To the Graduate Council:

I am submitting herewith a thesis written by Fred Dean Carter, Jr. entitled "Accuracy and Stability of a Finite Element Pseudo-Compressibility CFD Algorithm for Incompressible Flows." I have examined the final copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Engineering Science.

A J Baker

A.J. Baker, Major Professor

We have read this thesis
and recommend its acceptance:

Osama Soliman

Kenneth Kimble

Accepted for the Council:

C W Minkel

Vice Provost
and Dean of The Graduate School

STATEMENT OF PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a Master's degree at The University of Tennessee, Knoxville, I agree that the Library shall make it available to borrowers under rules of the Library. Brief quotations from this thesis are allowable without special permission, provided that accurate acknowledgement of the source is made.

Permission for extensive quotation from or reproduction of this thesis may be granted by my major professor, or in his absence, by the Head of Interlibrary Services when, in the opinion of either, the proposed use of the material is for scholarly purposes. Any copying or use of the material in this thesis for financial gain shall not be allowed without my written permission.

Signature Fred Dean Carter, Jr.

Date May 1990

ACCURACY AND STABILITY OF A FINITE ELEMENT PSEUDO-COMPRESSIBILITY CFD ALGORITHM FOR INCOMPRESSIBLE FLOWS

A Thesis

Presented for the Master of Science

Degree

The University of Tennessee, Knoxville

F. Dean Carter, Jr.

May 1990

ACKNOWLEDGMENTS

During the course of my graduate study, I have been assisted by a number of individuals. I would like to express my appreciation to my major professor, Dr. A.J. Baker, for his consultation, guidance, and friendship. I would also like to thank my committee members: Dr. M.O. Soliman for his constant support and encouragement and Dr. K.H. Kim for his constructive criticism of this thesis. Thanks are due to G.S. Iannelli and Albert M. Hines for their availability and unfailing willingness to provide assistance during my thesis work. I am grateful to Dr. J.E. Stoneking and the Department of Engineering Science and Mechanics for financial assistance provided through my graduate teaching assistantship. Special thanks are due to Eunice Hinkle whose friendship and help were invaluable. Thanks also are extended to the Computational Fluid Dynamics (CFD) Laboratory and the Computing Center at the University of Tennessee for generous use of their facilities.

The computer programs referred to in this thesis are resident in the CFD Laboratory of the Department of Engineering Science and Mechanics. Any errors occurring in this thesis or those programs are my sole responsibility and do not reflect on the individuals listed here.

ABSTRACT

The term "pseudo-compressibility" refers to a technique used in numerically solving incompressible flows. In this approach, the pressure field solution is obtained by introducing a pressure time derivative term into the continuity equation to form a set of hyperbolic-type equations. The objective of this thesis is to implement the method through modification of an existing 2-D compressible flow Navier-Stokes finite element code and to evaluate the algorithm's accuracy and stability.

The non-dimensional form of the pseudo-compressibility equation set is presented and some theoretical issues concerning the pseudo-compressibility parameter are discussed. A companion conservation law system is derived via identification of truncation error terms in a Taylor series expansion. A weak statement is then formed, and is subsequently semi-discretized in space by the finite element method. Finally, the algebraic construction of the algorithm is discussed, including an implicit time integration scheme and boundary condition enforcement.

Three benchmark test cases are considered for code and algorithm verification. The results obtained compare well with those documented in the literature. Stability and convergence problems were encountered, however, pertaining to oscillations in the pseudo-pressure field, especially near boundary corner geometrical singularities. This and other computational issues are evaluated and discussed.

Contents

List of Figures

List of Symbols

A_j	Jacobian matrix of the flux vector scalar components
$\mathbf{b}$	body force
$\mathcal{B}$	coefficient of volumetric expansion
c	pseudo sound speed
c_v	specific heat
C	Courant number
$\mathbf{D}_k$	diagonal TWS stability matrix
EJK_e	element metric data array
$\mathbf{f}$, f_j	kinematic–kinetic flux vector, scalar components
$\mathbf{f}^\nu$, f_j^ν	dissipation flux vector, scalar components
$\{F\}$	Newton algorithm residual column matrix
g	gravitational acceleration
$\hat{g}$	gravity unit vector
h	local length measure
J_e, $(e_{jk})_e$	element coordinate transformation Jacobian matrix
k	thermal conductivity

L	channel length
L_e	entrance length
M	pseudo Mach number
$[M]$	mass matrix
$[M\ldots]$	finite element master matrix
n	dimension of problem domain
$\mathbf{n}$, n_j	outwards pointing unit vector, scalar components
$\{N_k\}$	trial function basis set of degree k
p	pressure
p_{ref}	reference pressure
q	state variable
q^h	approximation of state variable
$\{Q\}$	approximation nodal state variable column matrix
R^n	n-dimensional problem domain
$\{R\}$	residual
s	source term
t	time
t_L	time required for upstream traveling pseudo pressure wave to travel the length L of a channel
t_{2L}	time required for pseudo pressure wave to make one round trip in a channel of length L
T	dimensional temperature
u, v	cartesian velocity components

$\mathbf{u}$, u_i	velocity vector, scalar components
u_{ref}	reference velocity
u_{avg}	average velocity in the streamwise direction
w_i	test function set
x_{ref}	reference length
$\{\mathbf{X}\}_e$	element nodal coordinate column matrix
α	thermal diffusivity
β	diagonal TWS theory stability matrix
Γ	boundary of problem domain
δ	thickness of vorticity spreading
δ_{ij}	Kronecker delta
ϵ_j	unit vector scalar components
ε	dimensionless pseudo-compressibility
$\tilde{\varepsilon}$	dimensional pseudo-compressibility
η_j	local orthonormal coordinate system
θ	implicitness parameter
μ	dynamic viscosity
ν	kinematic viscosity
ρ_0	constant fluid density
Φ	viscous dissipation function
Ω	problem domain
Ω^h	discretized problem domain

Ω_e	finite element domain
Gr	Grashof number
Pr	Prandtl number
Ra	Rayleigh number
Re	Reynolds number
$\bigcup_e$	union over all elements
$\otimes$	matrix tensor product
S_e	assembly operator
$\{\cdot\}_e$	element nodal column matrix

Chapter 1

Introduction

In seeking numerical solutions for the incompressible Navier-Stokes conservation law system, the special role of pressure as a dynamic parameter presents the major difficulty in developing a solution procedure. It is in one sense a "mathematical artefact – a Lagrange multiplier that constrains the velocity field to remain divergence free" [1], while in actuality, the pressure adjusts instantaneously to stay in equilibrium with a time-varying divergence-free velocity field. In direct contrast with the compressible flow case, however, no equation of state exists relating pressure and density, and the continuity equation is devoid of a time derivative. Hence, some method must be devised to handle the pressure in the design of any numerical solution procedure for an incompressible flow.

The only mathematically exact way to ensure a divergence free velocity field employs vector field theory to recast the differential system in terms of the vector potential (streamfunction) and vorticity vector. The eliminated pressure distribution may then be calculated in a post-processing operation. The major drawbacks to this method include enforcement of vorticity boundary conditions and extension to three-dimensions. Another widely applied method of solving for the pressure is the Poisson equation approach developed by Harlow and Welch [2], in which a pressure field is computed such that continuity is "satisfied" at each time step. Usually a relaxation scheme iterating on pressure is employed until a reasonably divergence free condition is achieved. This method can be very time consuming, especially in three dimensional flows. Other methods of handling pressure can be found in the literature (for example, see [3]).

If the goal is to obtain the steady-state solution only, any of the aforementioned methods can be computationally wasteful. Chorin [4] proposed introducing artificial compressibility by adding the time derivative of pressure to the continuity equation. This leads to a hyperbolic-type conservation law system in which pressure waves of finite speed are introduced into the incompressible flow computation. When the solution reaches steady-state, the pressure time term vanishes and a divergence free velocity field is obtained. The method has been applied to fluid flow and heat transfer problems in both two and three space dimensions and several numerical analyses have also been performed [4]-[14]. For a survey of this work, see Peyret and Taylor [15].

All of the cited numerical analyses utilizing the method of artificial compressibility have employed finite difference techniques, with the exception of [12] in which a finite volume procedure is utilized. In this research project, the method has been implemented into an existing finite element compressible flow code, see Iannelli [16]. This code, designed for compressible Navier-Stokes and Euler aerodynamics applications, required relatively minor modifications in the implementation of the pseudo-compressibility formulation. The prime objective of this project was this implementation of Chorin's method, and acquisition of computational results for several benchmark problems to confirm operation and determine limitations. In Chapter 2, the mathematical formulation is described in detail and some theoretical issues concerning the artificial compressibility are presented. In Chapter 3, the CFD algorithm formulation is described, and in Chapter 4, the test cases are presented and discussed.

Chapter 2

Mathematical Formulation

2.1 Governing Equation System

The two-dimensional, laminar flow of a viscous, incompressible fluid with constant properties is governed by a system of four partial differential equations derived from the basic conservation laws for the continuum description, i.e., conservation of mass, linear momentum, and energy. In rectangular cartesian vector form these equations are, respectively

$$L(\rho_0) = \nabla \cdot \mathbf{u} = 0 \tag{2.1}$$

$$L(\mathbf{u}) = \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - \nabla \cdot \nu \nabla \mathbf{u} + \frac{1}{\rho_0} \nabla p + \frac{\mathbf{b}}{\rho_0} = 0 \tag{2.2}$$

$$L(T) = c_v \frac{\partial T}{\partial t} + c_v (\mathbf{u} \cdot \nabla) T - \nabla \cdot \frac{k}{\rho_0} \nabla T - \Phi - s_T = 0 \tag{2.3}$$

where $\mathbf{u}$ is the n-dimensional velocity vector, ν is the kinematic viscosity of the fluid, ρ_0 is the constant fluid density, p is the pressure, $\mathbf{b}$ is the body force, c_v is the specific heat of the fluid, k is the thermal conductivity, Φ is the viscous dissipation function, and s_T is a source term.

Analysis will be limited to flows with vanishingly small Mach number such that the viscous dissipation Φ can be neglected. For this case, density variations exist principally due to temperature gradients in the flow field, giving rise to a significant buoyancy body force term in equation (2.2). The Boussinesq expression is

$$\mathbf{b} = \rho_0 g \mathcal{B}(T - T_{ref})\hat{g} \tag{2.4}$$

where $\hat{g}$ is the gravity unit vector, g is the gravitational acceleration, $\mathcal{B}$ is the coefficient of volumetric expansion, and the subscript "ref" refers to the reference state [18].

To obtain the pseudo-compressible reformulation of equation set (2.1)-(2.3) as proposed by Chorin [4], the continuity equation (2.1) is modified by adding a fictitous pressure time term yielding,

$$\frac{1}{\rho_0 \tilde{\varepsilon}} \frac{\partial p}{\partial t} + \nabla \cdot \mathbf{u} = 0 \tag{2.5}$$

where $\tilde{\varepsilon}$ is termed the "pseudo-compressibility." This added term changes the original elliptic type system of equations, (2.1)-(2.3), into a hyperbolic system, and introduces waves of finite speed to distribute the pressure. For a truly incompressible flow, the wave speed is infinite, whereas the speed of propagation of these pseudo-waves depends on the magnitude of the pseudo-compressibility $\tilde{\varepsilon}$. Note that a physically meaningful incompressible flow solution is achieved only when the effect of the pressure time term is negligible, corresponding to a steady state.

The pseudo-compressible equation set, consisting of equations (2.5), (2.2), and

(2.3) is made non-dimensional by utilizing the following dimensionless quantities:

$$\begin{array}{lll} \varepsilon^* = \frac{\bar{\varepsilon}}{u_{ref}^2} & \mathbf{u}^* = \frac{\mathbf{u}}{u_{ref}} & k^* = \frac{k}{k_{ref}} = 1 \\ s_\theta^* = \frac{s_T}{s_{T(ref)}} & \nabla^{2*} = \nabla^2 x_{ref}^2 & c_v^* = \frac{c_v}{c_{v(ref)}} = 1 \\ p^* = \frac{p - p_{ref}}{\rho_{ref} u_{ref}^2} & \Theta^* = \frac{T - T_{ref}}{T_{max} - T_{ref}} & \nu^* = \frac{\nu}{\nu_{ref}} = 1 \\ t^* = \frac{t u_{ref}}{x_{ref}} & \nabla^* = \nabla x_{ref} & \rho^* = \frac{\rho_0}{\rho_{ref}} = 1 \end{array} \tag{2.6}$$

An asterisk denotes a dimensionless quantity, the subscript "ref" refers to a reference quantity, and the subscript "max" refers to the maximum value of the quantity in the flow field. Substituting these expressions into equations (2.5), (2.2), and (2.3) yields the final non-dimensional form of the pseudo-compressibility conservation law system as

$$L(p) = \frac{1}{\varepsilon}\frac{\partial p}{\partial t} + \nabla \cdot \mathbf{u} = 0 \tag{2.7}$$

$$L(\mathbf{u}) = \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - \nabla \cdot \frac{1}{Re}\nabla \mathbf{u} + \nabla p + \frac{Gr}{Re^2}\Theta \hat{g} = 0 \tag{2.8}$$

$$L(\Theta) = \frac{\partial \Theta}{\partial t} + (\mathbf{u} \cdot \nabla)\Theta - \nabla \cdot \frac{1}{Re}\left(\frac{1}{Pr}\right)\nabla\Theta - s_\Theta = 0 \tag{2.9}$$

where all quantities are dimensionless, hence the asterisk notation is omitted. The important dimensionless parameters are

$$\text{Reynolds Number :} \quad Re = \frac{x_{ref} u_{ref}}{\nu_{ref}} \tag{2.10}$$

$$\text{Grashoff Number :} \quad Gr = \frac{g\mathcal{B}(T_{max} - T_{ref})x_{ref}^3}{\nu_{ref}^2} \tag{2.11}$$

$$\text{Prandtl Number :} \quad Pr = \frac{\nu_{ref}\rho_0 c_{v(ref)}}{k_{ref}} \tag{2.12}$$

2.2 Boundary Conditions

The pseudo-compressibility equation set (2.7)-(2.9) represents a non-linear, initial-value hyperbolic boundary value problem for infinite Reynolds number. Specification of proper boundary conditions is thus necessary for existence of the solution to the equation system. Through a characteristic analysis and using Chakravarthy's theory for well-posedness of hyperbolic initial boundary value problems [17], well-posed inlet/outlet boundary condition sets can be determined for the isothermal, inviscid form of the pseudo-compressibility equations (see also, [11] and [14]).

At the inlet, two Dirichlet specifications of data are needed for isothermal flow. The following sets of boundary conditions are considered:

1. Inlet pressure p and transverse velocity v specified.

2. Inlet principal and transverse velocities (u and v) specified.

3. Inlet pressure p and principal velocity u specified.

The theoretical analysis verifies that boundary condition sets 1 and 2 are admissible, whereas choice 3 leads to an ill posed mathematical model.

At the outlet only one Dirichlet constraint is admissible, and the possible options are

1. Outlet pressure p specified.

2. Outlet principal velocity u specified.

3. Outlet transverse velocity v specified.

The theory confirms that specifying either the pressure or principal velocity is admissible, whereas imposing the transverse velocity alone is not permitted.

For the inviscid case under consideration, it was also verified that specifying $\mathbf{u} \cdot \mathbf{n}$ is adequate along solid boundary walls. For a viscous flow, both components of $\mathbf{u}$ are set to zero to enforce the no-slip condition. In either case, viscous flow or inviscid flow, no pressure boundary condition is required or admissible along walls. The enforcement of these various boundary conditions is discussed in Chapter 3.

2.3 The Pseudo-compressibility

In formulating the pseudo-compressibility equations for incompressible flow, the pseudo-compressibility constant $\tilde{\varepsilon}$ is introduced, as is the question of what numerical value or range it should be given. An extensive discussion concerning optimum $\tilde{\varepsilon}$ values for inviscid flows can be found in [11]. For viscous flows, Chang and Kwak [8] discuss theoretical issues concerning $\tilde{\varepsilon}$, and a brief summary of their analysis is now presented.

In an incompressible flow, pressure disturbances travel with infinite speed. With the introduction of the pressure time term, these waves travel with finite speed, and the magnitude of the speed depends on $\tilde{\varepsilon}$. In using this method, there is thus a time lag between a flow disturbance and its effect on the pressure field, whereas in a true incompressible flow, the pressure field adjusts instantaneously to any such disturbance. In viscous flows, the behavior of the boundary layer is very sensitive to the streamwise pressure gradient, especially in regions where separation may occur. Therefore, if the pressure waves do not travel quickly enough, the solution may diverge.

As shown in references [8] and [11], the pseudo sound speed is defined to be

$$c \equiv \sqrt{u_{avg}^2 + \tilde{\varepsilon}} \tag{2.13}$$

where u_{avg} is the average speed in the streamwise direction and $\tilde{\varepsilon}$ is the dimensional pseudo-compressibility. Relative to this sound speed, the pseudo Mach number can be expressed as

$$M = \frac{u_{avg}}{c} = \frac{u_{avg}}{\sqrt{u_{avg}^2 + \tilde{\varepsilon}}} \tag{2.14}$$

For the pseudo waves to be able to travel upstream, M must be less than one, which is always true for $\tilde{\varepsilon} > 0$. This is a necessary condition so that an outlet boundary condition can affect the whole flow field.

Considering a channel of length L and width x_{ref}, the time required for an upstream traveling wave to travel the length L of the channel is

$$t_L = \frac{L}{c - u_{avg}} \tag{2.15}$$

The rate of vorticity spreading for laminar flow is given approximately by

$$\frac{d\left(\frac{\delta}{x_{ref}}\right)^2}{dt^*} \simeq \frac{4}{Re} \tag{2.16}$$

where t^* is defined as in (2.6) and Re is based on the width of the channel, x_{ref} [8]. Integrating (2.16), the time required for vorticity to spread a thickness δ is approximately given as

$$t_\delta^* \simeq \frac{Re}{4}\left(\frac{\delta}{x_{ref}}\right)^2 \tag{2.17}$$

Thus, in order for the viscous boundary layer to adjust to its new pressure environment, it is required to make

$$t_L^* \ll t_\delta^* \tag{2.18}$$

where t_L^* is obtained from the non-dimensionalization of (2.15). Setting $\frac{u_{avg}}{u_{ref}} = 1$, $\frac{\delta}{x_{ref}} = 1$, and utilizing (2.17), (2.18), and the non-dimensional forms of (2.13) and (2.15), the following restriction on ε can be derived [8]:

$$\varepsilon \gg \left[1 + \frac{4}{Re}\left(\frac{L}{x_{ref}}\right)\right]^2 - 1 \tag{2.19}$$

In [13], it is found that for most problems an ε in the range 1-10 is satisfactory.

Another important aspect of the parameter $\tilde{\varepsilon}$ is its effect upon the convergence of a problem's solution. To reach a converged steady-state solution, the pseudo-waves must make at least one round trip in distributing the pressure. The time required for these waves to travel downstream and back upstream, a total distance of $2L$, is

$$t_{2L} = \frac{L}{c+u} + \frac{L}{c-u} \tag{2.20}$$

From (2.13) and (2.20), the minimum time required to achieve a steady-state solution is

$$t_{2L} = \frac{\sqrt{u_{avg}^2 + \tilde{\varepsilon}}}{\tilde{\varepsilon}} 2L \tag{2.21}$$

Note that t_{2L} in (2.21) is a dimensional quantity. The validity of this requirement is discussed in Chapter 4.

Chapter 3

Numerical Computational Fluid Dynamics Algorithm

3.1 Companion Conservation Law System

The spatial semi-discretization of (2.7)-(2.9) introduces an approximation dispersion error that can corrupt the solution and possibly lead to iterative divergence. To smooth out high-frequency oscillations on inadequate discretizations, hence make the computation stable, numerical dissipation (smoothing) may be added. Herein, "artificial viscosity" terms are developed via a modification of (2.7)-(2.9) upon truncation error term identification as dictated by the Taylor weak statement (TWS) theory, see Baker and Kim [19]. This results in a companion conservation law system for the pseudo-compressible equation system (2.7)-(2.9).

Neglecting the heat source and body force terms in this analysis, the n-dimensional form of (2.7)-(2.9) can be symbolically expressed as

$$L(q) = \frac{\partial q}{\partial t} + \nabla \cdot (\mathbf{f} - \mathbf{f}^{\nu}) = 0 \tag{3.1}$$

where q is the state variable, $\mathbf{f} = \mathbf{f}(q)$ denotes the kinematic-kinetic flux vector, and $\mathbf{f}^\nu = \mathbf{f}^\nu(q)$ represents the dissipation fluxes. In index notation, the flux vectors $\mathbf{f}$ and $\mathbf{f}^\nu$ are defined as

$$\mathbf{f} \equiv f_j \epsilon_j \qquad \text{and} \qquad \mathbf{f}^\nu \equiv f_j^\nu \epsilon_j \tag{3.2}$$

where ϵ_j denotes the unit vector scalar components parallel to a cartesian system. The array components of q, f_j, and f_j^ν are defined as follows:

$$q \equiv \left\{ \begin{array}{c} p \\ u_i \\ \Theta \end{array} \right\} \quad , \quad f_j \equiv \left\{ \begin{array}{c} u_j \\ u_j u_i + p\delta_{ij} \\ u_j \Theta \end{array} \right\} \quad , \quad f_j^\nu \equiv \left\{ \begin{array}{c} 0 \\ \frac{1}{Re}\frac{\partial u_i}{\partial x_j} \\ \frac{1}{RePr}\frac{\partial \Theta}{\partial x_j} \end{array} \right\} \tag{3.3}$$

Further, u_i is the i-th component of the velocity vector, δ_{ij} is the Kronecker delta, and the tensor indices range $1 \leq (i,j) \leq n$.

To obtain the companion conservation law form, q is expanded in a temporal Taylor series as

$$q^{n+1} = q^n + \left(\frac{\partial q}{\partial t}\right)^n \Delta\, t + \left(\frac{\partial^2 q}{\partial t^2}\right)^n \frac{\Delta t^2}{2} + \ldots \tag{3.4}$$

Then, the inviscid form of (3.1) is utilized to replace the temporal derivatives of q with spatial derivatives of $\mathbf{f}$. Neglecting the higher order terms and taking the limit as $\Delta t \to 0$, (3.4) becomes

$$\frac{\partial q}{\partial t} + \frac{\partial f_j}{\partial x_j} - \frac{\partial}{\partial x_j}\left[A_j A_k \,\Delta\, t \beta^* \frac{\partial q}{\partial x_k}\right] = 0 \tag{3.5}$$

where $1 \leq (j,k) \leq n$, $A_j \equiv \frac{\partial f_j}{\partial q}$ is the Jacobian matrix of the flux vector components f_j, and $\beta^* \geq 0$ is arbitrary.

Equation (3.5) is now simplified as follows. To preclude time step dependence at steady state, Δt is replaced by the Courant number definition

$$\Delta\, t \equiv \frac{h}{|\mathbf{u}|} C \tag{3.6}$$

where $|\mathbf{u}|$ is a generalized local speed, h is a representative length measure, and C is the Courant number. Substituting (3.6) into (3.5) yields

$$\frac{\partial q}{\partial t} + \frac{\partial f_j}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\left(A_j A_k \frac{h}{|\mathbf{u}|}\right)\beta \frac{\partial q}{\partial x_k}\right] = 0 \tag{3.7}$$

where β contains C and is generalized to a square diagonal matrix of order equal to that of q. This generalization is achieved by admitting a separate stability parameter for each component of q. Thus, the user defined "β-terms" contained in β (i.e. β_p, β_u, β_v, β_Θ) are used to dose the level of artificial dissipation in the p, u, v, and Θ equations, respectively.

Finally, contracting the matrix product $A_j A_k$ with the Kronecker delta δ_{jk} and retaining only the diagonal terms, a diagonal matrix $\mathbf{D}_{\underline{j}}$ is formed. Reinserting the viscous fluxes $\mathbf{f}^\nu$ and the source term s, the final companion conservation law form of the pseudo-compressible equation system is obtained as,

$$\frac{\partial q}{\partial t} + \frac{\partial (f_j - f_j^\nu)}{\partial x_j} - s - \frac{\partial}{\partial x_j}\left[(\beta \mathbf{D}_{\underline{j}})\frac{\partial q}{\partial x_j}\right] = 0 \tag{3.8}$$

where $(\beta\mathbf{D}_{\underline{j}})$ is the matrix product $\beta\mathbf{D}_{\underline{j}}$ without sum on the index $\underline{j}$. Also, note that $h/\mathbf{u}$ has been included into the definition for $\mathbf{D}_{\underline{j}}$. Additional details can be found in [16] and the forms of β and $\mathbf{D}_{\underline{j}}$ are given in Appendix B.

In summary, an additional term becomes added to each of the pseudo-compressibility equations (2.7)-(2.9) to obtain the companion conservation law system (3.8). These additional "β-terms" may be used to moderate the dispersion error mechanism in the numerical solution, if necessary, and the level of this artificial viscosity is controlled by specifying the values of the multipliers contained in the matrix β.

3.2 Taylor Weak Statement

The weak statement for (3.8) is

$$\int_{\Omega} w_i \left[\frac{\partial q}{\partial t} + \nabla \cdot (\mathbf{f} - \mathbf{f}^{\nu}) - s - \nabla \cdot (\beta \mathbf{D}) \nabla q \right] d\Omega = 0 \tag{3.9}$$

for all functions w_i, where $(\beta\mathbf{D})$ is the diagonal matrix with components $(\beta\mathbf{D}_{\underline{j}})$ from (3.8), with the $\underline{j}$ subscript suppressed for vector notation, and $\Omega \subset \mathrm{R}^n$ is a domain on which the arbitrary test function w_i is non-vanishing. Utilizing the Green-Gauss Theorem, the symmetric form weak statement associated with (3.9) is

$$\begin{aligned} &\int_{\Omega} w_i \frac{\partial q}{\partial t}\, d\Omega - \int_{\Omega} \nabla w_i \cdot \mathbf{f}\, d\Omega - \int_{\Omega} \nabla w_i \cdot (-\mathbf{f}^{\nu})\, d\Omega \\ &- \int_{\Omega} w_i s\, d\Omega + \int_{\Omega} \nabla w_i \cdot (\beta\mathbf{D}) \nabla q\, d\Omega + \oint_{\partial\Omega} w_i \mathbf{f} \cdot \mathbf{n}\, d\Gamma \\ &+ \oint_{\partial\Omega} w_i(-\mathbf{f}^{\nu}) \cdot \mathbf{n}\, d\Gamma - \oint_{\partial\Omega} w_i((\beta\mathbf{D})\nabla q) \cdot \mathbf{n}\, d\Gamma = 0 \end{aligned} \tag{3.10}$$

for all w_i and where $\mathbf{n}$ is the outwards pointing unit normal vector. Equation (3.10) constitutes the weak statement for the Taylor series modified conservation law system (3.9).

3.3 Boundary Condition Enforcement

The Taylor weak statement (3.10) exposes the important flux vector surface integrals that permit efficient enforcement of key boundary conditions [16]. On any boundary segment, the kinematic flux vector surface integral is

$$\oint_{\partial\Omega} w \mathbf{f} \cdot \mathbf{n}\, d\Gamma \tag{3.11}$$

which becomes

$$\oint_{\partial\Omega} w(u_j n_j)\, d\Gamma \tag{3.12}$$

in the continuity equation,

$$\oint_{\partial\Omega} w(u_j n_j) u_i\, d\Gamma \tag{3.13}$$

in the i-th momentum equation, and

$$\oint_{\partial\Omega} w(u_j n_j)\Theta\, d\Gamma \tag{3.14}$$

in the energy equation. In (3.13), the pressure term is not included, because it is omitted from the integration by parts of (3.9). Thus, no surface integral involving the pressure p is created in the formation of (3.11).

The surface integral

$$-\oint_{\partial\Omega} w(\mathbf{f}^{\nu} + (\beta\mathbf{D})\nabla q) \cdot \mathbf{n}\, d\Gamma \tag{3.15}$$

becomes

$$-\oint_{\partial\Omega} w\left((\beta\mathbf{D})\frac{\partial p}{\partial n}\right) d\Gamma \tag{3.16}$$

in the continuity equation,

$$-\oint_{\partial\Omega} w\left(\frac{\partial u_i}{\partial n} + (\beta\mathbf{D})\frac{\partial u_i}{\partial n}\right) d\Gamma \tag{3.17}$$

in the i-th momentum equation, and

$$-\oint_{\partial\Omega} w\left(\frac{\partial \Theta}{\partial n} + (\beta\mathbf{D})\frac{\partial \Theta}{\partial n}\right) d\Gamma \tag{3.18}$$

in the energy equation. Boundary conditions are thus enforced through the inclusion or deletion of the surface integrals (3.12)-(3.14) and (3.16)-(3.18).

Wall Boundaries

Along solid wall boundaries, an inviscid flow specification implies that u_i is tangent to the wall, while for a viscous flow problem, the no-slip condition, $u_i = 0$, must be enforced. So, for an inviscid flow, surface integrals (3.12)-(3.14) are set equal to zero (deleted) since $u_j n_j = 0$, and $\mathbf{f}^\nu$ is zero identically since $Re \to \infty$. In viscous flows, all the surface integrals involving u_i default to zero, and the enforcement of the Dirichlet constraint $u_i = 0$ is a routine matter, see Baker [20]. Equation (3.18) provides the means to enforce a non-homogeneous Neumann constraint on the potential temperature Θ and is otherwise deleted.

Inlets and Outlets

Acceptable boundary conditions for inlets and outlets in an inviscid flow were given in Chapter 2, and as previously mentioned, Dirichlet data are enforced as described in [20]. For a viscous flow specification where $\mathbf{f}^\nu \neq 0$, if the velocity and temperature are not specified, their normal derivatives, $\frac{\partial u_i}{\partial n}$ and $\frac{\partial \Theta}{\partial n}$, are made to vanish by deletion of surface integrals (3.17) and (3.18), respectively, which is an appropriate requirement [20]. It would be erroneous, of course, to also delete any of the surface integrals (3.12)-(3.14) as this would imply that $u_i n_i = 0$, which is not the case. The problem is handled by not utilizing the Green-Gauss Theorem in the formation of the appropriate integrals of $\mathbf{f}$ along the inlet and/or outlet boundary under consideration, so as not to produce these terms requiring evaluation. Finally, for all boundaries (including solid walls) the surface integral terms involving $(\beta \mathbf{D})$ are assumed to vanish identically. This is convenient and acceptable, since these terms are relatively small (and are not even present if artificial dissipation is not being utilized).

3.4 Finite Element Formulation

In formulating the finite element algorithm, the semi-discrete approximation q^h to the exact solution q is constructed on the computational mesh $\Omega^h = \bigcup_e \Omega_e$ as,

$$q(\mathbf{x},t) \simeq q^h(\mathbf{x},t) = \bigcup_e \sum_{i=1}^{I} N_i(\eta_j) Q_i(t) = \bigcup_e \{N_k\}^T \{Q\}_e \tag{3.19}$$

where N_i is an admissible trial space function, I is the number of nodes per element and Q_i is the array of nodal state variables. Further, $\{Q\}_e$ is the column matrix form of the nodal state variable for any particular element, $\{N_k\}$ is the trial function basis set of degree k, and $\bigcup_e$ signifies the union over all elements. The discretization formation utilizes an isoparametric coordinate transformation, where η_j is the local orthonormal coordinate system. Note that the space and time dependence has been separated, with the time dependence remaining with the nodal variables $Q_i(t)$.

For the "Galerkin" criterion, the test function set w_i introduced in (3.9) is set equal to the trial function set, hence by elements in the discretization.

$$w_i = N_i \tag{3.20}$$

Now, inserting (3.19) and (3.20) into the Taylor weak statement (3.10), the finite element semi-discrete statement is

$$S_e \left(\int_{\Omega_e} \{N\}\{N\}^T \{Q\}'_e \, d\Omega - \int_{\Omega_e} \nabla\{N\} \cdot \mathbf{f}_e \, d\Omega + \int_{\Omega_e} \nabla\{N\} \cdot \mathbf{f}^{\nu}_e \, d\Omega \right.$$

$$+\int_{\Omega_e}\{(\beta\mathbf{D})\}_e^T\{N\}\nabla\{N\}\cdot\nabla\{N\}^T\{Q\}_e\,d\Omega-\int_{\Omega_e}\{N\}s_e\,d\Omega$$
$$+\oint_{\partial\Omega_e}\{N\}\mathbf{f}_e\cdot\mathbf{n}\,d\Gamma+\oint_{\partial\Omega_e}\{N\}(-\mathbf{f}_e^\nu)\cdot\mathbf{n}\,d\Gamma$$
$$-\oint_{\partial\Omega_e}\{(\beta\mathbf{D})\}_e^T\{N\}\{N\}\nabla\{N\}^T\{Q\}_e\cdot\mathbf{n}\,d\Gamma\Big)=\{0\} \tag{3.21}$$

In (3.21), superscript prime denotes the ordinary time derivative, $\mathbf{f}_e$ and $\mathbf{f}_e^\nu$ are the flux vector component semi-discretizations on the finite element Ω_e and S_e symbolizes the assembly operation performed over all elements of Ω^h. Note that the elements of the diagonal matrix $(\beta\mathbf{D})$ are also interpolated using the basis function $\{N\}$, since $\mathbf{D}$ contains terms involving the state variables.

The scalar components of the inviscid flux vector $(f_j)_e$ are

$$(f_j)_e=\left\{\begin{array}{c}\{N\}^T\{UJ\}_e\\ \{UJ\}_e^T\{N\}\{N\}^T\{UI\}_e+\{N\}^T\{P\}_e\delta_{IJ}\\ \{UJ\}_e^T\{N\}\{N\}^T\{\Theta\}_e\end{array}\right\} \tag{3.22}$$

for $1\le I,J\le n$, where UJ and UI denote the nodal velocity scalar components. For the viscous flux vector $(f_j^\nu)_e$ the corresponding definition is

$$(f_j^\nu)_e=\left\{\begin{array}{c}0\\ \frac{1}{Re}\left(\frac{\partial\{N\}^T}{\partial x_j}\{UI\}_e\right)\\ \frac{1}{RePr}\left(\frac{\partial\{N\}^T}{\partial x_j}\{\Theta\}_e\right)\end{array}\right\} \tag{3.23}$$

and the source array (with $s_\theta=0$) is

$$(s)_e=\left\{\begin{array}{c}0\\ -\frac{Gr}{Re^2}\{N\}^T\{\Theta\}_e\hat{g}\\ 0\end{array}\right\} \tag{3.24}$$

Inserting (3.22),(3.23) and (3.24) into (3.21) in the appropriate order, and deleting the surface integrals to enforce the previously presented boundary conditions, produces the weak statement semi-discretization. Suppressing the subscript e notation for clarity, the terminal form is

$$S_e\left(\frac{1}{\varepsilon}\int_{\Omega_e}\{N\}\{N\}^T\{P\}'\,d\Omega-\int_{\Omega_e}\frac{\partial\{N\}}{\partial x_j}\{N\}^T\{UJ\}\,d\Omega\right.$$
$$\left.+\int_{\Omega_e}\{(\beta_p\mathbf{D}_{\underline{j}})\}^T\{N\}\frac{\partial\{N\}}{\partial x_j}\frac{\partial\{N\}^T}{\partial x_j}\{P\}\,d\Omega\right)=\{0\} \tag{3.25}$$

$$S_e\left(\int_{\Omega_e}\{N\}\{N\}^T\{UI\}'\,d\Omega-\int_{\Omega_e}\{UJ\}^T\{N\}\frac{\partial\{N\}}{\partial x_j}\{N\}^T\{UI\}\,d\Omega\right.$$
$$+\int_{\Omega_e}\{N\}\frac{\partial\{N\}^T}{\partial x_j}\{P\}\delta_{IJ}\,d\Omega+\frac{1}{Re}\int_{\Omega_e}\frac{\partial\{N\}}{\partial x_j}\frac{\partial\{N\}^T}{\partial x_j}\{UI\}\,d\Omega$$
$$-\frac{Gr}{Re^2}\int_{\Omega_e}\{N\}\{N\}^T\{\Theta\}\delta_{I2}\,d\Omega$$
$$\left.+\int_{\Omega_e}\{(\beta_{u_i}\mathbf{D}_{\underline{j}})\}^T\{N\}\frac{\partial\{N\}}{\partial x_j}\frac{\partial\{N\}^T}{\partial x_j}\{UI\}\,d\Omega\right)=\{0\} \tag{3.26}$$

$$S_e\left(\int_{\Omega_e}\{N\}\{N\}^T\{\Theta\}'\,d\Omega-\int_{\Omega_e}\{UJ\}^T\{N\}\frac{\partial\{N\}}{x_j}\{N\}^T\{\Theta\}\,d\Omega\right.$$
$$+\frac{1}{RePr}\int_{\Omega_e}\frac{\partial\{N\}}{\partial x_j}\frac{\partial\{N\}^T}{\partial x_j}\{\Theta\}\,d\Omega$$
$$\left.+\int_{\Omega_e}\{(\beta_\Theta\mathbf{D}_{\underline{j}})\}^T\{N\}\frac{\partial\{N\}}{\partial x_j}\frac{\partial\{N\}^T}{\partial x_j}\{\Theta\}\,d\Omega\right)=\{0\} \tag{3.27}$$

Again, superscript prime denotes the temporal ordinary derivative, and the β_p, β_{u_i}, and β_Θ terms refer to the specific user-defined β-values in the continuity, momentum, and energy equations, respectively. The nodal state variable matrices $\{P\}$, $\{UI\}$, and $\{\Theta\}$ are independent of x_i, and can be extracted from the integrands. Finally, note that the buoyancy term is detailed in (3.26) for $n = 2$.

The assembly of equations (3.25)-(3.27) can be compactly written as

$$[M]\{P\}'-\varepsilon[M2J0]\{UJ\}+\varepsilon\{(\beta_p\mathbf{D}_{\underline{j}})\}^T[M30JJ]\{P\}=\{0\} \tag{3.28}$$

$$[M]\{UI\}'-\{UJ\}^T[M30J0]\{UI\}+[M20I]\{P\}+\frac{1}{Re}[M2JJ]\{UI\}$$
$$-\frac{Gr}{Re^2}[M200]\{\Theta\}\delta_{I2}+\{(\beta_{u_i}\mathbf{D}_{\underline{j}})\}^T[M30JJ]\{UI\}=\{0\} \tag{3.29}$$

$$[M]\{\Theta\}' - \{UJ\}^T[M30J0]\{\Theta\} + \frac{1}{RePr}[M2JJ]\{\Theta\}$$

$$+\{(\beta_\Theta \mathbf{D}_j)\}^T[M30JJ]\{\Theta\} = \{0\} \tag{3.30}$$

The definitions of the various matrices $[M2\ldots]$ and $[M3\ldots]$, resulting from the integrals of products of $\{N\}$ follow from inspection, and $[M]$ represents the mass matrix, see [20]. Also, notice that the continuity equation has been multiplied through by the pseudo-compressibility ε to isolate $\{P\}'$.

As mentioned previously, an isoparametric coordinate transformation is naturally introduced into the algorithm integrals by a change of variables from cartesian coordinates $\mathbf{x}$ to the finite element intrinsic coordinate system η_k [20]. The local coordinate transformation on the generic element domain Ω_e is

$$(\mathbf{x}(\eta_k))_e = \{N(\eta_k)\}^T\{\mathbf{X}\}_e \tag{3.31}$$

where $\{N\}$ is the trial space basis set and $\{\mathbf{X}\}_e$ contains the nodal coordinate scalar components $(x_j)_e$. The relation between partial derivatives with respect to $\mathbf{x}$ and η_k is

$$\frac{\partial}{\partial \eta_k} = \left[\frac{\partial x_j}{\partial \eta_k}\right]_e \frac{\partial}{\partial x_j} \tag{3.32}$$

The matrix $\left[\frac{\partial x_j}{\partial \eta_k}\right]_e$ is evaluated using (3.31). Then, (3.32) is transformed into

$$\frac{\partial}{\partial x_j} = \frac{1}{det\, J_e} J_e^{-1} \frac{\partial}{\partial \eta_k} \equiv \frac{1}{det\, J_e}[e_{jk}]_e \frac{\partial}{\partial \eta_k} \tag{3.33}$$

where

$$J_e^{-1} = [e_{jk}]_e \equiv \begin{bmatrix} \frac{\partial y}{\partial \eta_2} & , & -\frac{\partial y}{\partial \eta_1} \\ -\frac{\partial x}{\partial \eta_2} & , & \frac{\partial x}{\partial \eta_1} \end{bmatrix}_e \tag{3.34}$$

In implementing the coordinate transformation, $[e_{jk}]_e$ is inserted into each integral in (3.25)-(3.27) prior to evaluation on Ω_e. For example, in (3.25)

$$\int_{\Omega_e} \frac{\partial\{N\}}{\partial x_j}\{N\}^T\{UJ\}_e\,d\Omega = \int_{\Omega_e}[e_{jk}]_e\frac{\partial\{N\}}{\partial\eta_k}\{N\}^T\,d\Omega\{UJ\}_e$$
$$\simeq EJK_e\int_{\Omega_e}\frac{\partial\{N\}}{\partial\eta_k}\{N\}^T\,d\Omega\{UJ\}_e \equiv EJK_e[M2KO]\{UJ\}_e \qquad (3.35)$$

where the data array EJK_e, $1 \leq (J,K) \leq n$, contains the metric data evaluated at suitable element quadrature points and η_k is the local orthonormal coordinate system spanning any finite element domain Ω_e. Thus, the matrices $[M\ldots]$ in (3.28)-(3.30) are each made element independent. Finally, note in (3.35) that $\{UJ\}_e$ is extracted from the integral since it is independent of η_k.

3.5 Implicit Time Integration

Equations (3.28)-(3.30) express the finite element semi-discretization which yields, upon assembly on $\Omega^h = \bigcup_e \Omega_e$, the following global ordinary differential equation (ODE) system:

$$[M]\{Q\}' + \sum_{K=1}^{n}\{RK(Q)\} = \{0\} \qquad (3.36)$$

where the residual, $\{R\} = \sum\{RK\}$, is the sum of n contributions for a n-dimensional problem domain. To integrate (3.36), the generic "θ" implicit method is appropriate [20] and yields

$$[M]\{Q_{n+1} - Q_n\} + \Delta t\left[\theta\sum_{K=1}^{n}\{RK(Q)\}_{n+1} + (1-\theta)\sum_{K=1}^{n}\{RK(Q)\}_n\right] = \{0\} \quad (3.37)$$

System (3.37) defines a non-linear function in $\{Q\}$, solution of which is formulated via Newton's algorithm,

$$\left[M + \theta\,\Delta\,t \sum_{K=1}^{n} \left(\frac{\partial\{RK(Q)\}}{\partial\{Q\}}\right)\right]^{p} \{Q_{n+1}^{p+1} - Q_{n+1}^{p}\} = -\{F\}^{p} \tag{3.38}$$

$$\{F\}^{p} = [M]\{Q_{n+1}^{p} - Q_{n}\} + \Delta t \left(\theta \sum_{K=1}^{n} \{RK(Q)\}_{n+1}^{p} + (1-\theta) \sum_{K=1}^{n} \{RK(Q)\}_{n}\right) \tag{3.39}$$

where p denotes the iteration index.

The left-hand matrix of (3.38) is not actually formed in practice because, due to the large sparse nature, only relatively coarse grids would be practical. Instead, a tensor matrix product approximation factorization is utilized [21] and for the 2-D case, (3.38) is replaced by

$$\begin{aligned} &\left[M1 + \theta\,\Delta\,t \left(\frac{\partial\{R1\}}{\partial\{Q\}}\right)^{p}\right] \otimes \left[M2 + \theta\,\Delta\,t \left(\frac{\partial\{R2\}}{\partial\{Q\}}\right)^{p}\right] \\ &\{Q_{n+1}^{p+1} - Q_{n+1}^{p}\} = -\{F\}^{p} \end{aligned} \tag{3.40}$$

where $\otimes$ denotes the matrix tensor product, and $\{F\}^{p}$ remains the right side residual defined in (3.39). Equation (3.40) is then approximately solved via the sequence

$$\begin{aligned} \left[M1 + \theta\,\Delta\,t \left(\frac{\partial\{R1\}}{\partial\{Q\}}\right)^{p}\right] \{Z\} &= -\{F\}^{p} \\ \left[M2 + \theta\,\Delta\,t \left(\frac{\partial\{R1\}}{\partial\{Q\}}\right)^{p}\right] \{Q_{n+1}^{p+1} - Q_{n+1}^{p}\} &\equiv \{Z\} \end{aligned} \tag{3.41}$$

where $\{Z\}$ is an intermediate solution array. The matrices in (3.41) are block tridiagonal and are formed over one-dimensional coordinate curves during an "η_1-sweep" followed by an "η_2-sweep."

Chapter 4

Discussion and Results

The finite element CFD algorithm described in the preceding chapter was implemented through modification of an existing 2-D compressible flow Navier-Stokes code, Iannelli and Baker [16]. The code utilizes quadrilateral elements (four nodes per element) and bilinear trial functions $\{N_1\}$. The basic requirement of the pseudo-compressible finite element solution algorithm is prediction of separated (and non-separated) viscous, incompressible, laminar flows in two-dimensional geometries. The following three benchmark test cases are considered:

1. Developing flow in a rectangular cross-section channel.

2. Flow over a backward facing step (step wall diffuser).

3. Buoyant natural convection in a thermal cavity.

Before discussing each case individually, some comments regarding boundary conditions and the use of artificial dissipation are in order.

An unexpected and significant difficulty is encountered at the corner singularities of the solution domain for each of the three test cases. At corner nodes, the pressure p seems to be uncoupled from adjacent nodal pressures, as evidenced by the observation that as the pressure solution evolves, the corner pressures remained essentially unchanged from their initial condition. This phenomena causes local solution errors near the corners that are unacceptable. It was confirmed, however, that the residuals (the $\{F\}$ terms of (3.38)) at the corner nodes were being correctly computed, which suggested that the problem was in the nodal coupling contained in the pressure Jacobians $\frac{\partial\{RK\}}{\partial\{P\}}$. The inclusion of artificial dissipation into the continuity (pressure) equation was observed to moderate the problem by coupling the corner node pressures with the surrounding nodal pressures. This coupling is the result of the $\beta_p \nabla^2 p$ term introduced into the continuity equation discrete approximation Jacobians when artificial dissipation is included (see Appendix B). Thus, some level of smoothing in the continuity equation was required in every test case to avoid pressure solution stagnation at corners, regardless of whether or not the pressure solution was sufficiently smooth in the remainder of the domain. Generally, a relatively high value of β_p was employed until the solution began to approach steady state ($\frac{\partial p}{\partial t}$ approaches zero), at which time β_p was reduced to allow the divergence of the velocity field, which is heavily influenced by the added smoothing terms, to better approach computational zero. In the INS3D (Incompressible Navier-Stokes, 3-Dimensional) finite difference code [13], an exponentially decaying smoothing term is employed to produce the same effect.

Another way to avoid the pressure decoupling problem at corner singularities would be to enforce a Dirichlet constraint on the pressure at the inlets and outlets. For the two ducted flows considered, developing channel flow and a step-wall diffuser, the pressure level was imposed at the outlet, which eliminated any potential outlet corner difficulties. Imposing the pressure at the both the inlet and the outlet, however, poses a problem in that the pressure drop through the domain of interest is generally unknown. One exception is the case of a fully developed, laminar rectangular channel flow, in which the pressure drop through a given channel length can be analytically determined as a function of Reynolds number. Such a case was executed, with the pressure at the inlet and outlet being correctly specified. The imposed initial flow field was parabolic, but its' magnitude was incorrect for the given Reynolds number and pressures specified. It was observed that the velocity and pressure fields then converged to the correct solution with no artificial dissipation being required.

4.1 Developing Channel Flow

The first test case considered is developing flow in a two-dimensional rectangular channel of unit width and normalized length of 14. Since the problem is symmetric about the duct centerline, a half-duct discretization was employed. The prescribed inlet condition is uniform flow with the exception of the corner node, where the velocity is set to zero, see Figure A.1 in Appendix A. The pressure at the outlet is enforced as Dirichlet data and set to zero ($p_{ref} = 0$). The no-slip condition is imposed along the wall boundary and the transverse velocity v is zero at the duct centerline. The computational grid had 20 elements in the cross-stream direction and 40 elements in the streamwise direction, with local grid refinement along the wall boundary and near the inlet, see Figure A.2 (wherein the y-axis has been magnified by a factor of 10).

The initial condition consists of an uniform velocity field identical to that at the inlet with a linear pressure drop from the inlet to the outlet. Solutions were obtained for Reynolds numbers of 50, 100, and 200, with the Reynolds number based on the total duct width ($x_{ref} = 1.0$) and inlet flow velocity ($u_{ref} = 1.0$). The converged solutions for $Re = 50$ and $Re = 100$ were used as the initial conditions for the $Re = 100$ and $Re = 200$ solutions, respectively. Equation (2.19) places the following restrictions on the pseudo-compressibility ε:

$$\begin{array}{ll} Re = 50 & \varepsilon \gg 3.49 \\ Re = 100 & \varepsilon \gg 1.43 \\ Re = 200 & \varepsilon \gg 0.64 \end{array}$$

Hence, a value of $\varepsilon = 10$ was chosen for all solutions.

Equation (2.21) predicts the minimum time for convergence to be 9.3 seconds. It was observed that the pseudo pressure waves had distributed the pressure within this time limit, but the total time required for full convergence was approximately 30 seconds in each case. During the "extra" 20 seconds, the velocity field continued to adjust to the distributed pressure field until the converged solution was achieved. The solution was considered to be converged when the maximum change in any variable $|\Delta\{Q\}|_{max}$ at a time step and the maximum residual level $|\sum_{K=1}^{n}\{RK(Q)\}|_{max}$ (see (3.36)) were less than 10^{-4}. Each case took approximately 800 time steps to reach steady state, with Δt averaging 0.04 seconds.

An important consideration in developing duct flow is accurate prediction of entrance length. Schlichting [22] derives the analytical formula for prediction of entrance length for a channel of unit width as

$$L_e = 0.04Re \tag{4.1}$$

For the three cases considered, $Re = 50$, 100, and 200, the computed velocity profile at the entrance length predicted by (4.1) is within 2% of being fully developed.

Shown in Figures A.3-A.8 are the u and v velocity profiles at select x locations near the channel entrance. Also shown are the terminal u velocity profiles at the channel outlet, which are in good agreement with the exact (parabolic) solution. Note that a small axial velocity overshoot is evident near the channel wall at the inlet as required for this problem specification[20].

As previously discussed, some level of artificial dissipation was required to diffuse pressure solution errors resulting from inlet corner node decoupling. In this particular problem specification, the inlet corner is a critical area of the domain in that a large pressure buildup occurs there, see Figure A.9, which produces the cited axial velocity overshoot. A relatively large value of artificial pressure smoothing was required to control local pressure oscillations near the inlet, with $\beta_p = 0.15$, 0.3, and 0.5 for $Re = 50$, 100, and 200 respectively. Even with this added dissipation, some pressure fluctuations are evident in the pressure contours, cf. Figure A.9.

The local pressure buildup and velocity overshoot near the inlet are a consequence of the imposed boundary condition at the inlet, i.e. an uniform axial velocity profile and $v = 0$. In a laboratory, achieving an uniform velocity profile at the inlet of a channel would be practically impossible, since the velocity and pressure would anticipate the presence of the channel entrance and precursory "adjustments" in the pressure and velocity fields would occur upstream of the actual inlet. In an effort to simulate these "adjustments", while still maintaining a relatively uniform entrance velocity, an inviscid flow section was added preceding the viscous flow channel entrance. This added portion was the same width as the original channel and had a length of 3 units. An uniform velocity profile was imposed at the entrance, but the tangential flow boundary condition was imposed along the "inviscid" wall boundary of this inviscid region.

As can be seen in Figure A.10, a slight pressure buildup again occurred at the corner of the viscous channel. However, the magnitude of this buildup is much smaller and less localized than the previous case. Also, the local velocity overshoot was eliminated and the predicted entrance length was in agreement to within 1% of that predicted by (4.1). The singularity point created at the intersection of the inviscid and viscous regions of the channel caused oscillations in pressure and velocity to propagate upstream in the negative x-direction. To localize these oscillations, artificial smoothing was introduced in the pressure and momentum equations by setting $\beta_p = 0.2$ and $\beta_{u_i} = 0.1$. The velocity vector distributions near the channel inlet are shown in Figures A.11 and A.12 for the two cases, with the viscous segment of the channel beginning at $x = 0$ in A.12. Notice that some axial velocity oscillations dominate the inviscid section of channel flow along the wall boundary. Due to these oscillations, the expected velocity deacceleration along the wall just prior to the viscous channel entrance does not occur, but instead a local acceleration is observed.

4.2 Step Wall Diffuser

The second test case is two-dimensional flow over a backward facing step. The problem geometry is modeled after an experimental and numerical investigation reported by Armaly, et al. [23]. Shown in Figure A.13 is a schematic of the diffuser and in Figure A.14 the experimental data of [23] is summarized. The benchmark requirement is prediction of the reattachment length of the primary recirculation zone located behind the step, which is a direct function of Reynolds number for $Re \leq 400$, above which the experiment verifies transition to a fully three-dimensional flow.

The total length of the channel must be sufficient to allow the flow to attain a fully developed velocity profile at outflow. Since convergence time is linearly dependent on channel length (see equation (2.21)), as short a channel as possible is desirable for a given range of Reynolds numbers. Due to the cited transition to 3-D flow, the range of $10 \leq Re \leq 400$ was considered for verification purposes. An overall channel length of 40 step heights was sufficient to achieve a fully developed profile at outflow, with the height of the step being 0.49 units. The length of the inlet channel before the step is equal to 4 step heights. The Reynolds number employed in [23] is based on u_{ref} equal to two-thirds of the maximum inlet velocity and x_{ref} equal to the hydraulic diameter of the inlet channel ($x_{ref} = 2(.52) = 1.04$)

rather than the step height. Herein, u_{ref} is specified as the maximum inlet velocity ($u_{ref} = 1.0$) and x_{ref} is set to unity. Hence, the transition to 3-D flow occurs at a Reynolds number of 577 (corresponding to Armaly's Reynolds number of 400).

The prescribed boundary conditions are similar to that of the previous test case, now with a parabolic inlet velocity profile and a fixed outlet pressure enforced as Dirichlet data. The no-slip condition is enforced along walls. The computational grid had 20 elements in the cross-stream direction and 37 elements in the streamwise direction, with local refinement near the step, see Figure A.15. The initial condition consists of a parabolic velocity profile throughout the domain (with $u_{max} = 1.0$ in the inlet section and $u_{max} = 0.5$ downstream of the step) and a linear pressure gradient spanning the length of the channel (with $p_{ref} = p_{exit} = 0$). After the solution at $Re = 10$ was obtained, the sequence of solutions at $Re = 100$, 200, 288, 433, and 577 was achieved by increasing the Reynolds number with the previous solution serving as the initial condition.

The question of the value of the pseudo-compressibility constant for this problem specification now arises. In [13], a 2-D numerical investigation utilizing the pseudo-compressible finite difference code INS3D was conducted for the step wall diffuser, but the value of ε was not stated. Since a value of $\varepsilon = 10$ had been successfully employed in the previous test problem, this value was also initially chosen for the present case. At outflow, $u_{max} \approx 0.5$, so the average speed in the streamwise direction was approximated as $u_{avg} = \frac{2}{3}(.5) = .33$. Thus, equation (2.21) predicts the minimum time to achieve steady-state as approximately 13 seconds.

Another question that arises is what level, if any, of artificial diffusion to employ to control numerical dispersion error, especially in the pressure field. Based on the previous test case, it was expected that some level of artificial smoothing would be needed in the pseudo-compressible continuity equation to control pressure oscillations, especially near the corners of the inlet. Hence, a value of $\beta_p = 0.1$ was initially chosen. It was discovered, however, that this level of dissipation significantly hindered algorithm convergence rate as the solution proceeded, with Δt averaging less than .01 second. Decreasing β_p improved the convergence rate, although pressure oscillations began to appear in the inlet channel and in the vicinity of the step. However, these oscillations did not seem to affect stability of the u and v velocity solutions.

Converged solutions were obtained for $Re = 100$, 200, and 288 using $\varepsilon = 10$. Solutions were considered to be converged when the maximum change of variables at a time step and the maximum residuals (the $\sum_{K=1}^{n}\{RK(Q)\}$ terms of (3.36)) had decreased to the order of 10^{-4}. As previously stated, β_p was gradually lowered as the solution proceeded from an initial value of $\beta_p = 0.1$ to $\beta_p = 0.0$, with Δt generally being maintained in the range of 0.05–0.3 seconds and with the greatest Δt occurring at $\beta_p = 0$. A terminal β_p value of zero also promoted enforcement of conservation of mass, since a non-zero β_p introduces smoothing terms into the continuity (pressure) equation, yielding $\nabla^h \cdot \mathbf{u}^h$ – smoothing terms ≈ 0 rather than $\nabla^h \cdot \mathbf{u}^h \approx 0$. As observed in the previous test case, the pressure waves distributed the pressure within the minimum convergence time predicted by equation (2.21). Again, however, "extra" time was needed for the accelerated velocity field to "adjust" to the distributed pressure. Hence, solutions converged within approximately 37 seconds for $Re \leq 288$, which is about three times as long as the predicted minimum convergence rate. This is comparable with what occurred in the developing channel flow case.

As previously noted, pressure oscillations occurred near the inlet when β_p was decreased. These oscillations were principally in the cross-stream direction, with relatively minor oscillations occurring in the streamwise direction. After convergence was achieved, it was found that β_p could be increased to somewhat "smooth out" these oscillations. As an example, shown in Figure A.16 are pressure contours for the converged $Re = 288$ solution with $\beta_p = 0$. Depicted in Figures A.17 and A.18 are the pressure contours with β_p having been increased to 0.05 and 0.1, respectively. Notice that the oscillations have been extensively reduced, but not completely eliminated. If β_p is increased beyond approximately 0.1, the introduced smoothing terms begin to adversely affect the divergence-freeness of the velocity field, and the momentum equation residuals begin to increase. For example, at $\beta_p = 0.0$, the steady flow residuals of the momentum equations are on the order of 10^{-5}. Increasing β_p to 0.2 serves to decrease the pressure oscillations, but the residuals of the momentum equations are increased by an order of magnitude to 10^{-4}. Further increases in β_p continue to destabilize the momentum equations, due to the adverse affect on the velocity field's computationally-approximate divergence-free state corresponding to $\beta_p = 0.0$. Additionally, excessively increasing β_p causes automatic enforcement of the Neumann constraint $\frac{\partial p}{\partial n}$ along boundary segments (see Section 3.3), which is not everywhere a proper specification.

It is interesting to note that the severe pressure oscillations occurring at the inlet (with $\beta_p = 0$) do not adversely affect convergence of the u and v solutions. This is due to the fact that only the gradient of the pressure field is "seen" by the momentum equations. Since the Galerkin finite element method is fundamentally equivalent to a central differencing type of finite difference formula in the way spatial gradients are approximated, the odd-even nodal points are decoupled. Therefore, an oscillatory pseudo-pressure solution is certainly expected and allowed. Inclusion of artificial dissipation in the pressure equation, however, causes adjacent nodes to be coupled because of the introduced $\nabla^2 p$ term. Hence, pressure oscillations are detected and consequently "smoothed out."

Using the previous solution at $Re = 288$ as the initial condition, the Reynolds number was increased to 433. As the solution progressed, it was observed that the velocity solution in the recirculation zone behind the step became unstable. A proper recirculation zone did not form, and the solution reached a "stagnation" point where the velocity field in the vicinity of the recirculation zone would "oscillate" between incorrect solutions and convergence to steady state was not attained. These oscillations were not dispersion error "$2\triangle x$ waves", but seemed to be random in nature. Various trials with different β_p value were performed, with similar unsuccessful results. At this point, ε was decreased from 10 to a value of 1. This new value of the pseudo-compressibility resulted in a converged solution after approximately 37 seconds, which is comparable to the previous Reynolds number solutions, even though equation (2.21) predicts an increase in minimum convergence time.

With ε still equal to 1.0, a similar problem occurred when the Reynolds number was increased to 577. The velocity field in the vicinity of the recirculation zone was again "unstable" and non-converging. Decreasing ε to 0.1 resolved the problem. However, convergence time was increased to approximately 245 seconds, which is a substantial increase from previous solutions, but is only 33% greater than the minimum convergence time of 184 seconds predicted by (2.21). The reasons for the unstable recirculation zone at lower ε values are not clear, and further investigation is needed. Similar instabilities resulted when a solution at $Re = 721$ was attempted with ε remaining at 0.1.

Figures A.19-A.23 show the converged velocity fields in the step vicinity for the various Reynolds number solutions. The data in the channel downstream from the recirculation has been truncated to improve resolution. Figures A.24-A.28 present magnified views of the recirculation flowfields. The predicted primary reattachment length, $X1$, is in good agreement with the experimental data of [23] up to $Re = 300$, see Figure A.29 (where Re is defined as in [23]). Note that "S" refers to the step height, which is equal to 0.49 units. Also shown in Figure A.29 are results from a finite element code written for the streamfunction–vorticity ($\Omega - \Psi$) reformulation of the incompressible Navier-Stokes equations [24], which mathematically guarantees a divergence-free velocity field. Note that the predictions from the $\Omega - \Psi$ and pseudo-compressibility codes are essentially identical, and that both codes slightly underpredict the reattachment length at $Re = 400$, above which the flow becomes three-dimensional.

4.3 Thermal Cavity

The final benchmark test case considered is natural convection in a thermal cavity. Since there is no freestream onset velocity, the reference velocity is redefined in terms of the thermodynamic properties of the flow field as,

$$u_{ref} \equiv \frac{\alpha_{ref}}{x_{ref}} \tag{4.2}$$

where $\alpha_{ref} = \frac{k_{ref}}{\rho_0 c_{v(ref)}}$ is the thermal diffusivity. Using (4.2), the pseudo-compressibility conservation law system (2.7)-(2.9) takes the following non-dimensional form:

$$L(p) = \frac{1}{\varepsilon}\frac{\partial p}{\partial t} + \nabla \cdot \mathbf{u} = 0 \tag{4.3}$$

$$L(\mathbf{u}) = \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - Pr\nabla^2\mathbf{u} + \nabla p + PrRa\Theta\hat{g} = 0 \tag{4.4}$$

$$L(\Theta) = \frac{\partial \Theta}{\partial t} + (\mathbf{u} \cdot \nabla)\Theta - \nabla \cdot \nabla\Theta = 0 \tag{4.5}$$

The new dimensionless parameter introduced in (4.6) is

$$\text{Rayleigh Number :} \qquad Ra = \frac{\rho_0^2 c_{v(ref)} g x_{ref}^3 \mathcal{B}(T_{max} - T_{ref})}{\mu_{ref} k_{ref}} \tag{4.6}$$

The problem being considered is that of the two-dimensional flow of a Boussinesq fluid of Prandtl number 0.71 in an upright square cavity of side $x_{ref} = 1.0$. Both velocity components are zero on all boundaries. The horizontal walls are insulated ($\nabla T \cdot \mathbf{n} = 0$), and the left and right vertical walls are held at temperatures T_{max} and

T_{ref}, respectively (see Figure A.30). Unlike the previous two test cases, pressure is not fixed on any domain boundaries. However, since pressure is arbitrary to within a constant, the pressure was anchored at zero at the center of the right vertical wall ($p_{ref} = 0$). The initial conditions consisted of setting u, v, and p to zero everywhere on the domain with Θ varying linearly between the hot and cold walls.

Restrictions on the value of the pseudo-compressibility ε for this problem specification were unknown. An ε of 10 was initially chosen since this value had been used in the previous two test cases. Utilizing an uniform 16×16 mesh, a solution for $Ra = 1000$ was obtained by progressively increasing the buoyancy parameter by intervals of 200, using the steady-state solution at the previous Rayleigh number as the initial guess. Due to the previously discussed pressure decoupling at the corner nodes, it was expected that some level of artificial dissipation ($\beta_p > 0$) would be needed in the pressure equation to control localized pressure oscillations. This was indeed the case, and for $Ra = 1000$, $\beta_p = 0.01$ produced a smooth pressure distribution with the final residuals (the $\sum_{k=1}^{n}\{RK(Q)\}$ terms of equation (3.36)) of all four conservation law equations converging to less than 10^{-4}.

In Figures A.31-A.33, contour plots of u, v, and temperature Θ are presented, and in Figure A.34 the velocity field is depicted for $Ra = 1000$. Shown in Figure A.35 are pressure contour and perspective solution surface plots. The general shapes of the u, v, and Θ contours agree will with those presented in the benchmark numerical study de Vahl Davis [25], which utilized the stream function – vorticity reformulation. Summarized below is a direct velocity comparison for $Ra = 1000$, where u_{max} refers to the maximum horizontal velocity on the vertical mid-plane of the cavity and v_{max} refers to the maximum vertical velocity on the horizontal mid-plane of the cavity.

Streamfunction − Vorticity	Pseudo − compressibility
Mesh : 40 × 40	Mesh : 16 × 16
$u_{max} = 3.61$ at $y = .81$	$u_{max} = 3.67$ at $y = .81$
$v_{max} = 3.68$ at $x = .18$	$v_{max} = 3.72$ at $x = .18$

Note that the origin ($x = 0$, $y = 0$) is defined to be at the lower left corner of the cavity.

Shown in Figure A.36 are the pressure contour and perspective solution surface plots for $Ra = 1000$ with $\beta_p = 0$. The absence of artificial smoothing causes local pressure oscillations in the corners and the residuals of all four conservation law equations increase by an order of magnitude to about 10^{-3}. The pressure decoupling at the corners, the cause of these oscillations, proved to be a very detrimental aspect in efforts to obtain solutions for larger Rayleigh number.

Still utilizing $\varepsilon = 10$, larger Rayleigh number solutions were obtained by increasing Ra by intervals of 500. As Ra was increased, it was determined that an increasing value of β_p was also required to control pressure oscillations. Increasing the value of β_p, however, adversely affected the divergence-freeness of the velocity field prediction and it became increasingly difficult to obtain converged solutions. For example, at $Ra = 4500$, $\beta_p = 0.1$ was required to obtain a smooth pressure solution. The corresponding residuals were on the order of 10^{-1} for the pressure equation and 10^{-3} for the momentum and energy equations. For the same Rayleigh number, as β_p was decreased to zero, localized pressure oscillations appeared in the corners, similar to those depicted in Figure A.36, with the residuals staying at about the same level. For β_p values greater than 0.1, convergence deteriorated, and at $\beta_p = 1.0$ residuals only on the order of 10^{-2} were obtainable.

At this point, it was decided to evaluate a different value of ε. Working at a Rayleigh number of 5000, an ε value of 1.0 was defined in combination with various β_p values with divergent results. Then ε was increased to 50, with somewhat more successful consequences. At $\varepsilon = 50$, $Ra = 5000$, and $\beta_p = 0$ a solution was obtained with maximum residuals being on the order of 10^{-2} for the pressure equation and 10^{-4} for the momentum and energy equation. Shown in Figures A.37-A.41 are the u-velocity contours, v-velocity contours, Θ-contours, velocity vector field, and pressure solution, respectively. Notice in Figure A.41 that substantial pressure oscillations are present, especially near the corners. It is interesting that these oscillations occur without grossly affecting the convergence of the momentum equations. As discussed in the previous section, this is possibly due to the odd-even nodal decoupling of the discrete pressure imposition. Increasing β_p to a value of 0.1 serves to diminish the pressure oscillations, see Figure A.42, but again the divergence-freeness of the velocity field is adversely affected and the residuals of the momentum and energy equations increase to about 10^{-2}.

It was clear at this point that the decoupling of the pressure at the corner nodes presents a major stumbling block for obtaining solutions beyond the relatively low Rayleigh number of 5000. A comprehensive analysis concerning the exact reasons for this decoupling is obviously required, which is beyond the scope of this project.

Chapter 5

Summary

5.1 Conclusions

The goal of this thesis was to implement and critically evaluate a finite element CFD algorithm to solve incompressible flow problems utilizing the method of pseudo-compressibility. The test bed was accomplished through modification of an existing compressible flow Navier-Stokes code. Code and theory operation were verified by consideration of three benchmark test cases. Results from these trial problems compared favorably with documented analytical, numerical, and experimental data, for the appropriate range of flow parameters. However, certain difficulties were encountered and evaluated via sensitivity studies.

The fundamental and most troublesome problem is an apparent decoupling of the pressure solution at geometrical singularities of the solution domain. The pressure solution at these corner nodes does not "follow" the pressure evolution in the surrounding field, but instead remains essentially unchanged from the specified initial condition. Hence, pressure "spikes" become formed in corners which cause localized pressure oscillations. This error mechanism can be moderated by introduction of an artificial dissipation mechanism into the pseudo-pressure equation involving a Laplacian pressure term which serves to couple adjacent nodal pressures. However, this term compromises the attainment of acceptable divergence-freeness in the velocity solution.

Pressure oscillations were encountered in the step wall diffuser problem in the small inlet channel and in the vicinity of the step. Again, inclusion of artificial dissipation in the pressure equation was effective in diminishing the magnitude of these oscillations. The presence of these oscillations did not seem to adversely affect algorithm convergence because of the intrinsic odd-even decoupling of the pseudo-pressure discrete operator in the overall solution domain. Inclusion of artificial dissipation in the pseudo-pressure equation was deleterious in that the introduced smoothing terms affect the attainable divergence-freeness of the velocity field, which is zero for a truly incompressible flow. Hence, there is an upper bound on the level of artificial dissipation that can be employed if a converged solution satisfying an acceptable level of divergence-freeness is to be achieved.

The level of the pseudo-compressibility ε coefficient is an important variable aspect of the algorithm, since it affects convergence and convergence rate by controlling the speed of the traveling pressure waves. An ε value of 10 was used successfully for all three test cases, with the exception of the step wall diffuser where ε was lowered to 1.0 and finally to 0.1 to achieve converged solutions at $Re = 433$ and $Re = 577$, respectively. This suggests that the Reynolds number does indeed affect the range of acceptable pseudo-compressibility values, as implied by (2.19). The time required for predicted pressure distribution for a channel flow was generally confirmed, with convergence to an acceptable steady-state taking somewhat longer.

5.2 Recommendations

The evaluated pseudo-compressibility CFD algorithm shows potential for solving a wide range of incompressible fluid-thermal flow problems. The pressure decoupling at the corners, however, presents a significant obstacle which must be resolved before the developed benchmark problem range can be extended. Disregarding the corner difficulties, two primary aspects of the algorithm subject to future investigation are the pseudo-compressibility parameter and the artificial dissipation mechanism. The pseudo-compressibility ε is subject to optimization via its effect on solution stability, convergence, and convergence rate. Theoretical analysis is needed to determine lower *and* upper bounds on ε in the context of the finite element semi-discretization. Further numerical experiments would be useful in validating optimum values for ε for various problem specifications. Additionally, it may prove advantageous to permit ε to vary throughout the domain based on local velocity magnitudes (see [12]).

As shown in this study, artificial dissipation seems necessary in the pressure equation to control solution oscillations. The magnitude of these artificial smoothing terms throughout the solution domain is also subject to optimization in regard to effectiveness in controlling oscillations, improving convergence, and increasing convergence rate. It is clear that such an artificial smoothing term has a significant effect on the enforcement of conservation of mass, and consequently some measure of the divergence of the velocity field (without inclusion of smoothing terms) is needed. This measure could possibly be used in developing a way to automatically decrease artificial dissipation in regions of the solution domain where the divergence of the velocity is not converging to computational zero as the problem approaches a steady-state.

LIST OF REFERENCES

[1] Gresho, M.P. and Sani, R.L., "On Pressure Boundary Conditions for the Incompressible Navier-Stokes Equations," Int. J. Num. Mtd. Fluids, 7(1987), 1111-1145.

[2] Harlow, F.H. and Welch, J.E., "Numerical Calculation of Time-Dependent Viscous Incompressible Flow with Free Surface," Phys. Fluids, 8(1965), 2182-2189.

[3] Zienkiewicz, O.C. and Godbole, P.N., "Viscous Incompressible Flow with Special Reference to Non Newtonian (Plastic) Fluids," in R.H. Gallagher, et. al. (eds.), *Finite Elements in Fluids*, Vol. 1, Wiley-Interscience, London(1975), 25-55.

[4] Chorin, A.J., "A Numerical Method for Solving Incompressible Viscous Flow Problems," J. Comp. Phys., 2(1967), 12-26.

[5] Plows, W.H., "Some Numerical Results for Two-Dimensional Steady Laminar Benard Convection," Phys. Fluids, 11(1968), 1593-1599.

[6] Grabowski, N.J. and Berger, S.A., "Solutions of the Navier-Stokes Equations for Vortex Breakdown," AIAA Journal, 75(1976), 525-544.

[7] Steger, J.L. and Kutler, P., "Implicit Finite Difference Procedures for Computation of Vortex Wakes," AIAA Journal, 15(1977), 581-590.

[8] Chang, J.L.C. and Kwak, D., "On the Method of Pseudo Compressibility for Numerically Solving Incompressible Flows," AIAA Paper 84-0252(1984).

[9] Kwak, D., Chang, J.L.C., and Shanks, S.P., "An Incompressible Navier-Stokes Flow Solver in Three-Dimensional Curvilinear Coordinate System Using Primitive Variables," AIAA Paper 84-0253(1984).

[10] Soy, W.Y. and Berger, S.A., "Laminar Entrance Flow in a Curved Pipe," J. Fluid Mech., 148(1984), 109-135.

[11] Choi, D. and Merkle, C.L., "Application of Time-Iterative Schemes to Incompressible Flow," AIAA Journal, 23(1985), 1518-1524.

[12] Rizzi, A. and Eriksson, L., "Computation of Inviscid Incompressible Flow with Rotation," J. Fluid Mech., 153(1985), 275-312.

[13] Rogers, S.E., Kwak, D. and Kaul, U., "On the Accuracy of the Pseudocompressibility Method in Solving the Incompressible Navier-Stokes Equations," Appl. Math. Modeling, 11(1987), 35-44.

[14] Soh, W.Y., "Time Marching Solution of Incompressible Navier-Stokes Equations for Internal Flow," J. Comp. Phys., 70(1987), 232-251.

[15] Peyret, R. and Taylor, T.D., *Computational Methods for Fluid Flow*, Springer-Verlag, New York(1983).

[16] Iannelli, G.S. and Baker, A.J., "An Implicit and Stiffly Stable Finite Element CFD Algorithm for Unsteady Aerodynamics," AIAA Paper 89-0656(1989).

[17] Pulliam, T.H., "Efficient Solution Methods for the Navier-Stokes Equations," Lecture Notes: *Numerical Techniques for Viscous Flow Computation in Turbomachinery Bladings*, Von Karman Institute for Fluid Dynamics, Rhode St. Genese, Belgium, 20-24 Jan.(1986).

[18] White, F.M., *Viscous Fluid Flow*, McGraw-Hill, New York(1974).

[19] Baker, A.J. and Kim, J.W., "A Taylor Weak Statement Algorithm for Hyperbolic Conservations Laws," Int. J. Num. Mtds. Fluids, 7(1987), 489-520.

[20] Baker, A.J., *Finite Element Computational Fluid Mechanics*, Hemisphere, Washington, D.C.(1983).

[21] Lambert, J.D., *Computational Methods in Ordinary Differential Equations*, Wiley, New York(1973).

[22] Schlichting, H., *Boundary Layer Theory*, McGraw-Hill, New York(1955).

[23] Armaly, B.F., et al., "Experimental and Theoretical Investigation of Backward Facing Step Flow," J. Fluid Mech., 127(1983), 473-496.

[24] Noronha, W.P. and Baker, A.J., "A Taylor Weak Statement Finite Element CFD Algorithm for 2D Incompressible Navier Stokes Equations," AIAA Paper 89-0659(1989).

[25] de Vahl Davis, G., "Natural Convection of Air in a Square Cavity: A Bench Mark Numerical Solution," Int. J. Num. Mtd. Fluids, 3(1983), 249-264.

APPENDIXES

Appendix A

Figures

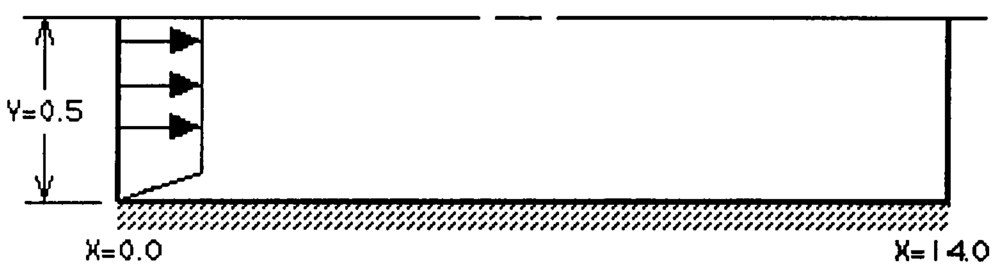

Figure A.1: Developing Channel Flow Geometry.

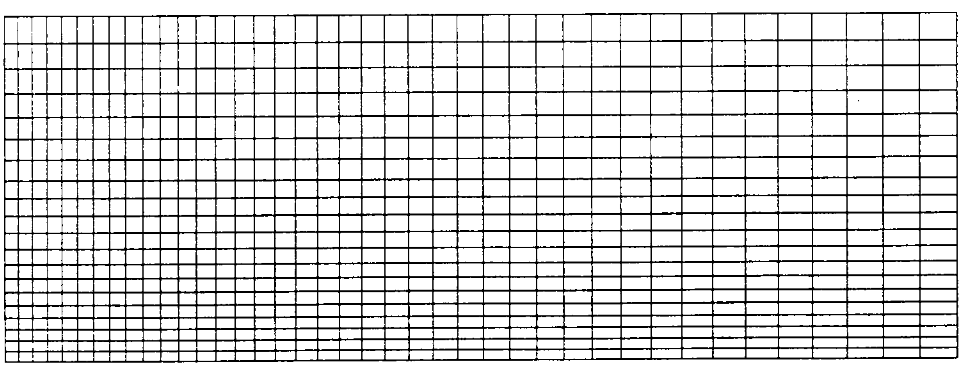

Figure A.2: Developing Channel Flow Mesh.

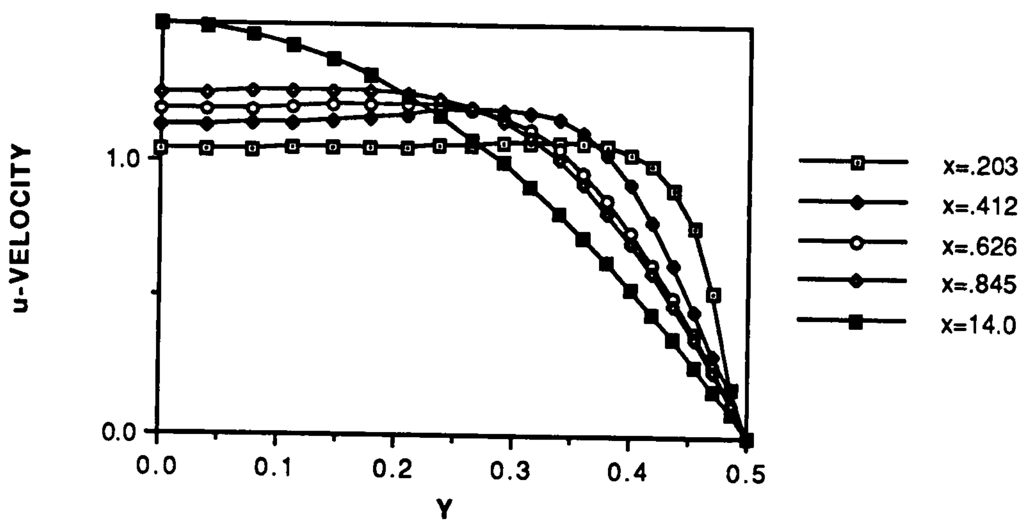

Figure A.3: Developing Channel Flow, u-Velocity Profiles, $Re = 50$.

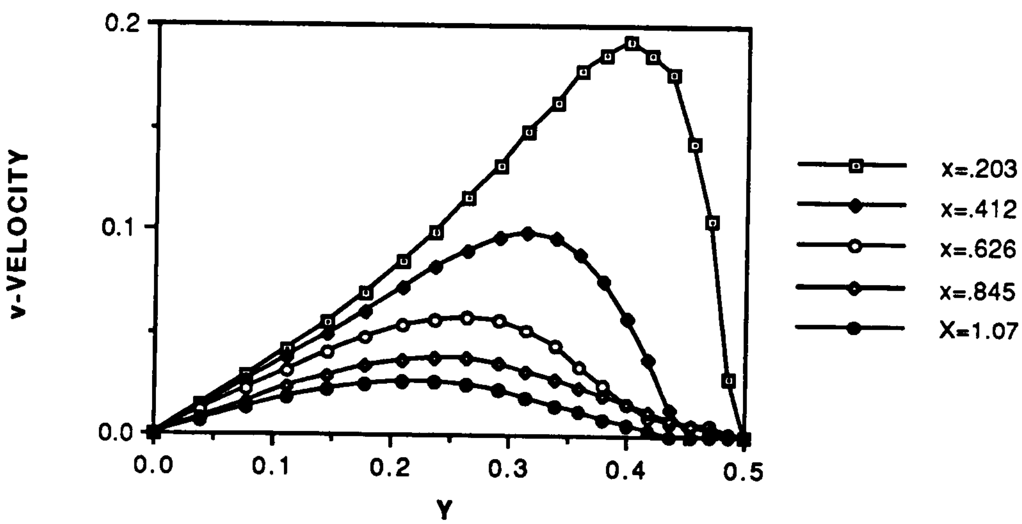

Figure A.4: Developing Channel Flow, v-Velocity Profiles, $Re = 50$.

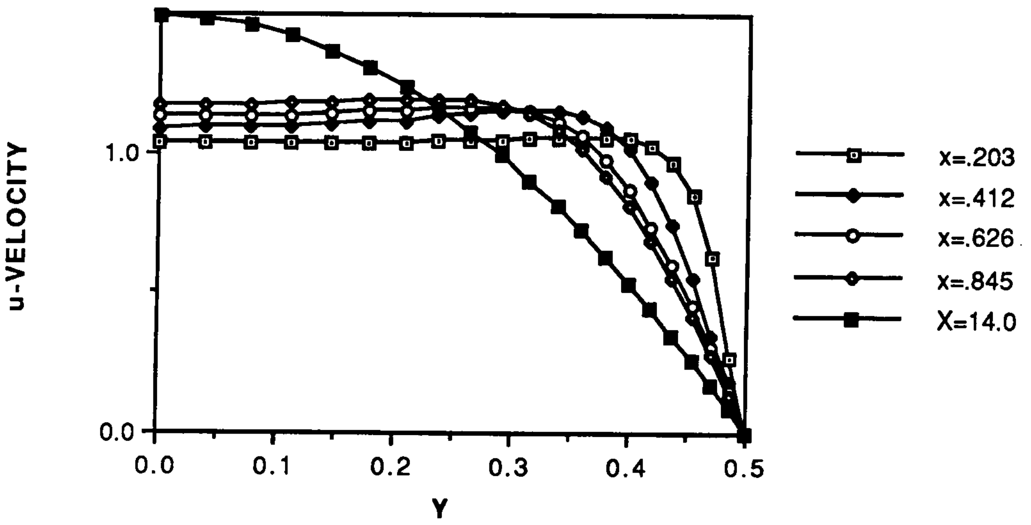

Figure A.5: Developing Channel Flow, u-Velocity Profiles, $Re = 100$.

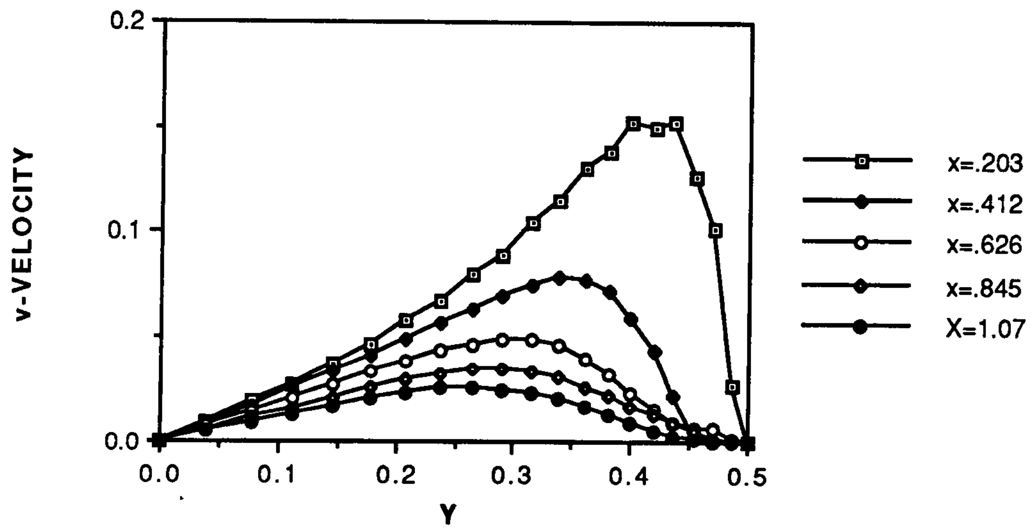

Figure A.6: Developing Channel Flow, v-Velocity Profiles, $Re = 100$.

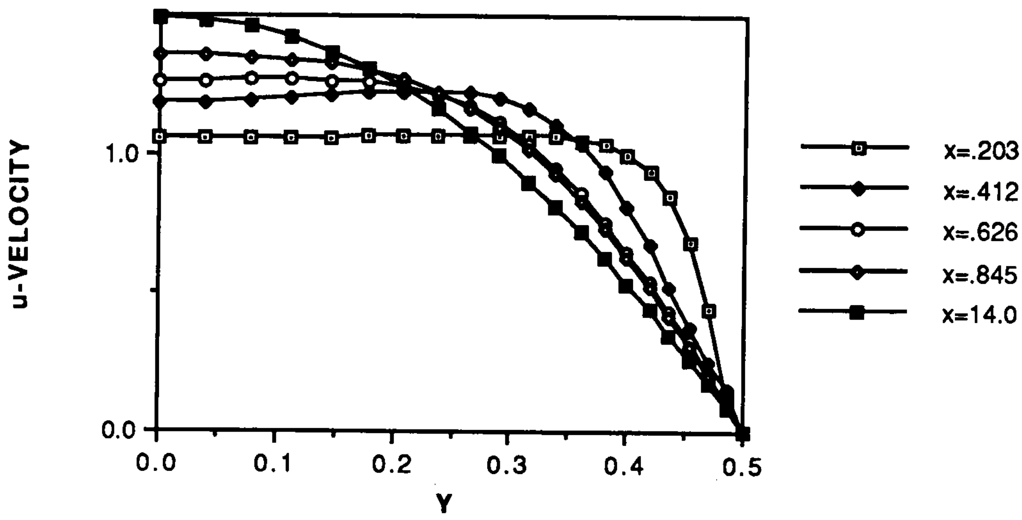

Figure A.7: Developing Channel Flow, u-Velocity Profiles, $Re = 200$.

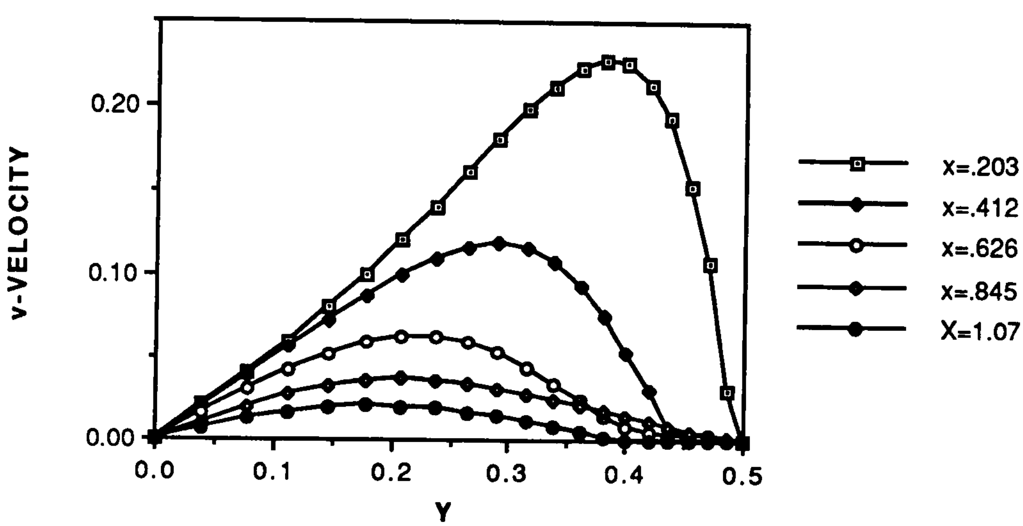

Figure A.8: Developing Channel Flow, v-Velocity Profiles, $Re = 200$.

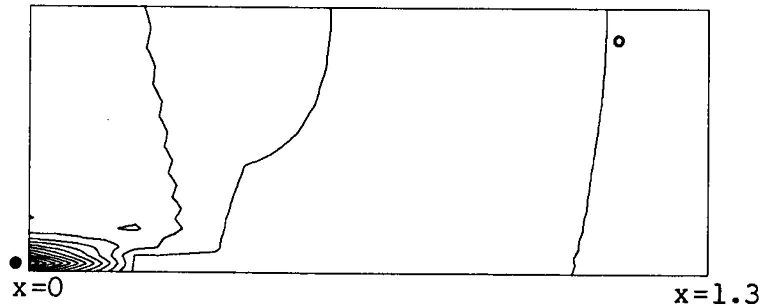

Figure A.9: Developing Channel Flow, Pressure Contours, $Re = 200$, $\beta_p = 0.5$, $\bullet - p_{max} = 22.3$, $\circ - p_{min} = 9.7$.

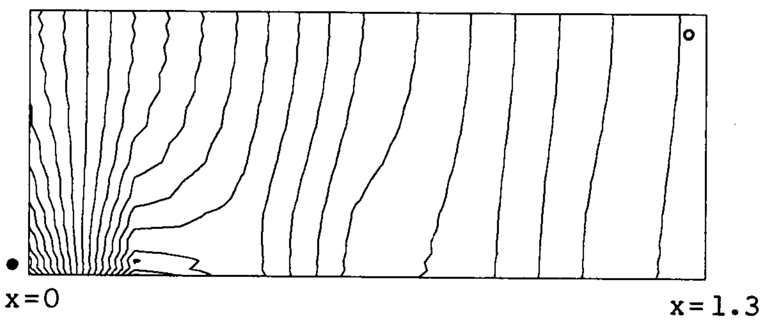

Figure A.10: Developing Channel Flow (with Upstream Inviscid Entrance Section), Pressure Contours, $Re = 200$, $\beta_p = 0.2$, $\bullet - p_{max} = 12.4$, $\circ - p_{min} = 9.0$.

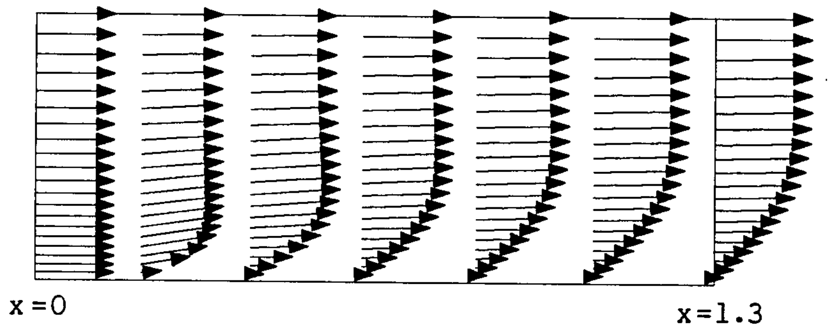

Figure A.11: Developing Channel Flow, Velocity Field (no Inviscid Approach Region), $Re = 200$, $\beta_u = 0.0$, $\beta_v = 0.0$.

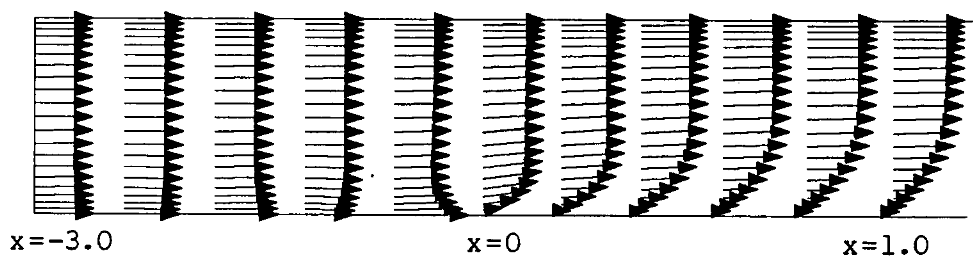

Figure A.12: Developing Channel Flow, Velocity Field (with Upstream Inviscid Entrance Section), Velocity Field, $Re = 200$, $\beta_u = 0.1$, $\beta_v = 0.1$.

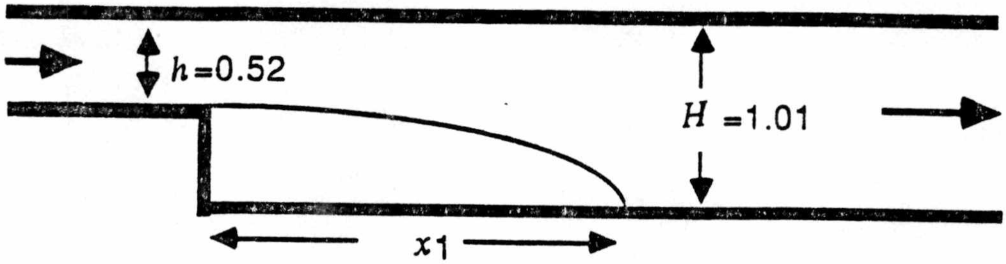

Figure A.13: Step Wall Diffuser Geometry.

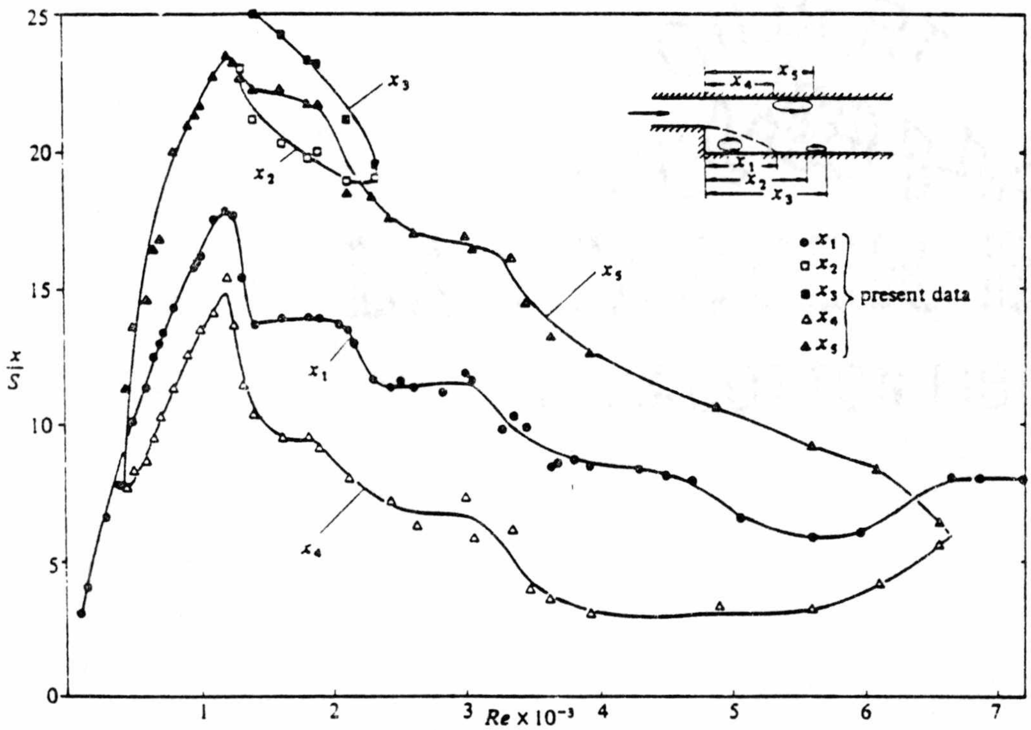

Figure A.14: Experimental Data of Armaly, et al. [17].

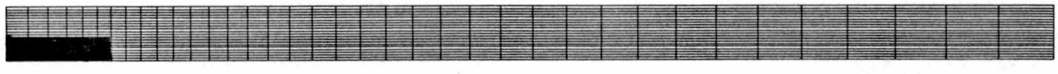

Figure A.15: Step Wall Diffuser Mesh.

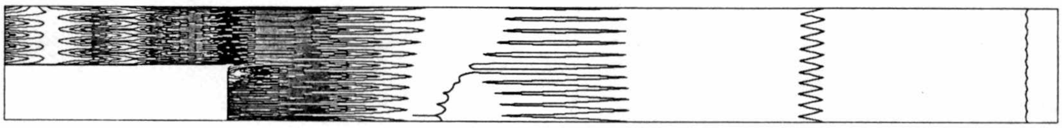

Figure A.16: Step Wall Diffuser, Pressure Contours, $Re = 288$, $\varepsilon = 10$, $\beta_p = 0.0$.

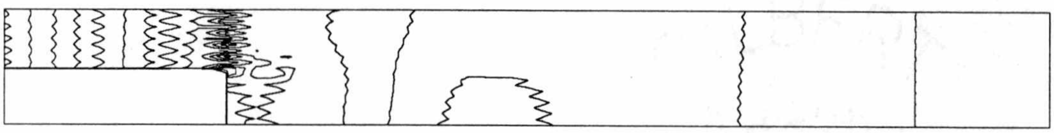

Figure A.17: Step Wall Diffuser, Pressure Contours, $Re = 288$, $\varepsilon = 10$, $\beta_p = 0.05$.

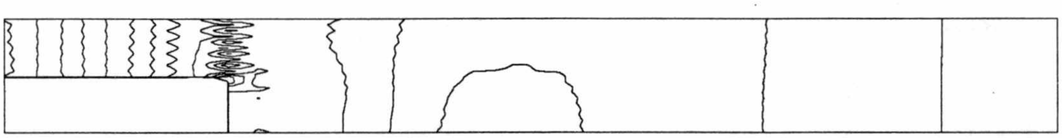

Figure A.18: Step Wall Diffuser, Pressure Contours, $Re = 288$, $\varepsilon = 10$, $\beta_p = 0.1$.

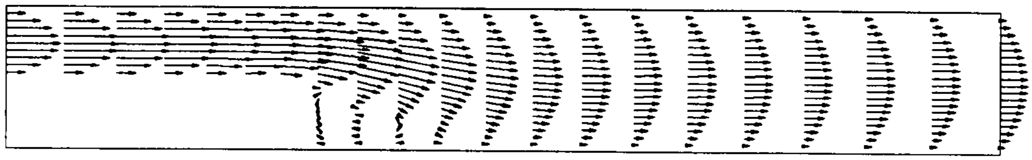

Figure A.19: Step Wall Diffuser, Velocity Field, $\varepsilon = 10$, $Re = 100$, $\beta_p = 0.0$.

Figure A.20: Step Wall Diffuser, Velocity Field, $\varepsilon = 10$, $Re = 200$, $\beta_p = 0.0$.

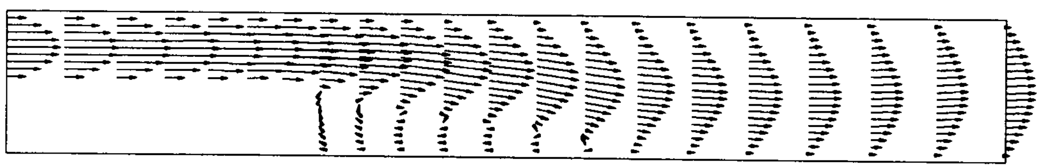

Figure A.21: Step Wall Diffuser, Velocity Field, $\varepsilon = 10$, $Re = 288$, $\beta_p = 0.0$.

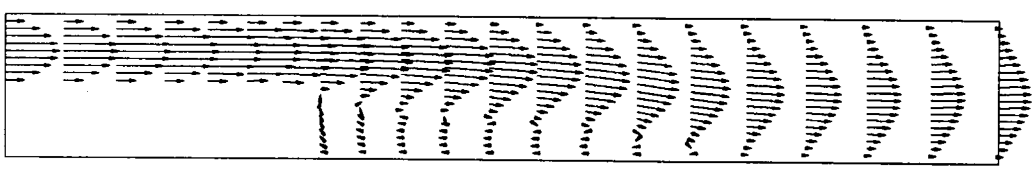

Figure A.22: Step Wall Diffuser, Velocity Field, $\varepsilon = 1.0$, $Re = 433$, $\beta_p = 0.0$.

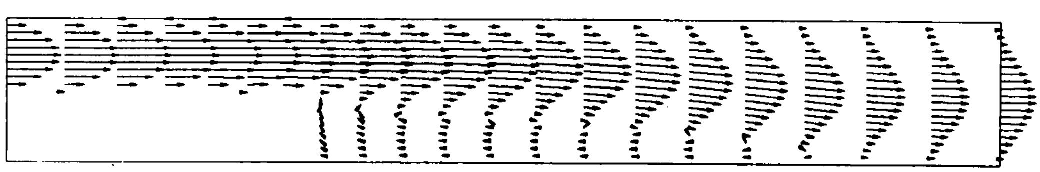

Figure A.23: Step Wall Diffuser, Velocity Field, $\varepsilon = 0.1$, $Re = 577$, $\beta_p = 0.0$.

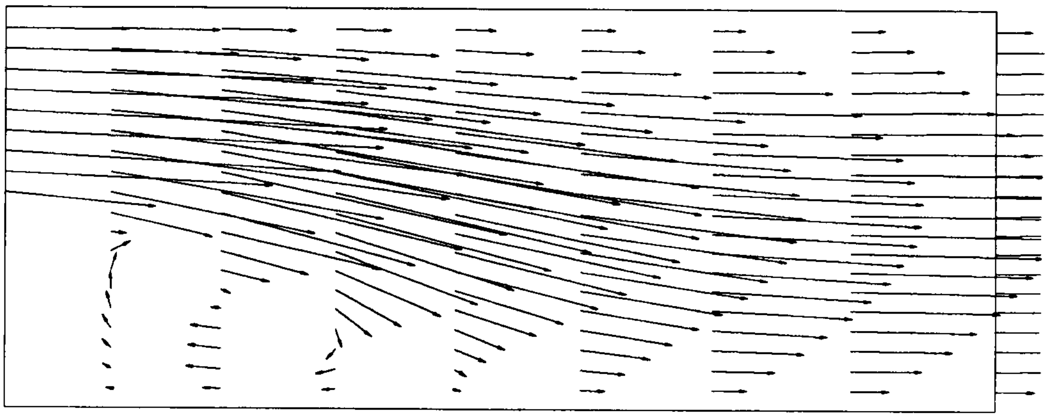

Figure A.24: Step Wall Diffuser, Velocity Field (Recirculation Zone), Re = 100, $\varepsilon = 10$, $\beta_p = 0.0$.

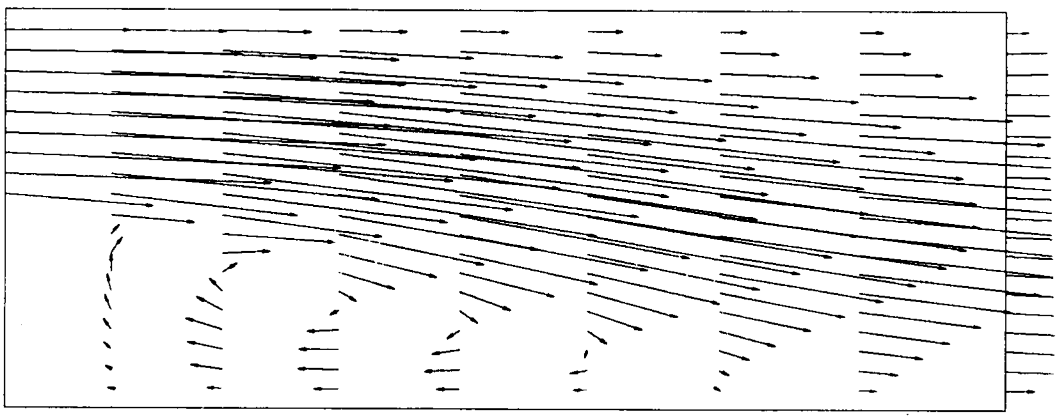

Figure A.25: Step Wall Diffuser, Velocity Field (Recirculation Zone), Re = 200, $\varepsilon = 10$, $\beta_p = 0.0$.

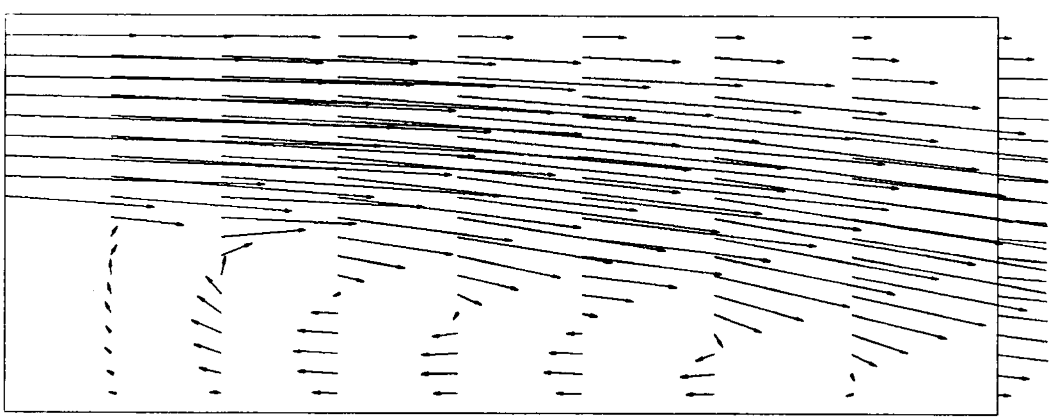

Figure A.26: Step Wall Diffuser, Velocity Field (Recirculation Zone), Re = 288, $\varepsilon = 10$, $\beta_p = 0.0$.

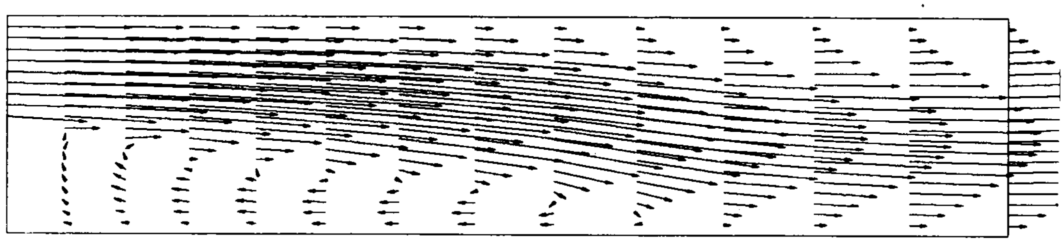

Figure A.27: Step Wall Diffuser, Velocity Field (Recirculation Zone), Re = 433, $\varepsilon = 1.0$, $\beta_p = 0.0$.

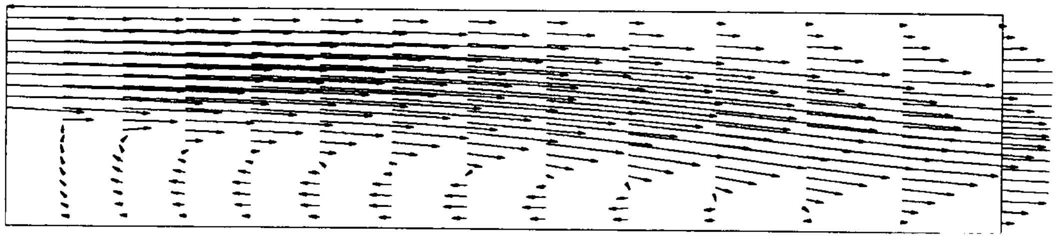

Figure A.28: Step Wall Diffuser, Velocity Field (Recirculation Zone), Re = 577, $\varepsilon = 0.1$, $\beta_p = 0.0$.

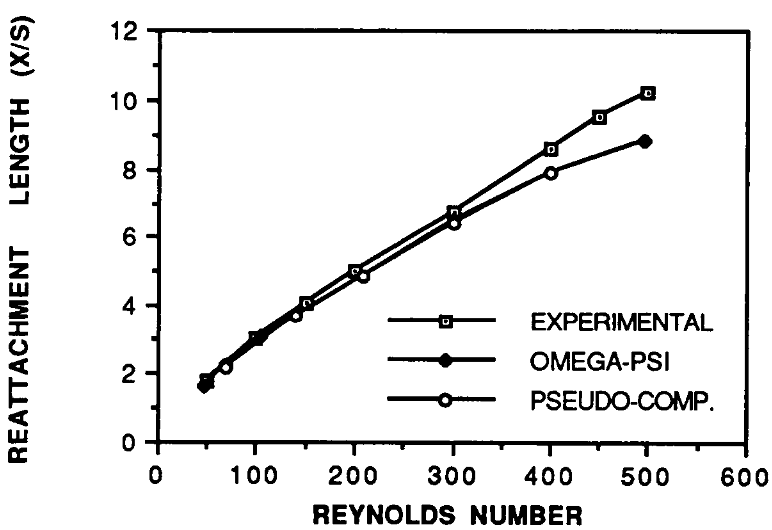

Figure A.29: Comparison of Experimental [17] and Predicted Primary Reattachment Length.

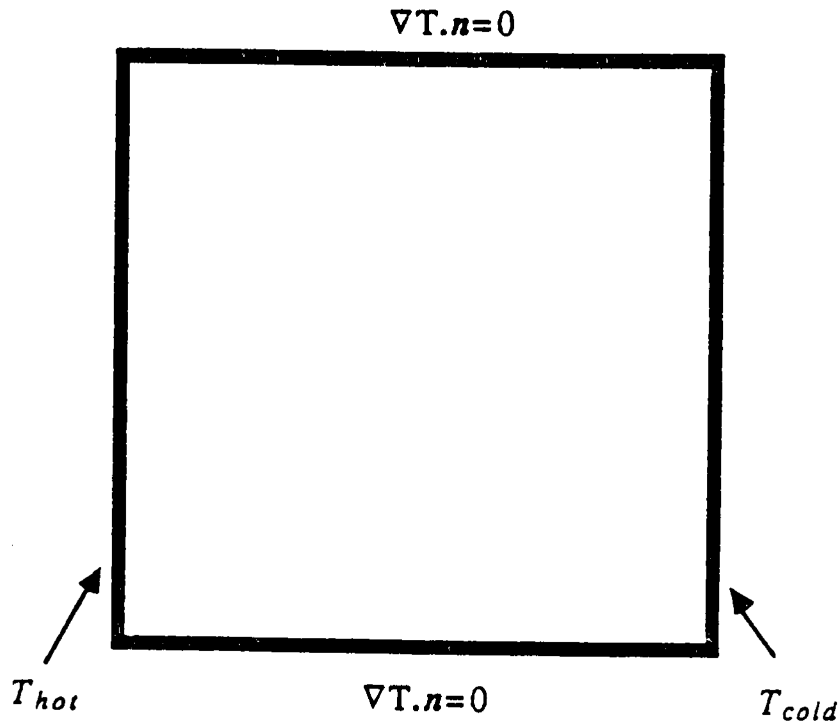

Figure A.30: Thermal Cavity Test Problem.

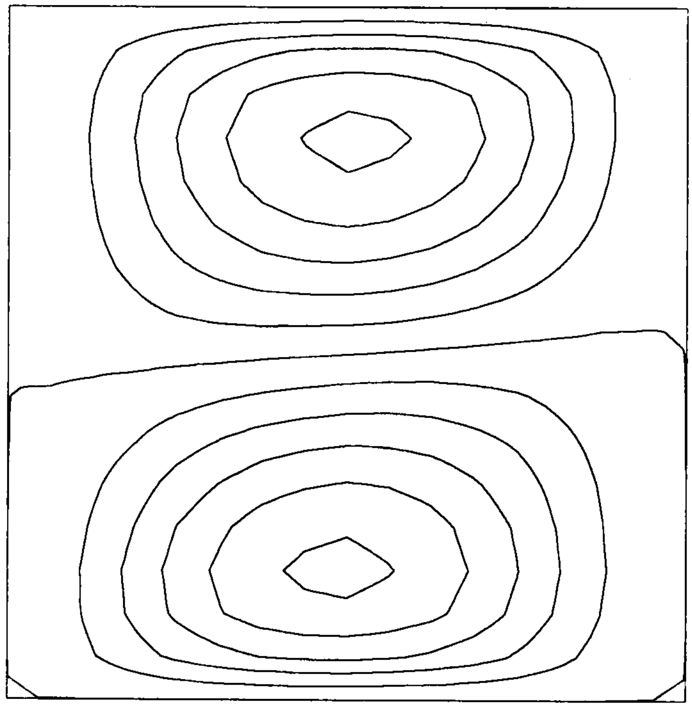

Figure A.31: Thermal Cavity, u-Velocity Contours, $Ra = 1000$, $\varepsilon = 10$, $\beta_p = .01$.

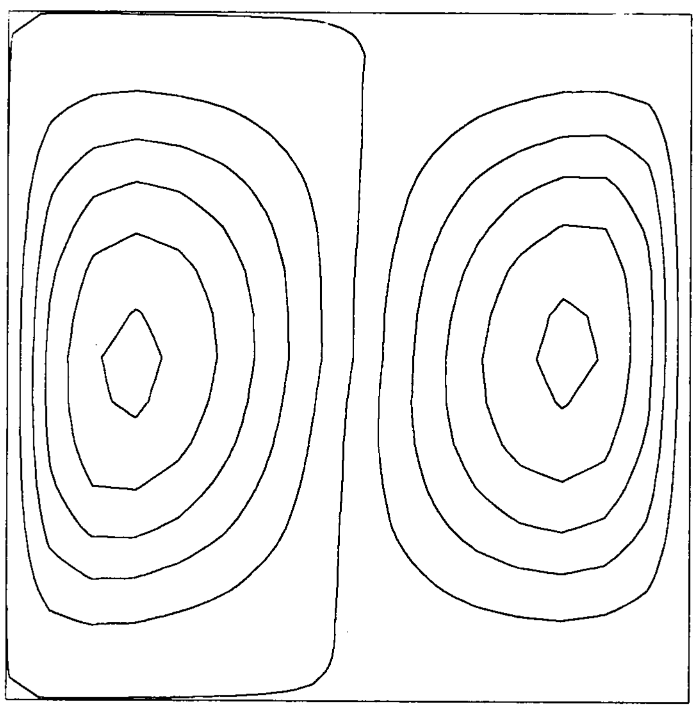

Figure A.32: Thermal Cavity, v-Velocity Contours, $Ra = 1000$, $\varepsilon = 10$, $\beta_p = .01$.

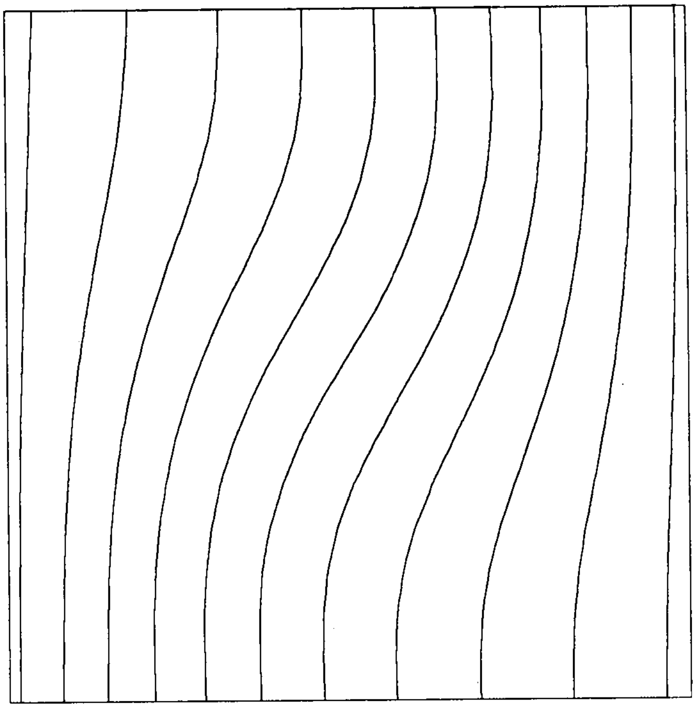

Figure A.33: Thermal Cavity, Temperature Contours, $Ra = 1000$, $\varepsilon = 10$, $\beta_p = .01$.

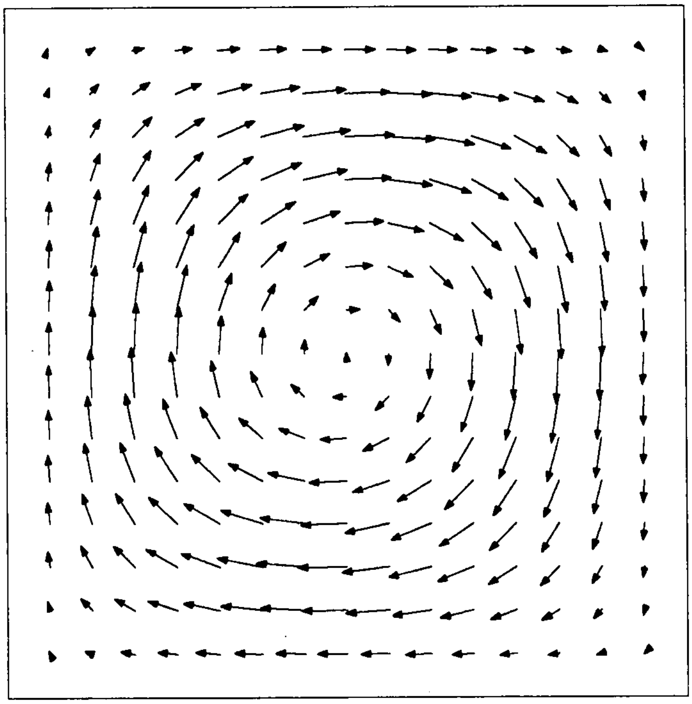

Figure A.34: Thermal Cavity, Velocity Field, $Ra = 1000$, $\varepsilon = 10$, $\beta_p = .01$.

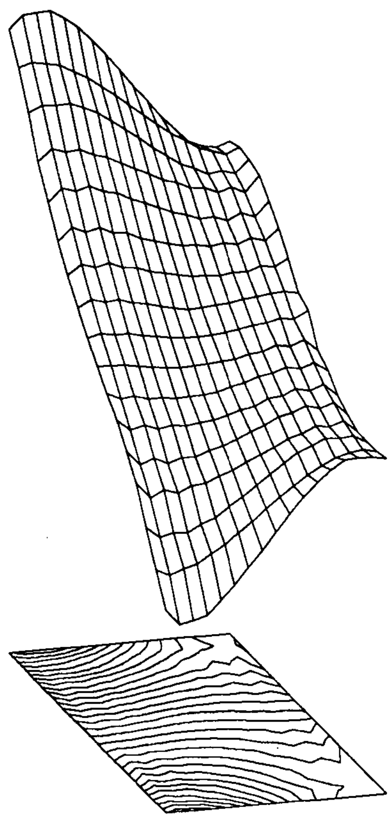

Figure A.35: Thermal Cavity, Pressure Solution, $Ra = 1000$, $\varepsilon = 10$, $\beta_p = .01$.

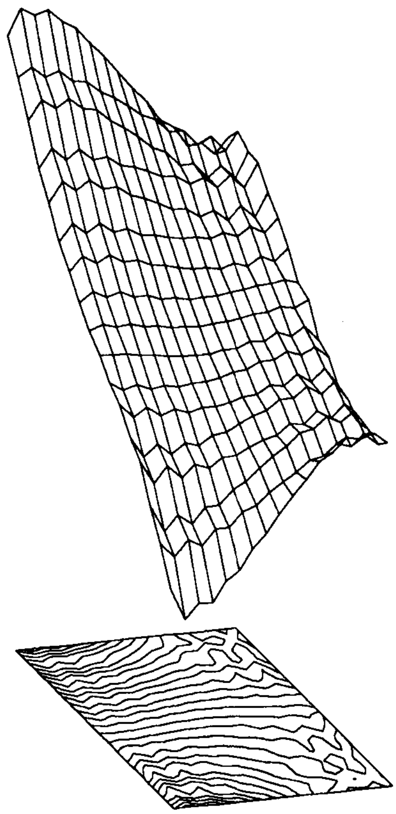

Figure A.36: Thermal Cavity, Pressure Solution, $Ra = 1000$, $\varepsilon = 10$, $\beta_p = 0.0$.

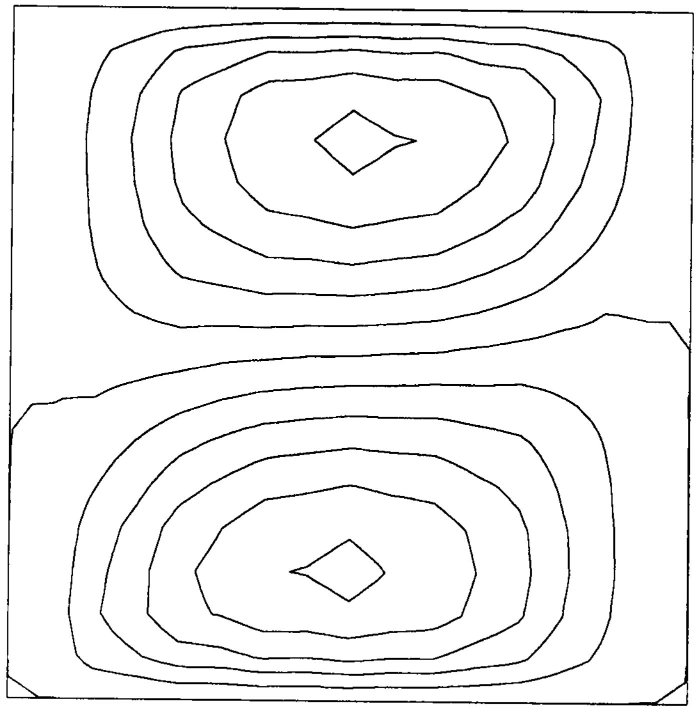

Figure A.37: Thermal Cavity, u-Velocity Contours, $Ra = 5000$, $\varepsilon = 50$, $\beta_p = 0.0$.

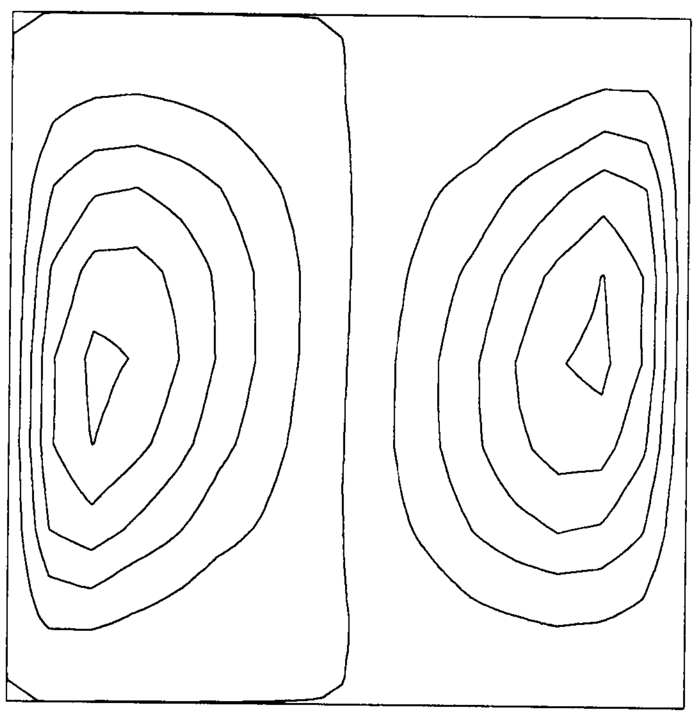

Figure A.38: Thermal Cavity, v-Velocity Contours, $Ra = 5000$, $\varepsilon = 50$, $\beta_p = 0.0$.

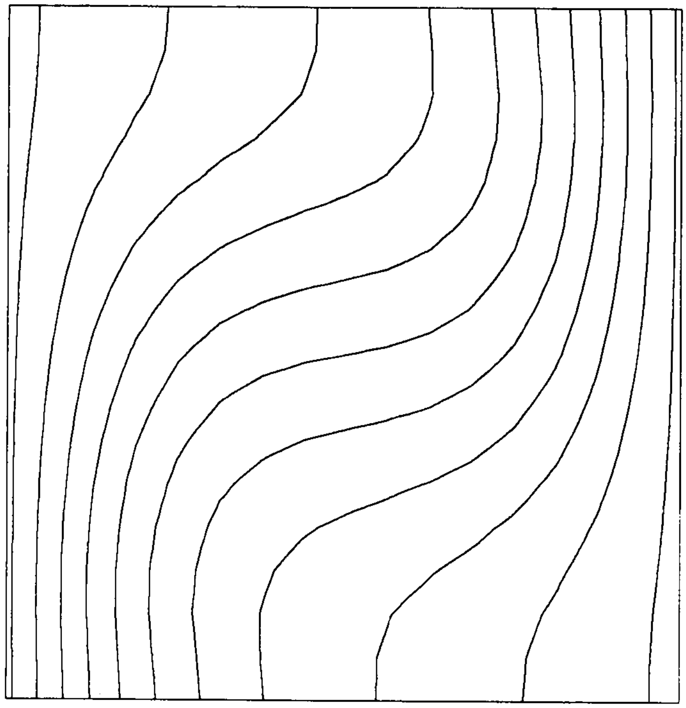

Figure A.39: Thermal Cavity, Temperature Contours, $Ra = 5000$, $\varepsilon = 50$, $\beta_p = 0.0$.

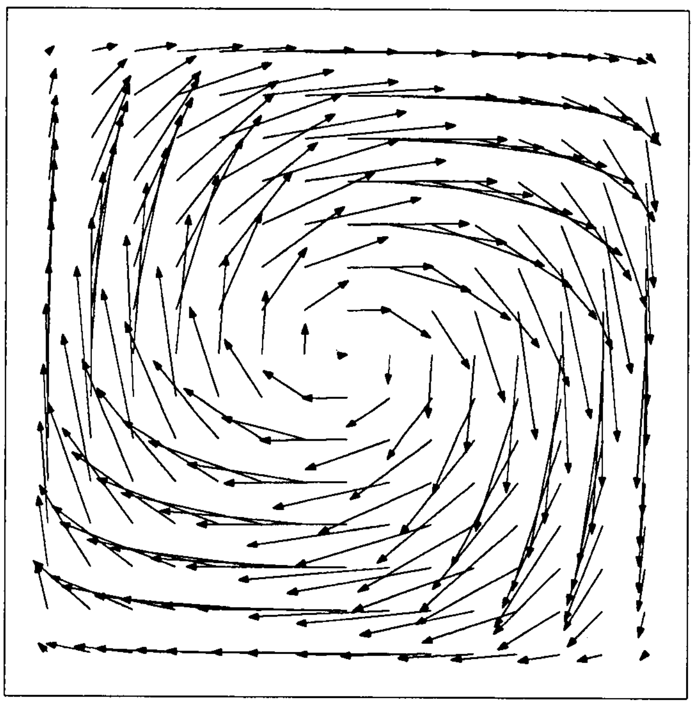

Figure A.40: Thermal Cavity, Velocity Field, $Ra = 5000$, $\varepsilon = 50$, $\beta_p = 0.0$.

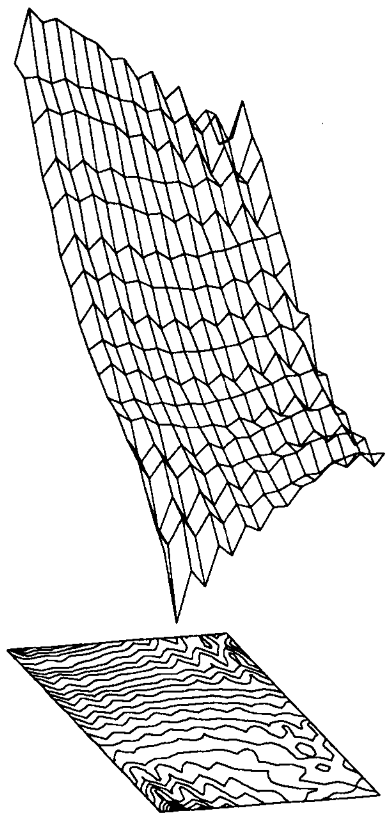

Figure A.41: Thermal Cavity, Pressure Solution, $Ra = 5000$, $\varepsilon = 50$, $\beta_p = 0.0$.

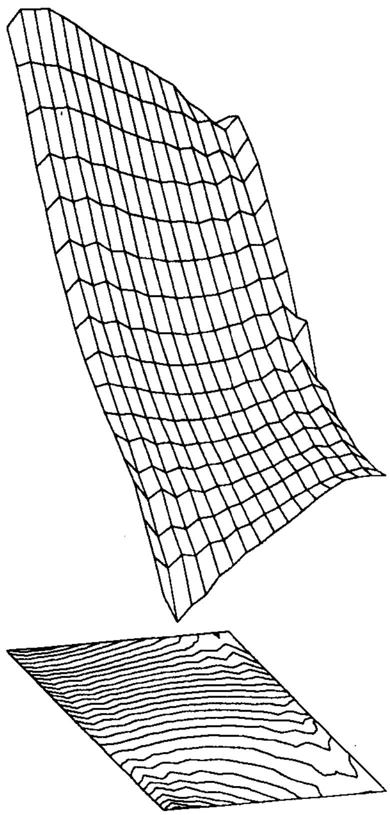

Figure A.42: Thermal Cavity, Pressure Solution, $Ra = 5000$, $\varepsilon = 50$, $\beta_p = 0.1$.

Appendix B

TWS Stability Matrices

As outlined in Section 3.1, the TWS stability matrices are formed utilizing a set of scalar parameters contained in the diagonal matrix β and the kinematic-kinetic flux vector Jacobian matrix $A_j \equiv \frac{\partial f_j}{\partial q}$. The matrix β in two dimensions is defined as

$$\beta \equiv \begin{bmatrix} \beta_p & 0 & 0 & 0 \\ 0 & \beta_u & 0 & 0 \\ 0 & 0 & \beta_v & 0 \\ 0 & 0 & 0 & \beta_\Theta \end{bmatrix} \tag{B.1}$$

where β_p, β_u, β_v, and β_Θ are scalar constants.

The diagonal matrix $\mathbf{D}_{\underline{j}}$ defined in (3.8) is derived by a simplification of the matrix product $A_j A_k$. For a two-dimensional problem, A_1 and A_2 are

$$A_1 \equiv \frac{\partial f_1}{\partial q} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 2u & 0 & 0 \\ 0 & v & u & 0 \\ 0 & \Theta & 0 & u \end{bmatrix} \quad , \quad A_2 \equiv \frac{\partial f_2}{\partial q} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & v & u & 0 \\ 1 & 0 & 2v & 0 \\ 0 & 0 & \Theta & v \end{bmatrix} \tag{B.2}$$

The matrix product $A_j A_k$ is then contracted to $A_{\underline{j}} A_{\underline{j}}$ and diagonalized by discarding the off-diagonal terms forming $\mathbf{D}_{\underline{j}}$ as

$$\mathbf{D}_1 = \frac{h}{|\mathbf{u}|} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1+4u^2 & 0 & 0 \\ 0 & 0 & u^2 & 0 \\ 0 & 0 & 0 & u^2 \end{bmatrix} \tag{B.3}$$

$$\mathbf{D}_2 = \frac{h}{|\mathbf{u}|}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & v^2 & 0 & 0 \\ 0 & 0 & 1+4v^2 & 0 \\ 0 & 0 & 0 & v^2 \end{bmatrix} \tag{B.4}$$

Note that the local reference values h and $|\mathbf{u}|$ from (3.7) have also been included in the definitions of $\mathbf{D}_{\underline{j}}$.

The terms $(1+4u^2)$ and $(1+4v^2)$ in (B.3) and (B.4), respectively, are simplified to u^2 and v^2. Next, all the squared velocity terms are "generalized" as $|\mathbf{u}|^2$, where $\mathbf{u} = \sqrt{u^2+v^2}$ and u and v are local flow speeds. Dividing through by $|\mathbf{u}|$, $\mathbf{D}_{\underline{j}}$ is obtained as

$$\mathbf{D}_{\underline{j}} = h\begin{bmatrix} |\mathbf{u}|^{-1} & 0 & 0 & 0 \\ 0 & |\mathbf{u}| & 0 & 0 \\ 0 & 0 & |\mathbf{u}| & 0 \\ 0 & 0 & 0 & |\mathbf{u}| \end{bmatrix} \tag{B.5}$$

Again, the subscript $\underline{j}$ refers to a coordinate direction, and $\mathbf{D}_1$ differs from $\mathbf{D}_2$ since the local reference length h depends on orientation, hence also in the tensor product Jacobians employed in the time integration scheme outlined in Section 3.5. Indicating these distinct reference lengths as $h_{\underline{j}}$, the final form for $\mathbf{D}_{\underline{j}}$ is

$$\mathbf{D}_{\underline{j}} \equiv \begin{bmatrix} |\mathbf{u}|^{-1}h_{\underline{j}} & 0 & 0 & 0 \\ 0 & |\mathbf{u}|h_{\underline{j}} & 0 & 0 \\ 0 & 0 & |\mathbf{u}|h_{\underline{j}} & 0 \\ 0 & 0 & 0 & |\mathbf{u}|h_{\underline{j}} \end{bmatrix} \tag{B.6}$$

Thus, the TWS stability mechanism for (3.8) is finally expressed as

$$\frac{\partial}{\partial x_j}\begin{bmatrix} \beta_p & 0 & 0 & 0 \\ 0 & \beta_u & 0 & 0 \\ 0 & 0 & \beta_v & 0 \\ 0 & 0 & 0 & \beta_\Theta \end{bmatrix}\begin{bmatrix} |\mathbf{u}|^{-1}h_{\underline{j}} & 0 & 0 & 0 \\ 0 & |\mathbf{u}|h_{\underline{j}} & 0 & 0 \\ 0 & 0 & |\mathbf{u}|h_{\underline{j}} & 0 \\ 0 & 0 & 0 & |\mathbf{u}|h_{\underline{j}} \end{bmatrix}\begin{Bmatrix} \partial p/\partial x_j \\ \partial u/\partial x_j \\ \partial v/\partial x_j \\ \partial \Theta/\partial x_j \end{Bmatrix} \tag{B.7}$$

The elements of the matrix $\mathbf{D}_{\underline{j}}$ are interpolated locally by the basis function $\{N\}$ as shown in the finite element semi-discrete statement (3.21).

Considering only the pseudo-pressure equation, the TWS stability mechanism can be expressed as

$$\frac{\partial}{\partial x_j}\left(\beta_p \frac{h_{\underline{j}}}{|\mathbf{u}|}\right)\frac{\partial p}{\partial x_j} \tag{B.8}$$

Since β_p is an user-defined constant and $h_{\underline{j}}/|\mathbf{u}|$ is a local constant, (B.9) can be rewritten as

$$\beta_p\left(\frac{h_{\underline{j}}}{|\mathbf{u}|}\right)\frac{\partial}{\partial x_j}\left(\frac{\partial p}{\partial x_j}\right) \tag{B.9}$$

or

$$\beta_p\left(\frac{h_{\underline{j}}}{|\mathbf{u}|}\right)\nabla^2 p \tag{B.10}$$

Hence, the origin of the $\nabla^2 p$ coupling term referred to in Chapter 4 is observed.

Appendix C

Sample Input Stream

Before running the main program, entitled PSEUDO, the necessary data files for the particular problem under consideration must be generated. As an example problem definition procedure, the input stream for the step wall diffuser is presented along with a brief description of each sub-program utilized to create the necessary data file.

MESH – The Mesh Generator Routine

MESH is an interactive program which builds the data file for a mesh suitable for the flow analysis. It allows the user to create arbitrary grids constituted of quadrilateral elements bounded by linear, parabolic, or cubic boundary segments. The output files created by MESH are:

FOR060.DAT containing the number of columns and rows, number of elements, nodes, boundary nodes, and number of nodes per element (four),

FOR062.DAT containing the element nodal connection array and the boundary node vector array, and

FOR064.DAT containing the x and y coordinates of the nodes.

The MESH input stream for the step wall diffuser problem specification is as follows. When MESH is accessed, the user (hereafter referred to as "you") is asked if a new grid is to be generated. Assuming that a new grid is to be created, the total number of element columns (IMAX) and the total number of element rows (JMAX) are requested.

```
$ RUN MESH
Do you wish to generate a new
grid or view a previous grid?

0  >>  view old grid only
1  >>  create a new grid

1
INPUT
IMAX ?
37
JMAX ?
20
OK,YOU HAVE              740 ELEMENTS
WHICH FORM              798 NODES ALTOGETHER
            114 OF WHICH ARE BOUNDARY NODES !
```

For the step wall diffuser, a rectangular grid is generated and the step is later "blocked out" by another sub-program (TSGEN). Hence, there is only one "y-coordinate zone" and one "x-coordinate zone" and no y-boundary or x-boundary cusps, see Figure C.1.

```
Y-CO-ORDINATE ZONES
TOTAL NUMBER OF ZONES ?
1
ANY Y-BOUNDARY CUSPS ?  (1=YES)
0
X-CO-ORDINATE ZONES
TOTAL NUMBER OF ZONES ?
1
ANY X-BOUNDARY CUSPS ?  (1=YES)
0
```

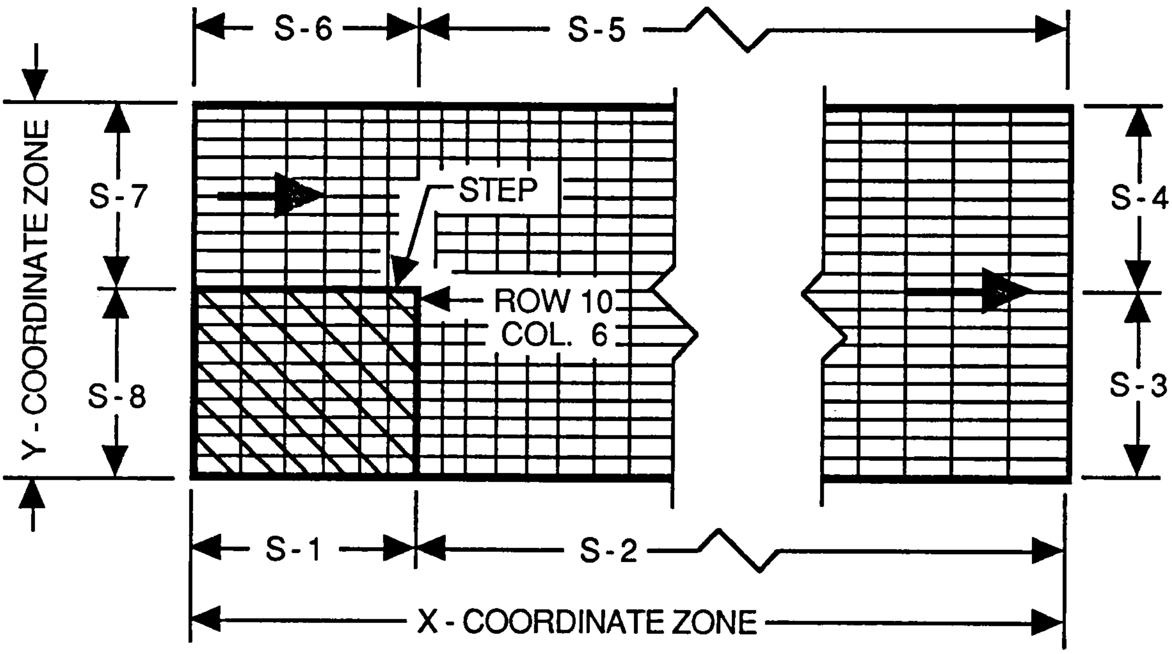

Figure C.1: Step Wall Diffuser Geometry

Next, you are requested to input the total number of boundary segments. Since the grid is rectangular, there are actually four boundary segments. However, due to the presence of the step and local grid refinement in the streamwise direction (x-direction) near the step, a total of eight boundary segments are specified. These segments are labeled in Figure C.1 as S-1 through S-8. The ending point (node) of each segment is located by inputting the appropriate row, column, and node number. The lower left element of the mesh is located at row one, column one, and element nodes are numbered counter-clockwise beginning with the lower left node of each element. The beginning of segment number one, as well as the end of the final segment, is assumed by the program to be the first row, first column, local node number one, with segments input sequentially running counter-clockwise. For this example, the "second extreme node of section 1" is located at the base of the step at row 1, column 6, and node 2.

```
 BOUNDARY GEOMETRY
 HOW MANY SECTIONS ?
8
SECOND EXTREME NODE OF SECTION:                   1
NODE LOCATION:
ELEMENT CO-ORDINATES:
ELEMENT ROW ?
1
ELEMENT COLUMN ?
6
NODE LOCAL NUMBER ?
2
ELEMENT              101 NODE          127
SECOND EXTREME NODE OF SECTION:                   2
NODE LOCATION:
ELEMENT CO-ORDINATES:
ELEMENT ROW ?
1
ELEMENT COLUMN ?
37
NODE LOCAL NUMBER ?
2
ELEMENT              721 NODE          778
SECOND EXTREME NODE OF SECTION:                   3
NODE LOCATION:
ELEMENT CO-ORDINATES:
ELEMENT ROW ?
10
ELEMENT COLUMN ?
37
NODE LOCAL NUMBER ?
3
ELEMENT              730 NODE          788
SECOND EXTREME NODE OF SECTION:                   4
NODE LOCATION:
ELEMENT CO-ORDINATES:
ELEMENT ROW ?
20
ELEMENT COLUMN ?
37
NODE LOCAL NUMBER ?
3
ELEMENT              740 NODE          798
SECOND EXTREME NODE OF SECTION:                   5
NODE LOCATION:
ELEMENT CO-ORDINATES:
ELEMENT ROW ?
20
ELEMENT COLUMN ?
6
NODE LOCAL NUMBER ?
```

```
3
ELEMENT            120 NODE          147
SECOND EXTREME NODE OF SECTION:            6
NODE LOCATION:
ELEMENT CO-ORDINATES:
ELEMENT ROW ?
20
ELEMENT COLUMN ?
1
NODE LOCAL NUMBER ?
4
ELEMENT             20 NODE           21
SECOND EXTREME NODE OF SECTION:            7
NODE LOCATION:
ELEMENT CO-ORDINATES:
ELEMENT ROW ?
10
ELEMENT COLUMN ?
1
NODE LOCAL NUMBER ?
4
ELEMENT             10 NODE           11
```

The x and y coordinates of the defined boundary section endpoints are next input. The coordinates of the first node of the first element (i.e. the lower left corner node) are initially requested. The coordinates of the final node of every segment are then specified in order. The program assumes that the beginning of each segment corresponds to the end of the previous segment. For the present example, the lower left hand corner is the beginning of segment 1 and has coordinates (0.0, 0.0). The end of the first boundary segment, located at the base of the step, has coordinates (2.0, 0.0).

```
EXTREME-NODE CO-ORDINATES
FIRST NODE CO-ORDINATES
X= ?
0.0
Y= ?
0.0
SECTION              1
LAST-NODE CO-ORDINATES
X= ?
2.0
```

```
Y= ?
0.0
SECTION            2
LAST-NODE CO-ORDINATES
X= ?
20.0
Y= ?
0.0
SECTION            3
LAST-NODE CO-ORDINATES
X= ?
20.0
Y= ?
0.49
SECTION            4
LAST-NODE CO-ORDINATES
X= ?
20.0
Y= ?
1.01
SECTION            5
LAST-NODE CO-ORDINATES
X= ?
2.0
Y= ?
1.01
SECTION            6
LAST-NODE CO-ORDINATES
X= ?
0.0
Y= ?
1.01
SECTION            7
LAST-NODE CO-ORDINATES
X= ?
0.0
Y= ?
0.49
```

To complete the boundary segment specification, the type of boundary segment is requested. If either of the first two choices is selected, the appropriate vertical or horizontal line is implemented. If either of the last two are chosen, then the degree polynomial of the dependent variable is additionally requested.

```
X<CONST  Y<VAR             CODE: 1
X<VAR    Y<CONST           CODE: 2
X<VAR    Y<VAR , Y<IND     CODE: 3
X<VAR    Y<VAR , X<IND     CODE: 4
2
SECTION          2
X<CONST  Y<VAR             CODE: 1
X<VAR    Y<CONST           CODE: 2
X<VAR    Y<VAR , Y<IND     CODE: 3
X<VAR    Y<VAR , X<IND     CODE: 4
2
SECTION          3
X<CONST  Y<VAR             CODE: 1
X<VAR    Y<CONST           CODE: 2
X<VAR    Y<VAR , Y<IND     CODE: 3
X<VAR    Y<VAR , X<IND     CODE: 4
1
SECTION          4
X<CONST  Y<VAR             CODE: 1
X<VAR    Y<CONST           CODE: 2
X<VAR    Y<VAR , Y<IND     CODE: 3
X<VAR    Y<VAR , X<IND     CODE: 4
1
SECTION          5
X<CONST  Y<VAR             CODE: 1
X<VAR    Y<CONST           CODE: 2
X<VAR    Y<VAR , Y<IND     CODE: 3
X<VAR    Y<VAR , X<IND     CODE: 4
2
SECTION          6
X<CONST  Y<VAR             CODE: 1
X<VAR    Y<CONST           CODE: 2
X<VAR    Y<VAR , Y<IND     CODE: 3
X<VAR    Y<VAR , X<IND     CODE: 4
2
SECTION          7
X<CONST  Y<VAR             CODE: 1
X<VAR    Y<CONST           CODE: 2
X<VAR    Y<VAR , Y<IND     CODE: 3
X<VAR    Y<VAR , X<IND     CODE: 4
1
SECTION          8
X<CONST  Y<VAR             CODE: 1
X<VAR    Y<CONST           CODE: 2
X<VAR    Y<VAR , Y<IND     CODE: 3
X<VAR    Y<VAR , X<IND     CODE: 4
1
```

The grid point distribution of each successive segment is the final item requested. For the step wall diffuser, the grid spacing is uniform in the y-direction and refined in the vicinity of the step in the streamwise direction. Hence, in segment 1 ($x = 0.0$ to $x = 2.0$) a right side concentration is specified while in segment 2 ($x = 2.0$ to $x = 20.0$) a left side concentration is selected. Reference to "left" or "right" is always made from a viewer outside the boundary segment looking toward the interior of the domain. The stretching strength is the intensity of the grid non-uniformity. A strength of 1.0 corresponds to uniform spacing, whereas a higher number yields a non-uniformly distributed grid.

```
POINT-DISTRIBUTION STRETCHING
SECTION              1
STRETCHING:
UNIFORM         CODE: 1
LEFT-SIDED      CODE: 2
RIGHT-SIDED     CODE: 3
BOTH-SIDED      CODE: 4
CENTERED        CODE: 5
3
STRETCHING STRENGTH   (1 THROUGH 20)
0.5
SECTION              2
STRETCHING:
UNIFORM         CODE: 1
LEFT-SIDED      CODE: 2
RIGHT-SIDED     CODE: 3
BOTH-SIDED      CODE: 4
CENTERED        CODE: 5
2
STRETCHING STRENGTH   (1 THROUGH 20)
1.35
SECTION              3
STRETCHING:
UNIFORM         CODE: 1
LEFT-SIDED      CODE: 2
RIGHT-SIDED     CODE: 3
BOTH-SIDED      CODE: 4
CENTERED        CODE: 5
1
```

```
SECTION              4
STRETCHING:
UNIFORM         CODE: 1
LEFT-SIDED      CODE: 2
RIGHT-SIDED     CODE: 3
BOTH-SIDED      CODE: 4
CENTERED        CODE: 5
1
SECTION              5
STRETCHING:
UNIFORM         CODE: 1
LEFT-SIDED      CODE: 2
RIGHT-SIDED     CODE: 3
BOTH-SIDED      CODE: 4
CENTERED        CODE: 5
3
STRETCHING STRENGTH  (1 THROUGH 20)
1.35
SECTION              6
STRETCHING:
UNIFORM         CODE: 1
LEFT-SIDED      CODE: 2
RIGHT-SIDED     CODE: 3
BOTH-SIDED      CODE: 4
CENTERED        CODE: 5
2
STRETCHING STRENGTH  (1 THROUGH 20)
0.5
SECTION              7
STRETCHING:
UNIFORM         CODE: 1
LEFT-SIDED      CODE: 2
RIGHT-SIDED     CODE: 3
BOTH-SIDED      CODE: 4
CENTERED        CODE: 5
1
SECTION              8
STRETCHING:
UNIFORM         CODE: 1
LEFT-SIDED      CODE: 2
RIGHT-SIDED     CODE: 3
BOTH-SIDED      CODE: 4
CENTERED        CODE: 5
1
```

STARDA – The Data Preparation Routine

The data preparation routine STARDA is an interactive program which builds the files necessary to run the PSEUDO finite element analysis code. The user must specify all initial and boundary conditions, problem parameters such as reference values and the pseudo-compressibility coefficient, as well as algorithm artificial dissipation coefficients, time step variation and time integration parameters. The input files required by STARDA are the three files generated by MESH: FOR060.DAT, FOR062.DAT, FOR064.DAT. The output files generated by STARDA are:

FOR059.DAT containing the numerical dissipation (beta) parameters,

FOR061.DAT containing the initial conditions distribution

FOR063.DAT containing, reference length, reference velocity, Re, pseudo-compressibility, Ra, reference pressure, Pr, T_{max}, and T_{ref},

FOR065.DAT containing the initial outlet pressures and the inlet flow angle,

FOR066.DAT containing the time integration parameters,

FOR067.DAT containing the current time, delta time, and time step variation parameters, and

FOR068.DAT containing information about each of the nodes such as boundary condition(s) applied (node codes) and number of variables per node.

The program begins by requesting information about the initial conditions and problem variables (such as reference values, Reynolds number Re, etc.). If you choose to set up initial conditions, the subroutine USDFIC (USer DeFined Initial Conditions) is accessed. Additionally, an exit pressure can be explicitly specified and parabolic or slug velocity profiles automatically generated throughout the domain (based on maximum velocities specified in USDFIC). Prescribing the inlet flow direction dictates the initial flow direction throughout the domain. The inlet flow direction is input in degrees measured counter-clockwise from the x-axis. If you choose not to set up initial conditions, then the previous version of FOR061.DAT is used as an intial condition.

```
$ RUN STARDA

DO YOU WISH TO SET UP INITIAL CONDITIONS ?
1<< YES
0<<  NO
1

PROBLEM PARAMETERS


PLEASE ENTER REFERENCE PRESSURE
0.0
PLEASE ENTER REFERENCE DENSITY
1.0
PLEASE ENTER REFERENCE VELOCITY
1.0
PLEASE ENTER REFERENCE LENGTH
1.0
PLEASE ENTER THE NON-DIM. PSEUDO-COMPRESSIBILITY
0.1

IS THE PROBLEM ISOTHERMAL?
1<< YES
2<< NO
1

PLEASE ENTER THE PROBLEM REYNOLDS NUMBER
10.0
```

```
THE PROBLEM-PARAMETER  FILE HAS BEEN
CREATED

DO YOU WISH TO SPECIFY AN OUTFLOW PRESSURE ?
1<< YES
2<<  NO
1

PLEASE ENTER DIMENSIONAL
OUTFLOW PRESSURE
0.0

PLEASE SELECT THE TYPE OF VELOCITY PROFILE
1 <<  CONSTANT U ALONG Y (SLUG FLOW)
2 <<  VARIABLE U ALONG Y (PARABOLIC PROFILE)
2

PLEASE SELECT A PARABOLIC PROFILE
1 << SYMMETRIC
2 << ASYMMETRIC,MAX AT UPPER WALL
3 << ASYMMETRIC,MAX AT LOWER WALL
1

INLET FLOW DIRECTION PRESCRIBED ?
1<< YES
2<<  NO
1

INLET FLOW DIRECTION  ? (ANGLE IN DEGREES)
0.0
THE INITIAL-CONDITION FILES HAVE BEEN
CREATED
```

The time integration parameters are input next. These control solution of the algebraic equation system resulting from the finite element assembly. In addition to the θ-implicit single step algorithm, an implicit Runge-Kutta scheme can also be specified. For the problems presented in this study, the backward Euler specification ($\theta = 1.0$) was utilized.

```
DO YOU WISH TO SET UP TIME-INTEGRATION
 ALGORITHM PARAMETERS ?
1<< YES
0<<  NO
1
```

```
TYPE OF INTEGRATION ALGORITHM ?
1=EULER BACKWARD
2=CRANK-NICOLSON
3=RUNGE-KUTTA 87
4=RUNGE-KUTTA 88
1

ENTER NUMBER OF NEWTON ITERATIONS (>=2)
3
THE TIME INTEGRATION ALGORITHM PARAMETER
FILE HAS BEEN CREATED
```

The parameters controlling the time step are now requested. The user-defined "zero" is the maximum value of the algorithm residual which represents convergence during a Newton iteration. Hence, when $\{F\}$ is less than or equal to "zero" the converged solution has been obtained at that particular time station. The time-step variation parameter controls the size of the maximum variation allowed for the flow variables. A larger value may permit more frequent time step increases, if convergence is obtained.

```
DO YOU WISH TO SET UP TIME STEP
 VARIATION PARAMETERS ?
1<< YES
0<<  NO
1

INITIAL TIME ?
0.0
TIME MAX ?
10.0
TOTAL NUMBER OF STEPS ?
10
MINIMUM TIME STEP ? (< INITIAL DT !)
0.001
INITIAL TIME STEP ?
0.01
MAXIMUM TIME STEP ?
0.05
USER DEFINED ZERO ?
0.0001
DT-VARIATION PARAMETER (0.0< PAR <=1.0)
0.3
THE TIME STEP VARIATION PARAMETER FILE
 HAS BEEN CREATED
```

Artificial dissipation mechanism coefficients are requested next. Beta coefficients 1-4 refer to the dissipation parameters operable in the pressure, u-velocity, v-velocity, and energy equations, respectively. Although different beta values may be specified for the x and y coordinate directions, for the three benchmark problems presented in this thesis, the x and y beta values were set equal.

```
DISSIPATION REQUESTED ?
1=YES
0=NO
1
PLEASE ENTER X-BETA COEFFICIENT #              1
0.1
PLEASE ENTER Y-BETA COEFFICIENT #              1
0.1
PLEASE ENTER X-BETA COEFFICIENT #              2
0.0
PLEASE ENTER Y-BETA COEFFICIENT #              2
0.0
PLEASE ENTER X-BETA COEFFICIENT #              3
0.0
PLEASE ENTER Y-BETA COEFFICIENT #              3
0.0
PLEASE ENTER X-BETA COEFFICIENT #              4
0.0
PLEASE ENTER Y-BETA COEFFICIENT #              4
0.0
THE DISSIPATION MECHANISM PARAMETER FILE
 HAS BEEN CREATED
```

The boundary condition information is the final sequence and is input by boundary segments. The referenced segments , however, are not necessarily the same as those used in the mesh generator. For the step wall diffuser, eight boundary condition segments are required: four boundary corners, two wall boundaries, one inlet, and one outlet. As previously mentioned, the step is "blocked out" in the subprogram TSGEN, described in the next section.

```
DO YOU WISH TO SET-UP BOUNDARY-CONDITION INFORMATION ?
1<< YES
0<<  NO
1

 BOUNDARY CONDITIONS

HOW MANY BOUNDARY SECTIONS ? (<=20)
8
          8 SECTIONS
NOW ENTER:  1 ,TO CONFIRM
1

SECTION                    1
 TYPE OF BOUNDARY ?
1<< INFLOW   BOUNDARY
2<< INFLOW   BOUNDARY CORNER
3<< OUTFLOW BOUNDARY CORNER
4<< OUTFLOW BOUNDARY
5<< WALL     BOUNDARY
2
THE TYPE SELECTED IS                2
NOW ENTER:  1 ,TO CONFIRM
1

SECTION                    1
 TYPE OF PRESCRIBED BC ?
1<<   du/dn=0 & dv/dn=0
2<<   p & du/dn=0 & dv/dn=0
3<<   u & dv/dn=0
4<<   v & du/dn=0
5<<   p & u & dv/dn=0
6<<   p & v & du/dn=0 & T
7<<   u & v
8<<   p & u & v & T
16<<  u & v & T
7
THE TYPE SELECTED IS                7
NOW ENTER:  1 ,TO CONFIRM
1

FIRST
EXTREME NODE OF SECTION                1
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
1
ELEMENT COLUMN ?
1
NODE LOCAL NUMBER ?
1
ELEMENT                1 NODE              1
NODE LOCAL NUMBER:                1
ROW              1 COLUMN              1
```

```
NOW ENTER:  1 ,TO CONFIRM
1


SECOND
EXTREME NODE OF SECTION             1
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
1
ELEMENT COLUMN ?
1
NODE LOCAL NUMBER ?
1
ELEMENT              1 NODE           1
NODE LOCAL NUMBER:               1
ROW              1 COLUMN            1
NOW ENTER:  1 ,TO CONFIRM
1


SECTION               2
 TYPE OF BOUNDARY ?
1<< INFLOW  BOUNDARY
2<< INFLOW  BOUNDARY CORNER
3<< OUTFLOW BOUNDARY CORNER
4<< OUTFLOW BOUNDARY
5<< WALL    BOUNDARY
5
THE TYPE SELECTED IS             5
NOW ENTER:  1 ,TO CONFIRM
1
SECTION               2
 TYPE OF PRESCRIBED BC ?
1<<   du/dn=0 & dv/dn=0
2<<   p & du/dn=0 & dv/dn=0
3<<   u & dv/dn=0
4<<   v & du/dn=0
5<<   p & u & dv/dn=0
6<<   p & v & du/dn=0 & T
7<<   u & v
8<<   p & u & v & T
16<<  u & v & T
7
THE TYPE SELECTED IS             7
NOW ENTER:  1 ,TO CONFIRM
1

FIRST
EXTREME NODE OF SECTION             2
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
1
```

```
ELEMENT COLUMN ?
1
NODE LOCAL NUMBER ?
2
ELEMENT            1 NODE            22
NODE LOCAL NUMBER:             2
ROW            1 COLUMN           1
NOW ENTER:  1 ,TO CONFIRM
1


SECOND
EXTREME NODE OF SECTION           2
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
1
ELEMENT COLUMN ?
37
NODE LOCAL NUMBER ?
1
ELEMENT          721 NODE           757
NODE LOCAL NUMBER:             1
ROW            1 COLUMN          37
NOW ENTER:  1 ,TO CONFIRM
1


SECTION              3
 TYPE OF BOUNDARY ?
1<< INFLOW  BOUNDARY
2<< INFLOW  BOUNDARY CORNER
3<< OUTFLOW BOUNDARY CORNER
4<< OUTFLOW BOUNDARY
5<< WALL    BOUNDARY
3
THE TYPE SELECTED IS           3
NOW ENTER:  1 ,TO CONFIRM
1

SECTION              3
 TYPE OF PRESCRIBED BC ?
1<<   du/dn=0 & dv/dn=0
2<<   p & du/dn=0 & dv/dn=0
3<<   u & dv/dn=0
4<<   v & du/dn=0
5<<   p & u & dv/dn=0
6<<   p & v & du/dn=0 & T
7<<   u & v
8<<   p & u & v & T
16<<  u & v & T
8
THE TYPE SELECTED IS           8
NOW ENTER:  1 ,TO CONFIRM
```

```
1

FIRST
EXTREME NODE OF SECTION              3
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
1
ELEMENT COLUMN ?
37
NODE LOCAL NUMBER ?
2
ELEMENT          721 NODE           778
NODE LOCAL NUMBER:             2
ROW            1 COLUMN            37
NOW ENTER:  1 ,TO CONFIRM
1


SECOND
EXTREME NODE OF SECTION              3
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
1
ELEMENT COLUMN ?
37
NODE LOCAL NUMBER ?
2
ELEMENT          721 NODE           778
NODE LOCAL NUMBER:             2
ROW            1 COLUMN            37
NOW ENTER:  1 ,TO CONFIRM
1


SECTION              4
 TYPE OF BOUNDARY ?
1<< INFLOW  BOUNDARY
2<< INFLOW  BOUNDARY CORNER
3<< OUTFLOW BOUNDARY CORNER
4<< OUTFLOW BOUNDARY
5<< WALL    BOUNDARY
4
THE TYPE SELECTED IS            4
NOW ENTER:  1 ,TO CONFIRM
1

SECTION              4
 TYPE OF PRESCRIBED BC ?
1<<   du/dn=0 & dv/dn=0
2<<   p & du/dn=0 & dv/dn=0
3<<   u & dv/dn=0
4<<   v & du/dn=0
```

```
5<<    p & u & dv/dn=0
6<<    p & v & du/dn=0 & T
7<<    u & v
8<<    p & u & v & T
16<<   u & v & T
2
THE TYPE SELECTED IS            2
NOW ENTER:  1 ,TO CONFIRM
1

FIRST
EXTREME NODE OF SECTION            4
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
1
ELEMENT COLUMN ?
37
NODE LOCAL NUMBER ?
3
ELEMENT          721 NODE          779
NODE LOCAL NUMBER:            3
ROW             1 COLUMN          37
NOW ENTER:  1 ,TO CONFIRM
1


SECOND
EXTREME NODE OF SECTION            4
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
20
ELEMENT COLUMN ?
37
NODE LOCAL NUMBER ?
2
ELEMENT          740 NODE          797
NODE LOCAL NUMBER:            2
ROW            20 COLUMN          37
NOW ENTER:  1 ,TO CONFIRM
1


SECTION              5
 TYPE OF BOUNDARY ?
1<< INFLOW  BOUNDARY
2<< INFLOW  BOUNDARY CORNER
3<< OUTFLOW BOUNDARY CORNER
4<< OUTFLOW BOUNDARY
5<< WALL    BOUNDARY
3
THE TYPE SELECTED IS            3
NOW ENTER:  1 ,TO CONFIRM
```

```
1

SECTION                  5
 TYPE OF PRESCRIBED BC ?
1<<   du/dn=0 & dv/dn=0
2<<   p & du/dn=0 & dv/dn=0
3<<   u & dv/dn=0
4<<   v & du/dn=0
5<<   p & u & dv/dn=0
6<<   p & v & du/dn=0 & T
7<<   u & v
8<<   p & u & v & T
16<<  u & v & T
8
THE TYPE SELECTED IS             8
NOW ENTER:  1 ,TO CONFIRM
1

FIRST
EXTREME NODE OF SECTION              5
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
20
ELEMENT COLUMN ?
37
NODE LOCAL NUMBER ?
3
ELEMENT              740 NODE          798
NODE LOCAL NUMBER:              3
ROW             20 COLUMN             37
NOW ENTER:  1 ,TO CONFIRM
1


SECOND
EXTREME NODE OF SECTION              5
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
20
ELEMENT COLUMN ?
37
NODE LOCAL NUMBER ?
3
ELEMENT              740 NODE          798
NODE LOCAL NUMBER:              3
ROW             20 COLUMN             37
NOW ENTER:  1 ,TO CONFIRM
1


SECTION                  6
```

```
 TYPE OF BOUNDARY ?
1<< INFLOW   BOUNDARY
2<< INFLOW   BOUNDARY CORNER
3<< OUTFLOW BOUNDARY CORNER
4<< OUTFLOW BOUNDARY
5<< WALL     BOUNDARY
5
THE TYPE SELECTED IS                5
NOW ENTER:  1 ,TO CONFIRM
1

SECTION                  6
 TYPE OF PRESCRIBED BC ?
1<<    du/dn=0 & dv/dn=0
2<<    p & du/dn=0 & dv/dn=0
3<<    u & dv/dn=0
4<<    v & du/dn=0
5<<    p & u & dv/dn=0
6<<    p & v & du/dn=0 & T
7<<    u & v
8<<    p & u & v & T
16<<   u & v & T
7
THE TYPE SELECTED IS                7
NOW ENTER:  1 ,TO CONFIRM
1

FIRST
EXTREME NODE OF SECTION                  6
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
20
ELEMENT COLUMN ?
37
NODE LOCAL NUMBER ?
4
ELEMENT              740 NODE            777
NODE LOCAL NUMBER:                 4
ROW              20 COLUMN             37
NOW ENTER:  1 ,TO CONFIRM
1


SECOND
EXTREME NODE OF SECTION                  6
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
20
ELEMENT COLUMN ?
1
NODE LOCAL NUMBER ?
3
```

```
ELEMENT              20 NODE              42
NODE LOCAL NUMBER:                  3
ROW            20 COLUMN                 1
NOW ENTER:  1 ,TO CONFIRM
1


SECTION                  7
 TYPE OF BOUNDARY ?
1<< INFLOW  BOUNDARY
2<< INFLOW  BOUNDARY CORNER
3<< OUTFLOW BOUNDARY CORNER
4<< OUTFLOW BOUNDARY
5<< WALL    BOUNDARY
2
THE TYPE SELECTED IS                2
NOW ENTER:  1 ,TO CONFIRM
1

SECTION                  7
 TYPE OF PRESCRIBED BC ?
1<<   du/dn=0 & dv/dn=0
2<<   p & du/dn=0 & dv/dn=0
3<<   u & dv/dn=0
4<<   v & du/dn=0
5<<   p & u & dv/dn=0
6<<   p & v & du/dn=0 & T
7<<   u & v
8<<   p & u & v & T
16<<  u & v & T
7
THE TYPE SELECTED IS                7
NOW ENTER:  1 ,TO CONFIRM
1

FIRST
EXTREME NODE OF SECTION                  7
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
20
ELEMENT COLUMN ?
1
NODE LOCAL NUMBER ?
4
ELEMENT              20 NODE              21
NODE LOCAL NUMBER:                  4
ROW            20 COLUMN                 1
NOW ENTER:  1 ,TO CONFIRM
1


SECOND
EXTREME NODE OF SECTION                  7
```

```
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
20
ELEMENT COLUMN ?
1
NODE LOCAL NUMBER ?
4
ELEMENT            20 NODE           21
NODE LOCAL NUMBER:            4
ROW            20 COLUMN           1
NOW ENTER:  1 ,TO CONFIRM
1

SECTION            8
 TYPE OF BOUNDARY ?
1<< INFLOW  BOUNDARY
2<< INFLOW  BOUNDARY CORNER
3<< OUTFLOW BOUNDARY CORNER
4<< OUTFLOW BOUNDARY
5<< WALL    BOUNDARY
1
THE TYPE SELECTED IS           1
NOW ENTER:  1 ,TO CONFIRM
1

SECTION            8
 TYPE OF PRESCRIBED BC ?
1<<   du/dn=0 & dv/dn=0
2<<   p & du/dn=0 & dv/dn=0
3<<   u & dv/dn=0
4<<   v & du/dn=0
5<<   p & u & dv/dn=0
6<<   p & v & du/dn=0 & T
7<<   u & v
8<<   p & u & v & T
16<<  u & v & T
7
THE TYPE SELECTED IS           7
NOW ENTER:  1 ,TO CONFIRM
1

FIRST
EXTREME NODE OF SECTION           8
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
20
ELEMENT COLUMN ?
1
NODE LOCAL NUMBER ?
1
ELEMENT            20 NODE           20
NODE LOCAL NUMBER:            1
```

```
ROW            20 COLUMN           1
NOW ENTER:  1 ,TO CONFIRM
1


SECOND
EXTREME NODE OF SECTION            8
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
1
ELEMENT COLUMN ?
1
NODE LOCAL NUMBER ?
4
ELEMENT              1 NODE            2
NODE LOCAL NUMBER:             4
ROW             1 COLUMN           1
NOW ENTER:  1 ,TO CONFIRM
1

THE BOUNDARY-CONDITION CODE FILE HAS BEEN
CREATED
FORTRAN STOP
```

TSGEN – The Training Structure Routine

TSGEN is the training structure generator which inserts flow blockage geometries. It automatically inserts proper boundary conditions necessary to simulate flow obstructions. MESH and STARDA are first run as if there were no obstructions in the flow. Then, TSGEN is run prior to running PSEUDO. In the case of the step wall diffuser, it is necessary to run TSGEN to create the step at the inlet, see Figure C.1.

The first request is for the total number of training structures desired. The choice of a viscous or inviscid simulation is required to implement proper boundary conditions. Then the lower left and upper right-most elements of the rectangular training structure are entered as initial and final elements, see Figure C.1. The variables at nodes interior to the obstacle are automatically set to nought, while boundary conditions along the boundary of the training structure are handled by modification of the node codes in FOR068.DAT.

```
$ RUN TSGEN
ENTER NUMBER OF TRAINING STRUCTURES
1

SELECT TYPE OF FLOW
1<< INVISCID  FLOW
2<<  VISCOUS  FLOW
2
TRAINING-STRUCTURE
TYPE AND LOCATION INFORMATION

TRAINING STRUCTURE #                   1
PLEASE SELECT A TYPE:
1<<TRIANGULAR,POSITIVE SLOPE,
   LOWER-BANK
2<<TRIANGULAR,POSITIVE SLOPE,
   UPPER-BANK
```

```
3<<TRIANGULAR,NEGATIVE SLOPE,
   LOWER-BANK
4<<TRIANGULAR,NEGATIVE SLOPE,
   UPPPER-BANK
5<<RECTANGULAR,LOWER-BOUNDARY BOTTOM
6<<RECTANGULAR,UPPER-BOUNDARY TOP
5

SPECIFY THE ELEMENT COORDINATES
THE INITIAL AND FINAL ELEMENTS
OF THE TRAINING STRUCTURE.


INITIAL ELEMENT:
ENTER ELEMENT ROW:
1

ENTER ELEMENT COLUMN :
1

FINAL   ELEMENT:
ENTER ELEMENT ROW:
10
ENTER ELEMENT COLUMN :
6

DO YOU WISH TO REPLACE THE EXISTING NODE CODES ?
1<< YES
2<<  NO
1
FORTRAN STOP
```

PSEUDO: The Finite Element Solving Routine

PSEUDO is the main finite element algorithm code. All necessary data files have been prepared by the previous three programs. The output files of this program are FOR070.DAT, the time parameter update, FOR071.DAT, the solution array, and FOR072.DAT, the progress report and final solution printout.

FOR072.DAT contains an integer map of the computational domain information, the boundary condition and initial condition information, problem parameters, time integration information, the residuals at each time step, along with the norms and temporal information at each time step. The final solution is then output after the last time step.

FOR071.DAT contains the final solution of the problem variables. This file must be renamed FOR061.DAT (the initial conditions) for any further runs of the same problem. Also, to begin subsequent runs at the final time step of the previous run, FOR070.DAT should be renamed FOR067.DAT (to update the time parameters).

VITA

Fred Dean Carter, Jr. was born in Boliver, Tennessee on September 18, 1964. He graduated from Bearden High School in Knoxville, Tennessee in June 1982. The following September he entered the University of Tennessee, Knoxville, and in March 1987 he received a Bachelor of Science Degree in Agricultural Engineering.

In September 1987, he entered the Graduate School of the University of Tennessee, Knoxville, to pursue a Master of Science Degree in Engineering Science, with an emphasis in Computational Fluid Dynamics. That degree was awarded in May 1990.

The author is a member of Phi Kappa Phi and Gamma Sigma Delta Honor Societies.