ratio of specific heats; Petrov-Galerkin stability parameter
Γ	boundary
δ_{ij}	Kronecker delta
ε	error
ζ	natural (nondimensional) coordinate (z)
η	natural (nondimensional) coordinate (y)
μ	dynamic viscosity
ρ	density
σ	standard deviation
$\boldsymbol{\sigma}$, σ_{ij}	stress tensor
ξ	natural (nondimensional) coordinate (x)
Ω	domain
∇	divergence operator

Subscripts

e	internal energy; element
i, in	inlet
i,j	node numbers; column-row reference in vectors
i,j,k	unit vectors in the x, y, and z directions
u,v,w	velocity components in the x,y,z directions
x,y,z	x,y,z coordinate directions
1	in front of shock
2	behind shock
∞	reference quantity (free stream)

Superscripts

i	node numbers
n	previous time level
$n+1$	new time level
*	dimensionless parameter

APPENDIX B

FLOWCHARTS

Flowcharts for the solution algorithms used in this study are presented in this section. Figures B-1 and B-2 show the flowcharts for the compressible flow solver. Figures B-3 through B-5 show the flowcharts for the h-adaptive process.

106
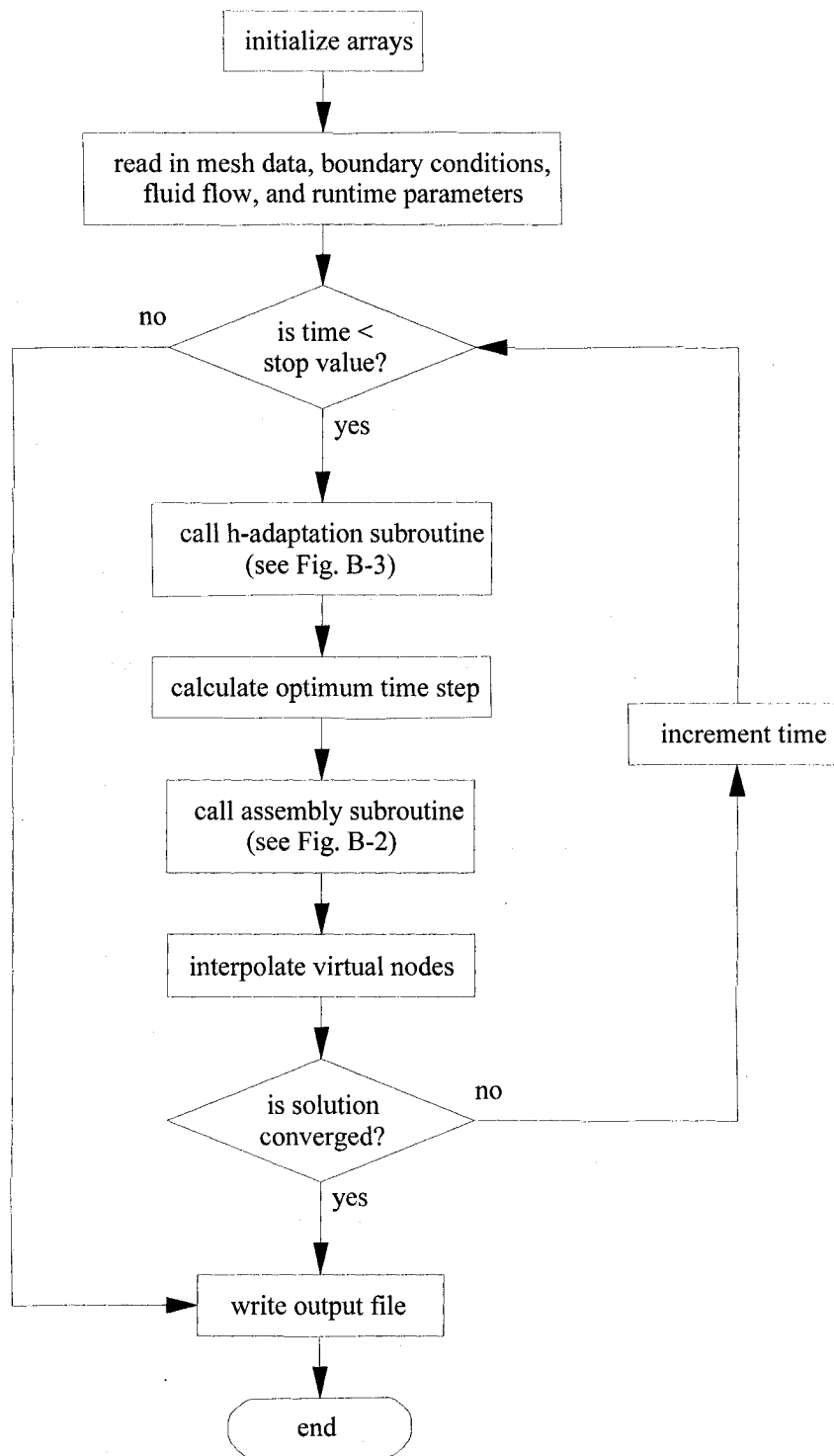
Figure B-1. Compressible flow solver flowchart
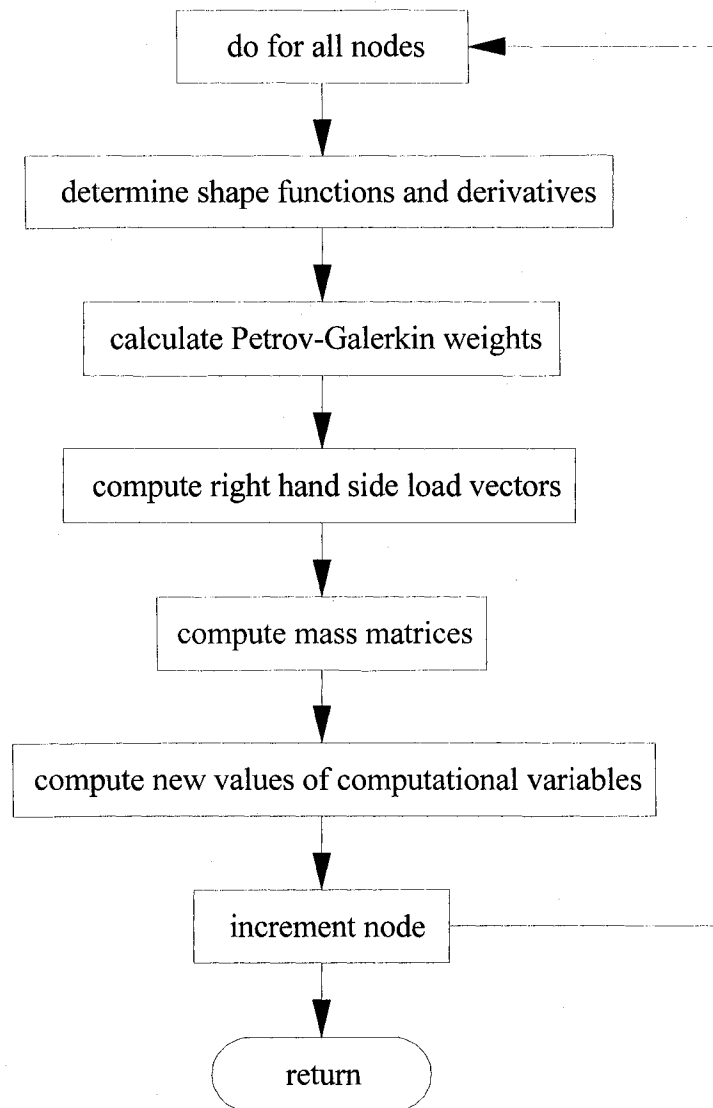
Figure B-2. Assembly flowchart
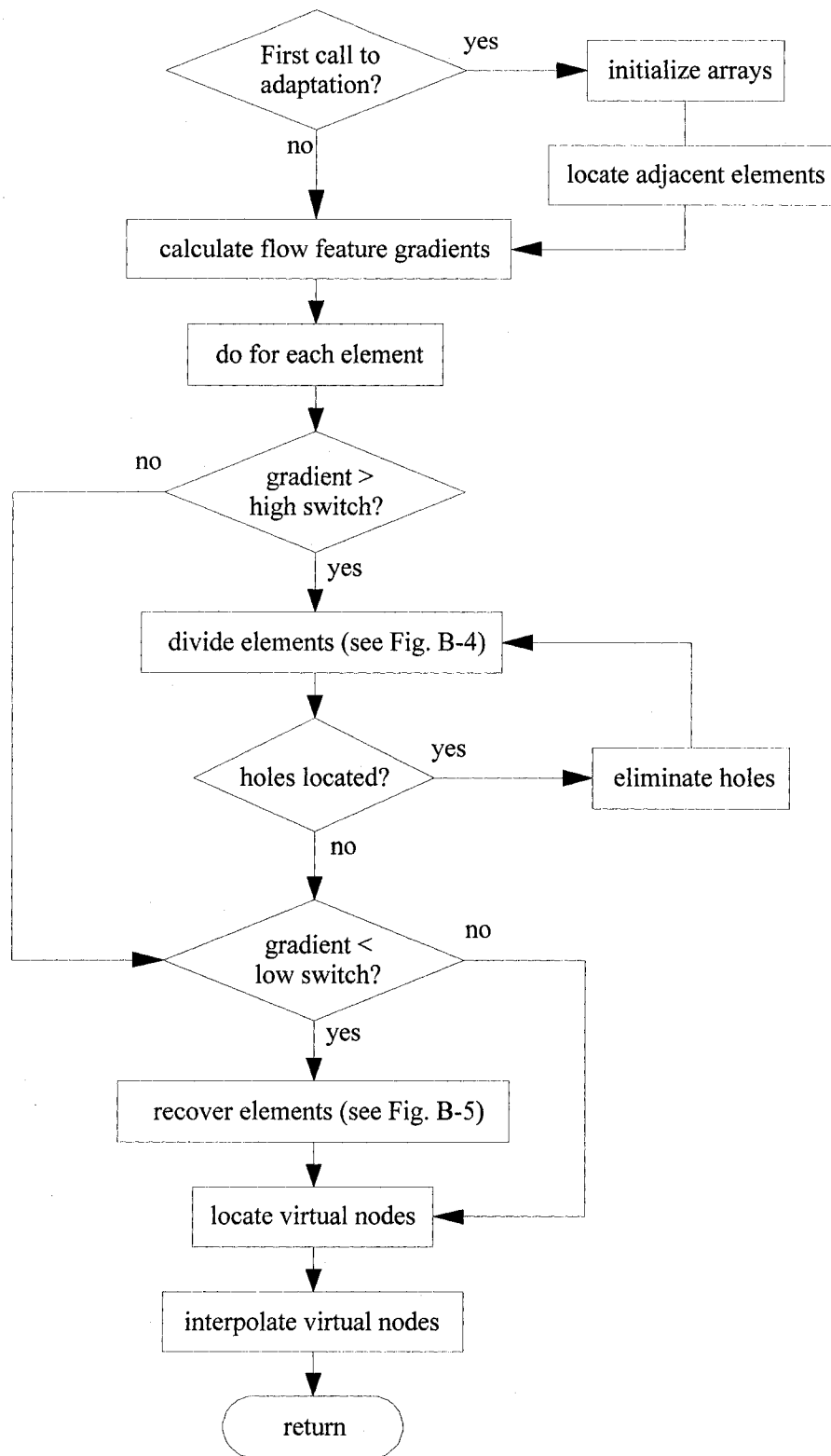
Figure B-3. *h*-adaptation flowchart

Figure B-4. Element division flowchart

do for all elements tagged for recovery

do for each face of element

is neighbor at same level?

no

is neighbor's neighbor at same level?

no

yes

yes

do not delete existing nodes they will become virtual nodes

delete existing nodes

delete children

update adjacent elements

update boundary conditions

increment face

increment element

return

Figure B-5. Element recovery flowchart

REFERECES

Almeida, R.C. and Galeao, A.C., "An Adaptive Petrov-Galerkin Formulation for the Compressible Euler and Navier-Stokes Equations", *Computer Methods in Applied Mechanics and Engineering*, Vol. 129, 1996, pp. 157-176.

Anderson, J.D., *Hypersonic and High Temperature Gas Dynamics*, American Institute of Aeronautics and Astronautics, Reston, VA, 2000.

Anderson, J.D., *Modern Compressible Flow with Historical Perspective*, McGraw-Hill, San Francisco, CA, 1990.

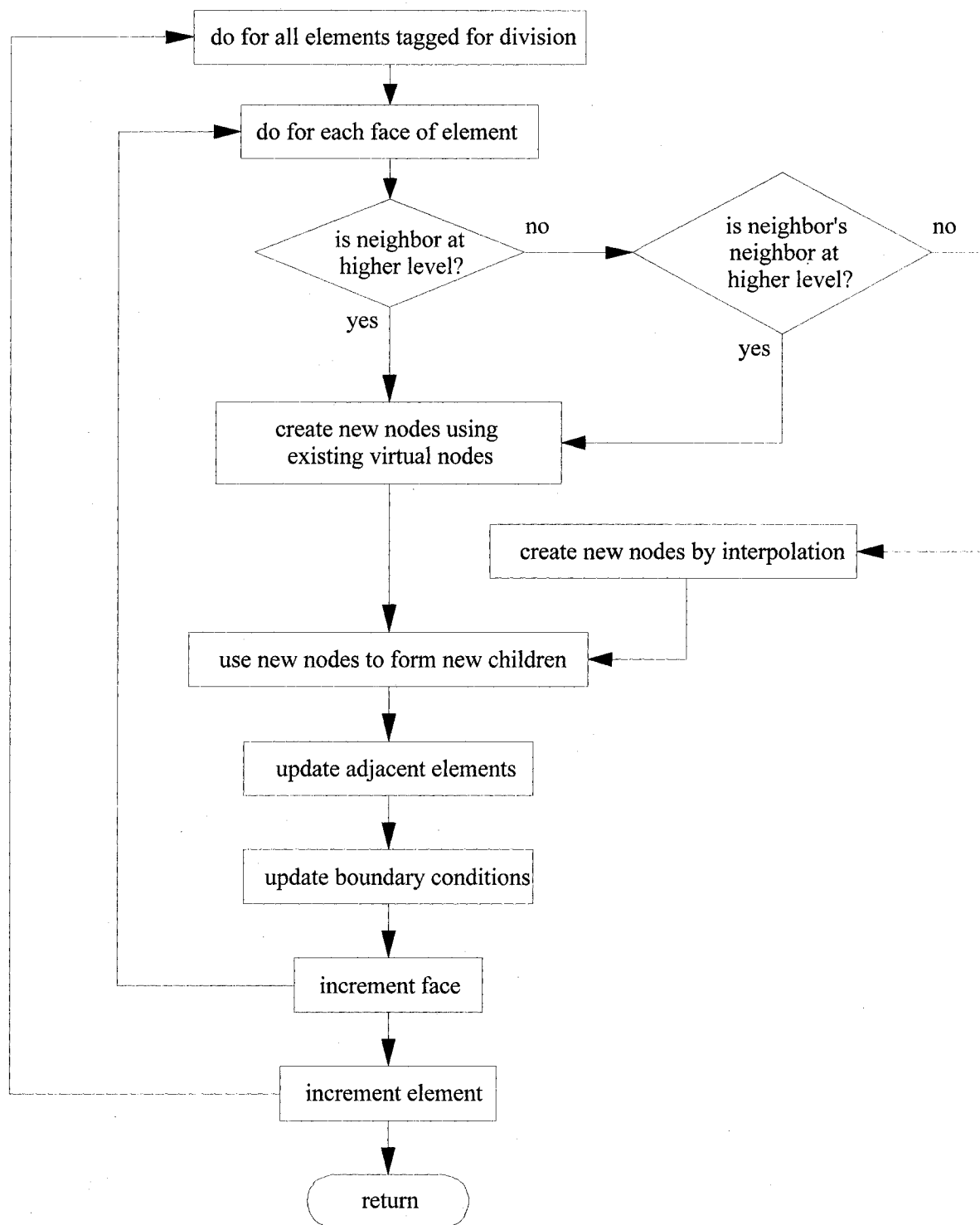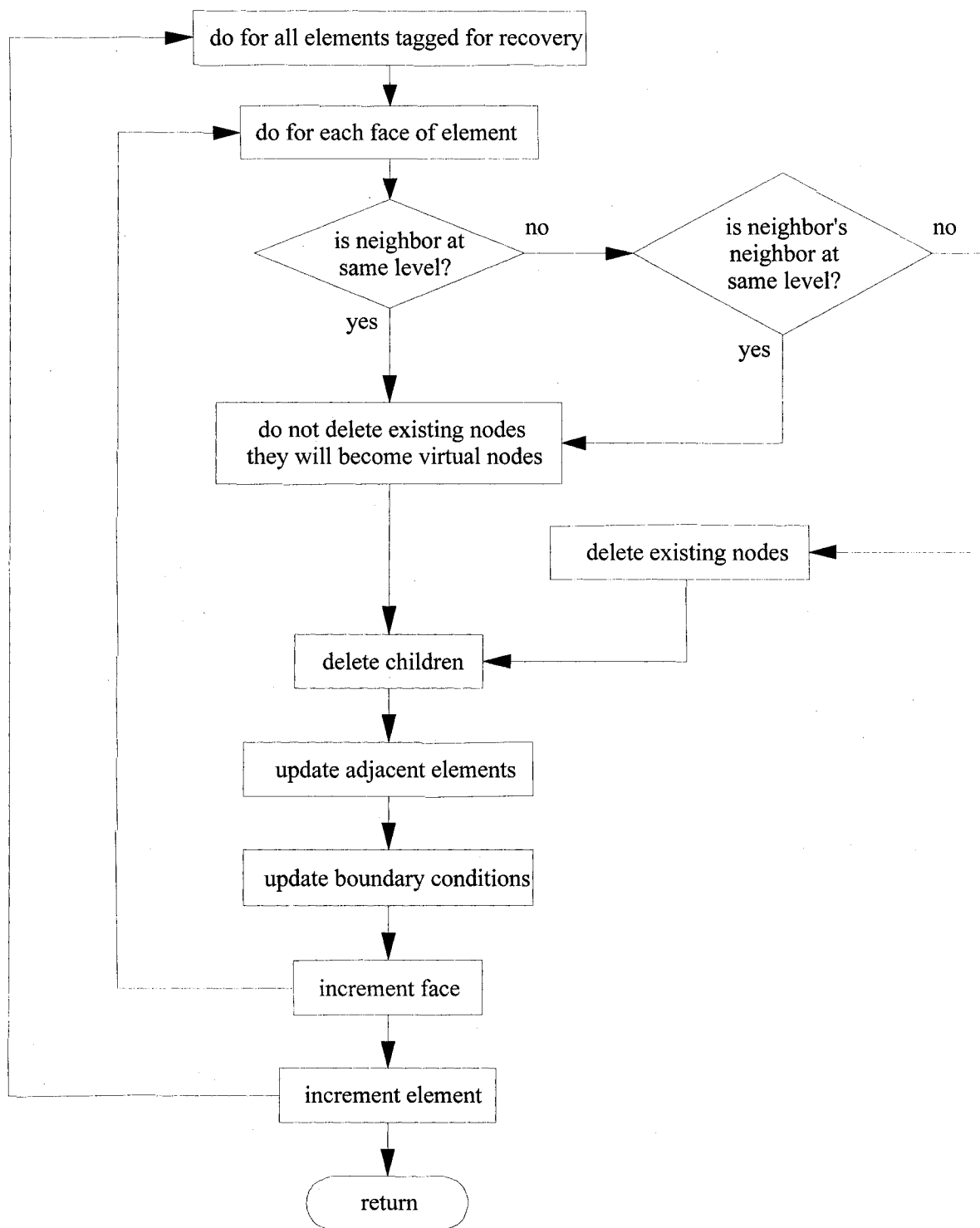Beer, G. and Watson, J.O., *Introduction to Finite and Boundary Element Methods for Engineers*, John Wiley & Sons, New York, NY, 1992

Braddy, R.W., Homes, N., Konrad, C.H., Martinez, M., "Joint Actinide Shock Physics Experimental Research (JASPER) Facility Overview", 52nd Meeting of the Aeroballistic Range Association, Quebec City, Quebec, Canada, September 9-14, 2001.

Brueckner, F.P., "Finite Element Analysis of High-Speed Flows with Application to the Ram Accelerator Concept", Ph.D. Dissertation, Univ. of Arizona, Tucson, AZ, 1991.

Brueckner, F.P. and Pepper, D.W., "Parallel Finite Element Algorithm for Three-Dimensional Inviscid and Viscous Flow", *Journal of Thermophysics and Heat Transfer*, Vol. 9, No.2, 1995, pp. 240-246.

Burton, K.L. and Pepper, D.W., "Use of Adaptive Finite Elements for Compressible Flow", *Journal of Thermophysics*, Vol. 7, No. 4, 1993, pp. 744-746.

Capon, P. and Jimack, P.K., "An Adaptive Finite Element Method for the Compressible Navier-Stokes Equations". *Numerical Methods for Fluid Dynamics V*, Oxford University Press, New York, NY, 1995, pp. 327-333.

Carter, J.E., "Numerical Solutions of the Navier-Stokes Equations for the Supersonic Laminar Flow Over a Two-Dimensional Compression Corner", NASA TR R-385, Langley Research Center, Hampton, VA, 1972.

Demkowicz, L., Pardo, D., and Rachowicz, W., "*3D hp-Adaptive Finite Element Package (3Dhp90). Version 2.0. The Ultimate (?) Data Structure for Three-Dimensional, Anisotropic hp Refinements*", TICAM Report 02-24, Texas Institute for Computational Engineering and Sciences, Austin, TX, 2002.

Devloo, P., Oden, J.T., and Pattani, P., "An h-p Adaptive Finite Element Method for the Numerical Simulation of Compressible Flow", *Computer Methods in Applied Mechanics and Engineering*, Vol. 70, 1988, pp. 203-235.

Dyne, B.R., "Finite Element Analysis of Incompressible, Compressible, and Chemically Reacting Flows, with Emphasis on the Pressure Approximation", Ph.D. Dissertation, Univ. of Arizona, Tucson, AZ, 1992.

Fletcher, C.A.J., *Computational Techniques for Fluid Dynamics, Vol. 2*, Springer-Verlag, New York, NY, 1997.

Fletcher, C.A.J., *Computational Galerkin Methods*, Springer-Verlag, New York, NY, 1984.

Heinrich, J.C. and Pepper, D.W., *Intermediate Finite Element Method: Fluid Flow and Heat Transfer Applications*. Taylor & Francis, Philadelphia, PA., 1999.

Hughes, T.J.R. and Brooks, A., "A Multi-Dimensional Upwind Scheme With No Crosswind Diffusion", *Finite Element Methods for Convection Dominated Flows*, AMD vol. 34, ASME, New York, 1979, pp. 19-35.

Hughes, T.J.R., Tezduyar, T.E., and Brooks, A.N., "A Petrov-Galerkin Finite Element Formulation for Systems of Conservation Laws with Special Reference to the Compressible Euler Equations", *Numerical Methods for Fluid Dynamics*, Academic Press, Orlando, 1982, pp. 97-125.

Ilinca, F., Pelletier, D., Ignat, L., "Adaptive Finite Element Solution of Compressible Turbulent Flows", *AIAA Journal*, Vol. 36, No. 12, 1998, pp. 2187-2194.

Lohner, R., Morgan, K., Peraire, J., Zienkiewicz, O.C., and Kong, L., "Finite Element Methods for Compressible Flow", *Numerical Methods for Fluid Dynamics II*, Oxford University Press, New York, NY, 1986, pp. 27-53.

Mespoulet, J., "JASPER: Internal Ballistics Analysis Related to the First Shots", *not published*, Thiot Ingenierie, France, 2001.

Mitchell, A.C. and Nellis, W.J., "Shock Compression of Aluminum, Copper, and Tantalum", *Journal of Applied Physics*, vol. 52, May 1981, pp. 3363-3374.

Nellis, W.J., Mitchell, A.C., Ree, F.H., Ross, M., Holmes, N.C., Trainor, R.J., and Erskine, D.J., "Equation of State of Shock-Compressed Liquids: Carbon Dioxide and Air", *Journal of Chemical Physics*, vol. 95, no. 7, Oct 1991, pp. 5268-5272.

Oden, J.T. and Demkowicz, L., "A Survey of Adaptive Finite Element Methods in Computational Mechanics", *State-of-the-Art Surveys on Computational Mechanics*, ASME, New York, NY, 1989.

Pepper, D.W. and Heinrich, J.C., *The Finite Element Method: Basic Concepts and Applications*. Taylor & Francis, Hemisphere Publishing Corp., 1992.

Pepper, D.W. and Stephenson, D.E., "An Adaptive Finite-Element Model for Calculating Subsurface Transport Contaminant", *Ground Water*, Vol. 33, No. 3, 1995, pp.486-496.

Raoche, P.J., *Computational Fluid Dynamics*, Hermosa Publishers, Albuquerque, NM, 1976.

Schlichting, H., *Boundary-Layer Theory*, McGraw-Hill, San Francisco, CA, 1979.

Schonberg, W.P. and Cooper, D., "Repeatability and Uncertainty Analyses of Light Gas Gun Test Data", *AIAA Journal*, Vol. 32, No. 5, May 1994, pp. 1058-1065.

Suli, E. and Houston, P., "Adaptive Finite Element Approximation of Hyperbolic Problems", *Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics*, Springer, New York, 2003.

Taylor, H.M. and Karlin, S., *An Introduction to Stochastic Modeling*, Academic Press, San Diego, CA, 1994.

Zienkiewicz, O.C. and Taylor, R.L., *The Finite Element Method: The Basis*, Vol. 1, 5th ed., Butterworth-Heinemann, Woburn, MA, 2000. (a)

Zienkiewicz, O.C. and Taylor, R.L., *The Finite Element Method: Fluid Dynamics*, Vol. 3, 5th ed., Butterworth-Heinemann, Woburn, MA, 2000. (b)

VITA

Graduate College
University of Nevada, Las Vegas

Timothy Todd de Bues

Home Address:
    8505 Timber Pine Ave.
    Las Vegas, NV 89143

Degrees:
    Bachelor of Science, Aerospace Engineering, 1991
    University of Southern California

    Master of Science, Aerospace Engineering, 1994
    California State University, Long Beach

Special Honors and Awards:
    NASA Space Grant Fellowship, 1999
    NASA Space Grant Fellowship, 2001
    Tau Beta Pi – Engineering Honor Society

Dissertation Title: An H-Adaptive Finite Element Compressible Flow Solver Applied to
Light-Gas Gun Design

Dissertation Examination Committee:
    Chairman, Dr. Darrell W. Pepper, Ph. D.
    Committee Member, Dr. Robert Boehm, Ph. D.
    Committee Member, Dr. William Culbreth, Ph. D.
    Committee Member, Dr. Samir Moujaes, Ph. D.
    Graduate Faculty Representative, Dr. George Miel, Ph. D.