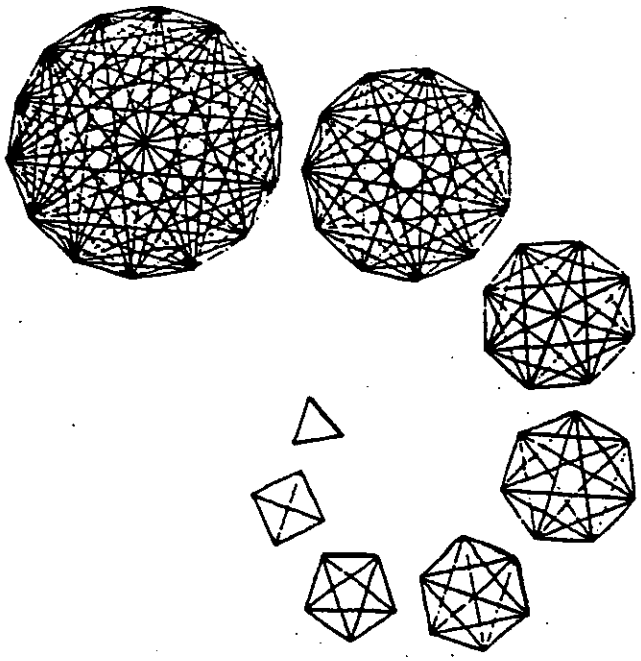
CENTER FOR PURE AND APPLIED MATHEMATICS
UNIVERSITY OF CALIFORNIA, BERKELEY

PAM- 601

# A PROJECTION METHOD FOR REACTING FLOW IN THE ZERO MACH NUMBER LIMIT

MINDY FRUCHTMAN LAI

FEBRUARY 1994



This report was done with support from the Center for Pure and Applied Mathematics. Any conclusions or opinions expressed in this report represent solely those of the author(s) and not necessarily those of the Center for Pure and Applied Mathematics or the Department of Mathematics.

# A Projection Method for Reacting Flow in the Zero Mach Number Limit

by

**Mindy Fruchtman Lai**

B.E. (The Cooper Union) 1988

A dissertation submitted in partial satisfaction of the requirements for the degree of

**Doctor of Philosophy**

in

**Engineering - Mechanical Engineering**

in the

GRADUATE DIVISION

of the

UNIVERSITY OF CALIFORNIA at BERKELEY

Committee in charge:

Professor Phillip Colella, Chair

Professor Alexandre J. Chorin

Professor Philip Marcus

1993

# A Projection Method for Reacting Flow in the Zero Mach Number Limit

Copyright 1993

by

Mindy Fruchtman Lai

# Abstract

A Projection Method for Reacting Flow in the

Zero Mach Number Limit

by

Mindy Fruchtman Lai

Doctor of Philosophy in Mechanical Engineering

University of California, at Berkeley

Professor Phillip Colella, Chair

An efficient numerical technique is presented for solving the reduced system of equations for reacting flow in the zero Mach number limit. The technique is based on the projection method which had been developed to solve the Navier-Stokes equations for incompressible flow, modified to account for the non-zero divergence of the velocity. The technique is second-order accurate in both space and time. Calculations are presented for reacting flow in a closed container.

# Table of Contents

**Abstract** ..... 1

**Table of Contents** ..... iv

**List of Figures** ..... vi

**List of Tables** ..... vii

**Acknowledgments** ..... viii

**Introduction** ..... 1

**Chapter 1: Review of Previous Work** ..... 3

1.1: Low Mach Number Models ..... 3

1.2: Numerical Methods for Low Mach Number Combustion ..... 4

1.2.1: Lagrangian Particle Methods ..... 4

1.2.2: Compressible Flow Schemes ..... 5

1.2.3: All-speed algorithms ..... 6

1.3: Projection Methods ..... 7

**Chapter 2: The Equations for Low Mach Number Combustion** ..... 10

2.1: Scaling - Non-reacting Case ..... 10

2.2: Scaling - Reacting Case ..... 17

2.3: The Low Mach Number System of Equations for Reacting Flow ..... 23

2.4: A One Dimensional Example ..... 25

**Chapter 3: Projection Methods** ..... 28

3.1: Finite Difference Methods and Nomenclature ..... 28

3.2: Chorin's Method ..... 28

3.3: Discretizing the Projection ..... 29

3.4: Accuracy and Stability of the Approximate Projection Operator ..... 35

3.5: Filtering ..... 40

3.6: Algorithm for the Non-Reacting Case ..... 44

3.6.1: Time Stepping Strategy ..... 45

3.6.2: The Treatment of the Nonlinear Terms ..... 47

3.6.3: Pressure Correction ..... 53
3.7: Algorithm for the Reacting Case ..... 55
3.7.1: Velocity Predictor ..... 58
3.7.2: Scalar Predictor and Update ..... 58
3.7.3: Velocity and Pressure Update ..... 62
3.8: Computational Effort ..... 64
**Chapter 4: Results ..... 65**
4.1: Rayleigh Taylor Instability ..... 65
4.2: Numerical Convergence ..... 65
4.3: Combustion in a Closed Container ..... 68
4.4: CPU Time and Memory Requirements ..... 70
4.5: Future Work ..... 72
**References ..... 80**
**Appendix ..... 86**
A.1: The Multigrid Method ..... 86

# List of Figures

Figure 2.1: Heat release vs. X for a one-dimensional flame front ..... 26

Figure 2.2: Divergence vs. X for a one-dimensional flame front ..... 27

Figure 2.3: Velocity vs. X for a one-dimensional flame front ..... 27

Figure 3.1: Divergent mode left in by the MAC/centered difference projection ..... 41

Figure 3.2: Two-dimensional mode removed by the filter ..... 41

Figure 3.3: Location of ghost cells ..... 51

Figure 3.4: Predicted value extrapolation for the Godunov predictor ..... 52

Figure 4.1: Diagram of temperature variations and reaction zones for a laminar flame ..... 69

Figure 4.2: Bulk thermodynamic pressure vs. time for three uniform grids ..... 71

Figure 4.3: The $L_2$ norm of the mass fraction vs. time for three uniform grids ..... 71

Figure 4.4: Early evolution of the Rayleigh-Taylor instability showing the formation of two counter-rotating vortices into a mushroom shape ..... 74

Figure 4.5: Late time evolution of the Rayleigh-Taylor instability ..... 75

Figure 4.6: Flame front location for the case of combustion in a closed container ..... 76

Figure 4.7: Vorticity contours for combustion in a closed container ..... 77

Figure 4.8: Temperature contours for combustion in a closed container ..... 78

Figure 4.9: The scalar field of the velocity potential for combustion in a closed container ..... 79

Figure A.1: One Multigrid V-cycle ..... 87

# List of Tables

Table 4.1: Velocity Convergence Results - Non-reacting Flow ..... 66

Table 4.2: Temperature Convergence Results - Non-reacting Flow ..... 66

Table 4.3: Mass Fraction Convergence Results - Non-reacting Flow ..... 67

Table 4.4: Velocity Convergence Results - Reacting Case ..... 67

Table 4.5: Temperature Convergence Results - Reacting Case ..... 67

Table 4.6: Mass Fraction Convergence Results - Reacting Case ..... 68

Table 4.7: CPU and Memory Usage ..... 70

# Acknowledgments

Special thanks goes to Phil Colella for his special brand of mentoring which blends equal parts of good humor, wisdom, friendship, and understanding. Phil tirelessly read about three thousand drafts of this work, and barely complained. I couldn't possibly have had a better thesis advisor. I would also like to thank Alexandre Chorin and Phil Marcus for thoroughly reviewing this work and adding their constructive, insightful comments.

Financial support from the following are gratefully acknowledged: DARPA and the National Science Foundation under grant DMS-8919074; the National Science Foundation Presidential Young Investigator award under grant ACS-8958522; and the Department of Energy Office of Scientific Computing HPCC program under grant DE-FG03-92ER25140.

I am most grateful to my family who patiently waited and always lent their enthusiastic support, and I especially wish to recognize Nick, who withstood black moods, pretended to be interested in Godunov methods, and believed in me always. Without you, Nick, this victory would not be nearly as sweet.

# Introduction

The goal of this thesis is to develop a numerical method for solving the system of equations of reacting flow in the limit of zero Mach number. The method presented, a modified projection method, is second-order accurate in both time and space. It is, furthermore, restricted only by an advective CFL number and is, therefore, an efficient means of solving the reduced reacting flow equations. In the limit of zero Mach number, the time scale of the acoustics is much smaller than the advective time scale, and this fact is used to remove detailed acoustic wave effects from the system. Despite this simplification, the model correctly represents the effects of large temperature and density variations, substantial heat release, and baroclinic generation of vorticity.

Projection methods are a class of finite difference methods introduced by Chorin in 1968 to solve the Navier-Stokes equations for incompressible flow. The fractional step scheme he proposed involves first advancing the advection-diffusion equation in time to find an intermediate velocity field, and then, in turn, applying the divergence constraint by projecting this intermediate field onto the space of divergence-free vector fields.

Chorin's original method was first-order accurate in time and second-order accurate in space. Various investigators have extended the projection method to second order accuracy, including Bell et. al.[6], [7], Kim and Moin [42], Maday and Patera [47], and van Kan [64]. Bell and Marcus [9] applied the method to variable density flow. Here, the approach developed in Bell, Colella and Howell [7] is extended to solve the system of equations for reacting flow in the zero Mach number limit. The major modification required is a way of compensating for the inhomogeneous divergence condition, since the divergence of the velocity is no longer equal to zero.

This thesis is divided into four chapters. In Chapter 1, previous work on low Mach number models is reviewed and various numerical methods which have been used to treat low Mach number flow in the past are discussed. The chapter also includes a review of the development of the projection method.

In Chapter 2, the scaling argument presented in Rehm and Baum [55], Sivashinsky [60], and Majda and Sethian [48] which leads to the derivation of the limiting system of equations for low Mach number reacting flow is discussed. To illustrate the ideas, the $M \rightarrow 0$ limit for the simpler case of inviscid, compressible flow is discussed first, followed by a discussion of the more general

reacting case.

In Chapter 3, the numerical method which is the subject of this thesis is presented. Again, for clarity, the method for incompressible flow with variable density is discussed first, and then it is shown how to adapt the incompressible method to solve the enlarged set of equations for reacting flow.

Finally, in Chapter 4, results are presented and numerical convergence is verified. Two calculations are included. One calculation shows the development of a non-reacting Rayleigh-Taylor instability, while the second calculation is of reacting flow in a closed adiabatic container. The first calculation is compared to that of Bell and Marcus [9] for validation purposes.

# Chapter 1
# Review of Previous Work

## 1.1 Low Mach Number Models

Landau and Lifshitz [46] briefly discussed the conditions under which unsteady flow could be considered incompressible. First, the characteristic local fluid velocity, U, must be much less than the sound speed, c; i.e. the Mach number $M = \frac{U}{c}$ must be small. Second, if L is the length scale of the domain and T is a characteristic time scale for the fluid where $U \sim \frac{L}{T}$, then $T \gg \frac{L}{c}$. That is, the time taken by a sound wave to traverse a distance L must be small compared to the time scale during which the fluid undergoes significant changes. In particular, high frequency or long wavelength sound waves are excluded by the condition $U \sim L/T$ since the wavelength of sound waves is of the order $cT \sim \frac{UT}{M} = \frac{L}{M} \ll L$.

Ebin [27] and Klainerman and Majda [43] more rigorously explored the relationship between low Mach number flows and incompressible flows. Ebin gave a first analysis, while Klainerman and Majda gave a different, more complete analysis which included long wavelength acoustic corrections. They proved that the solution of the system of equations for compressible flow reduces to the solution of the Euler equations as $\sqrt{\gamma A}M \rightarrow 0$ for a gas with equation of state $p(\rho) = A\rho^{\gamma} + B$ where $A > 0$ and $\gamma > 1$. The proof was not extended to the non-isentropic case (and, therefore, not extended to reacting flow.)

Chorin [20] presented a numerical method for solving incompressible viscous flow problems which involved taking the limit $M \rightarrow 0$ numerically by introducing an artificial compressibility, $\delta$. The artificial compressibility was used as a parameter to speed up the convergence to a steady-state solution; the final solution was independent of $\delta$. Rehm and Baum [55] were the first to use a hydrodynamic model for reacting flow that analytically removed acoustic wave effects. They derived a system of equations by nondimensionalizing the compressible flow equations and examining them in the limit of zero Mach number. Viscous effects were neglected. Sivashinsky [60] independently proposed a similar model for slow flame propagation where flows are characterized by low Mach numbers. He investigated the general three-dimensional case and included transport, kinetic, and viscous effects. Sivashinsky derived a full system of equations consisting of conserva-

tion of mass, momentum, energy, and species, and an equation of state. In addition, he included a numerical solution to this system of equation for the case of combustion in a closed container with an infinitely thin flame. Ghoniem, Chorin and Oppenheim [30] used a low Mach number model as well in their investigation of turbulent flow in a combustion tunnel, where the effects of compressibility were represented by volume sources at the front.

Majda and Sethian [48] introduced a modified system of equations for low Mach number reacting flow that was more tractable from a numerical standpoint than the system introduced by Sivashinsky [60]. Their system replaced the continuity equation by an evolution equation for the bulk thermodynamic pressure. This switch simplified the design and implementation of their numerical scheme and made it easier to enforce the constraint imposed by the equation of state. The work in the present thesis is based on the system developed by Majda and Sethian.

## 1.2 Numerical Methods for Low Mach Number Combustion

### 1.2.1 Lagrangian Particle Methods

Only a few numerical algorithms have been developed to treat low Mach number reacting flow. One approach is to use a Lagrangian particle method such as that first introduced by ghoniem, Chorin, and Oppenheim in [30], and further developed by Ghoniem et. al.[31-37, 44, 50] and Sethian [58]. These methods are based on the random vortex method (RVM) [23] which models turbulent eddies using vortex elements which are followed about in time, and a volume-of-fluid flame propagation algorithm [24]. The main advantage of such an approach is that it is inherently adaptive, putting the computational effort only into the regions where the vorticity is non-zero. Another advantage of the RVM is that it is grid free, and thus does not display effects such as numerical viscosity.

Ghoniem, Chorin and Oppenheim [30] advance their solution using a fractional step scheme. In the first step, the vortex elements are advected and the divergence-free component of the velocity is found. In the second step, the flame is propagated and the gradient component of the velocity is calculated. This method was applied to the problem of combustion in an open channel where the flame is stabilized by a backward-facing step. The fluid model in [30] is based on the representa-

tion of the flame as a thin moving surface, with the fluid on either side being incompressible. The flow is taken to be two-dimensional, and the flame is treated as a constant-pressure deflagration wave propagating at a prescribed rate. Baroclinic effects are neglected. The calculations presented by [30] were able to capture many effects observed in turbulent combustion, such as the formation of large scale flow structures. These results also assisted in explaining the role of hydrodynamics in flame ignition and the mechanism behind exothermic processes. The calculations also helped to show the details of entrainment and mixing.

Sethian [58] also used the RVM coupled with a flame propagation algorithm to analyze combustion in open and closed vessels. He was able to show how the growth and development of counter-rotating turbulent eddies affect the flame, as well as how exothermal and viscous effects affect the burning front in combusting channel flow. Rhee, Sethian and Talbot [56] add a mechanism for computing the baroclinic generation of vorticity to their Lagrangian particle method used to compute an infinitely thin flame in an open channel. The vorticity across the flame front is calculated using the Hayes model.

To date, however, no calculations using Lagrangian particle methods have been performed for the case of low Mach number reacting flow in a closed container with baroclinic effects. Such methods have not been used because of the difficulty involved in calculating the baroclinic generation of vorticity across the flame front when the flame connects two regions of compressible gas with continuous density variations.

### 1.2.2 Compressible Flow Schemes

Another approach to numerically simulating low Mach number reacting flow is to use a scheme such as KIVA [3, 4, 5, 41, 51]. KIVA, an algorithm originally developed at Los Alamos National Laboratory, solves the unsteady equations for a multi-component, chemically reacting mixture of ideal gases using a finite difference methodology. A particle method is used for calculating liquid sprays, and ALE (Arbitrary Lagrangian-Eulerian), a finite volume method for arbitrary hexahedrons, is utilized as a gridding method. KIVA uses turbulence modelling to compute kinetic energy dissipation and solves for the pressure iteratively.

In the ALE method, spatial differences are calculated on a mesh consisting of arbitrary quadri-

lateral cells in two space dimensions and hexahedral cells in three dimensions. All thermodynamic variables are defined at cell centers. (In KIVA-II, the velocities are defined at cell centers as well. The usual practice for ALE is to define the velocity at cell vertices.) The computational mesh is moved in a purely Lagrangian way while the chemical source terms, the heat and mass diffusion terms, the pressure gradient, and the changes in density and internal energy due to the divergence of the velocity are calculated. Next, the solution is remapped onto a fixed Eulerian mesh.

To handle flows with low Mach numbers, KIVA uses an acoustic subcycling algorithm. The terms associated with acoustic wave propagation (such as the pressure gradient term in the momentum equation and the terms involving the divergence of the velocity in the conservation of mass and energy equations) are differenced with a time step $\Delta t_s$ which satisfies the CFL condition $\frac{c\Delta t_s}{\Delta x} \leq 1$, where c is the sound speed and $\Delta x$ is the mesh spacing. The remaining terms are differenced using time step $\Delta t$ which satisfies the constraint $\frac{U\Delta t}{\Delta x} \leq 1$, where U is the advective fluid velocity. $\Delta t_s$ and $\Delta t$ are related by $\frac{\Delta t_s}{\Delta t} \approx M$. For every time step $\Delta t$, approximately 1/M time steps of size $\Delta t_s$ are performed to advance those terms of order $M^2$. Thus, when the Mach number is small, this scheme is very inefficient. To solve this problem, the Mach number is artificially increased while all other dimensionless parameters are held fixed to reduce the number of iterations required. A disadvantage of acoustic subcycling is that short acoustic waves are not damped. To remedy the problem, an explicit damping term is added to the momentum equation.

KIVA's advantage is its flexibility. It has capabilities for solving problems with complex chemistry and complex geometry. Its efficiency and accuracy could be improved, however, for low Mach number flow.

### 1.2.3 All-speed algorithms

Another approach to calculating low Mach number reacting flow is the use of "all-speed" algorithms, schemes that are meant to model flows for any value of Mach number. All-speed schemes are based on the ideas presented in Chorin [20]. In such schemes, the full non-linear problem is solved, and the equations are regularized to deal with the stiffness which arises from disparate time scales at low Mach numbers. These schemes can be used for both inviscid and viscous flows, and for any value of Reynolds number.

Preconditioning techniques have been developed to compensate for the effects of eigenvalue stiffness. Some of these techniques are general while others focus on the low Mach number regime. The preconditioning essentially scales the equations of motion so that the eigenvalues become of the same order of magnitude. The solution is then advanced in scaled time. For unsteady flow, an additional "pseudo-time" term is added to the equations, and the solution is advanced in "pseudo" time. Examples of these schemes can be seen in Pletcher and Chen [54], by Briley, et. al., [15], Choi and Merkel [18, 19, 28], Dannenhoffer and Giles [26], Viviand et. al. [53, 67], Turkel [63], and van Leer [66].

The advantage of an all-speed calculation is just what the name implies: it can deal effectively with flows that simultaneously display low and high Mach number regions, for example, high speed flows with low velocities near stagnation points. The disadvantage of this approach is that it is difficult to solve the full non-linear system because of the complexity of the linear algebra.

## 1.3 Projection Methods

The projection method is a primitive variable finite difference method introduced by Chorin in [18, 22] to solve the incompressible Navier-Stokes equations. The method is a predictor-corrector scheme where the predictor step advances the solution of the advection-diffusion equation in time to yield an intermediate velocity field. In the corrector step, this field is projected onto the space of divergence-free vector fields to satisfy the divergence constraint. The original scheme was $O(\Delta t + \Delta x^2)$ for a domain with periodic boundary conditions in two or three dimensions, and required a fairly restrictive time step.

Gal-Chen and Somerville [29] implemented a variation on the projection method to solve the Navier-Stokes equations for topography using the staggered grid system of Harlow and Welsh [39] and explicit pressure terms in the predictor. Kim and Moin [42] extended the method to second-order accuracy using an inhomogeneous boundary condition for the intermediate velocity field. They treated the advective terms using a second-order explicit Adams-Bashforth scheme and used a second-order implicit Crank-Nicolson discretization for the viscous terms. As in [29], they utilized a staggered grid representation. Van Kan [64] proposed another second-order accurate variation of the projection method. In this variation, time-centered advective terms were obtained using

linearized difference approximations to calculate spatial gradients.

Maday and Patera [47] developed a spectral element projection method which converted the second-order Crank-Nicolson discretization of the advection-diffusion equation into a linear system for the coefficients of Fourier modes in each element. Another example of a spectral projection-type algorithm is the pressure-step last method (PSL) used by Coughlin and Marcus in [25] to compute turbulent bursts.

In [38], Gresho reviewed projection methods in an attempt to clarify issues relating to applying boundary conditions to intermediate velocity fields and viscous terms in fractional step schemes. He concluded that all intermediate boundary conditions create non-physical solutions, e.g., spurious boundary layers, at the walls, but accuracy is recovered by solving a higher-order pressure Poisson equation. Simo and Armero [59] reviewed and tested projection methods as well in their work examining long-term behavior and unconditional stability for time-stepping algorithms. They compared different versions of fractional step/projection methods to direct schemes with an augmented Lagrangian strategy in finding the solution to the driven cavity problem. They concluded that even though the projection methods they tested are unconditionally stable, these methods are limited by a restrictive time step, and thus are unattractive for use in simulations involving steady-state solutions.

Bell, Colella and Glaz (BCG) [6] extended Chorin's method to second-order accuracy by introducing more coupling between the diffusion-convection step and the projection step. Advective terms were treated using a second-order Godunov procedure. Bell, Colella and Howell [7] improved upon BCG's efficiency and robustness by altering the treatment of the advective terms to eliminate an instability that was observed for CFL numbers greater than 1/2. They accomplished this by adding an additional MAC-type projection. Bell and Marcus [9] extended the projection method to variable density flows by defining the discrete approximation of the projection operator using a discrete divergence-gradient pair that was adjoint with respect to a discrete density weighted vector inner product. Strikwerda [61] researched projection methods as well in an attempt to determine a spatial discretization of the projection operator for cases where all components of velocity are defined at cell centers. The discrete divergence-gradient pair he suggests are discrete adjoints with respect to scalar and vector inner products and lead to a projection that is

third-order accurate.

# Chapter 2
# The Equations for Low Mach Number Combustion

## 2.1 Scaling - Non-reacting Case

The purpose of this thesis is to introduce a method to numerically solve the limiting system of equations which describes chemically reacting flow as the Mach number approaches zero. In flows with very low Mach numbers, it is observed that the pressure in the system remains nearly constant in space, exhibiting only small fluctuations around a mean value. The acoustic waves in these systems are weak and equilibrate very rapidly. These flows are frequently found in common engineering devices such as pumps, burners and internal combustion engines.

The advantage of solving the reduced system of equations is that by analytically eliminating the acoustic wave effects, the time step in a numerical method will no longer be restricted by acoustic wave CFL constraints. It will, therefore, be possible to take time steps based on the fluid velocity, which will lead to an efficient numerical algorithm.

The overall goal is to analyze reacting flow, but it is illustrative to first show how the scaling works in the simpler case of inviscid, compressible flow. By considering the simpler case, it is possible to establish that the analog of low Mach number flow is incompressible flow. The approach will be to first scale the governing equations so that the scaled pressure is of the same order of magnitude as the advective scales. Then, asymptotic expansions for the dependent variables will be substituted into the scaled equations, and terms of equal order will be equated to determine expressions for some of the terms in the expansions. Finally, those terms which vanish as $M \to 0$ will be dropped from the scaled equations to yield the limiting system.

Consider first the inviscid, compressible, non-reacting flow in a closed container. Variables are defined as follows: $\mathbf{u} = (u, v)$ is the velocity, $\rho$ is the density, p is the pressure, t is the time, $\Omega$ is the bounded spatial domain, and c is the speed of sound. (Vectors are denoted by bold-faced type.) The equations describing inviscid, compressible flow are given by

continuity,

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

conservation of momentum,

$$\rho \frac{D\mathbf{u}}{Dt} + \nabla p = 0, \tag{2.2}$$

pressure equation,

$$\frac{Dp}{Dt} + \rho c^2 (\nabla \cdot \mathbf{u}) = 0. \tag{2.3}$$

The boundary condition for the flow in a closed container is the "no-flow" condition $\mathbf{u} \cdot \mathbf{n}|_{\partial\Omega} = 0$ which states that the normal component of the velocity must be zero on the boundary.

Consider an ideal gas with an equation of state $p = \rho RT$ where T is the temperature and R is the universal gas constant. The assumption is made that $\gamma$, the ratio of specific heats, is a constant and that $\gamma > 1$. For such a gas, the square of the sound speed is equal to $\gamma p/\rho$. It is therefore possible to replace $\rho c^2$ in the pressure equation by $\gamma p$.

Let L be a characteristic length scale based on the size of the domain, $\Omega$; U be a characteristic velocity based on the initial fluid velocity, $U = \max(U(x,0))$; and $\bar{\rho}$ be a characteristic density, defined by $\bar{\rho} = \int \rho(x,0)\, d\Omega$. $\bar{\rho}$, L and U are all of order unity. The behavior being investigated is that of the fluid on time scales comparable to the advective time scale, $T \sim \frac{L}{U}$. It is assumed that the ratio of local fluid velocity to local sound speed, $\frac{|U|}{c}$, is uniformly small, and that the variation of the dependent variables in x and t is of the order of the scales defined by L and T; for example, $\frac{\partial \mathbf{u}}{\partial t}$ is of order $\frac{L}{T^2}$. In particular, high frequency or long wavelength sound waves are excluded by the condition $U \sim L/T$. Sound waves which vary on time scales comparable to T have frequencies, f, which satisfy $Tf \sim 1$. If $\lambda$ is the wavelength of such sound waves, then $c \sim \lambda f \sim \frac{\lambda}{T}$, and $\lambda \sim \frac{L}{M} \gg L$.

A scaled pressure, $\tilde{p}$, is introduced, given by

$$\tilde{p} = \frac{p - \bar{p}}{\gamma \bar{p}} (\rho U^2), \tag{2.4}$$

where $\bar{p}$ is the initial mean pressure, defined by $\bar{p} = \int p(x,0)\, d\Omega$. It is possible to use $\bar{p}$ in the definition of sound speed, $c^2 = \gamma \bar{p}/\rho$, and define a Mach number based on this sound speed and the characteristic velocity, $M^2 = \bar{\rho} U^2/\gamma \bar{p}$. $\tilde{p}$ can then be expressed as

$$\tilde{p} = (p - \bar{p}) M^2. \tag{2.5}$$

With this scaling, $\tilde{p}$ is of order unity with respect to the advective scales defined above and $\tilde{p}/\bar{p} \sim O(M^2)$. Equation (2.5) can then be substituted into the momentum equation which reduces

to

$$\rho \frac{Du}{Dt} + \frac{1}{M^2} \nabla \tilde{p} = 0. \tag{2.6}$$

It is assumed that asymptotic expansions in $M \ll 1$ can be created for the scaled pressure, the velocity, and the density, of the form

$$\begin{aligned} \tilde{p}(x,t) &= p_0(x,t) + Mp_1(x,t) + M^2\pi(x,t) + O(M^3), \\ u(x,t) &= u_0(x,t) + Mu_1(x,t) + M^2u_2(x,t) + O(M^3), \\ \rho(x,t) &= \rho_0(x,t) + M\rho_1(x,t) + M^2\rho_2(x,t) + O(M^3), \end{aligned} \tag{2.7}$$

where each term, $p_i$, $u_i$ and $\rho_i$, is of order unity as $M \to 0$. The same expansions are assumed to hold for the derivatives; for example,

$$\begin{aligned} \frac{\partial \tilde{p}}{\partial x}(x,t) &= \frac{\partial p_0}{\partial x}(x,t) + M\frac{\partial p_1}{\partial x}(x,t) + M^2\frac{\partial \pi}{\partial x}(x,t) + O(M^3), \\ \frac{\partial \tilde{p}}{\partial t}(x,t) &= \frac{\partial p_0}{\partial t}(x,t) + M\frac{\partial p_1}{\partial t}(x,t) + M^2\frac{\partial \pi}{\partial t}(x,t) + O(M^3), \end{aligned} \tag{2.8}$$

etc. Note that equation (2.8) will not be valid if there are long wavelength acoustic waves present. If these expansion are substituted into the momentum equation (2.2), then

$$\begin{aligned} (\rho_0 + M\rho_1 + O(M^2))\frac{D}{Dt}(u_0 + Mu_1 + O(M^2)) &= \\ -\frac{1}{M^2}\nabla\,(p_0 + Mp_1 + M^2\pi_2 + O(M^3)). \end{aligned} \tag{2.9}$$

Terms of equal order in equation (2.9) can be equated to obtain expressions for individual terms in the expansions. By equating terms of order $M^{-2}$ and $M^{-1}$, it is seen that

$$M^{-2}: \qquad \nabla p_0 = 0 \tag{2.10}$$

and

$$M^{-1}: \qquad \nabla p_1 = 0. \tag{2.11}$$

Thus, the first two terms in the asymptotic expansion for the pressure, $p_0$ and $p_1$, are functions of time only. By equating terms of order $M^0$, the momentum equation is recovered

$$M^0: \qquad \rho_0 \frac{Du_0}{Dt} = \nabla \pi, \tag{2.12}$$

where $\pi/\bar{p} = O(M^2)$. Thus, the gradient of the order $M^2$ pressure fluctuations is the forcing

term in the momentum equation.

By combining equations (2.7) - (2.12), it is seen that the scaled pressure is given by

$$\tilde{p}(x,t) = p_0(t) + Mp_1(t) + M^2\pi(x,t) + O(M^3) \tag{2.13}$$

where the $\pi$ term is associated with the pressure gradient in the momentum equation.

It is possible to find more precise expressions for the terms in this expansion by repeating the process of equating terms of equal order in the Mach number in the pressure equation. To do so, one begins by replacing $p(x,t)$ by $(\frac{1}{M^2}\tilde{p} + \bar{p})$ in the pressure equation (2.3) and inserting the asymptotic expansions of $\tilde{p}$ and $\mathbf{u}$. This gives

$$\frac{1}{\gamma\left[\frac{1}{M^2}(p_0 + Mp_1 + M^2\pi + O(M^3)) + \bar{p}\right]}\frac{D}{Dt}\left[\frac{1}{M^2}(p_0 + Mp_1 + M^2\pi + O(M^3)) + \bar{p}\right] = -\nabla\cdot(\mathbf{u}_0 + M\mathbf{u}_1 + M^2\mathbf{u}_2 + O(M^3)). \tag{2.14}$$

Recall that $\tilde{p}/\bar{p} = O(M^2)$ and, therefore, that $p_0/\bar{p} = O(M^2)$ as well. By equating terms of order $M^0$, one finds

$$M^0: \qquad \frac{1}{\gamma(p_0/M^2 + \bar{p})}\frac{D(p_0/M^2 + \bar{p})}{Dt} = -\nabla\cdot\mathbf{u}_0 = f_0(t). \tag{2.15}$$

Since $p_0$ is a function of time only and $\bar{p}$ is a constant, $\nabla\cdot\mathbf{u}_0$ is also a function of time only; i.e., $-\nabla\cdot\mathbf{u}_0 = f_0(t)$. The no-flow boundary condition can then be applied to the velocity to show that $f_0(t)$ must be identically zero. That is,

$$0 = \int(\mathbf{u}_0\cdot\mathbf{n})\,dA = \int(\nabla\cdot\mathbf{u}_0)\,d\Omega = \int -f_0(t)\,d\Omega = -Cf_0(t) \tag{2.16}$$

where C is a constant equal to the volume of domain $\Omega$. Thus $f_0(t) = 0$ which implied that $Dp_0/Dt = 0$ and $\nabla\cdot\mathbf{u}_0 = 0$. If $Dp_0/Dt$ equals zero, then $p_0(t) = p_0(0)$. Since $\bar{p}$ is the mean pressure, $p_0 = 0$. By substituting these results into equation (2.14) the equation reduces to

$$\frac{1}{\gamma(\frac{1}{M}(p_1 + M\pi + O(M^2)) + \bar{p})}\frac{D}{Dt}(\frac{1}{M}(p_1 + M\pi + O(M^2)) + \bar{p}) = -\nabla\cdot(M\mathbf{u}_1 + M^2\mathbf{u}_2 + O(M^3)). \tag{2.17}$$

Repeating the argument in (2.15) and (2.16), it is seen that equating the terms of order $M^1$ in equation (2.14) gives

$$M^1: \qquad \frac{1}{\gamma M \bar{p}} \frac{D}{Dt} (p_1/M + \bar{p}) = -\nabla \cdot \mathbf{u}_1 = f_1(t) \tag{2.18}$$

which leads to the condition that $p_1(t) = p_1(0) = 0$ and $\nabla \cdot \mathbf{u}_1 = 0$.

After equating the remaining terms of order $M^2$, recalling that $\pi/\bar{p} = O(M^2)$, one finds

$$M^2: \qquad \frac{1}{\gamma \bar{p}} \frac{D\pi}{Dt} = -\nabla \cdot \mathbf{u}_2. \tag{2.19}$$

Since $\pi$ is a function of both space and time, so is $\nabla \cdot \mathbf{u}_2$ and thus the divergence of the velocity is $O(M^2)$ for $t \geq 0$. The continuity equation (2.1) shows that the material derivative of the density is $O(M^2)$ as well.

It has so far been demonstrated that the total pressure for low Mach number non-reacting flow takes the form

$$p(x,t) = \bar{p} + \pi(x,t) \tag{2.20}$$

where $\pi/\bar{p} = O(M^2)$. This scaling result is consistent with Bernoulli's equation for steady flow, which states that $\Delta p \sim \rho U^2$. It has also been shown that $\nabla \cdot \mathbf{u} = O(M^2)$.

By taking the limit $M \rightarrow 0$ and dropping the terms that vanish in equations (2.1) - (2.3), it is possible to obtain the following form of the equations for low Mach number flow:

$$\rho \frac{D\mathbf{u}}{Dt} + \nabla \pi = 0, \tag{2.21}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{2.22}$$

$$\frac{D\rho}{Dt} = 0. \tag{2.23}$$

If the density is initially constant in space, equation (2.23) implies that the density will remain constant for all time. For $\rho(x,0) = \rho_0 = \rho(x,t)$, equation (2.22) combined with equation (2.23) reduces further to the familiar form of the incompressible Euler equations for a constant density fluid

$$\rho_0 \frac{D\mathbf{u}}{Dt} + \nabla \pi = 0, \tag{2.24}$$

$$\nabla \cdot \mathbf{u} = 0. \tag{2.25}$$

The same scaling analysis can be performed in the case of viscous flow, where the momentum

equation takes the form

$$\rho \frac{\mathbf{Du}}{\mathbf{Dt}} = -\nabla\pi + \nabla \cdot \tau \tag{2.26}$$

where $\tau$ is the stress tensor, defined as $\tau = \mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T) + \zeta(\nabla \cdot \mathbf{u})$. $\mu$ is the ordinary or first coefficient of viscosity, and $\zeta$ is the second coefficient of viscosity.

The scaling argument can be repeated to arrive at the incompressible Navier-Stokes equations for an incompressible constant density fluid:

$$\rho_0 \frac{\mathbf{Du}}{\mathbf{Dt}} = -\nabla\pi + \nabla \cdot \mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T), \tag{2.27}$$

$$\nabla \cdot \mathbf{u} = 0. \tag{2.28}$$

Viscous terms in equation (2.27) have been simplified by using equation (2.28).

Equations (2.27) and (2.28) are fundamentally different from the equations for compressible flow. The compressible flow equations are evolution equations for $(\rho, \mathbf{u}, p)$ and, given initial conditions, they form a well-posed initial value problem. In contrast, the incompressible Navier-Stokes equations are a combination of an evolution equation and a constraint on the velocity that is independent of time. The presence of $\nabla\pi$ further complicates matters; there is no obvious evolution equation for the pressure.

Equations (2.27) and (2.28) can be better understood by introducing the concept of a projection. This concept is based on the Hodge (or Helmholtz) vector decomposition which splits an arbitrary vector field into two orthogonal components, one divergence-free and the other the gradient of a scalar field. That is, if $\mathbf{w} = \mathbf{w}(\mathbf{x})$ is some vector field defined on a spatial domain $\Omega$, then w can be written as

$$\mathbf{w} = \mathbf{u}_d + \nabla\varphi \tag{2.29}$$

where $\nabla \cdot \mathbf{u}_d = 0$, $\mathbf{u}_d \cdot \mathbf{n} = 0$ on $\partial\Omega$, and $\varphi$ is the solution to the elliptic equation

$$\Delta\varphi = \nabla \cdot \mathbf{w}, \qquad \left.\frac{\partial\varphi}{\partial \mathbf{n}}\right|_{\partial\Omega} = \mathbf{w} \cdot \mathbf{n}|_{\partial\Omega}. \tag{2.30}$$

This is an orthogonal decomposition; i.e.,

$$\int (\mathbf{u}_d \cdot \nabla\varphi)\, d\mathbf{x} = 0. \tag{2.31}$$

Given this decomposition, it is possible to define a projection operator, $\mathbf{P}$, such that $\mathbf{P}$ acting

on $\mathbf{w}$ will extract its divergence-free part, thus projecting $\mathbf{w}$ onto the space of divergence-free vector fields. That is, $\mathbf{P}(\mathbf{w}) = \mathbf{u}_d$. The relationship between the projection operator and the gradient component of $\mathbf{w}$ is $(\mathbf{I}-\mathbf{P})\mathbf{w} = \nabla\varphi$. Note that $\mathbf{P}$ operating on a gradient is zero, $\mathbf{P}(\nabla\psi) = 0$, and $\mathbf{P}$ operating on a divergence-free field leaves the field unchanged, $\mathbf{P}(\mathbf{u}_d) = \mathbf{u}_d$ for $\nabla\cdot\mathbf{u}_d = 0$, $\mathbf{u}_d\cdot\mathbf{n}|_{\partial\Omega} = 0$. Thus, $\mathbf{P}^2 = \mathbf{P}$.

The projection operator can be applied to the Navier-Stokes equations to obtain the projection formulation

$$\frac{\partial\mathbf{u}}{\partial t} = \mathbf{P}(-(\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla\cdot\tau). \tag{2.32}$$

This formulation converts the Navier-Stokes equations into a single evolution equation for the velocity for which the constraint (2.28) is automatically satisfied if $\nabla\cdot\mathbf{u} = 0$ at time $t = 0$. Given initial conditions that satisfy the constraint, this is a well-posed initial value problem. In addition, the projection formulation also enables us to understand more clearly the function of the pressure gradient which satisfies the equation

$$\nabla\pi = (\mathbf{I}-\mathbf{P})(-(\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla\cdot\tau). \tag{2.33}$$

Thus, $\nabla\pi$ is the gradient piece that is stripped away when the projection operator is applied to satisfy the divergence constraint.

A projection operator for variable density flow is defined in an analogous manner. It is possible to decompose a vector field $\mathbf{w} = \mathbf{w}(\mathbf{x})$ defined on a domain $\Omega$ into two orthogonal components as

$$\mathbf{w} = \mathbf{u}_d + \frac{1}{\rho}\nabla\varphi, \tag{2.34}$$

where $\nabla\cdot\mathbf{u}_d = 0$, $\mathbf{u}_d\cdot\mathbf{n} = 0$ on $\partial\Omega$, and $\varphi$ is the solution to the elliptic equation

$$L_\rho\varphi = \nabla\cdot\mathbf{w}, \qquad \left.\frac{1}{\rho}\frac{\partial\varphi}{\partial\mathbf{n}}\right|_{\partial\Omega} = \mathbf{w}\cdot\mathbf{n}|_{\partial\Omega}, \tag{2.35}$$

where

$$L_\rho\varphi = \nabla\cdot\left(\frac{1}{\rho}\nabla\varphi\right). \tag{2.36}$$

This is an orthogonal decomposition with respect to a density weighted inner product; i.e.,

$$\int (\mathbf{u}_d \cdot \frac{1}{\rho} \nabla \varphi) \rho d\mathbf{x} = 0. \tag{2.37}$$

Equation (2.34) is the analog of the Hodge decomposition for a variable density fluid.

As in the constant density case, it is possible to define a variable density projection operator, $\mathbf{P}_\rho$, such that $\mathbf{P}_\rho$ acting on $\mathbf{w}$ projects it onto the space of divergence-free vector fields. That is, $\mathbf{P}_\rho(\mathbf{w}) = \mathbf{u}_d$ and $(\mathbf{I} - \mathbf{P}_\rho)\mathbf{w} = \frac{1}{\rho}\nabla\varphi$. This projection operator can be applied to the variable density Navier-Stokes equations and the projection formulation can be obtained:

$$\frac{\partial \mathbf{u}}{\partial t} = \mathbf{P}_\rho(-(\mathbf{u} \cdot \nabla)\mathbf{u} + \frac{1}{\rho}\nabla \cdot \tau). \tag{2.38}$$

Once again, an evolution equation and a constraint independent of time have been converted into a single evolution equation for the velocity. The pressure gradient for variable density flow is given by

$$\frac{1}{\rho}\nabla\pi = (\mathbf{I} - \mathbf{P}_\rho)(-(\mathbf{u} \cdot \nabla)\mathbf{u} + \frac{1}{\rho}\nabla \cdot \tau). \tag{2.39}$$

Equivalently, if one has a guess for the pressure field, $\pi^G$, equation (2.39) can be written in correction form as

$$\frac{1}{\rho}\nabla\delta = (\mathbf{I} - \mathbf{P}_\rho)(-(\mathbf{u} \cdot \nabla)\mathbf{u} + \frac{1}{\rho}\nabla \cdot \tau - \frac{1}{\rho}\nabla\pi^G) \tag{2.40}$$

where $\delta$, the pressure increment, satisfies $\pi = \pi^G + \delta$. This form will be useful in discussing discretization methods in Chapter 3.

There are several ways of applying a projection operator to a vector field $\mathbf{w} = \mathbf{w}(\mathbf{x})$, but they are all mathematically equivalent to performing the following steps if $\mathbf{w}$ is sufficiently smooth: first, take the divergence of $\mathbf{w}$, thus forming the right-hand side to the elliptic equation (2.35) which can be solved for the scalar field, $\varphi$. Once $\varphi$ is known, $grad(\varphi)$ is easily constructed and can be subsequently subtracted from $\mathbf{w}$ to find the divergence-free component, $\mathbf{u}_d$.

## 2.2 Scaling - Reacting Case

The previous scaling argument can be extended to the case of reacting flow in a closed adiabatic container where the following assumptions are made:

1. Reactants go to products irreversibly in one step.

2. All fluid properties, e.g., the diffusion coefficients, $\kappa$ and $\mathcal{D}$, and the viscosities, $\mu$ and $\zeta$, are constant.

3. There are only two species of gas, burned and unburned, which have the same molecular weight. Having only two species will enable the conservation of species equation to be written in terms of a single variable, the mass fraction, defined as the fraction of unburned mass to total mass.

4. Both reactants and products are governed by the same $\gamma$-gas law.

5. Radiative heat transfer effects are negligible.

6. The gas is polytropic; the specific heats $c_p$ and $c_v$ are constant, and $\gamma = \frac{c_p}{c_v}$.

Since both the first and the second coefficients of viscosity are constant, the divergence of the stress tensor reduces to $\nabla \cdot \tau = \mu \Delta \mathbf{u} + \zeta \nabla (\nabla \cdot \mathbf{u})$. The pressure can be combined with the second term of $\nabla \cdot \tau$ to form a modified pressure given by

$$\pi := \pi + \zeta \nabla \cdot \mathbf{u}, \tag{2.41}$$

where the notation $A := B$ denotes "update in place" notation: the value of A is replaced by B. Thus the gradient terms in the momentum equation can be consolidated.

Given these assumptions, the system of equations that describe the flow is given by:

continuity,

$$\frac{D\rho}{Dt} + \rho (\nabla \cdot \mathbf{u}) = 0 \qquad \left.\frac{\partial \rho}{\partial \mathbf{n}}\right|_{\partial\Omega} = 0, \tag{2.42}$$

conservation of momentum,

$$\rho \frac{D\mathbf{u}}{Dt} = -\nabla \pi + \mu \Delta \mathbf{u} \qquad \mathbf{u} \cdot \mathbf{n}|_{\partial\Omega} = 0, \tag{2.43}$$

conservation of energy,

$$\rho c_p \frac{DT}{Dt} = \kappa \Delta T + q_0 k \rho Z + \Phi + \frac{Dp}{Dt} \qquad \left.\frac{\partial T}{\partial \mathbf{n}}\right|_{\partial\Omega} = 0, \tag{2.44}$$

conservation of species,

$$\rho \frac{DZ}{Dt} = \mathcal{D}(\nabla \cdot \rho \nabla Z) - k \rho Z \qquad \left.\frac{\partial Z}{\partial \mathbf{n}}\right|_{\partial\Omega} = 0, \tag{2.45}$$

ideal gas equation of state,

$$\rho = \frac{p}{RT} \tag{2.46}$$

where $c_p$ is the constant pressure specific heat, $\kappa$ is the coefficient of thermal diffusion, $q_0$ is a the heat of reaction, $k = k(T)$ is the reaction rate, $\mathcal{D}$ is the coefficient of species diffusion, T is the temperature, Z is the mass fraction, and $\Phi$ is the viscous energy dissipation given by

$$\Phi = \nabla \cdot (\mathbf{u} \cdot \tau) - \mathbf{u} \cdot (\nabla \cdot \tau). \tag{2.47}$$

For a two-dimensional flow, the expression for $\Phi$ reduces to

$$\Phi = 2\mu\left[ (\frac{\partial u}{\partial x})^2 + (\frac{\partial v}{\partial y})^2 + \frac{1}{2}(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x})^2 \right] + \zeta(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y})^2. \tag{2.48}$$

An asymptotic analysis similar to the one performed for the non-reacting case can be performed for the reacting case. Once again, units are selected so that $\bar{\rho}$, L and T are of order unity, $U \sim L/T$, and $M \ll 1$. A scaled pressure, $\tilde{p}$, is introduced, defined as it was in equation (2.4) where $\bar{p}$ is the mean pressure, $\bar{p}(t) = \int p(x,t)\, d\Omega$. By substituting $(\frac{1}{M^2}\tilde{p} + \bar{p})$ for $p(x,t)$ in the momentum equation (2.43) and equating the terms of equal order in the Mach number, it is possible to arrive at the same conclusion that was reached for the non-reacting case, namely that the first two terms in the asymptotic expansion for $\tilde{p}$, $p_0$ and $p_1$, are functions of time only. By balancing the terms of order $M^0$, it can be shown that

$$M^0: \qquad \rho_0 \frac{Du_0}{Dt} = \nabla\pi + \mu\Delta u_0. \tag{2.49}$$

For the reacting case, the total pressure is given as the sum

$$p(x,t) = \bar{p}(t) + \pi(x,t) \tag{2.50}$$

where $\bar{p}$, the mean pressure, is a bulk thermodynamic pressure, and $\pi$ denotes the pressure fluctuations. Again, $\pi$ is of order unity with respect to the advective scales and $\pi/\bar{p} = O(M^2)$.

Now that the scaling for the total pressure is known, it is possible to obtain a generalization of the incompressibility constraint (2.28). In particular, the thermodynamic relations can be analyzed to find an expression for the divergence of the velocity and an evolution equation for the bulk thermodynamic pressure. To begin the analysis, it is necessary to combine the first law of thermodynamics,

$$de = dq - pd\left(\frac{1}{\rho}\right), \tag{2.51}$$

the definition of enthalpy,

$$h = e + p/\rho, \tag{2.52}$$

and relationships for internal energy and enthalpy for a polytropic gas,

$$\begin{aligned} de &= c_v dT, \\ dh &= c_p dT, \end{aligned} \tag{2.53}$$

where e is the internal energy, h is the enthalpy, q is the heat, and $c_v$ is the specific heat for constant volume. From this combination, two expressions for energy are obtained:

$$\rho c_p dT = dq + dp \tag{2.54}$$

and

$$\rho c_v dT = \rho dq + \frac{p}{\rho} d\rho . \tag{2.55}$$

If these energy equations are used to describe Lagrangian parcels of fluid, all differentials may be replaced by material derivatives, i.e, $d\eta \rightarrow \frac{D\eta}{Dt} = \frac{\partial \eta}{\partial x} + (\mathbf{u} \cdot \nabla)\eta$. The result is two energy equations,

$$\rho c_v \frac{DT}{Dt} = \rho \frac{Dq}{Dt} + \frac{p}{\rho}\frac{D\rho}{Dt} \tag{2.56}$$

and

$$\rho c_p \frac{DT}{Dt} = \rho \frac{Dq}{Dt} + \frac{Dp}{Dt} . \tag{2.57}$$

The Lagrangian energy derivative, $\rho \frac{Dq}{Dt}$, can be replaced by terms describing heat creation, heat diffusion, and the viscous dissipation in the system. That is, $\rho \frac{Dq}{Dt}$ can be set to $\nabla \cdot (\kappa \nabla T) + q_0 k \rho Z + \Phi$ in equations (2.56) and (2.57) in light of the assumptions listed previously. After these substitutions, the following equations result:

$$\rho c_v \frac{DT}{Dt} = \frac{p}{\rho}\frac{D\rho}{Dt} + \kappa \Delta T + q_0 k \rho Z + \Phi \tag{2.58}$$

and

$$\rho c_p \frac{DT}{Dt} = \frac{Dp}{Dt} + \kappa \Delta T + q_0 k \rho Z + \Phi. \tag{2.59}$$

It follows from the equation of state that

$$\frac{T}{\bar{p}} = O(1). \tag{2.60}$$

Equation (2.59) can then be written

$$\frac{\rho c_p}{\bar{p}} \frac{DT}{Dt} = \frac{1}{\bar{p}} \frac{Dp}{Dt} + \frac{\kappa \Delta T}{\bar{p}} + \frac{q_0 k \rho Z}{\bar{p}} + \frac{\Phi}{\bar{p}}. \tag{2.61}$$

If $q_0/T$ is of order unity, then all terms in equation (2.61) are of order unity except for the viscous energy dissipation term: $\frac{\Phi}{\bar{p}}$ is of order $\mu|\nabla \mathbf{u}|^2$, which, from the argument presented in equations (2.4) - (2.20), is of order $\pi$ and $\frac{\pi}{\bar{p}} \sim O(M^2)$. Note that the assumption that $(\mu|\nabla \mathbf{u}|^2)/\bar{p} \sim O(M^2)$ is only good if the viscosity is sufficiently small. This would not be a good assumption, for example, to apply to Stokes flow. In addition, the pressure term in equation (2.61) can be expressed

$$\frac{1}{\bar{p}} \frac{Dp}{Dt} = \frac{1}{\bar{p}} \frac{D\bar{p}}{Dt} + O(M^2) = \frac{1}{\bar{p}} \frac{d\bar{p}}{dt} + O(M^2). \tag{2.62}$$

Thus, to leading order in $M^2$, equation (2.61) is given by

$$\rho c_p \frac{DT}{Dt} = \frac{d\bar{p}}{dt} + \kappa \Delta T + q_0 k \rho Z, \tag{2.63}$$

and, similarly, equation (2.58) is given by

$$\rho c_v \frac{DT}{Dt} = \frac{\bar{p}}{\rho} \frac{D\rho}{Dt} + \kappa \Delta T + q_0 k \rho Z. \tag{2.64}$$

Equation (2.64) can be used to find the constraint on the velocity divergence. By substituting the equation for conservation of mass, $\frac{1}{\rho} \frac{D\rho}{Dt} = -(\nabla \cdot \mathbf{u})$, into the energy equation (2.64) and solving for the velocity divergence, the result is

$$\nabla \cdot \mathbf{u} = \frac{1}{\bar{p}} \left(-\rho c_v \frac{DT}{Dt} + (\kappa \Delta T + q_0 k \rho Z)\right) \tag{2.65}$$

which can be written equivalently, with the definition $\gamma = c_p/c_v$, as

$$\nabla \cdot \mathbf{u} = \frac{1}{\gamma \bar{p}} \left(-\frac{d\bar{p}}{dt} + (\gamma - 1)(\kappa \Delta T + q_0 k \rho Z)\right). \tag{2.66}$$

Equation (2.66) is a constraint on the velocity divergence for the reacting flow case. The right hand

side of this equation can be viewed as a distributed volume source, where a positive contribution results in expansion of the gas and a negative contribution results in compression. It is expected that the largest variations in the temperature the combustive heat release will be in the region of the flame. Consequently, the volume source is largest in this region and the gas expands only at the flame front. Away from the front, the volume source is nearly equal to $-\frac{d\bar{p}}{dt}$ and the gas is being compressed.

The role of the temporal derivative of the mean pressure can be understood by examining its evolution equation. This equation can be derived by using equation (2.66) in conjunction with the no-flow boundary condition. From the boundary condition,

$$\int_{\partial\Omega} (\mathbf{u} \cdot \mathbf{n})\, dA = 0. \tag{2.67}$$

This boundary condition can be expressed equivalently as

$$\int_{\Omega} (\nabla \cdot \mathbf{u})\, d\Omega = 0. \tag{2.68}$$

Applying this form of the boundary condition to the right hand side of equation (2.66), the result is

$$\int_{\Omega} \frac{1}{\gamma\bar{p}} \left[ -\frac{d\bar{p}}{dt} + (\gamma - 1)\,(\kappa\Delta T + q_0 k\rho Z) \right] d\Omega = 0. \tag{2.69}$$

Since $\bar{p}$ is a function of time only, it can be taken outside the integral. Equation (2.69) will then reduce to an evolution equation for the mean pressure

$$\frac{d\bar{p}}{dt} = \frac{(\gamma - 1)}{\mathrm{vol}\,(\Omega)} \int_{\Omega} [\,\kappa\Delta T + q_0 k\rho Z]\, d\Omega. \tag{2.70}$$

Because the flow is contained in a closed adiabatic vessel, this equation can be reduced further to

$$\frac{d\bar{p}}{dt} = \frac{(\gamma - 1)}{\mathrm{vol}\,(\Omega)} \int_{\Omega} q_0 k\rho Z d\Omega. \tag{2.71}$$

Equations (2.70) and (2.71) show that mathematically, the role of thermodynamic pressure is to insure that the solvability condition for equation (2.66) is satisfied. A physical interpretation of the role of the thermodynamic pressure will be given in section 2.4.

As in the non-reacting case, the previous scaling analysis is only valid if equation (2.66) is satisfied by the initial velocity. If it is not, it is possible for large acoustic variations to develop and for

the flow to exhibit large-amplitude acoustic waves. This initial constraint is referred to in Majda and Sethian [48] as chemical fluid balance. Note that this is a condition that affects only the scalar potential portion of the velocity and is not, therefore, particularly restrictive. The initial divergence-free piece of the velocity, the piece which is associated with vortical motions, can be chosen arbitrarily.

## 2.3 The Low Mach Number System of Equations for Reacting Flow

The system of equations used in this investigation is based on the system developed by Majda and Sethian [48] where the variables $\mathbf{u}$, T, Z, $\rho$, and $\pi$, i.e., the velocity, the temperature, the mass fraction, the density, and the pressure, respectively, all vary in x and t. $\bar{p}$, the bulk thermodynamic pressure, varies in time only. The system is composed of the following equations:

conservation of momentum,

$$\frac{D\mathbf{u}}{Dt} = -\frac{1}{\rho}\nabla\pi + \frac{\mu}{\rho}\Delta\mathbf{u} \qquad \mathbf{u}\cdot\mathbf{n}|_{\partial\Omega} = 0\ , \tag{2.72}$$

conservation of energy,

$$\rho c_p \frac{DT}{Dt} = \frac{d\bar{p}}{dt} + \kappa\Delta T + q_0 k\rho Z \qquad \left.\frac{\partial T}{\partial \mathbf{n}}\right|_{\partial\Omega} = 0, \tag{2.73}$$

conservation of species,

$$\rho\frac{DZ}{Dt} = \mathcal{D}(\nabla\cdot\rho\nabla Z) - k\rho Z \qquad \left.\frac{\partial Z}{\partial \mathbf{n}}\right|_{\partial\Omega} = 0, \tag{2.74}$$

ideal gas equation of state,

$$\rho = \frac{\bar{p}}{RT}, \tag{2.75}$$

evolution equation for the bulk thermodynamic pressure,

$$\frac{d\bar{p}}{dt} = \frac{(\gamma-1)}{\mathrm{vol}(\Omega)}\int_{\Omega}(q_0 k\rho Z + \kappa\Delta T)\,d\Omega, \tag{2.76}$$

divergence constraint on the velocity,

$$\nabla \cdot \mathbf{u} = S = \frac{1}{\gamma \bar{p}} \left( -\frac{d\bar{p}}{dt} + (\gamma - 1)\,(q_0 k \rho Z + \kappa \Delta T) \right),$$
$$S = \Delta\varphi, \qquad \left.\frac{\partial \varphi}{\partial \mathbf{n}}\right|_{\partial\Omega} = 0. \tag{2.77}$$

Note that there is not a separate evolution equation for $\rho$. Instead, the density is determined by $\bar{p}$ and T by using the equation of state (2.75). Conservation of mass is imposed in the divergence constraint (2.77).

As in the reacting case, the momentum equation and the divergence constraint present a combination of an evolution equation for the velocity and a constraint on the velocity that is independent of time. A projection formulation was introduced in the non-reacting case to convert the pair into a evolution equation for the velocity for which the constraint is satisfied automatically. The identical formulation is not appropriate for the reacting case because $\mathbf{P}\left(\frac{\partial \mathbf{u}}{\partial t}\right) \neq \frac{\partial \mathbf{u}}{\partial t}$ (i.e., the velocity is not a divergence-free field.)

However, A variation of this formulation can be used. This variation is also based on the Hodge decomposition to solve equations of the form

$$\frac{D\mathbf{u}}{Dt} = -\nabla\pi + \mu\Delta\mathbf{u} \tag{2.78}$$

$$\nabla \cdot \mathbf{u} = S \tag{2.79}$$

where S denotes the volume source in equation (2.77).

The fluid velocity can be rewritten as the sum of two orthogonal components, one divergence-free and the other the gradient of a scalar field

$$\mathbf{u} = \mathbf{u}_d + \nabla\psi, \tag{2.80}$$

where $\nabla \cdot \mathbf{u}_d = 0$, $\mathbf{u}_d \cdot \mathbf{n}|_{\partial\Omega} = 0$, and, $\int (\mathbf{u}_d \cdot \nabla\psi)\, d\mathbf{x} = 0$. Unlike in the non-reacting case, $\nabla\psi$ is non-zero. The gradient $\nabla\psi$ satisfies the divergence constraint on the velocity by satisfying the relationship

$$\nabla \cdot \mathbf{u} = \Delta\psi = S. \tag{2.81}$$

Given this definition of the velocity, the momentum equation can be rewritten as

$$\frac{\partial \mathbf{u}}{\partial t} - \frac{\partial}{\partial t}(\nabla\psi) + \frac{\partial}{\partial t}(\nabla\psi) = -(\mathbf{u} \cdot \nabla)\,\mathbf{u} - \frac{1}{\rho}\nabla\pi + \frac{\mu}{\rho}\Delta\mathbf{u}. \tag{2.82}$$

By using the fact that $\mathbf{u} - \nabla\psi = \mathbf{u}_d$ is divergence-free and applying the projection, an evolu-

tion equation for u without the constraint may be obtained. This evolution equation is given by

$$\frac{\partial \mathbf{u}}{\partial t} = \frac{\partial}{\partial t}\nabla\psi + \mathbf{P}_\rho\left(-\frac{\partial}{\partial t}(\nabla\psi) - (\mathbf{u}\cdot\nabla)\,\mathbf{u} + \frac{\mu}{\rho}\Delta\mathbf{u}\right), \tag{2.83}$$

where

$$\frac{\partial}{\partial t}\nabla\psi = \nabla\Delta^{-1}\frac{\partial S}{\partial t}. \tag{2.84}$$

By applying the chain rule, $\frac{\partial S}{\partial t}$ can be expressed in terms of the spatial derivatives of T and Z as

$$\frac{\partial S}{\partial t} = \frac{\partial S}{\partial T}\frac{\partial T}{\partial t} + \frac{\partial S}{\partial Z}\frac{\partial Z}{\partial t} \tag{2.85}$$

where the temporal derivatives in T and Z can be evaluated in terms of spatial derivatives by using the governing equations (2.73) and (2.74).

Equation (2.83) is the analog of the projection formulation of the Navier-Stokes equations for the reacting case, and given initial conditions, forms a well-posed initial value problem. The equation for the pressure gradient corresponding to equation (2.39) is given by

$$\frac{1}{\rho}\nabla\pi = (\mathbf{I} - \mathbf{P}_\rho)\left(-\frac{\partial}{\partial t}(\nabla\psi) - (\mathbf{u}\cdot\nabla)\,\mathbf{u} + \frac{\mu}{\rho}\Delta\mathbf{u}\right). \tag{2.86}$$

Equivalently, given a guess for the pressure field, $\pi^G$, the following equation can be derived for $\delta = \pi - \pi^G$:

$$\frac{1}{\rho}\nabla\delta = (\mathbf{I} - \mathbf{P}_\rho)\left(-\frac{\partial}{\partial t}(\nabla\psi) - (\mathbf{u}\cdot\nabla)\,\mathbf{u} + \frac{\mu}{\rho}\Delta\mathbf{u} - \frac{1}{\rho}\nabla\pi^G\right). \tag{2.87}$$

## 2.4 A One Dimensional Example

A one-dimensional example is presented to illustrate the nature of the compressibility effects. Consider an adiabatic container with homogeneous boundary conditions for the velocity at x = 0 and x = 1. To the left of the center is hot burned gas. To the right of the center is cold unburned gas. Where the hot and cold gas meets in the center, heat will diffuse from the hot region to the cold region, and the burned and unburned gas will begin to diffuse into one another. When sufficient heat has diffused into the unburned gas to raise it to the auto-ignition temperature, a flame front will develop. Heat will be released only locally at this front, as a result of the chemical reaction. The heat release profile is illustrated in Figure 2.1.

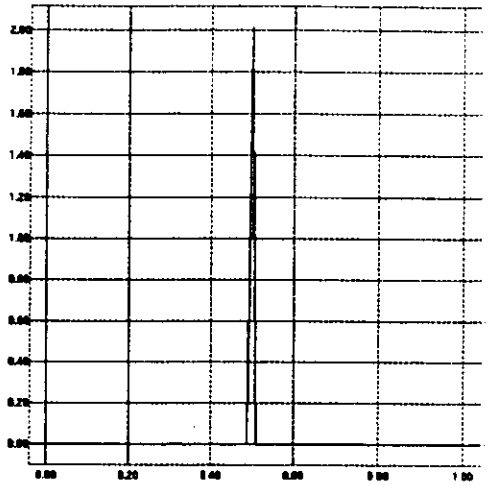


Figure 2.1 Heat release vs. X for a one-dimensional flame front

In one dimension, the velocity divergence is given by

$$\frac{\partial u}{\partial x} = \frac{1}{\gamma \bar{p}}\left(-\frac{d\bar{p}}{dt} + (\gamma - 1)\left(q_0 k \rho Z + \kappa \frac{\partial^2 T}{\partial x^2}\right)\right). \tag{2.88}$$

The divergence profile that corresponds to the heat release profile above is pictured in Figure 2.2. Note that the temporal derivative of the mean pressure adds a *uniform* contribution to the divergence, unlike the heat diffusion and energy source terms which are only non-zero at the location of the flame front.

Figure 2.2 shows clearly where compression is occurring and where expansion is occurring. From conservation of mass in one dimension, $\frac{\partial u}{\partial x} = -\frac{1}{\rho}\frac{D\rho}{Dt}$. Therefore, if $\frac{\partial u}{\partial x}$ is positive, $\frac{D\rho}{Dt}$ must be negative and expansion must be taking place. Conversely, if $\frac{\partial u}{\partial x}$ is negative, compression results. From the figure, the velocity divergence is seen to be negative except in the neighborhood of the flame front. Thus, the gas is being compressed everywhere except at the front, where the gas is expanding.

The velocity profile is shown in Figure 2.3. As indicated by the divergence profile, the velocity decreases linearly from zero from the left boundary to the flame front. At the front, the sign of the velocity flips from negative to positive, and the velocity once again decreases linearly to meet the

condition at the boundary.

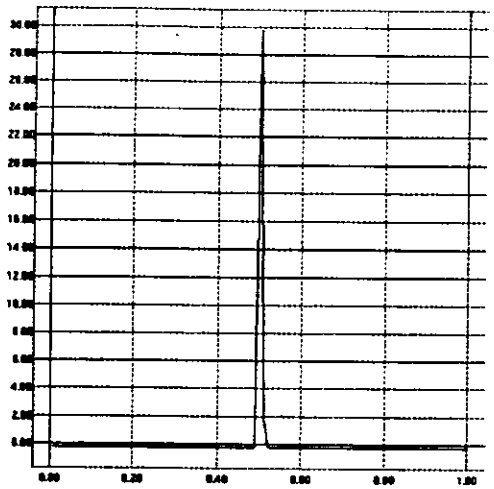


Figure 2.2 Divergence vs. X for a one-dimensional flame front

In summary, volume sources for the reacting flow are only non-zero in the vicinity of the flame front where the heat is being released. The thermodynamic pressure, however, is spatially uniform to $O(M^2)$, which means that this heat release results in a *uniform* compression within the spatial domain. This compression exactly balances the expansion that occurs along the flame front.

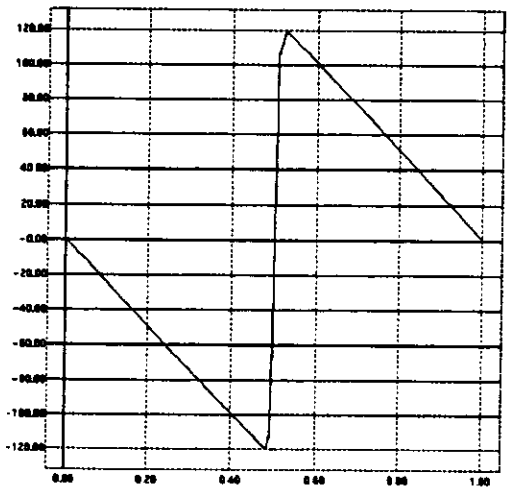


Figure 2.3 Velocity vs. X for a one-dimensional flame front

# Chapter 3
# Projection Methods

## 3.1 Finite Difference Methods and Nomenclature

Finite difference methods are a class of numerical methods for solving differential equations in which the derivatives in the governing equations and boundary conditions are replaced by divided difference approximations. Time is divided into small increments of $\Delta t$ called time steps. It is assumed that at the beginning of each time step, at time t, a discrete approximation to the solution which satisfies appropriate boundary conditions is known. The object of the scheme is to advance the solution in time and obtain an updated solution at time $t^n + \Delta t$. As many times step as necessary are performed until the desired final time is reached.

The spatial domain is discretized as well into finite difference cells, each with area $\Delta x \times \Delta y$. The center of each cell is located at $((i - 1/2)\Delta x,\ (j - 1/2)\Delta y)$ and is denoted as cell (i,j) where i = 1,..., nx; j = 1,..., ny. The left edge of each cell is located at $i - 1/2, j$, the right edge at $i + 1/2, j$, the top of the cell at $i, j + 1/2$, and the bottom at $i, j - 1/2$. Discrete variables are indexed by location and time. i and j are spatial indices, written as subscripts, which correspond to particular finite difference cells. n is a time index, written as a superscript, which corresponds to a particular time step.

## 3.2 Chorin's Method

Chorin [21] presented a numerical scheme to solve the Navier-Stokes equations based on the form written in equations (2.38) and (2.39). This algorithm is a fractional step scheme; in the first step, nonlinear and viscous terms are evaluated and used to advance the advection-diffusion equation in time by solving

$$bU^* - BU^n = F(U), \tag{3.1}$$

where $U^n = (u^n, v^n)$ is the discrete velocity at time $t^n$, $TU \equiv bU^{n+1} - BU^n$ is an approximation for $\frac{\partial u}{\partial t}$, and F(U) is a discrete approximation for the advective, viscous, and body force terms. An example of a discretization for equation (3.1) is

$$U^* = U^n + \Delta t\,[-(U \cdot \nabla)\,U + \mu \Delta U]\,. \tag{3.2}$$

The result of solving equation (3.1) is an intermediate velocity field, $U^*$, which does not, in general, satisfy the divergence constraint. The second step involves applying the projection operator to enforce the constraint, and using the divergence-free vector component to update the velocity.

$$U^{n+1} = \mathbf{P}(U^*) \tag{3.3}$$

There are several issues that must be addressed in implementing the projection method. One is the discretization of the projection operator, specifying how exactly $\mathbf{P}$ is going to return a discretely divergence-free vector field and defining "discretely divergence-free". Another issue is temporal discretization. Chorin's method is second-order accurate in space but only first-order accurate in time because the projection operator does not commute with the Laplace operator in the presence of boundaries. A second-order accurate method is desired. Spatial discretization is a third issue. The goal is to use high-resolution finite difference methods to discretize the advective terms which work well in the presence of strong gradients. Finally, a method that is restricted only by the advective CFL number, regardless of the value of viscosity, is desired.

## 3.3 Discretizing the Projection

The discretization of the projection operator is based on the discrete Hodge decomposition, which decomposes an arbitrary discrete vector field into two orthogonal components, one discretely divergence-free and the other the discrete gradient of a scalar. That is,

$$W = U_d + G\varphi \tag{3.4}$$

where $DU_d = 0$, and D and G are difference approximations to *div* and *grad*.

Chorin discretized the projection operator using operators that were adjoints with respect to an appropriate set of vector and scalar inner products. More precisely, he used discrete divergence and gradient operators that satisfied the adjoint relationship $D = -G^T$. The discrete Laplacian was defined by $L^h = DG$.

Applying a projection operator can be done in several ways, but all of them are mathematically equivalent to performing the following steps:

1) Solve

$$L^h\varphi = DW. \tag{3.5}$$

2) Find $U_d = W - G\varphi$.

Equation (3.5) is always solvable even though $L^h$ is not invertible; DW is in the range of $L^h$ because of the adjoint relationship $D = -G^T$. Also, despite the fact that the solution to equation (3.5) is not unique, $G\varphi$ is independent of the particular choice of solution to equation (3.5).

The projection operator has several properties. First, because of the adjoint relationship, $\mathbf{P} = \mathbf{P}^T$. Second, because it is a projection, the discrete operator satisfies $\mathbf{P}^2 = \mathbf{P}$. Third, it is easy to show using these two properties that

$$\|\mathbf{P}(W)\| \le \|W\|, \tag{3.6}$$

where $\| \ \|$ is the discrete $L^2$ norm derived from the vector inner product, $\langle U, V\rangle = \sum_{i,j} (U_{i,j} \cdot V_{i,j})\,\Delta x \Delta y$, and $\|Q\| = \langle Q, Q\rangle^{1/2}$. We shall refer to such operators as *discrete projection operators.*

Unfortunately, the discretization of the projection operators has some serious drawbacks for the case where the components of the velocity are centered at the same location, e.g., $w((i-1/2)\Delta x, (j-1/2)\Delta y) \approx W_{i,j} = (u_{i,j}, v_{i,j})$. One disadvantage is that an operator where $L^h = DG$ can lead to non-standard Poisson operators which complicate the linear algebra. For example, a divergence - gradient pair made of central difference operators yields an $L^h$ that is locally decoupled into independent operators defined on each of four disjoint subgrids. Another example is the complicated linear algebra which results from the discrete projection operator formed from the non-symmetric D and G operators introduced by Strikwerda [61]. This discrete projection operator is expensive to compute and is not robust in the presence of large density variations. More precisely, to compute a Strikwerda projection, it is necessary to take higher order differences of the density field, which, for large density variations, can lead to the generation of spurious high wavenumber modes, even for a smooth initial right hand side of equation (3.5).

The approach taken in this thesis is not to use a discrete projection operator where $\mathbf{P}^2 = \mathbf{P}$, but rather to use a consistent discretization of the continuous projection operator where $L^h \neq DG$, and, thus, an operator where $\mathbf{P}^2 \neq \mathbf{P}$. We shall refer to these discretizations as *approximate projection operators.* The use of approximate projection operators is an approach that has been taken by Almgren, et. al. [2] where the discretization of the operator is based on finite elements. In this

thesis, a discretization is used for the projection where the divergence and gradient are defined using centered differences, and $L^h$ is discretized by a standard five-point representation. The approximate projection is stable; a relationship analogous to equation (3.6) will be derived in section 3.4.

To apply the approximate projection operator to a discrete vector field $W = U_d + G\varphi$, several steps are performed. First, the Poisson equation

$$\Delta^h \varphi = D_0 W \tag{3.7}$$

is solved for $\varphi$. The divergence-free field is then found by subtracting $G\varphi$ from W. The result is $U_d$, the projection of vector field W. That is, $\mathbf{P}(W)$ is given by

$$\mathbf{P}(W) = U_d = W - G_0\varphi. \tag{3.8}$$

For variable density flow, first solve the Poisson equation

$$L_\rho^h \varphi = D_0 W. \tag{3.9}$$

The divergence-free field is then found by subtracting $G_{0,\rho}\varphi$, a discrete approximation to $\frac{1}{\rho}\nabla\varphi$, from W

$$\mathbf{P}(W) = U_d = W - G_{0,\rho}\varphi. \tag{3.10}$$

Next, the operators used in equations (3.7) - (3.10) are defined. The discrete divergence operator is given by

$$D_0 U = \frac{u_{i+1,j} - u_{i-1,j}}{2\Delta x} + \frac{v_{i,j+1} - v_{i,j-1}}{2\Delta y}. \tag{3.11}$$

The discrete gradient operator for the variable density case, $G_{0,\rho}$, is given by

$$(G_{0,\rho}\varphi)_{i,j} = \left(\frac{1}{\rho_{i,j}}\frac{\varphi_{i+1,j} - \varphi_{i-1,j}}{2\Delta x}, \frac{1}{\rho_{i,j}}\frac{\varphi_{i,j+1} - \varphi_{i,j-1}}{2\Delta y}\right). \tag{3.12}$$

$G_0$, the discrete gradient operator for the constant density case, is equal to $G_{0,\rho}$ if the density is set to unity. $\Delta^h$, the discrete elliptic operator for the constant density case, is a cell-centered five-point Laplacian given by

$$(\Delta^h\varphi)_{i,j} = \frac{1}{\Delta x^2}(\varphi_{i+1,j} - 2\varphi_{i,j} + \varphi_{i-1,j}) + \frac{1}{\Delta y^2}(\varphi_{i,j+1} - 2\varphi_{i,j} + \varphi_{i,j-1}). \tag{3.13}$$

$L_\rho^h$, the discrete elliptic operator for the variable density case which corresponds to a discretization of $L_\rho$ in equation (2.36), is given by

$$(L_\rho^h\varphi)_{i,j} = \frac{1}{\Delta x^2}\left(\frac{(\varphi_{i+1,j} - \varphi_{i,j})}{\rho_{i+1/2,j}} - \frac{(\varphi_{i,j} - \varphi_{i-1,j})}{\rho_{i-1/2,j}}\right) + \frac{1}{\Delta y^2}\left(\frac{(\varphi_{i,j+1} - \varphi_{i,j})}{\rho_{i,j+1/2}} - \frac{(\varphi_{i,j} - \varphi_{i,j-1})}{\rho_{i,j-1/2}}\right). \tag{3.14}$$

The density values at cell edges in equation (3.14) are obtained by averaging the values of density in adjacent cells. For edge i+1/2, j, the averaged density is expressed by

$$\rho_{i+1/2,j} = \frac{1}{\frac{1}{2}\left(\frac{1}{\rho_{i,j}} + \frac{1}{\rho_{i+1,j}}\right)}. \tag{3.15}$$

Three sets of boundary conditions are required to complete a projection: a set for $D_0$, a set for $G_0$, and a set for solving the Poisson equation [(3.7) or (3.9).] $D_0$ and $G_0$ carry their own boundary conditions, where those for $D_0$ are the physical boundary conditions for the problem, and those for $G_0$ are non-physical and are set using second-order polynomial extrapolation. The Poisson equation is solved using homogeneous Neumann boundary conditions.

Boundary conditions are applied to the divergence operator by setting fluxes at the boundary edges. $D_0$ is defined in terms of fluxes as

$$(D_0U)_{i,j} = \frac{F_{i+1/2,j} - F_{i-1/2,j}}{\Delta x} + \frac{F_{i,j+1/2} - F_{i,j-1/2}}{\Delta y} \tag{3.16}$$

where F denotes flux and the definition of flux on the domain interior edges is

$$\begin{aligned} F_{i+1/2,j} &= \frac{1}{2}(u_{i+1,j} + u_{i,j}) \qquad i = 1, \ldots, nx-1, \\ F_{i,j+1/2} &= \frac{1}{2}(v_{i,j+1} + v_{i,j}) \qquad j = 1, \ldots, ny-1. \end{aligned} \tag{3.17}$$

For example, to apply the physical boundary condition $W \cdot n = 0$, the flux at the edges of the domain is set to zero; e.g., $F_{1/2,j} = F_{nx+1/2,j} = F_{i,1/2} = F_{i,ny+1/2} = 0$.

Boundary conditions are applied to the gradient operator using second-order polynomial extrapolation to extend values of the scalar field, $\varphi$, to the first row of cells outside the domain. These extended values are then used to determine the gradient in the first row of cells in the domain interior. For example, the cells just outside the left hand side of the domain are calculated by

$$\varphi_{0,j} = 3(\varphi_{1,j} - \varphi_{2,j}) + \varphi_{3,j}, \tag{3.18}$$

and the expression for the gradient at the left-most interior cell is given by

$$(G_{0,\rho}\varphi)^x_{1,j} = \frac{1}{\rho_{1,j}} \frac{\varphi_{2,j} - \varphi_{0,j}}{2\Delta x}. \tag{3.19}$$

Second-order extrapolation of the scalar field is equivalent to a first-order extrapolation of the gradients on the mesh to determine the gradient at the boundary. In other words, $G_0$ can also be defined as the average of mesh-centered gradients

$$\begin{aligned} (G_{0,\rho}\varphi)^x_{i,j} &= \frac{1}{2\rho_{i,j}}\left(G^M_{i+\frac{1}{2},j} + G^M_{i-\frac{1}{2},j}\right), \\ (G_{0,\rho}\varphi)^y_{i,j} &= \frac{1}{2\rho_{i,j}}\left(G^M_{i,j+\frac{1}{2}} + G^M_{i,j-\frac{1}{2}}\right), \end{aligned} \tag{3.20}$$

where

$$G^M_{i+\frac{1}{2},j} = \frac{\varphi_{i+1,j} - \varphi_{i,j}}{\Delta x}. \tag{3.21}$$

These mesh-centered gradients can be linearly extrapolated to the domain boundary. Thus, the gradient at the left edge of the domain, $G^M_{1/2,j}$, can be expressed

$$G^M_{1/2,j} = 2G^M_{3/2,j} - G^M_{5/2,j}. \tag{3.22}$$

The gradient on the first row of cells in the interior is then calculated using these extrapolated values. The cells on the left interior edge, for example, are calculated by

$$(G_{0,\rho}\varphi)^x_{1,j} = \frac{1}{2\rho_{i,j}}(G^M_{3/2,j} + G^M_{1/2,j}). \tag{3.23}$$

Homogeneous Neumann boundary conditions are applied to $L^h_\rho$ in a similar way. A finite difference approximation is used to find the value of the scalar field in the cells just outside the

domain, and these values, in turn, are used to calculate $L_\rho^h$ at the interior edge cells. For example, the boundary condition $\left.\frac{\partial\varphi}{\partial x}\right|_{left} = 0$ is enforced at the left edge by setting cells just outside using the finite difference approximation for gradient on the boundary

$$\left.\frac{\partial\varphi}{\partial x}\right|_L \approx \frac{\varphi_{1,j} - \varphi_{0,j}}{\Delta x} = 0. \tag{3.24}$$

Thus, $\varphi_{0,j}$ is set to $\varphi_{1,j}$, etc.

Bell, Colella and Glaz [6] discuss at some length the issue of accuracy for the projection given sufficient smoothness of the pressure in the absence of boundaries. Their analysis, however, does not extend to the case where there are boundaries since, in that case, operators do not commute. They conclude that it is difficult to determine the order of accuracy of the projection with boundary conditions; an $O((\mu\Delta t)^{1/4})$ error term seems to be introduced in the pressure, and an $O(\mu\Delta t)$ error term seems to be introduced in the velocity. Both terms appear to be an artifact of the analysis since it is observed that their numerical convergence study indicates sufficient accuracy. Second-order accuracy is observed as well for the algorithm presented in this thesis, as discussed in section 4.2.

Regarding the question of solvability, note that $L_\rho^h$ with boundary conditions has a one-dimensional null space consisting of scalar fields that are constant throughout the mesh. Equation (3.9) is solvable, however; $D_0W$ is in the range of $L_\rho^h$ and is orthogonal to constant scalar fields which are the null space of $L_\rho^h$. That is, if the discrete scalar inner product is defined by $\langle\psi, \varphi\rangle \equiv \sum_{i,j} \psi_{i,j}\varphi_{i,j}\Delta x\Delta y$, then

$$\langle 1, D_0W\rangle = 0. \tag{3.25}$$

Furthermore, the non-uniqueness of the solutions to equation (3.9) poses no difficulty. Solutions will be unique up to an additive constant, and since $G_0\varphi = G_0(\varphi + c)$, the gradient will be independent of the choice of solution.

Next, it is shown that the solution obtained by the procedure described above gives a consistent discretization to equation (2.35),

$$L_\rho\varphi = \nabla\cdot\mathbf{w}, \qquad \frac{1}{\rho}\frac{\partial\varphi}{\partial n} = \mathbf{w}\cdot\mathbf{n} \text{ on } \partial\Omega, \tag{3.26}$$

where $L_\rho\varphi$ is defined in equation (2.36) and we assume $\int(\mathbf{w}\cdot\mathbf{n})\,dA = 0$ so that equation (3.26)

is solvable. The problem can be split into two pieces,

$$L_\rho \varphi_\Omega = \nabla \cdot \mathbf{w}, \qquad \left.\frac{\partial \varphi_\Omega}{\partial \mathbf{n}}\right|_{\partial\Omega} = 0, \tag{3.27}$$

and

$$L_\rho \varphi_{\partial\Omega} = 0, \qquad \frac{1}{\rho}\frac{\partial \varphi_{\partial\Omega}}{\partial \mathbf{n}} = \mathbf{w} \cdot \mathbf{n} \text{ on } \partial\Omega, \tag{3.28}$$

where $\varphi = \varphi_\Omega + \varphi_{\partial\Omega}$ satisfies equation (3.26). It can be shown that the solution to equation (3.9) can be written as an analogous sum: if $\varphi$ satisfies equation (3.9), then $\varphi = \varphi_I + \varphi_B$ where

$$L_\rho^h \varphi_I = D_I W \tag{3.29}$$

is a consistent discretization of equation (3.27), and

$$L_\rho^h \varphi_B = D_B W \tag{3.30}$$

is a consistent discretization of equation (3.28). $D_I W$ is defined by equation (3.16), except the fluxes at the boundary are set by linear extrapolation. For example, the flux at the left boundary is set by

$$F_{1/2,j} = 2F_{3/2,j} - F_{5/2,j}. \tag{3.31}$$

$D_B W$ is then defined as $D_B W = D_0 W - D_I W$. It can be shown that $D_B W = 0$, except at cells adjacent to the boundary, where

$$(D_B W)_{1,j} = \frac{F_{1/2,j}}{\Delta x}, \qquad D_B W_{nx,j} = -\frac{F_{nx+1/2,j}}{\Delta x}, \tag{3.32}$$

etc. If $W \cdot \mathbf{n} = 0$ on $\partial\Omega$, then $\varphi_B$ is zero and $\varphi = \varphi_I$. However, if $W \cdot \mathbf{n} \neq 0$ on the boundary, then $\frac{1}{\rho} G_0 \varphi_B$ is the potential field that corrects for the inhomogeneous boundary condition and accounts for the extent to which $W \cdot \mathbf{n} \neq 0$.

## 3.4 Accuracy and Stability of the Approximate Projection Operator

In this section, it will be shown that the approximate projection operator defined above is stable; that is,

$$\|\mathbf{P}\| \leq 1 \tag{3.33}$$

for the case of a constant density fluid with doubly periodic boundary conditions. This is one of the

key properties used by Chorin in proving the convergence of numerical methods used to solve the Navier-Stokes equation in [22]. It will also be shown that the approximate projection operator is a second-order accurate discretization of the continuous operator defined in equations (2.29) - (2.39). Note that the analysis that follows is only valid for the constant density case and does not easily extend to the case of a variable density fluid.

Fourier analysis will be used to prove stability, where continuous Fourier modes are given by

$$\mathbf{w}^{\mathbf{k}}(\mathbf{x}) = e^{i(\mathbf{k}\cdot\mathbf{x})} \tag{3.34}$$

and $i = \sqrt{-1}$; $\mathbf{k} = (k_1, k_2)$ and $k_i$ is any integer; and $\mathbf{x} = (x, y)$. The discrete analysis begins by considering the discrete velocity vector, $U = \{U_{i,j}\}$ where $i = 1, \ldots, N$, $j = 1, \ldots, N$ and N is even. The velocity can be expanded in discrete Fourier modes,

$$\omega_{\mathbf{j}}^{\mathbf{k}} = \omega^{\mathbf{k}}(\mathbf{j}h) = e^{i(\mathbf{k}\cdot\mathbf{j})h}, \tag{3.35}$$

where $\mathbf{k} = (k_1, k_2)$ and $k_i = -\frac{N}{2} + 1, \ldots, \frac{N}{2}$; $\mathbf{j} = (j_1, j_2)$, $j_i = 0, \ldots, N-1$; and $\Delta x = \Delta y = h = 2\pi/N$. The set of Fourier modes is orthonormal on $[0, 2\pi]$. The discrete transform of the velocity is

$$\hat{U}^{\mathbf{k}} = (\hat{u}^{\mathbf{k}}, \hat{v}^{\mathbf{k}}) = \sum_{\mathbf{j}} U_{\mathbf{j}} \omega_{\mathbf{j}}^{\mathbf{k}} h^2. \tag{3.36}$$

where $U_{\mathbf{j}} = (u_{\mathbf{j}}, v_{\mathbf{j}})$ are the coefficients in the expansion.

Fourier transforms are useful for examining the stability of operators because discrete Fourier modes are the eigenvectors of shift operators, the basic building blocks of finite difference operators. The shift operators $S_x^+$, $S_x^-$, $S_y^+$, $S_y^-$ are defined by

$$\begin{aligned}
(S_x^+ u)_{i,j} &= \{u_{i+1,j}\}, \\
(S_x^- u)_{i,j} &= \{u_{i-1,j}\}, \\
(S_y^+ v)_{i,j} &= \{v_{i,j+1}\}, \\
(S_y^- v)_{i,j} &= \{v_{i,j-1}\}.
\end{aligned} \tag{3.37}$$

All linear finite difference operators can be expressed as linear combinations of these shifts. Taking the discrete Fourier transform of equations (3.37) gives

$$
\begin{aligned}
S_x^+ \omega_j^k &= \omega_{j_1+1,\, j_2}^k = e^{ik_1 h} \omega_j^k, \\
S_x^- \omega_j^k &= \omega_{j_1-1,\, j_2}^k = e^{-ik_1 h} \omega_j^k, \\
S_y^+ \omega_j^k &= \omega_{j_1,\, j_2+1}^k = e^{ik_2 h} \omega_j^k, \\
S_y^- \omega_j^k &= \omega_{j_1,\, j_2-1}^k = e^{-ik_2 h} \omega_j^k.
\end{aligned}
\tag{3.38}
$$

Therefore, if $V = LU$, where L is a constant coefficient finite difference operator, then taking the Fourier transform of V is the same as multiplying $\hat{U}$ by an amplification factor, or symbol, composed of a linear combination of $e^{ik_1 h}$, $e^{-ik_1 h}$, $e^{ik_2 h}$ and $e^{-ik_2 h}$. (For more information on Fourier analysis, see Strikwerda [62].)

To begin the stability analysis of the approximate projection operator, consider the discrete centered difference divergence operator, $D_0$, which is given by

$$
D_0 = \begin{pmatrix} \dfrac{S_x^+ - S_x^-}{2h} & \dfrac{S_y^+ - S_y^-}{2h} \end{pmatrix}. \tag{3.39}
$$

The symbol of $D_0$, $\hat{D}_0$, can be expressed as

$$
\hat{D}_0(k) = (\hat{\delta}_0^x \quad \hat{\delta}_0^y), \tag{3.40}
$$

where

$$
\hat{\delta}_0^x = \frac{e^{ik_1 h} - e^{-ik_1 h}}{2h} \tag{3.41}
$$

and

$$
\hat{\delta}_0^y = \frac{e^{ik_2 h} - e^{-ik_2 h}}{2h}. \tag{3.42}
$$

Likewise, the discrete centered difference gradient, $G_0$, can be given by

$$
G_0 = \begin{pmatrix} (S_x^+ - S_x^-)/(2h) \\ (S_y^+ - S_y^-)/(2h) \end{pmatrix}. \tag{3.43}
$$

The symbol of $G_0$, $\hat{G}_0$, can be expressed

$$\hat{G}_0 = \begin{pmatrix} \hat{\delta}_0^x \\ \hat{\delta}_0^y \end{pmatrix}. \tag{3.44}$$

Given these definitions, it is possible to define the symbol of a projection operator. For example, consider the operators $\mathbf{Q}_0 = G_0(D_0G_0)^{-1}D_0$ and $\mathbf{P}_0 = I - \mathbf{Q}_0$. The operator $\mathbf{P}_0$ is the discrete projection operator based on centered difference operators, and it satisfies the relationship $\mathbf{P}_0^2 = \mathbf{P}_0$ Since the boundary conditions are doubly periodic, operators commute and $\mathbf{Q}_0$ can be written

$$\mathbf{Q}_0 = (D_0G_0)^{-1}G_0D_0. \tag{3.45}$$

The symbol of $\mathbf{Q}_0$, $\hat{Q}_0$, is given by

$$\hat{Q}_0(\mathbf{k}) = \frac{1}{(\hat{\delta}_0^x)^2 + (\hat{\delta}_0^y)^2} \begin{pmatrix} \hat{\delta}_0^x\hat{\delta}_0^x & \hat{\delta}_0^x\hat{\delta}_0^y \\ \hat{\delta}_0^x\hat{\delta}_0^y & \hat{\delta}_0^y\hat{\delta}_0^y \end{pmatrix}. \tag{3.46}$$

and satisfies

$$\mathbf{Q}_0(\mathbf{b}\omega_J^{\mathbf{k}}) = \hat{Q}_0(\mathbf{k})\,\mathbf{b}\omega_J^{\mathbf{k}}. \tag{3.47}$$

where $\mathbf{b} = (b_1, b_2)$. Specifically, $\hat{Q}(\mathbf{k})$ is a (2 x 2) matrix which depends on $\mathbf{k}$ with eigenvalues zero and one. By combining equations (3.46) and (3.47), it is not difficult to show that $\|\mathbf{Q}_0\| \le 1$ and that $\|I - \mathbf{Q}_0\| \le 1$.

The same analysis can be applied to $\mathbf{Q} = G_0(\Delta^h)^{-1}D_0$, where $\mathbf{P} = I - \mathbf{Q}$ represents an approximate projection. Since operators commute for the doubly periodic case, $\mathbf{Q}$ can be written

$$\mathbf{Q} = (\Delta^h)^{-1}G_0D_0, \tag{3.48}$$

where $\Delta^h$ is the five point Laplacian defined in equation (3.14). The symbol of $\mathbf{Q}$, $\hat{Q}$, is given by $(\hat{\Delta}^h)^{-1}\hat{G}_0\hat{D}_0$ or

$$\hat{Q}(\mathbf{k}) = \frac{1}{\hat{L}(\mathbf{k})} \begin{pmatrix} \hat{\delta}_0^x\hat{\delta}_0^x & \hat{\delta}_0^x\hat{\delta}_0^y \\ \hat{\delta}_0^x\hat{\delta}_0^y & \hat{\delta}_0^y\hat{\delta}_0^y \end{pmatrix}, \tag{3.49}$$

where

$$\hat{L}(\mathbf{k}) = \frac{e^{ik_1h} + e^{-ik_1h} + e^{ik_2h} + e^{-ik_2h} - 4}{h^2}. \tag{3.50}$$

If $\hat{L}_0(\mathbf{k})$ is defined as $(\hat{\delta}_0^x)^2 + (\hat{\delta}_0^y)^2$, then the relationship between $\hat{Q}$ and $\hat{Q}_0$ is

$$\hat{Q}(\mathbf{k}) = \hat{Q}_0(\mathbf{k}) \frac{\hat{L}_0(\mathbf{k})}{\hat{L}(\mathbf{k})}. \tag{3.51}$$

This relationship can be used to bound $\hat{Q}$, where

$$\|\hat{Q}(\mathbf{k})\| \le \|\hat{Q}_0(\mathbf{k})\| \max_{\mathbf{k}} \left| \frac{\hat{L}_0(\mathbf{k})}{\hat{L}(\mathbf{k})} \right|. \tag{3.52}$$

For $\mathbf{Q}$ to be stable, the eigenvalues of $\hat{Q}(\mathbf{k})$ must be between zero and one. Since the eigenvalues of $\hat{Q}_0(\mathbf{k})$ are zero and one, for $\mathbf{Q}$ to be stable, then $\beta = \hat{L}_0(\mathbf{k}) / (\hat{L}(\mathbf{k}))$ must satisfy

$$0 \le \beta \le 1. \tag{3.53}$$

To evaluate $\beta$, substitute the expressions for $\hat{\delta}_0^x$, $\hat{\delta}_0^y$ and $\hat{L}_0(\mathbf{k})$ in to the definition of $\beta$. The result is:

$$\beta = \frac{\cos^2(k_1h) + \cos^2(k_2h) - 2}{2\cos(k_1h) + 2\cos(k_2h) - 4}, \tag{3.54}$$

which varies between 0 and 1. Thus, equation (3.53) is satisfied and the approximate projection is a stable operator where the eigenvalues of $\hat{Q}(\mathbf{k})$ are zero and $\beta$, and the eigenvalues of $I - \hat{Q}(\mathbf{k})$ are 1 and $1 - \beta$.

Fourier analysis can also be used to show the accuracy of the approximate projection operator. Let $\mathbf{P}^E$ denote the continuous projection operator defined in equations (2.29) - (2.39) and let $\mathbf{P}$ denote an approximate projection operator. If $\mathbf{u}(\mathbf{x})$ is a smooth function and $\mathbf{u}_j^h = \mathbf{u}(jh)$ represents that function evaluated at grid points, then $\mathbf{P}$ is accurate of order p if

$$(\mathbf{P}^E\mathbf{u})(jh) - (\mathbf{P}\mathbf{u}^h)_j = O(h^p). \tag{3.55}$$

If $\hat{v} = \mathbf{b}\mathbf{w}^k(\mathbf{x})$, then for each fixed $\mathbf{k}$,

$$\mathbf{P}^E(\mathbf{b}\mathbf{w}^k)(jh) - (\mathbf{P}\mathbf{b}\omega^k)_j = O(h^2) \tag{3.56}$$

as $h \to 0$ is a necessary and sufficient condition for $\mathbf{P}$ to be a second-order accurate operator.

We will now show that equation (3.56) holds. Let $\hat{\mathbf{u}}$ denote the Fourier transform of $\mathbf{u}$. The spatial derivatives of $\hat{\mathbf{u}}$ are:

$$
\begin{aligned}
\hat{u}_x &= \imath k_1 \hat{u}, \\
\hat{u}_{xx} &= -k_1^2 \hat{u}, \\
\hat{u}_y &= \imath k_2 \hat{u}, \\
\hat{u}_{yy} &= -k_2^2 \hat{u}.
\end{aligned}
\tag{3.57}
$$

Let $\hat{q}$ denote the symbol of $I - \mathbf{P}^E$. Then $\hat{q}$ can be expressed as

$$
\hat{q}(\mathbf{k}) = \frac{1}{-k_1^2 - k_2^2} \begin{pmatrix} -k_1^2 & -k_1 k_2 \\ -k_1 k_2 & -k_2^2 \end{pmatrix}, \tag{3.58}
$$

where $-k_1^2 - k_2^2$ is the symbol of the exact Laplacian.

If the symbol of the discrete five point Laplacian is expanded in a Taylor series, the result is

$$
\hat{L}(\mathbf{k}) = \frac{(2 - k_1^2 h^2) + (2 - k_2^2 h^2) - 4 + O(h^4)}{h^2} = -k_1^2 - k_2^2 + O(h^2). \tag{3.59}
$$

Likewise, the definitions of $\delta_0^x$ and $\delta_0^y$ can also be expanded in a Taylor series to yield

$$
\delta_0^x = \frac{2\imath k_1 h + O(h^3)}{2h} = \imath k_1 + O(h^2) \tag{3.60}
$$

and

$$
\delta_0^y = \frac{2\imath k_2 h + O(h^3)}{2h} = \imath k_2 + O(h^2). \tag{3.61}
$$

Equations (3.59) - (3.61) and equation (3.49) can be combined to show the order of the difference between the continuous and discrete operators:

$$
\hat{q}(\mathbf{k}) - \hat{Q}(\mathbf{k}) = O(h^2), \tag{3.62}
$$

and, thus, that the approximate projection is a second-order accurate approximation to the continuous operator.

## 3.5 Filtering

Some care must be taken in using the approximate projection operator because the centered difference approximations used for D and G allow a non-physical oscillatory mode to persist. This mode, pictured in Figure 3.1, interacts badly with both the discretization of the advective terms $(U \cdot \nabla) U$ and with the volume sources that occur in the combustion case. The mode can be

removed by applying an additional projection operator to the "projected" field which has a different spatial discretization. If the second projection does not use a centered difference divergence,

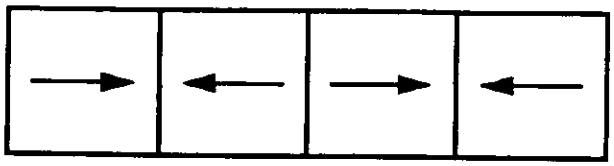


Figure 3.1 Divergent mode left in by the MAC/centered difference projection

this mode is removed.

Although applying a second projection is perfectly feasible, it is expensive computationally. A second solution, the one used in this thesis, is to create a filter which can eliminate the remaining mode. The filter consists of a single point-Jacobi iteration and is constructed using operators designed to catch a two-dimensional version of the mode. This two-dimensional mode is pictured in Figure 3.2.

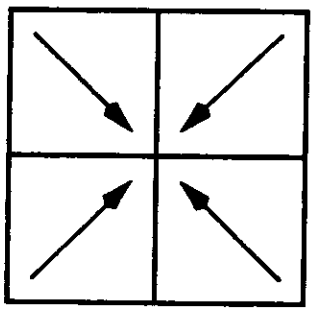


Figure 3.2 Two-dimensional mode removed by the filter

The operators used in the point-Jacobi iteration are denoted by a "d" superscript, which stands for "diagonal." The diagonal divergence, $D^d$, is defined at cell nodes and uses variables at cell centers and is given by

$$(D^dU)_{i-\frac{1}{2},j-\frac{1}{2}} = \frac{(u_{i,j}+u_{i,j-1}) - (u_{i-1,j}+u_{i-1,j-1})}{2\Delta x} + \frac{(v_{i,j}+v_{i-1,j}) - (v_{i,j-1}+v_{i-1,j-1})}{2\Delta y}. \tag{3.63}$$

The corresponding gradient is defined at cell centers using scalars defined at cell nodes and is given by

$$\begin{aligned}(G^d\varphi)^x_{i,j} &= \left[\frac{(\varphi_{i+1/2,j+1/2}+\varphi_{i+1/2,j-1/2}) - (\varphi_{i-1/2,j+1/2}+\varphi_{i-1/2,j-1/2})}{2\Delta x}\right] \\ (G^d\varphi)^y_{i,j} &= \left[\frac{(\varphi_{i+1/2,j+1/2}+\varphi_{i-1/2,j+1/2}) - (\varphi_{i+1/2,j-1/2}+\varphi_{i-1/2,j-1/2})}{2\Delta y}\right].\end{aligned} \tag{3.64}$$

To filter a vector field, V, the following steps are performed:

1. Compute $D^dV$ and initialize $\varphi$ to zero.
2. Relax once on V using

$$\varphi := (\varphi - \lambda D^dG^d\varphi + \lambda D^dV) = \lambda D^dV. \tag{3.65}$$

$\lambda$ is a relaxation parameter chosen to provide maximal damping in the point-Jacobi iteration for the diagonal Laplacian $D^dG^d$. For example, if $\Delta x = \Delta y = h$, then $\lambda = -\frac{(\sqrt{2}h)^2}{8} = -\frac{h^2}{4}$.

3. Replace V by $\mathcal{F}(V)$

$$V := \mathcal{F}(V) = V - G^d\varphi. \tag{3.66}$$

Applying the filter is sufficient to suppress the high wave number mode described above. Effectively, the filter behaves as an additional second-order accurate projection which is based on the divergence-gradient pair $D^d$ and $G^d$. It is expected that second-order accuracy should be maintained when applying the filter, and, in fact, second-order accuracy is observed in the numerical convergence study in Chapter 4.

Note that filtering is not used to maintain the stability of the projection. The mode removed is, technically, a divergence-free mode, and therefore, if left in, would not cause the scheme to become unstable. The mode is removed only because it is non-physical, an artifact of the centered difference divergence operator.

Fourier analysis can be used to show that the filter is accurate for the case of doubly periodic boundary conditions. If vector field $V = \{V_{i,j}\}$ where $i = 1, \ldots, N, j = 1, \ldots, N$ is expanded in

the discrete Fourier modes defined in equation (3.35), the discrete transform of V can be written

$$\hat{V}^k = \sum_j V_j \omega_j^k h^2. \tag{3.67}$$

The stability and accuracy of the filter can be demonstrated. The analysis begins with the observation that

$$I - (D^d G^d)^{-1} G^d D^d \tag{3.68}$$

is a projection operator with the symbol $\hat{I} - (\hat{D}^d \hat{G}^d)^{-1} \hat{G}^d \hat{D}^d$. The eigenvalues of the symbol are zero and one. The filter can be expressed by

$$\mathcal{F}(V) = V - G^d \varphi = (I - \lambda G^d D^d)\, V, \tag{3.69}$$

or

$$\mathcal{F} = \left( I - \left[ \frac{\lambda}{(D^d G^d)^{-1}} \right] (D^d G^d)^{-1} G^d D^d \right). \tag{3.70}$$

The operator in equation (3.70) is very similar to the projection operator in equation (3.68) with an extra factor of $\beta = \lambda / (D^d G^d)^{-1}$. By comparing the two operators, it is easy to see that the eigenvalues of the symbol of the operator in equation (3.70) are zero and $\hat{\beta}$. Thus, if it is possible to show that $0 \le \hat{\beta} \le 1$, then that is equivalent to showing that the filter is a stable operator.

Using the shift operators defined in equation (3.37), $D^d$ and $G^d$ are given by

$$D^d = \begin{pmatrix} \dfrac{(I + S_y^+)(I - S_x^+)}{2h} & \dfrac{(I + S_x^+)(I - S_y^+)}{2h} \end{pmatrix} \tag{3.71}$$

and

$$G^d = \begin{pmatrix} \dfrac{(I + S_y^-)(S_x^- - I)}{2h} \\ \dfrac{(I + S_x^-)(S_y^- - I)}{2h} \end{pmatrix}. \tag{3.72}$$

Let $\hat{D}^d$ and $\hat{G}^d$ be the symbols of $D^d$ and $G^d$, respectively. $\hat{D}^d \hat{G}^d$ can be expressed

$$\hat{D}^d\hat{G}^d = \frac{1}{4h^2}\left(\left(1+e^{ik_2h}\right)\left(1-e^{ik_1h}\right)\left(1+e^{-ik_2h}\right)\left(e^{-ik_1h}-1\right)\right) + \frac{1}{4h^2}\left(\left(1+e^{ik_1h}\right)\left(1-e^{ik_2h}\right)\left(1+e^{-ik_1h}\right)\left(e^{-ik_2h}-1\right)\right), \tag{3.73}$$

which reduces to

$$\hat{D}^d\hat{G}^d = \frac{2\cos(k_1h)\cos(k_2h)-2}{h^2}. \tag{3.74}$$

Therefore,

$$\frac{-4}{h^2} \le \hat{D}^d\hat{G}^d \le 0. \tag{3.75}$$

Since $\lambda$, the relaxation parameter in the filter, is selected to be $\frac{-h^2}{4}$, then $0 \le \hat{\beta} \le 1$ and the filter is a stable operator.

Taylor series extrapolation can be used to show the order of accuracy of the filter. If $u_j^h = u(jh)$ represents a smooth divergence-free field evaluated at grid points, and $\mathcal{F}(u^h) - u^h$ is extrapolated, the result is

$$\mathcal{F}(u^h) - u^h = \lambda G^d D^d u^h = \frac{h^2}{4}\nabla(\nabla\cdot u) + O(h^4) = O(h^4). \tag{3.76}$$

Thus, the filter acting on a divergence-free velocity field adds an $O(h^4)$ correction to that field.

## 3.6 Algorithm for the Non-Reacting Case

A fractional step method based on Bell, et. al. in [6], [7] and [9] is used to solve the Navier-Stokes equations for variable density in a closed container. The equations that describe this flow are given by

$$\rho\frac{Du}{Dt} = -\nabla\pi + \mu\Delta u \qquad u\cdot n|_{\partial\Omega} = 0, \tag{3.77}$$

$$\frac{D\rho}{Dt} = 0, \tag{3.78}$$

$$\nabla\cdot u = 0. \tag{3.79}$$

$U^n = (u^n, v^n)$ represents the discrete velocity field and $\rho^n$ approximates density on the finite difference grid at the beginning of time step n. $\pi^{n-1/2}$ represents the discrete pressure and is centered in time at the half step, at time $t^n - \frac{\Delta t}{2}$. All discrete variables are defined at cell centers

where $q_{i,j} = q(i\Delta x, j\Delta y)$.

The basic structure of the algorithm remains similar to that in Bell, Colella, and Glaz [6], but several additional steps are required to treat the variable density. The approximate projection described above is also used. The algorithm can be outlined as follows:

1. Compute $[(U \cdot \nabla) U]^{n+1/2}$ and $[(U \cdot \nabla) \rho]^{n+1/2}$, estimates to the advective derivatives centered at time $t^n + \frac{\Delta t}{2}$, using a Godunov method.
2. Solve a discrete form of the continuity equation (3.78) to obtain $\rho^{n+1}$.
3. Solve a discrete form of the momentum equation (3.77) to find $U^*$, an intermediate velocity field.
4. Apply the projection to $U^*$ to split $U^*$ into a divergence-free and a gradient component. Use the gradient component to update the pressure gradient, $\nabla \pi^{n+1/2}$, and use the divergence-free component to find $U^{n+1}$.
5. Apply the filter to the velocity, $U^{n+1} \leftarrow \mathcal{F}(U^{n+1})$.

### 3.6.1 Time Stepping Strategy

In this section the details involved in steps 2, 3 and 4 of the outline are discussed. It is assumed that $U^n$, $\rho^n$ and $G\pi^{n-1/2}$, discrete approximations to velocity, density, and pressure gradient that satisfy the boundary conditions and governing equations, are known. The solutions can be advanced in time by solving second-order accurate discretizations of equations (3.77) and (3.78) given by

$$U^* = U^n + \Delta t\left(-[(U \cdot \nabla) U]^{n+1/2} - \frac{1}{\rho^{n+1/2}} G\pi^{n-1/2} + \frac{\mu}{\rho^{n+1/2}} \Delta^h \left(\frac{U^* + U^n}{2}\right)\right) \quad (3.80)$$

and

$$\rho^{n+1} = \rho^n - \Delta t\,[(U \cdot \nabla) \rho]^{n+1/2}. \quad (3.81)$$

$U^*$, the intermediate velocity field obtained by solving equation (3.80), satisfies the same boundary conditions as U, and $\Delta^h$, the viscous operator in equation (3.80), is the discretization of Laplacian in equation (3.14). $\rho^{n+1/2}$ is determined by averaging $\rho^n$ and $\rho^{n+1}$, and the advective flux terms are evaluated to second-order accuracy by a Godunov procedure. (This procedure will be described in the next section.)

Because of the lagged pressure gradient in equation (3.80), $U^*$ is not second-order accurate in time. To achieve second-order accuracy, an updated pressure gradient is required which satisfies the relationship

$$U^{n+1} = U^n + \Delta t\left(-[(U\cdot\nabla)U]^{n+1/2} - \frac{1}{\rho^{n+1/2}}G\pi^{n+1/2} + \frac{\mu}{\rho^{n+1/2}}\Delta^h\left(\frac{U^* + U^n}{2}\right)\right). \quad (3.82)$$

This updated pressure gradient can be found by solving the Poisson equation

$$L_\rho^h \pi^{n+1/2} = D_0\left(\frac{U^* - U^n}{\Delta t} + \frac{1}{\rho^{n+1/2}}G\pi^{n-1/2}\right) \quad (3.83)$$

where $L_\rho^h$ is the difference approximation defined in equation (3.14) with coefficients $\rho = \rho^{n+1/2}$. Given a solution to equation (3.83), it is possible to calculate the velocity update by using

$$U^{n+1} = U^n + \Delta t\left(\frac{U^* - U^n}{\Delta t} + \frac{1}{\rho^{n+1/2}}G\pi^{n-1/2} - \frac{1}{\rho^{n+1/2}}G\pi^{n+1/2}\right). \quad (3.84)$$

Equation (3.83) is a consistent discretization of equation (2.39). In equation (2.39), the solution is required to the equation

$$\nabla\cdot\frac{1}{\rho}\nabla\pi = \nabla\cdot\left(-(\mathbf{u}\cdot\nabla)\mathbf{u} + \frac{\mu}{\rho}\Delta\mathbf{u}\right). \quad (3.85)$$

If the advective and viscous terms in the right hand side of equation (3.85) are replaced by the definition of $U^*$, then this equation discretely reduces to equation (3.83).

Alternately, an incremental pressure formulation can be used to find the updated pressure gradient. This formulation involves solving a Poisson equation for the increment in the pressure, $\delta = \pi^{n+1/2} - \pi^{n-1/2}$. This alternate Poisson equation is given by

$$L_\rho^h \delta = D_0\left(\frac{U^* - U^n}{\Delta t}\right). \quad (3.86)$$

where $L_\rho^h$ is, again, the difference approximation defined in equation (3.14) with coefficients $\rho = \rho^{n+1/2}$. Once equation (3.86) has been solved for $\delta$, it is possible to solve for the velocity update by subtracting the gradient of the pressure increment from $U^*$,

$$U^{n+1} = U^* - \frac{1}{\rho^{n+1/2}} G_0 \delta. \tag{3.87}$$

Equation (3.86) is a consistent discretization of equation (2.39). In equation (2.39), the solution is required to the equation

$$\nabla \cdot \frac{1}{\rho} \nabla \delta = \nabla \cdot \left( - (\mathbf{u} \cdot \nabla) \mathbf{u} + \frac{\mu}{\rho} \Delta \mathbf{u} - \frac{1}{\rho} \nabla \pi^G \right) \tag{3.88}$$

If the advective and viscous terms in the right hand side of equation (3.88) are replaced by the definition of $U^*$, then this equation discretely reduced to equation (3.86).

The incremental pressure projection formulation provides an alternate method of updating the velocity and the pressure. For example, in the constant density inviscid case, updates can be found by either of the following two formulations:

$$\begin{aligned} U^* &= U^n + \Delta t \left( - [ (U \cdot \nabla) U ]^{n+1/2} - G\pi^{n-1/2} \right), \\ U^{n+1} &= U^n + \Delta t \mathbf{P} \left( \frac{U^* - U^n}{\Delta t} \right), \\ G\pi^{n+1/2} &= G\pi^{n-1/2} + (\mathbf{I} - \mathbf{P}) \left( \frac{U^* - U^n}{\Delta t} \right), \end{aligned} \tag{3.89}$$

or

$$\begin{aligned} U^* &= U^n + \Delta t \left( - [ (U \cdot \nabla) U ]^{n+1/2} \right), \\ U^{n+1} &= U^n + \Delta t \mathbf{P} \left( \frac{U^* - U^n}{\Delta t} \right), \\ G\pi^{n+1/2} &= (\mathbf{I} - \mathbf{P}) \left( \frac{U^* - U^n}{\Delta t} \right). \end{aligned} \tag{3.90}$$

If $\mathbf{P}^2 = \mathbf{P}$, these two methods would give identical results. However, for the approximate projection operator where $\mathbf{P}^2 \neq \mathbf{P}$, the two methods differ.

### 3.6.2 The Treatment of the Nonlinear Terms

This section describes the Godunov method used to calculate the time-centered nonlinear flux terms used in equations (3.80) and (3.81). The method is similar to the one used by Bell, et. al. [9] which is conservative and second-order accurate. It can be broken down as follows:

1. Extrapolate along characteristics to construct time-centered left and right states, $(U, \rho)^{n+1/2, L}$ and $(U, \rho)^{n+1/2, R}$.

2. Find the upwind direction at each edge to select $U^{n+1/2}$, $\rho^{n+1/2}$, single values from the left and right state.

3. Difference the left and right states terms to find the advective terms, $[(U \cdot \nabla) U]^{n+1/2}$ and $[(U \cdot \nabla) \rho]^{n+1/2}$. The terms are calculated with centered differences, and are given by

$$\begin{aligned} uU_x + vU_y = & \frac{1}{2}(u^{n+1/2}_{i+1/2,j} + u^{n+1/2}_{i-1/2,j}) \frac{(U^{n+1/2}_{i+1/2,j} - U^{n+1/2}_{i-1/2,j})}{\Delta x} \\ & + \frac{1}{2}(v^{n+1/2}_{i+1/2,j} + v^{n+1/2}_{i-1/2,j}) \frac{(U^{n+1/2}_{i,j+1/2} - U^{n+1/2}_{i,j-1/2})}{\Delta y} \end{aligned} \tag{3.91}$$

and

$$\begin{aligned} u\rho_x + v\rho_y = & \frac{1}{2}(u^{n+1/2}_{i+1/2,j} + u^{n+1/2}_{i-1/2,j}) \frac{(\rho^{n+1/2}_{i+1/2,j} - \rho^{n+1/2}_{i-1/2,j})}{\Delta x} \\ & + \frac{1}{2}(v^{n+1/2}_{i+1/2,j} + v^{n+1/2}_{i-1/2,j}) \frac{(\rho^{n+1/2}_{i,j+1/2} - \rho^{n+1/2}_{i,j-1/2})}{\Delta y}. \end{aligned} \tag{3.92}$$

Various specialized Godunov schemes can be developed by refining any of these steps (for example, as in Bell, Dawson and Shubin [8].)

Since the Godunov scheme is explicit, a Courant-Friedrichs-Lewy (CFL) condition must be satisfied on the time step to insure stability. This constraint is based on the advective velocity only and is given by

$$\begin{aligned} \Delta t \le \sigma \min \left( \frac{\Delta x_{i,j}}{|u^n_{i,j}|}, \frac{\Delta y_{i,j}}{|v^n_{i,j}|} \right), \\ \sigma \le 1. \end{aligned} \tag{3.93}$$

This constraint dictates the size of the time increment for the entire scheme. $\sigma$ is known as the CFL number.

### Extrapolation

The first step of the Godunov method involves extrapolating along characteristics to find values of the density and the velocity on the mesh at time $t^{n+1/2}$. A simple second-order accurate Taylor series is used to perform the extrapolation. Given below are the expressions at the left cell edge, $i+1/2, j$, extrapolated from cell i,j:

$$U^{n+1/2,L}_{i+1/2,j} = U^n_{i,j} + \frac{\Delta x}{2} U^n_{x,ij} + \frac{\Delta t}{2} U^n_{t,ij}, \tag{3.94}$$

$$\rho^{n+1/2,L}_{i+1/2,j} = \rho^n_{i,j} + \frac{\Delta x}{2} \rho^n_{x,ij} + \frac{\Delta t}{2} \rho^n_{t,ij}. \tag{3.95}$$

The x subscript denotes differentiation with respect to the x-direction and the t subscript denotes differentiation with respect to time. The expressions for the density and the velocity at the same edge extrapolated from cell i+1,j are

$$U^{n+1/2,R}_{i+1/2,j} = U^n_{i+1,j} - \frac{\Delta x}{2} U^n_{x,i+1,j} + \frac{\Delta t}{2} U^n_{t,i+1,j}, \tag{3.96}$$

$$\rho^{n+1/2,R}_{i+1/2,j} = \rho^n_{i+1,j} - \frac{\Delta x}{2} \rho^n_{x,i+1,j} + \frac{\Delta t}{2} \rho^n_{t,i+1,j}. \tag{3.97}$$

The temporal derivatives in equations (3.94) - (3.97) can be replaced by substituting in the differential equation. After doing so, these equations can be expressed as

$$\begin{aligned} U^{n+1/2,L}_{i+1/2,j} = U^n_{i,j} &+ \left(\frac{\Delta x}{2} - u_{i,j}\frac{\Delta t}{2}\right) U_{x,ij} \\ &- \frac{\Delta t}{2}\left( (vU_y)_{i,j} - \frac{\mu}{\rho_{i,j}} (\Delta^h U)_{i,j} + \frac{1}{\rho_{i,j}} (G\pi)_{i,j} \right), \end{aligned} \tag{3.98}$$

$$\rho^{n+1/2,L}_{i+1/2,j} = \rho^n_{i,j} + \left(\frac{\Delta x}{2} - u_{i,j}\frac{\Delta t}{2}\right) \rho_{x,ij} - \frac{\Delta t}{2} (v\rho_y)_{i,j}, \tag{3.99}$$

and

$$\begin{aligned} U^{n+1/2,R}_{i+1/2,j} = U^n_{i+1,j} &- \left(\frac{\Delta x}{2} + u_{i+1,j}\frac{\Delta t}{2}\right) U_{x,i+1,j} \\ &- \frac{\Delta t}{2}\left( (vU_y)_{i+1,j} - \frac{\mu}{\rho_{i+1,j}} (\Delta^h U)_{i+1,j} + \frac{1}{\rho_{i+1,j}} (G\pi)_{i+1,j} \right), \end{aligned} \tag{3.100}$$

$$\rho^{n+1/2,R}_{i+1/2,j} = \rho^n_{i+1,j} - \left(\frac{\Delta x}{2} + u_{i+1,j}\frac{\Delta t}{2}\right) \rho_{x,i+1,j} - \frac{\Delta t}{2} (v\rho_y)_{i+1,j}. \tag{3.101}$$

Spatial derivatives are evaluated by centered difference approximations in the normal (x) direction and upwind differences in the transverse (y) direction. In addition, the pressure gradient is omitted in equations (3.98) and (3.100) and corrected for later. Given these substitutions, equations (3.98) - (3.101) can be written

$$U^{n+1/2,L}_{i+1/2,j} = U^n_{i,j} + \left(\frac{1}{2} - \max(u_{i,j}, 0)\frac{\Delta t}{2\Delta x}\right)\delta U_{x,ij} - \frac{\Delta t}{2}\left(v_{i,j}\frac{[U]^y_{i,j}}{\Delta y} - \frac{\mu}{\rho_{i,j}}(\Delta^h U)_{i,j}\right), \quad (3.102)$$

$$\begin{aligned} U^{n+1/2,R}_{i+1/2,j} = U^n_{i+1,j} &+ \left(-\frac{1}{2} - \min(u_{i+1,j}, 0)\frac{\Delta t}{2\Delta x}\right)\delta U_{x,i+1,j} \\ &- \frac{\Delta t}{2}\left(v_{i+1,j}\frac{[U]^y_{i+1,j}}{\Delta y} - \frac{\mu}{\rho_{i+1,j}}(\Delta^h U)_{i+1,j}\right), \end{aligned} \quad (3.103)$$

$$\rho^{n+1/2,L}_{i+1/2,j} = \rho^n_{i,j} + \left(\frac{1}{2} - \max(u_{i,j}, 0)\frac{\Delta t}{2\Delta x}\right)\delta\rho_{x,ij} - \frac{\Delta t}{2}v_{i,j}[\rho]^y_{i,j}, \quad (3.104)$$

$$\rho^{n+1/2,R}_{i+1/2,j} = \rho^n_{i+1,j} + \left(-\frac{1}{2} - \min(u_{i+1,j}, 0)\frac{\Delta t}{2\Delta x}\right)\delta\rho_{x,i+1,j} - \frac{\Delta t}{2}v_{i+1,j}[\rho]^y_{i+1,j}. \quad (3.105)$$

$\delta U_{x,ij}$ and $\delta\rho_{x,ij}$, normal spatial differences in equations (3.102) - (3.105), denote the limited fourth-order accurate finite difference approximations to the derivatives in the normal direction. They are calculated in the x-direction, for example, by

$$\begin{aligned} (\delta^x q)_{i,j} &= \mathrm{sign}(q_{i+1,j} - q_{i-1,j}) \times \\ &\min\left[\left|\frac{2}{3}(q_{i+1,j} - q_{i-1,j}) - \frac{1}{6}((\delta^f q)_{i+1,j} - (\delta^f q)_{i-1,j})\right|, (\delta^{\lim} q)_{i,j}\right] \\ (\delta^f q)_{i,j} &= \min(2|q_{i+1,j} - q_{i-1,j}|, (\delta^{\lim} q)_{i,j}) \times \mathrm{sign}(q_{i+1,j} - q_{i-1,j}) \end{aligned} \quad (3.106)$$

$$(\delta^{\lim} q)_{i,j} = \begin{cases} \min(2|q_{i+1,j} - q_{i,j}|, 2|q_{i,j} - q_{i-1,j}|) & \text{if } (q_{i+1,j} - q_{i,j})(q_{i,j} - q_{i-1,j}) > 0 \\ 0 & \text{otherwise} \end{cases}$$

where $q = (u, v, \rho)$.

$[U]^y_{i,j}$ and $[\rho]^y_{i,j}$, normal spatial differences in equations (3.102) - (3.105), denote the upwind slopes used to obtain the spatial derivatives in the transverse direction. They are calculated by

$$[U]^y_{i,j} = \begin{cases} U_{i,j} - U_{i,j-1} & \text{if } (v_{i,j} > 0) \\ U_{i,j+1} - U_{i,j} & \text{if } (v_{i,j} \le 0) \end{cases} \quad (3.107)$$

for the velocity and

$$[\rho]^y_{i,j} = \begin{cases} \rho_{i,j} - \rho_{i,j-1} & \text{if } (v_{i,j} > 0) \\ \rho_{i,j+1} - \rho_{i,j} & \text{if } (v_{i,j} \le 0) \end{cases} \quad (3.108)$$

for the density. Viscous terms are calculated using the five point Laplacian in equation (3.14).

To evaluate the viscous terms and limited slopes near the boundary, rows of "ghost cells" are

created around the domain and filled with values as indicated by the boundary conditions. More precisely, if within the domain there are cells labelled (i,j) where i = 1,..., nx and j = 1,..., ny, ghost cells (0,j) and (nx+1,j) where j = 0,..., ny+1 and cells (i,0) and (i,ny+1) where i = 0,..., nx+1 are specified. These cells are used, for example, to find the five point Laplacian at cell (1,1), which requires ghost cells (0,1) and (1,0).

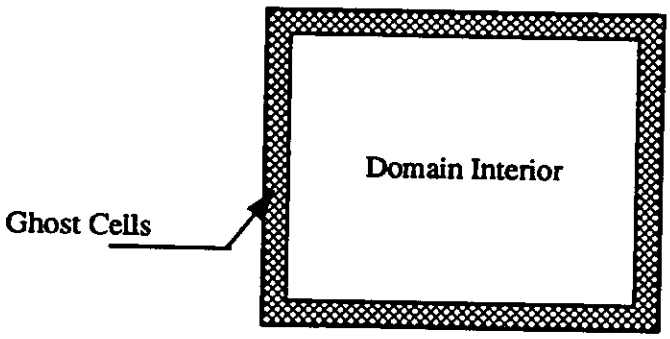



Figure 3.3 Location of ghost cells

For inviscid problems, homogeneous Dirichlet boundary conditions are enforced on the normal component of velocity, and polynomial extrapolation is used to fill ghost cells with the tangential component of velocity. For example, in evaluating $[U]^{y}_{i,j}$ and $\delta^x U_{ij}$, set ghost cells in the first row of exterior cells with

$$\begin{aligned} u_{0,j} &= -u_{1,j}, \\ v_{i,0} &= -v_{i,1}, \end{aligned} \tag{3.109}$$

and

$$\begin{aligned} v_{0,j} &= 3\,(v_{1,j} - v_{2,j}) + v_{3,j}, \\ u_{i,0} &= 3\,(u_{i,1} - u_{i,2}) + u_{i,3}, \end{aligned} \tag{3.110}$$

etc. To compute the normal spatial derivatives for the velocity at cells adjacent to the boundary on the domain interior, $\delta^x U_{ij}$ is set to $\delta^f U_{ij}$, as defined in equation (3.106).

To evaluate viscous terms and compute $\Delta^h$, the no-slip condition is enforced on both the normal and tangential components of velocity. Thus, ghost cells are filled by

$$u_{0,j} = -u_{1,j},$$
$$v_{0,j} = -v_{1,j}, \tag{3.111}$$

etc. Spatial derivatives for density adjacent to the boundary in both the normal and transverse directions, $[\rho]^y_{i,j}$ and $\delta^x\rho_{ij}$, are computed using second-order polynomial extrapolation. Ghost cells are filled, for example, on the left by

$$\rho_{0,j} = 3(\rho_{1,j} - \rho_{2,j}) + \rho_{3,j},$$
$$\rho_{i,0} = 3(\rho_{i,1} - \rho_{i,2}) + \rho_{i,3}. \tag{3.112}$$

## Upwinding

The next step in the Godunov method is to resolve the ambiguity at cell edges by selecting one of the two extrapolated values. Recall that at each cell edge, two values have been extrapolated, as illustrated for edge i+1/2, j in Figure 3.3.

A single value is selected between the left and right states by solving a Riemann problem.

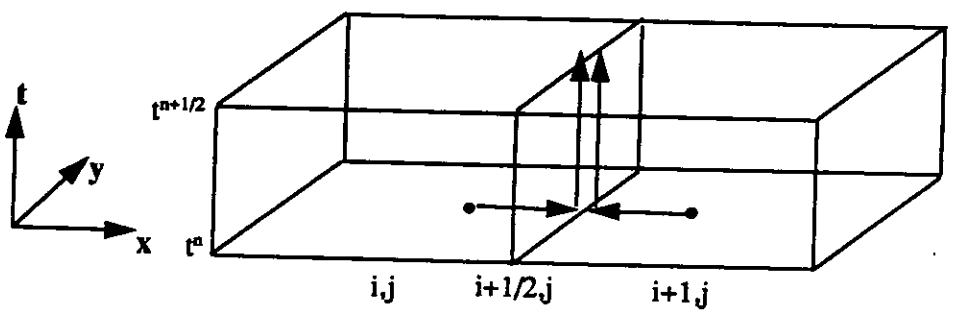



Figure 3.4 Predicted value extrapolation for the Godunov predictor

Mathematically, the solution to a Riemann problem is found by determining which characteristics cross the path at the point of interest. Data are chosen which comes from the upwind side and downwind data are discarded.

The upwind solution is shown here for the normal velocity at the right edge of cell i,j where a single state is selected by

$$u_{i+1/2,j} = \begin{cases} u^L & \text{if } \dfrac{u^n_{i,j} + u^n_{i+1,j}}{2} \geq 0 \\ u^R & \text{otherwise.} \end{cases} \tag{3.113}$$

The normal component of the velocity is then used to determine the tangential component of the velocity and the upwinded density by

$$v_{i+1/2,j} = \begin{cases} v^L & \text{if } u_{i+1/2,j} > 0 \\ v^R & \text{if } u_{i+1/2,j} < 0 \\ \frac{1}{2}(v^L + v^R) & \text{if } u_{i+1/2,j} = 0 \end{cases} \tag{3.114}$$

and

$$\rho_{i+1/2,j} = \begin{cases} \rho^L & \text{if } u_{i+1/2,j} > 0 \\ \rho^R & \text{if } u_{i+1/2,j} < 0 \\ \frac{1}{2}(\rho^L + \rho^R) & \text{if } u_{i+1/2,j} = 0 \end{cases} . \tag{3.115}$$

Boundary conditions are easily enforced on these predicted velocities, $U^{n+1/2}$, since they lie on the mesh. The normal components on the boundary are set using Dirichlet boundary conditions, and the tangential components on the boundary are set using second-order polynomial extrapolation.

After the upwinding is complete, the predicted densities are used to construct the time-centered nonlinear terms, $[(U \cdot \nabla)\rho]^{n+1/2}$, described in equation (3.92).

### 3.6.3 Pressure Correction

Before calculating the advective terms for the velocity, a pressure correction step is performed to estimate for the effect of the omitted pressure term, $\frac{\Delta t}{2\rho}\nabla\pi$. The purpose of the $\nabla\pi$ term is to ensure that the divergence constraint is satisfied to the appropriate order in time. In Bell, Colella and Glaz [6], $\nabla\pi$ was approximated by the lagged pressure gradient, $\nabla\pi^{n-1/2}$, but it was observed by Bell, Colella and Howell [7] that using the lagged pressure gradient in the extrapolation resulted in a non-linear instability for CFL numbers greater than 0.5. The constraint is enforced by performing an additional MAC-type projection designed to return a stable approximation to $\nabla\pi$ for CFL numbers up to one. The corrected pressure gradient satisfies the relationship

$$\frac{\Delta t}{2\rho}\nabla\pi = (\mathbf{I} - \mathbf{P}^{MAC})\, U^{n+1/2}. \tag{3.116}$$

In detail, this MAC-projection is performed by first solving the Poisson equation

$$L_\rho^h \varphi = D^M U \tag{3.117}$$

where $L_\rho^h$ is defined as in equation (3.14) using $\rho = \rho^n$, and $D^M$, the MAC divergence, is given by

$$(D^M U)_{i,j} = \frac{1}{\Delta x}(u_{i+1/2,j} - u_{i-1/2,j}) + \frac{1}{\Delta y}(v_{i,j+1/2} - v_{i,j-1/2}). \tag{3.118}$$

No-normal flow boundary conditions are enforced for $D^M$ by specifying the normal velocity components on the boundary, as in

$$u_{1/2,j} = v_{i,1/2} = u_{nx+1/2,j} = v_{i,ny+1/2} = 0. \tag{3.119}$$

Equation (3.117) is solved with homogeneous Neumann boundary conditions applied in the same manner as discussed in section 3.3.

Given a solution to equation (3.117), the MAC gradient of the scalar field, $G_\rho^M \varphi$, is constructed by

$$(G_\rho^M \varphi)_{i+1/2,j} = \begin{cases} \dfrac{1}{\rho_{i+1/2,j}} \dfrac{(\varphi_{i+1,j} - \varphi_{i,j})}{\Delta x} \\ \dfrac{1}{\rho_{i+1/2,j}} \dfrac{(\varphi_{i+1,j+1} + \varphi_{i,j+1} - \varphi_{i,j-1} - \varphi_{i+1,j-1})}{4\Delta y} \end{cases} \tag{3.120}$$

where $\rho = \rho^n$. The values of the density on the mesh are obtained by averaging as in equation (3.15). $G_\rho^M$ carries homogeneous Neumann boundary conditions which are enforced by setting ghost cell values for $\varphi$ by

$$\begin{aligned} \varphi_{0,j} &= \varphi_{1,j} \\ \varphi_{i,0} &= \varphi_{i,1} \end{aligned} \tag{3.121}$$

etc. Given $G_\rho^M \varphi$, the predicted velocities are corrected by subtracting off this pressure gradient. That is,

$$\begin{aligned} U_{i+1/2,j}^{n+1/2} &:= U_{i+1/2,j}^{n+1/2} - (G_\rho^M \varphi)^x_{i+1/2,j}, \\ U_{i,j+1/2}^{n+1/2} &:= U_{i,j+1/2}^{n+1/2} - (G_\rho^M \varphi)^y_{i,j+1/2}. \end{aligned} \tag{3.122}$$

This formulation is consistent with the overall accuracy of the method since it is only required

to estimate the predicted values of the velocity to $O(\Delta t^2)$. An additional order of accuracy is gained in the corrector step. It is for this same reason that it is sufficient to use explicit viscous terms in the predictor. They too are only $O(\Delta t^2)$, but an additional order of $\Delta t$ is gained in the corrector.

The stability and accuracy of the algorithm for the non-reacting case is difficult to determine for two reasons: the problem is non-linear, and the scheme is non-linear. Stability can be proven, however, for the linearized Euler equations for constant density,

$$\begin{aligned} \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}_0 \cdot \nabla)\,\mathbf{u} &= -\nabla \pi, \\ \nabla \cdot \mathbf{u} &= 0, \end{aligned} \tag{3.123}$$

where $\mathbf{u}_0$ is a constant. For these equations, a scheme can be written

$$U^{n+1} = U^n + \Delta t\,(\mathbf{P} - A(U^n)), \tag{3.124}$$

where

$$A(U^n) = (\mathbf{u}_0 \cdot \nabla)\,\mathbf{u} \tag{3.125}$$

is a linear finite difference approximation to $(\mathbf{u}_0 \cdot \nabla)\,\mathbf{u}$. If $\|A(V)\| \le K\|V\|$ where K is a constant, then

$$\|U^{n+1}\| \le (1 + K\Delta t)\,\|U^n\| \tag{3.126}$$

since $\|\mathbf{P}\| \le 1$.

## 3.7 Algorithm for the Reacting Case

It is possible to extend the approach that was implemented for the non-reacting case and apply it to solving the system of equations for reacting flow discussed in the previous chapter. The most significant design change occurs in enforcing the divergence constraint. For incompressible flow, the velocity is divergence-free and an approximate projection is used to insure that the updated velocity met this homogeneous constraint. In the case of reacting flow, the velocity divergence is non-zero, and so the algorithm must be modified to account for this inhomogeneous constraint.

Once again, the equations for low Mach number combustion in a closed container are conservation of momentum:

$$\frac{Du}{Dt} = -\frac{1}{\rho}\nabla\pi + \frac{\mu}{\rho}\Delta u \qquad u \cdot n|_{\partial\Omega} = 0 \ , \tag{3.127}$$

conservation of energy:

$$\rho c_p \frac{DT}{Dt} = \frac{d\bar{p}}{dt} + \kappa\Delta T + q_0 k\rho Z \qquad \left.\frac{\partial T}{\partial n}\right|_{\partial\Omega} = 0, \tag{3.128}$$

conservation of species:

$$\rho\frac{DZ}{Dt} = \mathcal{D}(\nabla \cdot \rho\nabla Z) - k\rho Z \qquad \left.\frac{\partial Z}{\partial n}\right|_{\partial\Omega} = 0, \tag{3.129}$$

ideal gas equation of state:

$$\rho = \frac{\bar{p}}{RT}, \tag{3.130}$$

mean pressure equation:

$$\frac{d\bar{p}}{dt} = \frac{(\gamma-1)}{vol(\Omega)}\int_{\Omega} (q_0 k\rho Z + \kappa\Delta T)\, d\Omega, \tag{3.131}$$

divergence constraint on the velocity:

$$\nabla \cdot u = S = \frac{1}{\gamma\bar{p}}\left(-\frac{d\bar{p}}{dt} + (\gamma-1)\,(q_0 k\rho Z + \kappa\Delta T)\right). \tag{3.132}$$

The discrete variables are defined as follows: $U^n = (u^n, v^n)$ is the velocity field, $\rho^n$ is the density, $T^n$ is the temperature, $\bar{p}^n$ is the bulk thermodynamic pressure which varies in time only, $Z^n$ is the mass fraction, $k^n$ is the reaction rate, and

$$S^n = \frac{1}{\gamma\bar{p}^n}\left[-\dot{\bar{p}}^n + (\gamma-1)\,(q_0 k\rho Z + \kappa\Delta^h T)^n\right] \tag{3.133}$$

denotes the volume sources. All the above variables are defined on the finite difference grid at the beginning of time step n. $\pi^{n-1/2}$ is the discrete pressure and is centered in time at $t^n - \frac{\Delta t}{2}$. All the variables $q = U, \pi, \rho, T, Z$, and $k$ are defined at cell centers, (i,j), where $q_{i,j} \approx q\,((i-1/2)\,\Delta x,\ (j-1/2)\,\Delta y)$.

The general procedure will remain the same as for the non-reacting case, with similar time-stepping strategy and treatment of the nonlinear terms. Note that the continuity equation is no longer solved explicitly for density. Instead, the temperature and the mean pressure are calculated through the conservation of energy equation (3.128) and the mean pressure evolution equation

(3.131) and then, in turn, the density is updated through the equation of state (3.130).

In outline form, the reacting algorithm consists of the following steps:

A. Velocity Predictor

Perform a Godunov procedure to find the advective terms for the velocity, $[(U \cdot \nabla) U]^{n+1/2}$. The divergence constraint is enforced on the predicted velocities, $U^{n+1/2}$.

B. Scalar Predictor

Perform a Godunov procedure to find the advective terms for the temperature and the mass fraction, $[(U \cdot \nabla) T]^{n+1/2}$ and $[(U \cdot \nabla) Z]^{n+1/2}$. Estimate the temperature, the mean pressure and the density at $t^{n+1/2}$.

C. Scalar Update

1. Solve a second-order accurate Crank-Nicolson discretization for the updated mass fraction, $Z^{n+1}$. The discretization is given by

$$Z^{n+1} = Z^n + \Delta t\left[-[(\mathbf{u} \cdot \nabla) Z]^{n+1/2} + \frac{1}{\rho}\mathcal{D}L^h_{\mathcal{D}}\left(\frac{Z^{n+1}+Z^n}{2}\right) - k\rho\left(\frac{Z^{n+1}+Z^n}{2}\right)\right] \quad (3.134)$$

where $L^h_{\mathcal{D}}$ is a suitable discretization of the species diffusion operator in equation (3.129). $Z^{n+1/2}$ is obtained by averaging $Z^n$ and $Z^{n+1}$.

2. Estimate the temporal derivative of the bulk thermodynamic pressure, $\dot{\bar{p}}^n$, and solve a second-order accurate Crank-Nicolson discretization for the updated temperature, $T^{n+1}$. The discretization is given by

$$\begin{aligned} T^{n+1} &= T^n + \Delta t\,[-[(\mathbf{u} \cdot \nabla) T]^{n+1/2}] \\ &\quad + \frac{\Delta t}{\rho c_p}\kappa\Delta^h\left(\frac{T^{n+1}+T^n}{2}\right) + \dot{\bar{p}}^n + q_0 k\rho Z^{n+1/2}. \end{aligned} \quad (3.135)$$

Average $T^n$ and $T^{n+1}$ to find $T^{n+1/2}$.

3. Correct estimates for $\dot{\bar{p}}^{n+1/2}$ and $\rho^{n+1/2}$. Calculate the volume sources, $S^{n+1}$.

D. Velocity and Pressure Gradient Update

1. Solve a second-order accurate Crank-Nicolson discretization of the momentum equation for the intermediate velocity field, $U^*$. The discretization is given by

$$U^* = U^n + \Delta t\left[-[(u\cdot\nabla)U]^{n+1/2} - \frac{1}{\rho^{n+1/2}}G\pi^{n-1/2} + \frac{\mu}{\rho^{n+1/2}}\Delta^h\left(\frac{U^*+U^n}{2}\right)\right]. \quad (3.136)$$

This step in analogous to advancing the solution in the non-reacting case as described in equation (3.80).

2. Enforce the divergence constraint. Find the incremental pressure gradient, $G_0\delta$, and update $G\pi$. Apply the filter to the divergence-free component of the velocity.

### 3.7.1 Velocity Predictor

For the most part, the velocity predictor is the same as in the non-reacting case. The only difference is in the pressure correction step which insures that the divergence constraint is satisfied to the appropriate order in time. Recall that in the non-reacting case, a MAC-type projection was performed to calculate an estimate for the pressure term that had been omitted in the extrapolation step, $\frac{\Delta t}{2\rho}\nabla\pi$. For the reacting case, the MAC-projection is modified to account for the non-zero divergence of the predicted velocities, $U^{n+1/2}$.

In detail, the pressure correction step for the reacting case is performed by first solving the Poisson equation

$$L_\rho^h\varphi = D^M U^{n+1/2} - S^n \quad (3.137)$$

where $L_\rho^h$ is the discretization of $L_\rho$ as defined in equation (3.14) and $D^M$ is the MAC-divergence defined in equation (3.118). $S^n$, the volume source term defined in equation (3.133), is calculated using the following estimate for $\dot{\bar{p}}^n$:

$$\dot{\bar{p}}^n = \frac{(\gamma-1)}{\mathrm{vol}(\Omega)}\sum_{i,j}(q_0 k\rho Z + \kappa\Delta T)^n. \quad (3.138)$$

Equation (3.137) is solved using homogeneous boundary conditions for $L_\rho^h$ and no-normal flow boundary conditions for $D^M$. They are applied in the same manner as they were for the non-reacting case. Given the solution for equation (3.137), the MAC-gradient of the scalar field, $G_\rho^M\varphi$, is calculated and the predicted velocities are corrected using equation (3.122).

### 3.7.2 Scalar Predictor and Update

In this section, the fractional step method used to find scalar updates is discussed in more

detail. The approach will be similar to the approach in the non-reacting case where the solutions were advanced in time by solving second-order accurate discretizations of the governing equations. The implementation, however, is more complicated in the reacting case due to the presence of source terms associated with combustion.

The goal of the scalar predictor is to estimate nonlinear flux terms and reaction source terms. Nonlinear terms are evaluated using a Godunov procedure. The Godunov method used to evaluate the nonlinear terms for temperature is very similar to that described for density in the non-reacting case. The only difference between the two Godunov methods is that diffusion terms for temperature must be calculated for the predictor. These terms are approximated by a standard five-point discretization to which homogenous Neumann boundary conditions are applied.

In addition, a slightly different version of the Godunov method must be used in calculating the advective terms for mass fraction. It was observed that using the unmodified Godunov procedure introduces a mild non-linear instability for CFL numbers greater than about 0.5 in the reacting flow case. Using the following alternative version of the extrapolation step alleviates the problem.

First calculate auxiliary field, $\tilde{Z}$, by

$$\tilde{Z} = Z + \left(\frac{\Delta t - \tau}{2}\right)\left(\mathcal{D}L^h_{\mathcal{D}}Z - k^n Z\right) \tag{3.139}$$

where $\tau = \min\ (\rho h^2/8\mathcal{D}, \Delta t)$ and $L^h_{\mathcal{D}}Z$ is a discrete approximation to $\nabla \cdot (\rho \nabla Z)$ and is given by

$$\begin{aligned} L^h_{\mathcal{D}}Z = & \frac{\rho_{i+1/2,j}(Z_{i+1,j} - Z_{i,j}) - \rho_{i-1/2,j}(Z_{i,j} - Z_{i-1,j})}{\Delta x^2} \\ & + \frac{\rho_{i,j+1/2}(Z_{i,j+1} - Z_{i,j}) - \rho_{i,j-1/2}(Z_{i,j} - Z_{i,j-1})}{\Delta y^2}. \end{aligned} \tag{3.140}$$

The $\rho$ coefficients in the expression above are obtained by averaging $\rho^n$ as in equation (3.15). Next, a second auxiliary field, $\bar{Z}$, is calculated by

$$\bar{Z} = \tilde{Z} + \frac{\tau}{2}\mathcal{D}L^h_{\mathcal{D}}\tilde{Z} - \frac{\tau}{2}k^n\tilde{Z}. \tag{3.141}$$

This second auxiliary field is used to extrapolate to cell edges at time $t^{n+1/2}$ with a Taylor series

$$Z^{n+1/2,L}_{i+1/2,j} = \bar{Z}^{n}_{i,j} + \left(\frac{1}{2} - u_{i,j}\frac{\Delta t}{2\Delta x}\right)\delta Z_{x,ij} - \frac{\Delta t}{2}[vZ]^{y}_{i,j}. \tag{3.142}$$

Essentially, this modified algorithm smooths the source terms that come from the chemical reaction by performing a single point-Jacobi iteration. The smoothing is consistent with the governing differential equation and introduces no loss of accuracy.

The algorithm continues with the estimation of the time-centered reaction source terms. To find these, we use the temporal derivative of mean pressure, $\dot{\bar{p}}^{n}$, which has been previously calculated in equation (3.138) to get an approximation to $\bar{p}^{n+1/2}$ given by

$$\bar{p}^{n+1/2} = \bar{p}^{n} + \Delta t\dot{\bar{p}}^{n}. \tag{3.143}$$

Next, an approximation to $T^{n+1/2}$, $\tilde{T}$, is estimated by the Taylor series

$$\tilde{T} = T^{n} + \frac{\Delta t}{2}\frac{\partial T}{\partial t}. \tag{3.144}$$

The temporal derivative in equation (3.144) is eliminated by substituting the energy conservation equation

$$\tilde{T} = T^{n} + \frac{\Delta t}{2}\left[-[(U\cdot\nabla)T]^{n+1/2} + \frac{1}{\rho^{n}c_{p}}(\kappa L^{h}T^{n} + \dot{\bar{p}}^{n} + (q_{0}k\rho Z)^{n})\right]. \tag{3.145}$$

$\tilde{T}$ and $\bar{p}^{n+1/2}$ are used to find an estimate for density at $t^{n+1/2}$, $\tilde{\rho}$, through the equation of state. $\tilde{\rho}$ is expressed as

$$\tilde{\rho} = \frac{\bar{p}^{n+1/2}}{R\tilde{T}}. \tag{3.146}$$

$\tilde{T}$ is also used to find an approximation for the time-centered reaction rate, $\tilde{k}$, given by

$$\tilde{k} = Ae^{-E/(R\tilde{T})}. \tag{3.147}$$

The scalar correction step is now performed to find updates for the temperature, the mass fraction, the density and the mean pressure. The previous estimate of the reaction source term and mean pressure derivative is also corrected.

The correction step begins by solving equation (3.134), a second-order accurate Crank-Nicolson discretization of the species conservation equation to find the mass fraction update, $Z^{n+1}$. In the discretization, all the values of $\rho$, including those coefficients of $L^{h}_{\mathcal{D}}$ defined in equation (3.140), are given by averaging $\tilde{\rho}$, the density defined in equation (3.146), as in equation (3.15). k

is given by $\bar{k}$, the reaction rate defined in equation (3.147) Homogeneous Neumann boundary conditions are enforced by setting ghost cell values by extension. for example,

$$\begin{aligned} Z_{0,j} &= Z_{1,j} \\ Z_{i,0} &= Z_{i,1} \end{aligned} \tag{3.148}$$

etc. Next, mass fraction is averaged to find the value at $t^{n+1/2}$,

$$Z^{n+1/2} = \frac{Z^{n+1} + Z^n}{2}. \tag{3.149}$$

The mass fraction update is next used to solve the second-order accurate discretization of the energy conservation equation for $T^{n+1}$, equation (3.135). In this discretization, $\rho = \tilde{\rho}$, the density defined in equation (3.146) and $k = \tilde{k}$, the reaction rate defined in equation (3.147). $\Delta^h$ is the five-point discretization of Laplacian defined in equation (3.14). Homogeneous Neumann boundary conditions are enforced as they are for Z, by setting ghost cell values by extension, as in

$$\begin{aligned} T_{0,j} &= T_{1,j} \\ T_{i,0} &= T_{i,1} \end{aligned} \tag{3.150}$$

etc. Again, temperature at the old and new times is averaged to determine $T^{n+1/2}$,

$$T^{n+1/2} = \frac{T^{n+1} + T^n}{2}. \tag{3.151}$$

The corrector step continues by recalculating reaction source terms and the temporal derivative of the mean pressure. First, reaction rates are determined at time $t^{n+1/2}$ with the expression

$$k^{n+1/2} = Ae^{-E/(RT^{n+1/2})}. \tag{3.152}$$

The reaction rates are used to correct the mean pressure derivative and update mean pressure. These quantities are calculated by

$$\begin{aligned} \dot{\bar{p}}^{n+1/2} &= \frac{(\gamma-1)}{\mathrm{vol}(\Omega)} \sum_{i,j} (q_0 k^{n+1/2} \tilde{\rho} Z^{n+1/2} + \kappa \Delta^h T^{n+1/2}), \\ \bar{p}^{n+1} &= \bar{p}^n + \Delta t \dot{\bar{p}}^{n+1/2}, \\ \bar{p}^{n+1/2} &= \frac{\bar{p}^n + \bar{p}^{n+1}}{2}. \end{aligned} \tag{3.153}$$

Since $\dot{\bar{p}}$ was not centered properly in time in when the discretization of the energy equation (3.135) was solved, the temperature is corrected using the new estimate of $\dot{\bar{p}}^{n+1/2}$ by

$$T^{n+1} := (T^{n+1} + \Delta t\,(\dot{\bar{p}}^{n+1/2} - \dot{\bar{p}}^{n})). \tag{3.154}$$

Given the corrected temperature, $T^{n+1/2}$ is obtained by averaging $T^{n+1}$ and $T^{n}$ as in equation (3.151).

The density can now be found at any point in time by applying the equation of state using the appropriately centered values of the pressure and the temperature. That is,

$$\begin{aligned} \rho^{n+1/2} &= (\bar{p}^{n+1/2}/(RT^{n+1/2})), \\ \rho^{n+1} &= (\bar{p}^{n+1}/(RT^{n+1})). \end{aligned} \tag{3.155}$$

Volume source terms can be evaluated at $t^{n+1}$ by the expression

$$S^{n+1} = \frac{1}{\gamma\bar{p}^{n+1}}\,[-\dot{\bar{p}}^{n+1/2} + (\gamma-1)\,(q_0 k\rho Z + \kappa\Delta^h T)^{n+1}]. \tag{3.156}$$

### 3.7.3 Velocity and Pressure Update

The corrected scalars are used in the predictor-corrector method used to update the velocity and the pressure gradient. After the Godunov procedure is performed and the advective terms are found, $U^*$ is computed, where $U^*$ is an intermediate velocity field which does not satisfy the divergence constraint accurately enough in time. $U^*$ is obtained by solving a second- order accurate Crank-Nicolson discretization for the momentum equation, equation (3.136), where both the advective terms and the pressure gradient are treated as sources. Note that the lagged pressure gradient is used in this expression. This lagged term is the reason that $U^*$ only satisfies the divergence constraint to first-order accuracy in time. To achieve second-order accuracy, an expression for $U^{n+1}$ is required with the pressure gradient properly centered in time, as in the expression

$$U^{n+1} = U^{n} + \Delta t\left[-[(\mathbf{u}\cdot\nabla)U]^{n+1/2} - \frac{1}{\rho^{n+1/2}}G\pi^{n+1/2} + \frac{\mu}{\rho^{n+1/2}}\Delta^h\left(\frac{U^* + U^n}{2}\right)\right]. \tag{3.157}$$

This centering for the pressure gradient can be obtained by solving a pressure Poisson equation for the updated pressure,

$$L_\rho^h \pi^{n+1/2} = D_0\left(\frac{U^* - U^n}{\Delta t} + \frac{1}{\rho^{n+1/2}}G\pi^{n-1/2}\right) - \left(\frac{S^{n+1} - S^n}{\Delta t}\right). \tag{3.158}$$

Given a solution to equation (3.158), it is possible to calculate the velocity update by evaluating the expression

$$U^{n+1} = U^n + \Delta t\left(\frac{U^* - U^n}{\Delta t} + \frac{1}{\rho^{n+1/2}} G\pi^{n-1/2} - \frac{1}{\rho^{n+1/2}} G\pi^{n+1/2}\right). \tag{3.159}$$

Equation (3.158) is a consistent discretization of equation (2.86). Equation (2.86) requires the solution of the following Poisson equation:

$$\nabla \cdot \frac{1}{\rho}\nabla\pi = \nabla \cdot \left(\frac{\partial}{\partial t}(\nabla\psi) - (\mathbf{u}\cdot\nabla)\,\mathbf{u} + \frac{\mu}{\rho}\Delta\mathbf{u}\right). \tag{3.160}$$

If the advective and viscous terms in the right hand side of equation (3.160) are replaced by the definition of $U^*$ and $\nabla \cdot \left(\frac{\partial}{\partial t}(\nabla\psi)\right)$ is replaced by $\frac{\partial S}{\partial t}$, then discretely this equation reduces to equation (3.158).

Alternately, a pressure Poisson equation can be solved for the increment in the pressure, $\delta = \pi^{n+1/2} - \pi^{n-1/2}$. This alternate Poisson equation is given by

$$L_\rho^h \delta = D_0\left(\frac{U^* - U^n}{\Delta t}\right) - \left(\frac{S^{n+1} - S^n}{\Delta t}\right) \quad . \tag{3.161}$$

Given a solution to equation (3.161), it is possible to solve for the velocity update by subtracting the gradient of the pressure increment from $U^*$,

$$U^{n+1} = U^* - \frac{1}{\rho^{n+1/2}} G_0 \delta. \tag{3.162}$$

Equation (3.161) is a consistent discretization of equation (2.86). Equation (2.86) requires the solution to the Poisson equation

$$\nabla \cdot \frac{1}{\rho}\nabla\delta = \nabla \cdot \left(\frac{\partial}{\partial t}(\nabla\psi) - (\mathbf{u}\cdot\nabla)\,\mathbf{u} + \frac{\mu}{\rho}\Delta\mathbf{u} - \frac{1}{\rho}\nabla\pi^G\right). \tag{3.163}$$

If the advective and viscous terms in the right hand side of equation (3.163) are replaced by the definition of $U^*$ and the guess for the pressure, $\pi^G$, is taken to be the pressure at the old time step, discretely this equation reduces to equation (3.161).

In equations (3.161) and (3.163), $L_\rho^h$ is the difference approximation defined in equation (3.14) with coefficients $\rho = \rho^{n+1/2}$ given by equation (3.155). Boundary conditions for each operator are treated as in the non-reacting case.

Given $S^{n+1}$, the volume source terms at time $t^{n+1}$, the potential for the gradient component of the velocity, $G_0\psi^{n+1}$, can be computed by solving

$$\Delta^h \psi^{n+1} = S^{n+1}. \tag{3.164}$$

The divergence-free component of the velocity is then obtained by subtracting the gradient component from $U^{n+1}$; i.e., $U^{n+1} - G_0\psi^{n+1}$ is approximately divergence-free.

Finally, the divergence-free component of $U^{n+1}$ is filtered and added to the gradient component to determine the velocity update. That is,

$$U^{n+1} := (\mathcal{F}(U^{n+1} - G_0\psi^{n+1}) + G_0\psi^{n+1}) \tag{3.165}$$

where $\mathcal{F}$ denotes the filter given by equations (3.65) - (3.66) and $U^{n+1}$ satisfied equation (3.157).

## 3.8 Computational Effort

Finally, we address the question of computational effort and show which portions of the algorithm are likely to be the most expensive. The velocity predictor is explicit and, therefore, inexpensive. The pressure correction step requires the solution of Poisson equation (3.137). This linear system is solved using a multigrid method. The scalar predictor is entirely explicit. The scalar update section requires the solution of two second-order accurate parabolic equations, (3.134) and (3.135). Again, these linear systems are solved using a multigrid method. The velocity and pressure update is the most expensive section of the code. It involves solving the two parabolic problems in equation (3.136), one to find each component of the intermediate velocity field, $U^*$. In addition, the Poisson equation (3.161) must be solved to find the incremental pressure gradient, and the Poisson equation (3.164) must be solved to find the velocity potential due to combustive heat release. Still, these systems are all linear, and again a multigrid method is used to expedite the linear algebra.

All together, four parabolic equations and three elliptic equations are solved per time step. The details of what percentage of CPU time gets devoted to solving each of the major elements are discussed in Chapter 4.

# Chapter 4
# Results

## 4.1 Rayleigh Taylor Instability

In this chapter, the results of our calculations are presented and numerical convergence is demonstrated for the method discussed in Chapter 3. The first calculation shows the development of a Rayleigh-Taylor instability which occurs when a heavy fluid accelerates into a lighter fluid due to the force of gravity. Initially, the density interface between the two fluids is perturbed by

$$\eta(x) = 0.05L\cos\left(\frac{2\pi x}{L}\right) \tag{4.1}$$

where L is the width of the computational domain. The interface has been smoothed by a tanh function. The initial density ratio is 7:1, and the fluids are assumed to be inviscid. A uniform 32 x 128 grid was used to perform this calculation.

At early times, we see the interface begin to roll-up into two counter-rotating vortices. As time progresses, the vortical structure becomes quite convoluted. The significance of these results is their favorable comparison to those in Bell and Marcus [9]. This, along with the results of the numerical convergence study, is an indication that the formulation using the approximate projection indeed captures the physics of the problem.

## 4.2 Numerical Convergence

Convergence is established by solving the same problem with smooth initial data on coarse and fine grids, outputting the results at a fixed time, and comparing the difference on adjacent grids. The initial stream function is given by

$$\Psi^0 = \pi^{-1}\sin^2(\pi y)\sin^2(\pi x) \tag{4.2}$$

while the initial temperature and mass fraction are expressed by

$$T^0 = \frac{1}{1-\tanh(y-0.5)} \tag{4.3}$$

and

$$Z^0 = \frac{1}{1-\tanh(y-0.5)} - 0.5. \tag{4.4}$$

The results of the convergence study for the non-reacting cases are summarized in tables 1, 2, and 3. We tested the following cases:

Case I: $Pr = 0$, $Sc = 0$, $Le = 0$, Euler

Case II: $Pr = 0$, $Sc = \infty$ (i.e., no species diffusivity), $Le = 0$, $Re = 100$

Case III: $Pr = 1$, $Sc = 1$, $Le = 0$, $Re = 100$

Where Pr, Sc, Le and Re are the Prandtl, Schmidt, Lewis and Reynolds numbers, respectively, defined $Pr = \frac{c_p \mu}{\kappa}$, $Sc = \frac{\nu}{\mathcal{D}}$, $Le = \frac{\kappa}{\rho c_p \mathcal{D}}$, and $Re = \frac{\rho U L}{\mu}$.

**Table 4.1 Velocity Convergence Results - Non-reacting Flow**

| Norm | Case | 32-64 | Rate | 64-128 | Rate | 128-256 |
|---|---|---|---|---|---|---|
| L1 | I | 3.801e-3 | 2.12 | 8.516e-4 | 2.07 | 2.005e-4 |
| | II | 2.154e-3 | 1.97 | 5.547e-4 | 1.97 | 1.424e-4 |
| | III | 2.153e-3 | 1.97 | 5.545e-4 | 1.97 | 1.423e-4 |
| L2 | I | 3.479e-3 | 2.10 | 7.898e-4 | 2.06 | 1.863e-4 |
| | II | 1.898e-3 | 2.00 | 4.741e-4 | 1.98 | 1.208e-4 |
| | III | 1.897e-3 | 2.00 | 4.740e-4 | 1.98 | 1.209e-4 |

**Table 4.2 Temperature Convergence Results - Non-reacting Flow**

| Norm | Case | 32-64 | Rate | 64-128 | Rate | 128-256 |
|---|---|---|---|---|---|---|
| L1 | I | 1.046e-4 | 2.15 | 2.822e-5 | 2.12 | 5.109e-6 |
| | II | 9.548e-5 | 2.03 | 2.324e-5 | 2.03 | 5.667e-6 |
| | III | 4.494e-5 | 2.06 | 1.057e-5 | 2.03 | 2.564e-6 |
| L2 | I | 1.496e-4 | 2.15 | 3.234e-5 | 2.12 | 7.251e-6 |
| | II | 1.239e-4 | 2.01 | 3.059e-5 | 2.00 | 7.605e-6 |
| | III | 5.838e-5 | 2.07 | 1.369e-5 | 2.03 | 3.329e-6 |

**Table 4.3 Mass Fraction Convergence Results - Non-reacting Flow**

| Norm | Case | 32-64 | Rate | 64-128 | Rate | 128-256 |
|---|---|---|---|---|---|---|
| L1 | I | 1.046e-4 | 2.14 | 2.282e-5 | 2.12 | 5.109e-6 |
| | II | 9.547e-5 | 2.03 | 2.34e-5 | 2.03 | 5.668e-6 |
| | III | 4.493e-5 | 2.06 | 1.057e-5 | 2.03 | 2.565e-6 |
| L2 | I | 1.496e-4 | 2.16 | 3.234e-5 | 2.12 | 7.251e-6 |
| | II | 1.239e-4 | 2.01 | 3.059e-5 | 2.00 | 7.604e-6 |
| | III | 5.840e-5 | 2.07 | 1.369e-5 | 2.03 | 3.332e-6 |

A combusting case was used as well in the convergence study. For this case, the initial stream function is the same as in equation (4.2). The initial temperature and mass fraction profiles are the same as for the cases pictured in Figure 4.6 through Figure 4.8 which consist of two "hot spots", one at (0.25, 0.25) and another at (0.75, 0.75) defined by

$$T^0 = \frac{1}{1000} + \frac{\tanh(((r-0.125)\,32)+1)}{2} + \left(\frac{1}{300} - \frac{1}{1000}\right) \tag{4.5}$$

and

$$Z^0 = \frac{\tanh(((r-0.125)\,32)+1)}{2}. \tag{4.6}$$

The results of this test case for which Sc = 1, Pr = 1, and Le = 1, and Re = 100 is presented in Table 4.4 - Table 4.6

**Table 4.4 Velocity Convergence Results - Reacting Case**

| Norm | 32-64 | Rate | 64-128 | Rate | 128-256 |
|---|---|---|---|---|---|
| L1 | 9.937e-2 | 2.08 | 2.310e-3 | 2.05 | 5.514e-4 |
| L2 | 1.041e-2 | 2.09 | 2.390e-3 | 2.05 | 5.708e-4 |

**Table 4.5 Temperature Convergence Results - Reacting Case**

| Norm | 32-64 | Rate | 64-128 | Rate | 128-256 |
|---|---|---|---|---|---|
| L1 | 2.650e-3 | 2.00 | 6.596e-4 | 2.03 | 1.595e-4 |
| L2 | 5.641e-3 | 2.10 | 1.279e-3 | 2.07 | 2.976e-4 |

Table 4.6 Mass Fraction Convergence Results - Reacting Case

| Norm | 32-64 | Rate | 64-128 | Rate | 128-256 |
|---|---|---|---|---|---|
| L1 | 1.802e-3 | 2.05 | 4.318e-4 | 1.88 | 1.202e-4 |
| L2 | 4.093e-e | 2.03 | 9.991e-4 | 1.87 | 2.809e-4 |

## 4.3 Combustion in a Closed Container

The next calculation shown is that of combustion in a closed container. Initially, a smooth initial velocity profile is defined inside a unit square with homogeneous Dirichlet boundary conditions. The initial temperature profile consists of two smooth "hot spots" in the lower left and upper right corners, corresponding to areas where the fluid contains only products of combustion. The remainder of the fluid consists of low temperature reactants. The Reynolds number is approximately 5000, and the Prandtl, Lewis and Schmidt numbers are all of order unity. A 128 x 128 uniform grid was used to perform this calculation.

We note especially the complex vortical structure that forms in the burnt gas due to the baroclinic generation of vorticity at the flame front and the enhanced mixing of the lower density gases. This vorticity is generated due to the application of the same force is applied both in front and in back of the flame. In the rear of the flame front, the temperature of the gas is higher and, therefore, the density is lower. The lower density gas is accelerated more rapidly than the high density gas, creating a velocity gradient. The strongest baroclinic vorticity generation is thus displayed at early times since the effect is strongest when the density ratio is highest. As the flame front moves outwards, this vorticity is left behind and serves to enhance the mixing in the center of the container.

In addition, the low Mach number effects are apparent in the temperature profiles. Note that the high temperature zone is located not at the flame front where the energy is released, but rather in the center of the domain. This effect is due to the uniform rise in bulk thermodynamic pressure; as the pressure rises, the temperature rises highest in the center where combustion is completed and the density is lowest. This behavior of the temperature field can be predicted by examining the conservation of energy equation which shows that $dT \sim \frac{1}{\rho} d\bar{p}$. The factor of $\rho$ on the right hand side has the effect of increasing temperatures in the post-combustion, low-density regions.

In this calculation it was assumed that the reactants are perfectly mixed before chemical reac-

tion and the flame is laminar. According to laminar flame theory, the flame zone is divided into two sections: zone 1, the preheat zone, where the gases are heated by conduction and diffusion until they reach ignition temperature; and zone 2, the reaction zone (see [45] for more information.) A diagram of a one-dimensional laminar flame is shown in Figure 4.1.

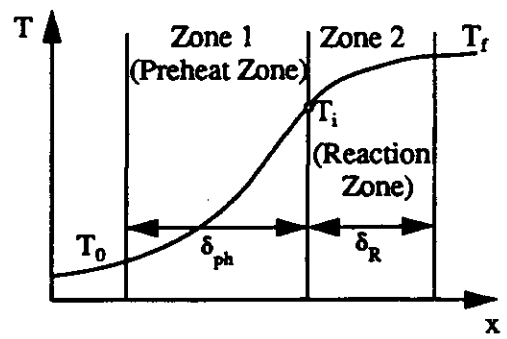



Figure 4.1 Diagram of temperature variations and reaction zones for a laminar flame

A scaling argument can be used to show that the flame front is being resolved in space and time if the flame is described by this theory. An energy balance in zone 1 shows that

$$\dot{m}c_p(T_i - T_0) = \kappa\frac{(T_f - T_0)}{\delta_R} \tag{4.7}$$

where $T_0$ is the temperature in the unburned region, $T_i$ is the temperature at the border between zone 1 and zone 2, $T_f$ is the temperature of the combustion products, and $\delta_R$ is the length of the reaction zone. The mass flow rate, $\dot{m}$, is defined as

$$\dot{m} = \rho S_L \tag{4.8}$$

where $\rho$ is the density of the unburned gas and $S_L$ is the laminar flame speed. By combining equations (4.7) and (4.8), a definition for the flame speed can be found given by

$$S_L = \frac{\kappa}{\rho c_p}\frac{(T_f - T_0)}{(T_i - T_0)}\frac{1}{\delta_R}. \tag{4.9}$$

If $\tau_R$, the reaction time, is the reciprocal of the reaction rate, k, then

$$\delta_R \propto \frac{S_L}{k} \tag{4.10}$$

and

$$\delta_R \propto \sqrt{\frac{\kappa}{\rho c_p} \frac{(T_f - T_0)}{(T_i - T_0)} \frac{1}{k}} \quad . \tag{4.11}$$

An ignition temperature reaction rate model is used where $k \sim 1\ s^{-1}$. It is assumed that $T_f \approx 1000$ K, $T_0 \approx 300$ K, $T_i \approx 600$ K, $\kappa = 1.4 \times 10^{-5}$ cal/(s cm K), $\rho \approx 3 \times 10^{-6}$ g/cm$^3$ and $c_p \approx 1.0$ cal/(s g K). By substituting these parameters into equation (4.11) we can see that our flame front should be spread over approximately four cells.

Another way convergence is demonstrated is by showing that the rise in bulk thermodynamic pressure and the rate of fuel consumption both approach a limit. Thermodynamic pressure vs. time and the $L_2$ norm of the mass fraction field vs. X are plotted in Figure 4.2 and Figure 4.3 for three uniform meshes: 64 x 64, 128 x 128, and 256 x 256

## 4.4 CPU Time and Memory Requirements

Performance tests were run on a Cray Y-MP C90 to asses the algorithm's efficiency. The results are summarized in Table 4.7 for the case of combustion in a closed container on a uniform grid. Two cases were tested, one on a 128 x 128 uniform mesh and another on a 256 x 256 uniform mesh. The first two columns in the table show the percentage of CPU time spent solving elliptic problems and the percentage spent solving parabolic problems. Column 3 shows the average number of Mflops (millions of floating point operations per second), an indication of whether the code is properly vectorized. Column 4 shows the maximum amount of memory the program required in units of Mw (mega words), and column 5 indicates the rate of throughput by showing, on average, how many CPU seconds are required to complete one time step.

**Table 4.7 CPU and Memory Usage**

| N | Elliptic% | Parabolic% | Mflops | Memory (Mw) | CPU secs/ time step |
|---|---|---|---|---|---|
| 128 | 62.2 | 8.4 | 381.4 | 1.62 | .358 |
| 256 | 62.6 | 9.1 | 484.9 | 5.85 | 1.1 |

The table clearly shows that the vast majority of CPU cycles are spent in Multigrid solving elliptic problems. However, the table also indicates that the program is dominated by vector operations and that good execution speeds are being obtained. For the 128 x 128 case, of the 381.4

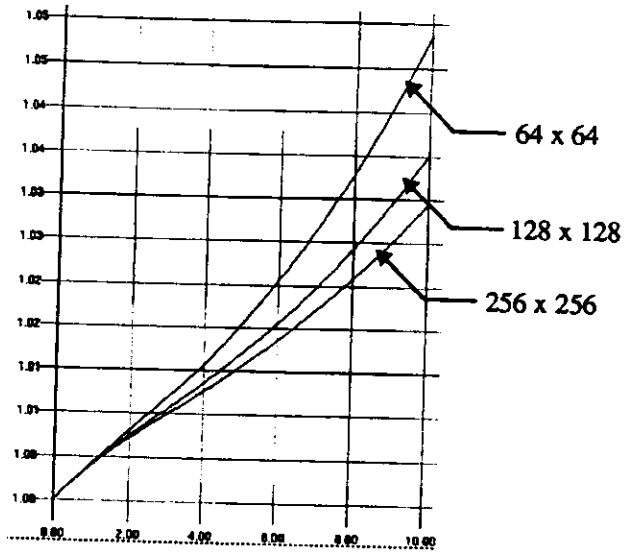



Figure 4.2 Bulk thermodynamic pressure vs. time for three uniform grids

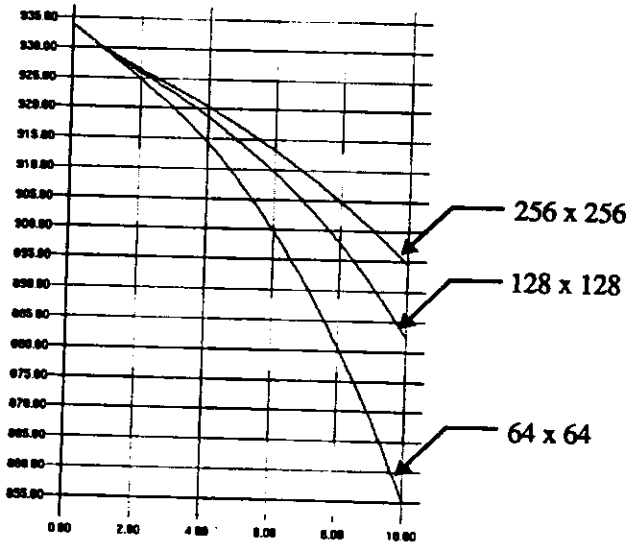



Figure 4.3 The $L_2$ norm of the mass fraction vs. time for three uniform grids

Mflops overall, 380.1 were performed in the vector units. For the 256 x 256 case, 484.0 of the 484.9 Mflops were performed in the vector units. To lend some perspective to these figures, a rate of over 200 Mflops is considered to be adequate, and a rate of over 400 Mflops is considered to be very good. The memory requirements at first glance may seem high. However, the figure is inflated due to the fact all memory required by multigrid is allocated once at the beginning of the program. If lower memory overhead is desired, memory could easily be allocated and deallocated in the

multigrid routines themselves.

## 4.5 Future Work

This algorithm was developed as part of a long-term plan to create an engineering tool which could be used to design components that involve low Mach number reacting and non-reacting fluid flow. Of course, major modifications must be made to achieve this goal; in particular, many of the simplifying assumptions made previously must be relaxed, changes must be made to improve the physical model, and new additions must be put into place to handle more complex spatial domains. Some specific suggestions for future modifications are:

1. Improved Gridding Methods

Adaptive Mesh Refinement (AMR) as presented in Berger and Oliger [11] and Berger and Colella [12] is one way that gridding could be improved. AMR is a means of locally refining a uniform mesh so that the computational power is only put into regions which high gradients. To implement AMR, a refinement criteria is determined, the domain is gridded accordingly, and then the algorithm is applied to both coarse and fine grids. Regridding is performed every few steps. By following such an approach, the CPU time required to perform these computations can be reduced significantly, as shown by Almgren, et. al. [2] who used AMR in performing a calculation to solve the incompressible Euler equations.

In addition to adding AMR, an improved boundary treatment should be used. Two approaches for handling complicated domain geometries are the Cartesian mesh method as presented by Chern and Colella [16] and Young, et. al. [68], and the composite mesh method described in Chesshire and Henshaw [17] and Benek, et. al. [10]. To implement a Cartesian mesh strategy, a uniform grid is placed over the entire domain, and bodies and domain boundaries are allowed to intersect cells. Intersected cells are treated as though they contain a tracked front. The advantage to the Cartesian mesh method is it's simplicity; it is easily implemented for any domain or for flow over any body. The disadvantage to this method is that some accuracy is lost at the boundary. Whether a Cartesian mesh approach is appropriate or not is problem-dependent; if the flow is primarily driven by boundary layer interactions, the loss of accuracy at the boundary makes this method a poor choice. However, for flows where boundary flow may not be of primary importance, a Cartesian mesh

approach is a viable option. A Cartesian mesh approach has already been used by Pember, et. al. [52] for inviscid compressible flow, and methods for automatic grid generation are being currently investigated by Melton, et. al. [49].

A composite mesh approach is implemented by creating multiple meshes and interpolating between them. For example, a uniform mesh may be laid over the majority of the domain, except over bodies in the flow and at boundaries where a body-fitted grid would be used. The body-fitted grid would cover the boundary layer and extend partially into the free stream. Boundary data would be interpolated onto the body-fitted grid from the uniform background mesh. The advantage of such an approach is its preservation of accuracy at the boundary. The disadvantage is that such an algorithm is correspondingly more complicated.

2. Improved Physical Models

One way to improve the physical model would be to replace the ignition temperature kinetics model by a more complex model. This addition could be useful for predicting pollutant generation in burners or internal combustion engines where mechanisms must be included in the kinetics model to account for the formation of intermediate species. Another way to improve the physical model would be to add a radiation model. A third improvement would be to relax the assumption that fluid properties such as viscosity, diffusivity, thermal conductivity, etc., are constant. To perform a more realistic computation, clearly these properties must be allowed to vary with the thermodynamic variables. Finally, other reduced models for combustion can be implemented, such as algorithms for tracking thin flame fronts, and the conserved scalar model for calculating non-premixed combustion flames in Bilger [13]. Thin flame models have been used in the past by Ghoniem, Chorin and Oppenheim [30], Zhu and Sethian [69], and Hilditch [40].

Figure 4.4 Early evolution of the Rayleigh-Taylor instability showing the formation of two counter-rotating vortices into a mushroom shape

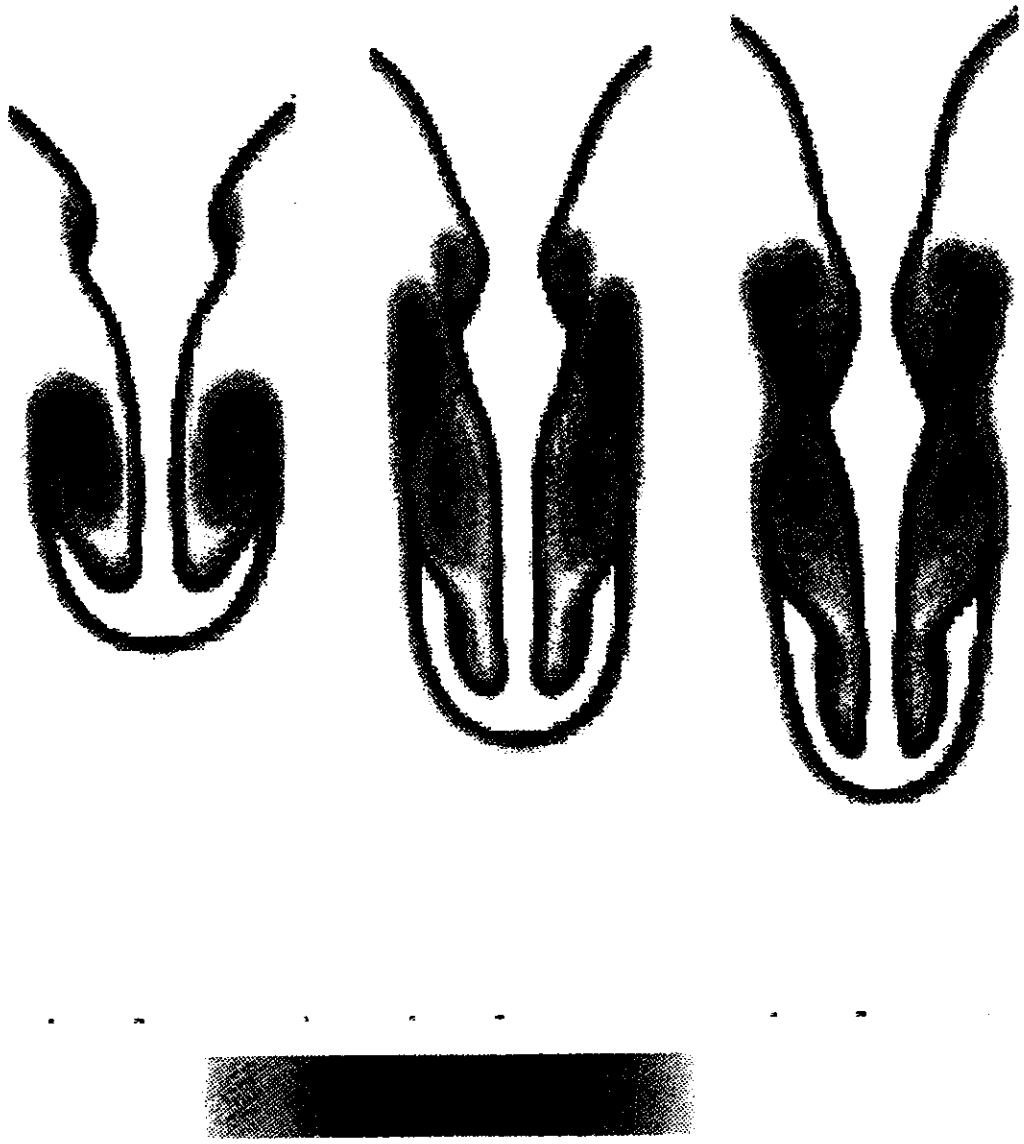


Figure 4.5 Late time evolution of the Rayleigh-Taylor instability

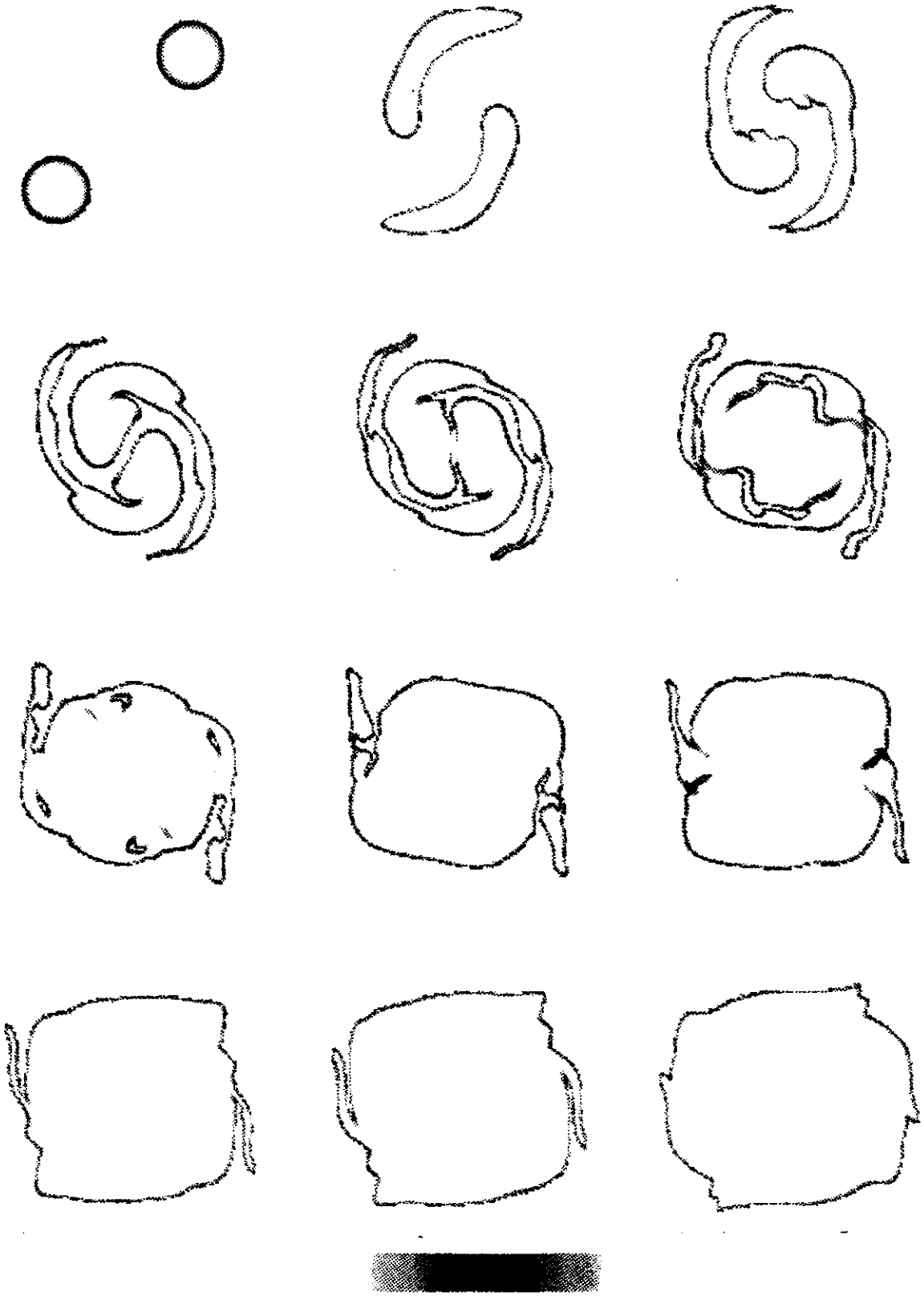


Figure 4.6 Flame front location for the case of combustion in a closed container

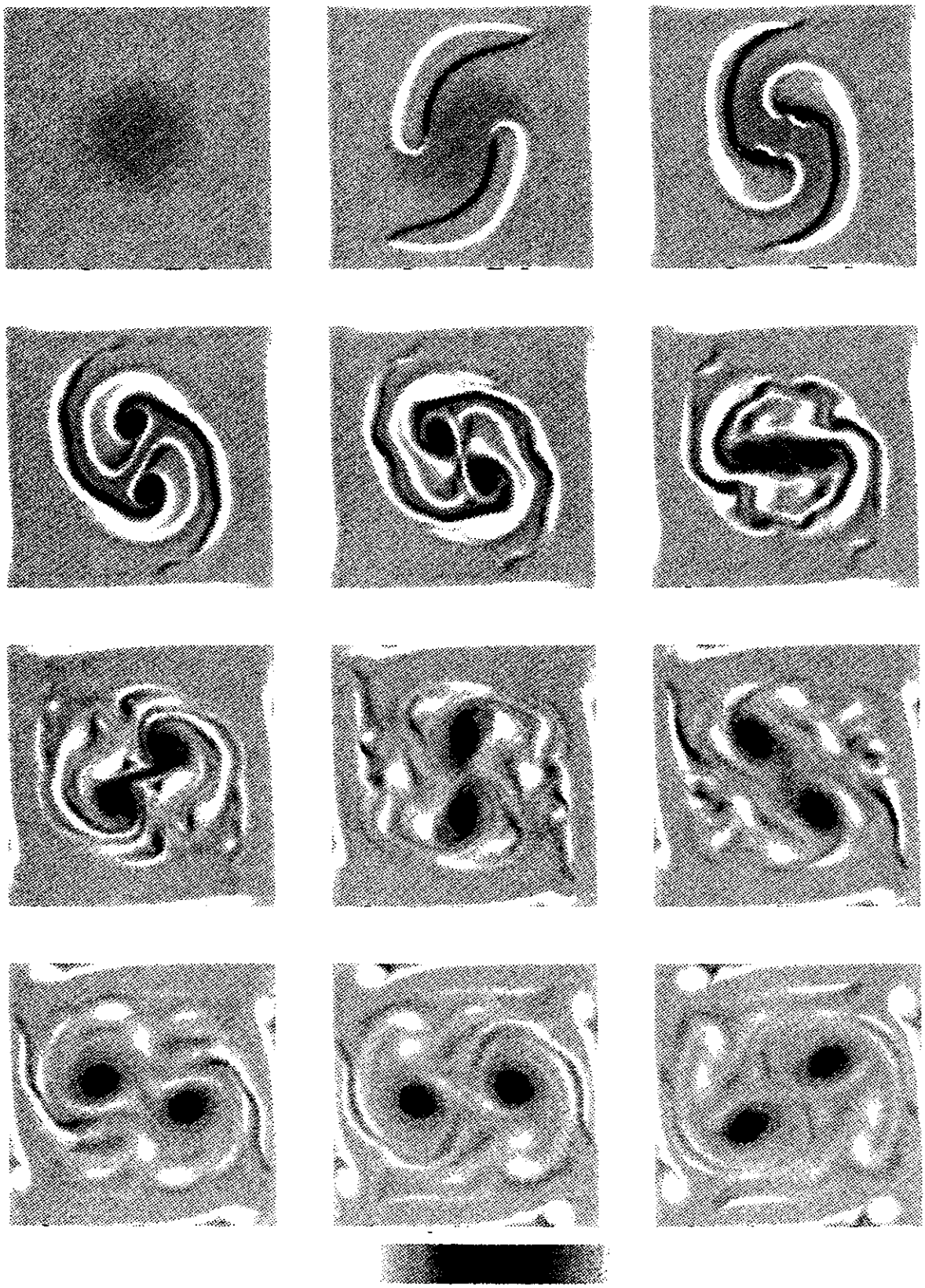


Figure 4.7 Vorticity contours for combustion in a closed container

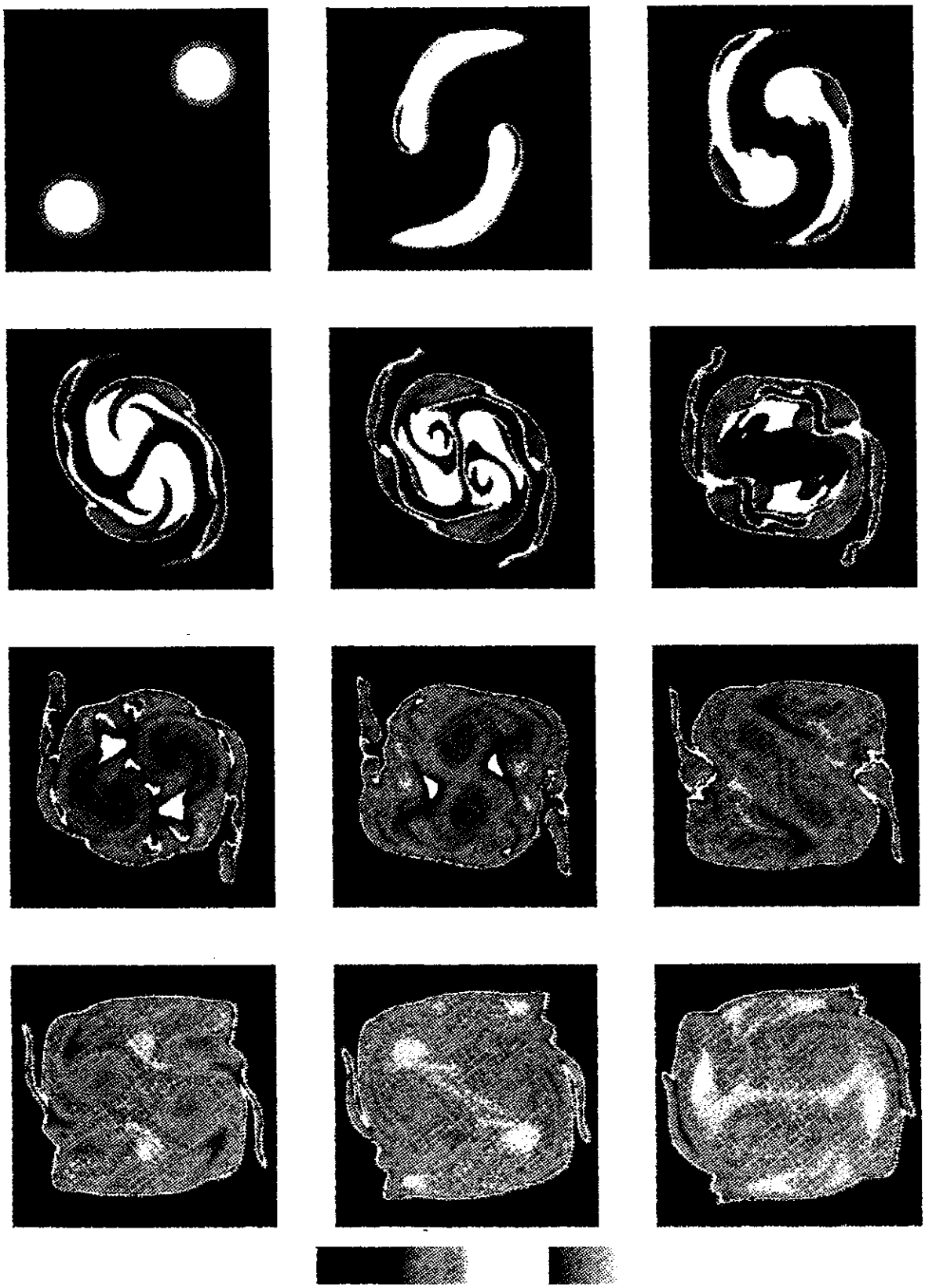


Figure 4.8 Temperature contours for combustion in a closed container

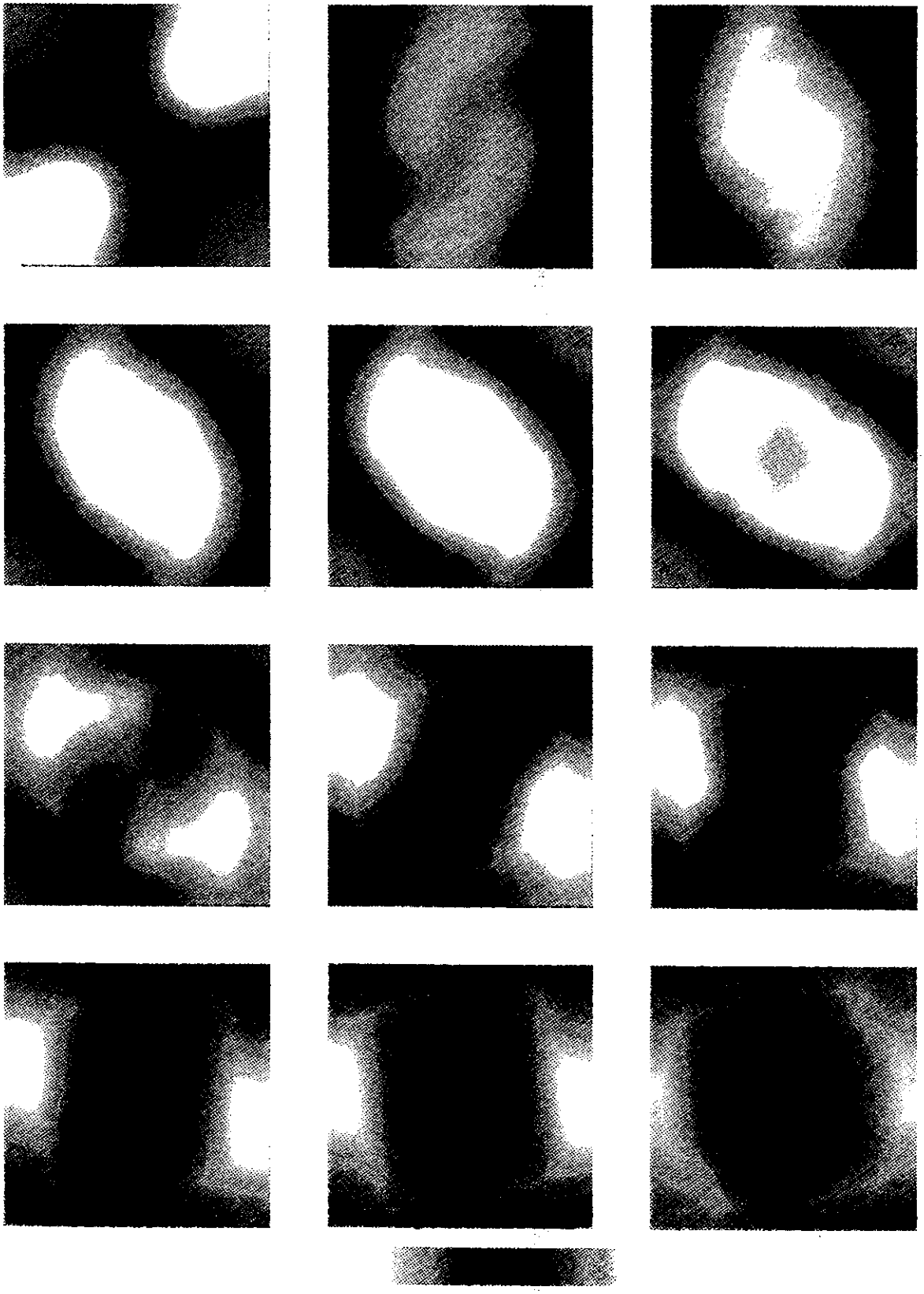


Figure 4.9 The scalar field of the velocity potential for combustion in a closed container

## References


[1] A.S. Almgren, J.B. Bell, P. Colella and L.H. Howell, "An Adaptive Projection Method for the Incompressible Euler Equations," *Proceedings, AIAA 11th Comp. Fluid Dynamics Conf.*, Orlando, FL, July 6-9 1993, pp. 530-539.

[2] A.S. Almgren, J.B. Bell, W.G. Szymczak, "A Numerical Method for the Incompressible Navier-Stokes Equations Based on an Approximate Projection," to appear.

[3] A.A. Amsden, J.D. Ramshaw, P.J. O'Rourke and J.K. Dukowicz, "KIVA: A Computer Program for Two- and Three-Dimensional Fluid Flows with Chemical Reactions and Fuel Sprays," *Los Alamos National Laboratory Report LA-10245-MS*, 1985.

[4] A.A. Amsden, J.D. Ramshaw, L.D. Cloutman, and P.J. O'Rourke, "Improvements and Extensions to the KIVA Computer Program," *Los Alamos National Laboratory Report LA-10534-MS*, 1985.

[5] A.A. Amsden, T.D. Butler, P.J. O'Rourke and J.D. Ramshaw "KIVA: A Comprehensive Model for 2D and 3D Engine Simulations," *SAE Paper 850554*, 1985.

[6] J.B. Bell, P. Colella and H.M. Glaz, "A Second Order Projection Method for the Incompressible Navier-Stokes Equations," *J. Comp. Phys.*, v. 85, pp. 257-283, 1989.

[7] J.B. Bell, P. Colella, and L.H. Howell, "An Efficient Second Order Projection Method for Viscous Incompressible Flow," *Proceedings, AIAA 10th Comp. Fluid Dynamics Conf.*, Honolulu, HI, June 24-26, 1991, pp.360-367.

[8] J.B. Bell, C.N. Dawson, and G.R. Shubin, "An Unsplit, Higher Order Godunov Method for Scalar Conservation Laws in Multiple Dimensions," *J. Comp. Phys.*, v. 74, no. 1, pp. 1-24, 1988.

[9] J.B. Bell and D.L. Marcus, "A Second Order Projection Method for Variable Density Flows," *J. Comp. Phys.*, v. 101, pp. 334-348, 1992.

[10] J.A. Benek, J.L. Steger, and F.C. Dougherty, "A Flexible Grid Embedding Technique with Application to the Euler Equations," *Proceedings, AIAA 6th Comp. Fluid Dynam-*

*ics Conf.*, Danvers, MA, July 13-15, 1983, pp.373-383.

[11] M.J. Berger and J. Oliger, "Adaptive Mesh Refinement for Hyperbolic Partial Differential Equations," *J. Comp. Phys.*, v. 53, pp. 484-512, 1984.

[12] M.J. Berger and P. Colella, "Local Adaptive Mesh Refinement for shock Hydrodynamics," *J. Comp. Phys.*, v. 82, no. 1, pp. 64-84, 1989.

[13] R.W. Bilger, "Turbulent Diffusion Flames," *Ann. Rev. Fluid Mech.*, v. 21, pp. 101-135, 1989.

[14] W.L. Briggs, *A Multigrid Tutorial*, SIAM, Philadelphia, PA, 1987.

[15] W.R. Briley, H. McDonald, and S.J. Shamroth, *AIAA Journal*, v. 21, p. 1467, 1983.

[16] I.-L. Chern and P. Colella, "A Conservative Front Tracking Method for Hyperbolic Conservation Laws," Technical Report UCRL-97200, Lawrence Livermore National Laboratory, July, 1987.

[17] G. Chesshire and W.D. Henshaw, "composite Overlapping Meshes for the Solution of Partial Differential Equations," *J. Comp. Phys.*, v. 90, no. 1, pp. 1-64, 1990.

[18] D. Choi and C.L. Merkle, *AIAA Journal*, v. 23, p.1518, 1985.

[19] Y.-H. Choi, Ph.D. thesis, Pennsylvania State University, University Park, PA, 1989 (unpublished.)

[20] A.J. Chorin, "A Numerical Method for Solving Incompressible Viscous Flow Problems," *J. Comp. Phys.*, v. 2, pp. 12-26, 1967.

[21] A.J. Chorin, "Numerical Solutions of the Navier-Stokes Equations," *Math. Comp.*, v. 22, pp. 745-762, 1968.

[22] A.J. Chorin, "On the Convergence of Discrete Approximations to the Navier-Stokes Equations," *Math. Comp.*, v. 23, pp. 341-353, 1969.

[23] A.J. Chorin, "Numerical Analysis of slightly Viscous Flow," *J. Fluid Mech.*, v. 57, p. 785, 1973.

[24] A.J. Chorin, "Flame Advection and Propagation Algorithms," *J. Comp. Phys.* v. 35, pp. 1-11, 1980.

[25] K.T. Coughlin and P. Marcus, "Turbulent Bursts in a Three-Dimensional Fluid," submitted to *Phys. Rev. Letters*.

[26] J.F. Dannenhoffer III and M.B. Giles, *AIAA Journal*, v. 28, p. 1457, 1990.

[27] D.G. Ebin, "The Motion of Slightly Compressible Fluid Viewed as Motion with a Strong Constraining Force,", *Ann. of Math.*, v. 105, pp. 141-200, 1977.

[28] J. Feng and C.L. Merkle, *AIAA Paper 90-0016*, 1990.

[29] T. Gal-Chen and R.C.J. Somerville, "Numerical Solution of the Navier-Stokes Equations with Topography," *J. Comp. Phys.*, v. 17, pp. 276-310, 1975.

[30] A.F. Ghoniem, A.J. Chorin and A.K. Oppenheim, "Numerical Modelling of Turbulent Flow in a Combustion Tunnel," *Phil. Trans., R. Soc. Lond., A*, v. 304, pp. 303-325, 1982.

[31] A.F. Ghoniem and J.A. Sethian, "Dynamics of Turbulent Structure in a Recirculating Flow; A Computational Study," *AIAA-85-0146*, AIAA 23rd Aerospace Sciences Meeting, Reno, NV, January, 1985.

[32] A.F. Ghoniem, "Computational Methods in Turbulent Reacting Flow," *Lectures in Applied Math.*, v. 24, pp.199-265, 1986.

[33] A.F. Ghoniem, "Effect of Large Scale Structures on Turbulent Flame Propagation," *Combustion and Flame*, v. 64, pp.321-336, 1986.

[34] A.F. Ghoniem and O.M. Knio, "Numerical Simulation of Flame Propagation in Constant Volume Containers," *Twenty-First Symposium (International) on Combustion*, The Combustion Institute, pp. 1313-1320, 1986.

[35] A.F. Ghoniem and P. Givi, "Vortex-Scalar Element Calculations of a Diffusion Flame," *AIAA-87-0225*, AIAA 25th Aerospace Sciences Meeting, Reno, NV, 1987.

[36] A.F. Ghoniem, G. Heidarinejad and A. Krishnan, "On Mixing, Baroclinicity and the Effect of Strain in a Chemically-Reacting Shear Layer," *AIAA-88-0729*, AIAA 26th Aerospace Sciences meeting, Reno, NV, January, 1988.

[37] A.F. Ghoniem and H.N. Najm, "Numerical Simulation of the Coupling Between Vor-

ticity and Pressure Oscillation in a Combustion Instability," *AIAA-89-2665*, AIAA/ASME/SAE/ASEE 25th Joint Propulsion Conference, Monterey, CA, July, 1989.

[38] P.M. Gresho, "On the Theory of Semi-Implicit Projection Methods for Viscous Incompressible Flow and Its Implementation Via a Finite Element Method that also Introduces a Nearly Consistent Mass Matrix, Part I," *International Journal for Numerical Methods in Fluids*, v. 11, pp. 587-620, 1990.

[39] F.J. Harlow and J.E. Welsh, "Numerical Calculation of Time Dependant Viscous Incompressible Flow with Free Surface," *Phys. Fluids*, v. 8, 1965.

[40] J.A. Hilditch, "A Front Tracking Method for Compressible Flames in One Dimension," Master's Thesis, University of California, Berkeley, Spring, 1993.

[41] M.J. Holst, "A Look at the KIVA-II Computer Program," Numerical Mathematics Group, Lawrence Livermore National Laboratory, 1992.

[42] J. Kim and P. Moin, "Application of a Fractional-Step Method to Incompressible Navier-Stokes Equations," *J. Comp. Phys.*, v. 59, pp. 308-323, 1985.

[43] S. Klainerman and A. Majda, "Compressible and Incompressible Fluids,", *Comm. on Pure and Appl. Math.*, v. 35, pp. 629-651, 1982.

[44] O.M. Knio and A.F. Ghoniem, "Vortex Simulation of a Three-Dimensional Reacting Shear Layer with Infinite-Rate Kinetics," AIAA Journal, v. 30, no. 1, pp. 105-116, 1992.

[45] K.K. Kuo, *Principles of Combustion*, John Wiley & Sons, New York, 1986.

[46] L.D. Landau and E.M. Lifshitz, *Fluid Mechanics*, Pergamon Press, Oxford, 1959.

[47] Y. Maday and A.T. Patera, "Spectral Element Method for the Incompressible Navier-Stokes Equations," in *State-of-the-Art Surveys on Computational Mechanics*, edited by A.K. Noor and J.T. Oden, ASME, New York, 1989.

[48] A. Majda and J.A. Sethian, "The Derivation and Numerical Solution of the Equations for Zero Mach Number Combustion," *Combust. Sci. Tech.*, v. 42, pp. 185-205, 1985.

[49] J.E. Melton, F.Y. Enomoto, and M.J. Berger, "3D Automatic Cartesian Grid Genera-

tion for Euler Flows," *Proceedings, AIAA 11th Comp. Fluid Dynamics Conf.*, Orlando, FL, July 6-9 1993, pp. 959-969.

[50] H.N. Najm and A.F. Ghoniem, "Modeling Pulsating Combustion due to Flow-Flame Interactions in Vortex-Stabilized Pre-mixed Flames," *Proceedings of the International Symposium on Pulsating Combustion*, Monterey, CA, August, 1991.

[51] P.J. O'Rourke, "The KIVA Computer Program for Multidimensional Chemically Reactive Fluid Flows with Fuel Sprays," *Numerical Simulations of Combustion Phenomena*, edited by R. Glowinski, B. Larrouturou, and R. Temam, Springer-Verlag, Berlin, p. 74, 1985.

[52] R.B. Pember, J.B. Bell, P. Colella, W.Y. Crutchfield, and M.W. Welcome, "Adaptive Cartesian Grid Methods for Representing Geometry in Inviscid Compressible Flow," *Proceedings, AIAA 11th Comp. Fluid Dynamics Conf.*, Orlando, FL, July 6-9 1993, pp. 948-958.

[53] R. Peyret and H. Viviand, *Recent Advanced in the Aerospace Sciences*, edited by C. Casci, Plenum, New York, 1985.

[54] R.H. Pletcher and K.-H. Chen, "On Solving the Compressible Navier-Stokes Equations for Unsteady Flows at Very Low Mach Numbers," *Proceedings, AIAA 11th Comp. Fluid Dynamics Conf.*, Orlando, FL, July 6-9 1993, pp. 765-775.

[55] R.G. Rehm and H. R. Baum, "The Equations of Motion for Thermally Driven, Buoyant Flows," *Journal of Research of the National Bureau of Standards*, v. 83, no. 3, pp. 297-308, 1978.

[56] C. Rhee, J.A. Sethian, and L. Talbot, "Dynamic Behavior of a Premixed Open V-Flame with Exothermicity and Baroclinicity," to be submitted, *J. Fluid Mechanics*, Sept. 1993.

[57] A.A. Samarskii, "An Efficient Difference Method for Solving a Multi-Dimensional Parabolic Equation in an Arbitrary Domain," *Z. Vycisl. Mat. i Mat. Fiz.*, v. 2, 1962, pp. 787-811 = *U.S.S.R. Comput. Math. and Math. Phys.*, 1963, 1964, no. 5, pp 894-926.

MR 32 #609.

[58] J. Sethian, "Turbulent Combustion in Open and Closed Vessels," *J. Comp. Phys.*, v. 54, no. 3, pp. 425-456, 1984.

[59] J.C. Simo and F. Armero, "Unconditional Stability and Long-Term Behavior of Transient Algorithms for the Incompressible Navier-Stokes and Euler Equations," to appear in *Comp. Meth. Appl. Mech. Eng.*

[60] G.I. Sivashinsky, "Hydrodynamic Theory of Flame Propagation in an Enclosed Volume," *Acta Astronautica*, v. 6, pp. 631-645, 1979.

[61] J.C. Strikwerda, "Finite Difference Methods for the Stokes and Navier-Stokes Equations," *SIAM J. Sci. Stat. Comput.*, v. 5, no. 1, pp 56-68, 1984.

[62] J.C. Strikwerda, *Finite Difference Schemes and Partial Differential Equations*, Wadsworth and Brooks/Cole Mathematics Series, Pacific Grove, CA, 1989.

[63] E. Turkel, *J. Comp. Phys.*, v. 72, p. 277, 1987.

[64] J. van Kan, "A Second-Order Accurate Pressure-Correction Scheme for Viscous Incompressible Flow," *SIAM J. Sci. Stat. Comput.*, v. 7, no. 3, pp. 870-891, 1986.

[65] B. van Leer, "Multidimensional Explicit Difference Schemes for Hyperbolic Conservation Laws," in *Computing Methods in Applied Sciences and Engineering, VI*, pp. 493 (Elsevier Science, Amsterdam, 1984).

[66] B. van Leer, C.-H. Tai and K.G. Powell, AIAA Paper 89-1933, 1989.

[67] H. Viviand, *Numerical Methods for the Euler Equations of Fluid Dynamics*, edited by F. Angrand et. al., SIAM, Philadelphia, 1985.

[68] D.P. Young, R.G. Melvin, M.B. Bieterman, F.T. Johnson, S.S. Samani, and J.E. Bussoletti, "A Locally Refined Rectangular Grid Finite Element Method: Application to Computational Fluid Dynamics and Computational Physics," J. Comp. Phys, v. 92, no. 1, pp. 1-66, 1991.

[69] J.Y. Zhu and J.A. Sethian, "Projection Methods Coupled to Level Set Interface Techniques," J. Comp. Phys, v. 102, no. 1, pp 128-138, 1992.

# Appendix

## A.1 The Multigrid Method

The multigrid method is a technique for reducing the computational time required to solve linear systems, and it can be used in the numerical scheme described in this thesis to considerably lower the run-time and expense. See Briggs [3a] for an introduction to multigrid.

Multigrid is used for solving problems of the form $Lu = f$ on a domain $\Omega$ where L is a linear operator, given some boundary conditions for u. Let v denote the numerical solution to the problem. It is possible to find the solution iteratively using

$$v_{i,j}^{l+1} := (v_{i,j}^{l} - \lambda (L^h v^l)_{i,j} + \lambda f_{i,j}) \tag{A.1}$$

where l is an iteration index, $L^h$ is the discrete version of our linear operator, and $\lambda$ is a relaxation parameter. After a sufficient number of iterations, v will be close to u. When equation (A.1) is applied to v, we are said to "be relaxing on" v.

The choice for $\lambda$ can significantly affect the speed of the calculation; $\lambda$ is often defined in a way that cancels all terms of $v_{i,j}^{l}$ from the right hand side of equation (A.1). For example, for a standard five-point Laplacian where $\Delta x = \Delta y = h$, $\lambda$ would be set to $h^2/4$ so that

$$\begin{aligned} v_{i,j}^{l+1} &:= \left( v_{i,j}^{l} - \frac{h^2}{4} \left( \frac{v_{i+1,j}^{l} + v_{i-1,j}^{l} + v_{i,j+1}^{l} + v_{i,j-1}^{l} - 4v_{i,j}^{l}}{h^2} \right) + \frac{h^2}{4} f_{i,j} \right) \\ &= -(v_{i+1,j}^{l} + v_{i-1,j}^{l} + v_{i,j+1}^{l} + v_{i,j-1}^{l}) + \frac{h^2}{4} f_{i,j}. \end{aligned} \tag{A.2}$$

Let $v^l = u + \delta^l$ where $\delta$ is the error between the exact and computed solution. Also let $R = L^h v^l - f$ where R stands for *residual*. Given these definitions, the problem for v can be equivalently rewritten as a problem for $\delta$

$$\begin{aligned} L^h v^l &= L^h u + L^h \delta^l, \\ L^h \delta^l &= L^h v^l - f = R. \end{aligned} \tag{A.3}$$

Therefore, as $v^l \to u$, $\delta^l \to 0$.

Gauss-Seidel and Jacobi are two common choices for relaxation schemes. They are virtually identical methods; in Gauss-Seidel, $v^{l+1}$ depends on $v^l$ and values of $v^{l+1}$ that have already been found, while in Jacobi relaxation $v^{l+1}$ depends on $v^l$ only. These schemes can be used with

red-black ordering, a term which refers to the order in which one solves for $v^{l+1}$. To implement red-black ordering, first solve on the "red" cells (referring to a checkerboard pattern), then finish the iteration by solving on the "blacks."

Relaxation alone, however, is too slow for our purposes. While high frequencies in the solution are damped efficiently, low frequencies will persist. Multigrid solves this problem by performing the same relaxation on both coarse and fine grids, the rationale being that a low frequency mode on a fine grid will appear as a high frequency mode on a coarse grid. By systematically relaxing on these difference grids and interpolating the solution on adjacent levels, a solution can be found in a much shorter time.

Multigrid has been implemented using a V-cycle, where "V" describes the succession of grids relaxed upon. A sample V-cycle for a grid with 16 cells per side is shown in Figure A.1.

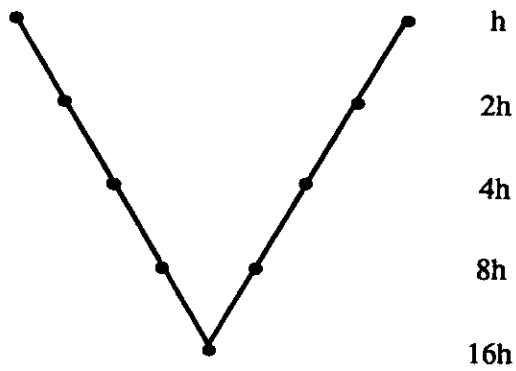



Figure A.1 One Multigrid V-cycle

Here are the steps in detail that make up a single V-cycle for a grid with the coarsest spacing given by Nh. Begin on the finest mesh given v, f, and h and proceed as follows:

Relax on $L^h u^h = f^h$

Coarsen $f^{2h} = I_h^{2h} R^h$

Relax on $L^{2h} u^{2h} = f^{2h}$

Compute $f^{4h} = I_{2h}^{4h} R^{2h}$

Relax on $L^{4h} u^{4h} = f^{4h}$

Compute $f^{8h} = I_{4h}^{8h} R^{4h}$

...

Solve $L^{Nh} u^{Nh} = f^{Nh}$

...

Correct $u^{4h} := (u^{4h} + I_{4h}^{8h} u^{4h})$

Relax on $L^{4h} u^{4h} = f^{4h}$

Correct $u^{2h} := (u^{2h} + I_{2h}^{4h} u^{2h})$

Relax on $L^{2h} u^{2h} = f^{2h}$

Correct $u^{h} := (u^{h} + I_{h}^{2h} u^{h})$

Relax on $L^{h} u^{h} = f^{h}$

To coarsen, an interpolating function is used. This function is given by

$$(I_h^f R^f)_{i,j} = \frac{1}{4}(R^f_{2i,2j} + R^f_{2i-1,2j} + R^f_{2i,2j-1} + R^f_{2i-1,2j-1}) . \tag{A.4}$$

where an f superscript denotes fine mesh values and a c superscript denotes coarse mesh values. Since values of the density appear in the operator, these too need to be averaged. In the MAC projection, where values of density are required on the mesh, averaging is done by

$$\begin{aligned} \rho^c_{i-1/2,j} &= \frac{2}{1/\rho^f_{2i-3/2,2j} + 1/\rho^f_{2i-3/2,2j-1}}, \\ \rho^c_{i,j-1/2} &= \frac{2}{1/\rho^f_{2i,2j-3/2} + 1/\rho^f_{2i-1,2j-3/2}}. \end{aligned} \tag{A.5}$$

In the parabolic equations where values of density are required at cell centers, averaging is done by

$$\rho^c_{i,j} = \frac{1}{4}(\rho^f_{2i,2j} + \rho^f_{2i-1,2j} + \rho^f_{2i,2j-1} + \rho^f_{2i-1,2j-1}) . \tag{A.6}$$

The correction step where the coarse correction is interpolated onto the fine mesh is given by

$$\begin{aligned} v^f_{2i,2j} &= v^f_{2i,2j} - \delta^c_{i,j}, \\ v^f_{2i,2j-1} &= v^f_{2i,2j-1} - \delta^c_{i,j}, \\ v^f_{2i-1,2j} &= v^f_{2i-1,2j} - \delta^c_{i,j}, \\ v^f_{2i-1,2j-1} &= v^f_{2i-1,2j-1} - \delta^c_{i,j}. \end{aligned} \tag{A.7}$$

## Legal Notice

This report was prepared as an account of work sponsored by the Center for Pure and Applied Mathematics. Neither the Center nor the Department of Mathematics makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information or process disclosed.