To the Graduate Council:

I am submitting herewith a thesis written by Atish Jyoti Mitra entitled "On Performance of A Pressure-Projection Algorithm for Incompressible Navier-Stokes." 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, Major Professor

We have read this thesis
and recommend its acceptance:

Accepted for the Council:

Associate Vice Chancellor
and Dean of the Graduate School

# ON PERFORMANCE OF A PRESSURE-PROJECTION ALGORITHM FOR INCOMPRESSIBLE NAVIER STOKES

A Thesis

Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Atish J Mitra

May 2000

## Acknowledgment

I am indebted to Dr. A.J Baker, whose supervision has been a perfect blend of advice, encouragement and indulgence. In the period I have spent as a member of his group, he has emerged as an inspiring model in his teaching and his enthusiasm for research. I am also grateful to my committee members Dr. J. Iannelli and Dr. M. Parang for their useful comments on my thesis.

Many other people, specifically the members of the CFD laboratory at UT, provided a friendly and stimulating environment at work Joe Orzechowski helped me a lot with understanding and assimilating aPSE, and my collegues Alexy, Zac and Mike helped with Latex. Nina at the computational laboratory has been patient with my daily queries on unix operations.

My parents have been specially supportive over the years, specially in encouraging me to do what I liked

## Abstract

A time accurate implicit Galerkin finite element algorithm for the incompressible Navier Stokes equations is assessed for convergence and conservation properties. The algorithm inherits it's philosophy from the group of algorithms known as Pressure Projection methods. The implementation of such a method typically employs numerical linear algebra procedures that are non-Newton, which strongly affects algorithm convergence, hence conservation properties Using sparse matrix methods, a full Newton process is acheived for solving the two-dimensional thermal cavity benchmark problem at large Rayleigh number.

The Newton algorithm performance, fully coupling the velocity, temperature and continuity constraint function degrees of freedom, is clearly superior Sobolev norms of these variables are used as primary indicators of solution convergence. Time evolution of these norms, along with the norm of the physically motivated pressure Poission equation solution, provides ample insight into the superior performance and conservation properties compared to the common segregated quasi-Newton procedure. The results of this study generate benchmark quality solutions for Rayleigh numbers ranging from $10^6$ to $10^8$

# Contents


**1 INTRODUCTION** **2**

1.1 Key Points of Research . . . . . 3

**2 PRIOR RESEARCH** **4**

2 1 Review of Earlier Work on INS Algorithms . . . . . 4

2.2 The Continuity Constraint Algorithm . . . . . 4

**3 THEORETICAL DEVELOPEMENT** **6**

3 1 Governing Equations . . . . . 6

3.2 Weak Statement Formulation . . . . . 8

3 3 Algorithm Structure and Linear Algebra . . . . . 11

3.4 Error Measures . . . . . 13

3.5 Computational Platform . . . . . 13

3 5.1 Software . . . . . 13

3.5.2 Hardware . . . . . 14

**4 A BENCHMARK** **15**

4.1 The Problem - Thermal Cavity . . . . . 15

4.2 Past Research . . . . . 15

4.2.1 Computational Work . . . . . 15

4.2.2 Experimental Work . . . . . 16

4.3 Key Issues of Solution . . . . . 16

4.3.1 Non-Dimensionalization Issues . . . . . 16

4.3.2 Time Evolution of Norms . . . . . 17

4.3.3 Solution "Steady State" . . . . . 17

4.3.4 The "Residual Divergence" . . . . . 17

4.3.5 A comment on solution strategy . . . . . 18

4.4 Algorithm Performance . . . . . 18

4.4.1 Convergence issues . . . . . 18

4.4.2 Time Stepping . . . . . 19

4.4.3 Conservation issues . . . . . 20

4.5 Results . . . . . 21

**5 SUMMARY AND CONCLUSIONS** **28**

**Bibliography** **30**

**Appendices** **33**

**A Input Files** **34**

A.0.1 Model File . . . . . 35

A 0.2 Template File . . . . . 38

**B Tables** **47**

**C Figures** **51**

**Vita** **93**

# List of Tables

B.1 Benchmark Numerical Data, Ra=1,000,000 . . . . . 48

B.2 Benchmark Numerical Data, Ra=10,000,000 . . . . . 49

B.3 Benchmark Numerical Data, Ra=100,000,000 . . . . . 50

# List of Figures

C.1 Time evolution of Norms (segregated jacobian), 32 mesh, Ra=$10^6$ . . . . 52

C.2 Time evolution of Norms, 32 mesh, Ra=$10^6$ . . . . . 53

C.3 Time evolution of Norms, 64 mesh, Ra=$10^7$ . . . . . 54

C.4 Time evolution of Norms, 64 mesh, Ra=$10^8$ . . . . . 55

C.5 Optimal mesh Progression Ratio, Ra=$10^6$ . . . . . 56

C.6 Optimal mesh Progression Ratio, Ra=$10^7$ . . . . . 57

C.7 Optimal mesh Progression Ratio, Ra=$10^8$ . . . . . 58

C.8 Temperature contour, Ra=$10^7$, 32x32 mesh . . . . . 59

C.9 Pressure contour, Ra=$10^7$, 32x32 mesh . . . . . 60

C.10 Vorticity contour, Ra=$10^7$, 32x32 mesh . . . . . 61

C.11 Streamlines, Ra=$10^7$, 32x32 mesh . . . . . 62

C.12 Velocity Field, Ra=$10^7$, 32x32 mesh . . . . . 63

C.13 Temperature contour, Ra=$10^7$, 64x64 mesh . . . . . 64

C.14 Pressure contour, Ra=$10^7$, 64x64 mesh . . . . . 65

C.15 Vorticity contour, Ra=$10^7$, 64x64 mesh . . . . . 66

C.16 Streamlines, Ra=$10^7$, 64x64 mesh . . . . . 67

C.17 Velocity Field, Ra=$10^7$, 64x64 mesh . . . . . 68

C.18 Temperature contour, Ra=$10^8$, 64x64 mesh . . . . . 69

C.19 Pressure contour, Ra=$10^8$, 64x64 mesh . . . . . 70

C.20 Vorticity contour, Ra=$10^8$, 64x64 mesh . . . . . 71

C.21 Streamlines, Ra=$10^8$, 64x64 mesh . . . . . 72

C.22 Velocity Field, Ra=$10^8$, 64x64 mesh . . . . . 73

C.23 Temperature contour, Ra=$10^8$, 128x128 mesh . . . . . 74

C.24 Pressure contour, Ra=$10^8$, 128x128 mesh . . . . . 75

C.25 Vorticity contour, Ra=$10^8$, 128x128 mesh . . . . . 76

C.26 Streamlines, Ra=$10^8$, 128x128 mesh . . . . . 77

C.27 Velocity Field, Ra=$10^8$, 128x128 mesh . . . . . 78

C 28 Streamlines (X1=0.25,X2=0.75), Ra=$10^8$, 128x128 mesh . . . . . 79

C.29 Streamlines (X1=0.35,X2=0.75), Ra=$10^8$, 128x128 mesh . . . . . 80

C.30 Streamlines (X1=0.75,X2=0.25), Ra=$10^8$, 128x128 mesh . . . . . 81

C.31 Streamlines (X1=0.75,X2=0.45), Ra=$10^8$, 128x128 mesh . . . . . 82

C 32 Streamlines (X1=0.75,X2=0.08), Ra=$10^8$, 128x128 mesh . . . . . 83

C 33 Streamlines (X1=0.85,X2=0 175), Ra=$10^8$, 128x128 mesh . . . . . 84

C.34 Velocity Field (X1=0.15,X2=0 85), Ra=$10^8$, 128x128 mesh . . . . . 85

C.35 Velocity Field (X1=0.1,X2=0.75), Ra=$10^8$, 128x128 mesh . . . . . 86

C.36 Velocity Field (X1=0.25,X2=0.9), Ra=$10^8$, 128x128 mesh . . . . . 87

C.37 Velocity Field (X1=0.4,X2=0.85), Ra=$10^8$, 128x128 mesh . . . . . 88

C 38 Velocity Field (X1=0.37,X2=0.97), Ra=$10^8$, 128x128 mesh . . . . . 89

C 39 Velocity Field (X1=0.9,X2=0.35), Ra=$10^8$, 128x128 mesh . . . . . 90

C.40 Verification of convergence, Ra=$10^6$ . . . . . 91

C 41 Verification of convergence, Ra=$10^8$ . . . . . 92

**Nomenclature**

| | |
|---|---|
| Ar | Archimedes number |
| $f_j$ | kinematic flux vector |
| $f_j^v$ | viscous flux vector |
| $\{FQ\}$ | Newton residual vector |
| g | acceleration due to gravity |
| k | thermal conductivity, iteration index |
| M | mass matrix |
| $n_i$ | boundary surface normal unit vector |
| $\{N\}$ | vector of element basis functions |
| p | pressure |
| P | kinematic pressure |
| Pr | Prandtl number |
| q | state variable |
| $\{Q\}$ | discrete approximation of state variables |
| Ra | Rayleigh number |
| Re | Reynolds number |
| $Re^t$ | turbulent Reynolds number |
| $\{R\}$ | residual vector |
| s | source term |
| t | time |
| T | temperature |
| $x_i$ | spatial coordinate |
| $\alpha$ | thermal diffusivity |
| $\beta$ | coefficient of volume expansion |
| $\delta_{ij}$ | Kroneckar delta |
| $\Delta t$ | time step |
| $\epsilon$ | convergence tolerance |
| $\eta_h$ | transform coordinates |
| $\Theta$ | dimensionless potential temperature |
| $\kappa$ | bulk viscosity |
| $\mu$ | dynamic viscosity |
| $\nu$ | kinematic viscosity |
| $\rho$ | density |
| $\Phi$ | continuity constraint function |
| $\psi$ | streamfunction |
| $\Psi$ | approximation trial space |
| $\Omega$ | problem domain |
| $\partial\Omega$ | problem boundary |

# Chapter 1

# INTRODUCTION

Incompressible thermal fluid flow continues to remain an important area of interest to engineers and scientists This is not surprising considering the extent of applications of the theory of incompressible fluids and isochoric flows in practical engineering. The incompressibility condition has proven a very practical and useful assumption for a large range of flows. Hence the interest in primitive-variable CFD formulations continues, as manifest in the continued study of the direct computational analysis of the Reynolds-averaged forms of the Navier-Stokes equations.

The objective of this research is to generate and implement an improved version of the Continuity Constraint Algorithm [1], as developed and implemented in a quasi-Newton construction in the CFD laboratory of the University of Tennessee in the early nineties.

## 1.1 Key Points of Research

As original contributions to the study of incompressible CFD, this presents ·

- A modification of the Continuity Constraint Algorithm by incorporating the continuity constraint function, $\Phi$ , into the jacobian, yielding a genuine Newton algorithm.

- An implementation of the resulting modified CCA using a finite element semi-discretization of a Galerkin Weak Statement

- A "benchmarking" of the improved algorithm performance against existing numerical results for a classic benchmark - the thermal cavity.

- Extension of the existing benchmark to higher previously unattainable Rayleigh numbers, hence extending the available benchmark problem class

# Chapter 2

# PRIOR RESEARCH

## 2.1 Review of Earlier Work on INS Algorithms

An extensive and comprehensive review of CFD algorithms for Incompressible flows has been compiled by P T. Williams and A.J. Baker [2]. This publication identifies the solenoidal kinematic constraint $\nabla \cdot \mathbf{u} = \mathbf{0}$ as the basis of all such algorithms, and presents a taxonomy of established algorithms.

## 2.2 The Continuity Constraint Algorithm

The roots of the general class of pressure-relaxation algorithms [2] can be traced back to the finite-difference SMAC method, to Imperial College (UK) research in the late sixties, and also to developments in research in Incompressible Navier-Stokes Algorithms in the CFD laboratory of the University of Tennessee in the late eightees

The incompressibility assumption bestows upon the pressure the dual roles of a Lagrange multiplier, enforcing an isochoric constraint on the flow field, at the same time acting as a dynamical force variable in the momentum equations. This had been recognised and utilised in the developement of the Continuity Constraint Algorithm by Williams [1].

# Chapter 3

# THEORETICAL DEVELOPEMENT

## 3.1 Governing Equations

The conservation law system under study can be represented by a coupled, non-linear set of partial differential equations.

$$\mathcal{L}(q) = \frac{\partial q}{\partial t} + \frac{\partial (f_j - f_j^v)}{\partial x_j} - s = 0 \tag{3.1}$$

The system is defined on the domain $\Omega \times t$ , q thereby being a function of position $(x_j)$ and time(t).

In the coupled Navier-Stokes equations formulation in question, and assuming the Boussinesq buoyancy body force, the state variable q and the kinematic flux vector $f_j$

and viscous flow vector $f_j^v$ have the following forms.

$$q = \begin{Bmatrix} u1 \\ u2 \\ u3 \\ \Theta \\ 0 \end{Bmatrix} \tag{3.2}$$

$$f_j = \begin{Bmatrix} u_j u_1 + p\delta_{1j} \\ u_j u_2 + p\delta_{2j} \\ u_j u_3 + p\delta_{3j} \\ u_j \Theta \\ 0 \end{Bmatrix} \tag{3.3}$$

$$f_j^v = \begin{Bmatrix} \frac{1+Re^t}{Re}\frac{\partial u_1}{\partial x_j} \\ \frac{1+Re^t}{Re}\frac{\partial u_2}{\partial x_j} \\ \frac{1+Re^t}{Re}\frac{\partial u_3}{\partial x_j} \\ \frac{1}{Re}\left(\frac{1}{Pr} + \frac{Re^t}{Pr^t}\right)\frac{\partial \Theta}{\partial x_j} \\ \frac{\partial \Phi}{\partial x_j} \end{Bmatrix} \tag{3 4}$$

$$S = \left\{ \begin{array}{c} -A_r \Theta g_1 \\ -A_r \Theta g_2 \\ -A_r \Theta g_3 \\ s_\Theta \\ \nabla \cdot \tilde{u} \end{array} \right\} \tag{3 5}$$

An important point to note in the above formulation is that $\Phi$ is a mathematically motivated function serving as an enforcer of mass conservation in the system. It's action in conserving mass is analogous to that of a pressure-like term in the system (3.1).

On the other hand, the genuine (kinematic) pressure is obtained from a pressure Poisson equation established in the form

$$\mathcal{L}(\mathrm{P}) = \nabla^2 \mathrm{P} - \nabla \cdot (\mathcal{L}(\mathrm{u}) - \nabla \mathrm{P}) = 0, \mathrm{with} BCs \tag{3.6}$$

## 3.2 Weak Statement Formulation

The problem defined by the nonlinear coupled PDE set along with the established-boundary conditions constitutes a well posed problem. In its equivalent weak form, the mathematical flexibility and robustness is enhanced by the resultant widening of the integrability conditions. The problem is reduced to seeking a solution $u_i^N(x_j, t)$, $\Theta^N(x_j, t)$,$\Phi^N(x_j, t)$,$P^N(x_j, t)$ as expansions onto a trial space $\Psi_j(x_i)$ such that

$$\int w(x_i,t)\mathcal{L}(\mathrm{q^N})\mathrm{d}\tau = 0, \forall \mathrm{w} \tag{3 7}$$

The requirements of the solution spaces to ensure a well posed weak statement is fully developed in [3]

A continuum approximation assumes the separability of space and time. One further assumes the test function in (3.7) can be accurately interpolated.

$$w^M(x_j,t) = \sum \Phi_i(x_j)W_i(t) \tag{3 8}$$

Substituting (3.8) and forming the extremum yields

$$WS^N = \frac{\partial}{\partial W_i}\int_\Omega w^M(x_j,t)\mathcal{L}(\mathrm{q^N})\mathrm{d}\tau = \int_\Omega \Phi_{\mathrm{i}}(\mathrm{x_j})\mathcal{L}(\mathrm{q^N})\mathrm{d}\tau = 0, \forall \mathrm{i} \tag{3 9}$$

The *Galerkin Weak Statement* is the mathematically optimal choice of the test function set to be identical to the trial space ([4])· hence

$$GWS^N = \int_\Omega \Psi_i(x_j)\mathcal{L}(\mathrm{q^N})\mathrm{d}\tau = 0 \tag{3.10}$$

For (3.1), the $GWS^N$ is therefore

$$\int_\Omega \Psi_i(\frac{\partial q^N}{\partial t} + \frac{\partial}{\partial x_j}(f_j - f_j^v)^N - s^N)d\tau = 0 \tag{3.11}$$

Using the Green-Gauss theorem the divergence operator is projected from the flux vectors $f_j$ and $f_j^v$ to the approximation trial space $\Psi_i$, and a surface integral is produced as

$$\int_\Omega \Psi_i(\frac{\partial q^N}{\partial t} - s^N)d\tau - \int_\Omega \frac{\partial \Psi_i}{\partial x_j}(f_j - f_j^v)^N d\tau + \oint_{\partial\Omega_e \cup \partial\Omega} \Psi_i(f_j - f_j^v)\hat{n}_j d\tau = 0 \quad (3.12)$$

that is a placeholder for all *natural* boundary conditions in the $GWS^N$.

The finite-element method uses a spatial semi-discretization $\Omega^h$ of the continuum domain $\Omega$, made up of the union of a set of disjoint subdomains $\Omega_e$ or finite elements, such that

$$\Omega \approx \Omega^h \equiv \cup_e \Omega_e \quad (3\ 13)$$

The approximation $q^N$ is then formed as $q^h$, the union of finite-element approximations $q_e$ on $\Omega_e$

$$q(x_j, t) \approx q^N(x_j, t) \equiv q^h(x_j, t) = \cup_e q_e(x_j, t) \quad (3.14)$$

On any finite-element domain $\Omega_e$, the generic form for $q_e$ is

$$q_e(x_j, t) \equiv N_k(\eta_j)^T Q(t) \quad (3.15)$$

where each element in the row vector ${N_k}^T$ (called the trial space finite-element basis set) is a k-th degree polynomial. There are as many of these polynomials as there are nodal degrees of freedom in $\Omega_e$

The indicated integrals in (3.12) are evaluated at the element level, and the resulting expressions are assembled into a global matrix statement of the form

$$GWS^h = [M]\frac{d\{Q\}}{dt} + R(Q) = 0 \tag{3.16}$$

The *time marching* algorithm selected for the formulation is $\theta$ -implicit Euler, and its use produces

$$\{FQ\} = [M]\{Q_{n+1} - Q_n\} + \Delta t(\theta\{R\}_{n+1} + (1-\theta)\{R\}_n) = \{0\} \tag{3.17}$$

as the algorithm terminal algebraic equation system.

The pressure Poisson equation and the $\Phi$-constraint equation , through their $GWS^N$, each produce the algebraic system

$$\{FQ_P\} = [D]\{Q_P\} - \{S_P(Q(t))\} \tag{3.18}$$

where $\{Q_p\}$ denotes Poission state variable members.

Summing up, the procedure described above converts the problem into a nonlinear coupled matrix statement, which is herein solved iteratively by a new Newton algorithm.

## 3.3 Algorithm Structure and Linear Algebra

The theoretical developments outlined above produce two distinct processes that are to be implemented in parallel. One is the $\theta$-implicit momentum-energy-phi system, and the other is the Pressure Poisson equation system.

The present algorithm treats $\Phi$ as a *mathematically motivated* function that is integrated into the main primitive variable Newton algorithm jacobian The pressure

Poisson equation solution on the other hand, is theoretically predicted to be a postprocessing equation in evaluating the actual pressure.

The philosophy of the algorithm is outlined below .

1. initialize the state variable at time step **n+1** from the values obtained from time step **n**

2. solve the momentum-energy-$\Phi$ system for $u^p$, $\Theta^p$, $\Phi^p$

3 update the constraint function as $C_{n+1}^{p+1} = C_{n+1}^1 + (\frac{1}{\theta \Delta t}) \sum_k^p \Phi^k$

4. repeat steps 2 - 3 till $\|\Phi^p\|_E \leq \epsilon$, $\epsilon$ being the desired tolerence

5. converge the Newton iteration process and update solution

6. solve the pressure Poisson equation for $P^{n+1}$

7. solve for vorticity and streamfunction as *post-processing* operation

8. evaluate the *residual divergence* $div_R = \nabla \cdot \tilde{u}$ as an *a posteriori* error estimate

9. advance time step and return to 1

The distinctive feature of the modification to the Continuity Constraint Algorithm described earlier is the incorporation of the continuity constraint function $\Phi$ into the jacobian The original CCA, which used a $\Phi$-segregated jacobian, enjoyed a convergence rate which was only quasi-Newton Coupling the jacobian is done here with the aim of obtaining better convergence characteristics, hence enforcement of the continuity constraint, while enjoying (possibly) a genuine Newton convergence rate.

## 3.4 Error Measures

Measures of solution errors are the energy semi-norms [4, 12] of the state variables As the "stopping-test" of the iterations, the energy semi-norms of the variable q has been considered.

$$\|q^p\|_E = \frac{\frac{1}{2}\int_\Omega \frac{\partial q}{\partial x_j}\frac{\partial q}{\partial x_j} d\Omega}{\Omega^h} \tag{3 19}$$

Energy semi-norms of all variables phi, velocity, pressure and temperature provide useful insight into the solution convergence and mesh optimality. Time evolutions of such norms have been used extensively in this research to gain insight into the progress of solution

## 3.5 Computational Platform

### 3.5.1 Software

**aPSE** , a finite element engineering software under developement at the UT CFD laboratory, is used for this research. aPSE is a template based problem solving environment that has been developed with the philosophy of keeping programming issues like data management and numerical analysis separate from the problem specific algorithm theoretical issues.

**aPSE system shell organisation**

1. JOBFILE · simulation record instruction set

2. TEMPLATE : discrete model instruction set

3 MODEL : simulation specifics; boundary and initial conditions

4 HOOKS : user written math , physics nuances

5. MACRO · macro/multi-grid data base

6. SOUT . screen file for execution tracking

7. COUT : output file

8. GOUT : output file for graphics

9. IOUT : integer print fields

10. NORM files : time evolution of norms

The GOUT files produced by **aPSE** contain data that are viewable using Tecplot (registered trademark of Amtec Engineering, Inc), a graphic visualization software. Here Tecplot is used to visualize the flow situation in the domain at various time stations, providing another way to estimate the convergence and conservation measures of the solution Tecplot is also used to generate the isoclines of various variables in the domain after solution convergence.

### 3.5.2 Hardware

For this research, aPSE has been run on Unix workstations in the Computational Fluid Dynamics Laboratory of the Engineering College at University of Tennessee.

# Chapter 4

# A BENCHMARK

## 4.1 The Problem - Thermal Cavity

A fluid (air) is considered that obeys the Boussinesq approximation [9]. The problem domain is described as a two-dimensional enclosure, being a square within which the air is enclosed. The vertical walls of the enclosures are maintained at constant temperatures $T_{right}$ and $T_{left}$ , and the horizontal walls ( top and bottom walls) are adiabatic. It is assumed that the Prandtl number $P_r$ of the fluid is maintained at 0.71

## 4.2 Past Research

### 4.2.1 Computational Work

The thermally driven square cavity (sometimes referred to as the "natural cavity" in the literature) has been a classic problem since de Vahl Davis and I.P. Jones's comparison

exercise ([5]) and subsequent establishment as a benchmark ([6]) for NS algorithms in 1983. The numerical results and flow patterns established in their exercise are quality benchmarks in the Rayleigh number range Ra=$10^3$ to $10^6$. In the next decade and the half, various researchers have computed results to the problem in the higher Ra ($10^7$-$10^8$) range ([7]).

### 4.2.2 Experimental Work

"Validation" is at least as important a part of CFD algorithm testing as "benchmarking". Little, if any, experimental data exist to "validate" the numerical data generated through the various contributions to the CFD literature for the thermal cavity case. The reason for this clearly lies in the almost intractable challenge of accurate experimental data extraction in the ranges of Ra = $10^3$ -$10^6$. While the recent contributions in numerical results in the ranges of Ra = $10^7$ -$10^8$ is a logical extension of the "benchmark" problem, extensions to still higher Ra ranges may encourage experimental work.

## 4.3 Key Issues of Solution

### 4.3.1 Non-Dimensionalization Issues

To generate benchmark quality solution(s), the numerical data for comparison has to be non-dimensionalized by the same scales as that of available benchmark solutions. For the Ra= $10^6$ case, the parameters D, $\alpha$, $\frac{D^2}{\alpha}$ are scale factors for length, streamfunction and time respectively. In the relatively higher convection regimes of Ra= $10^7$ and Ra=

$10^8$, the benchmark solution literature has used the parameters D, $\alpha$ and $\frac{D^2}{\alpha}Ra^{-0.5}$ as the scale factors.

### 4.3.2 Time Evolution of Norms

The norms of the various parameters are plotted against time for every run at all Rayleigh numbers. In this research, speacial emphasis has been given to interpretation of these time evolutions, mainly to observe the dependence of transient behaviour on Rayleigh Numbers (figs C 1,C.2,C.3)[1].

### 4.3.3 Solution "Steady State"

In the course of the present analysis, it has been observed that the "steady state" solutions are necessarily "numerically asymptotic" results of sufficient time evolutions. The dispersive numerical errors of $2\Delta x$ nature diminish with every mesh refinement, with the concomitant reduction in time step size and increase of number of time steps for the required convergence

### 4.3.4 The "Residual Divergence"

In view of the asymptotic nature of the time evolutions, an efficient global measure of solution accuracy is selected as the "Residual Divergence" - the divergence measure of the final "converged" velocity solution field. This measure, which is the $\Phi$-norm for the final converged solution, is included in tables[2] B.1,B.2 and B.3

[1] All figures may be found in Appendix C

[2] All tables may be found in Appendix B

### 4.3.5 A comment on solution strategy

The following solution strategy is consistently followed through out this benchmark generation exercise·

- run a 32×32 uniform mesh for $Ra = 10^6$, the upper limit of the de Vahl Davis benchmark data set
- restart from the previous (hot) run output as "Initial Conditions" (I.C.) , experimenting with non-uniform mesh refining , towards increasing convergence accuracy till the $\nabla \cdot u^* \leq \epsilon$ condition is reached.
- use the previous (converged) run as I.C. for the next higher Ra run.
- study the integer field data for the convergence characteristics
- use the 32×32 mesh data to generate a 64×64 mesh data by a linear interpolation scheme.
- redo the earlier steps as necessary on the refined mesh.

## 4.4 Algorithm Performance

### 4.4.1 Convergence issues

Typically pressure relaxation algorithms have quasi-Newton convergence rates; this is the case with the original Continuity Constraint Algorithm ([1]). One of the principal

reasons behind the incorporation of the constraint function $\Phi$ into the jacobian in the present algorithm was a quest for better convergence characteristics.

For comparing the convergence characteristics of the two algorithms, output data at Ra=$10^6$ is considered. Figure C 39 shows convergence data from the segregated jacobian algorithm and also from the present coupled jacobian algorithm. While the former shows a quasi-Newton convergence rate, the present algorithm shows Newton convergence. In figure C.40, a verification of convergence rates at Ra =$10^8$ is recorded; Newton convergence of iterations is clearly demonstrated at two randomly chosen time steps

### 4.4.2 Time Stepping

In the course of preliminary studies during formulation of the coupled algorithm, exhaustive experimentation was carried out with the segregated algorithm. The segregated algorithm was run for the range of Rayleigh numbers $10^3$ up to $10^6$ At a Rayleigh number, a sequence of runs were generated at progressive increments of time step size to determine the highest allowable time step that produced a true convergence A summary of the results obtained is recorded here for comparison.

At Ra=$10^3$, with a 16×16 element mesh, a maximum time step of 58.0 was supported by the algorithm. A higher time stepping resulted in oscillatory convergence characteristics At a Ra =$10^4$, the same time step of 58.0 produced oscillatory convergence. Progressive reduction of time step was carried on, till a stable solution was reached at a time step of 10.0. The next higher Ra number $10^5$ supported a time step up to 1.0.

Any higher time step produced an oscillatory solution at this Rayleigh number.

At Ra $=10^6$, a 32×32 element mesh was used with the segregated algorithm No non-oscillatory convergence for this Ra number could be extracted even for small time stepping. For example, a time stepping of 0.1 for a 32×32 element mesh produced an oscillatory convergence pattern till over 600 time steps (fig C.1).

The newly developed algorithm, at Ra $=10^6$, on a 32×32 element mesh, and with a time step of 5.0, shows excellent convergence, as is evident from the corresponding plot of state variable norms (fig C.2). Thus the superiority of the Newton algorithm lies in its ability to effect robust convergence at Ra numbers higher than those achievable by the segregated jacobian algorithm, and also in supporting larger time steps, which in turn assures faster solution at steady state.

### 4.4.3 Conservation issues

Conservation of mass is the most basic issue in incompressible Navier-Stokes algorithms. During formulation of the present algorithm, it was postulated that one of its strong points would be the robustness of conservation of mass. As discussed earlier, a direct indicator of this would be the residual divergence value. The best achievable value for this parameter for the Ra number $=10^6$ case on a 32×32 element mesh for the segregated algorithm was $5\times10^{-5}$ The corresponding value obtained from the present algorithm is $5\times10^{-8}$. This clearly shows the superiority of the coupled algorithm in achieving better conservation properties. The tabulated values of this norm of $\Phi$ (tables B.1,B.2, B.3) show a consistent and strong correlation of this parameter with the general solution

convergence.

## 4.5 Results

As one primary objective of this study is benchmarking of the numerical algorithm developed, numerical values obtained for certain parameters which are representative of the flow pattern are compared against corresponding values obtained by earlier researchers using different algorithms.

The de Vahl Davis solution ([6]) is well acknowledged as accurate for Ra=$10^4 - 10^6$. Hence the Ra=$10^6$ case is taken to be the beginning of the flow regime for the present study. Using the presented Newton algorithm, numerical results are obtained, which are then non-dimensionalised by scaling with respect to the scaling parameters defined in ([6]).

An outline of the software used (aPSE) has been given (3.5.1). For the new coupled u-v-T-$\Phi$ jacobian, a template file [3]was written, thus implementing the PDE(s) describing the problem, with full cross-coupling among the variables. The companion model file was also written, completing the thermal cavity problem description with initial conditions and boundary conditions.

A uniform mesh of 32×32 elements is taken as the starting point A run is typically started with small initial time step, with an incrementing factor allowing the integration time step size to grow to within a iterative convergence bound, as set by avoidance

[3]Typical program input files may be found in Appendix A

of oscillating convergence (linear algebra) encountered during the run Upon convergence to steady state of this initial run, a sequence of restarts on geometric progression non-uniform mesh refinements is undertaken, the purpose of which is to estimate the "optimal mesh" via norms of the state variables. Under this meshing scheme, the non-uniform mesh measure distribution is given by the equation

$$l_{e+1} = (1 + PR)l_e \tag{4.1}$$

where PR is the progression ratio of the mesh.

A plot (fig C.4) of the norm values against the mesh progression ratio shows a value in the range of 1.10 - 1.2 to produce the optimal solution for the selected Ra For subsequent experimentation and final data extraction, the progression ratio of 1.2 is selected for this Rayleigh number.

Convergence of any particular solution to steady state is directly assessed by analyzing the time evolution of the state variable energy norms. A typical time evolution plot from a "nice" run is shown in fig C.2. In this case, it can be seen that norms of the primitive variables are the true indicators of the physics behind the problem simulation. The $u^h$ and $T^h$ norms are particularly informative, with the oscillations being smoothed out by 250 time steps, verifying that a true steady state is acheived as the limit of the evolutionary process.

These non-dimensionalized numerical values are then tabulated along with corresponding values from ([6, 7]) in Table B 1. The value of the streamfunction at the

midpoint of the domain $\Psi_{mid}$ shows good agreement with the corresponding benchmark value given in [6]. It is noted that while the positions of occurance of $u_{max}$ and $w_{max}$ agree with the corresponding reference values, the actual values of those parameters show underprediction, pointing to the need for mesh refinement

Rocognizing the need for mesh refinement, a doubling of the uniform mesh to 64×64 elements is undertaken. In keeping with the philosophy of minimal runtime, the converged velocity and temperature fields from the earlier "coarse mesh" case is established as the initial condition using an elementary linear interpolation scheme Then, the similar sequence of progression ratio non-uniform mesh refinement study follows A tabulation of the non-dimensionalized numerical values (table B.1) clearly shows good agreement for all parameters of the reference benchmark at Ra=$10^6$.

The (r,h) meshing study sequence is repeated for Ra=$10^7$. As before, the uniform 32×32 mesh is the starting point. The run is started with small initial time step, using the time increment factor to allow the integration time step to grow just short of oscillating convergence As before, after convergence to steady state, a sequence of non-uniform refinements is undertaken - with the norms of the variables being recorded A plot (fig C.5) of the norm values against the mesh progression ratio shows a value of 1.10 - 1.15 to be the optimal zone for this Ra number. For subsequent experimentation and final data extraction, the progression ratio of 1.15 is selected for Rayleigh number of $10^7$.

The assertion that a true steady state is the limit of an evolutionary process in

verified for the Ra=$10^7$ case The time evolutions of the energy norms are recorded and analyzed for all the runs. A typical time evolution plot from a "nice" run is shown in fig C.3. It is again noted that the energy norms of the primitive variables and also the mass conservation constraint $\Phi$ smooth out by approximately 250-300 time steps.

Other than the computation and comparison of numerical values, plots of some important parameters representative of the flow are obtained for Ra$10^7$ (fig C.7 - C.11), using the 32×32 meshing. The plots provide a good qualitative visualization of the nature of the flow. The temperature contour plot (fig C.7) show the expected steep gradients along the hot and cold walls. The streamfunction contours (fig C.10) show the development of two additional (to Ra=$10^6$) circulation zones with the expected mirror symmetry All the contour plots, including the pressure (fig C.8) and vorticity (fig C.9), are visually satisfactory, agreeing with the plots obtained by other researchers.

One point to note for this case (Ra=$10^7$, 32 mesh) is the lack of smoothness of some of the contour plots. This clearly points to insufficient meshing where solution gradients change steeply. The plot of velocity vectors (fig C.11) shows clearly the development of flow from the solution boundaries (walls) to the developed zone But an "eyeball estimate" of the velocity plot also indicates the need of more mesh-points to bring out the flow details in the entire domain

The need for mesh refinement is thus apparent, hence progression to a mesh of 64×64 is undertaken. In keeping with the philosophy of minimal runtime, the converged velocity and temperature fields are linearly interpolated from the 32×32 solution for

the initial data set. The similar sequence of non-uniform mesh refinement follows the described procedure.

Plots of the relevent parameters (fig C.12 - C.16) again form the key part of the solution. The improvement in the quality of the countour plots is immediately visible via "eyeball estimates". As expected, improvement in resolution of $2\Delta x$ information is most evident at the zones of high gradients, as at the hot and cold walls. The improved meshing is found adequate to capture smoothly the temperature variation (fig C.12) near the walls. The plot of the velocity vectors (fig C 16) also show adequate resolution of the magnitude and direction of the velocity field

For the numerical comparism with results obtained by other researchers, non dimensionalized numerical values are calculated and tabulated along with corresponding values from ([7]) in Table B.2 Excellent agreement of the numerical results obtained in this experiment with the existing benchmark data is confirmed.

The study then progresses to the case of $Ra=10^8$, the starting point being a uniform mesh of $32\times32$. The need for adding a refined mesh for this solution is immediately apparent. The steep gradients near the walls and the intermediate variation in temperature gradiants are not adequately captured using the $32\times32$ mesh.

For the subsequent $64\times64$ mesh, the run is started as described with small initial time step. After convergence of the initial run, a sequence of non-uniform refinements is undertaken with the state variable norms being recorded. A plot (fig C 6) of the norm values against the mesh progression ratio shows a value of 1.10 - 1.15 to be the optimal

zone for this Ra number. For subsequent final data extraction, a progression ratio of 1.15 is taken.

The contour plots (fig C.17 - C.20) for the Ra=$10^8$, 64×64 mesh converged solution are particularly revealing. The temperature (fig C.17), pressure (fig C.18) and vorticity (fig C 19) solutions show distinct changes in patterns from the smaller Ra cases A notable point in the temparature contour is the presence of an undulation in two of the contour lines. This is symmetric and is retained even after oscillations in norms had died away. This solution, though acceptable, shows there is need for improvement to a finer mesh. The velocity vector plot (fig C.21) of this solution shows the several symmetric recirculation zones created at this temperature difference. This is also distinctly brought out in the streamfunction plot (fig C.20).

Hence, for Ra =$10^8$, another refinement is done to a mesh of 128×128 elements. The temperature and velocity fields of the converged 64×64 mesh solution are linearly interpolated for the initial conditions, and the solution built up through the described route of non-uniform re-meshing.

The contour plots (fig C.22 - C 25) for the Ra =$10^8$, 128×128 mesh case shows the effective capturing of the steep temperature gradients (fig C 22) at the hot and cold walls. The contours of the pressure (fig C.23) and vorticity (fig C.24) variables are also smooth The streamfunction plot (fig C.25) shows the formation of several symmetrically placed recirculation zones, as is also corroborated by the corresponding velocity vector plot (fig C.26). The process of extracting a "optimal" solution for this Ra number on the

128×128 mesh is particularly challanging, and ultimately a quality solution satisfactory to both the "eyeball estimate" and agreeing with numerical values obtained in [7] is obtained

Non dimensionalized numerical values are calculated and tabulated along with corresponding values from ([7]) in Table B.3 for both the 64×64 and 128×128 element meshing. Excellent agreement of the numerical results obtained in this experiment with the existing benchmark data is confirmed.

The richness of flow information obtained from this algorithm at Ra $=10^8$ is fully brought out by a few close ups of the streamfunction contours and the velocity vector plots (fig C.27 - C.38).

# Chapter 5

# SUMMARY AND CONCLUSIONS

A time accurate implicit Galerkin finite element algorithm is formulated via a Newton iteration. For evaluating performance, this algorithm is used to solve the two-dimensional thermal cavity benchmark problem at large Rayleigh numbers. The same problem is solved using the predecessor quasi-Newton algorithm for comparison.

The new algorithm, fully coupling the velocity, temperature and continuity constraint function degrees of freedom, achieves a genuine Newton convergence against the quasi-Newton convergence characteristics demonstrated by the segregated jacobian algorithm. The new algorithm is demonstrated to effect robust convergence at Ra numbers higher than those acheivable by the segregated jacobian algorithm, and also to support larger time stepping and hence faster steady state convergence

The ability to effect a strong conservation of mass has emerged as the most basic issue for algorithm performance The new algorithm is demonstrated to have consistent superiority in acheiving solutions with better mass conservation properties Moreover, a measure of the mass conservation state is shown to have strong correlation with the general solution convergence.

It is claimed that the present work provides a benchmark quality solution for the thermal cavity problem at Ra number $10^7$ and $10^8$. It is also recommended that future proposed solutions of this benchmark be accompanied by the determined - preferably energy norm measure of the mass conservation state of the final steady state solution.

# Bibliography

# Bibliography


[1] Williams,P T., "A Three Dimensional Time Accurate Incompressible Navıer Stokes Finite Element CFD Algorithm", PhD Dıssertation, University Of Tennessee, Knoxville (1993)

[2] Williams,P.T and A J.Baker "Incompressible Computational Fluid Dynamics and the Continuity Constraint Method for the Three Dimensıonal Navier Stokes Equatıons", *J.Numerıcal Heat Transfer, Fundamentals* , (1996)

[3] Baker,A.J., *Fınite Element Computatıonal Fluıd Mechanıcs* ,Hemisphere Publishing Corp., New York, (1983)

[4] Baker,A.J., *Handbook Of Fluıd Mechanıcs* ,Ch. 28, CRC Press, (1998)

[5] G. De Vahl Davıs and I.P. Jones " Natural Convection in a Square Cavity : A Comparison Exercıse", *Internatıonal Journal of Numerıcal Methods ın Fluıds* ,vol3, 227-248 (1983)

[6] De Vahl Davis "Natural Convection of air in a Square Cavity: A Bench Mark Numerical Solution", *International Journal of Numerical Methods in Fluids* , vol3, 249-264 (1983)

[7] P. Le Quéré "Accurate Solutions To The Square Thermally Driven Cavity at High Rayleigh Number", *Computers and Fluids* , Vol 21, No1, 29-41 (1991)

[8] P. Le Quéré and Masud Behnia "From Onset of Unsteadiness to Chaos in a Differentially Heated Square Cavity", *J. Fluid Mechanics* , vol359, 81-107 (1998)

[9] Gebhart, B. et al, *Buoyancy Induced Flow and Transport* , Hemisphere Publishing Corp. (1988)

[10] Batchelor, G.K., *An Introduction to Fluid Dynamics* , Cambridge University Press (1967)

[11] Gunzburger, M.D , *Finite Element Methods for Viscous Incompressible Flow* , Academics Press (1989)

[12] Strang, G. and G.J. Fix, *An Analysis of the Finite Element Method* , Prentice Hall(1973)

[13] White, F.M., *Viscous Fluid Flow* , McGraw-Hill Publishing Co. (1974)

[14] Hirsch, C, *Numerical Computation of Internal and External Flows* , (2 vols), John Wiley and Sons (1990)

# Appendices

# Appendix A

# Input Files

### A.0.1 Model File

```
TITLE       **** THERMAL CAVITY  MOD.CAV **** (2/22/00)
     PHI THERMAL CAVITY
INTEGRATION FACTORS
   0.        $   INITIAL_TIME
   6.E5      $   FINAL_TIME
   1.0E-5    $   PROBLEM_CONVERGENCE_CRITERIA
   0.5       $   MAXIMUM_CHANGE_IN_Q_(DQ)
   -0.0001   $   INITIAL_TIME_STEP
   1.25      $   TIME_STEP_MULTIPLIER
   1.        $   MAXIMUM_TIME_STEP
   4         $   CRITERIA_TO_RAISE_MAX_TIME_STEP
   200       $   MAXIMUM_NUMBER_OF_STEPS
   5         $   MAXIMUM_NUMBER_OF_ITERATIONS_PER_STEP
   1.0E-3    $   ITERATION_CONVERGENCE_CRITERIA
   0.5       $   THETA_IMPLICITNESS_FACTOR
   1.        $   CONVERGENCE_VARIABLE
GLOBAL SCALARS
  [ 0. 1. -1. 0. 0.  $  HT ONE MONE REI RE2I
    0. 0. .5 0. 0.0  $  GRSH PEI HALF SPHISW BETT
    0.0 2.0  ] $  BETA TWO
MATERIALS
    2      1
  [ 0.  ] $  DUM1
  [ 0.  ] T  DUM1
OTHERS
 0. 0 26           T RARRAY(26)
  1.0E-1           T PHI_DQ_LEVEL  (EPSPHI)
  0. 0 71          T RARRAY(71)
   1.0             T PHIFCT
  7.              T VECTOR 7
$     COMMON /CTHERM/
$    1    ALC,       GAMMAF,       AEXTO,       AEXT1,       AEXT2,
       1.          1.4          0.0          0.0          0.        # BGU
$    2   UREF,        AREF,      GASCON,        PREF,       PZERO,
     1.591337E-2  0.         4.972E4     2.1168E3     2.1168E3    # BGU
$    3 CONDCT,       GRAVCN,      SPHEAT,       VISCK,         XMW,
       4.175889E-6  32.17398    .237    1.591337E-4     28.95748   # BGU
$    4   TREF,        TMAX,        TMIN,       TZERO,      DELTEM,
       70.0        61.0126      0.0        459.6       0.          # BGU
$    5    BTU,       XMAREF,       XMAIN,      XMAOUT,      TINTFT,
       778.0       0.          0.          0.          12.         # BGU
```

```
$    6     CF,         HREF,        HZERO,        CEXT1,        CEXT2,
       0.          0.           0.           0.           0.          # BGU
$    7   SREF,         SMAX,         SMIN,        SZERO,      DELSAL,
       0.          0.           0.           0.           0.          # BGU
$    8  DEXT1,        DEXT2,        DEXT3,        DEXT4,        DEXT5,
       0.          0.           0.           0.           0.         T
$    9 ZXT(60)
#     You can add (or replace) variables that you have incorporated
#        into your 'hooks' files.
#     Numeric entries are free format.
#     Entries, such as AEXT1 etc. are open for your use.
#  N.B.  ALC is the ONLY variable used in the 'main' subroutines.
#          It is set to 1.0, if you do not set it here.
#          It MUST be in position 1 in this COMMON block.
#   END OF CTHERM
#     COMMON /WALLFN/ TINTPR(100),THKNES(100),SKNFRN(100),TMHICK(100)
  0  32             T
  0                 T  ISMPHI
   0 10             T  ISCALR(10)
   1                T  IETKJA
PRINT
   129   129      1   # GRID SIZE    TOTAL OF   16641 NODES
                 20   # PRINT INTERVAL (STEPS)
           1    129   # I WINDOW
           1    129   # J WINDOW
           1      1   # K WINDOW
NONE # VARIABLES IN COUT
NONE # VARIABLES FIRST STEP ONLY
(1P,25(1X,,5E14.6/)1X,4E14.6))  #  FORMAT FOR 129x129
GRAPHICS
 X1 X2 TEMP U1 U2 PHI SPHI PRES OMGA PSI OMGE PSIE U1E U2E PRSE TMPE  T
THREE_INTEGER    T  INTEGER PRINT FIELD
 TEMP U1 U2 PHI SPHI PRES X1 X2 OMGA PSI OMGE PSIE U1E U2E PRSE TMPE  T
DEBUG
 NONE $ STEP NUMBER TO PRINT ALL ITERATIONS
 NONE $ TYPE OF DEBUG PRINT
 1 $ ITERATION NUMBER
 102,1,1 $ I,J,K  LOCATION, (K=0, ALL PLANES)
 ALL $ VARIABLE OR ALL
 ALL $ TERM NUMBER OR (- FOR SET NO.), OR ALL
BOUNDARY CONDITIONS
 WALL_NS  1 [ 129I1 1 127I129 130 127I129 258 129I-1 16641 ]
```

```
 DIRI_T   1 [ 129I129 1 129I129 129 ]
INITIAL CONDITIONS
  [16641*0.] $ U1
  [16641*0.] $ U2
  [S 129(0. 64R1. .305063 64R1. .610126) S100.] $ TEMP # INTER., FOR RAYLEIGH NO.
  [16641*0.] $ PHI
  [16641*0.] $ PRES
  [16641*0.] $ OMGA
  [16641*0.] $ PSI
  [ 129(0. 20R1.25 .05 88R1.0 .95 20R.8 1.0) ] $ X1    74
  [ -129(0. 20R1.25 .05 88R1.0 .95 20R.8 1.0) ] $ X2    75
# [129(0. 64R1.0 .5 64R1.0 1.0)] $ X1 # INTERPOLATED
#[-129(0. 64R1.0 .5 64R1.0 1.0)] $ X2 # INTERPOLATED
  [16641*0.] T ONES
END
```

### A.0.2 Template File

```
##### **** temp.pns **** {U1 U2 TEMP PHI} {PRES} {OMGA} {PSI} {NORMS}
INTEGRATION FACTORS
   INITIAL_TIME
   FINAL_TIME
   PROBLEM_CONVERGENCE_CRITERIA
   MAXIMUM_CHANGE_IN_Q_(DQ)
   INITIAL_TIME_STEP
   TIME_STEP_MULTIPLIER
   MAXIMUM_TIME_STEP
   CRITERIA_TO_RAISE_MAX_TIME_STEP
   MAXIMUM_NUMBER_OF_STEPS
   MAXIMUM_NUMBER_OF_ITERATIONS_PER_STEP
   ITERATION_CONVERGENCE_CRITERIA
   THETA_IMPLICITNESS_FACTOR
   CONVERGENCE_VARIABLE

TRANSFORMATION ARRAYS
      ETKJ      1.
      DETJ      1
#      DETE      0.

BOUNDARY CONDITIONS
          U1 U2 TEMP PHI PRES OMGA PSI    # ORDER
  FLUX     3  0  0    0   0    0    0     # CONVECTION HEAT FLUX
  HFLUX    4  0  0    0   0    0    0     # HEAT FLUX
  RFLUX    5  0  0    0   0    0    0     # RADIATION HEAT FLUX
  HRFLX    6  0  0    0   0    0    0     # RADIATION HEAT FLUX
  INLT_P   3  0  0    0   0    0    0     # INLET
  WALL_SL  4  0  0    0   0    0    0     # SLIP WALL
  DIRI_U   D  0  0    0   0    0    0     # DIRICHLET U
  DIRI_V   0  D  0    0   0    0    0     # DIRICHLET V
  WALL_NS  D  D  0    0   0    0    D     # NO SLIP WALL
  DIRI_T   0  0  D    0   0    0    0     # WALL TEMPERATURE
  DIRI_PHI 0  0  0    D   0    0    0     # THROUGHFLOW PHI
  DIRI_P   0  0  0    0   D    0    0     # THROUGHFLOW PRESSURE
  DIRI_OMG 0  0  0    0   0    D    0     # DIRICHLET OMGA
  DIRI_PSI 0  0  0    0   0    0    D     # DIRICHLET PSI
  BLANK    D  D  D    D   D    D    D     # BLANK REGION

TITLE
     PHI CONSTRAINT INS GWS ALGORITHM, 2D, NEWTON JACOBIAN (12/21/99)
```

```
RESIDUALS
  U1   1    #  VARBL, SET NO.,  --- TEMPORAL SET (U1)
 ()()()(0;1)(B200)(-U1)
+()()()(1;0)(B201)(PHI)
+()()()(3;0)(B202)(PHI)
+()()()(1;0)(B201)(SPHI)
+()()()(3;0)(B202)(SPHI)
  U1   2    #  VARBL, SET NO.,  --- SPATIAL  SET (U1)
 ()()(U1+U2)(1020;0)(B3001)(U1)
+()()(U1+U2)(3040;0)(B3002)(U1)
+()()()(1;0)(B201)(PRES)
+()()()(3;0)(B202)(PRES)
+(REI)()()(1122;-1)(B211)(U1)
+(REI)()()(3344;-1)(B222)(U1)
+(REI)()()(1324;-1)(B221)(U1)
+(REI)()()(1324;-1)(B212)(U1)

  U2   1    #  VARBL, SET NO.,  --- TEMPORAL SET (U2)
 ()()()(0;1)(B200)(-U2)
+()()()(2;0)(B201)(PHI)
+()()()(4;0)(B202)(PHI)
+()()()(2;0)(B201)(SPHI)
+()()()(4;0)(B202)(SPHI)
  U2   2   #  VARBL, SET NO.,  --- SPATIAL  SET (U2)
 ()()(U1+U2)(1020;0)(B3001)(U2)
+()()(U1+U2)(3040;0)(B3002)(U2)
+()()()(2;0)(B201)(PRES)
+()()()(4;0)(B202)(PRES)
+(REI)()()(1122;-1)(B211)(U2)
+(REI)()()(3344;-1)(B222)(U2)
+(REI)()()(1324;-1)(B221)(U2)
+(REI)()()(1324;-1)(B212)(U2)
+(-,GRSH,RE2I)()()(0;1)(B200)(TEMP)

  TEMP   1    #  VARBL, SET NO.,  --- TEMPORAL SET (TEMP)
 ()()()(0;1)(B200)(-TEMP)
  TEMP   2   #  VARBL, SET NO.,  --- SPATIAL  SET (TEMP)
 ()()(U1+U2)(1020;0)(B3001)(TEMP)
+()()(U1+U2)(3040;0)(B3002)(TEMP)
+(PEI)()()(1122;-1)(B211)(TEMP)
+(PEI)()()(3344;-1)(B222)(TEMP)
```

```
+(PEI)()()(1324;-1)(B221)(TEMP)
+(PEI)()()(1324;-1)(B212)(TEMP)
  TEMP   3   #  VARBL, SET NO.,  --- BOUNDARY SET (TEMP)
 ()(NUSL)()(0;1)(A200)(TEMP)
+(-)(NUSL)()(0;1)(A200)(TRBC)
  TEMP   5   #  VARBL, SET NO.,  --- BOUNDARY SET (TEMP)
 ()()()(0;1)(A200)(SRCT)

  PHI   1     #  VARBL, SET NO.,  --- SPATIAL  SET (PHI)
 ()()()(1;0)(B3001)(U1)
+()()()(3;0)(B3002)(U1)
+()()()(2;0)(B3001)(U2)
+()()()(4;0)(B3002)(U2)
+()()()(1122;-1)(B3011)(PHI)
+()()()(3344;-1)(B3022)(PHI)
+()()()(1324;-1)(B3021)(PHI)
+()()()(1324;-1)(B3012)(PHI)

  PHI   3  #  VARBL, SET NO.,-- BOUNDARY SET (PHI)
 (-)()()(1;0)(A200)(U1)
+()()()(2;0)(A200)(U2)

JACOBIANS
  U1  U1  1  1  #  VARBL, VARDIF, SET, DIRECTION 1
 ()()()(;1)(B200)()
  U1  U1  2  1  #  VARBL, VARDIF, SET, DIRECTION 1
+()()(U1+U2)(1020;0)(B3001)()
+()()(U1+U2)(3040;0)(B3002)()
+()()(U1)(1;0)(B3100)()
+()()(U1)(3;0)(B3200)()
+(REI)()()(1122;-1)(B211)()
+(REI)()()(1324;-1)(B212)()
+(REI)()()(1324;-1)(B221)()
+(REI)()()(3344;-1)(B222)()

  U1  U2  2  1  #
 ()()(U1)(2;0)(B3100)()
+()()(U1)(4;0)(B3200)()

   U1  PHI 1 1 #
 ()()()(1;0)(B201)()
+()()()(3;0)(B202)()
```

```
  U2  U1  2  1  #
 ()()(U2)(1;0)(B3100)()
+()()(U2)(3;0)(B3200)()

  U2  U2  1  1  #
 ()()()(;1)(B200)()
  U2  U2  2  1  #
+()()(U1+U2)(1020;0)(B3001)()
+()()(U1+U2)(3040;0)(B3002)()
+()()(U2)(2;0)(B3100)()
+()()(U2)(4;0)(B3200)()
+(REI)()()(1122;-1)(B211)()
+(REI)()()(1324;-1)(B212)()
+(REI)()()(1324;-1)(B221)()
+(REI)()()(3344;-1)(B222)()

  U2 TEMP 2  1 #
+(-,GRSH,RE2I)()()(;1)(B200)()

   U2  PHI  1 1 #
 ()()()(2;0)(B201)()
+()()()(4;0)(B202)()

 TEMP U1 2 1 #
()()(TEMP)(1;0)(B3100)()+()()(TEMP)(3;0)(B3200)()

 TEMP U2 2 1 #
()()(TEMP)(2;0)(B3100)()+()()(TEMP)(4;0)(B3200)()

  TEMP TEMP 1  1     #
 ()()()(0;1)(B200)()
  TEMP TEMP 2  1    #
+()()(U1+U2)(1020;0)(B3001)()
+()()(U1+U2)(3040;0)(B3002)()
+(PEI)()()(1122;-1)(B3011)()
+(PEI)()()(3344;-1)(B3022)()
+(PEI)()()(1324;-1)(B3021)()
+(PEI)()()(1324;-1)(B3012)()
 TEMP TEMP 3  1 #
 ()(NUSL)()(0;1)(A200)()
```

```
  PHI U1  1 1 #
 ()()()(1;0)(B3001)()
+()()()(3;0)(B3002)()
  PHI U1  3 1 #
 (-)()()(1;0)(A200)()

  PHI U2 1 1 #
 ()()()(2;0)(B3001)()
+()()()(4;0)(B3002)()
  PHI U2 3 1 #
 ()()()(2;0)(A200)()

  PHI PHI  1  1  #  VARBL, VARDIF, SET, ALL DIRECTIONS
 ()()()(1122;-1)(B3011)()
+()()()(3344;-1)(B3022)()
+()()()(1324;-1)(B3021)()
+()()()(1324;-1)(B3012)()

GROUP FREQUENCY
      1
SOLUTION TYPE
  DELTA_Q
  ILU_GMRES
  IMPLICIT_EULER
END

TITLE
   PHI ALGORITHM,  DELSQ PRESSURE SOLVE

RESIDUALS
  PRES   2   #  VARBL, SET NO.,  --- SPATIAL  SET (PRES)
 ()(U1)()(11;-1)(B3011)(U1)
+()(U2)()(12;-1)(B3011)(U1)
+()(U1)()(21;-1)(B3011)(U2)
+()(U2)()(22;-1)(B3011)(U2)
+()(U1)()(13;-1)(B3012)(U1)
+()(U2)()(14;-1)(B3012)(U1)
+()(U1)()(23;-1)(B3012)(U2)
+()(U2)()(24;-1)(B3012)(U2)
+()(U1)()(31;-1)(B3021)(U1)
+()(U2)()(32;-1)(B3021)(U1)
```

```
+()(U1)()(41;-1)(B3021)(U2)
+()(U2)()(42;-1)(B3021)(U2)
+()(U1)()(33;-1)(B3022)(U1)
+()(U2)()(34;-1)(B3022)(U1)
+()(U1)()(43;-1)(B3022)(U2)
+()(U2)()(44;-1)(B3022)(U2)
+(-,GRSH,RE2I)()()(2;0)(B210)(TEMP)
+(-,GRSH,RE2I)()()(4;0)(B220)(TEMP)
  PRES   6   #  VARBL, SET NO.,  --- BOUNDARY  SET (PRES)
 (REI)()(RET)(0;-1)(A3011)(U1)
#()(U1)()(11;-1)(B3011R)(U1)
  PRES   7   #  VARBL, SET NO.,  --- BOUNDARY  SET (PRES)
 (-,REI)()(RET)(0;-1)(A3011)(U1)
  PRES   8   #  VARBL, SET NO.,  --- BOUNDARY  SET (PRES)
 (TWO,REI)()(RET)(0;0)(A3001)(U1)
  PRES   9   #  VARBL, SET NO.,  --- BOUNDARY  SET (PRES)
 (-,TWO,REI)()(RET)(0;0)(A3001)(U1)

JACOBIANS
  PRES PRES  2  1  #  VARBL, VARDIF, SET, ALL DIRECTIONS
 ()()()(1122;-1)(B3011)()
+()()()(3344;-1)(B3022)()
+()()()(1324;-1)(B3021)()
+()()()(1324;-1)(B3012)()

GROUP FREQUENCY
    -1
SOLUTION TYPE
  Q
  ILU_GMRES
  IMPLICIT_EULER
END

TITLE            **** TEMPLATE TEMP.PHI ****
   OMEGA - FROM DELXU  (1/99)

RESIDUALS
  OMGA   2    #  VARBL, SET NO.,  --- SPATIAL  SET (OMGA)
 ()()()(2;0)(B201)(U1)
+()()()(4;0)(B202)(U1)
+(-)()()(1;0)(B201)(U2)
+(-)()()(3;0)(B202)(U2)
```

```
JACOBIANS
  OMGA OMGA  2  1  #  VARBL, VARDIF, SET, ALL DIRECTIONS
 ()()()(0;1)(B200)()

GROUP FREQUENCY
      -10
SOLUTION TYPE
  Q
  ILU_GMRES
  IMPLICIT_EULER
END

TITLE            **** TEMPLATE TEMP.PHI ****
   DELSQ PSI FROM OMEGA  (1/99)

RESIDUALS
  PSI   2  #  VARBL, SET NO.,  --- SPATIAL  SET (PSI)
 (-)()()(0;1)(B200)(OMGA)

JACOBIANS
  PSI PSI  2  1 # VARBL, VARDIF, SET, ALL DIRECTIONS
 ()()()(1122;-1)(B211)()
+()()()(3344;-1)(B222)()
+()()()(1324;-1)(B221)()
+()()()(1324;-1)(B212)()

GROUP FREQUENCY
      -10
SOLUTION TYPE
  Q
  ILU_GMRES
  IMPLICIT_EULER
END

TITLE            **** TEMPLATE TEMP.PHI ****
  ENERGY NORM COMPUTATIONS, ALL VARIABLES/PARAMETERS  (1/99)

RESIDUALS
  DUNC   1   #                    TEMPORAL TERM IN {F(Q)}
 ()()()(0;1)(B200)(-DUNC)
```

```
JACOBIANS
  DUNC DUNC   1 1   #              TIME TERM IN {F(Q)}
 ()()()(0;1)(A200)()
  DUNC DUNC   1 2   #              TIME TERM IN {F(Q)}
 ()()()(0;1)(A200)()

NORMS
  OMGE  1  T                                ENERGY NORM FOR OMGA
 (HALF,REI)()(OMGA)(1122;-1)(B211)(OMGA)
 (HALF,REI)()(OMGA)(1324;-1)(B212)(OMGA)
 (HALF,REI)()(OMGA)(1324;-1)(B221)(OMGA)
 (HALF,REI)()(OMGA)(3344;-1)(B222)(OMGA)

  PSIE  1  T                                ENERGY NORM FOR PSI
 (HALF)()(PSI)(1122;-2)(B211)(PSI)
 (HALF)()(PSI)(1324;-2)(B212)(PSI)
 (HALF)()(PSI)(1324;-2)(B221)(PSI)
 (HALF)()(PSI)(3344;-2)(B222)(PSI)

  U1E  1  T                                 ENERGY NORM FOR U1
 (HALF)()(U1)(1122;-1)(B211)(U1)
 (HALF)()(U1)(1324;-1)(B212)(U1)
 (HALF)()(U1)(1324;-1)(B221)(U1)
 (HALF)()(U1)(3344;-1)(B222)(U1)

  U2E  1  T                                 ENERGY NORM FOR U2
 (HALF)()(U2)(1122;-1)(B211)(U2)
 (HALF)()(U2)(1324;-1)(B212)(U2)
 (HALF)()(U2)(1324;-1)(B221)(U2)
 (HALF)()(U2)(3344;-1)(B222)(U2)

  PRSE  1  T                                ENERGY NORM FOR PRES
 (HALF)()(PRES)(1122;-1)(B211)(PRES)
 (HALF)()(PRES)(1324;-1)(B212)(PRES)
 (HALF)()(PRES)(1324;-1)(B221)(PRES)
 (HALF)()(PRES)(3344;-1)(B222)(PRES)

  TMPE  1  T                                ENERGY NORM FOR TEMP
 (HALF,PEI)()(TEMP)(1122;-1)(B211)(TEMP)
 (HALF,PEI)()(TEMP)(1324;-1)(B212)(TEMP)
 (HALF,PEI)()(TEMP)(1324;-1)(B221)(TEMP)
 (HALF,PEI)()(TEMP)(3344;-1)(B222)(TEMP)
```

```
  PHIE  1  T                                ENERGY NORM FOR PHI
 (HALF)()(PHI)(1122;-1)(B211)(PHI)
 (HALF)()(PHI)(1324;-1)(B212)(PHI)
 (HALF)()(PHI)(1324;-1)(B221)(PHI)
 (HALF)()(PHI)(3344;-1)(B222)(PHI)

GROUP FREQUENCY
      1
SOLUTION TYPE
  DELTA_Q
  FACTORED_GAUSS_ELIMINATION
  IMPLICIT_EULER

END
```

# Appendix B

# Tables

Table B.1. Benchmark Numerical Data

| **Rayleigh Number = $10^6$** | | | | | |
|---|---|---|---|---|---|
| Parameters | 32×32 (quasi Newton) | 32×32 | 64×64 | Ref [6] | Ref [7] |
| $\Psi_{mid}$ | 15 48 | 16 87 | 16 37 | 16 32 | 16 386 |
| $u_{max}$ | 76 02 | 70 93 | 64 8 | 64 63 | 64 83 |
| z | 0 85 | 0.85 | 0 85 | 0 85 | 0 85 |
| $w_{max}$ | 209.11 | 213 32 | 219 33 | 219.36 | 220 60 |
| x | 0 04 | 0 04 | 0 04 | 0 038 | 0 038 |
| div | $5\times10^{-5}$ | $5\times10^{-8}$ | $5\times10^{-8}$ | - | $5\times10^{-9}$ |

Table B.2: Benchmark Numerical Data

| **Rayleigh Number** $= 10^7$ | | | | | |
|---|---|---|---|---|---|
| Parameters | 32×32 | 64×64 | —— | Ref [6] | Ref [7] |
| $\Psi_{mid}$ | 0 0103 | 0 0091 | - | - | $9\ 235\times10^{-3}$ |
| $u_{max}$ | 0 06401 | 0 04501 | - | - | 0 04699 |
| z | 0 85 | 0 88 | - | - | 0 879 |
| $w_{max}$ | 0 2057 | 0 2190 | - | - | 0 2211 |
| x | 0 02 | 0 02 | - | - | 0 021 |
| div | $2\times10^{-6}$ | $5\times10^{-8}$ | - | - | $5\times10^{-9}$ |

Table B 3 Benchmark Numerical Data

| Rayleigh Number = $10^8$ | | | | | |
|---|---|---|---|---|---|
| Parameters | 32×32 | 64×64 | 128×128 | Ref [6] | Ref [7] |
| $\Psi_{mid}$ | 5 637×$10^{-3}$ | 5 22×$10^{-3}$ | 5 36×$10^{-3}$ | - | 5 232×$10^{-3}$ |
| $u_{max}$ | 0 0306 | 0 0322 | 0 0322 | - | 0 03219 |
| z | 0 93 | 0 926 | 0 926 | - | 0 928 |
| $w_{max}$ | 0 2074 | 0 221 | 0.222 | - | 0 2222 |
| x | 0 01 | 0 01 | 0 01 | - | 0.012 |
| div | 2×$10^{-5}$ | -2×$10^{-7}$ | 2×$10^{-7}$ | - | 5×$10^{-8}$ |

# Appendix C

# Figures

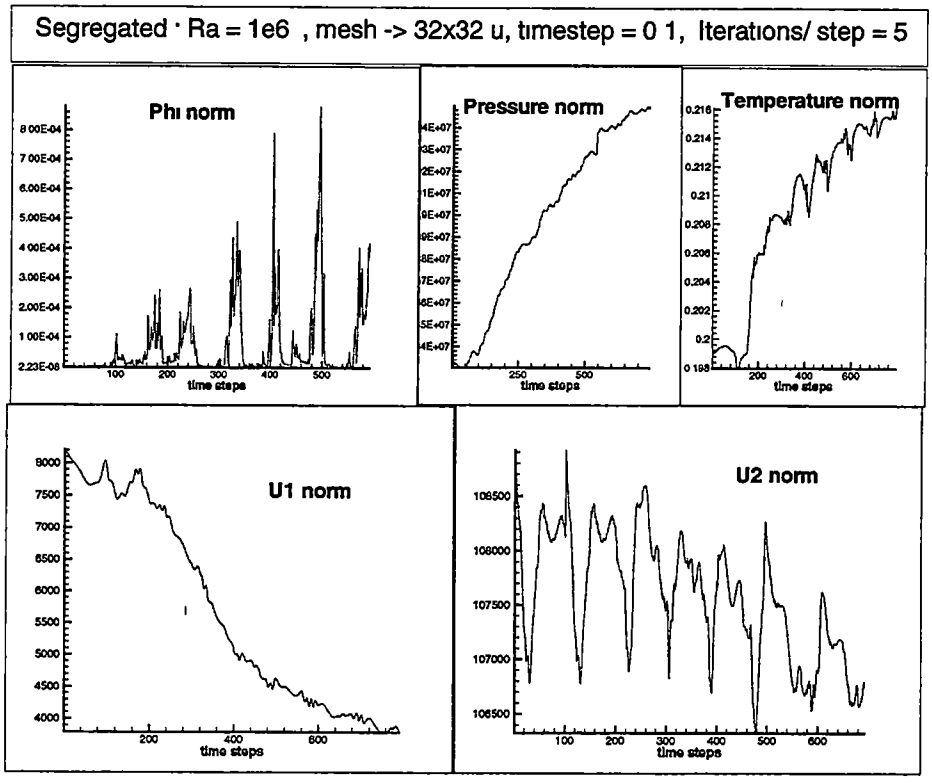


Figure C.1: Time evolution of Norms (segregated jacobian), 32 mesh, Ra=$10^6$

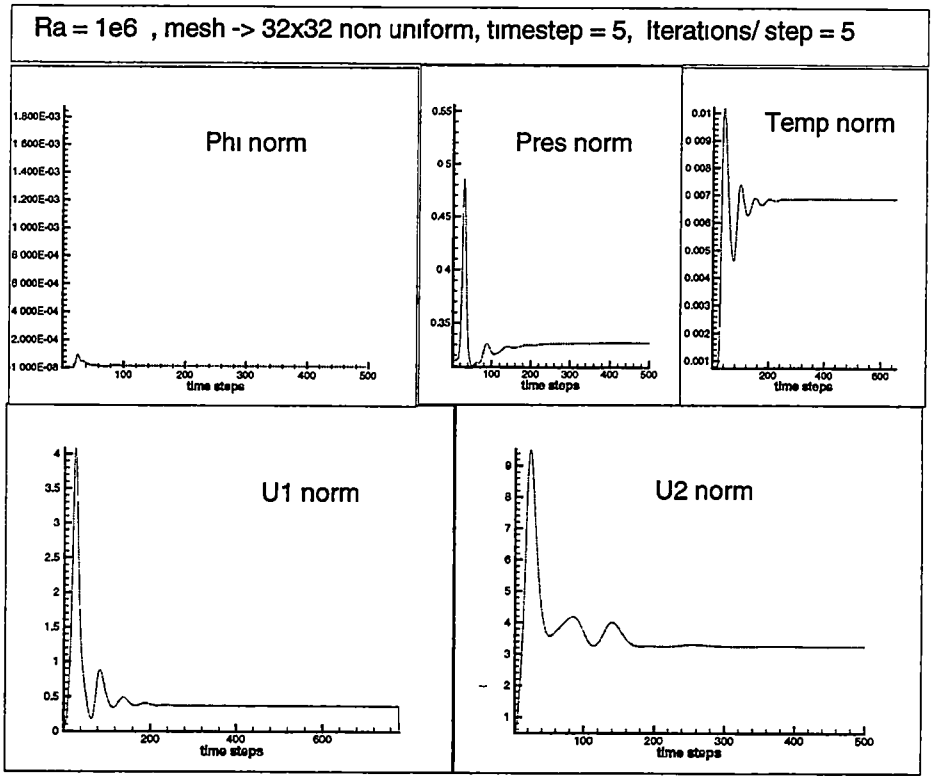


Figure C.2: Time evolution of Norms, 32 mesh, Ra=$10^6$

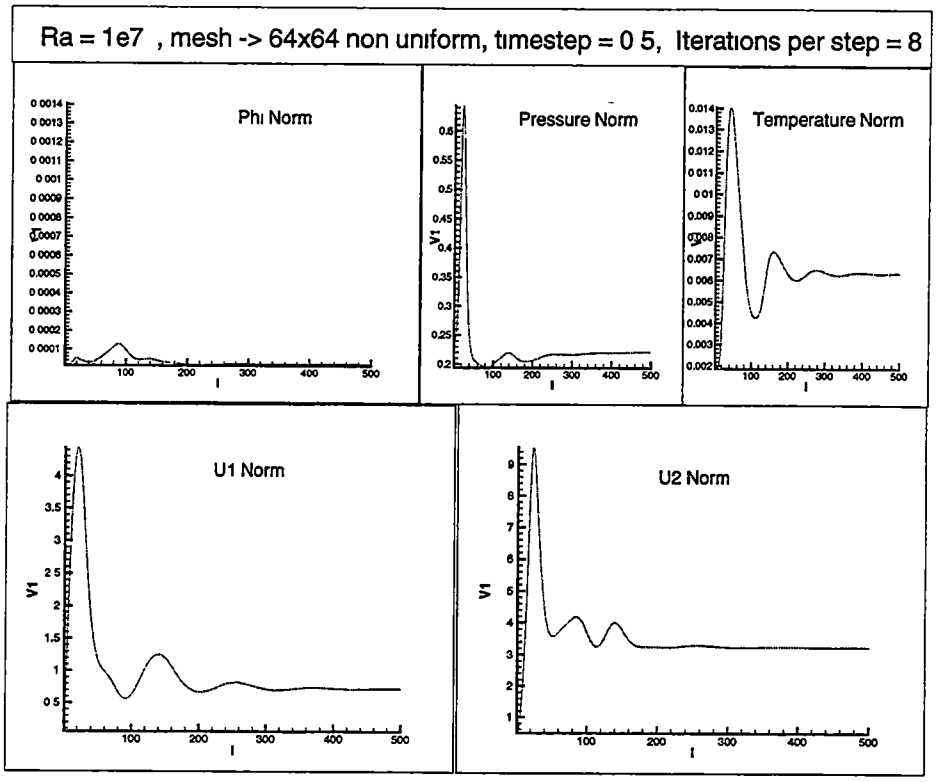


Figure C.3: Time evolution of Norms, 64 mesh, Ra=$10^7$

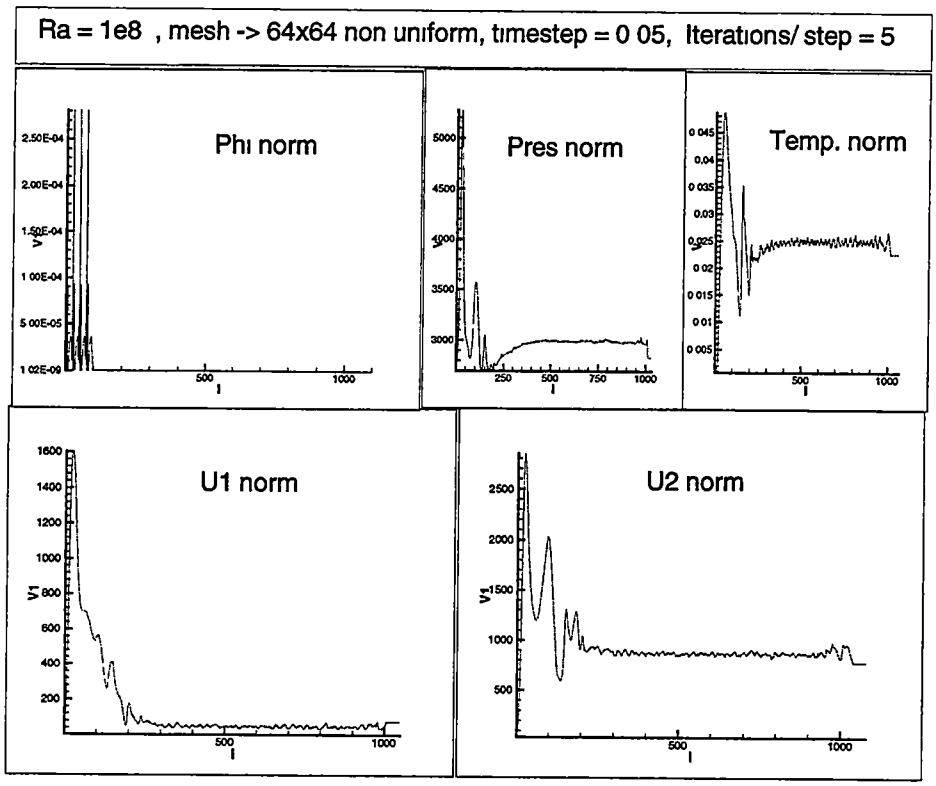


Figure C 4 Time evolution of Norms, 64 mesh, Ra=$10^8$

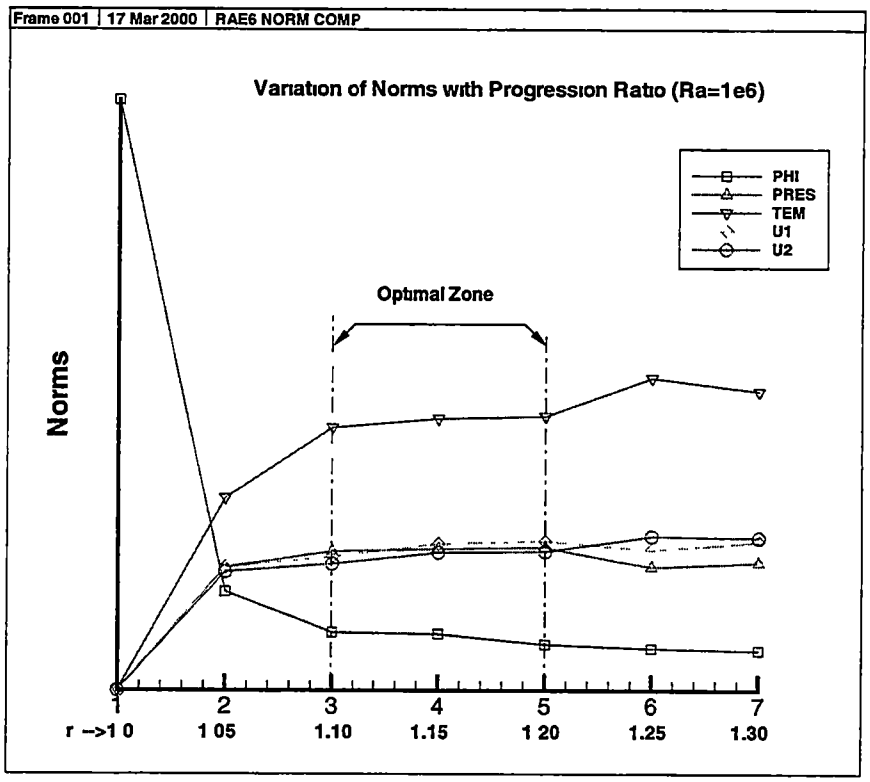


Figure C.5: Optimal mesh Progression Ratio, Ra=$10^6$

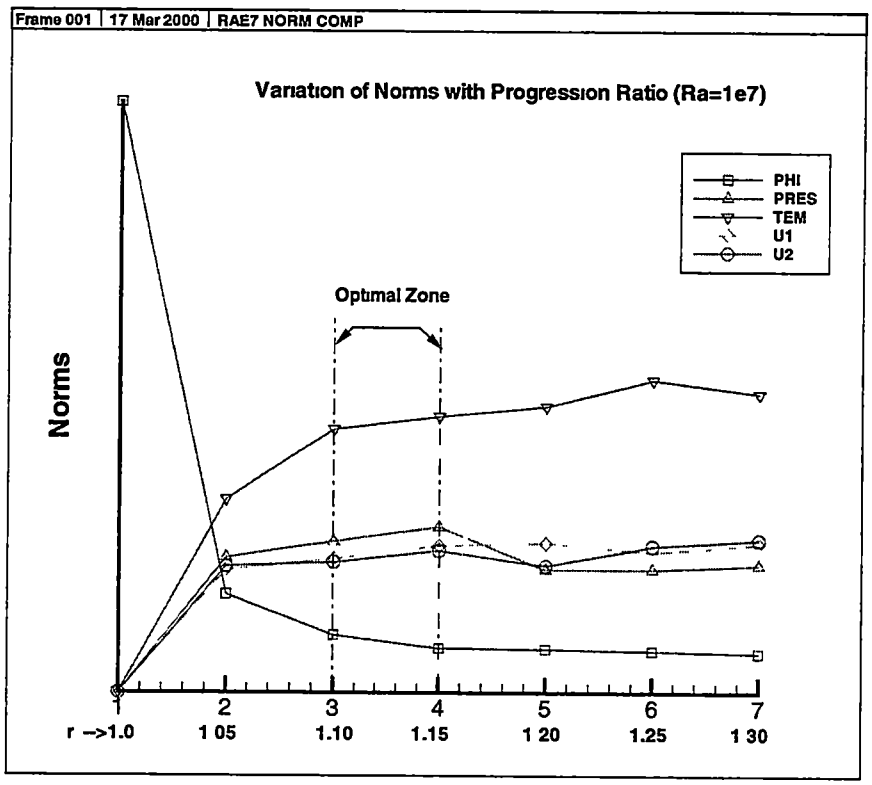


Figure C.6: Optimal mesh Progression Ratio, Ra=$10^7$

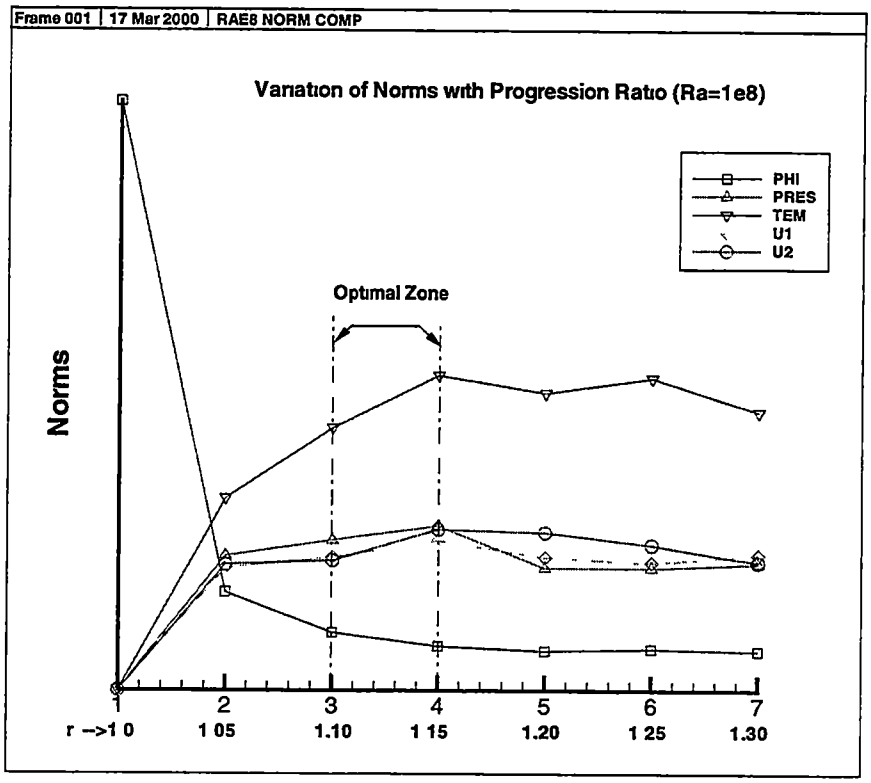


Figure C 7: Optimal mesh Progression Ratio, Ra=$10^8$

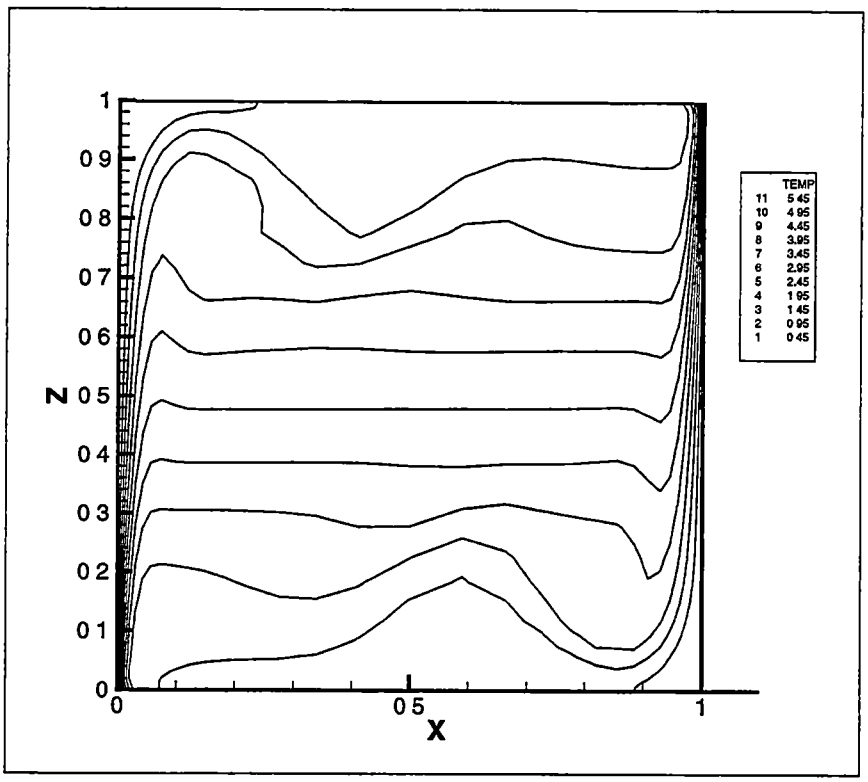


Figure C 8: Temperature contour, Ra=$10^7$, 32x32 mesh

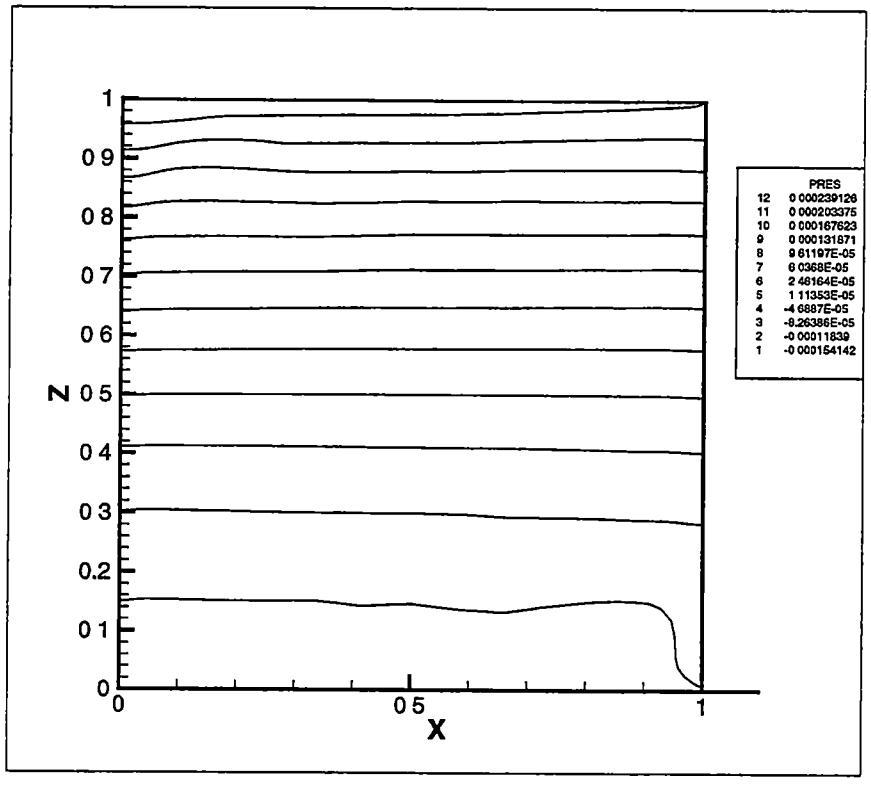


Figure C.9: Pressure contour, Ra=$10^7$, 32x32 mesh

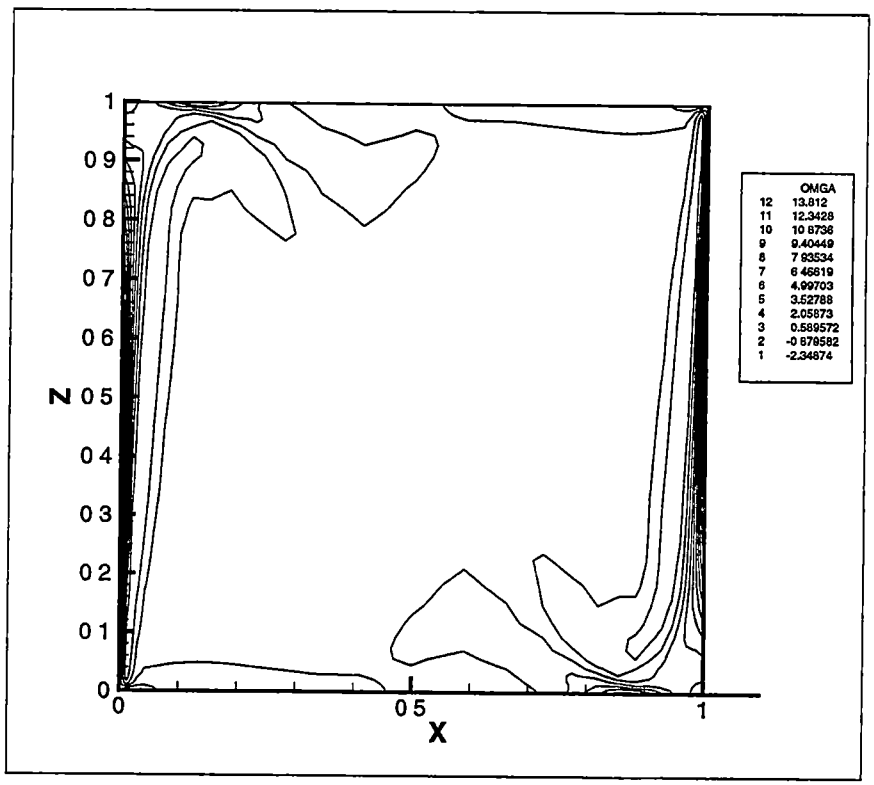


Figure C.10: Vorticity contour, Ra=$10^7$, 32x32 mesh

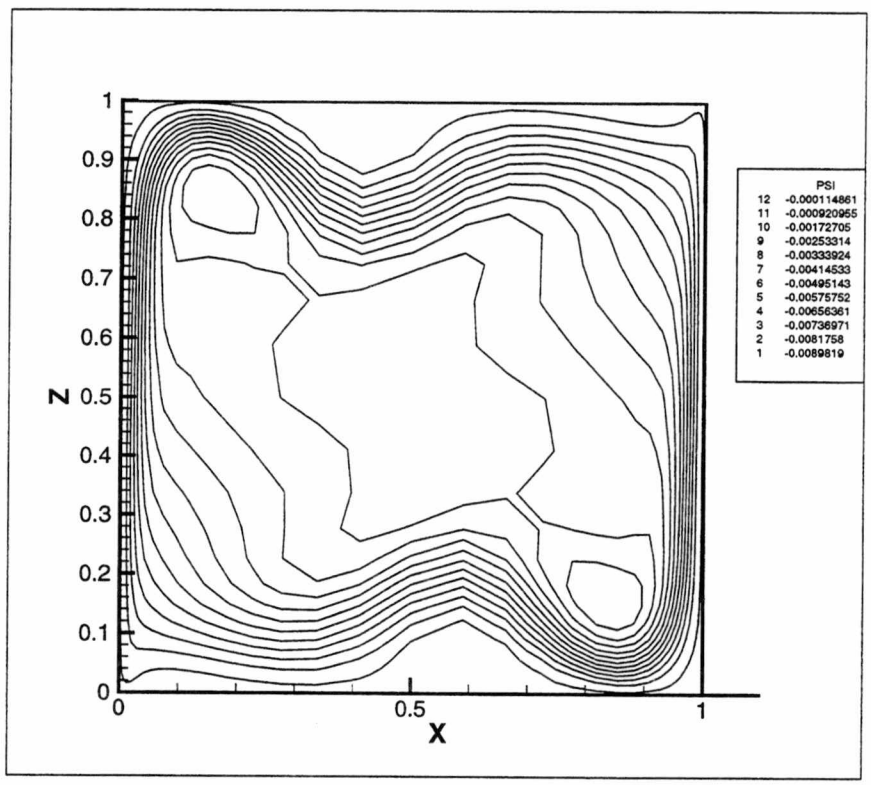


Figure C.11: Streamlines, Ra=$10^7$, 32x32 mesh

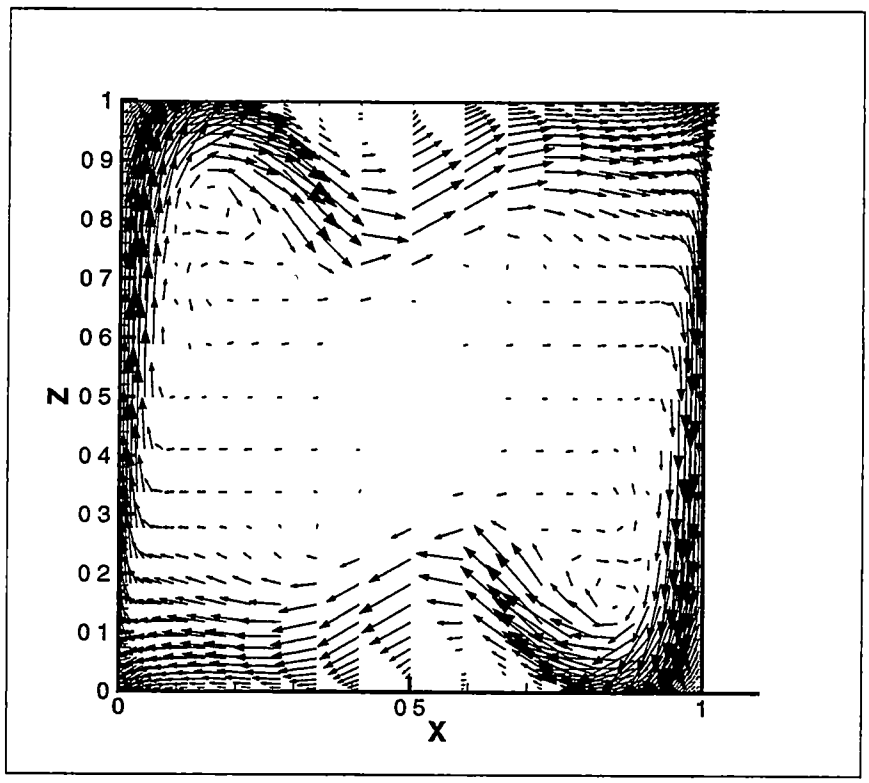


Figure C.12: Velocity Field, Ra=$10^7$, 32x32 mesh

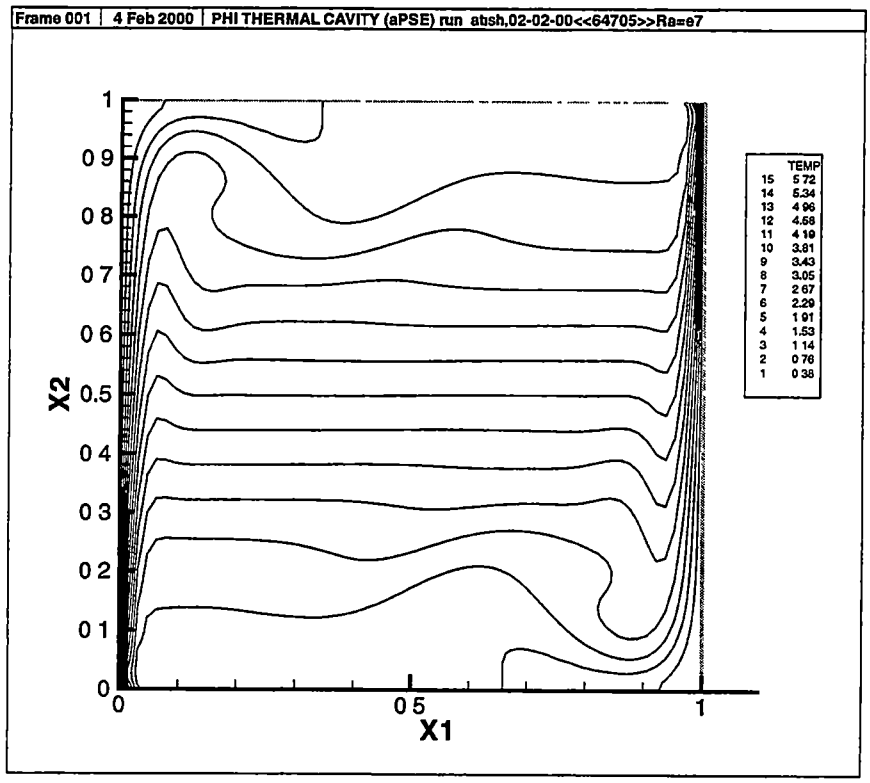


Figure C.13: Temperature contour, Ra=$10^7$, 64x64 mesh

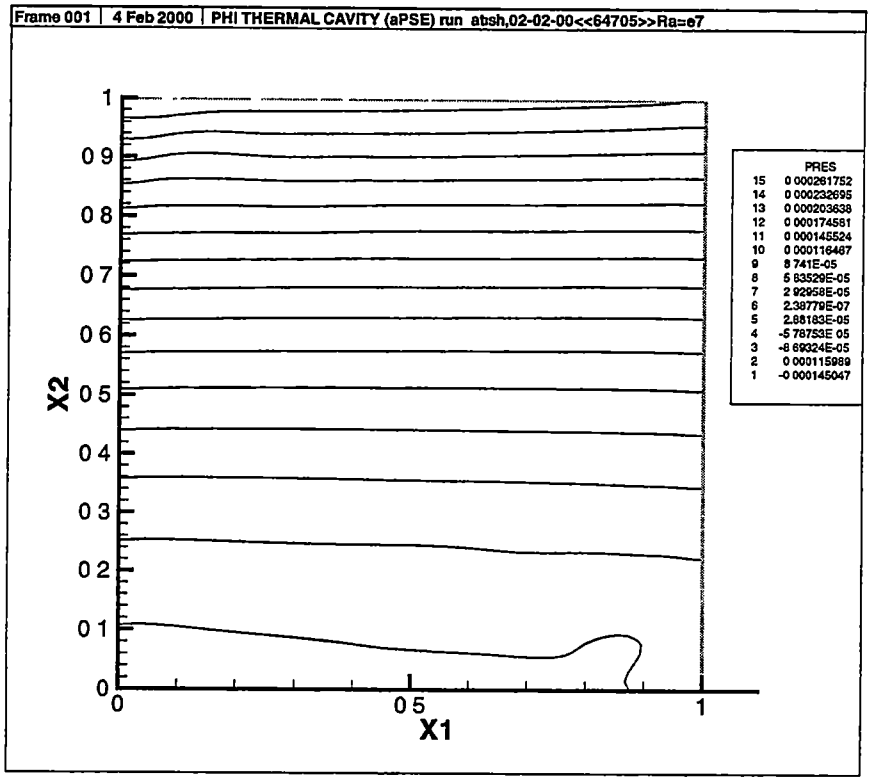


Figure C 14. Pressure contour, Ra=$10^7$, 64x64 mesh

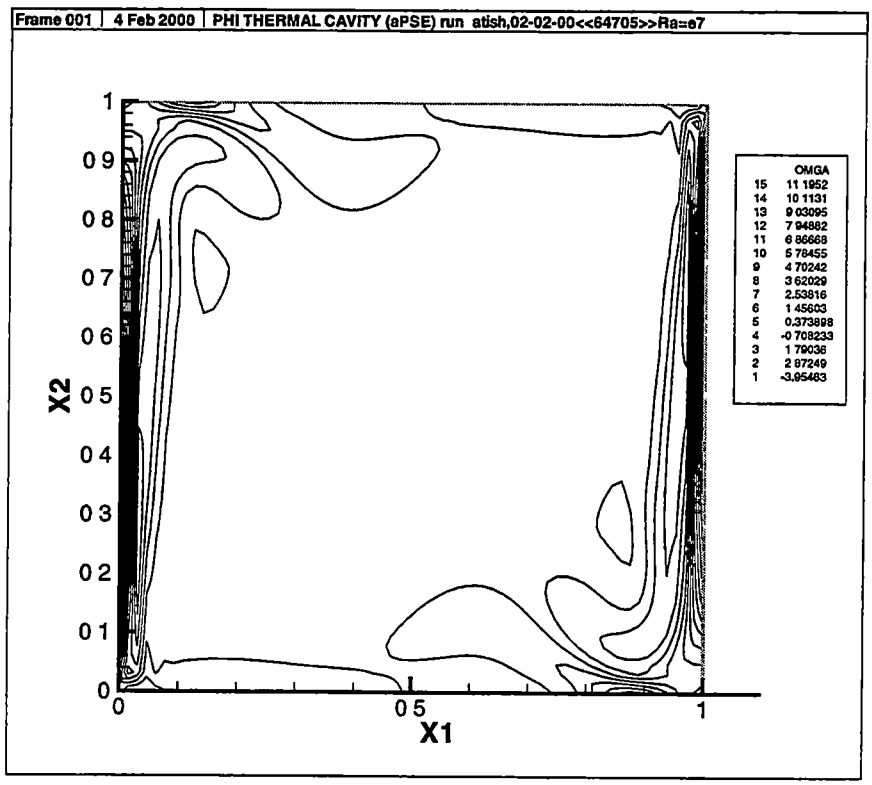


Figure C.15 Vorticity contour, Ra=$10^7$, 64x64 mesh

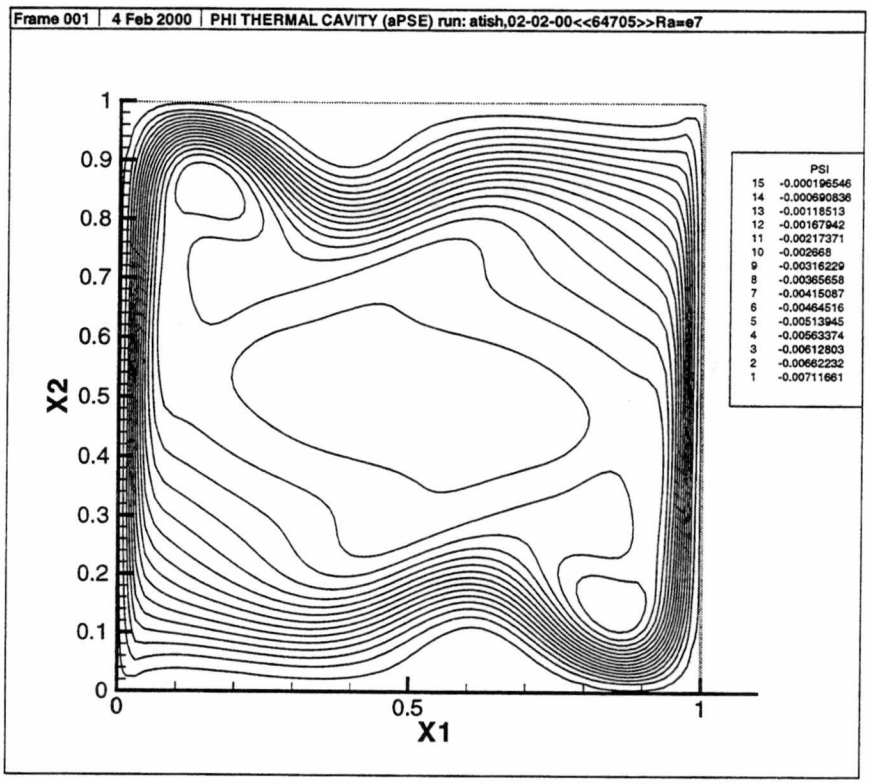


Figure C.16: Streamlines, Ra=$10^7$, 64x64 mesh

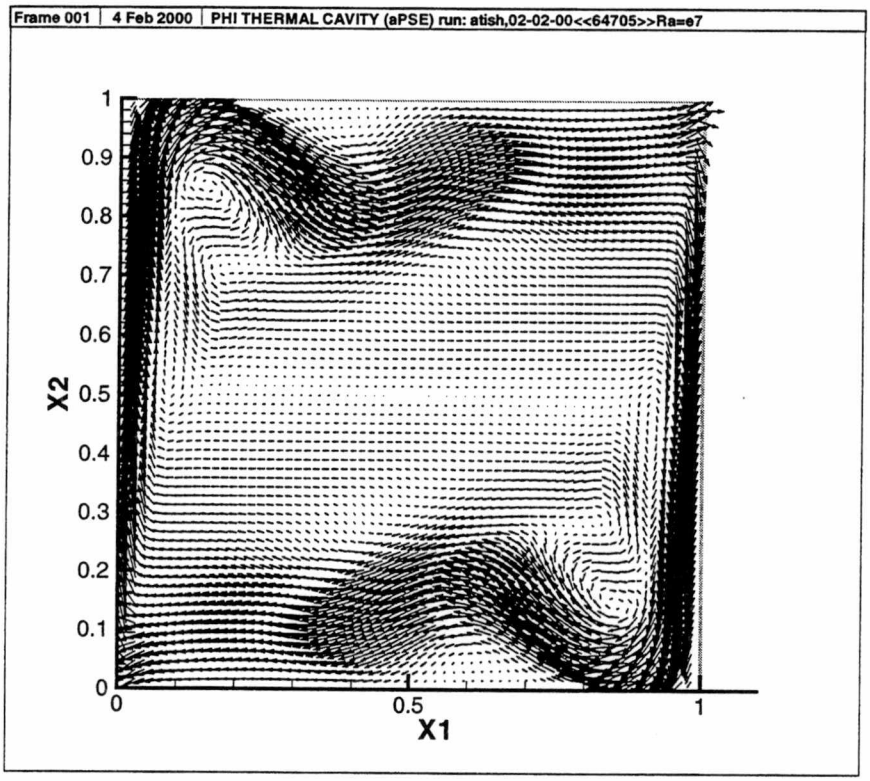


Figure C.17: Velocity Field, Ra=$10^7$, 64x64 mesh

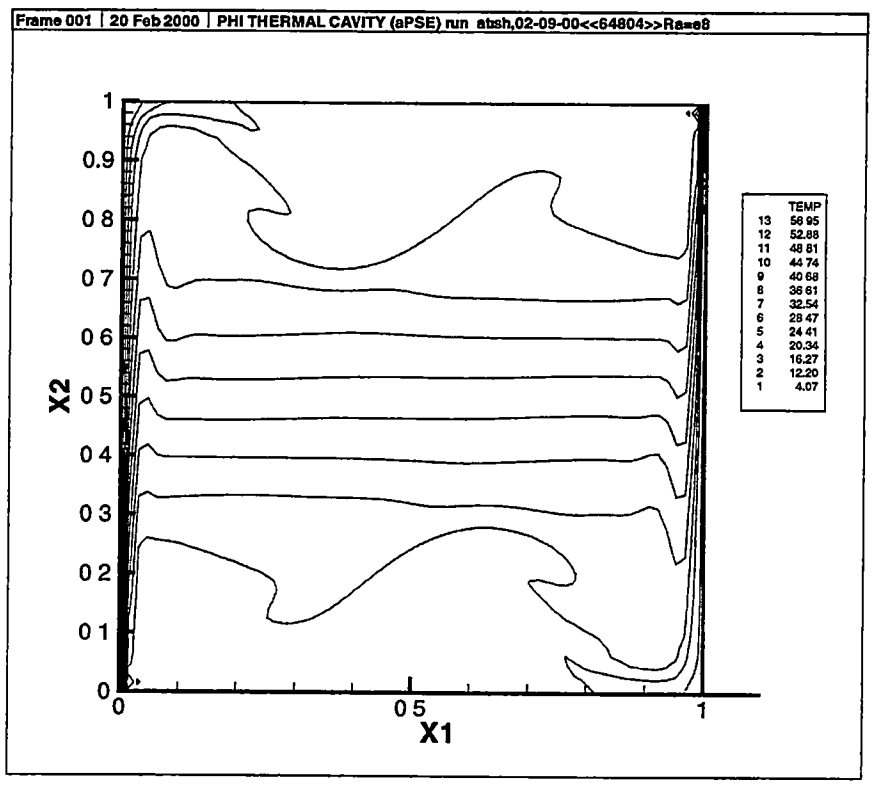


Figure C.18. Temperature contour, Ra=$10^8$, 64x64 mesh

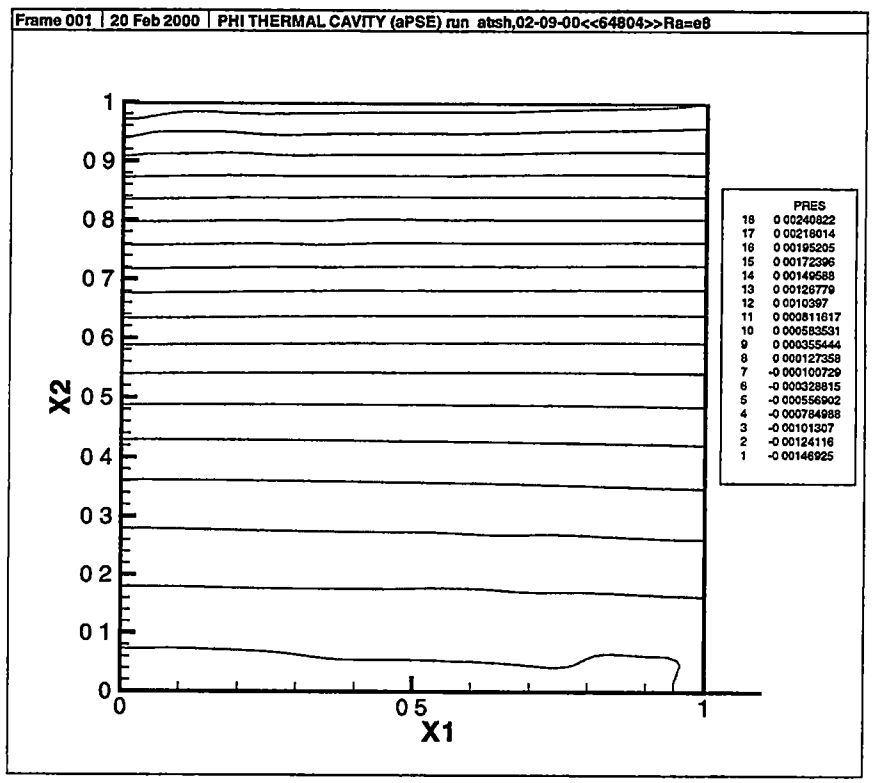


Figure C.19: Pressure contour, Ra=$10^8$, 64x64 mesh

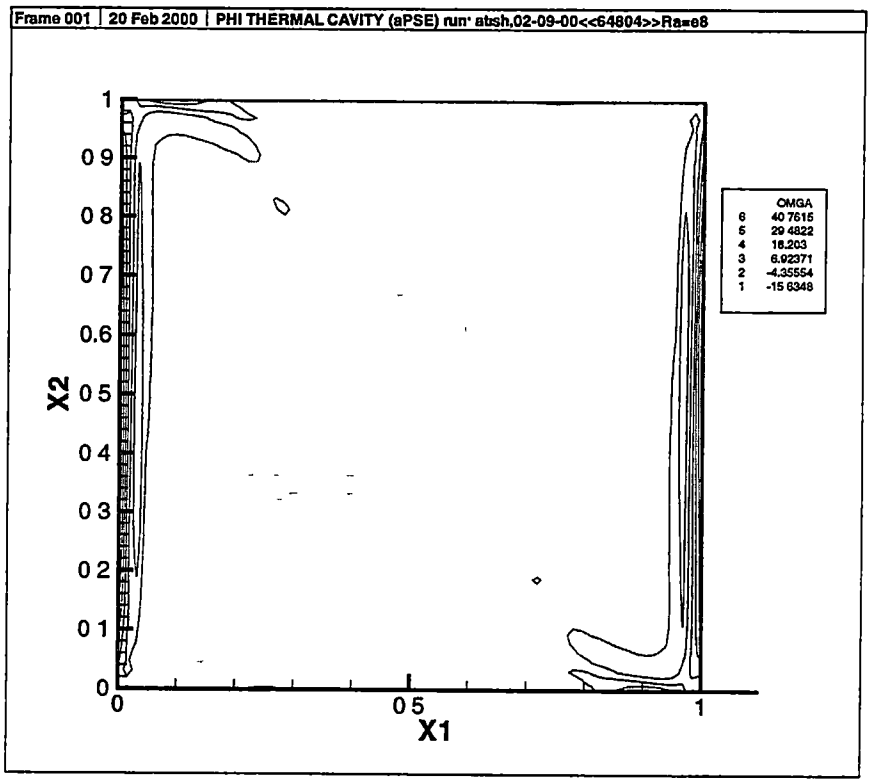


Figure C 20. Vorticity contour, Ra=$10^8$, 64x64 mesh

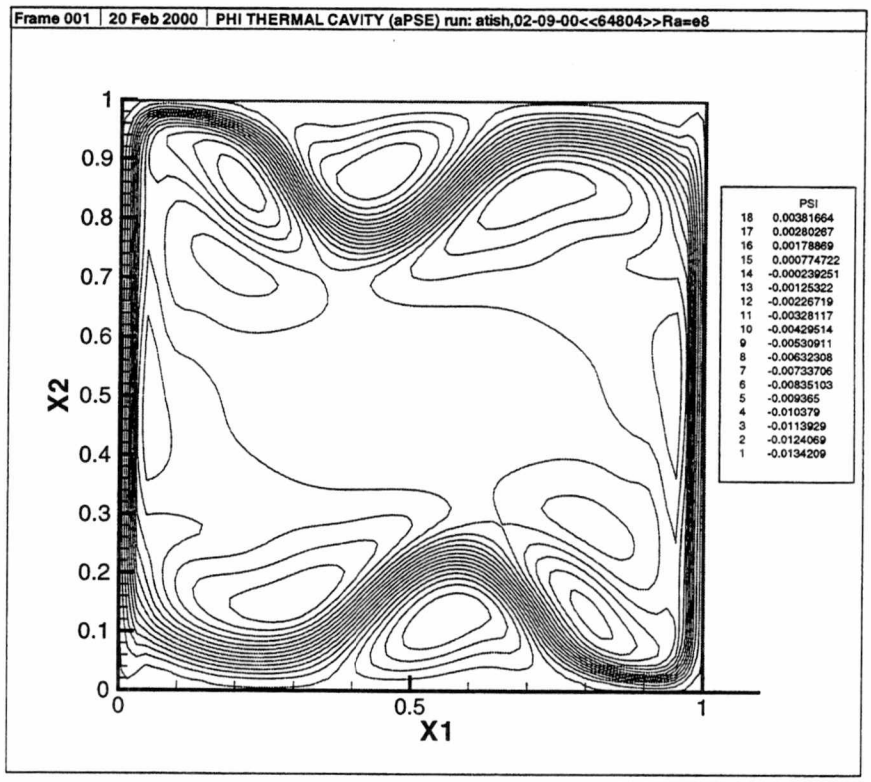


Figure C.21: Streamlines, Ra=$10^8$, 64x64 mesh

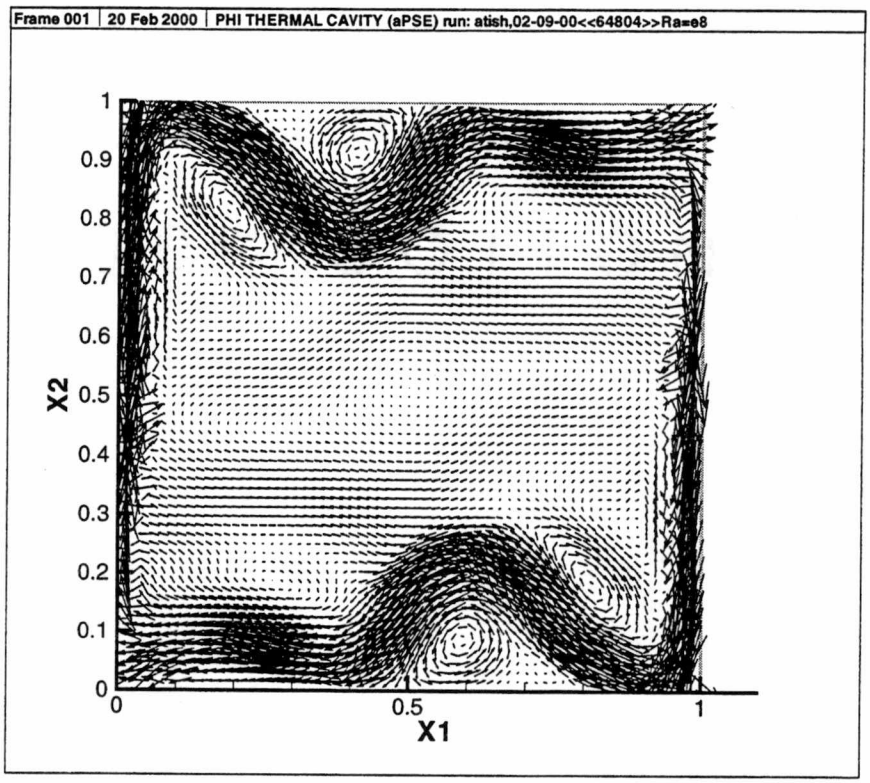


Figure C.22: Velocity Field, Ra=$10^8$, 64x64 mesh

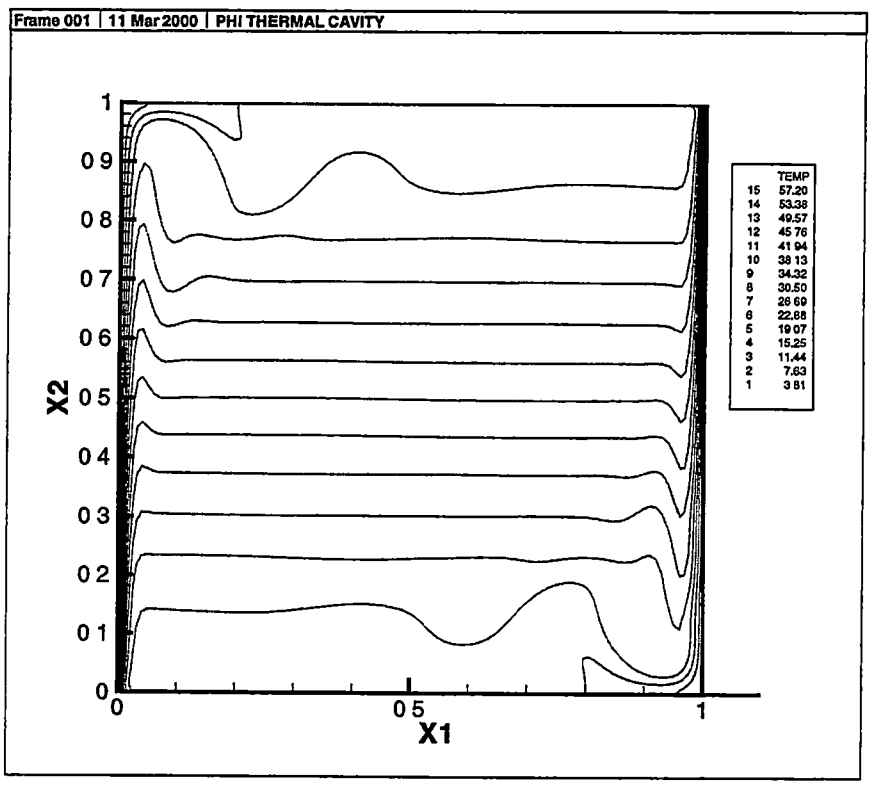


Figure C.23· Temperature contour, Ra=$10^8$, 128x128 mesh

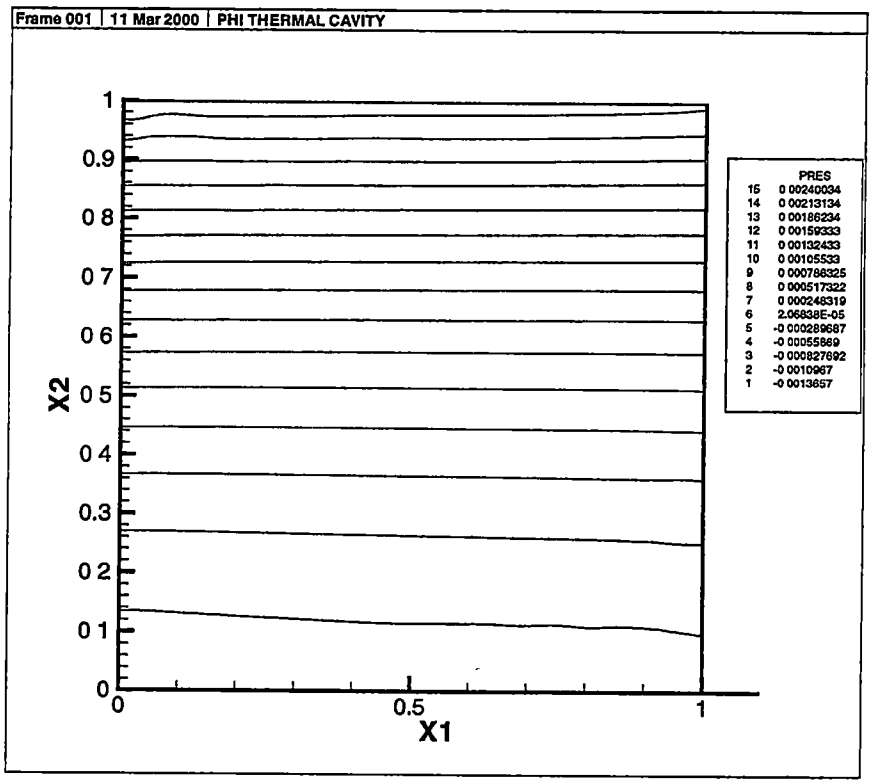


Figure C.24. Pressure contour, Ra=$10^8$, 128x128 mesh

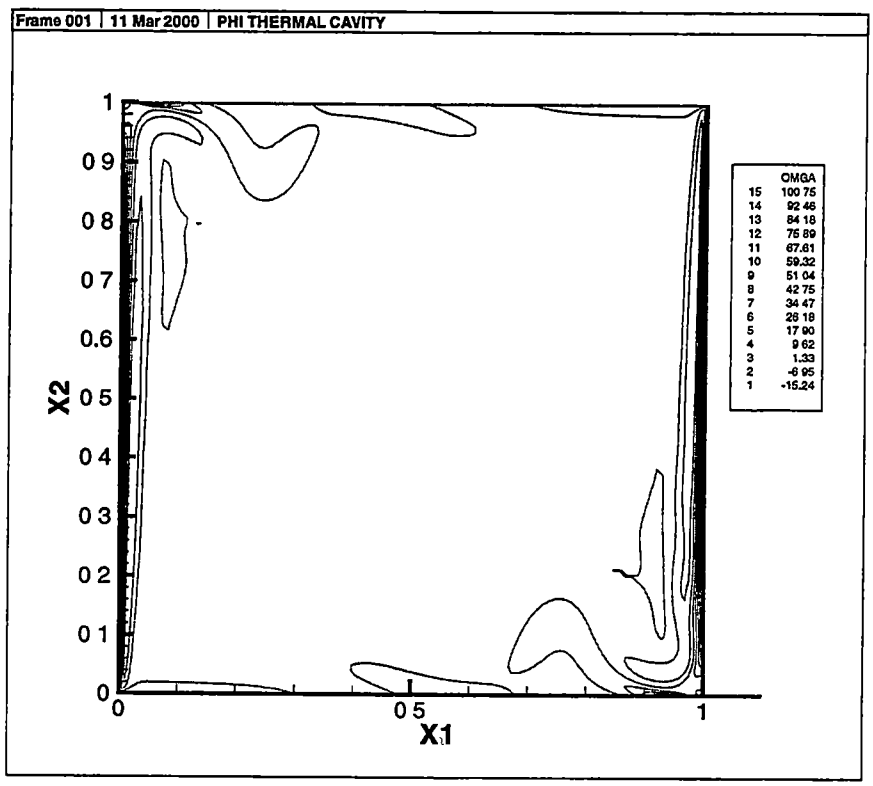


Figure C.25: Vorticity contour, Ra=$10^8$, 128x128 mesh

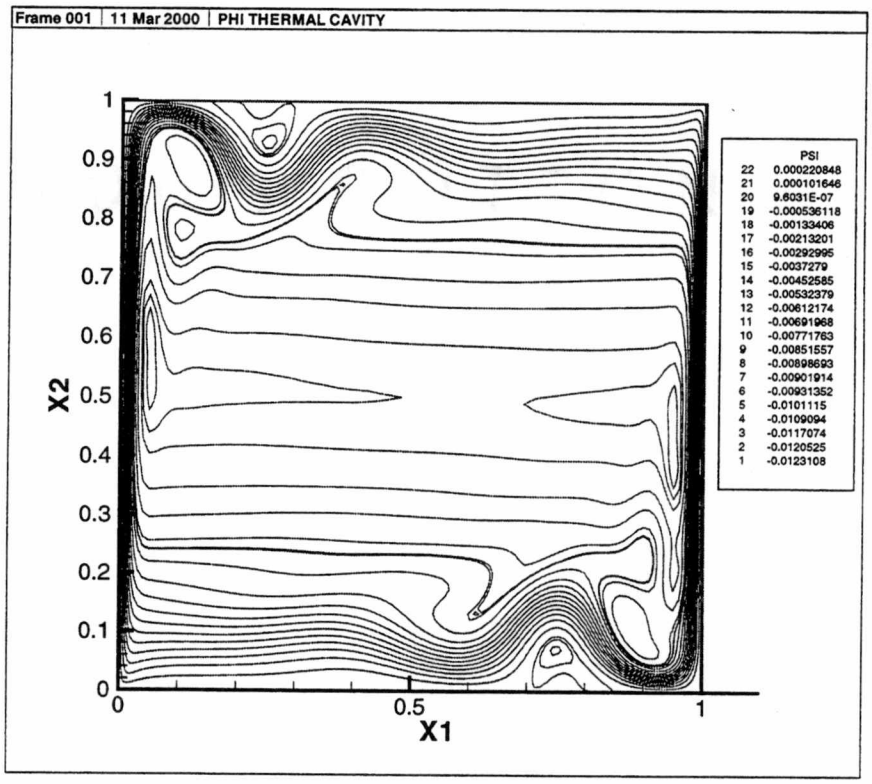


Figure C.26: Streamlines, Ra=$10^8$, 128x128 mesh

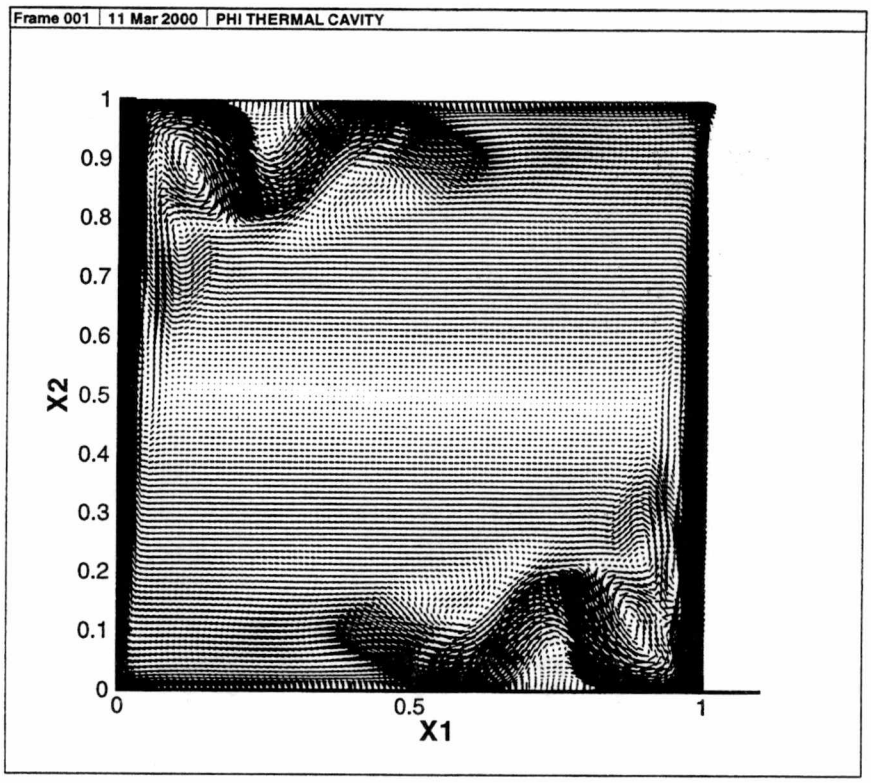


Figure C.27: Velocity Field, Ra=$10^8$, 128x128 mesh

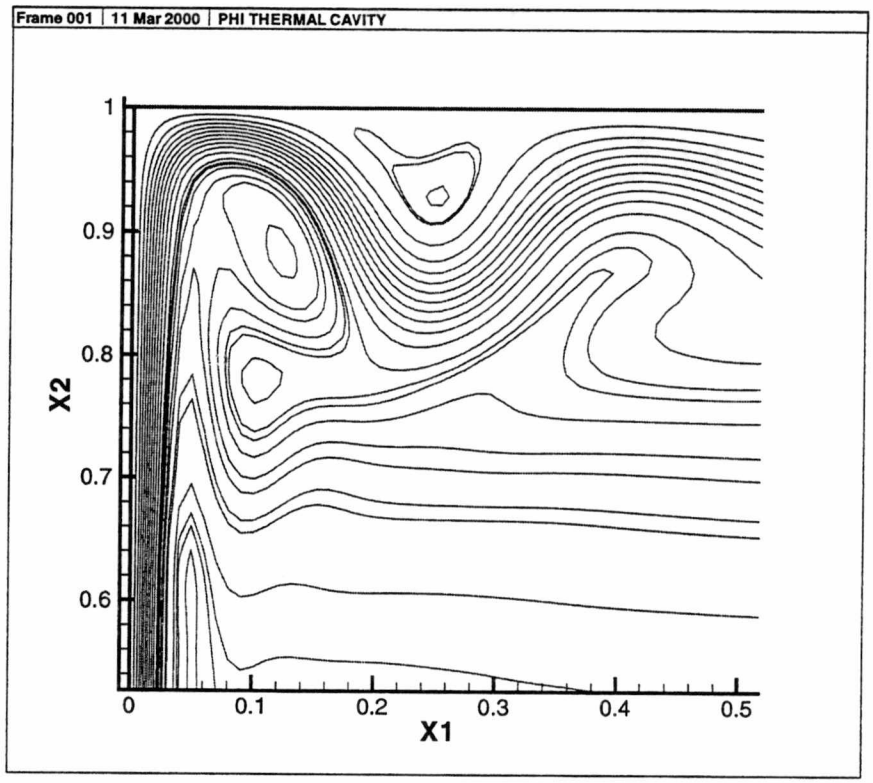


Figure C.28: Streamlines (X1=0.25,X2=0.75), Ra=$10^8$, 128x128 mesh

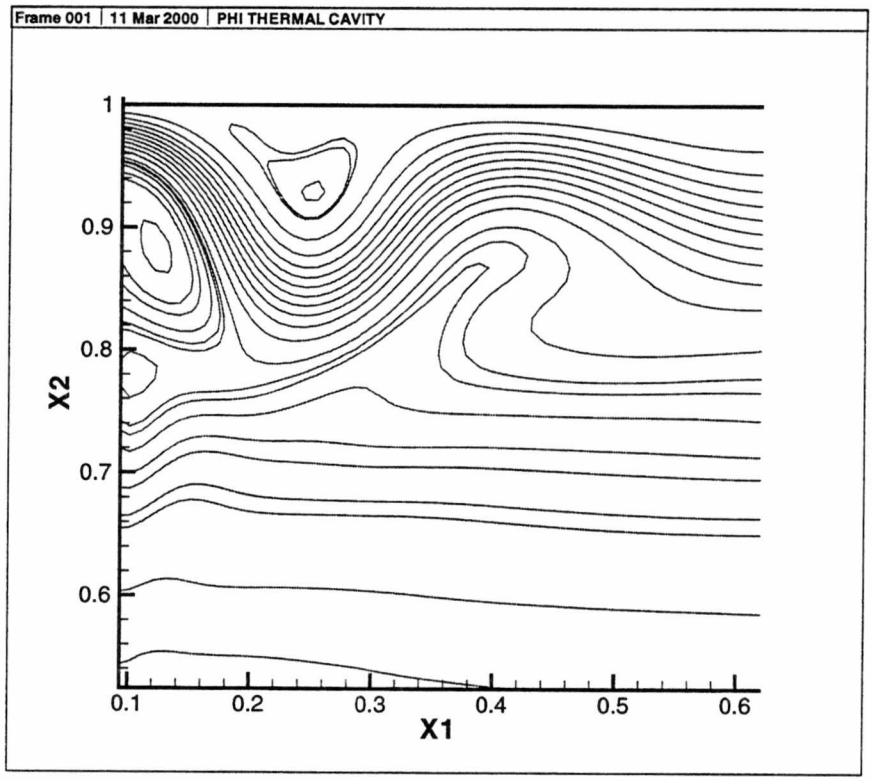


Figure C.29: Streamlines (X1=0.35,X2=0.75), Ra=$10^8$, 128x128 mesh

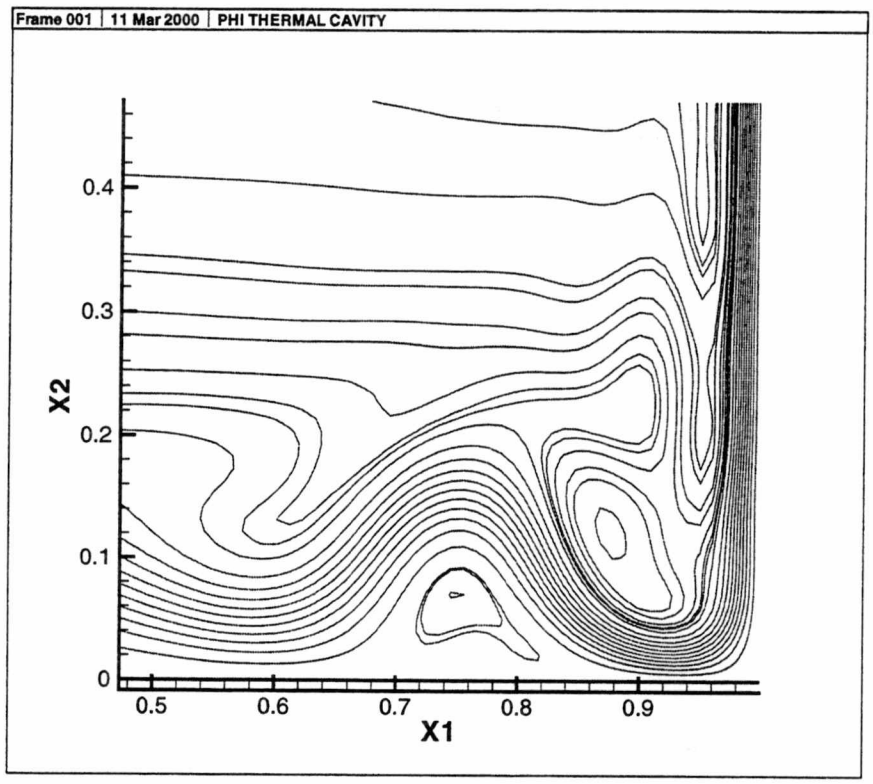


Figure C.30: Streamlines (X1=0.75,X2=0.25), Ra=$10^8$, 128x128 mesh

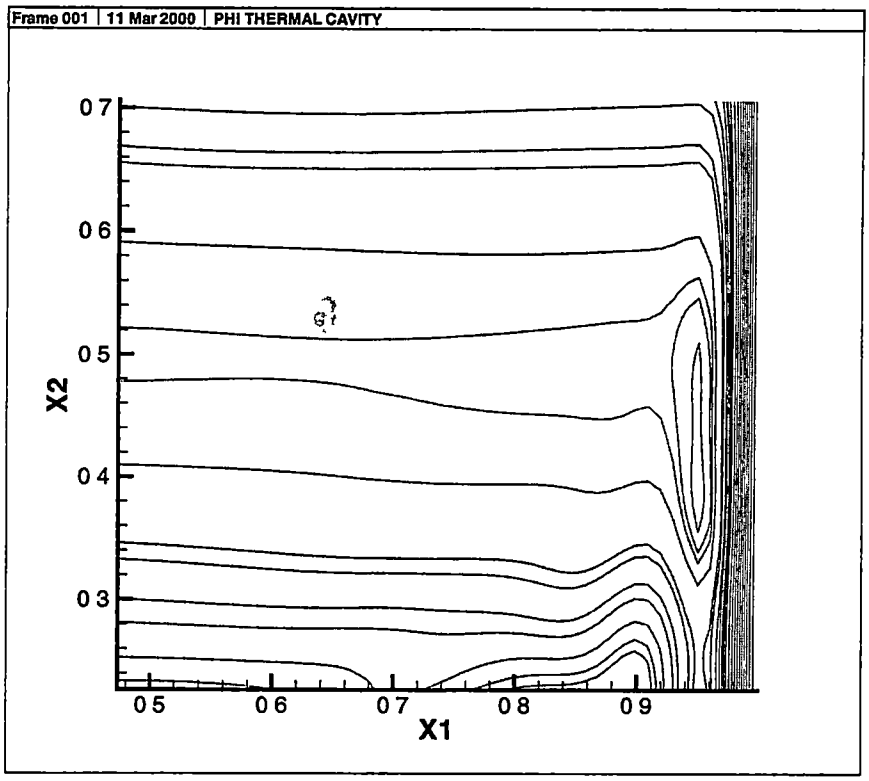


Figure C.31: Streamlines (X1=0.75,X2=0.45), Ra=$10^8$, 128x128 mesh

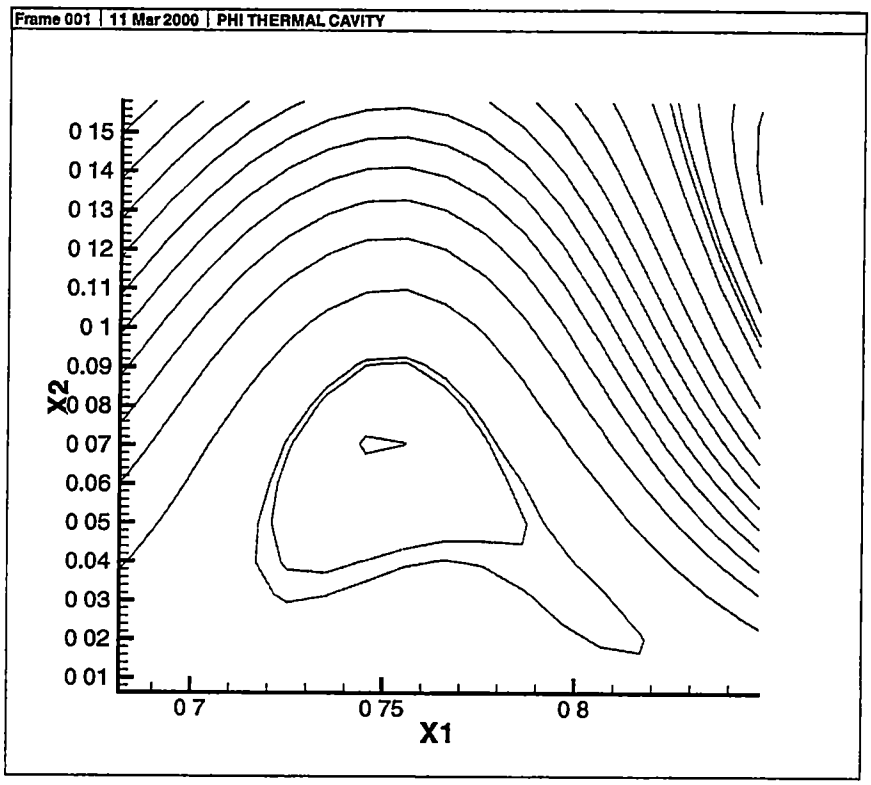


Figure C.32: Streamlines (X1=0 75,X2=0.08), Ra=$10^8$, 128x128 mesh

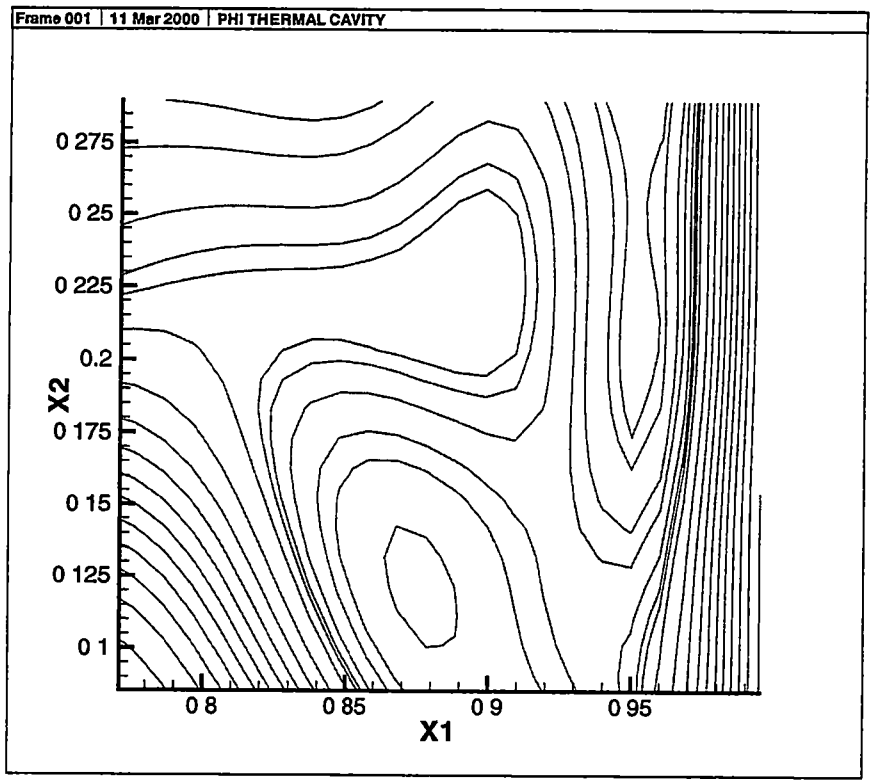


Figure C.33. Streamlines (X1=0.85,X2=0.175), Ra=$10^8$, 128x128 mesh

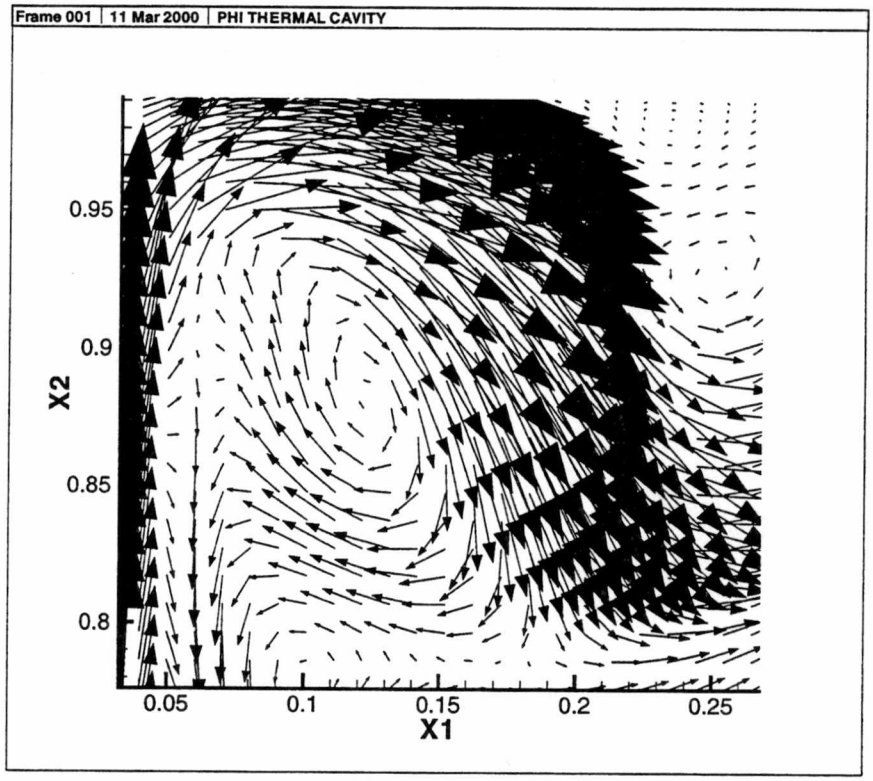


Figure C.34: Velocity Field (X1=0.15,X2=0.85), Ra=$10^8$, 128x128 mesh

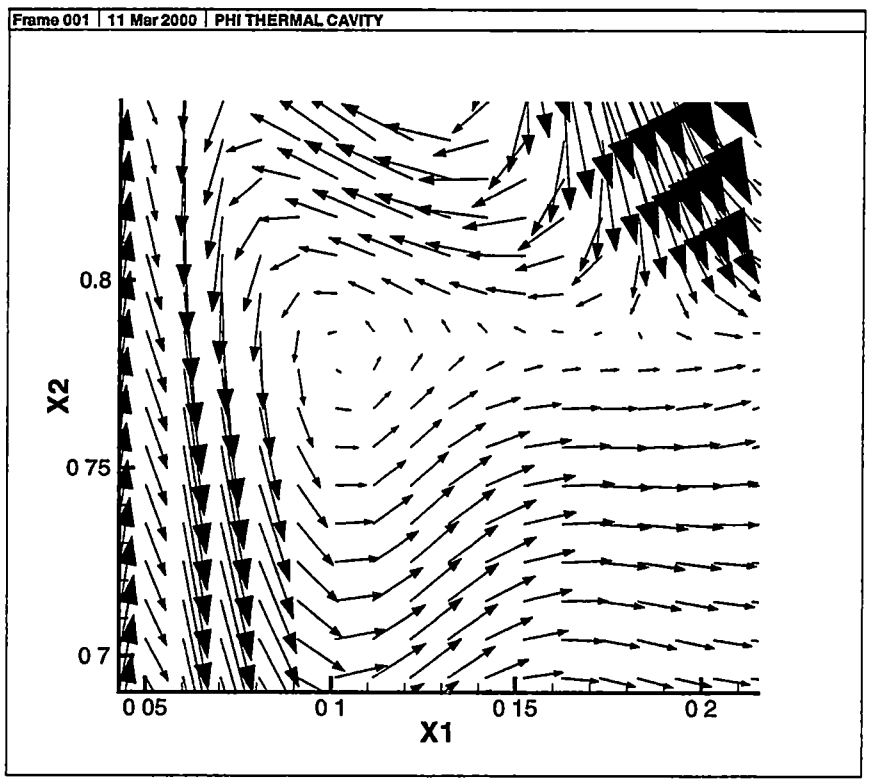


Figure C.35: Velocity Field (X1=0.1,X2=0.75), Ra=$10^8$, 128x128 mesh

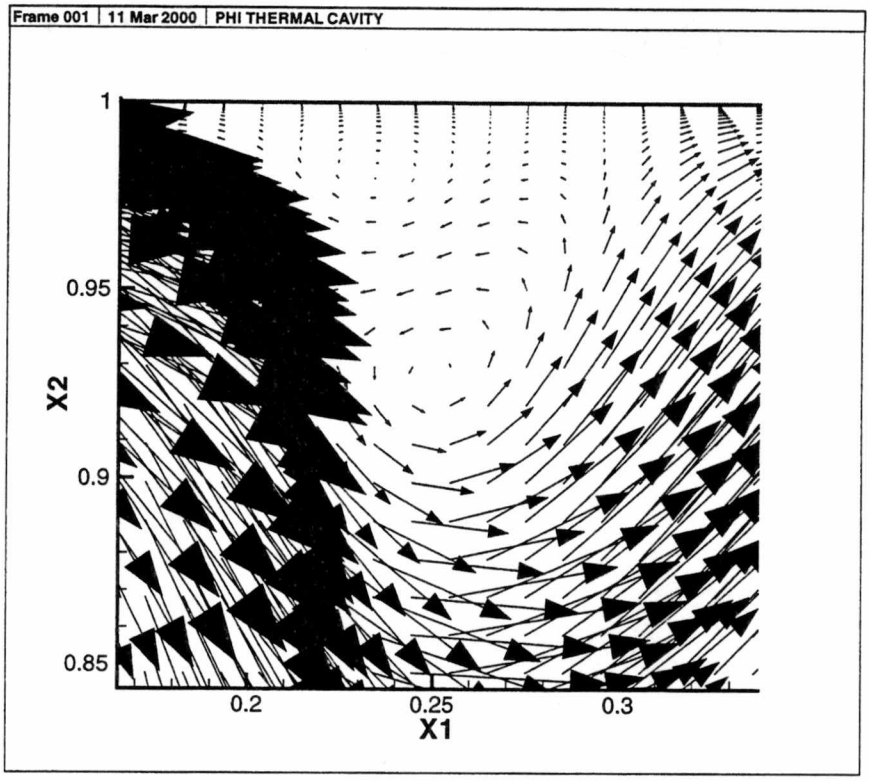


Figure C.36: Velocity Field (X1=0.25,X2=0.9), Ra=$10^8$, 128x128 mesh

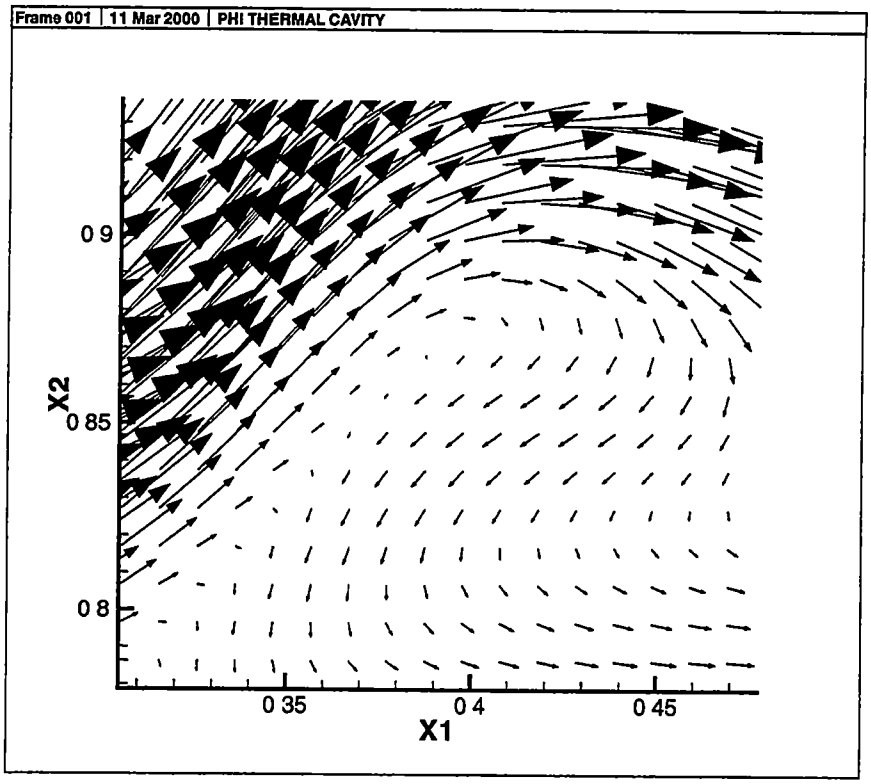


Figure C.37: Velocity Field (X1=0.4,X2=0.85), Ra=$10^8$, 128x128 mesh

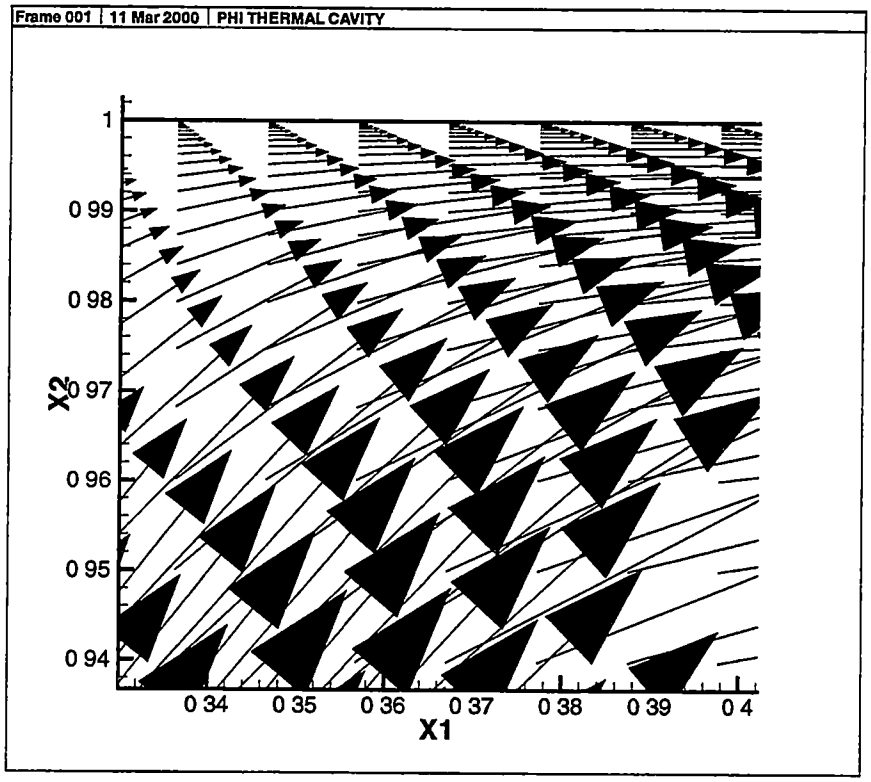


Figure C.38 Velocity Field (X1=0.37,X2=0.97), Ra=$10^8$, 128x128 mesh

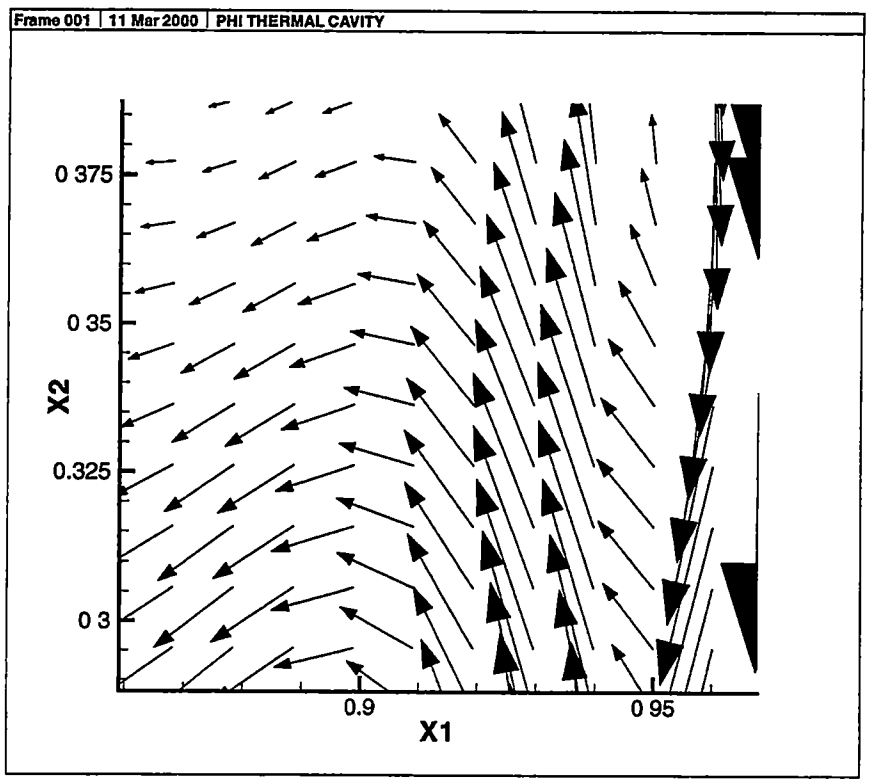


Figure C.39: Velocity Field (X1=0.9,X2=0.35), Ra=$10^8$, 128x128 mesh

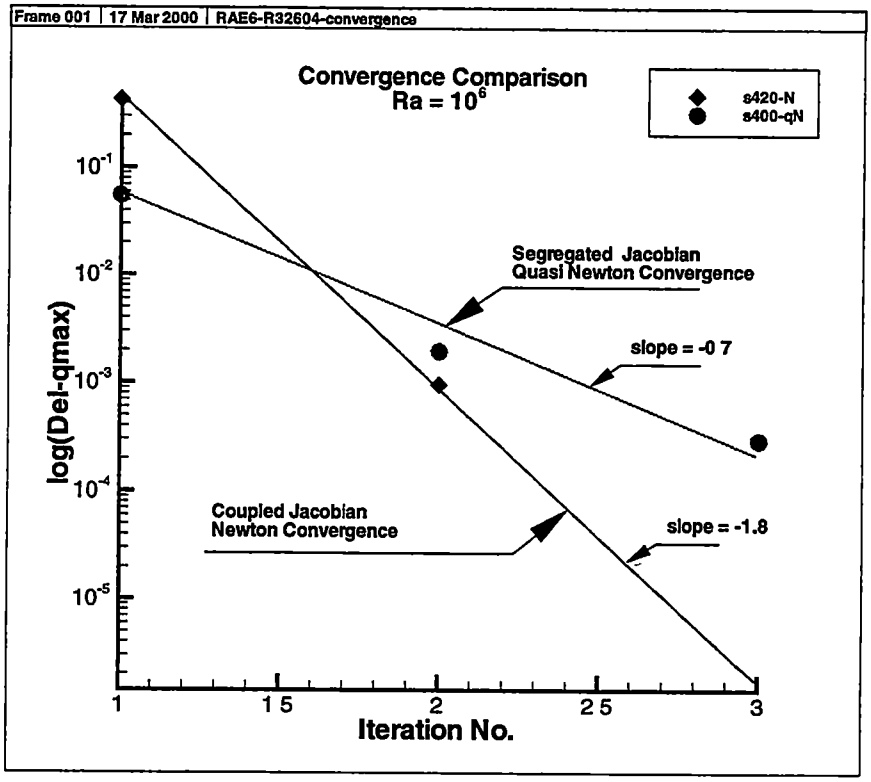


Figure C 40: Verification of convergence, Ra=$10^6$

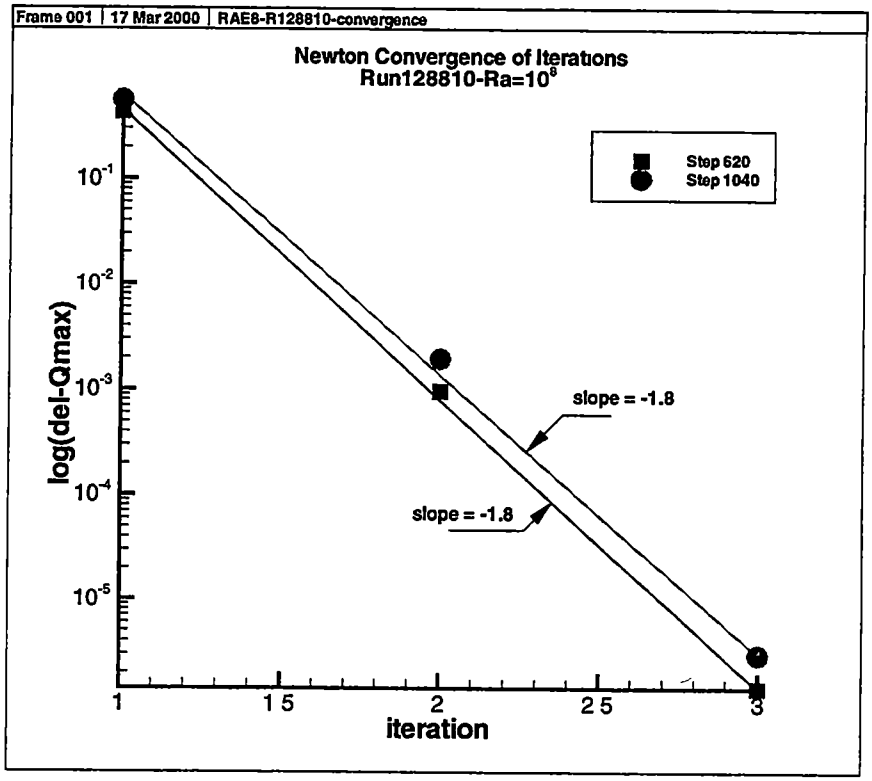


Figure C.41· Verification of convergence, Ra=$10^8$

# Vita

Atish Jyoti Mitra was born in Calcutta, India on April 8th, 1970. He attended South Point High School, Calcutta from where he passed the Higher secondary examination in July 1988. From August 1988 he studied at the Department of Mechanical Engineering, Jadavpur University, Calcutta and received the Bachelor of Mechanical Engineering (B.M.E.) degree in July 1992.

After a five-year tenure at an private sector engineering concern in Calcutta, he accepted a graduate assistantship at the University of Tennessee, Knoxville and began study towards a Master Of Science degree in Engineering Science with an emphasis on Computational Fluid Dynamics (CFD).

Mr Mitra is at present studying towards a PhD in Mathematics at the University of Tennessee, Knoxville.