INERTIAL MODES OF THE EARTH'S FLUID CORE

MD. KAMRUZZAMAN
B. Sc. in Physics, Shahjalal University of Science and Technology, 2008
M. S. in Physics, Shahjalal University of Science and Technology, 2009
Postgraduate Diploma in ESP, Abdus Salam ICTP, 2011

A Thesis
Submitted to the School of Graduate Studies
of the University of Lethbridge
in Partial Fulfillment of the
Requirements for the Degree

MASTER OF SCIENCE

Department of Physics and Astronomy
University of Lethbridge
LETHBRIDGE, ALBERTA, CANADA

INERTIAL MODES OF THE EARTH'S FLUID CORE

MD. KAMRUZZAMAN

Date of Defense: March 25, 2015

Dr. Behnam Seyed-Mahmoud Supervisor	Associate Professor	Ph.D.
Dr. David Kaminski Thesis Examination Committee Member	Associate Professor	Ph.D.
Dr. Steve Patitsas Thesis Examination Committee Member	Associate Professor	Ph.D.
Dr. Craig Coburn Chair, Thesis Examination Committee	Associate Professor	Ph.D.

Dedication

This thesis is dedicated to my parents.

Abstract

The Earth's outer core is a rotating ellipsoidal shell of compressible, stratified and self-gravitating fluid. As such, in the treatment of geophysical problems a realistic model of this body needs to be considered. In this work we consider a compressible and stratified fluid core model with different stratification parameters, related to the local Brunt- V*äisälä* frequency. We use the three potential description (3PD) to study the effects of the core's density stratification on the frequencies of some of the inertial modes of this model. As a first approximation, however, we ignore the ellipticity of the core's figure. The 3PD scheme describes the exact linearized dynamics of rotating, self-gravitating, stratified, compressible and inviscid fluids. The inertial modes of the core are the long-period free oscillations which have the Coriolis force as their restoring force. Historically an incompressible and homogeneous fluid sphere is considered to study these modes and analytical solutions are known for the frequencies and the displacement eigenfunctions of this model. We show that the effects of non-neutral density stratification may be significant, and the frequencies of these modes may change from model to model depending on the stratification parameters and the spatial structure of the modes. We further explore the role of the characteristic surfaces on the excitation and structure of these modes.

Acknowledgments

I am indebted to my supervisor Dr. Behnam Seyed-Mahmoud for his supervision, encouragement, suggestion and discussion at every stage of, for affording me an opportunity to study at the University of Lethbridge, this research work. I am grateful to my supervisor for the intense training in modelling techniques using FORTRAN in which I have become so confident.

I would also like to thank my supervisory committee members, Dr. Steve Patitsas and Dr. David Kaminski, for sparing time to have meetings during the course of this work during the past two years. I thank you all for the constructive suggestions and encouragements. I would like to thank the rest of the staff of the Department of Physics and Astronomy and my research mates Ali Moradi and Hossein Naseri for constructive discussions and encouragements.

I would like to thank the administration staff for helping me in various ways during the last two years. Special thanks to my friends at U of L for their warm friendship we have experienced together during our stay at U of L.

I wish to express my gratitude to the NSERC, and the School of Graduate Studies for their kind hospitality and for providing me facilities at the University of Lethbridge during the M.Sc. program which made this work possible.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

As long as a branch of knowledge offers an abundance of problems, it is full of vitality

– David Hilbert.

The structure of the Earth's interior is separated into three main layers, the crust, the mantle and the core, based on their chemical composition or material properties. The deepest region from the Earth's surface is the core which consists of a solid inner core with radius of 1221.5 km and a liquid outer core which is responsible for most of the Earth's magnetic field, and has a thickness of about 2258.5 km. Unlike the Earth's surface, the core is not directly accessible for study, and much of its properties is still unknown. The theoretical and observational studies of ray seismology, free oscillation, tides and wobble/nutation have established the fluidity of the outer core and the solidity of the inner core. These studies have generated a great deal of information about the core. Jeffreys [1] established the fluidity of the Earth's core by showing that the presence of a large zero-rigidity core surrounded by a mantle with a rigidity from ray seismology. In 1981, Dziewonski and Anderson [2] established the preliminary reference Earth model (PREM) from a compilation of seismic, free oscillation and nutation data. This model provides a basic reference state for many Earth parameters for consistent studies of the Earth, and is widely accepted as a reference Earth model.

On the other hand, some properties of the core cannot be firmly established from the theoretical and observational studies of ray seismology. One of the most important is the

stability parameter β, a dimensionless quantity that measures the deviation of the equilibrium density from neutral stratification. The stratification is the process leading to the formation or deposition of layers. The layers are organized under gravity so that the higher densities are found beneath lower densities. Pekeris and Accad [3] introduced β which is related to the density gradient and is not directly observable. This profile is, therefore, poorly controlled, uncertain and varies from one model of the Earth to another. The stability parameter is related to the square of the local Brunt-Väisälä frequency, N^2, which is a parameter characteristic of a stratified fluid. This frequency refers to the oscillations of a parcel of fluid about its equilibrium position. If the value of N^2 is positive ($\beta < 0$) then a parcel of fluid performs small oscillations about its original position with frequency N, and the density profile will be stably stratified. If the value of N^2 is negative ($\beta > 0$), i.e., N becomes an imaginary, then a parcel of fluid exhibits an exponential decay or growth, and it will be unstably stratified. If $N^2 = 0$ ($\beta = 0$) then the density profile will be neutrally stratified.

The sudden radiated energy caused by a very large earthquake can oscillate into the whole Earth. The oscillation frequencies are determined by the elastic properties and the structure of the Earth's interior [4]. The two long period oscillations, one of a period of 57 min and another of a period of 100 min, of the Kamchatka earthquake of 1952 were discovered by Benioff et al., [5]. They suggested that the 57 min oscillation represents one of the free spheroidal oscillations of the Earth. The Chandler oscillation was discovered in 1891 by S. C. Chandler from available astronomical data, and has a period of about 435 days. The spectrum of the Earth's free oscillations is divided into two distinct groups: (1) the short period free oscillations with elasticity as their primary restoring force which have periods shorter than a few hours, and (2) the long-period free oscillations, with periods longer than half a day, and are considerably affected by the rotation and in some cases the ellipticity of the Earth. There are four types of long-period free oscillations that could be excited in the Earth:

(i) the wobble and nutation modes depend on the Earth's composition and shape;

(ii) the translational modes of the solid inner core, known as the Slichter modes, with gravity as their restoring force;

(iii) the gravity modes with negative buoyancy as their restoring force;

(iv) inertial modes, which depend on the Coriolis force as their restoring force.

In astrophysical and geophysical problems, fluid rotation often plays an important part. Consider an ideal (homogeneous, incompressible and inviscid) fluid in a steady state of rotation about the z-axis, and imagine what would happen if a tiny volume is horizontally displaced with a low velocity, $\vec{v}$. Since at all points in the fluid an equilibrium would be maintained between the centrifugal force and pressure gradient force, the motion of the chosen particle would be affected solely by the Coriolis force, $\vec{F}$. With respect to the rotating frame of reference, the particle would be continuously deflected around a circular trajectory (see Figure 1.1). This type of oscillation is known as an inertial wave. These waves, also known as inertial oscillations or inertial modes of oscillations, travel through the interior of the fluid, and exist in any rotating fluid. The properties of a rotating fluid such as the compressibility and shape of the container may significantly affect on the frequencies of these modes [6]. So the identification of these modes have an impact on our understanding of the interiors of rotating bodies like the Earth's fluid core, other planets, and rotating stars.

McElhinny et al., [7] studied the palaeomagnetic data and their work supported that the Earth's geomagnetic field has existed for at least 3.5 billion years. It is now understood that the Earth's magnetic field is generated in the Earth's fluid core, a process called the geodynamo. The commonly accepted theory for the source of energy to power the geodynamo is based on thermal convection in the core. Elsasser [8] has given an explanation of the geodynamo based on the thermal convection, and has suggested that convection in the core may be compositional. However, if Earth's core is predominantly neutrally stratified

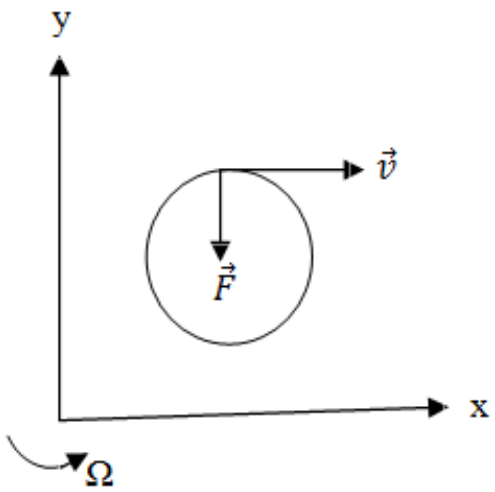

Figure 1.1: The restoring effect of the Coriolis force on a displaced particle of fluid.

then it is unlikely that thermal convection would produce enough energy to drive the geodynamo [9]. In the absence of a convection driven dynamo, an elliptical instability in the core may be capable of driving the dynamo [9]. Elliptical instability may occur in planetary fluid cores due to the gravitational pull of secondary bodies (e.g., moons) [10, 11, 12, 13]. The elliptical instability is excited as a result of interactions between two or more of the inertial modes of a contained fluid [9, 14]. Hence, the inertial modes in the Earth's fluid core may maintain the core dynamo.

The equation for oscillations in an homogeneous, incompressible, rotating, and inviscid fluid can be traced back to the works of Poincaré [15], Bryan [16] and Hough [17] and it's termed the Poincaré equation. This equation is a hyperbolic equation subject to a boundary condition, a condition that makes the equation ill-posed in the sense of Hadamard that it admits closed form solutions in certain geometry such as an ellipsoid or a cylinder. Stewartson and Roberts [18] pointed out that there are no existence and uniqueness theorems for hyperbolic equation in that case so that the existence of continuous solutions appears doubtful [19]. Bryan [16] established a method of solution using a double transformation to obtain an "oblate spheroidal" coordinate system, which allowed separable solutions to the Poincaré problem. Solutions were worked out by Kelvin [20] for a cylinder and by

Bryan [16] for a sphere and an ellipsoid.

Kudlick [21] extended the linear theory to include the effects of viscosity by using superposition of the natural oscillatory modes of the inviscid problem that had been corrected to first order for the effects of viscosity. Stewartson and Rickard [22] obtained an expansion in power of the shell thickness starting with a solution to the thin shell of the Poincaré equation. This asymptotic solution has a singularity on the inner boundary. An excellent summary of rotating fluid dynamic theory was provided by Greenspan [23]. He considered rotating fluids exclusively, and included a comprehensive investigation of viscosity effects. Although the amplitude of the inertial modes depends on the viscosity, the frequencies of these modes are independent of viscosity.

The geometry of the Earth's fluid core (i.e., a thick shell) presents mathematical difficulties in the theory of inertial waves [22]. Aldridge [24] observed that there are fundamental difficulties in obtaining the analytical solution of the Poincaré equation in a thick shell, although his experimental results suggested that such waves exist in the thick shell configuration [19]. Numerical solutions are, therefore, investigated whenever possible. Rieutord [25] and Henderson [26] applied numerical methods to approximately solve for the inertial modes of a thick, rotating, spherical shell. Rieutord [25] investigated the inertial modes of a shell by assuming an incompressible fluid of small viscosity. Henderson [26], however, considered an ideal fluid and used a finite element method. He placed the mesh along the characteristic surfaces to solve for what he termed the weak solutions of the Poincaré problem.

In order to solve the dynamical equations in the fluid core using the traditional approach, the vector field is represented by its spheroidal and toroidal components, and all the scalar fields are represented by the spherical harmonics [27, 28, 29, 30]. Although this may be a favorable approach for cases where Earth's rotation may be ignored, it is not suitable in a rotating case because it leads to the infinitely coupled chains of equations, and the periods of the oscillations of the fluid body then depend on heavy truncation of these

coupled chains. In addition, Rogister and Valette [31] pointed out the possible influence of the rotational modes on the dynamics of the Earth's liquid outer core. They consider core models with a different stability parameter to study these effects. They agree that the traditional approach is not adequate to fully describe these modes, hence they label the modes they computed using severe truncations as pseudo-modes. They show that these modes interact with at least three of the Earth's wobble and nutation modes, the Chandler Wobble, the Free Inner Core Nutation and the Free Core Nutation modes.

Recently, Seyed-Mahmoud and Rochester [32] developed the 3PD (Three Potential Description) that describes the exact linearized dynamics of rotating, self-gravitating, stratified, compressible and inviscid fluids, and used a Galerkin method to solve the dynamical equations. Seyed-Mahmoud and Rochester [32], and Seyed-Mahmoud et al., [6] considered a compressible and a neutrally stratified fluid to study the inertial modes of rotating and self-gravitating fluids of a spherical and spherical shell geometries using the 3PD. Their results show that for a spherical geometry, to the degree they investigated, the frequencies of the inertial modes of a neutrally stratified fluid may be significantly affected by such properties as the fluid compressibility. They also pointed out that the eigenfunctions of these modes are similar to those of the Poincaré model.

In this thesis, we use the 3PD to investigate the influence of non-zero β on the frequencies of the inertial modes for several models based on the PREM core model but modified so that the stratification parameter is different for each model. We first derive the web of characteristics, which gives more information about the eigenfunctions of these modes, as functions of frequency and stratification for compressible and inviscid fluids. We show that, depending on the size of β, some modal frequencies and eigenfunctions are practically unaffected by stratification, some are adjusted and some modes may disappear, that is, we have not been able to locate them.

This thesis is organized into five chapters as follows: In chapter 2 as we present the governing equations of the fluid core with the boundary conditions and the three potential

description (3PD). We also derive the web of characteristics for compressible and inviscid fluids, and discuss the core models. In chapter 3, we introduce the Galerkin method applicable to a system of simultaneous partial differential equations subject to boundary conditions. This method is then applied to the Poincaré equation and to the 3PD. These equations are expanded and the orthogonality relation among spherical harmonics is used to integrate the equation with respect to the radial component. Then, the matrix representations of the Poincaré equation and the 3PD are discussed. In chapter 4, we discuss the results of this work . First, the procedures for finding of the non-dimensional frequencies of the inertial modes of the fluid core are presented. We then present and describe the non-dimensional frequencies and displacement eigenfunctions of these inertial modes for the modified sphere and spherical shell core models with different stability parameters. Finally, chapter 5 contains the conclusions of this work.

Chapter 2

Formulation

In every mathematical investigation, the question will arise whether we can apply our mathematical results to the real world.

– V.I. Arnold.

In this chapter, we will first discuss the governing equations of the fluid core including the boundary conditions. In section 2.4, we will show the derivation of the web of characteristics which will give useful information about the nature of the eigenfunctions. We will also discuss the core models that will be used to investigate the inertial modes of both a fluid sphere and a spherical shell.

2.1 Governing Equations of the Fluid Core

In order to study the dynamics of the Earth's fluid core, we take the reference state as being one of hydrostatic equilibrium in a coordinate system which rotates with a constant angular velocity

$$\vec{\omega}_r = \Omega \hat{e}_3, \tag{2.1}$$

where Ω is the rate of rotation of the Earth and $\hat{e}_3$ is a unit vector along the rotation axis. As a first order approximation, we consider a spherical geometry in which $\vec{g}_0 = g_0 \hat{r}$ [33, 32]. The fluid core is assumed as inviscid. It has a low viscosity (typically the numerical value of kinematic viscosity: $\eta = 9.1 \times 10^{-8} m^2 s^{-1}$) [34], thus making use of an inviscid fluid a

good first approximation. The periods of the inertial modes are not affected by viscosity, however, their amplitudes are damped, much like a damped harmonic oscillator, at the rate $E^{\frac{1}{2}}$ [23], where E is the Ekman number. The Ekman number is, a dimensionless number, defined as $E = \frac{\eta}{\Omega L^2}$, where L characterize the typical length.

In this reference frame, the density ρ_0, the pressure p_0, the gravitational potential W_0 and the acceleration due to gravity $\vec{g}_0$ are related [35, 36] by

$$\nabla p_0 = \rho_0 \vec{g}_0, \tag{2.2}$$

$$\vec{g}_0 = \nabla W_0, \tag{2.3}$$

$$\nabla^2 W_0 = -4\pi G\rho_0 + 2\Omega^2, \tag{2.4}$$

$$\nabla \rho_0 = (1-\beta)\rho_0 \frac{\vec{g}_0}{\alpha^2}, \tag{2.5}$$

where G, α and β are the gravitational constant, the local compressional wave (P-wave) speed and the stability parameter respectively. The stability parameter measures the deviation of the equilibrium density from neutral stratification. The stability parameter is related to the square of the local Brunt-Väisälä frequency N^2, rendered dimensionless by dividing it by $4\Omega^2$, as

$$N^2 = -\frac{\beta g_0^2}{4\Omega^2\alpha^2}. \tag{2.6}$$

A positive value of N^2 (i.e., N is real) would permit a parcel of fluid slightly displaced parallel (or anti-parallel) to $\vec{g}_0$ and compressed (or expanded) by the change in pressure, to be lighter (or denser) than the fluid surrounding it, and therefore to perform small oscillations about its original position with frequency N. On the other hand, a negative value of N^2 (i.e., N is imaginary) would result in the displaced parcel as being denser (or lighter) than its new surrounding, and leads to sinking (or rising) and a parcel of fluid exhibits an exponential decay or growth. Thus one says that if $N^2 > 0$ ($\beta < 0$) then the density profile

is stably stratified, if $N^2 < 0$ ($\beta > 0$) then the density profile is unstably stratified and if $N^2 = 0$ ($\beta = 0$) then the density profile is neutrally stratified (i.e., satisfying the Adams-Williamson equation). The value of β is estimated, from seismic data analysis [37], to be in the range $|\beta| < 0.03 - 0.05$.

The linearized equations describing the dynamics of the fluid core are those of the conservation of mass, momentum, gravitational flux and entropy [27, 36]. These equations are:

$$\frac{\partial \rho_1}{\partial t} = -\nabla \cdot (\rho_0 \vec{v}), \tag{2.7}$$

$$\frac{\partial \vec{v}}{\partial t} + 2\Omega \hat{e}_3 \times \vec{v} = -\frac{1}{\rho_0}\nabla p_1 + \nabla V_1 + \frac{\rho_1}{\rho_0}\vec{g}_0, \tag{2.8}$$

$$\nabla^2 V_1 = -4\pi G \rho_1, \tag{2.9}$$

$$\frac{\partial p_1}{\partial t} = \alpha^2 \frac{\partial \rho_1}{\partial t} - \beta \rho_0 \vec{v} \cdot \vec{g}_0, \tag{2.10}$$

where $\vec{v} = \frac{\partial \vec{u}}{\partial t}$ is the velocity; ρ_1, p_1 and V_1 are the Eulerian perturbation in density, pressure and gravitational potential, respectively. Here $\vec{u}$ is the Lagrangian displacement from equilibrium. The Eqs. (2.7), (2.8), (2.8) and (2.9) are also known as the continuity equation, the Navier- Stokes (i.e., momentum) equation, Poisson's equation and the equation of state, respectively.

Since we are dealing with small oscillations, we assume that all the field variables have $e^{i\omega t}$ dependence, where ω is the modal frequency. Under this assumption and doing some mathematical operations in Eqs. (2.7) - (2.10), the fundamental linearized dynamical equations may be written as

$$\rho_1 = -\nabla \cdot (\rho_0 \vec{u}), \tag{2.11}$$

$$-\omega^2 \vec{u} + 2i\omega\Omega \hat{e}_3 \times \vec{u} = -\frac{1}{\rho_0}\nabla p_1 + \nabla V_1 + \frac{\rho_1}{\rho_0}\vec{g}_0, \tag{2.12}$$

$$\nabla^2 V_1 = -4\pi G \rho_1, \tag{2.13}$$

$$p_1 = \alpha^2 \rho_1 - \beta \rho_0 \vec{u} \cdot \vec{g}_0. \tag{2.14}$$

Using Eqs. (2.5) and (2.11) in Eqs.(2.12)-(2.14), then we get

$$-\omega^2\vec{u}+2i\omega\Omega\hat{e}_3\times\vec{u} = -\frac{1}{\rho_0}\nabla p_1+\nabla V_1-\frac{\vec{g}_0}{\rho_0}\nabla\cdot(\rho_0\vec{u}), \quad (2.15)$$

$$\nabla^2 V_1 = 4\pi G\nabla\cdot(\rho_0\vec{u}), \quad (2.16)$$

$$\frac{p_1}{\rho_0} = -(\alpha^2\nabla\cdot\vec{u}+\vec{u}\cdot\vec{g}_0). \quad (2.17)$$

The three components of the vector Eq. (2.15), Eqs. (2.16) and (2.17) are five linear partial differential equations (PDEs) in five variables: p_1, V_1 and the three components of $\vec{u}$. These equations represent the dynamics of the Earth's fluid core.

2.2 Boundary Conditions

These PDEs [Eq. (2.15), Eqs. (2.16) and (2.17)] are subject to certain continuity conditions at material interfaces, i.e., surfaces where one or more material properties are discontinuous. The boundary conditions of these equations are:

(i) continuity of the normal component of the displacement, $\hat{n}\cdot\vec{u}$,

(ii) continuity of the perturbation in the gravitational potential, V_1,

(iii) continuity of the normal component of the gravitational flux, $\hat{n}\cdot(\nabla V_1-4\pi G\rho_0\vec{u})$,

(iv) continuity of the normal component of the stress tensor, $\hat{n}\cdot\tilde{\tau}$,

where $\tilde{\tau}$ is the additional stress tensor due to deformation/flow, superimposed on the hydrostatic pressure stress of the equilibrium configuration, and $\hat{n}$ is the unit vector normal to the undeformed boundary surface. In the outer core the stress tensor has the form

$$\tilde{\tau} = -(p_1+\vec{u}\cdot\nabla p_0)\tilde{\mathbf{1}}, \quad (2.18)$$

and in the solid mantle it can be written as

$$\tilde{\tau} = (\lambda \nabla \cdot \vec{u})\tilde{\mathbf{1}} + 2\mu[\nabla\vec{u} + (\nabla\vec{u})^T], \tag{2.19}$$

where μ and λ are the Lamé parameters and $\tilde{\mathbf{1}}$ is the unit dyadic, $\nabla\vec{u} + (\nabla\vec{u})^T$ called the strain dyadic or strain tensor is a symmetric tensor, and the ij'th component of $(\nabla\vec{u})^T$ is the ji'th component of $\nabla\vec{u}$.

In the traditional approach [27, 28, 29, 30], to solve these dynamical equations [i.e., Eqs. (2.15), (2.16) and (2.17)] plus the boundary conditions, the field variables are represented in spherical polar coordinates system by spherical harmonics

$$\vec{u} = \sum_{m=-\infty}^{\infty} \sum_{n=|m|}^{\infty} \vec{S}_n^m + \vec{T}_n^m, \tag{2.20}$$

$$V_1 = \sum_{m=-\infty}^{\infty} \sum_{n=|m|}^{\infty} \phi_n^m Y_n^m, \tag{2.21}$$

$$p_1 = \sum_{m=-\infty}^{\infty} \sum_{n=|m|}^{\infty} \psi_n^m Y_n^m, \tag{2.22}$$

with being the spheroidal $\vec{S}_n^m$ and toroidal $\vec{T}_n^m$ components of $\vec{u}$:

$$\vec{S}_n^m = [u_n^m \hat{r} + r v_n^m \nabla] Y_n^m, \tag{2.23}$$

$$\vec{T}_n^m = -t_n^m \hat{r} \times \nabla Y_n^m, \tag{2.24}$$

$$Y_n^m = P_n^m(\cos\theta) e^{im\phi}, \tag{2.25}$$

where u_n^m, v_n^m, t_n^m, ϕ_n^m and ψ_n^m are functions of r only, and $P_n^m(\cos\theta)$ is the associated Legendre function of degree n and azimuthal order m.

The traditional approach is an effective tool to solve the governing equations for the computation of short period, shorter than a few hours, free oscillations. The short period free oscillations (acoustic modes) were computed by Alterman et al. [38]. They showed

that the effect of rotation and ellipticity is negligible for these types of oscillations. For the long period normal modes (longer than half a day), this approach leads to one or the other of two coupled chains of infinite length [27]. The calculation period of the oscillations of the fluid body then depends on the heavy truncation of these coupled chains. This approach gives reasonable results for the period of Chandler wobble. However, the approach is not adequate to compute to periods of the inertial modes and Slichter modes which are more sensitive to Earth's rotation [39]. Therefore, new alternative approaches were considered for the solution of the governing equations of the fluid core oscillations [40, 34, 35].

Smylie and Rochester [34] were able to reduce the governing equations of the fluid core in terms of only two scalar variables from the perturbation in the pressure and the gravitational potential based on the subseismic approximation (SSA). In this method, the contribution of the pressure perturbation $\frac{p_1}{\rho_0} << |\vec{g}_0 \cdot \vec{u}|$ is ignored in the equation (2.17). Rochester and Peng [41] implemented the SSA to show frequency dependence of the load Love numbers at the Inner Core Boundary (ICB) and Core Mantle Boundary (CMB). They applied a variational principle to solve for the frequencies of the Slichter modes for a rotating, spherical and neutrally stratified Earth model. Peng [42] noticed that the SSA violates the conservation of the linear momentum near the CMB.

Wu and Rochester [40] introduced the two potential description (TPD) that consists of two scalar governing equations in two scalar potentials, one the same that for the SSA and other the Eulerian perturbation in the gravitational potential. One of the shortcomings of the TPD is that one of the equations depend on a term proportional to $1/\beta$ (see [40]) which leads to instability at $\beta = 0$.

2.3 The Three Potential Description (3PD)

In order to overcome the above limitations, a new approach, the three potential description(3PD), was proposed by Seyed-Mahmoud and Rochester [32] (see a detailed derivation of the 3PD in [32]). The three scalars constituting the three potentials are constructed from

the perturbation in pressure $\chi = \frac{p_1}{\rho_0}$, the dilatation $\zeta = \nabla \cdot \vec{u}$ and the perturbation in gravitational potential V_1. The dynamical equations [i.e., Eqs. (2.15), (2.16) and (2.17)] are written in terms of the three potentials as:

$$\nabla^2 V_1 - 4\pi G\rho_0\{\beta\zeta - \tfrac{1-\beta}{\alpha^2}\chi\} = 0, \tag{2.26}$$

$$\nabla \cdot \{\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla(\chi - V_1) - \beta\vec{C}^*\zeta\} - \omega^2(\omega^2 - 4\Omega^2)\zeta = 0, \tag{2.27}$$

$$\vec{C} \cdot \nabla(\chi - V_1) - \omega^2(\omega^2 - 4\Omega^2)\chi - B\zeta = 0, \tag{2.28}$$

where

$$\tilde{\Gamma}_{\mathbf{p}} = \omega^2\tilde{\mathbf{1}} - 4\Omega^2\hat{e}_3\hat{e}_3 + 2i\omega\Omega\hat{e}_3 \times \tilde{\mathbf{1}}, \tag{2.29}$$

$$B = \alpha^2\omega^2(\omega^2 - 4\Omega^2) + \beta[\omega^2 g_0^2 - 4\Omega^2(\hat{e}_3 \cdot \vec{g}_0)^2], \tag{2.30}$$

$$\vec{C} = -\omega^2\vec{g}_0 + 4\Omega^2\hat{e}_3 \cdot \vec{g}_0\hat{e}_3 + 2i\omega\Omega^2\hat{e}_3 \times \vec{g}_0, \tag{2.31}$$

with $\vec{C}^*$ is the complex conjugate of $\vec{C}$, and the displacement vector, $\vec{u}$ is given by

$$\omega^2(\omega^2 - 4\Omega^2)\vec{u} = \tilde{\Gamma}_{\mathbf{p}} \cdot \nabla\,(\chi - V_1) - \beta\vec{C}^*\zeta. \tag{2.32}$$

Thus the Eqs. (2.26) and (2.27) are second order differential equations, and (2.28) is first order. These equations are known as the Three Potential Description (3PD) of core dynamics. They describe the exact linearized dynamics of rotating, self-gravitating, stratified, compressible and inviscid fluids. The Eqs. (2.26), (2.27) and (2.28) are also called the Poisson's, momentum and entropy equation [35], respectively.

For computational purposes, we render the 3PD non-dimensional as follows:

$\chi' = \frac{\chi}{4\Omega^2R^2}$, $V_1' = \frac{V_1}{4\Omega^2R^2}$, $\alpha' = \frac{\alpha}{2\Omega R}$, $\tilde{\Gamma'}_{\mathbf{p}} = \frac{\tilde{\Gamma}_{\mathbf{p}}}{4\Omega^2}$, $\sigma = \frac{\omega}{2\Omega}$, $\vec{g_o}' = \frac{\vec{g}_0}{4\Omega^2R}$,

$G' = \frac{G\rho_0}{4\Omega^2}$, $\nabla'() = \frac{1}{2R}\nabla()$, and $\nabla'^2() = \frac{1}{4R^2}\nabla^2()$

in Eqs. (2.26), (2.27) and (2.28), where R is the radius of Earth, σ is the non-dimensional modal frequency and the notation $()'$ is represented the dimensionless terms. The non-dimensional form of the 3PD is written as:

$$\nabla^2 V_1 - 4\pi G\{\beta\zeta - \tfrac{1-\beta}{\alpha^2}\chi\} = 0, \tag{2.33}$$

$$\nabla\cdot\{\tilde{\Gamma}_{\mathbf{p}}\cdot\nabla(\chi - V_1) - \beta\vec{C}^*\zeta\} - \sigma^2(\sigma^2-1)\zeta = 0, \tag{2.34}$$

$$\vec{C}\cdot\nabla(\chi - V_1) - \sigma^2(\sigma^2-1)\chi - B\zeta = 0, \tag{2.35}$$

where

$$\tilde{\Gamma}_{\mathbf{p}} = \sigma^2\tilde{\mathbf{1}} - \hat{e}_3\hat{e}_3 + i\sigma\hat{e}_3\times\tilde{\mathbf{1}}, \tag{2.36}$$

$$B = \alpha^2\sigma^2(\sigma^2-1) + \beta[\sigma^2 g_0^2 - (\hat{e}_3\cdot\vec{g}_0)^2], \tag{2.37}$$

$$\vec{C} = -\sigma^2\vec{g}_0 + \hat{e}_3\cdot\vec{g}_3\hat{e}_3 + i\sigma\hat{e}_3\times\vec{g}_0, \tag{2.38}$$

and the displacement vector, $\vec{u}$ is given by

$$\sigma^2(\sigma^2-1)\vec{u} = \tilde{\Gamma}_{\mathbf{p}}\cdot\nabla(\chi - V_1) - \beta\vec{C}^*\zeta, \tag{2.39}$$

where for convenience we have dropped the notation $()'$ from all terms. We, therefore, solve for the non-dimensional frequencies and the displacement eigenfunctions of the inertial modes using the Eqs. (2.33), (2.34), (2.35) and (2.39). The advantages of the 3PD [32] are that:

(i) there is no need to present the spheroidal and toroidal components of vector fields because all dependent field variables are scalar,

(ii) there are three equations involved rather than five as in the traditional approach,

(iii) the dynamical equations are stable for any range of frequency,

(iv) this approach leads to one or the other of two decoupled chains of infinite length,

(v) the description involves no derivatives higher than second and when a Galerkin method (or the variational principle) is used to solve the governing equations, application of the divergence theorem replaces the volume integrals involving second derivatives with surface and volume integrals involving first derivatives in the form for which the boundary conditions are natural.

2.3.1 The Poincaré Equation

If the fluid is homogeneous and incompressible (i.e., $\zeta = 0$), the boundary is rigid and no perturbation of density takes outside the container (i.e., $V_1 = 0$), the Eqs. (2.34) and (2.39) become

$$\nabla \cdot (\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla \chi) = 0, \tag{2.40}$$

and

$$\sigma^2(\sigma^2 - 1)\vec{u} = \tilde{\Gamma}_{\mathbf{p}} \cdot \nabla \chi, \tag{2.41}$$

which are the Poincaré equations for the pressure and the displacement. Analytical solutions exist for these equations of a sphere [23]. However, we solve them numerically as a test of our approach.

2.4 The Web of Characteristics

A hyperbolic partial differential equation can be decomposed into ordinary differential equations along curves known as characteristics. The web of characteristics is the signature of the hyperbolic nature of the governing equation. The governing equation is of hyperbolic type so that some discontinuities (gradient discontinuities) occur across the characteristic lines. If a characteristic line is revolved around the rotational axis, a surface called a characteristic cone is swept out (see Figure 2.2). These characteristic lines act as rays and reflect off the boundary and travel through the fluid. For this reason they are sometime called

attractors. If the slope of a characteristic line is associated with the frequency of an inertial mode then the attractor forms a closed loop, i.e., the ray returns to a reflected position on the boundary and repeats the loop (see figure 2.2 below).

For a derivation of the web of characteristics, the dynamical equations (2.15) and (2.17) are written in terms of the three scalar fields [32]

$$-\omega^2\vec{u}+2i\omega\Omega\hat{e}_3\times\vec{u} = -\nabla\Phi-\beta\vec{g}_0\zeta, \tag{2.42}$$

$$\vec{u}\cdot\vec{g}_0 = -(\alpha^2\zeta+\chi) \tag{2.43}$$

with $\Phi=\chi-V_1$.

Using Eqs. (2.6) and (2.43) in Eq. (2.42), then we get

$$-\omega^2\vec{u}+2i\omega\Omega\hat{e}_3\times\vec{u}=-\nabla\Phi-N^2U_r\hat{r} \tag{2.44}$$

with

$$U_r = \frac{4\Omega^2(\vec{u}\cdot\vec{g}_0+\chi)}{g_0}, \tag{2.45}$$

$$\vec{g}_0 = g_0\hat{r}. \tag{2.46}$$

First operating with $(\hat{e}_3\cdot)$ on Eq. (2.44),

$$\hat{e_3}\cdot\vec{u}=\frac{1}{\omega^2}[\hat{e}_3\cdot\nabla\Phi+N^2U_r\hat{e}_3\cdot\hat{r}]. \tag{2.47}$$

Now operating with $(\hat{e}_3\times)$ on Eq.(2.44) to get

$$-\omega^2\hat{e}_3\times\vec{u}+2i\omega\Omega\hat{e}_3\times\hat{e}_3\times\vec{u}=-\hat{e}_3\times\nabla\Phi-N^2U_r\hat{e}_3\times\hat{r},$$

$$\therefore -\omega^2\hat{e}_3\times\vec{u}+2i\omega\Omega[\hat{e}_3(\hat{e_3}\cdot\vec{u})-\vec{u}]=-\hat{e}_3\times\nabla\Phi-N^2U_r\hat{e}_3\times\hat{r}. \tag{2.48}$$

Using Eqs.(2.44) and (2.47) in Eq. (2.48), then we get

$$\omega^2\nabla\Phi - 4\Omega^2\hat{e}_3(\hat{e}_3\cdot\nabla)\Phi + N^2U_r[\omega^2\hat{r} - 4\Omega^2\hat{e}_3(\hat{e}_3\cdot\hat{r})] + \omega^2(4\Omega^2-\omega^2)\vec{u}$$
$$= -2i\omega\Omega[\hat{e}_3\times\nabla\Phi + N^2U_r\hat{e}_3\times\hat{r}]. \tag{2.49}$$

Now operating with $(\nabla\cdot)$ on Eq. (2.49), and using Eqs. (2.43) and (2.45) to get,

$$\nabla\cdot[\omega^2\nabla\Phi - 4\Omega^2\hat{e_3}(\hat{e_3}\cdot\nabla)\Phi + N^2U_r\{\omega^2\hat{r} - 4\Omega^2\hat{e}_3(\hat{e}_3\cdot\hat{r}) + 2i\omega\Omega\hat{e}_3\times\hat{r}\}]$$
$$+(4\Omega^2-\omega^2)\left(\frac{\omega^2}{4\Omega^2}\right)\left(\frac{-g_0}{\alpha^2}\right)U_r = 0, \tag{2.50}$$

where using $\nabla\cdot\vec{u} = \zeta$ as in previous section 2.3.

Again, operate $(\hat{r}\cdot)$ on Eq. (2.49) and use Eq. (2.45) to get

$$\omega^2\hat{r}\cdot\nabla\Phi - 4\Omega^2(\hat{r}\cdot\hat{e}_3)(\hat{e}_3\cdot\nabla)\Phi + N^2U_r[\omega^2 - 4\Omega^2(\hat{e}_3\cdot\hat{r})^2] + \omega^2(4\Omega^2-\omega^2)\hat{r}\cdot\vec{u}$$
$$= -2i\omega\Omega[\hat{r}\cdot\hat{e}_3\times\nabla\Phi + N^2U_r\hat{r}\cdot\hat{e}_3\times\hat{r}],$$
$$\therefore \omega^2\hat{r}\cdot\nabla\Phi - 4\Omega^2(\hat{r}\cdot\hat{e_3})(\hat{e_3}\cdot\nabla)\Phi + N^2U_r[\omega^2 - 4\Omega^2(\hat{e}_3\cdot\hat{r})^2] + \omega^2(4\Omega^2-\omega^2)$$
$$\times\left(U_r - \frac{\chi}{g_0}\right) = -2i\omega\Omega[\hat{r}\cdot\hat{e_3}\times\nabla\Phi + N^2U_r\hat{r}\cdot\hat{e_3}\times\hat{r}]$$

which gives

$$U_r = \frac{[\omega^2\hat{r} - 4\Omega^2\hat{e}_3(\hat{e}_3\cdot\hat{r}) - 2i\omega\Omega\hat{e}_3\times\hat{r}]\cdot\nabla\Phi - (4\Omega^2-\omega^2)\left(\frac{\omega^2}{4\Omega^2}\right)\frac{\chi}{g_0}}{D}, \tag{2.51}$$

where

$$D \equiv N^2[4\Omega^2(\hat{e_3}\cdot\hat{r})^2 - \omega^2] - (4\Omega^2-\omega^2)\left(\frac{\omega^2}{4\Omega^2}\right). \tag{2.52}$$

The parameter, $\frac{g_0}{\alpha^2}$ is a measure of the compressibility of the fluid core. Clearly substitution of Eq. (2.51) with Eq. (2.52) in Eq. (2.50) gives a second order partial differential equation.

The leading order terms are independent of $\frac{g_0}{\alpha^2}$. The type of differential equation only depends on their leading order derivatives (see Appendix A.1). Using Eqs. (2.51) and (2.52) in Eq. (2.50), we rewrite the Eq.(2.50) in the cylindrical coordinates, (s,ϕ,z), by keeping second order derivative terms only:

$$\begin{aligned}(D\omega^2+N^2\omega^4\sin^2\theta)\frac{\partial^2\Phi}{\partial s^2}+\left[D(\omega^2-4\Omega^2)-N^2(\omega^2-4\Omega^2)^2\cos^2\theta\right]\frac{\partial^2\Phi}{\partial z^2}\\ +\left[\frac{D\omega^2}{s^2}+\frac{4\omega^2\Omega^2N^4\sin\theta}{s^2}\right]\frac{\partial^2\Phi}{\partial\phi^2}+\frac{2i\omega^3\Omega N^2\sin^2\theta}{s}\frac{\partial^2\Phi}{\partial\phi\partial s}-\frac{2i\omega\Omega N^2\cos\theta}{s}(\omega^2-4\Omega^2)\\ \times\frac{\partial^2\Phi}{\partial\phi\partial z}-\frac{2i\omega\Omega N^2\sin\theta\cos\theta}{s}(\omega^2-4\Omega^2)\frac{\partial^2\Phi}{\partial z\partial\phi}+H\left(\frac{\partial\Phi}{\partial s},\frac{\partial\Phi}{\partial z},\frac{\partial\Phi}{\partial\phi},\Phi\right)=0,\end{aligned}\tag{2.53}$$

where

$$D=N^2(4\Omega^2\cos^2\theta-\omega^2)-\frac{\omega^2}{4\Omega^2}(4\Omega^2-\omega^2).\tag{2.54}$$

Let $\chi=a_1F(s,z)e^{im\phi}$, $V_1=a_2F(s,z)e^{im\phi}$ and $\Phi=\chi-V_1$, where a_1 and a_2 are constants, i.e., $\Phi\propto F(s,z)e^{im\phi}$, then Eq. (2.53) becomes

$$\begin{aligned}(D\omega^2+N^2\omega^4\sin^2\theta)\frac{\partial^2F}{\partial s^2}+\left[D(\omega^2-4\Omega^2)-N^2(\omega^2-4\Omega^2)^2\cos^2\theta\right]\frac{\partial^2F}{\partial z^2}\\ +G\left(\frac{\partial F}{\partial s},\frac{\partial F}{\partial z},F\right)=0.\end{aligned}\tag{2.55}$$

Thus Eq. (2.55) is a second order linear partial differential equation in the two independent variables s,z. The *characteristics* of Eq. (2.55) depend only on the coefficients of second order derivatives (see Appendix A.1). Now comparing Eq.(2.55) with Eq. (A.1), we get

$$a = D\omega^2+N^2\omega^4\sin^2\theta,\tag{2.56}$$

$$b = 0,\tag{2.57}$$

$$c = D(\omega^2-4\Omega^2)-N^2(\omega^2-4\Omega^2)^2\cos^2\theta.\tag{2.58}$$

The characteristics of Eq. (2.55) are

$$\begin{aligned}\frac{dz}{ds} &= \pm\sqrt{\frac{N^2(\omega^2-4\Omega^2)^2\cos^2\theta - D(\omega^2-4\Omega^2)}{D\omega^2+N^2\omega^4\sin^2\theta}},\\ &= \pm\sqrt{\frac{N^2(\omega^2-4\Omega^2)^2 z^2 - D(\omega^2-4\Omega^2)r^2}{D\omega^2 r^2+N^2\omega^4 s^2}},\end{aligned} \tag{2.59}$$

where, $D = N^2\left(4\Omega^2\frac{z^2}{r^2}-\omega^2\right)-(4\Omega^2-\omega^2)\left(\frac{\omega^2}{4\Omega^2}\right)$. We obtain the web of characteristics, which is functions of the frequency and stratification, after integrating Eq. (2.59). In this thesis, we wish to study only pure inertial modes of the core so that the values of β are very small. In this case $N^2(r)\sim 0$, for example if $\beta=-0.005$, $N^2(r)\sim 10^{-15}$. Figure 2.1 shows the numerical solutions of Eq. (2.59) for the (4,1,0) mode for different stability parameter, β. The characteristics lines for small values of β are straight lines. The characteristic line for $\beta=-0.005$ is shifted counterclockwise from that for a neutrally stratified, while the characteristic line for $\beta=0.005$ is shifted clockwise. We see that Eq. (2.59) is independent of m. The characteristics of Eq. (2.55) are, therefore, independent of *azimuthal symmetry*. Because of the symmetry of the problem, the azimuthal variable ϕ is separable from the two others (s,z).

For a neutrally stratified core model, $N^2=0$, Eq. (2.59) becomes

$$\frac{dz}{ds} = \pm\sqrt{\frac{4\Omega^2}{\omega^2}-1} \tag{2.60}$$

which gives

$$\begin{aligned}c_\pm &= z \pm s\sqrt{\frac{4\Omega^2}{\omega^2}-1},\\ &= z \pm s\sqrt{\frac{1}{\sigma^2}-1},\end{aligned} \tag{2.61}$$

where $c_\pm$ will designate the characteristic coordinates; c_+ and c_- are constant along characteristics of positive and negative slope respectively, and σ is the non-dimensional frequency

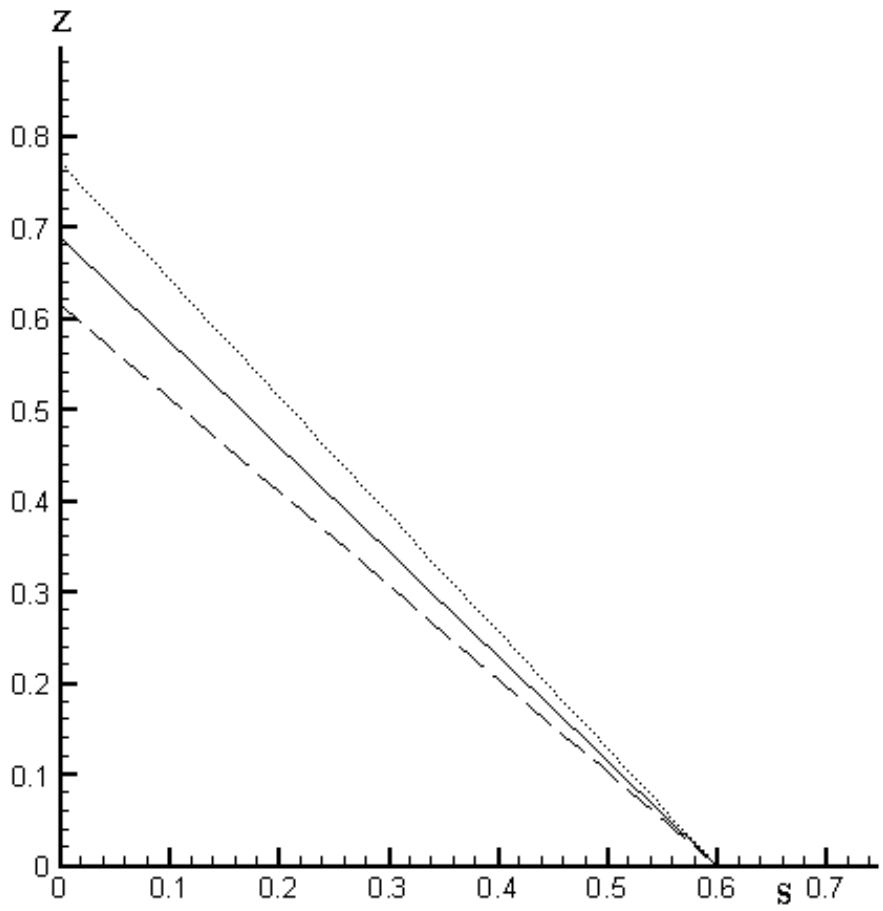

Figure 2.1: The numerical solutions of equation (2.59) for the (4,1,0) mode for different stability parameter, β, solid line: $\beta = 0.00$ with $\sigma_{0.00} = 0.657$, dashed line: $\beta = -0.005$ with $\sigma_{-0.005} = 0.698$ and dotted line: $\beta = 0.005$ with $\sigma_{0.005} = 0.614$. Note that the non-dimensional frequencies, σ, are taken here from the tables 4.4-4.6

as in the previous section 2.3. The equation (2.61) $c_+ = c_1$, where c_1 is an arbitrary constant, defines a characteristic line with slope, $+\sqrt{\frac{1}{\sigma^2} - 1}$, in the sz- plane. Eq. (2.61) corresponds to the two families of characteristics which are straight lines with slopes $\pm\sqrt{\frac{1}{\sigma^2} - 1}$. Moreover, we can get the similar characteristic equations for the Poincaré equation (i.e., (2.40)). We see from Eq. (2.61) that the slopes of the characteristics depend on the frequency of modes. Figure 2.2 shows the compressed potential, $\Phi = \chi - V_1$, contours with the characteristic lines of the (6,4,1) mode. It is clear that two families of the characteristics, the dashed lines and the solid lines, form closed loops. Both families also pass through the discontinuities in the pressure gradients, which is one of the properties of the characteristics

of hyperbolic equations.

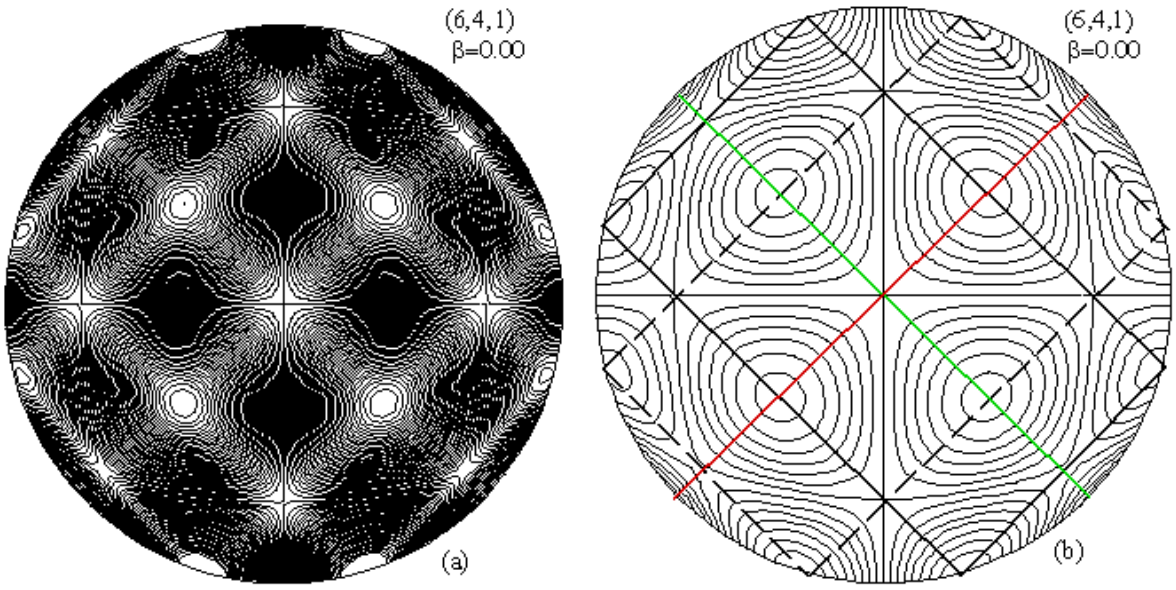

Figure 2.2: The compressed potential, $\Phi = \chi - V_1$, contours in a meridional plane, $\phi = 0$, of the (6,4,1) ($\sigma_{0.00} = 0.652$) mode, (a) without the characteristic lines, and (b) with the web characteristic lines. Note that the horizontal axis is the equator and the vertical axis is the rotational axis.

To find the relation between the meridional velocity and the characteristics, the momentum equation [i.e., Eq. (2.44)] in the cylindrical coordinates system (s, ϕ, z) is expressed in terms of components for axisymmetric modes [i.e., $\partial_\phi() = 0$] as

$$-\omega^2 u_s - 2i\omega\Omega u_\phi = -\frac{\partial\Phi}{\partial s} - N^2 U_r \sin\theta, \tag{2.62}$$

$$-\omega^2 u_\phi + 2i\omega\Omega u_s = 0, \tag{2.63}$$

$$-\omega^2 u_z = -\frac{\partial\Phi}{\partial z} - N^2 U_r \cos\theta. \tag{2.64}$$

Substituting Eq. (2.63) in Eq. (2.62) to obtain from Eqs. (2.62) and (2.64)

$$u_s = \frac{1}{\omega^2 - 4\Omega^2}\left[\frac{\partial\Phi}{\partial s} + N^2 U_r \sin\theta\right], \tag{2.65}$$

$$u_z = \frac{1}{\omega^2}\left[\frac{\partial\Phi}{\partial z} + N^2 U_r \cos\theta\right]. \tag{2.66}$$

We find that

$$\vec{u}\cdot\nabla\Phi = \frac{1}{\omega^2-4\Omega^2}\left[\frac{\partial\Phi}{\partial s}+N^2U_r\sin\theta\right]\frac{\partial\Phi}{\partial s}+\frac{1}{\omega^2}\left[\frac{\partial\Phi}{\partial z}+N^2U_r\cos\theta\right]\frac{\partial\Phi}{\partial z}. \tag{2.67}$$

The characteristic surfaces are also isobars (lines connecting points of equal pressure), i.e., the right hand side of Eq. (2.67) is always zero along these lines. The meridional displacement of axisymmetric modes is therefore parallel to the characteristic lines. Høiland [43] and Henderson [26] reported that a *special characteristic line* locates inside the cells where the velocity (i.e., displacement) is zero at some points. Thus, a special characteristic line indicates the location of the cell. Dintrans et al. [44] pointed out that the characteristics of non-axisymmetric modes exist in the meridional displacement eigenfunctions but are modulated by $e^{im\phi}$.

2.5 Core Models

In this section, we follow Seyed-Mahmoud [35] and take the PREM (Preliminary Reference Earth Model) [2] as the base for our core models to investigate the inertial modes of both a fluid sphere and a spherical shell. Initially, we assume that the mantle is rigid and the presence of the solid inner core is ignored for simplicity. For PREM, the density profile (see Figure 2.3) of the fluid core is a third order polynomial and the stability parameter β has numerical values of approximately -0.028 at the ICB (Inner Core Boundary) and 0.010 at CMB (Core Mantle Boundary); see Figure 2.4. The density profile of the fluid core of PREM is not valid at the center of a spherical model because the derivative of the density profile does not ensure the gravitational acceleration at the center. The density profile has to be regular at the center, which means its first derivative must vanish. This is the same for the P wave speed. We make sure that the density profile ρ and the P-wave speed α in the fluid core are expanded to the center of the Earth as smooth functions of the radius and that the mass of the core is conserved.

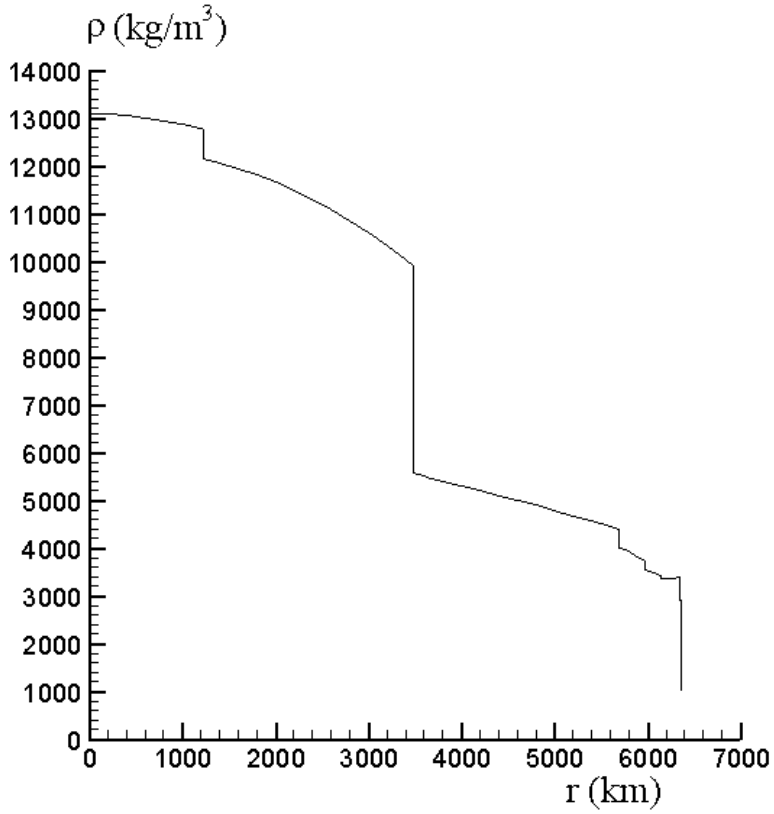

Figure 2.3: The density profile of PREM.

2.5.1 Ellipticity

In section 2.1, we have mentioned that the reference state is one of hydrostatic equilibrium in a frame rotating steadily at the rate $\Omega\hat{e_3}$. In this state the figure of a rotating body is a spheroid and the radius of an equipotential surface [45] is given as

$$r = r_0[1 - \frac{2}{3}\varepsilon P_2(\cos\theta)], \tag{2.68}$$

where r_0 is the mean radius of the spherical equipotential surface, P_2 is the degree two Legendre polynomial, θ is the co-latitude and ε is the ellipticity which has numerical values ranging from about 0.00247 at the ICB to about 0.00255 at the CMB. The effects of ellipticity on the frequencies of the core's inertial modes are small, of the order of ellipticity [46]. For example, the non-dimensional frequency, $\sigma = \frac{\omega}{2\Omega}$, of the (2,1,1) mode of the

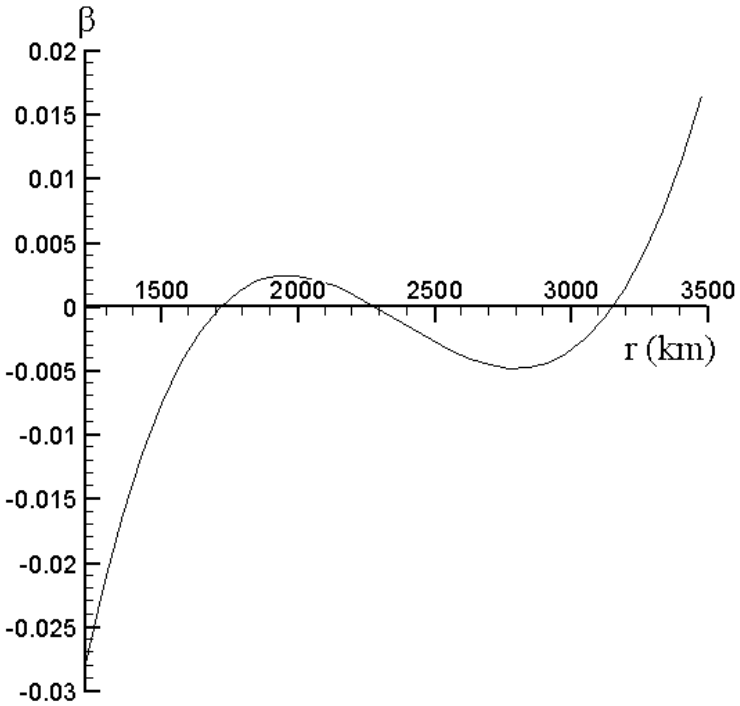

Figure 2.4: The stability parameter profile of PREM. Note that the horizontal axis starts at $r = 1221.5$km.

Poincaré model has numerical values of 0.5000 and 0.5013 for a spherical and a spheroidal geometries, respectively. Since in this thesis we study the effects of non-zero β on the frequencies of the inertial modes of the core, we consider a first order approximation in terms of the ellipticity, i.e., $\varepsilon = 0$.

2.5.2 The Modified P-Wave Profile

The P- wave speed, α, of the fluid core of PREM is given as

$$\alpha = (c_0 + c_1 x + c_2 x^2 + c_3 x^3)\text{km s}^{-1}, \tag{2.69}$$

where $c_0 = 11.0487$, $c_1 = -4.0362$, $c_2 = 4.8023$, $c_3 = -13.5732$, $x = \frac{r}{R}$ and R is the mean radius of the surface of the Earth. Since the first derivative of Eq. (2.69) with respect to r

is not zero at $r = 0$, this profile can not be used when the inner core is neglected as $\frac{d\alpha}{dr}$ is not zero.

The P-wave speed, α, of the PREM in the solid inner core is a smooth function of the radius and has the form

$$\alpha = (11.2622 - 6.3640x^2)\text{km s}^{-1}. \tag{2.70}$$

If we use this profile as a modified core model, we get a maximum error of about 17% at the CMB because the P-wave travels faster in solid media than in fluids. Compromising to reduce this discrepancy, we modify α to the form

$$\alpha = (10.6776 - 8.7572x^2)\text{km s}^{-1}. \tag{2.71}$$

The first derivative of this modified P-wave profile vanishes at $r = 0$ and its numerical values are close at ICB and CMB, and involves a maximum error of about 0.7% at $r \simeq 2900$ km. Figure 2.5 shows α distributions in the fluid core of PREM ($r = 1221.5$ km to $r = 3480$ km) and the modified PREM.

2.5.3 The Modified Density Profile

As mentioned in the Eq. (2.5), the density gradient in the fluid core is given as

$$\nabla\rho_0 = (1-\beta)\rho_0\frac{\vec{g}_0}{\alpha^2}, \tag{2.72}$$

which can be written as

$$\frac{d\rho_0}{dr} = -(1-\beta)\rho_0\frac{g_0}{\alpha^2}, \tag{2.73}$$

for a spherical Earth, since ρ_0 is a function of r only.

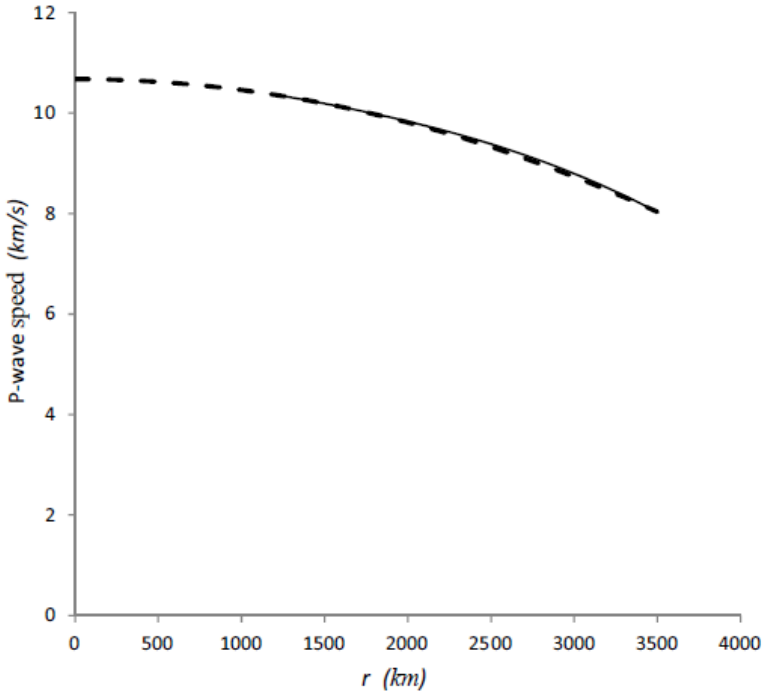

Figure 2.5: The P-wave speeds in the fluid core of PREM (solid line) and the fluid core of the modified PREM with no inner core (dashed line).

We choose the density profile as

$$\rho_0 = \sum_{j=1}^{N'} d_j x^{j-1}, \tag{2.74}$$

where $x = \frac{r}{R}$ is defined as in previous subsection 2.5.2, N' is an integer and d_j are constants. However, this profile [i.e., (2.74)] would satisfy that its first derivative vanishes at $r = 0$, if $d_2 = 0$. From Eq. (2.74) with this condition, we get

$$\frac{d\rho_0}{dr} = \frac{d\rho_0}{dx}\frac{dx}{dr} = \frac{1}{R}\sum_{j=3}^{N'}(j-1)d_j x^{j-2}. \tag{2.75}$$

To solve Eq. (2.73) for a best fitting density profile ρ_0, we use a Galerkin method with weight functions x^{i-1} $(i = 1, \cdots, N'-2)$. Using Eq. (2.75), the Galerkin representation of

Eq. (2.73) is

$$\sum_{j=3}^{N'}(j-1)d_j\int_0^{\frac{b}{R}}\frac{\alpha^2}{\rho_0 g_0 R}x^{i+j-3}dx+(1-\beta)\int_0^{\frac{b}{R}}x^{i-1}dx=0, \tag{2.76}$$

for each i, and b is the mean radius of CMB. Eq. (2.76) can be written as

$$\sum_{j=3}^{N'}A_{ij}d_j=F_i, \tag{2.77}$$

where

$$A_{ij}\equiv(j-1)\int_0^{\frac{b}{R}}\frac{\alpha^2}{\rho_0 g_0 R}x^{i+j-3}dx, \tag{2.78}$$

and

$$F_i\equiv-\frac{(1-\beta)}{i}\left(\frac{b}{R}\right)^i. \tag{2.79}$$

The gravitational acceleration, g_0, in the fluid core is given as

$$g_0(r)=\frac{GM(r)}{r^2}, \tag{2.80}$$

where $M(r)$ is the total mass of the body enclosed by the shell of radius r, which is given as

$$M(r)=4\pi\int_0^r\rho_0 r^2dr. \tag{2.81}$$

Using Eqs. (2.74) and (2.81) in Eq. (2.80), we get

$$g_0(r)=4\pi GR\sum_{j=1}^{N'}\frac{d_j x^j}{j+2}. \tag{2.82}$$

The exact value of β is not known in the core, but a possible range of β is given, $|\beta|<0.03-0.05$, by Masters [37]. In this study, we have chosen the values of β (e.g. $\beta=-0.001,-0.002$) much smaller than 0.03 because we wish to study only pure inertial modes of the Earth fluid core.

Now β is set to a desired value, β_d, (e.g., $-0.001, -0.002$, etc) in Eq. (2.79), and the density profile of PREM is used as a starting value to find ρ_0 [by Eq. (2.74)] and g_0 [by Eq. (2.82)] in Eq.(2.78). We set the starting values of the coefficients of the density profile: $d_1 = 12.5815 \times 10^3$ kg m^{-3}, $d_2 = 0$ kg m^{-3}, $d_3 = -3.6426 \times 10^3$kg m^{-3}, $d_4 = -5.5281 \times 10^3$ kg m^{-3} and all other $d_i = 0$. IMSL (International Mathematics and Statistics Library) [47] subroutines DQDAG and DLSARG are then called to solve Eqs. (2.77) and (2.78) respectively for the coefficients $d_3, \cdots, d_{N'}$. In PREM, $N' = 4$; but in our modified PREM it is $N' = 12$ because we have shown that a higher value of N' results in a much faster convergence of β to β_d.

To find the value d_1, we use the mass conservation of the fluid core (FC) as a constraint and proceed as follows:

$$\int_{FC} \rho_0 dV = \frac{4}{3}\pi\overline{\rho}b^3, \tag{2.83}$$

where $\overline{\rho}$ is the average density of the outer core. Substituting Eq. (2.74) and $dV = r^2 \sin^2\theta dr d\theta d\phi$ in Eq. (2.83), then

$$4\pi R^3 \sum_{j=1}^{N'} d_j \int_0^{\frac{b}{R}} x^{j+1} dx = \frac{4}{3}\pi\overline{\rho}b^3,$$

which can be written as

$$d_1 = \overline{\rho} - 3\left(\frac{R}{b}\right)^3 \sum_{j=3}^{N'} \frac{d_j}{j+2}\left(\frac{b}{R}\right)^{j+2}. \tag{2.84}$$

From Eq. (2.73), we get

$$\beta = 1 + \frac{\alpha^2}{\rho_0 g_0}\frac{d\rho_0}{dr}. \tag{2.85}$$

Once d_j are known, we use the equations (2.71), (2.74), (2.75) and (2.82) in equation (2.85) to solve for β. If $|\beta - \beta_d| > \varkappa$, where $\varkappa$ is the desired accuracy, we use the new d_j as the starting values, this process is repeated until $|\beta - \beta_d| \leq \varkappa$. In this thesis, we set $\varkappa = 10^{-6}$, and compute the coefficients of the modified density of the fluid core (see

Appendix C.1). In Tables 2.1 and 2.2, we show d's for different values of β. Figure 2.6 shows the density profile for a core model with $\beta = -0.001$.

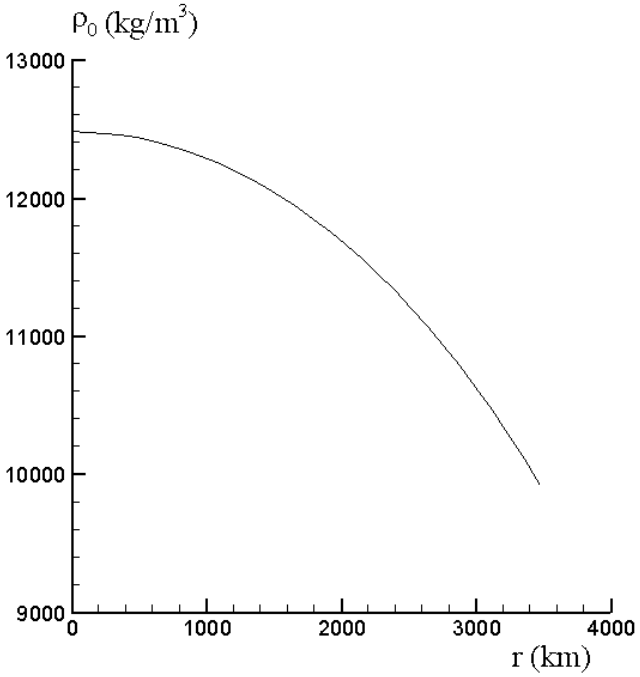

Figure 2.6: The density profile of the modified PREM with the stability parameter set at -0.001.

Table 2.1: The coefficients of the density profile ($\mathrm{kg\,m^{-3}}$) for the neutrally and stably stratified core models with no inner core

d_j	$\beta = 0.0$	$\beta = -0.001$	$\beta = -0.002$	$\beta = -0.005$
d_1	1.2477×10^4	1.2482×10^4	1.2480×10^4	1.2485×10^4
d_2	0.0	0.0	0.0	00
d_3	-7.7409×10^3	-7.7507×10^3	-7.7606×10^3	-7.7790×10^3
d_4	5.1132×10^{-2}	4.2523×10^{-2}	1.5008×10^{-2}	5.0953×10^{-2}
d_5	-2.5078×10^3	-2.5065×10^3	-2.5072×10^3	-2.5018×10^3
d_6	1.6271×10^1	1.4672×10^1	3.2642×10^1	1.6297×10^1
d_7	-9.8071×10^2	-9.6947×10^2	-1.0696×10^3	-9.7174×10^2
d_8	5.4545×10^2	5.1131×10^2	8.7488×10^2	5.4864×10^2
d_9	-1.8448×10^3	-1.7655×10^3	-2.5853×10^3	-1.8507×10^3
d_{10}	2.7434×10^3	2.6347×10^3	3.7762×10^3	2.7708×10^3
d_{11}	-2.8131×10^3	-2.7286×10^3	-3.6198×10^3	-2.8435×10^3
d_{12}	1.2135×10^3	1.1867×10^3	1.4871×10^3	1.2335×10^3

Table 2.2: The coefficients of the density profile ($\mathrm{kg\,m^{-3}}$) for the unstably stratified core models with no inner core

d_j	$\beta = 0.001$	$\beta = 0.002$	$\beta = 0.005$
d_1	1.2475×10^4	1.2473×10^4	1.2468×10^4
d_2	0.0	0.0	0.0
d_3	-7.7308×10^3	-7.7210×10^3	-7.6915×10^3
d_4	1.3254×10^{-2}	6.0311×10^{-2}	1.1308×10^{-2}
d_5	-2.5104×10^3	-2.5102×10^3	-2.5146×10^3
d_6	2.9605×10^1	1.7672×10^1	2.6303×10^1
d_7	-1.0576×10^2	-9.9227×10^2	-1.0461×10^3
d_8	8.1070×10^2	5.7153×10^2	7.4178×10^2
d_9	-2.4402×10^3	-1.9032×10^3	-2.2857×10^3
d_{10}	3.5617×10^3	2.8157×10^3	3.3319×10^3
d_{11}	-3.4462×10^3	-2.8644×10^3	-3.2560×10^3
d_{12}	1.4227×10^3	1.1226×10^3	1.3523×10^3

Chapter 3

The Galerkin Method and its Application of to the Governing Equations

---

The profound study of nature is the most fertile source of mathematical discoveries.

– Joseph Fourier

In this chapter, following Seyed-Mahmoud [35], we introduce the Galerkin method applicable to a system of simultaneous partial differential equations subject to boundary conditions. We use this method to solve the 3PD for the inertial modes of the sphere and spherical shell core models with various values of β.

3.1 The Galerkin Method

Russian mathematician Boris Grigoryevich Galerkin introduced the Galerkin method. This method is a tool to approximate the solution of an operator equation in the form of a linear combination of the elements of a linearly independent system. Nzikou [48] used this method to solve the 3PD for the eigenperiods and eigenfunctions of the Earth's Slichter modes. For testing the validity of this method, he also computed the frequencies and displacement eigenfunctions of some of the inertial modes of a compressible and neutrally stratified Earth's fluid core. Following Seyed-Mahmoud [35] and Nzikou [48], and we consider a set of functions $X = (X_1, X_2, X_3, \cdots, X_{N'})$ which satisfies a set of simultaneous PDEs in a region V,

$$\sum_{j=1}^{N'} L_{ij} X_j = 0 \tag{3.1}$$

for every i $(i = 1, \cdots, N')$, where L_{ij} are linear (maybe complex) partial differential operators.

Suppose that there are a number of associated boundary conditions satisfied on the boundary S of volume V, such that

$$\sum_{j=1}^{N'} B_{ij} X_j = 0 \tag{3.2}$$

for every i $(i = 1, \cdots, N')$, where B_{ij} are linear operators. Using a basis set f_k, $k = 1, \cdots, L$, we introduce trial functions

$$X_j = \sum_{k=1}^{L} C_{jk} f_k \tag{3.3}$$

for every j $(j = 1, \cdots, N')$, which need not a priori satisfy the boundary conditions. The Galerkin method tries to make $\sum_j L_{ij} X_j$ as nearly null as possible by requiring

$$\sum_{j=1}^{N'} \sum_{k=1}^{L} \int_V f_l^* L_{ij} C_{jk} f_k dV = 0, \tag{3.4}$$

where $l = 1, \cdots, L$, and * denotes the complex conjugate. The Eq. (3.4) can be written as a matrix form by setting up 1-1 correspondence $m = N'(l-1) + i$ and $n = N'(k-1) + j$,

$$\sum_n G_{mn} a_n = 0, (m = 1, \cdots, LN') \tag{3.5}$$

with $a_n = C_{jk}$, and

$$G_{mn} = H_{lijk} = \int_V f_l^* L_{ij} f_k dV. \tag{3.6}$$

In general the trial functions do not a priori satisfy the boundary conditions. We choose a set of basis functions ψ_l (i.e., weight functions) equal in number to the basis functions defined in the trial functions X_j is used to reconstruct in Eq. (3.4) as

$$\sum_{j=1}^{N'} \sum_{k=1}^{L} [\int_V f_l^* L_{ij} C_{jk} f_k dV + \int_S \psi_l^* B_{ij} C_{jk} f_k dS] = 0,$$

which has a form

$$\sum_{j=1}^{N'} \sum_{k=1}^{L} F_{lijk} C_{jk} = 0 \tag{3.7}$$

with

$$F_{lijk} = H_{lijk} + \int_S \psi_l^* B_{ij} f_k dS. \tag{3.8}$$

The surface integral may be removed by applying the divergence theorem, which converts the volume integral to a surface one. When this is possible with a proper choice of weight functions, the boundary conditions are called natural.

3.2 Application of the Galerkin Method to the Governing Equations of Core Dynamics

As mentioned in chapter 2, the 3PD has been constructed by the three scalar potentials χ, V_1 and ζ. Spherical harmonics are used to construct basis functions for a Galerkin method [6, 32]

$$\chi = \sum_{n=|m|}^{N'} \sum_{l=1}^{L} D^m_{L(n-1)+l} f_l(x) P_n^m(\cos\theta) \exp(im\phi), \tag{3.9}$$

$$V_1 = \sum_{n=|m|}^{N'} \sum_{l=1}^{L} D^m_{L(N'+n-1)+l} f_l(x) P_n^m(\cos\theta) \exp(im\phi), \tag{3.10}$$

$$\zeta = \sum_{n=|m|}^{N'} \sum_{l=1}^{L} D^m_{L(2N'+n-1)+l} f_l(x) P_n^m(\cos\theta) \exp(im\phi), \tag{3.11}$$

for any m. Because of the orthogonality relation among spherical harmonics, the solutions (i.e., (3.9)-(3.11)) for any m take the form of one or other of two decoupled chains involving the spherical harmonics of order m and degree $|m|+2(j-1)$ or $|m|+2j-1$ respectively, with $j = 1, \cdots, N'$, where N' is the maximum degree of the spherical harmonics. In the Eqs. (3.9)-(3.11), D's are constants, $P_n^m(\cos\theta)\exp(im\phi)$ are the spherical harmonics of degree n and azimuthal order m, and P_n^m are the associated Legendre polynomials. In theory, L and N' are infinite, but in practice they represent the truncation levels in r and θ directions,

respectively. Here f_l are the Legendre functions of degree l. The argument x for f_l is $-1 \leq x \leq 1$. Now we choose x as

$$x = \frac{1}{b-a}[2r-(b+a)] \tag{3.12}$$

with a and b are the inner and outer radii of the spherical shell respectively ($a = 0$ for a spherical container) [32]. The orthogonality relation among the spherical harmonics are first used to remove the θ dependence of these equations.

3.2.1 Galerkin Formulation of the Poincaré Equation

As a test of the validity of our approach, we first apply a Galerkin method to the case of a homologous and incompressible core mode, the Poincaré core model, for which analytical solutions exist for the frequencies and the displacement eigenfunctions of its inertial modes. The Poincaré equation (i.e., Eq. (2.40)) is rewritten as

$$\nabla \cdot (\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla \chi) = 0, \tag{3.13}$$

where $\tilde{\Gamma}_{\mathbf{p}}$ is defined as Eq. (2.36). The Galerkin representation of Eq. (3.13) is

$$\int_V \psi_q^m \nabla \cdot (\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla \chi) dV = 0 \tag{3.14}$$

with

$$\psi_q^m = (f_k(r) P_q^m(\cos\theta) \exp(im\phi))^* \tag{3.15}$$

for $k = 1, \cdots, L$. We know

$$\nabla \cdot (\psi_q^m \tilde{\Gamma}_{\mathbf{p}} \cdot \nabla \chi) = \nabla \psi_q^m \cdot (\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla \chi) + \psi_q^m \nabla \cdot (\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla \chi). \tag{3.16}$$

Now Eq. (3.16) is used in Eq. (3.14) to get

$$\int_V \nabla \cdot (\psi_q^m \tilde{\Gamma}_{\mathbf{p}} \cdot \nabla\chi) dV - \int_V \nabla\psi_q^m \cdot (\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla\chi) dV = 0. \tag{3.17}$$

Applying the divergence theorem in the 1st term of Eq. (3.17) to get

$$\int_S \hat{n} \cdot (\psi_q^m \tilde{\Gamma}_{\mathbf{p}} \cdot \nabla\chi) dS - \int_V \nabla\psi_q^m \cdot (\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla\chi) dV = 0, \tag{3.18}$$

where $\hat{n}$ is the unit vector normal to the surface boundary (core mantle boundary). Using the Eq. (2.41) in Eq. (3.18), we get

$$\int_S \sigma^2(\sigma^2-1)\psi_q^m(\hat{n}\cdot\vec{u}) dS - \int_V \nabla\psi_q^m \cdot (\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla\chi) dV = 0. \tag{3.19}$$

Since the mantle is rigid in the sense that the mass elements in this region remains at solid body rotation, the boundary condition (i) [see section 2.2] that the normal component of displacement requires that

$$\hat{n} \cdot \vec{u} = 0. \tag{3.20}$$

Substituting Eq. (3.20) in Eq. (3.19) to get

$$\int_V \nabla\psi_q^m \cdot (\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla\chi) dV = 0, \tag{3.21}$$

with $\tilde{\Gamma}_{\mathbf{p}}$ as in Eq. (2.36).

Substituting Eqs. (3.9) and (3.15) in Eq. (3.21), and using $dV = x^2 \sin\theta d\theta dx d\phi$ and integrating with respect to ϕ, we get

$$\sum_{n=|m|}^{N'} \sum_{l=1}^{L} D^m_{L(n-1)+l} [\int_0^{\pi} \int_0^{x} \{(\sigma^2 - \cos^2\theta) P_n^m P_q^m \frac{df_l}{dx}\frac{df_k}{dx} x^2 + f_l P_q^m \frac{df_k}{dx} \sin\theta\cos\theta \frac{dP_n^m}{d\theta} x$$

$$+f_k\frac{df_l}{dx}\sin\theta\cos\theta\frac{dP_q^m}{d\theta}P_n^m x+\sigma^2 f_l f_k\frac{dP_q^m}{d\theta}\frac{dP_n^m}{d\theta}-f_l f_k\sin^2\theta\frac{dP_q^m}{d\theta}\frac{dP_n^m}{d\theta}$$
$$+m\sigma P_n^m P_q^m\left(f_k\frac{df_l}{dx}+f_l\frac{df_k}{dx}\right)x+m\sigma f_l f_k\frac{\cos\theta}{\sin\theta}\left(P_q^m\frac{dP_n^m}{d\theta}+P_n^m\frac{dP_q^m}{d\theta}\right)$$
$$+\frac{m^2\sigma^2}{\sin^2\theta}f_l f_k P_n^m P_q^m\}\sin\theta d\theta dx]=0, \tag{3.22}$$

where the arguments of the associated Legendre polynomials, $P_n^m(\cos\theta)$ and $P_q^m(\cos\theta)$, are $\cos\theta$ (i.e., $-1\le\cos\theta\le 1$), and for convenience we have dropped those from all terms in the below.

Using the following identities

$$\cos^2\theta=\frac{1}{3}(2P_2+1), \tag{3.23}$$
$$\sin\theta\cos\theta=-\frac{1}{3}P_2^1, \tag{3.24}$$

and using integration by parts and the orthogonality relation among the spherical harmonics, we get

$$\int_0^\pi\frac{\cos\theta}{\sin\theta}\left(P_q^m\frac{dP_n^m}{d\theta}+P_n^m\frac{dP_q^m}{d\theta}\right)\sin\theta d\theta=\int_0^\pi P_n^m P_q^m\sin\theta d\theta, \tag{3.25}$$
$$\int_0^\pi\sin\theta\cos\theta\frac{dP_q^m}{d\theta}P_n^m\sin\theta d\theta=-\frac{1}{3}\int_0^\pi\left(6P_2P_n^m-P_2^1\frac{dP_n^m}{d\theta}\right)P_q^m\sin\theta d\theta, \tag{3.26}$$
$$\int_0^\pi\sin^2\theta\frac{dP_q^m}{d\theta}\frac{dP_n^m}{d\theta}\sin\theta d\theta=-\int_0^\pi[(m^2-\frac{2}{3}n(n+1)+\frac{2}{3}n(n+1)P_2)P_n^m$$
$$-\frac{2}{3}P_2^1\frac{dP_n^m}{d\theta}]P_q^m\sin\theta d\theta, \tag{3.27}$$
$$\int_0^\pi\left(\frac{dP_q^m}{d\theta}\frac{dP_n^m}{d\theta}+\frac{m^2}{\sin^2\theta}P_q^m P_n^m\right)\sin\theta d\theta=n(n+1)\int_0^\pi P_q^m P_n^m\sin\theta d\theta. \tag{3.28}$$

Eq. (3.22) then becomes

$$\sum_{n=|m|}^{N'}\sum_{l=1}^{L}D_{L(n-1)+l}^m[\int_0^\pi\int_0^x\{[(\sigma^2-\frac{1}{3})\frac{df_l}{dx}\frac{df_k}{dx}x^2+m\sigma(f_k\frac{df_l}{dx}+f_l\frac{df_k}{dx})x$$
$$+f_l f_k(m^2-\frac{2}{3}n(n+1))+f_l f_k(n(n+1)\sigma^2+m\sigma)]P_n^m P_q^m$$

$$-\frac{2}{3}[\frac{df_l}{dx}\frac{df_k}{dx}x^2+3f_k\frac{df_l}{dx}x-n(n+1)f_lf_k]P_2P_n^mP_q^m$$
$$-\frac{1}{3}[(f_l\frac{df_k}{dx}-f_k\frac{df_l}{dx})x+2f_lf_k]P_2^1\frac{dP_n^m}{d\theta}P_q^m\}\sin\theta d\theta dx]=0. \tag{3.29}$$

To integrate this equation with respect to θ, we need the following two identities:

$$P_2P_n^m = A_n^mP_{n-2}^m+B_n^mP_n^m+C_n^mP_{n+2}^m, \tag{3.30}$$
$$P_2^1\frac{dP_n^m}{d\theta} = 2(n+1)A_n^mP_{n-2}^m+3B_n^mP_n^m-2nC_n^mP_{n+2}^m, \tag{3.31}$$

where

$$A_n^m = \frac{3}{2}\frac{(n+m)(n+m-1)}{(2n+1)(2n-1)}, \tag{3.32}$$
$$B_n^m = \frac{n(n+1)-3m^2}{(2n+3)(2n-1)}, \tag{3.33}$$
$$C_n^m = \frac{3}{2}\frac{(n+2-m)(n+1-m)}{(2n+3)(2n+1)}. \tag{3.34}$$

Substituting Eqs. (3.30), and (3.31) in Eq. (3.29), we find the Galerkin formulation of the Poincaré equation

$$\sum_{n=|m|}^{N'}\sum_{l=1}^{L}\int_0^\pi P_n^mP_q^m\sin\theta d\theta[\int_0^x\{D_{L(n-1)+l}^m[(\sigma^2-\frac{1}{3})\frac{df_l}{dx}\frac{df_k}{dx}x^2+m\sigma(f_k\frac{df_l}{dx}$$
$$+f_l\frac{df_k}{dx})x+f_lf_k(m^2-\frac{2}{3}n(n+1))+f_lf_k(n(n+1)\sigma^2+m\sigma)$$
$$-B_n^m\{\frac{2}{3}\frac{df_l}{dx}\frac{df_k}{dx}x^2+(f_l\frac{df_k}{dx}+f_k\frac{df_l}{dx})x-(\frac{2}{3}n(n+1)-2)f_lf_k\}]$$
$$-\frac{2}{3}D_{L(n+1)+l}^mA_{n+2}^m[\frac{df_l}{dx}\frac{df_k}{dx}x^2+\{n(f_l\frac{df_k}{dx}-f_k\frac{df_l}{dx})+3f_l\frac{df_k}{dx}\}x$$
$$-n(n+3)f_lf_k]-\frac{2}{3}D_{L(n-3)+l}^mC_{n-2}^m[\frac{df_l}{dr}\frac{df_k}{dx}x^2-\{(n-2)$$
$$\times(f_l\frac{df_k}{dx}-f_k\frac{df_l}{dx})-3f_k\frac{df_l}{dx}\}x-(n+1)(n-2)f_lf_k]\}dx]=0. \tag{3.35}$$

The orthogonality relation is given by

$$\int_0^{\pi} P_n^m P_q^m \sin\theta d\theta = \frac{2}{2q+1}\frac{(q+m)!}{(q-m)!}\delta_{q,n}, \tag{3.36}$$

where $\delta_{q,n}$ is the Kronecker delta, which is zero for $n \neq q$ and 1 only for $n = q$. We will drop the constant in front of the orthogonality relation because the right side of Eq. (3.35) is equal to zero and this term is a common factor. The Galerkin formulation of the Poincaré equation now only depends on, $x = r/R$

$$\begin{aligned}
\sum_{l=1}^{L}\int_0^x \{D^m_{L(q-1)+l}[(\sigma^2-\frac{1}{3})\frac{df_l}{dx}\frac{df_k}{dx}x^2 + m\sigma(f_k\frac{df_l}{dx}+f_l\frac{df_k}{dx})x + f_l f_k\{m^2 \\
-q(q+1)(\frac{2}{3}-\sigma^2)+m\sigma\} - B^m_q\{\frac{2}{3}\frac{df_l}{dx}\frac{df_k}{dx}x^2 + (f_l\frac{df_k}{dx}+f_k\frac{df_l}{dx})x \\
-(\frac{2}{3}q(q+1)-2)f_l f_k\}] - \frac{2}{3}D^m_{L(q+1)+l}A^m_{q+2}[\frac{df_l}{dx}\frac{df_k}{dx}x^2 + \{q(f_l\frac{df_k}{dx} \\
-f_k\frac{df_l}{dx})+3f_l\frac{df_k}{dx}\}x - q(q+3)f_l f_k] - \frac{2}{3}D^m_{L(q-3)+l}C^m_{q-2}[\frac{df_l}{dx}\frac{df_k}{dx}x^2 \\
-\{(q-2)(f_l\frac{df_k}{dx}-f_k\frac{df_l}{dx})-3f_k\frac{df_l}{dx}\}x-(q+1)(q-2)f_l f_k]\}dx = 0,
\end{aligned} \tag{3.37}$$

for every $k = 1,\cdots,L$ and $q = 1,\cdots,N'$. The equation (3.37) leads to the matrix representation (given in appendix B.1) as

$$A\psi = 0, \tag{3.38}$$

where A is a $N'L \times N'L$ matrix, and ψ is a $N'L$ vector representing the coefficients of D_i in Eq. (3.37). Non-trivial solutions of Eq. (3.38) exist only if the determinant of A vanishes. The non-dimensional frequencies, σ, of the Poincaré model are then computed by finding the roots of $detA = 0$. Our numerical results for the Poincaré core model are reported in the next chapter 4 (see, Table 4.4 in column 2).

3.2.2 Galerkin Formulation of the Momentum Equation

The momentum equation (2.34) is

$$\nabla \cdot \{\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla(\chi - V_1) - \beta \vec{C}^* \zeta\} - \sigma^2(\sigma^2 - 1)\zeta = 0,$$

where $\tilde{\Gamma}_{\mathbf{p}}$ and $\vec{C}$ are defined as in Eqs. (2.36) and (2.38), respectively. The Galerkin representation of Eq. (2.34) is

$$\int_V \psi_q^m [\nabla \cdot \{\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla(\chi - V_1) - \beta \vec{C}^* \zeta\} - \sigma^2(\sigma^2 - 1)\zeta] dV = 0, \tag{3.39}$$

and ψ_q^m is given by Eq. (3.15). Note that

$$\begin{aligned} \nabla \cdot (\psi_q^m \{\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla (\chi - V_1) - \beta \vec{C}^* \zeta\}) = \nabla \psi_q^m \cdot \{\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla (\chi - V_1) - \beta \vec{C}^* \zeta\} \\ + \psi_q^m \nabla \cdot \{\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla (\chi - V_1) - \beta \vec{C}^* \zeta\}. \end{aligned} \tag{3.40}$$

Eq. (3.39) then becomes

$$\begin{aligned} \int_V \nabla \cdot (\psi_q^m \{\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla (\chi - V_1) - \beta \vec{C}^* \zeta\}) dV - \int_V [\nabla \psi_q^m \cdot \{\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla (\chi - V_1) - \beta \vec{C}^* \zeta\} \\ + \sigma^2(\sigma^2 - 1)\psi_q^m \zeta] dV = 0. \end{aligned} \tag{3.41}$$

Applying the divergence theorem in the 1st term of Eq.(3.41) and using the boundary condition (i) [see section 2.2] that the normal component of displacement requires that $\hat{n} \cdot \vec{u} = 0$ because the mantle is rigid, then the 1st term of Eq.(3.41) vanishes (similar process as shown in the case of the Poincaré equation in subsection 3.2.1); and Eq.(3.41) becomes

$$\int_V [\nabla \psi_q^m \cdot \{\tilde{\Gamma}_{\mathbf{p}} \cdot \nabla (\chi - V_1)\} - \{\nabla \psi_q^m \cdot \beta \vec{C}^* \zeta - \sigma^2(\sigma^2 - 1)\psi_q^m \zeta\}] dV = 0. \tag{3.42}$$

Using the results from the Poincare model (i. e., Eq. (3.35)), the 1st term of Eq. (3.42)

is written as

$$\int_V \nabla\psi_q^m \cdot \tilde{\Gamma_{\mathbf{p}}} \cdot \nabla\,(\chi - V_1)dv = 2\pi \sum_{n=|m|}^{N'} \sum_{l=1}^{L} \int_0^{\pi} P_n^m P_q^m \sin\theta d\theta [\int_0^x \{(D^m_{L(n-1)+l}$$
$$-D^m_{L(N'+n-1)+l})[(\sigma^2 - \frac{1}{3})\frac{df_l}{dx}\frac{df_k}{dx}x^2 + m\sigma(f_k\frac{df_l}{dx} + f_l\frac{df_k}{dx})x + f_l f_k(m^2$$
$$-\frac{2}{3}n(n+1)) + f_l f_k(n(n+1)\sigma^2 + m\sigma) - B_n^m\{\frac{2}{3}\frac{df_l}{dx}\frac{df_k}{dx}x^2 + (f_l\frac{df_k}{dx} + f_k\frac{df_l}{dx})x$$
$$-(\frac{2}{3}n(n+1) - 2)f_l f_k\}] - \frac{2}{3}(D^m_{L(n+1)+l} - D^m_{L(N'+n+1)+l})A^m_{n+2}[\frac{df_l}{dx}\frac{df_k}{dx}x^2$$
$$+\{n(f_l\frac{df_k}{dx} - f_k\frac{df_l}{dx}) + 3f_l\frac{df_k}{dx}\}x - n(n+3)f_l f_k] - \frac{2}{3}(D^m_{L(n-3)+l} - D^m_{L(N'+n-3)+l})$$
$$\times C^m_{n-2}[\frac{df_l}{dr}\frac{df_k}{dx}x^2 - \{(n-2)(f_l\frac{df_k}{dx} - f_k\frac{df_l}{dx}) - 3f_k\frac{df_l}{dx}\}x$$
$$-(n+1)(n-2)f_l f_k]\}dx]. \tag{3.43}$$

In the spherical coordinates system, Eq. (2.38) becomes

$$\vec{C} = -\sigma^2 g_r \hat{r} + g_r \cos\theta(\hat{r}\cos\theta - \hat{\theta}\sin\theta) + i\sigma g_r \sin\theta\hat{\phi},$$
$$\therefore \vec{C}^* = [-\sigma^2 g_r + g_r\cos^2\theta]\hat{r} - g_r\cos\theta\sin\theta\hat{\theta} - i\sigma g_r\sin\theta\hat{\phi} \tag{3.44}$$

with

$$g_r = -g_0 + \frac{1}{6}x. \tag{3.45}$$

Using Eqs. (3.11), (3.15) and (3.44), and integrating with respect to ϕ, the 2nd term of Eq. (3.42) becomes

$$\int_V \{\nabla\psi_q^m \cdot \beta\vec{C}^*\zeta - \sigma^2(\sigma^2-1)\psi_q^m\zeta\}dv = 2\pi \sum_{n=|m|}^{N'} \sum_{l=1}^{L} \int_0^x \int_0^{\pi} D^m_{L(2N'+n-1)+l}$$
$$\times[\beta f_l P_n^m P_q^m(-\sigma^2 g_r + g_r\cos^2\theta)\frac{df_k}{dx} - \frac{\beta}{x}f_l f_k g_r \cos\theta\sin\theta\frac{dP_q^m}{d\theta}P_n^m$$
$$-\frac{m\sigma\beta}{x}f_l f_k P_n^m P_q^m g_r - \sigma^2(\sigma^2-1)f_l f_k P_n^m P_q^m]x^2\sin\theta d\theta dx. \tag{3.46}$$

Using the following identities:

$$\cos^2\theta = \frac{1}{3}(2P_2+1), \tag{3.47}$$

$$\int_0^\pi \cos\theta\sin\theta\frac{dP_q^m}{d\theta}P_n^m\sin\theta d\theta = -\int_0^\pi\left(2P_2P_n^m-\frac{1}{3}P_2^1\frac{dP_n^m}{d\theta}\right)P_q^m\sin\theta d\theta, \tag{3.48}$$

equation (3.46) becomes

$$\int_V\{\nabla\psi_q^m\cdot\beta\vec{C}^*\zeta-\sigma^2(\sigma^2-1)\psi_q^m\zeta\}dv = 2\pi\sum_{n=|m|}^{N'}\sum_{l=1}^{L}\int_0^x\int_0^\pi D^m_{L(2N'+n-1)+l}$$
$$\times\{[\beta f_l(-\sigma^2 g_r+\frac{1}{3}g_r)\frac{df_k}{dx}-\frac{m\sigma\beta}{x}f_lf_kg_r-\sigma^2(\sigma^2-1)f_lf_k]P_n^mP_q^m$$
$$+[\frac{2}{3}\beta f_lg_r\frac{df_k}{dx}+\frac{2}{x}\beta f_lf_kg_r]P_2P_n^mP_q^m-\frac{\beta}{3x}f_lf_kg_rP_2^1\frac{dP_n^m}{d\theta}P_q^m\}x^2\sin\theta d\theta dx. \tag{3.49}$$

Substituting Eqs. (3.30), and (3.31) in Eq. (3.49) we get

$$\int_V\{\nabla\psi_q^m\cdot\beta\vec{C}^*\zeta-\sigma^2(\sigma^2-1)\psi_q^m\zeta\}dv = 2\pi\sum_{n=|m|}^{N'}\sum_{l=1}^{L}\int_0^\pi P_n^mP_q^m\sin\theta d\theta$$
$$\times\int_0^x\{D^m_{L(2N'+n-1)+l}[\beta f_l(-\sigma^2g_r+\frac{1}{3}g_r)\frac{df_k}{dx}x^2-(m\sigma\beta g_rx+\sigma^2(\sigma^2-1)x^2)f_lf_k$$
$$+B_n^m\beta g_r\{\frac{2}{3}f_l\frac{df_k}{dx}x^2+f_lf_kx\}]+D^m_{L(2N'+n+1)+l}A^m_{n+2}\beta g_r[\frac{2}{3}f_l\frac{df_k}{dx}x^2+(2-\frac{2}{3}(n+3))$$
$$\times f_lf_kx]+D^m_{L(2N'+n-3)+l}C^m_{n-2}\beta g_r[\frac{2}{3}f_l\frac{df_k}{dx}x^2+(2+\frac{2}{3}(n-2))f_lf_kx]dx. \tag{3.50}$$

Substituting Eqs. (3.43) and (3.50) in Eq. (3.42) and applying the orthogonality relation (3.36), we get the Galerkin formulation of the momentum equation

$$\sum_{l=1}^{L}\int_0^x\{(D^m_{L(q-1)+l}-D^m_{L(N'+q-1)+l})[(\sigma^2-\frac{1}{3})\frac{df_l}{dx}\frac{df_k}{dx}x^2+m\sigma(f_k\frac{df_l}{dx}+f_l\frac{df_k}{dx})x$$
$$+f_lf_k(m^2-\frac{2}{3}q(q+1))+f_lf_k(q(q+1)\sigma^2+m\sigma)-B_q^m\{\frac{2}{3}\frac{df_l}{dx}\frac{df_k}{dx}x^2+(f_l\frac{df_k}{dx}$$
$$+f_k\frac{df_l}{dx})x-(\frac{2}{3}q(q+1)-2)f_lf_k\}]-\frac{2}{3}(D^m_{L(q+1)+l}-D^m_{L(N'+q+1)+l})A^m_{q+2}$$
$$\times[\frac{df_l}{dx}\frac{df_k}{dx}x^2+\{q(f_l\frac{df_k}{dx}-f_k\frac{df_l}{dx})+3f_l\frac{df_k}{dx}\}x-q(q+3)f_lf_k]-\frac{2}{3}(D^m_{L(q-3)+l}$$

$$
\begin{aligned}
&-D^m_{L(N'+q-3)+l})C^m_{q-2}[\frac{df_l}{dr}\frac{df_k}{dx}x^2-\{(q-2)(f_l\frac{df_k}{dx}-f_k\frac{df_l}{dx})-3f_k\frac{df_l}{dx}\}x\\
&-(q+1)(q-2)f_lf_k]-D^m_{L(2N'+q-1)+l}[\beta f_l(-\sigma^2 g_r+\frac{1}{3}g_r)\frac{df_k}{dx}x^2-(m\sigma\beta g_r x\\
&+\sigma^2(\sigma^2-1)x^2)f_lf_k+B^m_q\beta g_r(\frac{2}{3}f_l\frac{df_k}{dx}x^2+f_lf_kx)]-D^m_{L(2N'+q+1)+l}A^m_{q+2}\beta g_r\\
&\times[\frac{2}{3}f_l\frac{df_k}{dx}x^2+\{2-\frac{2}{3}(q+3)\}f_lf_kx]-D^m_{L(2N'+q-3)+l}C^m_{q-2}\beta g_r\\
&\times[\frac{2}{3}f_l\frac{df_k}{dx}x^2+\{2+\frac{2}{3}(q-2)\}f_lf_kx\}dx=0, \qquad (3.51)
\end{aligned}
$$

for every $k=1,\cdots,L$ and $q=1,\cdots,N'$.

3.2.3 Galerkin Formulation of the Poisson's Equation

The Poisson's equation (2.33) is

$$\nabla^2 V_1-4\pi G\left\{\beta\zeta-\frac{1-\beta}{\alpha^2}\chi\right\}=0.$$

The Galerkin representation of Eq.(2.33) is

$$\int_V \psi^m_q\left[\nabla^2 V_1-4\pi G\left\{\beta\zeta-\frac{1-\beta}{\alpha^2}\chi\right\}\right]dV=0, \qquad (3.52)$$

and ψ^m_q is given by Eq. (3.15). Using

$$\nabla\cdot(\psi^m_q\nabla V_1)=\nabla\psi^m_q\cdot\nabla V_1+\psi^m_q\nabla^2 V_1. \qquad (3.53)$$

Eq.(3.52) becomes

$$\int_V\left[\nabla\cdot(\psi^m_q\nabla V_1)-\nabla\psi^m_q\cdot\nabla V_1-4\pi G\psi^m_q\left\{\beta\zeta-\frac{1-\beta}{\alpha^2}\chi\right\}\right]dV=0. \qquad (3.54)$$

Applying the divergence theorem we get

$$\int_S\hat{n}\cdot(\psi^m_q\nabla V_1)dS-\int_V\left[\nabla\psi^m_q\cdot\nabla V_1+4\pi G\psi^m_q\left\{\beta\zeta-\frac{1-\beta}{\alpha^2}\chi\right\}\right]dV=0,$$

$$\therefore \int_S \psi_q^m (\hat{n}\cdot\nabla V_1) dS - \int_V [\frac{\partial \psi_q^m}{\partial x}\frac{\partial V_1}{\partial x} + \frac{1}{x^2}\frac{\partial \psi_q^m}{\partial \theta}\frac{\partial V_1}{\partial \theta} + \frac{1}{x^2\sin^2\theta}\frac{\partial \psi_q^m}{\partial \phi}\frac{\partial V_1}{\partial \phi} - 4\pi G\psi_q^m \left\{\beta\zeta - \frac{1-\beta}{\alpha^2}\chi\right\}]dV = 0. \tag{3.55}$$

As mentioned in the section 2.5, we assume that the mantle is rigid, therefore the gravitational potential V_1 satisfies the Laplace's equation in this region

$$\nabla^2 V_1 = 0, \tag{3.56}$$

which has solutions of the form

$$V_1 = \sum_{n=|m|}^{N'} [A_n x^n + B_n x^{-(n+1)}] P_n^m(\cos\theta)\exp(im\phi). \tag{3.57}$$

In the mantle, $A_n = 0$, since otherwise V_1 would blow up at infinity. Hence, the perturbation in the gravity potential is

$$V_1 = \sum_{n=|m|}^{N'} B_n x^{-(n+1)} P_n^m(\cos\theta)\exp(im\phi). \tag{3.58}$$

From Eqs. (3.10) and (3.58), we get

$$B_n = x^{n+1}\sum_{l=1}^{L} D^m_{L(N'+n-1)+l} f_l(x), \tag{3.59}$$

for $n = 1,\cdots,N'$. Since V_1 is continuous across CMB, i.e., $(V_1)_{R_{CMB+}} = (V_1)_{R_{CMB-}}$ and $(\hat{n}\cdot\nabla V_1)_{R_{CMB+}} = (\hat{n}\cdot\nabla V_1)_{R_{CMB-}}$, then we get

$$(\hat{n}\cdot\nabla V_1)_{R_{CMB-}} = \frac{-(n+1)}{R_{CMB}}(V_1)_{R_{CMB-}}, \tag{3.60}$$

where R_{CMB+} is the radius of the CMB on the mantle side while R_{CMB-} is the same on the liquid side.

Using Eqs. (3.9) - (3.11), (3.15) and (3.60) in Eq. (3.55), we get

$$\sum_{n=|m|}^{N'}\sum_{l=1}^{L}\int_0^{\pi} P_n^m P_q^m \sin\theta d\theta \{D^m_{L(N'+n-1)+l}[(n+1)R_{CMB}f_l(R_{CMB})f_k(R_{CMB}) + \int_0^x \left(\frac{df_k}{dx}\frac{df_l}{dx}x^2 + n(n+1)f_l f_k\right)dx] + 4\pi G D^m_{L(2N'+n-1)+l}\int_0^x \beta f_k f_l x^2 dx - 4\pi G D^m_{L(n-1)+l}\int_0^x \left(\frac{1-\beta}{\alpha^2}\right) f_l f_k x^2 dx\} = 0. \tag{3.61}$$

In the above equation we have used the relation:

$$\int_0^{\pi}\left(\frac{dP_q^m}{d\theta}\frac{dP_n^m}{d\theta} + \frac{m^2}{\sin^2\theta}P_q^m P_n^m\right)\sin\theta d\theta = n(n+1)\int_0^{\pi} P_q^m P_n^m \sin\theta d\theta.$$

Applying the orthogonality relation (3.36) in Eq. (3.61), we get the Galerkin formulation of the Poisson's equation

$$\sum_{l=1}^{L}\{D^m_{L(N'+q-1)+l}[(q+1)R_{CMB}f_l(R_{CMB})f_k(R_{CMB}) + \int_0^x \left(\frac{df_k}{dx}\frac{df_l}{dx}x^2 + q(q+1)f_l f_k\right)dx] + 4\pi G D^m_{L(2N'+q-1)+l} \times \int_0^x \beta f_k f_l x^2 dx - 4\pi G D^m_{L(q-1)+l}\int_0^x \left(\frac{1-\beta}{\alpha^2}\right) f_l f_k x^2 dx\} = 0, \tag{3.62}$$

for $k = 1, \cdots, L$ and $q = 1, \cdots, N'$.

3.2.4 Galerkin Formulation of the Entropy Equation

The entropy equation (2.35) is

$$\vec{C}\cdot\nabla(\chi - V_1) - \sigma^2(\sigma^2-1)\chi - B\zeta = 0,$$

where B and $\vec{C}$ are defined as Eqs. (2.37) and (2.38), respectively.

The Galerkin representation of Eq. (2.35) is

$$\int_V \psi_q^m[\vec{C}\cdot\nabla\chi-\sigma^2(\sigma^2-1)\chi-\vec{C}\cdot\nabla V_1-B\zeta]dV=\ 0, \tag{3.63}$$

and ψ_q^m is given by Eq. (3.15). In the spherical coordinates system, Eq. (2.37) and (2.38) become

$$B = \alpha^2\sigma^2(\sigma^2-1)+\beta[\sigma^2 g_0^2-g_r^2\cos^2\theta], \tag{3.64}$$

$$\vec{C} = [-\sigma^2 g_r+g_r\cos^2\theta]\hat{r}-g_r\cos\theta\sin\theta\hat{\theta}+i\sigma g_r\sin\theta\hat{\phi}, \tag{3.65}$$

where g_r is given by Eq. (3.45). Now, substituting Eqs. (3.9) - (3.11), (3.15) and (3.64) - (3.65) in Eq. (3.2.4), inserting $dV=x^2\sin\theta d\theta dx d\phi$ and integrating with respect to ϕ, we get

$$\begin{aligned}
&\sum_{n=|m|}^{N'}\sum_{l=1}^{L}\int_0^x\int_0^\pi[D^m_{L(n-1)+l}\left\{f_k\left(-\sigma^2 g_r+\frac{1}{3}g_r\right)\frac{df_l}{dx}-\left(\frac{m\sigma}{x}g_r+\sigma^2(\sigma^2-1)\right)f_l f_k\right\}\\
&-D^m_{L(N'+n-1)+l}\left\{f_k\left(-\sigma^2 g_r+\frac{1}{3}g_r\right)\frac{df_l}{dx}-\frac{m\sigma}{x}g_r\right\}-D^m_{L(2N'+n-1)+l}f_l f_k\{\alpha^2\sigma^2(\sigma^2-1)\\
&+\beta(\sigma^2 g_0^2-\frac{1}{3}g_r^2)\}]P_q^m P_n^m x^2\sin\theta d\theta dx+\sum_{n=|m|}^{N'}\sum_{l=1}^{L}\int_0^x\int_0^\pi\frac{2}{3}[D^m_{L(n-1)+l}g_r f_k\frac{df_l}{dx}\\
&-D^m_{L(N'+n-1)+l}g_r f_k\frac{df_l}{dx}+D^m_{L(2N'+n-1)+l}\beta g_r^2 f_l f_k]P_2 P_q^m P_n^m x^2\sin\theta d\theta dx\\
&+\sum_{n=|m|}^{N'}\sum_{l=1}^{L}\int_0^x\int_0^\pi\frac{1}{3x}\left[\left(D^m_{L(n-1)+l}-D^m_{L(N'+n-1)+l}\right)g_r f_l f_k\right]P_2^1\frac{dP_n^m}{d\theta}P_q^m x^2\sin\theta d\theta dx=\ 0,
\end{aligned} \tag{3.66}$$

for $k=1,\cdots,L$, and we have used the relations: $\cos^2\theta=\frac{1}{3}(2P_2+1)$ and $\sin\theta\cos\theta=-\frac{1}{3}P_2^1$. To integrate Eq. (3.66) with respect to θ, we use the Eqs. (3.30) - (3.31) in Eq. (3.66) and

get

$$\sum_{n=|m|}^{N'}\sum_{l=1}^{L}\int_0^{\pi}P_q^mP_n^m\sin\theta d\theta\int_0^x[D^m_{L(n-1)+l}\{f_k(-\sigma^2g_r+\frac{1}{3}g_r)\frac{df_l}{dx}-(\frac{m\sigma}{x}g_r+\sigma^2(\sigma^2-1))$$
$$\times f_lf_k+B_n^m(\frac{2}{3}g_rf_k\frac{df_l}{dx}+\frac{1}{x}g_rf_lf_k)\}-D^m_{L(N'+n-1)+l}\{f_k(-\sigma^2g_r+\frac{1}{3}g_r)\frac{df_l}{dx}-\frac{m\sigma}{x}g_r$$
$$+B_n^m(\frac{2}{3}g_rf_k\frac{df_l}{dx}+\frac{1}{x}g_rf_lf_k)\}-D^m_{L(2N'+n-1)+l}f_lf_k\{\alpha^2\sigma^2(\sigma^2-1)+\beta(\sigma^2g_0^2-\frac{1}{3}g_r^2$$
$$-\frac{2}{3}B_n^mg_r^2)\}+D^m_{L(n+1)+l}A^m_{n+2}\{\frac{2}{3}g_r(f_k\frac{df_l}{dx}+\frac{1}{x}(n+3)f_lf_k)\}-D^m_{L(N'+n+1)+l}A^m_{n+2}$$
$$\times\{\frac{2}{3}g_r(f_k\frac{df_l}{dx}+\frac{1}{x}(n+3)f_lf_k)\}+D^m_{L(2N'+n+1)+l}A^m_{n+2}\frac{2}{3}\beta g_r^2f_lf_k+D^m_{L(n-3)+l}C^m_{n-2}$$
$$\times\{\frac{2}{3}g_r(f_k\frac{df_l}{dx}-\frac{1}{x}(n-2)f_lf_k)\}-D^m_{L(N'+n-3)+l}C^m_{n-2}\{\frac{2}{3}g_r(f_k\frac{df_l}{dx}-\frac{1}{x}(n-2)f_lf_k)\}$$
$$+D^m_{L(2N'+n-3)+l}C^m_{n-2}\frac{2}{3}\beta g_r^2f_lf_k]x^2dx=\ 0. \tag{3.67}$$

Using the orthogonality relation (3.36), Eq. (3.67) becomes

$$\sum_{l=1}^{L}\int_0^x\int_0^x[D^m_{L(q-1)+l}\{f_k(-\sigma^2g_r+\frac{1}{3}g_r)\frac{df_l}{dx}x^2-(m\sigma g_rx+\sigma^2(\sigma^2-1)x^2)f_lf_k$$
$$+B_q^m(\frac{2}{3}g_rf_k\frac{df_l}{dx}x^2+g_rf_lf_kx)\}-D^m_{L(N'+q-1)+l}\{f_k(-\sigma^2g_r+\frac{1}{3}g_r)\frac{df_l}{dx}x^2-m\sigma g_rf_lf_kx$$
$$+B_q^m(\frac{2}{3}g_rf_k\frac{df_l}{dx}x^2+g_rf_lf_kx)\}-D^m_{L(2N'+q-1)+l}f_lf_k\{\alpha^2\sigma^2(\sigma^2-1)+\beta(\sigma^2g_0^2$$
$$-\frac{1}{3}g_r^2-\frac{2}{3}B_q^mg_r^2)\}x^2+D^m_{L(q+1)+l}A^m_{q+2}\{\frac{2}{3}g_r(f_k\frac{df_l}{dx}x^2+(q+3)f_lf_kx)\}$$
$$-D^m_{L(N'+q+1)+l}A^m_{q+2}\{\frac{2}{3}g_r(f_k\frac{df_l}{dx}x^2+(q+3)f_lf_kx)\}+D^m_{L(2N'+q+1)+l}A^m_{q+2}$$
$$\times\frac{2}{3}\beta g_r^2f_lf_kx^2+D^m_{L(q-3)+l}C^m_{q-2}\{\frac{2}{3}g_r(f_k\frac{df_l}{dx}x^2-(q-2)f_lf_kx)\}$$
$$-D^m_{L(N'+q-3)+l}C^m_{q-2}\{\frac{2}{3}g_r(f_k\frac{df_l}{dx}x^2-(q-2)f_lf_kx)\}$$
$$+D^m_{L(2N'+q-3)+l}C^m_{q-2}\frac{2}{3}\beta g_r^2f_lf_kx^2]dx=\ 0, \tag{3.68}$$

for every for $k=1,\cdots,L$, and $q=1,\cdots,N'$.

3.3 Matrix Representation of the 3PD and Eigenvalues

We have shown in previous section that the use of the orthogonality relation among spherical harmonics leads to the removal of the θ and ϕ dependence. Note that the Eqs. (3.51), (3.62) and (3.68) are functions of x only. These three equations lead to the matrix representation (see in appendix B.2) as

$$A'\psi' = 0, \tag{3.69}$$

where A' is a $3N'L \times 3N'L$ matrix, and ψ' is a $3N'L$ vector representing the coefficients of D_i in Eqs. (3.51), (3.62) and (3.68). Non-trivial solutions of Eq. (3.69) exist and only if the determinant of A' is zero. The non-dimensional frequencies, σ, of the modified core models are computed by finding the roots of the determinant of A'.

Chapter 4

Results and Discussions

True Laws of Nature cannot be linear.

– Albert Einstein

In this chapter, we first report the frequencies and displacements for some of the inertial modes for the Poincaré core model, for which analytical solutions exist, in order to validate our approach. We will then report the results for the modes of several stratified core models with different stability parameters. We will also explore the dependence of the frequencies on the stability parameter. At the end we report the results for spherical shell models of the core with various stability parameters. We also discuss the displacement and pressure eigenfunctions with the characteristics lines of those inertial modes for both modified sphere and spherical shell core models with different stability parameter.

4.1 Non-dimensional Frequencies of the Spherical Core Models

As mentioned in chapter 3, the non-dimensional frequencies, σ, of the inertial modes are the roots of the secular determinant of the coefficient matrix. We search for the zeros of the determinant of the coefficient matrix in the frequency range, $0 \leq \sigma \leq 1$ for axisymmetric modes (i.e., $m = 0$), and $|\sigma| \leq 1$ for $m = 1$. The FORTRAN program we used to compute this matrix is given Appendix C.2. We search to restrict the frequency range $|\sigma| \leq 1$ because the frequency range of these modes is $|\omega| \leq 2\Omega$ [23] (i.e., $|\sigma| \leq 1$). The frequency range, $0 \leq \sigma \leq 1$ for the $m = 0$ is selected because the frequency spectrum for $\sigma < 0$ is identical to that for $\sigma > 0$. However, for $m = 1$ there is no such symmetry, therefore, every possible frequency will be detected. Note that there are two truncations involved: N' represents

the truncation level along θ, and L represents the truncation level along r. To compute the desired non-dimensional frequencies, we first fix the value of N' and then increased L until the computed frequency shows convergence, then we will increase N' and repeat the process until the computed frequency converges for any N' or L. Tables 4.1, 4.2 and 4.3 show the convergence patterns for the (6,2,0) mode of the Poincaré model, the (6,1,0) mode of a sphere with $\beta = 0$, and the (4,2,1) mode of a sphere with $\beta = -0.005$, respectively. We use Greenspan's [23] notation to identify a mode for labeling (n,k,m), where n, k and m refer to the degree of the spherical harmonics, the order of the mode and the azimuthal wavenumber, respectively. We identify a mode using two criteria: (1) the frequency of the mode which must converge, and (2) the displacement patterns for which must be satisfied the boundary condition ($\hat{n} \cdot \vec{u}$ must be continuous across any the boundary) and they need to be regular at the center and similar to those of the Poincaré core model [32].

Table 4.1: The convergence pattern for the (6,2,0) mode of the Poincaré model.

L	$\sigma(N'=4)$	N'	$\sigma(L=6)$
4	0.827	3	0.826
5	0.829	4	0.828
6	0.830	5	0.830
7	0.830	6	0.830
8	0.830	7	0.830

Table 4.4 shows the non-dimensional frequencies for the some low order inertial modes of the Poincaré model, and a compressible and neutrally stratified fluid sphere. In Table 4.4, column 1 represents some of the low order modes of azimuthal wavenumber 0 and 1 and degree up to 6, columns 2 and 3 show the non-dimensional frequencies of those inertial modes for the Poincaré core model, for which analytical solutions exist [23], and a modified neutrally stratified core model, respectively. Our results for the Poincaré model are identical to the analytical results for the same model. The non-dimensional frequencies (column 4) of these inertial modes of the same modified neutrally stratified core model were

Table 4.2: The convergence pattern for the (6,1,0) mode of a sphere with $\beta = 0$.

L	$\sigma(N' = 4)$	N'	$\sigma(L = 5)$
4	0.463	5	0.471
5	0.471	6	0.470
6	0.472	7	0.472
7	0.472	8	0.472
8	0.472	9	0.472
9	0.472	10	0.472

Table 4.3: The convergence pattern for the (4,2,1) mode of a sphere with $\beta = -0.005$.

L	$\sigma(N' = 4)$	N'	$\sigma(L = 5)$
3	0.323	3	0.328
4	0.326	4	0.329
5	0.329	5	0.329
6	0.329	6	0.329
7	0.329	7	0.329
8	0.329	8	0.329

computed by Seyed-Mahmoud et al. [6]. We see a high agreement between our results in column 3 and their results.

In Tables 4.5 and 4.6, we show the non-dimensional frequencies of some of the low order inertial modes of a compressible and stratified fluid spheres with various values of the stability parameter β. In columns 2, 4 and 6 of these tables we show the modal frequencies, and in columns 3, 5 and 7 we show the absolute percentage differences for the respective frequencies compare to those for the Poincaré core model. It is clear that the frequencies of the (5,2,0), (6,2,0), (2,1,1), (3,2,1), (4,3,1), (5,4,1) and (6,5,1) modes are least affected by stratification. We note that these modes have the highest middle index number k for the respective n and m values. In addition, the frequencies of the (5,1,0), (6,1,0), (3,1,1), (4,1,1), (5,1,1) and (6,2,1) modes are significantly affected by stratification. These modes,

Table 4.4: Non-dimensional frequencies, $\sigma = \omega/2\Omega$, of some of the low order inertial modes for the Poincaré model, and a compressible and neutrally stratified fluid sphere. Column 3 is our results and column 4 (i.e., $\sigma'_{0.00}$) is Seyed-Mahmoud et al. [6].

Mode	σ_{pc}	$\sigma_{0.00}$	$\sigma'_{0.00}$
$(3,1,0)$	0.447	0.456	0.456
$(4,1,0)$	0.655	0.657	0.657
$(5,1,0)$	0.285	0.292	0.292
$(5,2,0)$	0.765	0.766	0.766
$(6,1,0)$	0.469	0.472	0.471
$(6,2,0)$	0.830	0.831	0.831
$(2,1,1)$	0.500	0.500	0.500
$(3,1,1)$	-0.088	-0.101	-0.101
$(3,2,1)$	0.755	0.748	0.748
$(4,1,1)$	-0.410	-0.421	-0.422
$(4,2,1)$	0.306	0.310	0.310
$(4,3,1)$	0.854	0.849	0.849
$(5,1,1)$	-0.592	-0.598	-0.598
$(5,3,1)$	0.523	0.522	0.522
$(5,4,1)$	0.903	0.899	0.899
$(6,1,1)$	-0.702	-0.706	-0.706
$(6,2,1)$	-0.269	-0.276	-0.276
$(6,3,1)$	0.220	0.224	0.224
$(6,4,1)$	0.653	0.652	0.652
$(6,5,1)$	0.931	0.928	0.928

Table 4.5: Non-dimensional frequencies, $\sigma = \omega/2\Omega$, of some of the low order inertial modes of a compressible and stably stratified fluid spheres with different stability parameter β. Columns next to the frequency columns show the absolute percentage difference between the respective modal frequencies of the specific core model and that of the Poincaré core model.

Mode	$\sigma_{-0.001}$	% diff	$\sigma_{-0.002}$	% diff	$\sigma_{-0.005}$	% diff
$(3,1,0)$	0.463	3.6	0.469	4.9	0.488	9.2
$(4,1,0)$	0.666	1.7	0.674	2.9	0.698	6.6
$(5,1,0)$	0.300	5.3	0.309	8.4	0.333	16.8
$(5,2,0)$	0.774	1.2	0.781	2.1	0.804	5.1
$(6,1,0)$	0.483	3.0	0.493	5.1	0.525	11.9
$(6,2,0)$	0.838	1.0	0.845	1.8	0.866	4.3
$(2,1,1)$	0.500	0.0	0.500	0.0	0.500	0.0
$(3,1,1)$	-0.119	35.2	-0.133	51.1	-0.170	93.1
$(3,2,1)$	0.754	0.1	0.759	0.5	0.776	2.8
$(4,1,1)$	-0.434	5.9	-0.447	9.0	-0.483	17.8
$(4,2,1)$	0.314	5.9	0.318	3.9	0.329	7.5
$(4,3,1)$	0.854	0.0	0.859	0.6	0.874	2.3
$(5,1,1)$	-0.609	2.9	-0.621	4.9	-0.655	10.6
$(5,3,1)$	0.532	1.7	0.541	3.4	0.569	8.8
$(5,4,1)$	0.904	0.1	0.909	0.7	0.923	2.2
$(6,1,1)$	-0.716	2.0	-0.726	3.4	-0.757	7.8
$(6,2,1)$	-0.290	7.8	-0.303	12.6	-0.340	26.4
$(6,3,1)$	0.231	5.0	0.237	7.7	0.255	15.9
$(6,4,1)$	0.660	1.1	0.669	2.5	0.694	6.3
$(6,5,1)$	0.933	0.2	0.937	0.6	0.950	2.0

Table 4.6: Non-dimensional frequencies, $\sigma = \omega/2\Omega$, of some low of the order inertial modes of a compressible and unstably stratified fluid spheres with different stability parameter β. Columns next to the frequency columns show the absolute percentage difference between the respective modal frequencies of the specific core model and that of the Poincaré core model.

Mode	$\sigma_{0.001}$	% diff	$\sigma_{0.002}$	% diff	$\sigma_{0.005}$	% diff
$(3,1,0)$	0.449	0.4	0.442	1.1	0.422	5.6
$(4,1,0)$	0.649	0.9	0.640	2.3	0.614	6.3
$(5,1,0)$	0.282	1.1	0.273	4.2	0.243	14.7
$(5,2,0)$	0.758	0.9	0.751	1.8	0.727	5.0
$(6,1,0)$	0.460	1.9	0.449	4.3	0.412	12.2
$(6,2,0)$	0.823	0.8	0.816	1.7	0.794	4.3
$(2,1,1)$	0.500	0.0	0.500	0.0	0.500	0.0
$(3,1,1)$	-0.079	10.2	-	-	-	-
$(3,2,1)$	0.743	1.6	0.737	2.4	0.720	4.6
$(4,1,1)$	-0.408	0.5	-0.394	3.9	-0.350	14.6
$(4,2,1)$	0.306	0.0	0.302	1.3	0.290	5.2
$(4,3,1)$	0.844	1.2	0.839	1.8	0.824	3.5
$(5,1,1)$	-0.586	1.0	-0.574	3.0	-0.536	9.5
$(5,3,1)$	0.512	2.1	0.502	4.0	0.471	9.9
$(5,4,1)$	0.895	0.9	0.890	1.4	0.876	3.0
$(6,1,1)$	-0.703	0.1	-0.685	2.4	-0.652	7.1
$(6,2,1)$	-0.261	3.0	-0.246	8.6	-0.194	27.9
$(6,3,1)$	0.217	1.4	0.210	4.5	0.190	13.6
$(6,4,1)$	0.643	1.5	0.634	2.9	0.607	7.0
$(6,5,1)$	0.924	0.8	0.919	1.3	0.906	2.7

except for (6,2,1) mode, have the smallest middle index number k for the respective n and m. We were not able to compute the frequency of the (3,1,1) mode for $\beta = 0.002$ and $\beta = 0.005$ (see Table 4.6) as the frequency of this mode did not converge.

The non-dimensional frequencies, σ, as a function of stratification parameter, β, for each mode are shown in figures 4.1 and 4.2. We see from these figures that the frequencies of all modes, except the (2,1,1) mode where the frequency is independent of β, linearly depend on the density stratification, but their are not parallel to each other because the slopes of these lines are not same. The frequencies of all modes with positive frequency, except for the (2,1,1) mode, decrease with respect to β. Conversely, the frequencies of all modes with negative frequency increase with respect to β. The different signs of the frequencies indicate that the modes with positive frequency are prograde and negative frequencies are retrograde. The frequencies of the (3,1,0) and (6,1,0) modes around $\beta = 0.0034$, and of the (2,1,1) and (5,3,1) modes around $\beta = 0.0022$ are same, but their structures are different. This frequency crossing is not unusual. Rieutord and Valdetaro [49] reported observing the frequency crossing (having the same frequency) for two inertial modes, resembling the (4,1,0) mode, of an incompressible but viscous fluid shell in the limit as viscosity tends to zero.

4.2 Eigenfunctions of the Spherical Core Models

As mentioned above, the displacement eigenfunctions of the modes must be regular and similar to those of the Poincaré model. Here we represent some of the eigenfunctions corresponding to the frequencies shown in tables 4.4, 4.5, and 4.6 computed using the FORTRAN program (see Appendix C.3). The displacement eigenfunctions and the compressed potential eigenfunctions, $\Phi = \chi - V_1$, contours were plotted using the engineering software package TecPlot 9.0 [Ref: software company AMTEC ENGINEERING,INC.]. Figures 4.3, 4.4, 4.5 and 4.6 show the meridional displacement vectors and compressed potential contours of the (4,1,0), (5,2,0), (6,1,0) and (6,2,0) modes for the modified spherical

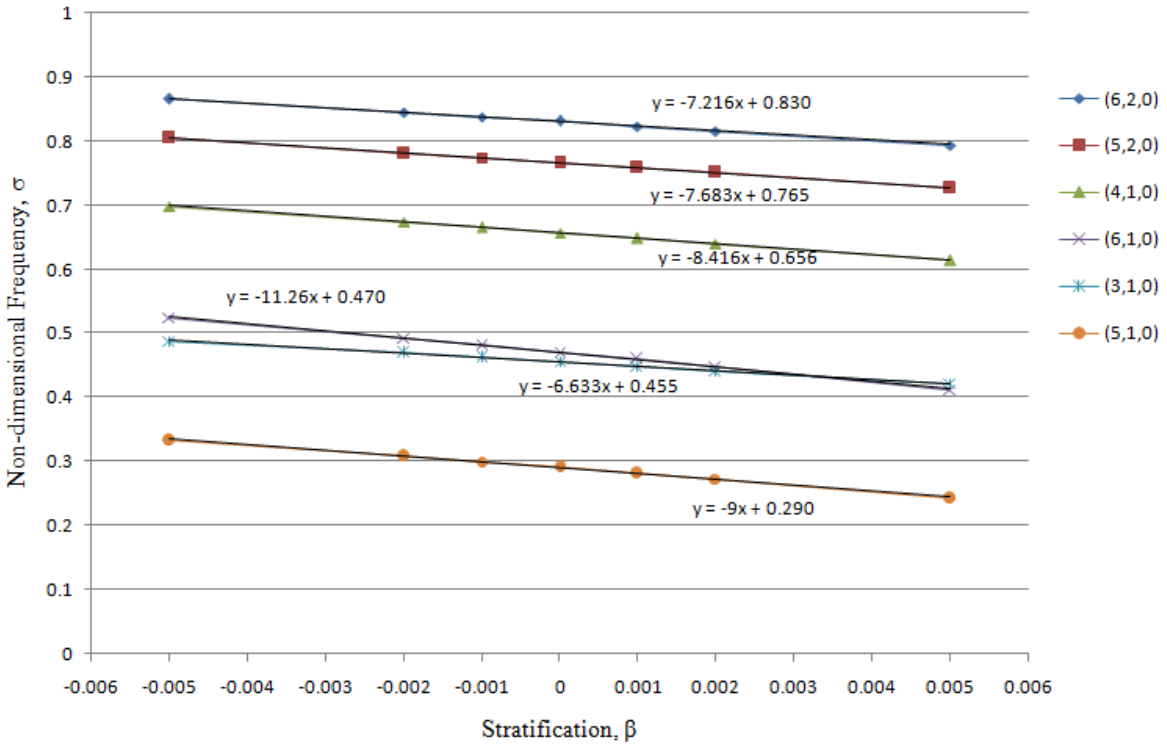

Figure 4.1: The non-dimensional frequency, σ, as a function of stratification parameter, β, for the axisymmetric (i.e., $m = 0$) modes of a compressible and stratified fluid spheres.

core models with different stability parameter, and with characteristic lines, respectively. The structure of the compressed potential eigenfunction is divided into its various cells by several color contours. In addition, red contours indicate the cells where the compressed potential is higher and blue contours indicate the cells where the compressed potential is lower. In the contour plots, a characteristic line is crossed where an abrupt change in the slope or trend of the contours occurs. Moreover, a vector plot in the meridional plane shows the relative magnitude of the radial and vertical components of the displacement. In the vector plots, the characteristic lines for axisymmetric modes pass through the center of the displacement cells. Thus a characteristic line indicates the location of the cell. In all figures (Figures 4.3- 4.14), the displacement is parallel to the boundary as prescribed by the boundary condition. We see from these figures that a characteristic line is slightly shifted counterclockwise from a neutrally stratified core model to stably stratified core models, but a characteristic line is slightly shifted clockwise from a neutrally stratified core model to

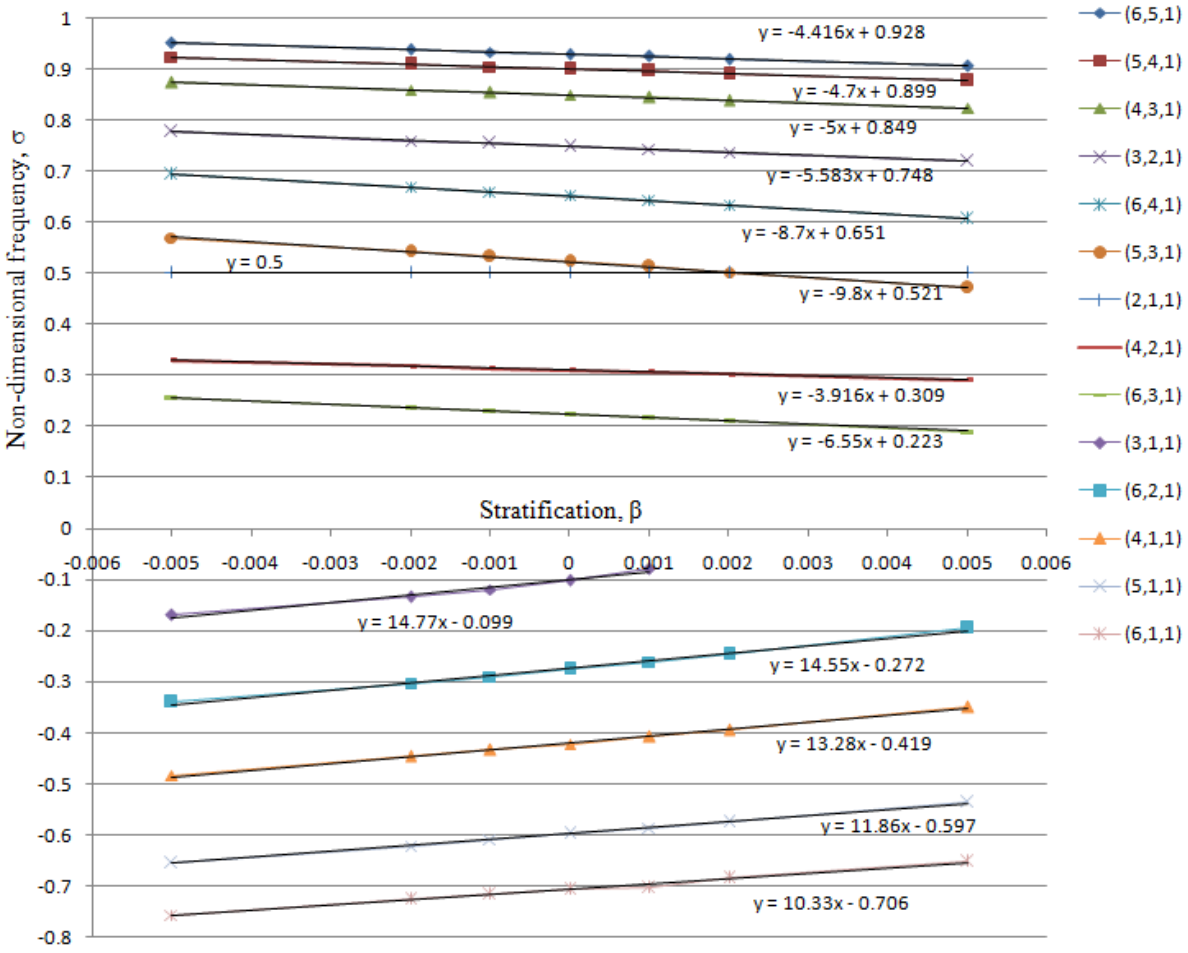

Figure 4.2: The non-dimensional frequency, σ, as a function of stratification parameter, β, for the non- axisymmetric modes with the azimuthal wavenumber, $m = 1$ of a compressible and stratified fluid spheres.

unstably stratified core models. Thus, the locations of the cells are affected by stratification so that the structures of eigenfunctions are affected by the stratification.

Figures 4.7 - 4.14 show the meridional displacement vectors and the compressed potential, $\Phi = \chi - V_1$, contours of some non-axisymmetric modes with azimuthal wavenumber 1 of the modified spherical core models with different stability parameter. In these figures, we were not able to figure out the relation between the characteristic lines and displacement vectors, because the azimuthal variable ϕ is separable from the other two variables (r, θ) to lead that the characteristics (i.e., Eq. (2.59)) are independent of m, however, the characteristics exist in the meridional plane of non-axisymmetric modes but are modulated

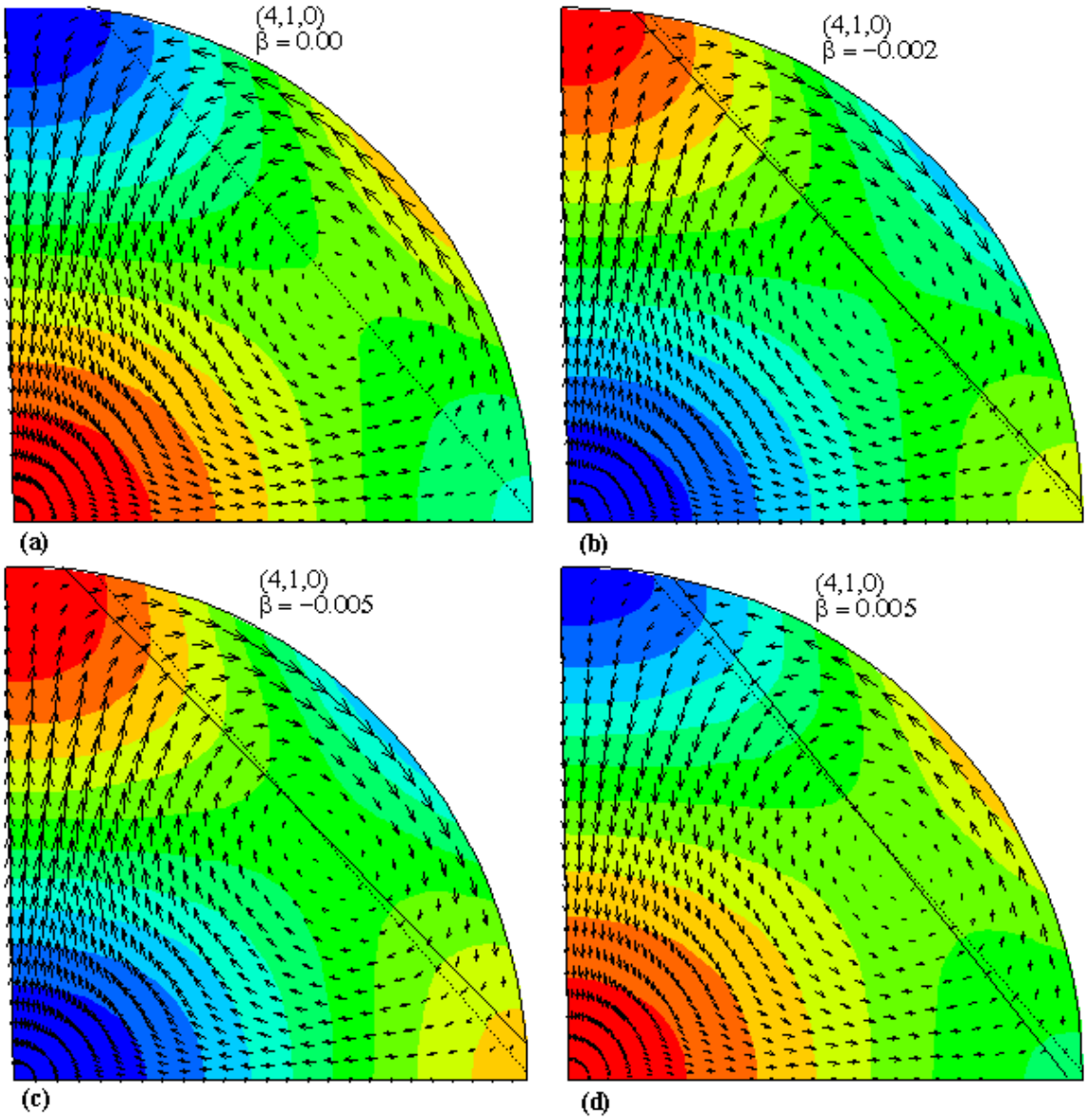

Figure 4.3: The meridional displacement vectors and the compressed potential, $\Phi = \chi - V_1$, contours of the (4,1,0) mode of the modified spherical core models with different stability parameter: (a) $\beta = 0.00$, (b) $\beta = -0.002$, (c) $\beta = -0.005$ and (d) $\beta = 0.005$, and a special characteristic line: dotted lines for $\beta = 0.00$ and solid lines for respective core models for different β. Note that the horizontal axis is the equator and the vertical axis is the rotational axis.

by $e^{im\phi}$ [44]. Our results show that the (2,1,1) mode is least affected by stratification. Figure 4.7 shows that the flow is divergence free. For this case equation (2.34) reduces to the Poincaré equation, therefore, we expect that the frequencies of this mode remain the same for all core models. We see from figures 4.8 - 4.14 that the modal cells for the displacement

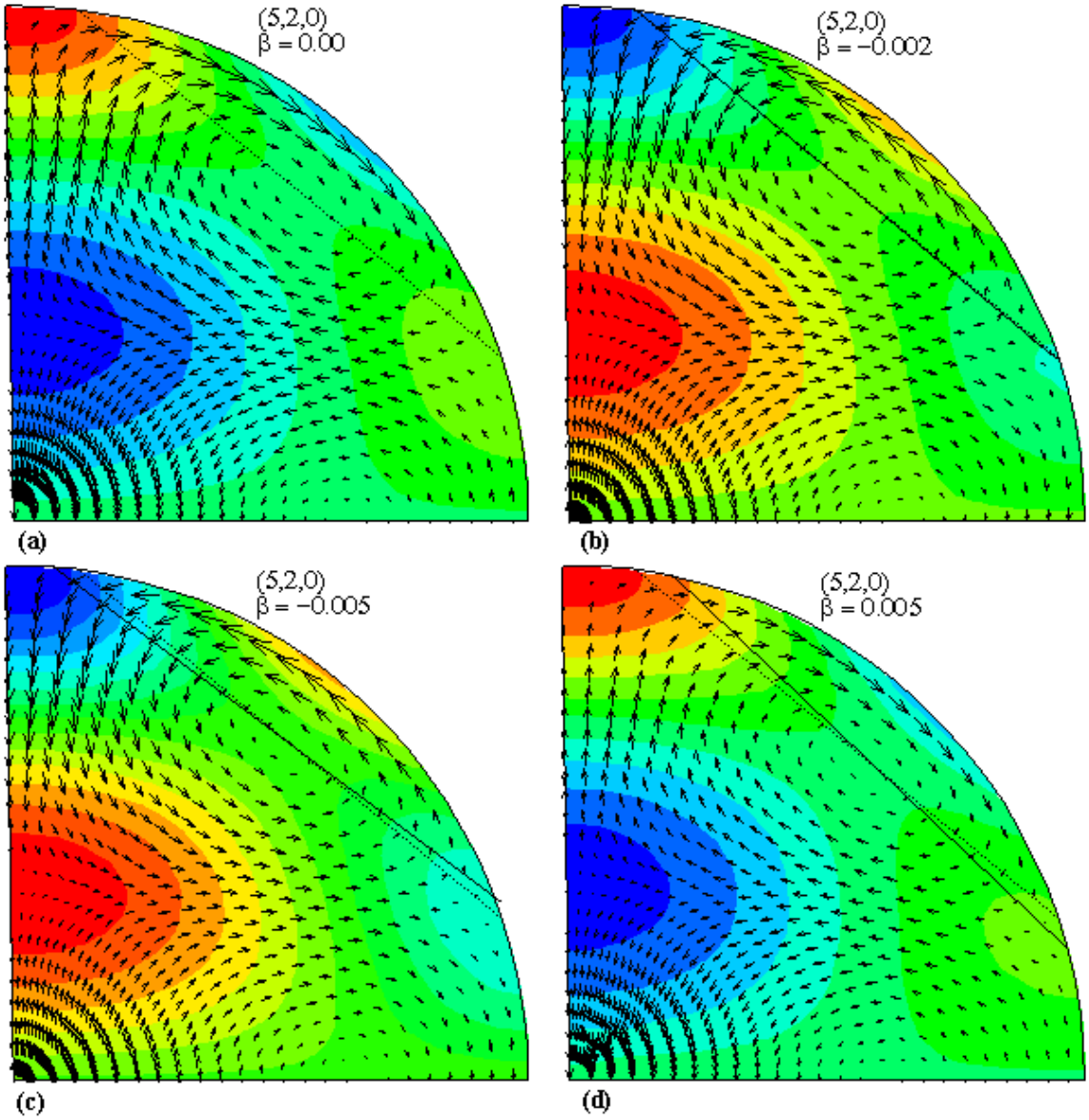

Figure 4.4: The meridional displacement vectors and the compressed potential, $\Phi = \chi - V_1$, contours of the (5,2,0) mode of the modified spherical core models with different stability parameter: (a) $\beta = 0.00$, (b) $\beta = -0.002$, (c) $\beta = -0.005$ and (d) $\beta = 0.005$, and a special characteristic line: dotted lines for $\beta = 0.00$ and solid lines for respective core models for different β. Note that the horizontal axis is the equator and the vertical axis is the rotational axis.

of (3,2,1), (4,3,1), (5,4,1) and (6,5,1) modes, least affected by stratification, are formed predominantly perpendicular to the rotation axis, while the modal cells for the (4,1,1), (5,1,1) and (6,2,1) modes, most affected by stratification, are formed predominantly parallel to the rotation axis. In equation (2.34), the term $\beta\vec{C}^*\zeta$ has a strong component, $\hat{e}_3 \cdot \vec{g}_0\hat{e}_3$ which is

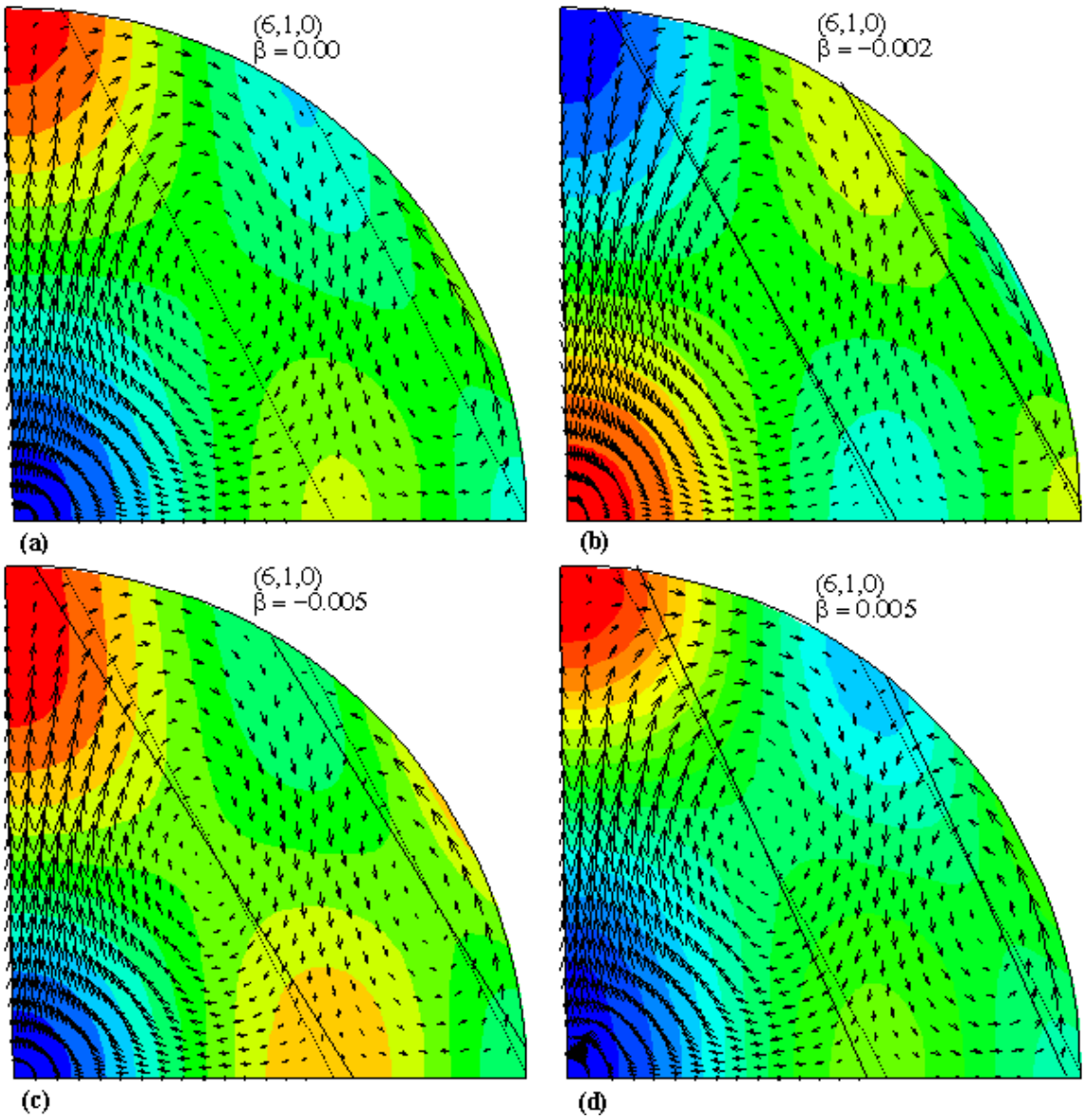

Figure 4.5: The meridional displacement vectors and the compressed potential, $\Phi = \chi - V_1$ contours of the (6,1,0) mode of the modified spherical core models with different stability parameter: (a) $\beta = 0.00$, (b) $\beta = -0.002$, (c) $\beta = -0.005$ and (d) $\beta = 0.005$, and the special characteristic lines: dotted lines for $\beta = 0.00$ and solid lines for respective core models for different β. Note that the horizontal axis is the equator and the vertical axis is the rotational axis.

parallel to the rotation axis. It seems that the stronger the parallel components of the displacement vectors to the rotation axis, the stronger the effects of the stratification is on the frequency of the mode. Moreover, in these figures, the structure of the compressed potential eigenfunction is divided into its various cells by several color contours. The structure

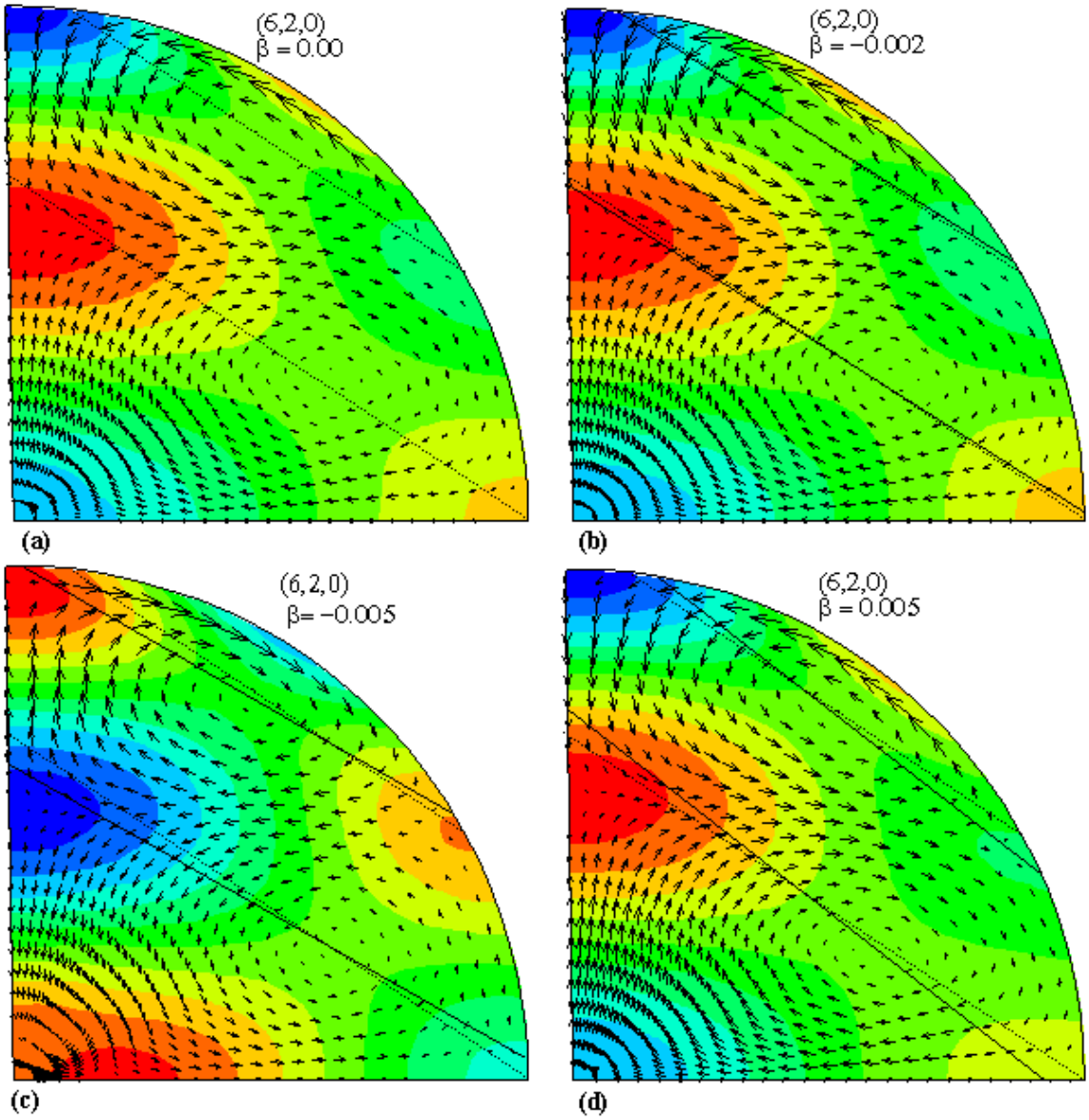

Figure 4.6: The meridional displacement vectors and the compressed potential, $\Phi = \chi - V_1$, contours of the (6,2,0) mode of the modified spherical core models with different stability parameter: (a) $\beta = 0.00$, (b) $\beta = -0.002$, (c) $\beta = -0.005$ and (d) $\beta = 0.005$, and the special characteristic lines: dotted lines for $\beta = 0.00$ and solid lines for respective core models for different β. Note that the horizontal axis is the equator and the vertical axis is the rotational axis.

and magnitude of the contours plots (i.e., Figure 4.7) of the (2,1,1) mode are same for different core models. On the other hand, we see from figures 4.8 - 4.14 that the structure and magnitude of the contour plots are significantly changed by the density stratification.

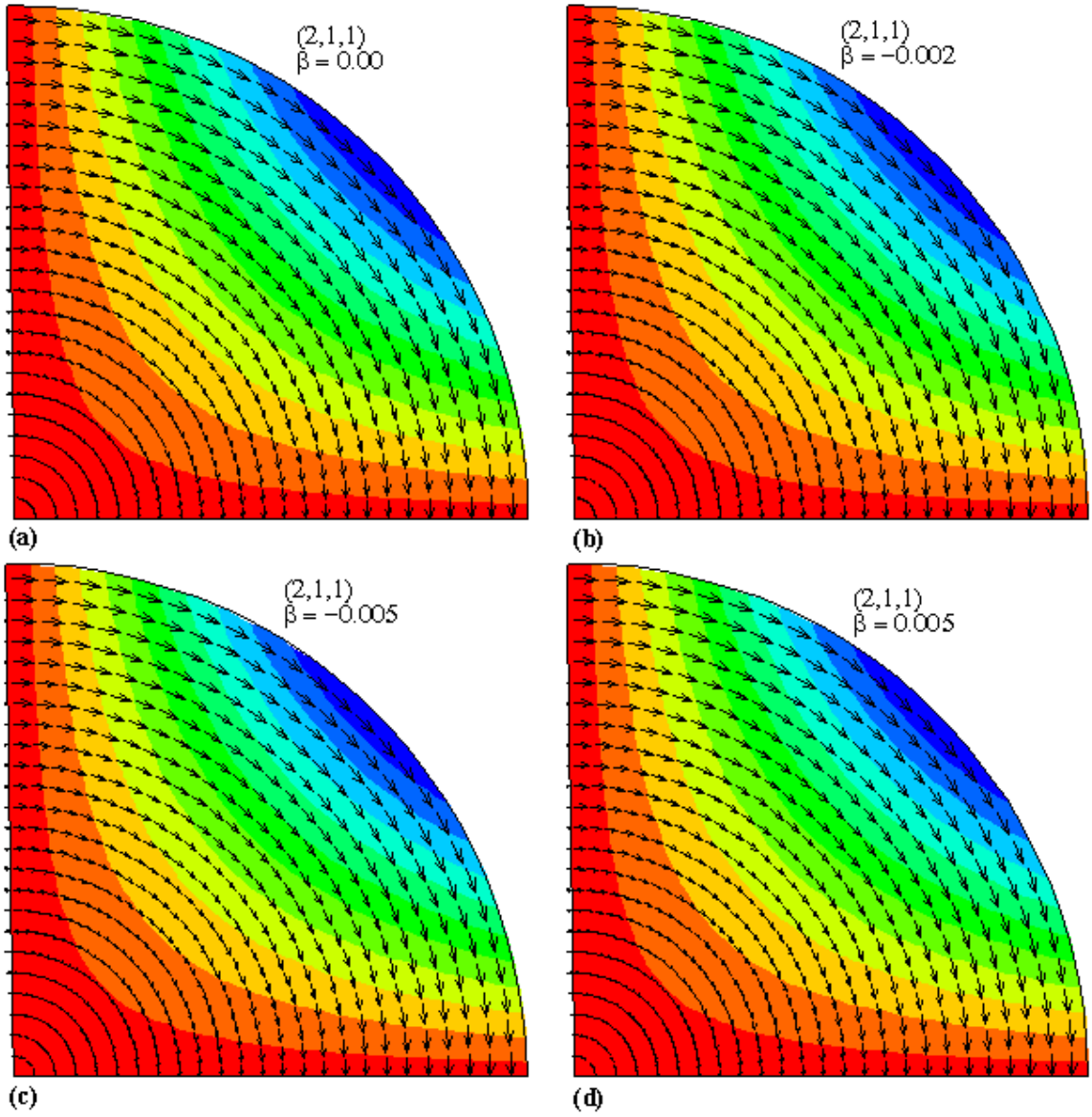

Figure 4.7: The meridional displacement vectors and the compressed potential, $\Phi = \chi - V_1$, contours of the (2,1,1) mode of the modified spherical core models with different stability parameter: (a) $\beta = 0.00$, (b) $\beta = -0.002$, (c) $\beta = -0.005$ and (d) $\beta = 0.005$. Note that the horizontal axis is the equator and the vertical axis is the rotational axis.

4.3 Non-dimensional Frequencies of the Spherical Shell Models

The computational approach for finding the inertial modes of a shell is the same as that for a sphere. In Table 4.7 we show the non-dimensional frequencies of some of the low order inertial modes of a compressible and neutrally stratified fluid, and stably stratified fluid shell proportional to the Earth's core with different stability parameter β. In this

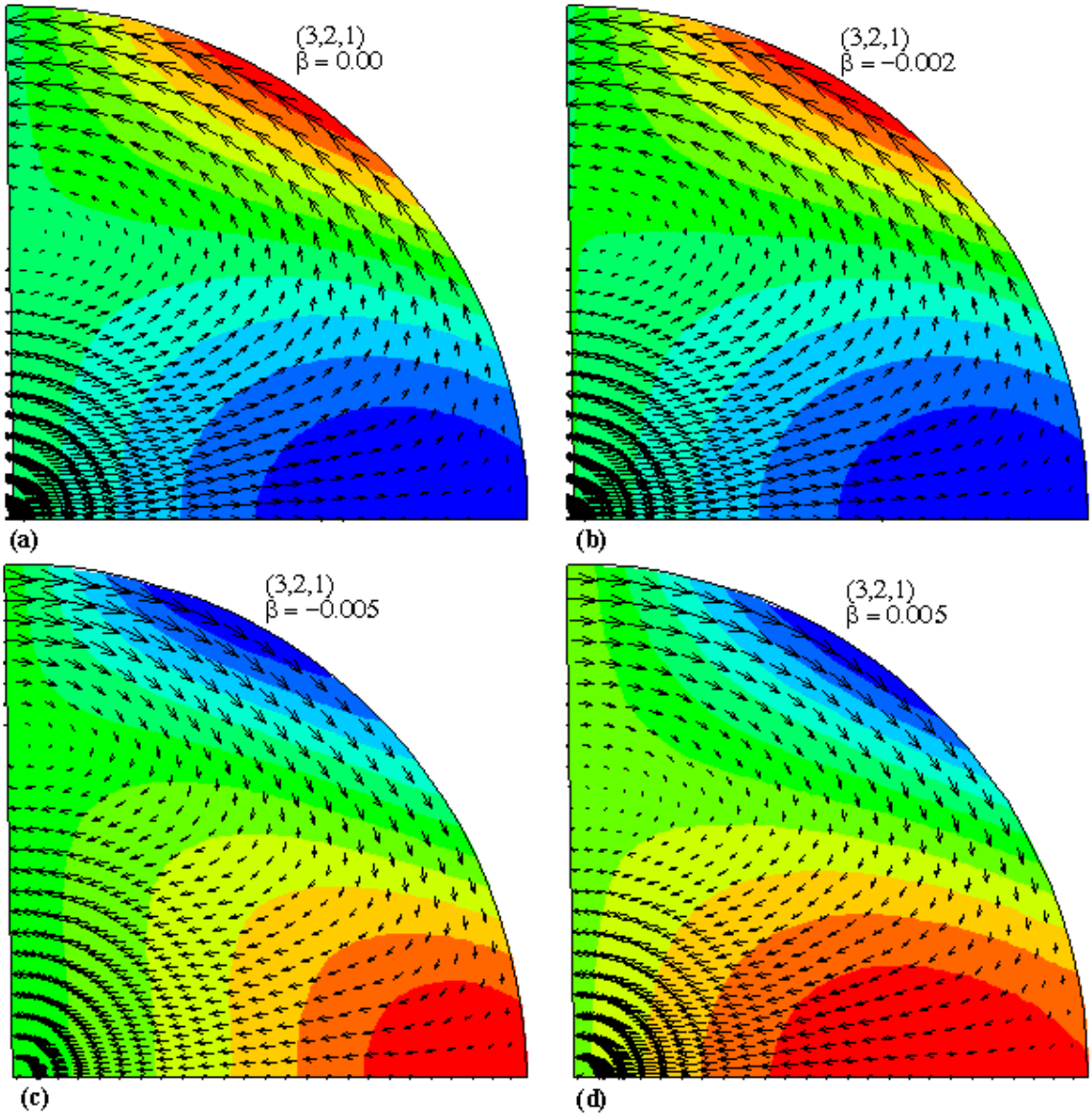

Figure 4.8: The meridional displacement vectors and the compressed potential, $\Phi = \chi - V_1$, contours of the (3,2,1) mode of the modified spherical core models with different stability parameter: (a) $\beta = 0.00$, (b) $\beta = -0.002$, (c) $\beta = -0.005$ and (d) $\beta = 0.005$. Note that the horizontal axis is the equator and the vertical axis is the rotational axis.

table, column 2 shows the frequencies of the inertial modes of a compressible and neutrally stratified spherical shell proportional to the Earth's fluid core. Note that our results (column 2) agree well with those of Seyed-Mahmoud et al. [6] (see their table 2). In columns 3, 4 and 5 of table 4.7 we show the modal frequencies of a compressible and stably stratified fluid spherical shell models for $\beta = -0.001$, $\beta = -0.002$ and $\beta = -0.005$, respectively. We

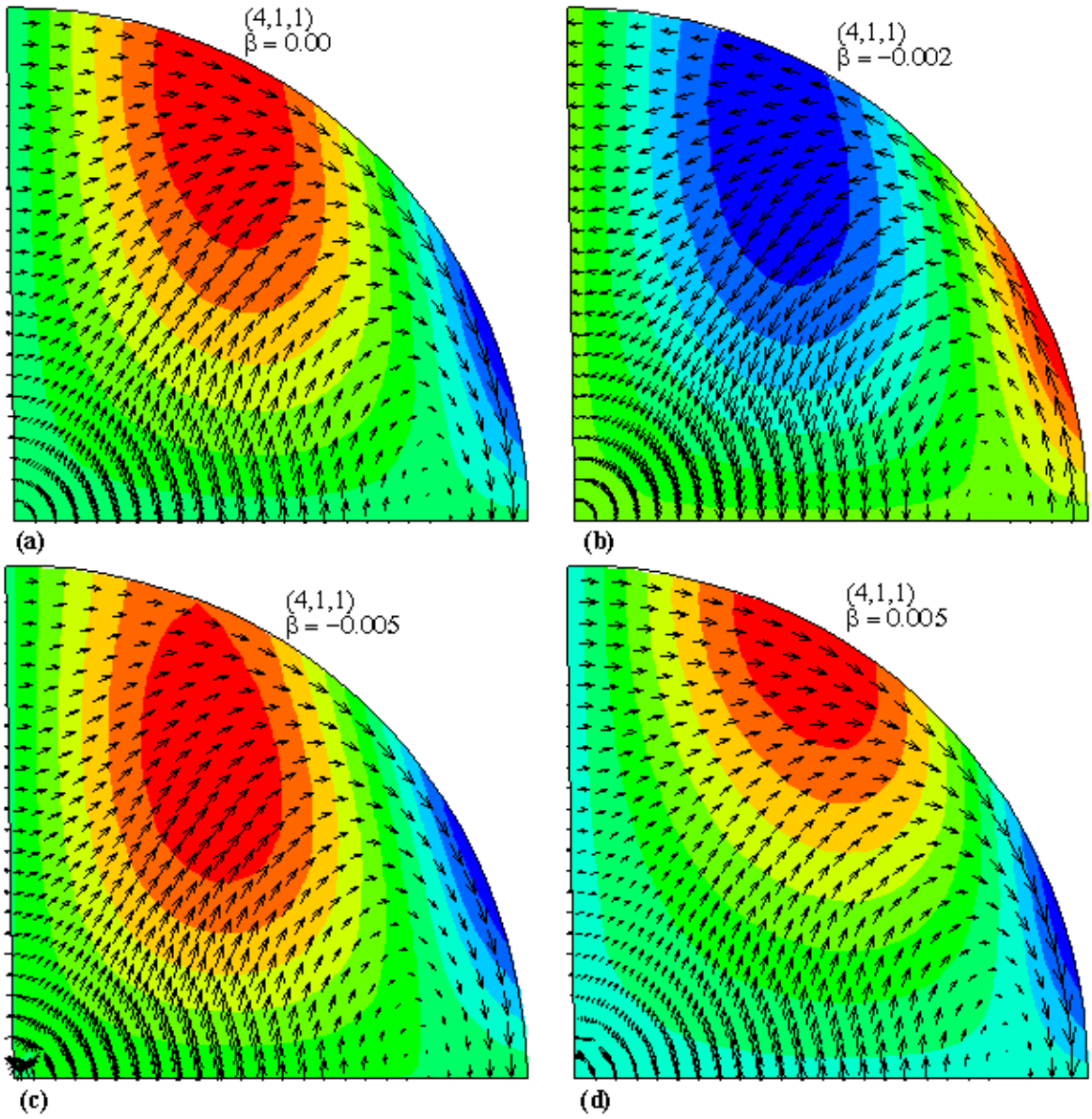

Figure 4.9: The meridional displacement vectors and the compressed potential, $\Phi = \chi - V_1$, contours of the (4,1,1) mode of the modified spherical core models with different stability parameter: (a) $\beta = 0.00$, (b) $\beta = -0.002$, (c) $\beta = -0.005$ and (d) $\beta = 0.005$. Note that the horizontal axis is the equator and the vertical axis is the rotational axis.

were not able to compute the frequencies of the (6,1,0) and (4,2,1) modes for $\beta = 0.005$ (see Table 4.7) as the frequencies of these modes did not converge.

Figures 4.15 and 4.16 show the non-dimensional frequency, σ, as a function of stratification parameter, β, for each mode. From these figures we see that the frequencies of all modes with positive frequency, except for the (2,1,1) mode, linearly decrease with respect

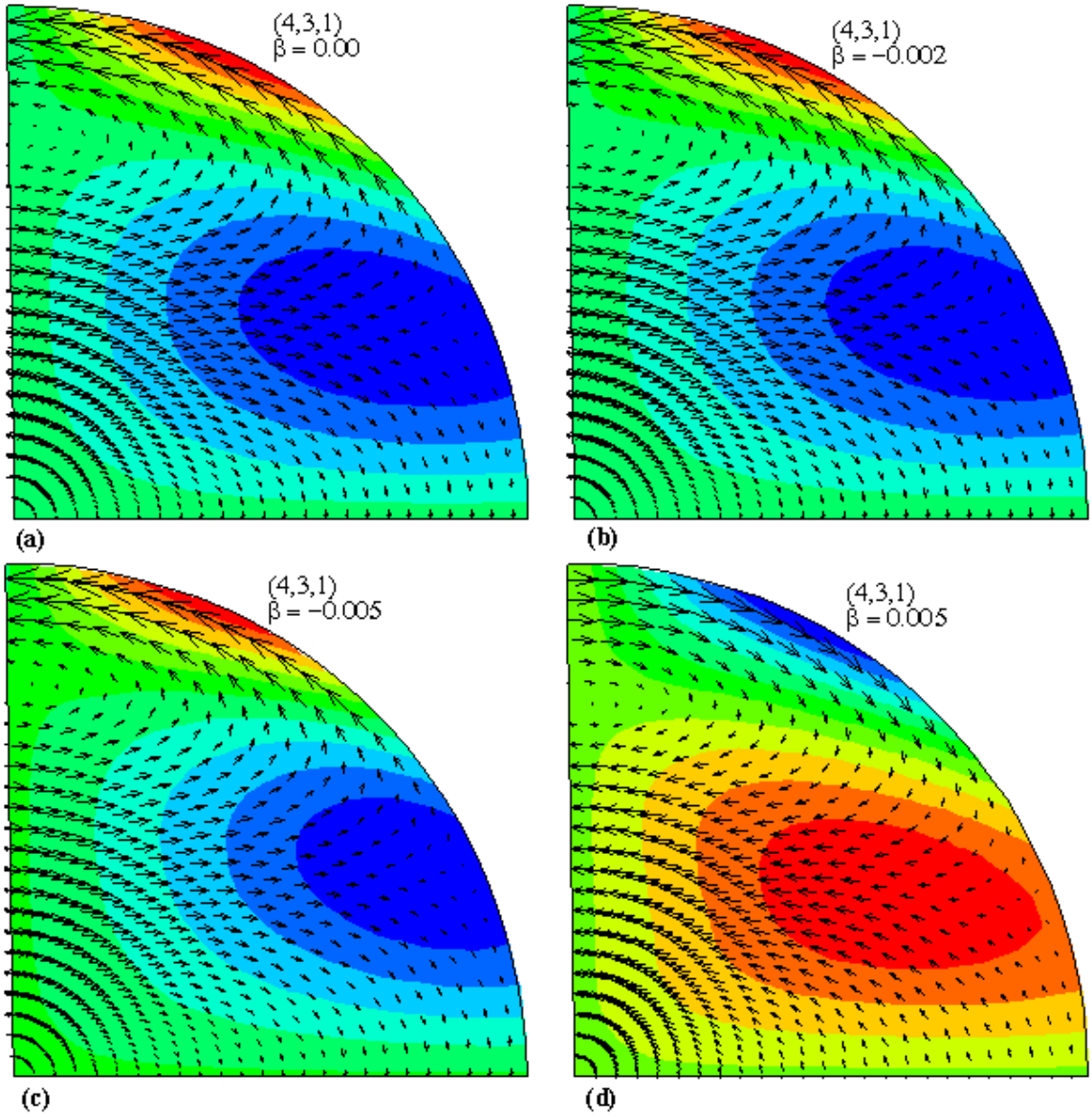

Figure 4.10: The meridional displacement vectors and the compressed potential, $\Phi = \chi - V_1$, contours of the (4,3,1) mode of the modified spherical core models with different stability parameter: (a) $\beta = 0.00$, (b) $\beta = -0.002$, (c) $\beta = -0.005$ and (d) $\beta = 0.005$. Note that the horizontal axis is the equator and the vertical axis is the rotational axis.

to β same as in the case of a sphere. On the other hand, the frequency of the (6,1,1) mode linearly increases with β, where the frequency of that mode is negative. The slopes of these lines are different, so they are not parallel to each other.

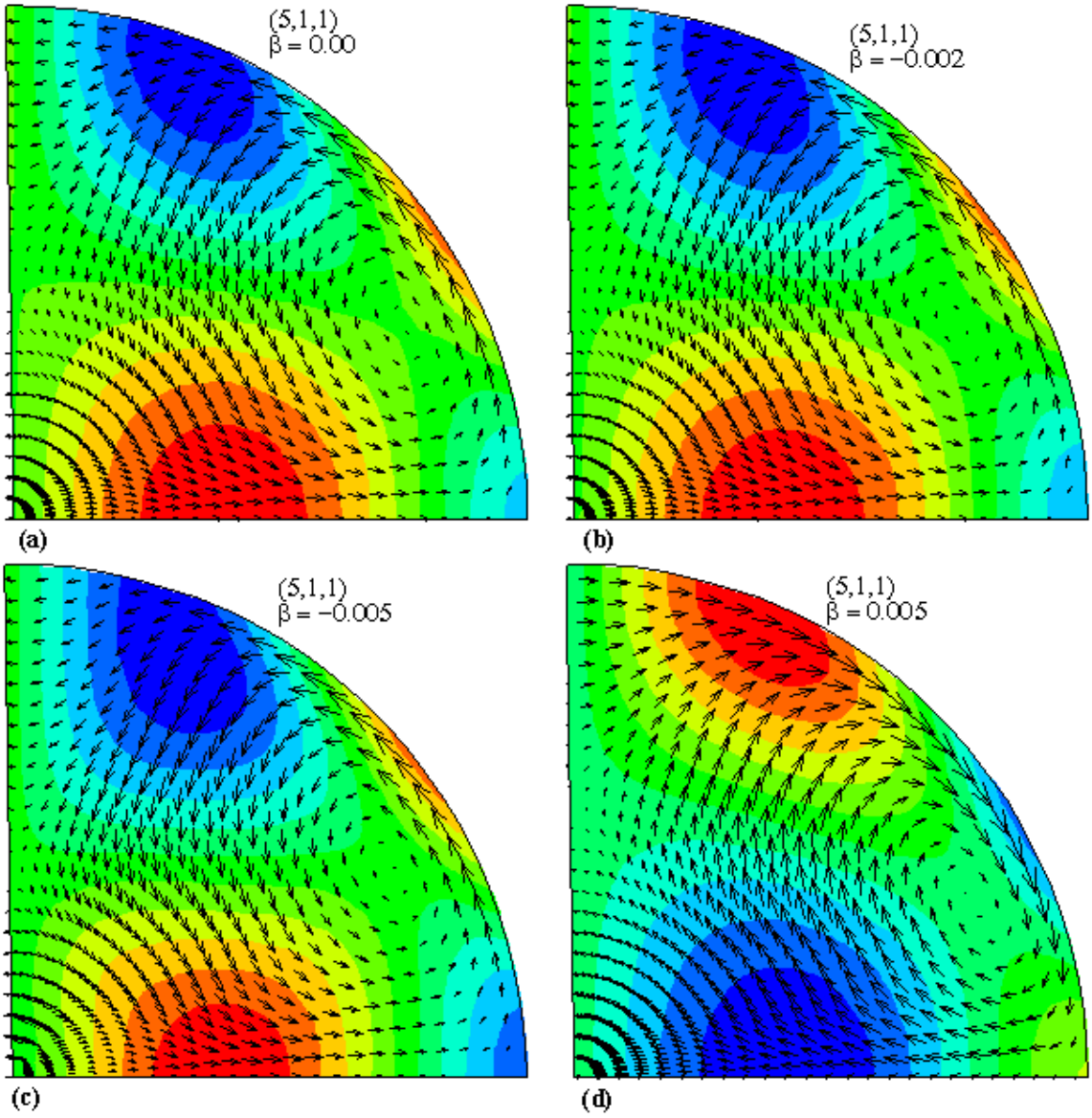

Figure 4.11: The meridional displacement vectors and the compressed potential, $\Phi = \chi - V_1$, contours of the (5,1,1) mode of the modified spherical core models with different stability parameter: (a) $\beta = 0.00$, (b) $\beta = -0.002$, (c) $\beta = -0.005$ and (d) $\beta = 0.005$. Note that the horizontal axis is the equator and the vertical axis is the rotational axis.

4.4 Eigenfunctions of the Spherical Shell Models

We expect that the displacement vector is parallel to near the boundaries in a fluid shell. Figures 4.17 and 4.18 show the meridional displacement vectors and compressed potential contours of the (4,1,0) and (6,1,0) modes with a characteristic line superimposed for the modified spherical shell models with different stability parameter, respectively. From these

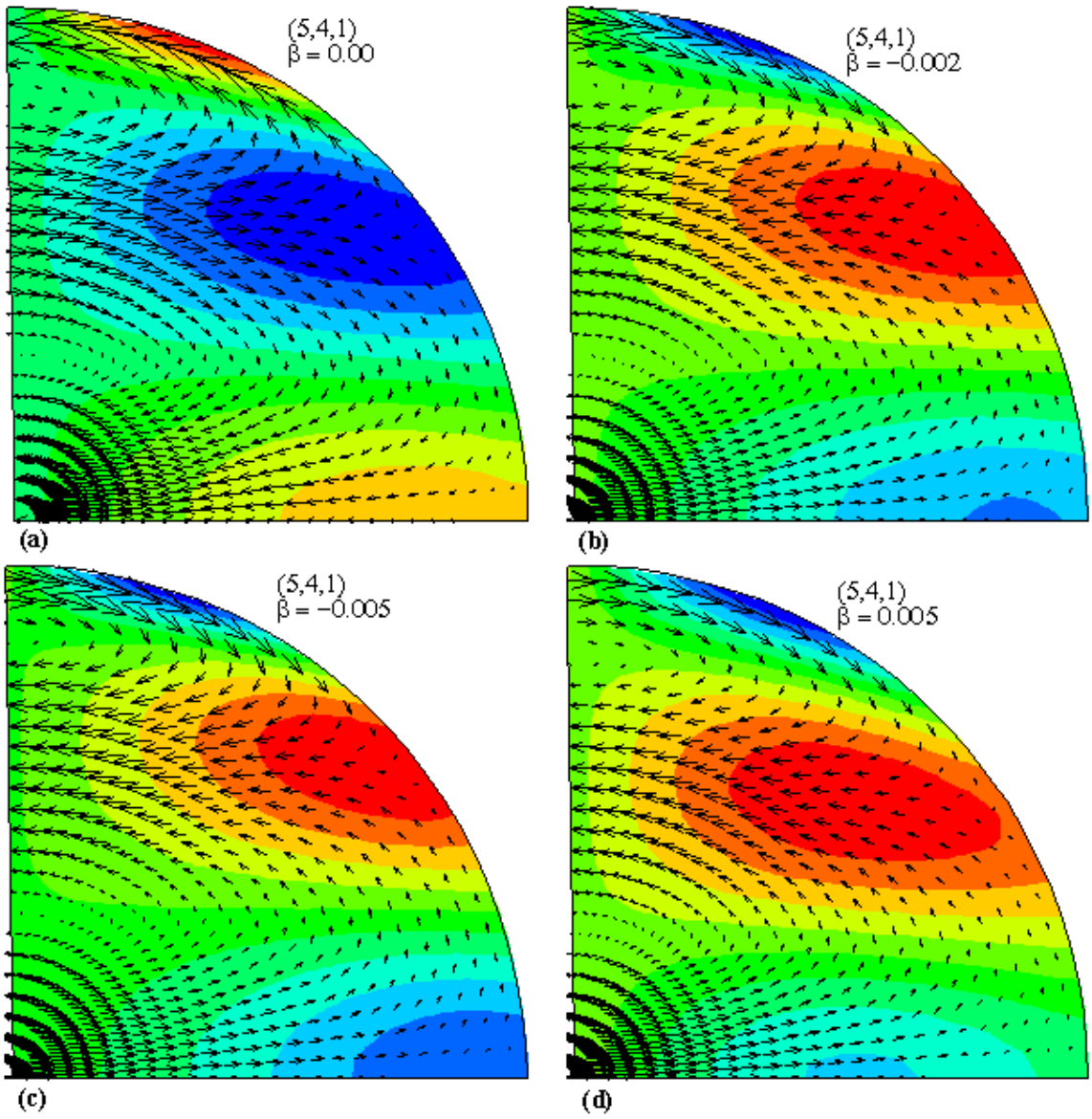

Figure 4.12: The meridional displacement vectors and the compressed potential, $\Phi = \chi - V_1$, contours of the (5,4,1) mode of the modified spherical core models with different stability parameter: (a) $\beta = 0.00$, (b) $\beta = -0.002$, (c) $\beta = -0.005$ and (d) $\beta = 0.005$. Note that the horizontal axis is the equator and the vertical axis is the rotational axis.

figures we deduce that the displacement vectors are parallel to the boundaries and the magnitude of the compressed potential contours are significantly affected by stratification. In these figures, we see that a characteristic line is slightly shifted counterclockwise with the change in the value of β same as in the case of a sphere. Thus, the locations of the cells are affected by stratification as well as the spherical core models.

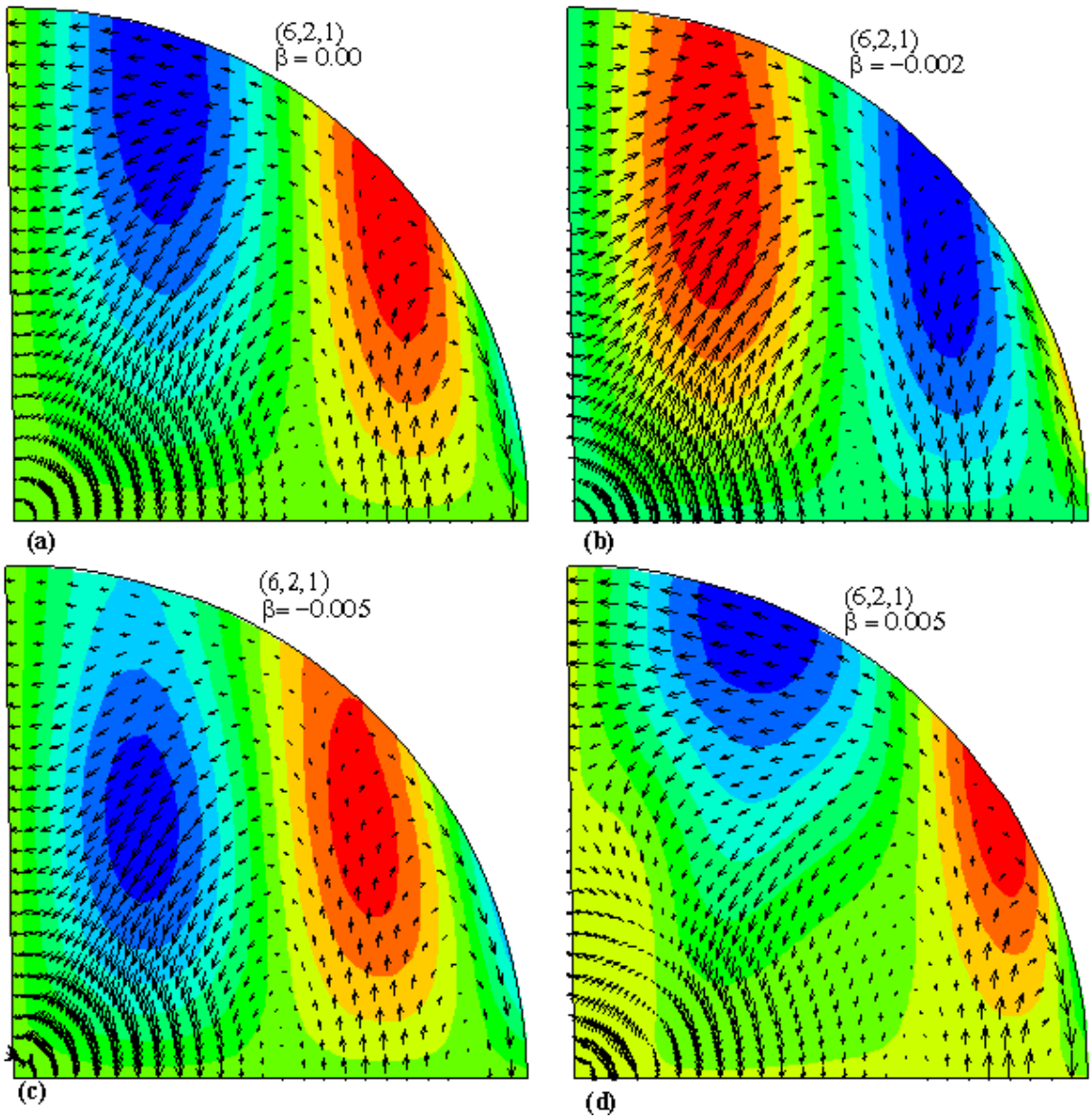

Figure 4.13: The meridional displacement vectors and the compressed potential, $\Phi = \chi - V_1$, contours of the (6,2,1) mode of the modified spherical core models with different stability parameter: (a) $\beta = 0.00$, (b) $\beta = -0.002$, (c) $\beta = -0.005$ and (d) $\beta = 0.005$. Note that the horizontal axis is the equator and the vertical axis is the rotational axis.

Moreover, the displacement eigenfunctions and the compressed potential contours of the modes with azimuthal wavenumber 1 for a neutrally ($\beta = 0.00$) stratified shell model, and their counterparts for stably stratified shell models are shown in Figures 4.19 and 4.20. In these figures, except the (2,1,1) mode, the structure and magnitude of contours plots are significantly affected by the changing in value of β.

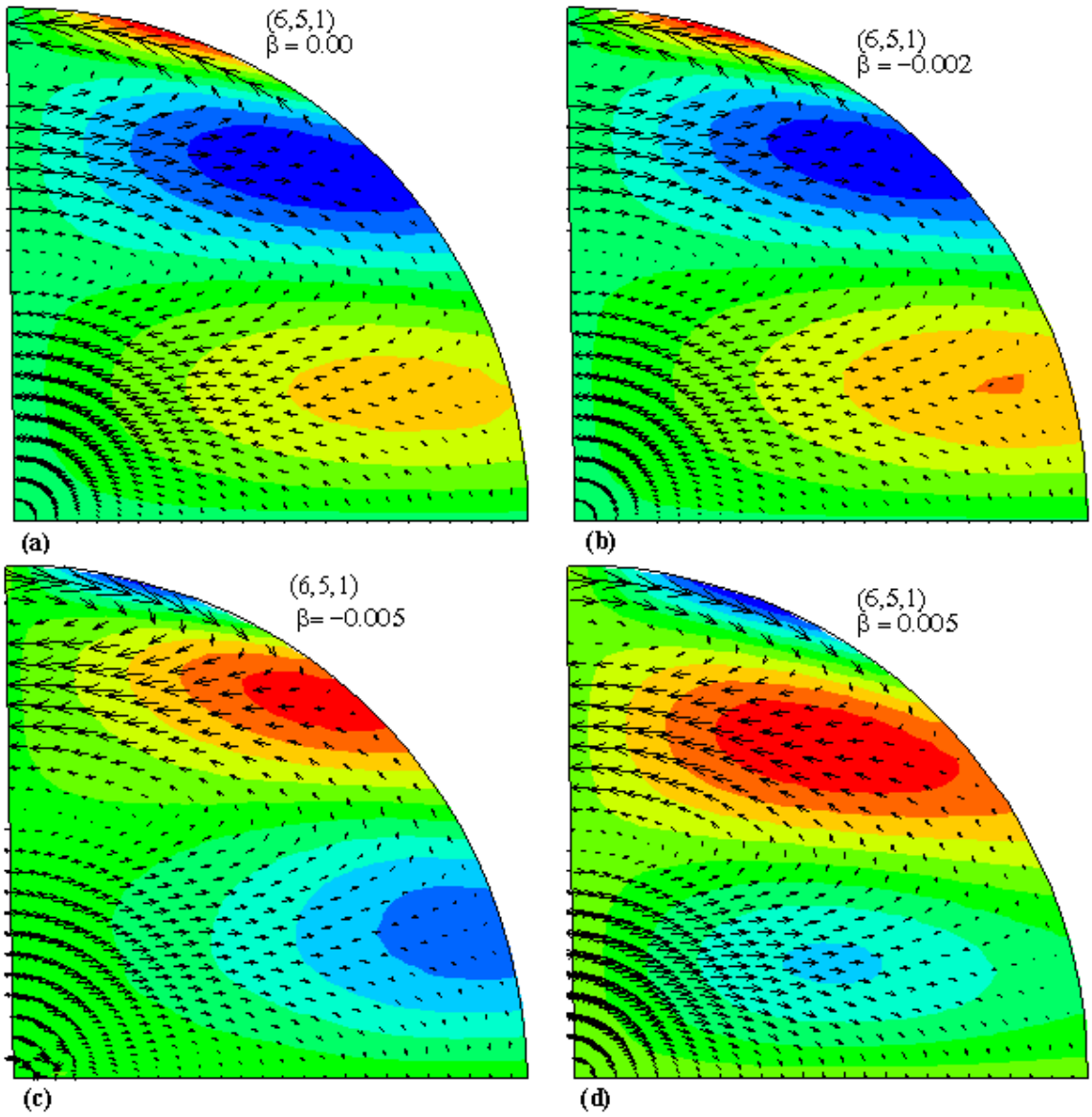

Figure 4.14: The meridional displacement vectors and the compressed potential, $\Phi = \chi - V_1$, contours of the (6,5,1) mode of the modified spherical core models with different stability parameter: (a) $\beta = 0.00$, (b) $\beta = -0.002$, (c) $\beta = -0.005$ and (d) $\beta = 0.005$. Note that the horizontal axis is the equator and the vertical axis is the rotational axis.

Table 4.7: Non-dimensional frequencies, $\sigma = \omega/2\Omega$, for some of the low order inertial modes of a compressible and neutrally stratified fluid, and those stably stratified fluid spherical shell models with different stability parameter β.

Mode	$\sigma_{0.00}$	$\sigma_{-0.001}$	$\sigma_{-0.002}$	$\sigma_{-0.005}$
$(4,1,0)$	0.664	0.673	0.684	0.701
$(5,2,0)$	0.766	0.772	0.781	0.800
$(6,1,0)$	0.475	0.489	0.504	-
$(6,2,0)$	0.833	0.842	0.851	0.877
$(2,1,1)$	0.500	0.500	0.500	0.500
$(4,2,1)$	0.304	0.303	0.306	-
$(4,3,1)$	0.849	0.846	0.850	0.866
$(5,4,1)$	0.930	0.931	0.942	0.947
$(6,1,1)$	-0.700	-0.709	-0.719	-0.756
$(6,4,1)$	0.657	0.658	0.669	0.682
$(6,5,1)$	0.924	0.923	0.930	0.949

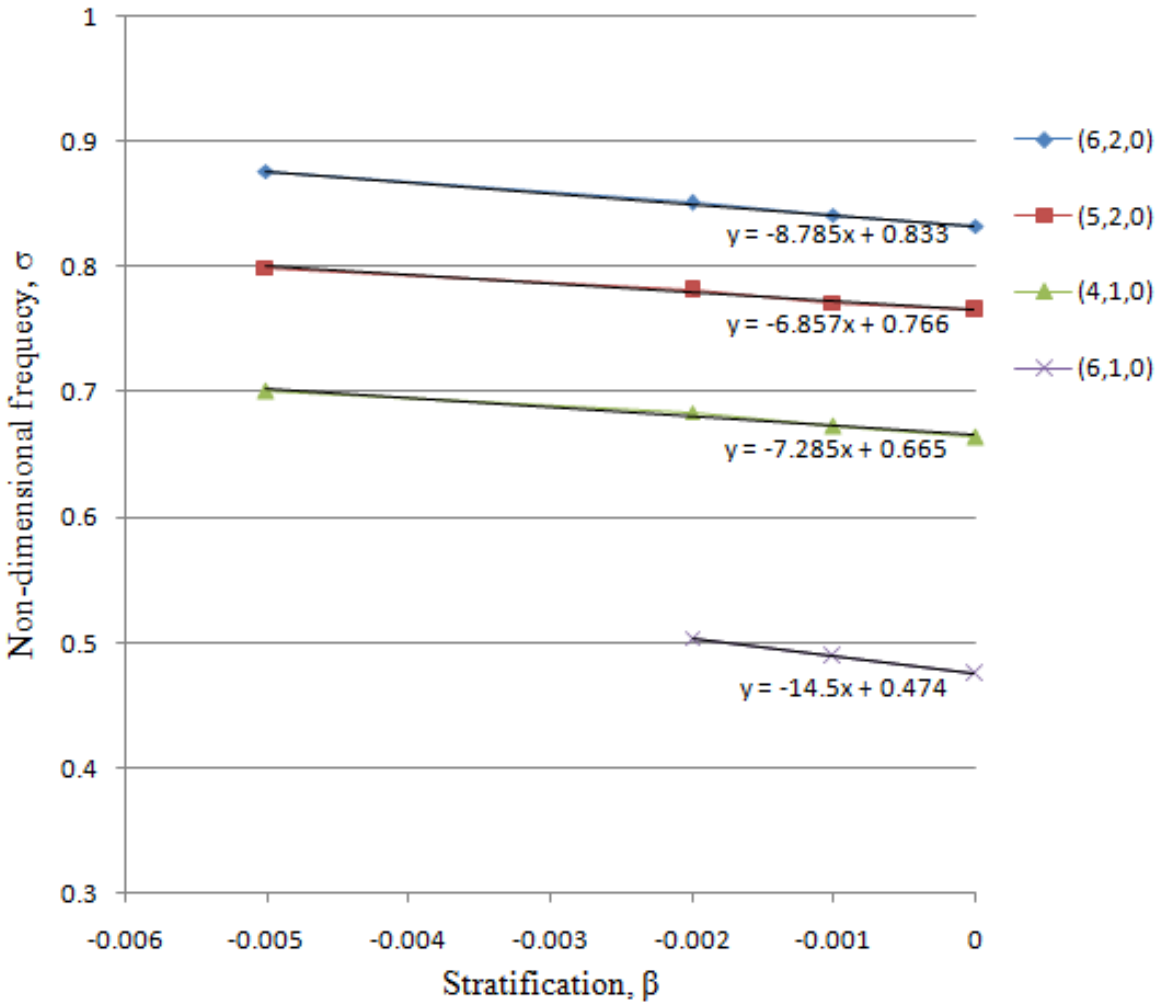

Figure 4.15: The non-dimensional frequency, σ, as a function of stratification parameter, β, for the axisymmetric (i.e., $m = 0$) modes of the spherical shell models.

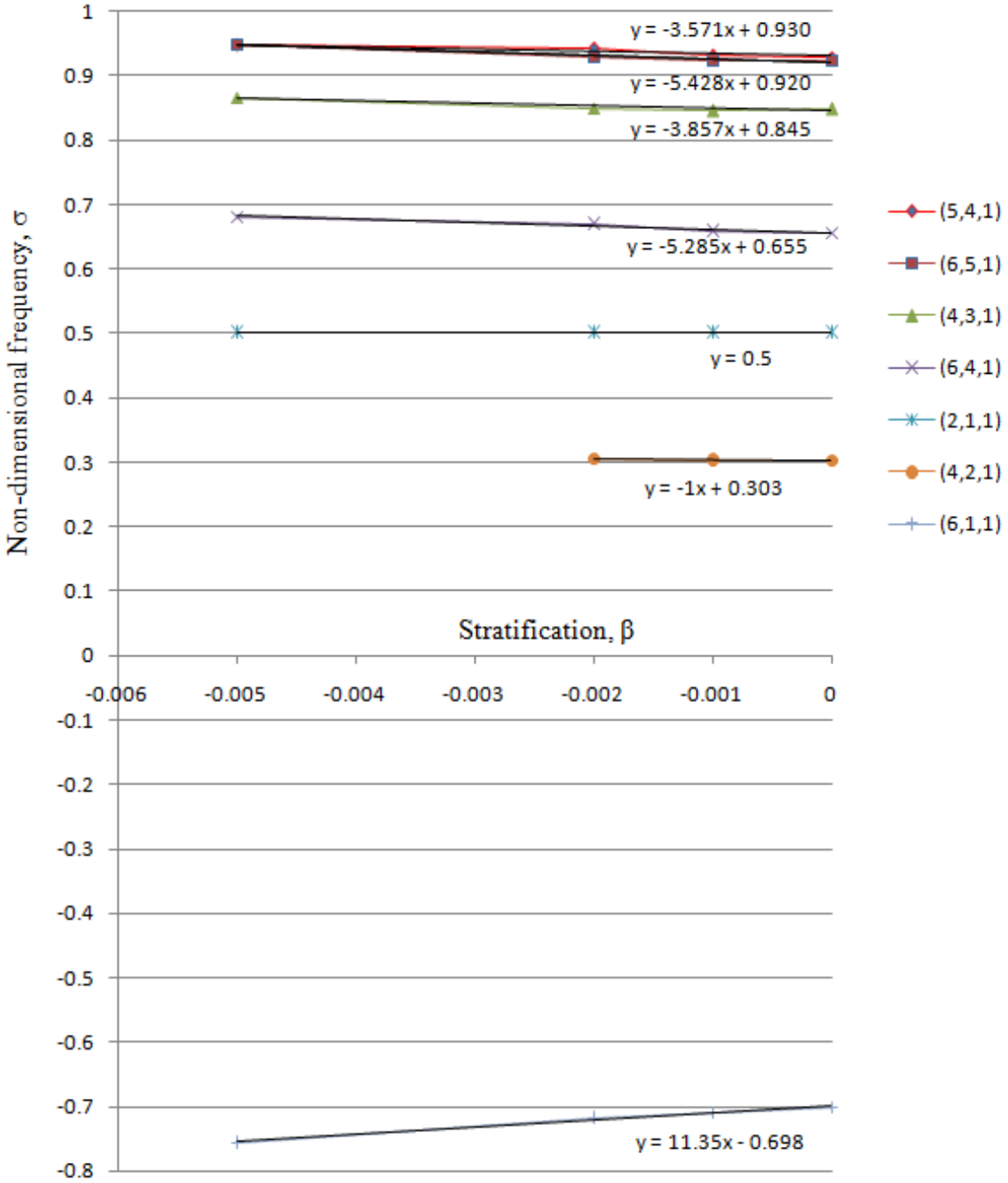

Figure 4.16: The non-dimensional frequency, σ, as a function of stratification parameter, β, for the non- axisymmetric modes with the azimuthal wavenumber, $m = 1$ of the spherical shell models.

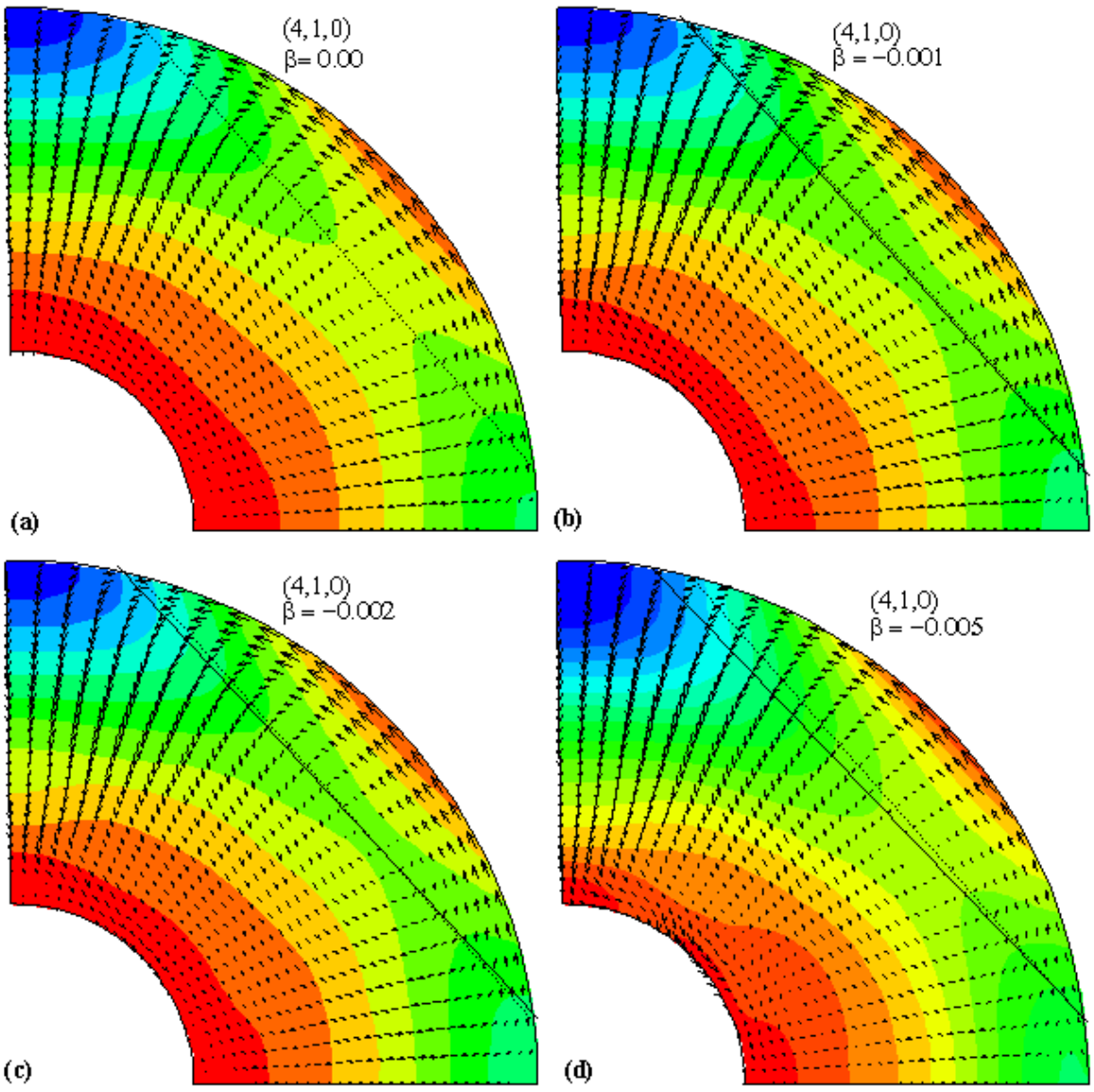

Figure 4.17: The meridional displacement vectors and the compressed potential, $\Phi = \chi - V_1$, contours of the (4,1,0) mode of the modified spherical shell core models with different stability parameter: (a) $\beta = 0.00$, (b) $\beta = -0.001$, (c) $\beta = -0.002$ and (d) $\beta = -0.005$, and a special characteristic line: dotted lines for $\beta = 0.00$ and solid lines for respective core models for different β. Note that the horizontal axis is the equator and the vertical axis is the rotational axis.

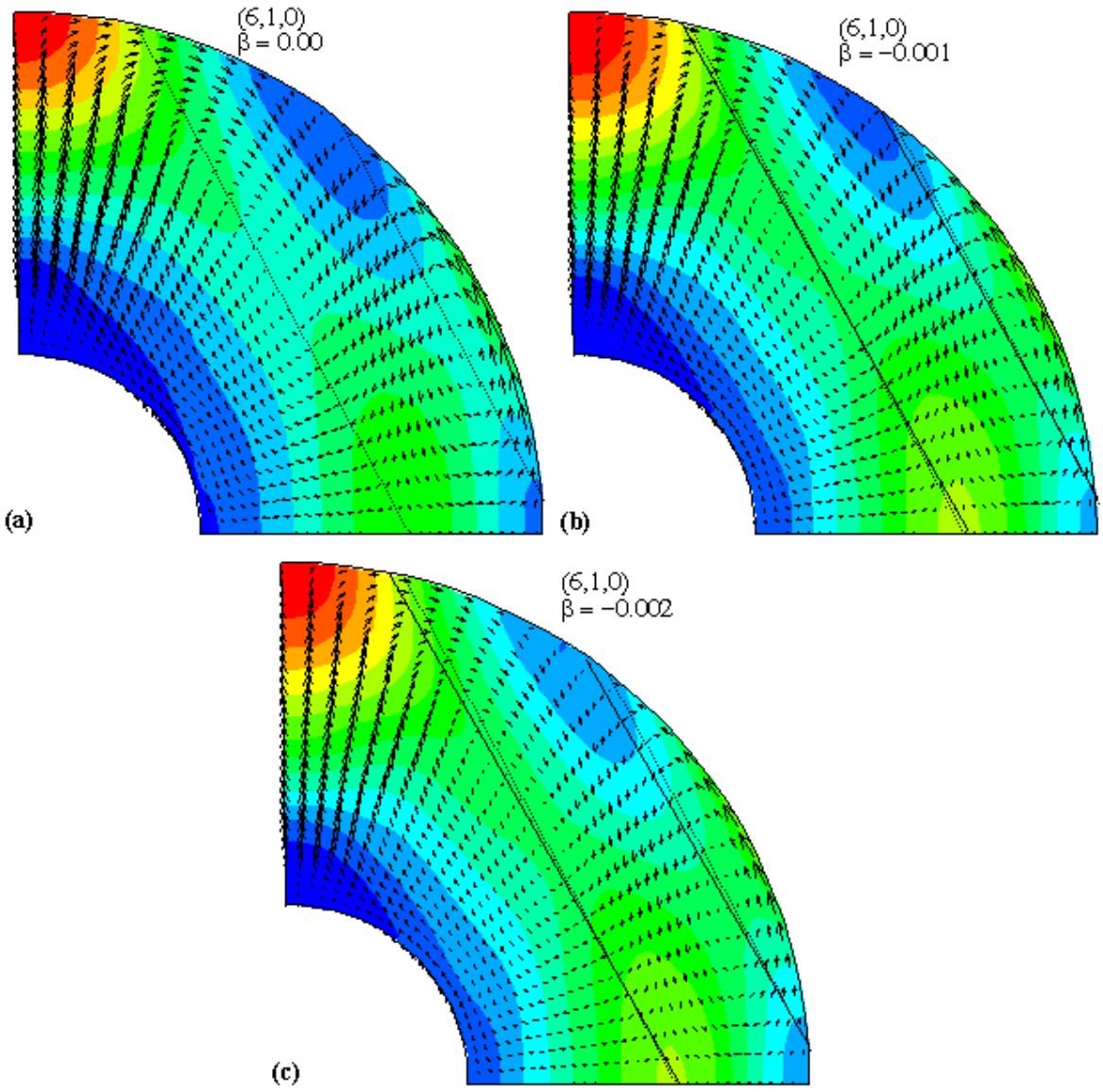

Figure 4.18: The meridional displacement vectors and the compressed potential, $\Phi = \chi - V_1$, contours of the (6,1,0) mode of the modified spherical shell core models with different stability parameter: (a) $\beta = 0.00$, (b) $\beta = -0.001$ and (c) $\beta = -0.002$, and the special characteristic lines: dotted lines for $\beta = 0.00$ and solid lines for respective core models for different β. Note that the horizontal axis is the equator and the vertical axis is the rotational axis.

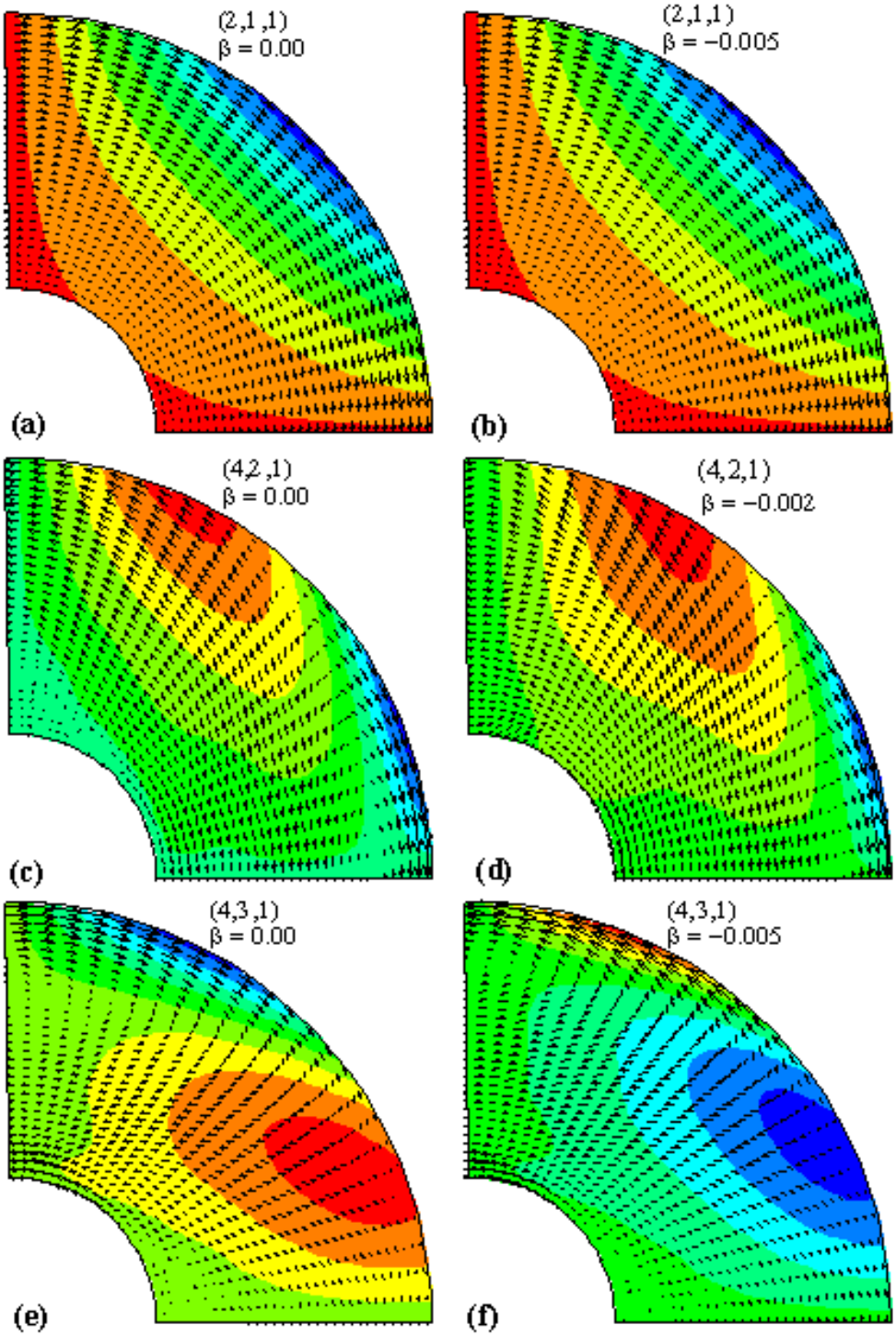

Figure 4.19: The displacement vectors and the compressed potential, $\Phi = \chi - V_1$, contours of the (2,1,1), (4,2,1) and (4,3,1) modes of a neutrally stratified shell model, left, and their counterparts for stably stratified shell models, right. Note that the horizontal axis is the equator and the vertical axis is the rotational axis.

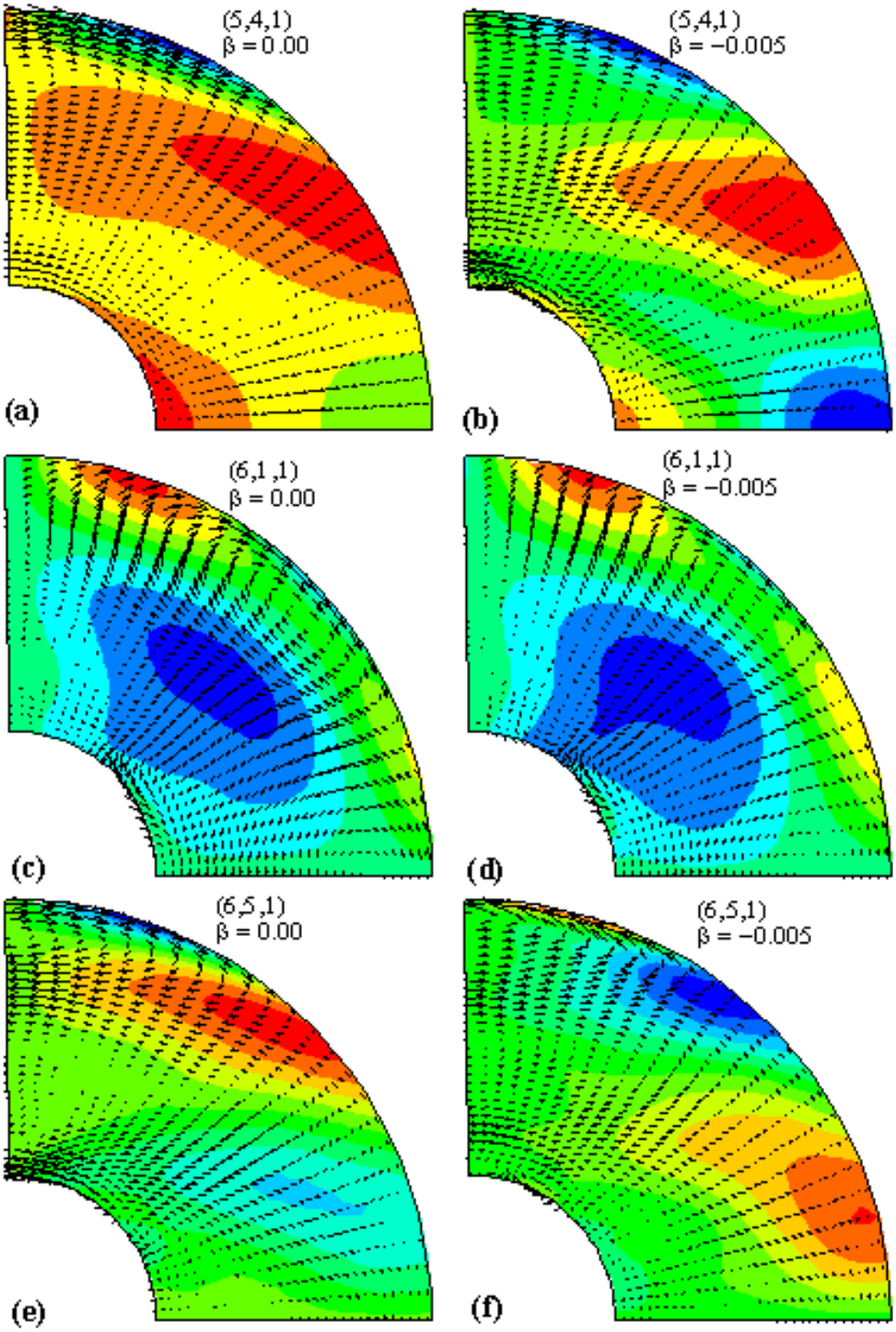

Figure 4.20: The displacement vectors and the compressed potential, $\Phi = \chi - V_1$, contours of the (5,4,1), (6,1,1) and (6,5,1) modes of a neutrally stratified, $\beta = 0.00$ shell model, left, and their counterparts for a shell model with $\beta = -0.005$, right. Note that the horizontal axis is the equator and the vertical axis is the rotational axis.

Chapter 5

Conclusions

Science is built up with facts, as a house is with stones. But a collection of facts is no more a science than a heap of stones is a house.

– Jules Henri Poincaré

In this thesis, we have shown that the effects of the density stratification on the frequencies of the inertial modes of the Earth's core. In order to study these effects, we have computed numerically the non-dimensional frequencies and eigenfunctions of some of the low order inertial modes for the sphere and spherical shell core models with different values of β. First, we have established the modified Earth core model based on PREM [2]. A Galerkin method is used to solve the dynamical equations of the core. We have expanded these equations using a spherical harmonic representation of the scalar fields V_1, χ and ζ, and have used the orthogonality relation and the linear independence properties of the spherical harmonics to remove the θ and ϕ dependence, so that the final coupled equations are functions of the radius only. We have also used the natural boundary conditions to reduce the second order derivatives to first order ones. These equations are then solved numerically for the frequencies, the displacement eigenfunctions and the compressed potential of the inertial modes of the core. We have shown that the frequencies of these modes are significantly affected by the density stratification for the different Earth's core models depending the values of β. For example, the non-dimensional frequency of the $(6,2,1)$ in-

ertial mode of a sphere is adjusted from -0.269 for the Poincaré core model to -0.340 for a core mode with the stability parameter of -0.005, a change of about 26%. The frequency of the (2,1,1) mode is not changed because this mode represents a tilt of the rotation axis, and the flow is solenoidal (i.e., $\nabla \cdot \vec{u} = 0$) which reduces the dynamical equations to the Poincaré equation.

We have also derived the web of characteristics which is the signature of the hyperbolic nature of the governing equation. The governing equation is of hyperbolic type so that some discontinuities in the pressure gradient occur the characteristic lines. We have shown that the characteristic lines are crossed where the an abrupt change in the slope or trend of the compressed potential contours occurs. In this study, we have explored the relation between the meridional displacement vectors and the characteristics for axisymmetric modes, and have computed a special characteristic line [26, 43] which indicates the location of the cell. We have shown that a special characteristic line is slightly shifted counterclockwise from a neutrally stratified core model to stably stratified core models, but slightly shifted clockwise for the unstably stratified core models. Thus the structures of eigenfunctions, except the (2,1,1) mode, are affected by the density stratification. In future work, we will include the ellipticity of the equipotential surfaces and the elasticity of the solid parts of the Earth to study the normal modes of more realistic Earth models.

Bibliography

[1] Jeffreys H. The rigidity of the earth's central core. *Mon. Notes R. Astr. Soc., Geophy. Suppl.*, 1:371–383, 1926.

[2] Dziewonski A. M. and Anderson D. L. Preliminary reference earth model. *Phys. Earth Planet. Inter.*, 25:297–356, 1981.

[3] Pekeris C. L. and Accad Y. Dynamics of the liquid core of the earth. *Phil. Trans. R. Soc. Lond.*, A273:237–260, 1972.

[4] Lowrie Willam. *Fundamentals of Geophysics*. Cambridge University Press, 2007.

[5] Benioff H., Gutenburg B., and Richter C. F. Progress report, seismological laboratory,. *Calif. Inst. Tech. 1953; Trans. Amer. Geophys. Union*, 35:979–987, 1954.

[6] Seyed-Mahmoud B., Heikoop J., and Seyed-Mahmoud R. Inertial modes of a compressible fluid core model. *Geophys. and Atro. Fluid Dyn.*, 101:1– 17, 2007.

[7] McElhinny M. W. and Senanayake W. E. Paleomagnetic evidence for existence of the geomagnetic field 3.5 ga ago. *J. of Geophys. Res.*, 85:3523–3528, 1980.

[8] Elsasser W. M. The earth's interior and geomagnetism. *Revs. Mod. Phys.*, 22:1–35, 1950.

[9] Seyed-Mahmoud B., Aldridge K. D., and Henderson G. A. Elliptical instability in rotating spherical fluid shells: application to earths fluid core. *Phys. Earth Planet. Inter.*, 117:257– 282, 2004.

[10] Kerswell R. R. The instaility of precessing flow. *Geophys. Atrso. Fluid Dyn.*, 72:107–114, 1993.

[11] Aldridge K. D., Seyed-Mahmoud B., Henderson G. A., and van-Wijngaarden W. Elliptical instability of the earths fluid core. *Phys. Earth Planet. Inter.*, 103:365–374, 1997.

[12] Seyed-Mahmoud B., Henderson G. A., and Aldridge K. D. A numerical model for the elliptical instability of the earths fluid core. *Phys. Earth Planet. Inter.*, 117:51–61, 2000.

[13] Moradi A. Elliptical instability in the planetary fluid cores. Master's thesis, The University of Lethbridge, AB, Canada, 2014.

[14] Kerswell R. R. Tidal excitation of hydrodynamics waves and their damping in the earth. *J. Fluid Mech.*, 274:219–241, 1994.

[15] Poincaré H. Sur la pré cession des corps déformables. *Bull. Astronomique.*, 27:321–335, 1910.

[16] Bryan G. The waves on a rotating liquid spheroid of finite ellipticity. *Phil. Trans. R. Soc. Lond.*, 180:187–219, 1989.

[17] Hough S. S. The oscillations of a rotating ellipsoidal shell containing fluid. *Phil. Trans. R. Soc. Lond.*, 186:469–506, 1895.

[18] Stewartson K. and Robers P. H. On the motion of a liquid in a spheroidal cavity of a precessing rigid body. *J. Fluid Mech.*, 17:1–20, 1963.

[19] Aldridge K. D. Axisymmetric inertial oscillations of a fluid in a rotating spherical shell. *Mathematika*, 19:163–168, 1972.

[20] Kelvin L.(William Thomson). Vibrations of a columnar vortex. *Phil. Mag.*, 10:155–168, 1880.

[21] Kudlick M. D. *On the transient motions in a contained rotating fluid*. PhD thesis, Massachusetts Institute of Technology, 1966.

[22] Stewartson K. and Rickard J. Pathological oscillations of a rotating fluid. *J. Fluid Mech.*, 35:759–773, 1969.

[23] Greenspan H.P. *The theory of rotating fluids*. Cambridge University Press, 1968.

[24] Aldridge K. D. Inertial waves and the earth's outer core. *Geophys. J. R. Astr. Soc.*, 42:337–345, 1975.

[25] Rieutord M. Inertial modes in the liquid core of the earth. *Phys. Earth Planet. Inter.*, 91:41– 46, 1995.

[26] Henderson G. A. *A finite-element method for weak solution of the Poincaré problem*. PhD thesis, York University, Ontario, Canada, 1996.

[27] Smith M. L. The scalar equations of an infinitestimal elastic-gravitational motion for a rotating slightly elliptical earth. *Geophys. J. R. Astr. Soc.*, 37:491–526, 1974.

[28] Crossley D. J. Core undertones with rotation. *Geophys. J. R. Astr. Soc.*, 42:477–488, 1975.

[29] Crossley D. J. Oscillatory flow in the liquid core. *Phys. Earth Planet. Inter., in press*, 1984.

[30] Shen P. Y. and Mansinha L. Oscillations, nutation and wobble of an elliptical rotating earth with liquid outer core. *Geophys. J. R. Astr. Soc.*, 46:467–494, 1976.

[31] Rogister Y. and Valette B. Influence of liquid core dynamics on rotational modes. *Geophys. J. Int.*, 176:368–388, 2009.

[32] Seyed-Mahmoud B. and Rochester M. G. Dynamics of rotating fluids described by scalar potentials. *Phys. Earth Planet. Inter.*, 156:143–151, 2006.

[33] Chandrasekhar S. *Hydrodynamics and Hydromagnetic Stability*. Oxford University Press, Oxford, 1961.

[34] Smylie D. E. and Rochester M. G. Compressibility, core dynamics and the subseismic wave equation. *Phys. Earth Planet. Inter.*, 24:308–319, 1981.

[35] Seyed-Mahmoud B. Wobble/nutation of a rotating ellipsoidal earth with liquid outer core: Implementation of a new set of equations describing dyanmics of rotating fluids. Master's thesis, Memorial University of Newfoundland, Canada, 1994.

[36] Smylie D. E., Szeto A. M. K., and Rochester M. G. The dynamics of the earth's inner and outer cores. *Req. Prog. Phys.*, 47:855–906, 1984.

[37] Masters G. Observational constraints on the chemical and thermal structure of the earth's deep interior. *Geophys. J. R. Astr. Soc.*, 57:507–534, 1979.

[38] Alterman Z., Jarosch H., and Pekeris C. L. Oscillations of the earth,. *Proc. Soc. London*, A 252:80–95, 1959.

[39] Smith M. L. Wobble and nutation of the earth,. *Geophys. J. R. Astr. Soc.*, 50:103–140, 1977.

[40] Wu W. J. and Rochester M. G. Core dynamics: the two-potential description and a new variational principle. *Geophys. J. Int.*, 103:697–706, 1990.

[41] Rochester M. G. and Peng Z. R. The slichter modes of a rotating earth: A test of the subseimic approximation,. *Geophys. J. Int.,*, 113:575–585, 1993.

[42] Peng Z. R. *The Slichter Modes in a Realistic Earth Model.* PhD thesis, Memorial University of Newfoundland, Canada, 1995.

[43] Hø iland E. Discussion of a hyperbolic equation relating to inertia and gravitational fluid oscillations. *Geofys. Publ.*, 91:211–227, 1962.

[44] Dintrans B., Rieutord M., and Valdettaro L. Gravito-inertial waves in rotating stratified sphere or spherical shell,. *Geophys. J. Int.*, 398:271–297, 1999.

[45] Chandrasekhar S. and Roberts P. H. The ellipticity of a slowly rotating configuration. *Astrophysical Journal*, 138:801–808, 1963.

[46] Seyed-Mahmoud B. and Moradi A. Dynamics of the earths fluid core: implementation of a clairaut coordinate system. *Phys. Earth Planet. Inter.*, 227:61–67, 2014.

[47] Rogue Wave Software. Imls fortran numerical library: Mathematical functions in fortran. *www.absoft.com/Support/Documentation/MathV1.pdf.*

[48] Nzikou M. Mamboukou. The earth's slichter modes. Master's thesis, The University of Lethbridge, AB, Canada, 2013.

[49] Rieutord M. and Valdettaro L. Inertial waves in a rotating spherical shell. *J. Fluid Mech.*, 341:77–99, 1997.

[50] Debnath L. *Nonlinear Partial Differential Equations for Scientists and Engineers. 2nd edu.* Birkhäuser, Boston, 2004.

Appendix A

Method of Characteristics

A.1 Method of Characteristics

The general second order linear partial differential equation in the two independent variables x, y is given by [50]

$$au_{xx} + bu_{xy} + cu_{yy} + du_x + eu_y + fu = h, \tag{A.1}$$

where $u_{xx} = \frac{\partial^2 u}{\partial x^2}, u_{yy} = \frac{\partial^2 u}{\partial y^2}, u_{xy} = \frac{\partial^2 u}{\partial x \partial y}, u_x = \frac{\partial u}{\partial x}, u_y = \frac{\partial u}{\partial y}$. a, b, c, d, e, f, and h are given functions of x and y or constants.

We consider the transformation from x, y to ξ, η is defined by

$$\xi = \phi(x, y), \tag{A.2}$$
$$\eta = \psi(x, y), \tag{A.3}$$

where ϕ and ψ are twice continuously differentiable and the Jacobian $J(x, y) = \phi_x \psi_y - \psi_x \phi_y$ is nonzero in a domain of interest. We now use the chain rule in the following:

$$\begin{aligned}
u_x &= u_\xi \xi_x + u_\eta \eta_x, \\
u_y &= u_\xi \xi_y + u_\eta \eta_y, \\
u_{xx} &= u_{\xi\xi} \xi_x^2 + 2u_{\xi\eta} \xi_x \eta_x + u_{\eta\eta} \eta_x^2 + u_\xi \xi_{xx} + u_\eta \eta_{xx}, \\
u_{yy} &= u_{\xi\xi} \xi_y^2 + 2u_{\xi\eta} \xi_x \eta_y + u_{\eta\eta} \eta_y^2 + u_\xi \xi_{yy} + u_\eta \eta_{yy}, \\
u_{yy} &= u_{\eta\eta} \xi_x \xi_y + u_{\xi\eta} (\xi_x \eta_y + \xi_y \eta_x) + u_{\eta\eta} \eta_x \eta_y + u_\xi \xi_{xy} + u_\eta \eta_{yy}
\end{aligned}$$

Substituting these results in Eq. (A.1);

$$a^* u_{\xi\xi} + b^* u_{\eta\xi} + c^* u_{\eta\eta} + d^* u_\xi + e^* u_\eta + f^* u = h^*, \tag{A.4}$$

where,

$$\begin{aligned}
a^* &= a\xi_x^2 + b\xi_x \xi_y + c\xi_y^2, \\
b^* &= 2a\xi_x \eta_x + b(\xi_x \eta_y + \xi_y \eta_x) + 2c\xi_y \eta_y, \\
c^* &= a\eta_x^2 + b\eta_x \eta_y + c\eta_y^2, \\
d^* &= a\xi_{xx} + b\xi_{xy} + c\xi_{yy} + d\xi_x + e\xi_y, \\
e^* &= a\eta_{xx} + b\eta_{xy} + c\eta_{yy} + d\eta_x + e\eta_y, \\
f^* &= f, \\
h^* &= h.
\end{aligned}$$

The problem is now to determine ξ and η so that Eq.(A.4) takes the simplest possible

form. We choose that $a^* = c^* = 0$ and $b^* \neq 0$, i.e.,

$$a^* = a\xi_x^2 + b\xi_x\xi_y + c\xi_y^2 = 0, \tag{A.5}$$

$$c^* = a\eta_x^2 + b\eta_x\eta_y + c\eta_y^2 = 0. \tag{A.6}$$

These two equations can be combined into a single quadratic equation for $\varsigma = \xi$ or η

$$a(\frac{\varsigma_x}{\varsigma_y})^2 + b(\frac{\varsigma_x}{\varsigma_y}) + c = 0. \tag{A.7}$$

We consider level curves $\xi = \phi(x,y) = \ constant = C_1$ and $\eta = \psi(x,y) = \ constant = C_2$. On these curves

$$d\xi = \xi_x dx + \xi_y dy = 0, \tag{A.8}$$

$$d\eta = \eta_x dx + \eta_y dy = 0, \tag{A.9}$$

that is, the slopes of these curves are given by

$$\frac{dy}{dx} = -\frac{\xi_x}{\xi_y}, \tag{A.10}$$

$$\frac{dy}{dx} = -\frac{\eta_x}{\eta_y}. \tag{A.11}$$

Thus, the slopes of both level curves are the roots of the same quadratic equation which is obtained fron Eq. (A.7) as

$$a(\frac{dy}{dx})^2 - b(\frac{dy}{dx}) + c = 0, \tag{A.12}$$

and the roots of this equation are given by

$$\frac{dy}{dx} = \frac{1}{2a}(b \pm \sqrt{b^2 - 4ca}). \tag{A.13}$$

These equations are known as the *characteristic* equations for Eq.(A.1), and their solutions are called the *characteristic* curves or simply the *characteristics* of Eq. (A.1). We also see from Eq. (A.13) that the *characteristic* equations for Eq. (A.1) depend only the coefficients of second order derivatives. The solution of the two ordinary differential equations (A.13) define two distinct families of characteristics $\phi(x,y) = C_1$ and $\psi(x,y) = C_2$, where C_1 and C_2 are constants of integration. There are three possible cases:

Case I: $\sqrt{b^2 - 4ac} > 0$

Equations for which $\sqrt{b^2 - 4ac} > 0$ are called *hyperbolic*. Integration of Eqs. (A.13) gives two real and distinct families of characteristics $\phi(x,y) = C_1$ and $\psi(x,y) = C_2$. It is important to point out that characteristics play a fundamental role in the theory of hyperbolic equations.

Case II: $\sqrt{b^2-4ac}=0$
Equations for which $\sqrt{b^2-4ac}=0$ are called *parabolic*. Integration of Eqs. (A.13) gives only one family of real characteristics $\phi(x,y)=C_1$ or $\psi(x,y)=C_2$.

Case III: $\sqrt{b^2-4ac}<0$
Equations for which $\sqrt{b^2-4ac}<0$ are called *elliptic*. In this case, equations (A.13) have no real solutions. So there are two families of complex characteristics.

Appendix B

Detailed Matrix Formulation from Chapter 3

(Here we show the matrix formulation of the Poincaré equation and the 3PD)

B.1 Matrix Formulation of the Poincaré Equation

The Galerkin formulation of the Poincaré equation (3.37) can be rewritten as

$$\begin{aligned}\sum_{l=1}^{L}\int_0^x \{D^m_{L(q-1)+l}[(\sigma^2-\frac{1}{3})\frac{df_l}{dx}\frac{df_k}{dx}x^2+m\sigma(f_k\frac{df_l}{dx}+f_l\frac{df_k}{dx})x+f_lf_k\{m^2\\ -q(q+1)(\frac{2}{3}-\sigma^2)+m\sigma\}-B^m_q\{\frac{2}{3}\frac{df_l}{dx}\frac{df_k}{dx}x^2+(f_l\frac{df_k}{dx}+f_k\frac{df_l}{dx})x\\ -(\frac{2}{3}q(q+1)-2)f_lf_k\}]-\frac{2}{3}D^m_{L(q+1)+l}A^m_{q+2}[\frac{df_l}{dx}\frac{df_k}{dx}x^2+\{q(f_l\frac{df_k}{dx}\\ -f_k\frac{df_l}{dx})+3f_l\frac{df_k}{dx}\}x-q(q+3)f_lf_k]-\frac{2}{3}D^m_{L(q-3)+l}C^m_{q-2}[\frac{df_l}{dx}\frac{df_k}{dx}x^2\\ -\{(q-2)(f_l\frac{df_k}{dx}-f_k\frac{df_l}{dx})-3f_k\frac{df_l}{dx}\}x-(q+1)(q-2)f_lf_k]\}dx=\ 0,\end{aligned} \tag{B.1}$$

for every $k=1,\cdots,L$ and $q=1,\cdots,N$, which [i.e., Eq. (B.1)] gives

$$\begin{aligned}\sum_{l=1}^{L}[F_{L(q-1)+k,L(q-1)+l}D_{L(q-1)+l}+F'_{L(q-1)+k,Lq+l}D_{Lq+l}\\ +F''_{L(q-1)+k,L(q-2)+l}D_{L(q-2)+l}]=\ 0,\end{aligned} \tag{B.2}$$

where,

$$\begin{aligned}F_{L(q-1)+k,L(q-1)+l}=\int_0^x[(\sigma^2-\frac{1}{3})\frac{df_l}{dx}\frac{df_k}{dx}x^2+m\sigma(f_k\frac{df_l}{dx}+f_l\frac{df_k}{dx})x\\ +f_lf_k\{m^2-q(q+1)(\frac{2}{3}-\sigma^2)+m\sigma\}-B^m_q\{\frac{2}{3}\frac{df_l}{dx}\frac{df_k}{dx}x^2\\ +(f_l\frac{df_k}{dx}+f_k\frac{df_l}{dx})x-(\frac{2}{3}q(q+1)-2)f_lf_k\}]dx,\end{aligned} \tag{B.3}$$

$$\begin{aligned}F'_{L(q-1)+k,Lq+l}=-\frac{2}{3}A^m_{q+2}\int_0^x[\frac{df_l}{dx}\frac{df_k}{dx}x^2+\{q(f_l\frac{df_k}{dx}-f_k\frac{df_l}{dx})\\ +3f_l\frac{df_k}{dx}\}x-q(q+3)f_lf_k]dx,\end{aligned} \tag{B.4}$$

$$F''_{L(q-1)+k,L(q-2)+l} = -\frac{2}{3}C^m_{q-2}\int_0^x [\frac{df_l}{dx}\frac{df_k}{dx}x^2 - \{(q-2)(f_l\frac{df_k}{dx} - f_k\frac{df_l}{dx}) \\ -3f_k\frac{df_l}{dx}\}x - (q+1)(q-2)f_lf_k]dx. \tag{B.5}$$

The equation (B.2) leads to the matrix representation as

$$A\psi = 0, \tag{B.6}$$

where A is a $NL \times NL$ matrix which makes from Eqs. (B.3) - (B.5), and ψ is a NL vector representing the coefficients of D_i in Eq. (B.2). An example of the shape of a matrix is given by assuming $N = 2$ and $L = 3$. Then, for the above choice, the matrix representation of Eq. (B.6) is shown below:

$$\begin{bmatrix} F_{1,1} & F_{1,2} & F_{1,3} & F'_{1,4} & F'_{1,5} & F'_{1,6} \\ F_{2,1} & F_{2,2} & F_{2,3} & F'_{2,4} & F'_{2,5} & F'_{2,6} \\ F_{3,1} & F_{3,2} & F_{3,3} & F'_{3,4} & F'_{3,5} & F'_{3,6} \\ F''_{4,1} & F''_{4,2} & F''_{4,3} & F_{4,4} & F_{4,5} & F_{4,6} \\ F''_{5,1} & F''_{5,2} & F''_{5,3} & F_{5,4} & F_{5,5} & F_{5,6} \\ F''_{6,1} & F''_{6,2} & F''_{6,3} & F_{6,4} & F_{6,5} & F_{6,6} \end{bmatrix} \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ D_4 \\ D_5 \\ D_6 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{B.7}$$

B.2 Matrix Formulation of the 3PD

The Galerkin formulation of the momentum equation (3.51) is given by

$$\sum_{l=1}^{L}[M_{L(q-1)+k,L(q-1)+l}D_{L(q-1)+l} + M'_{L(q-1)+k,Lq+l}D_{Lq+l} + M''_{L(q-1)+k,L(q-2)+l}D_{L(q-2)+l} \\ +M1_{L(q-1)+k,L(N+q-1)+l}D_{L(N+q-1)+l} + M1'_{L(q-1)+k,L(N+q)+l}D_{L(N+q)+l} \\ +M1''_{L(q-1)+k,L(N+q-2)+l}D_{L(N+q-2)+l} + M2_{L(q-1)+k,L(2N+q-1)+l}D_{L(2N+q-1)+l} \\ +M2'_{L(q-1)+k,L(2N+q)+l}D_{L(2N+q)+l} + M2''_{L(q-1)+k,L(2N+q-2)+l}D_{L(2N+q-2)+l}] = 0, \tag{B.8}$$

where

$$M_{L(q-1)+k,L(q-1)+l} = \int_0^x [(\sigma^2 - \frac{1}{3})\frac{df_l}{dx}\frac{df_k}{dx}x^2 + m\sigma(f_k\frac{df_l}{dx} + f_l\frac{df_k}{dx})x \\ +f_lf_k\{m^2 - q(q+1)(\frac{2}{3} - \sigma^2) + m\sigma\} - B^m_q\{\frac{2}{3}\frac{df_l}{dx}\frac{df_k}{dx}x^2 \\ +(f_l\frac{df_k}{dx} + f_k\frac{df_l}{dx})x - (\frac{2}{3}q(q+1) - 2)f_lf_k\}]dx, \tag{B.9}$$

$$M'_{L(q-1)+k,Lq+l} = -\frac{2}{3}A^m_{q+2}\int_0^x [\frac{df_l}{dx}\frac{df_k}{dx}x^2 + \{q(f_l\frac{df_k}{dx} - f_k\frac{df_l}{dx}) \\ +3f_l\frac{df_k}{dx}\}x - q(q+3)f_lf_k]dx, \tag{B.10}$$

$$M''_{L(q-1)+k,L(q-2)+l} = -\frac{2}{3}C^m_{q-2}\int_0^x [\frac{df_l}{dx}\frac{df_k}{dx}x^2 - \{(q-2)(f_l\frac{df_k}{dx} - f_k\frac{df_l}{dx}) \\ -3f_k\frac{df_l}{dx}\}x - (q+1)(q-2)f_lf_k]dx, \tag{B.11}$$

$$M1_{L(q-1)+k,L(N+q-1)+l} = -\int_0^x [(\sigma^2 - \frac{1}{3})\frac{df_l}{dx}\frac{df_k}{dx}x^2 + m\sigma(f_k\frac{df_l}{dx} + f_l\frac{df_k}{dx})x \\ +f_lf_k\{m^2 - q(q+1)(\frac{2}{3} - \sigma^2) + m\sigma\} - B^m_q\{\frac{2}{3}\frac{df_l}{dx}\frac{df_k}{dx}x^2 \\ +(f_l\frac{df_k}{dx} + f_k\frac{df_l}{dx})x - (\frac{2}{3}q(q+1) - 2)f_lf_k\}]dx, \tag{B.12}$$

$$M1'_{L(q-1)+k,L(N+q)+l} = \frac{2}{3}A^m_{q+2}\int_0^x [\frac{df_l}{dx}\frac{df_k}{dx}x^2 + \{q(f_l\frac{df_k}{dx} - f_k\frac{df_l}{dx}) \\ +3f_l\frac{df_k}{dx}\}x - q(q+3)f_lf_k]dx, \tag{B.13}$$

$$M1''_{L(q-1)+k,L(N+q-2)+l} = \frac{2}{3}C^m_{q-2}\int_0^x [\frac{df_l}{dx}\frac{df_k}{dx}x^2 - \{(q-2)(f_l\frac{df_k}{dx} - f_k\frac{df_l}{dx}) \\ -3f_k\frac{df_l}{dx}\}x - (q+1)(q-2)f_lf_k]dx, \tag{B.14}$$

$$M2_{L(q-1)+k,L(2N+q-1)+l} = -\int_0^x [\beta f_l(-\sigma^2 g_r + \frac{1}{3}g_r)\frac{df_k}{dx}x^2 - (m\sigma\beta g_r x \\ +\sigma^2(\sigma^2 - 1)x^2)f_lf_k + B^m_q\beta g_r(\frac{2}{3}f_l\frac{df_k}{dx}x^2 + f_lf_kx)]dx, \tag{B.15}$$

$$M2'_{L(q-1)+k,L(2N+q)+l} = -A^m_{q+2}\int_0^x [\beta g_r[\frac{2}{3}f_l\frac{df_k}{dx}x^2 + \{2 - \frac{2}{3}(q+3)\}f_lf_kx]]dx, \tag{B.16}$$

$$M2''_{L(q-1)+k,L(2N+q-2)+l} = -C^m_{q-2}\int_0^x [\beta g_r[\frac{2}{3}f_l\frac{df_k}{dx}x^2 + \{2 + \frac{2}{3}(q-2)\}f_lf_kx\}]dx, \tag{B.17}$$

for every $k = 1, \cdots, L$ and $q = 1, \cdots, N$.

Again, the Galerkin formulation of the Poisson's equation (3.62) is

$$\sum_{l=1}^{L}[P_{L(N+q-1)+k,L(q-1)+l}D_{L(q-1)+l} + P1_{L(N+q-1)+k,L(N+q-1)+l}D_{L(N+q-1)+l} \\ +P2_{L(N+q-1)+k,L(2N+q-1)+l}D_{L(2N+q-1)+l}] = 0, \tag{B.18}$$

where,

$$P_{L(N+q-1)+k,L(q-1)+l} = -4\pi G \int_0^x (\frac{1-\beta}{\alpha^2}) f_l f_k x^2 dx, \tag{B.19}$$

$$P1_{L(N+q-1)+k,L(N+q-1)+l} = (q+1) R_{CMB} f_l(R_{CMB}) f_k(R_{CMB}) + \int_0^x (\frac{df_k}{dx}\frac{df_l}{dx}x^2 + q(q+1) f_l f_k) dx, \tag{B.20}$$

$$P2_{L(N+q-1)+k,L(2N+q-1)+l} = 4\pi G \int_0^x \beta f_k f_l x^2 dx, \tag{B.21}$$

for $k = 1, \cdots, L$ and $q = 1, \cdots, N$.

Now, the Galerkin formulation of the entropy equation (3.68) is given by

$$\begin{aligned}
\sum_{l=1}^{L} [&E_{L(2N+q-1)+k,L(q-1)+l} D_{L(q-1)+l} + E'_{L(2N+q-1)+k,Lq+l} D_{Lq+l} \\
&+ E''_{L(2N+q-1)+k,L(q-2)+l} D_{L(q-2)+l} + E1_{L(2N+q-1)+k,L(N+q-1)+l} D_{L(N+q-1)+l} \\
&+ E1'_{L(2N+q-1)+k,L(N+q)+l} D_{L(N+q)+l} + E1''_{L(2N+q-1)+k,L(N+q-2)+l} D_{L(N+q-2)+l} \\
&+ E2_{L(2N+q-1)+k,L(2N+q-1)+l} D_{L(2N+q-1)+l} + E2'_{L(2N+q-1)+k,L(2N+q)+l} D_{L(2N+q)+l} \\
&+ E2''_{L(2N+q-1)+k,L(2N+q-2)+l} D_{L(2N+q-2)+l}] = 0,
\end{aligned} \tag{B.22}$$

where

$$\begin{aligned}
E_{L(2N+q-1)+k,L(q-1)+l} = \int_0^x [&f_k(-\sigma^2 g_r + \frac{1}{3} g_r)\frac{df_l}{dx}x^2 - (m\sigma g_r x + \sigma^2(\sigma^2-1)x^2) f_l f_k \\
&+ B_q^m (\frac{2}{3} g_r f_k \frac{df_l}{dx}x^2 + g_r f_l f_k x)] dx,
\end{aligned} \tag{B.23}$$

$$E'_{L(2N+q-1)+k,Lq+l} = A_{q+2}^m \int_0^x [\frac{2}{3} g_r (f_k \frac{df_l}{dx}x^2 + (q+3) f_l f_k x)] dx, \tag{B.24}$$

$$E''_{L(2N+q-1)+k,L(q-2)+l} = C_{q-2}^m \int_0^x [\frac{2}{3} g_r (f_k \frac{df_l}{dx}x^2 - (q-2) f_l f_k x)] dx, \tag{B.25}$$

$$\begin{aligned}
E1_{L(2N+q-1)+k,L(N+q-1)+l} = -\int_0^x [&f_k(-\sigma^2 g_r + \frac{1}{3} g_r)\frac{df_l}{dx}x^2 - m\sigma g_r f_l f_k x \\
&+ B_q^m (\frac{2}{3} g_r f_k \frac{df_l}{dx}x^2 + g_r f_l f_k x)] dx,
\end{aligned} \tag{B.26}$$

$$E1'_{L(2N+q-1)+k,L(N+q)+l} = -A^m_{q+2}\int_0^x [\frac{2}{3}g_r(f_k\frac{df_l}{dx}x^2 + (q+3)f_l f_k x)]dx, \tag{B.27}$$

$$E1''_{L(2N+q-1)+k,L(N+q-2)+l} = -C^m_{q-2}\int_0^x [\frac{2}{3}g_r(f_k\frac{df_l}{dx}x^2 - (q-2)f_l f_k x)]dx, \tag{B.28}$$

$$\begin{aligned} E2_{L(2N+q-1)+k,L(2N+q-1)+l} = -\int_0^x [f_l f_k\{\alpha^2\sigma^2(\sigma^2-1) + \beta(\sigma^2 g_0^2 - \frac{1}{3}g_r^2 \\ -\frac{2}{3}B^m_q g_r^2)\}x^2]dx, \end{aligned} \tag{B.29}$$

$$E2'_{L(2N+q-1)+k,L(2N+q)+l} = \frac{2}{3}A^m_{q+2}\int_0^x \beta g_r^2 f_l f_k x^2 dx, \tag{B.30}$$

$$E2''_{L(2N+q-1)+k,L(2N+q-2)+l} = \frac{2}{3}C^m_{q-2}\int_0^x \beta g_r^2 f_l f_k x^2 dx, \tag{B.31}$$

for every for $k = 1,\cdots,L$, and $q = 1,\cdots,N$. The Eqs. (B.8), (B.18) and (B.22) lead to the matrix representation as

$$A'\psi' = 0, \tag{B.32}$$

where A' is a $3NL \times 3NL$ matrix which makes from Eqs. (B.9) - (B.17), (B.19) - (B.21) and (B.23) - (B.31), and ψ' is a $3NL$ vector representing the coefficients of D_i. An example of the shape of a matrix is given by assuming $N = 2$ and $L = 2$. Then, for the above choice, the Eq. (B.32) can be written in terms of matrix as:

$$
\left[\begin{array}{cccc|cccc|cccc}
M_{1,1} & M_{1,2} & M'_{1,3} & M'_{1,4} & M1_{1,5} & M1_{1,6} & M1'_{1,7} & M'_{1,8} & M2_{1,9} & M2_{1,10} & M2'_{1,11} & M2'_{1,12} \\
M_{2,1} & M_{2,2} & M'_{2,3} & M'_{2,4} & M1_{2,5} & M1_{2,6} & M1'_{2,7} & M1'_{2,8} & M2_{2,9} & M2_{2,10} & M2'_{2,11} & M2'_{2,12} \\
M''_{3,1} & M''_{3,2} & M_{3,3} & M_{3,4} & M1''_{3,5} & M1''_{3,6} & M1_{3,7} & M1_{3,8} & M2''_{3,9} & M2''_{3,10} & M2_{3,11} & M2_{3,12} \\
M''_{4,1} & M''_{4,2} & M_{4,3} & M_{4,4} & M1''_{4,5} & M1''_{4,6} & M1_{4,7} & M1_{4,8} & M2''_{4,9} & M2''_{4,10} & M2_{4,11} & M2_{4,12} \\
\hline
P_{5,1} & P_{5,2} & 0 & 0 & P1_{5,5} & P1_{5,6} & 0 & 0 & P2_{5,9} & P2_{5,10} & 0 & 0 \\
P_{6,1} & P_{6,2} & 0 & 0 & P1_{6,5} & P1_{6,6} & 0 & 0 & P2_{6,9} & P2_{6,10} & 0 & 0 \\
0 & 0 & P_{7,3} & P_{7,4} & 0 & 0 & P1_{7,7} & P1_{7,8} & 0 & 0 & P2_{7,11} & P2_{7,12} \\
0 & 0 & P_{8,3} & P_{8,4} & 0 & 0 & P1_{8,7} & P1_{8,8} & 0 & 0 & P2_{8,11} & P2_{8,12} \\
\hline
E_{9,1} & E_{9,2} & E'_{9,3} & E'_{9,4} & E1_{9,5} & E1_{9,6} & E1'_{9,7} & E1'_{9,8} & E2_{9,9} & E2_{9,10} & E2'_{9,11} & E2'_{9,12} \\
E_{10,1} & E_{10,2} & E'_{10,3} & E'_{10,4} & E1_{10,5} & E1_{10,6} & E1'_{10,7} & E1'_{10,8} & E2_{10,9} & E2_{10,10} & E2'_{10,11} & E2'_{10,12} \\
E''_{11,1} & M''_{11,2} & E_{11,3} & E_{11,4} & E1''_{11,5} & E1''_{11,6} & E1_{11,7} & E1_{11,8} & E2''_{11,9} & E2''_{11,10} & E2_{11,11} & E2_{11,12} \\
E''_{12,1} & E''_{12,2} & E_{12,3} & E_{12,4} & E1''_{12,5} & E1''_{12,6} & E1_{12,7} & E1_{12,8} & E2''_{12,9} & E2''_{12,10} & E2_{12,11} & E2_{12,12}
\end{array}\right]
\times
\begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ D_4 \\ D_5 \\ D_6 \\ D_7 \\ D_8 \\ D_9 \\ D_{10} \\ D_{11} \\ D_{12} \end{bmatrix}
=
\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{B.33}
$$

Appendix C

Codes to Compute the Inertial Modes

C.1 The Code for the Modified PREM Model

The FORTRAN code computes the P-wave speed, gravity and the coefficient of density for the stability parameter $\beta = 0.00$ of the modified core model as below:

```
    program PREMLC
implicit none
integer :: irule, nout,ipath
integer, parameter::k= 29, p=12, q=3, LDA=p-2, N=p-2
double precision ::c(q), d(p) double precision ::x1,dx, x, PVM, RO, g, gro, beta, re, ri, rl,
dbeta, d1, mro, tolerance, db, y
double precision ::F, xi, xf, result,errabs, errest, errrel, A(LDA,LDA), B(N), CD(N)
integer ::l, i1,i2,j
character*40 ::flname1
external ::F
common/xxx/c,d,i1,i2

    re= 6371.0d0 ! radius of earth
ri= 1221.5d0/re ! radius of inner centre
rl= 3480.0d0/re ! radius of liquid core
x1=0.0d0
dx=3480.0d0/k
c(1)= 10.6776d3
c(2)= 0.0d0
c(3)= -8.7572d3
mro=10986.8795d0 ! mean density of outer core
d(1)= +12.5815d3
d(2)= +0.0d0
d(3)= -3.6426d3
d(4)= -5.5281d3
dbeta=0.00d0
tolerance= 1.0d-6

    OPEN(unit=10, file=flname1, access= 'append')
write(10,11) dbeta
write(*,11) dbeta
close(10)
```

```
OPEN(unit=20, file='density1.dat', status='replace', action= 'write')

    call UMACH (2, nout)
! Set limits of integration
xi= 0.d0
xf = rl
! Set error tolerances
errabs= 1.0d-8
! errest= 1.0d-8
errrel= 0.0d0
! Parameter for non-oscillatory function
irule = 6

    200 do i1=1, p-2
do i2=3, p
call DQDAG (F, xi, xf, errabs, errrel, irule, result, errest)
A(i1,i2-2)=(i2-1)*result
end do
B(i1)=-(1.0d0-dbeta)/(i1+2)*rl**(i1+2) !i is replaced by i+1
end do

    ipath=1 !means the system AX=B is solved
call DLSARG(N, A, LDA, B, ipath, CD)
do j=1, p-2
d(j+2)=CD(j)
end do

    d1=0.0d0
do j=3, p
d1=d1+d(j)/(j+2)*rl**(j+2)
end do

    d(1)=mro-3*(1.0d0/rl)**3*d1
d(2)=0.0d0
do l= 1, k+1
x= x1+(l-1)*dx
y=x/6371
call PREM(y, PVM, RO, g, gro, beta)
db= abs(beta - dbeta)
if (db.ge.tolerance) goto 200
write (20,*) x, PVM, RO
print*,x, PVM, RO
end do
close(20)
```

```
    do j=1, p
OPEN(unit=10, file=flname1, access= 'append')
write (10,12) d(j)
write (*,12) d(j)
close(10)
end do
11 format (' ',,'dbeta.=', f16.9)
12 format(' ','d(j)=',D16.8)
end program PREMLC
!'-------------------------------------------------------
double precision function F(x)
double precision, intent(in)::x
integer, parameter::p=12, q=3
double precision ::c(q), d(p)
double precision ::re, PVM, RO, g, gro, beta
integer ::i1,i2
common/xxx/c,d,i1,i2

    re= 6371.0d3 ! earth radius
call PREM(x, PVM, RO, g, gro, beta)

    F=PVM**2/(RO*g)/re*x**(i1+i2-2) ! i is replaced by i+1
end function F
!-------------------------------------------------------
subroutine PREM (y,PVM, RO, g, gro, beta)
integer, parameter::p=12, q=3
double precision ::c(q), d(p)
double precision ::y, re, PVM, RO, g, pi, GO, k, gro, m, beta
integer ::j,i1,i2
common/xxx/c,d,i1,i2
PVM = 0.00d0
RO = 0.00d0
g= 0.00d0
gro= 0.00d0
beta= 0.00d0
re= 6371.0d0 ! earth radius
pi= dacos(-1.0d0)
GO = 6.6690941D-11 !gravitational constant (PREM)
k= 4*pi*GO*re*1.0d3
m= 1.0d0/(re*1.0d3)
do j=1, q
PVM = PVM + c(j)*(y**(j-1)) !speed
end do
do 100 j=1, p
RO = RO + d(j)*(y**(j-1)) ! density
```

```
g = g+ k*d(j)/(j+2)*y**(j-1) ! g=graviry/y
if (j .le. 2) goto 100
gro= gro + m*(j-1)*d(j)*(y**(j-3)) !gro=gradient of ro/y
100 continue
beta = 1.0d0+ (PVM**2/(RO*g))*gro
end subroutine PREM
!————————————————————————————————————
```

C.2 The Code for the Frequencies of Inertial Modes

The FORTRAN code computes the non-dimensional frequencies of the inertial modes for a compressible and modified spherical core models as below:

```
    program trPD
implicit none
integer :: LM,N,imode
integer:: ne, LDA, ldfact, mv, mu
common/xix/imode
ne=3
1000 continue
write(*,*) 'Enter LM'
read(*,*) LM
write(*,*) 'Enter N'
read(*,*) N
write(*,*) 'Enter imode, 1 for b=-0.0, 2 for b=-0.001... 6 for b=-0.005'
read(*,*) imode
LDA=ne*LM*N
ldfact=LDA
mv=LDA
mu=2

    call poincare(LM,N,LDA,ldfact,mv,mu,ne)
write(*,*) 'Passt PC'
goto 1000
stop
end program trPD
!————————————————————————————————————-
subroutine poincare(LM,N,LDA,ldfact,mv,mu,ne)
implicit none
integer:: LM,N,LDA,ldfact,mv,mu,ne
double precision ::AX, BX, CX, FX, dFX, E1, E2, E3, E4, E5, E6, E7, E8, E9
double precision ::M1, M2, M3, M4, M5, M6, M7, M8, M9, P1, P2, P3
double precision ::xi, xf, resultE1,resultE2,resultE3,errabs, errest, errrel
double precision ::resultE4,resultE5,resultE6, resultE7,resultE8,resultE9,resultP1, resultP2,
```

```
resultP3
double precision ::resultM1,resultM2,resultM3,resultM4,resultM5,resultM6, resultM7,
resultM8, resultM9
double precision ::sig, omg,det1, det2, etol, det, delta, delta1, sigp, sigs,rg, rcm,ds
double precision ::CM(LDA,LDA), fact(ldfact,ldfact), ipvt(mv), dett(mu), bt, x, PVM, RO,
g, gro, beta
character*35 :: flname
integer :: irule, nout
integer :: m, q, k, l, lq,lx,ly,lz,lw ,i,j, t, lx1, lx2, ly1, ly2, lw1, lw2, lz1,lz2
integer :: ltt, lflag, li, lj, j2, il
external:: AX, BX, CX, FX, dFX, E1, E2, E3, E4, E5, E6, E7, E8, E9
external:: M1, M2, M3, M4, M5, M6, M7, M8, M9, P1, P2, P3
common/xxx/m,q,l,k,sig,omg
m=1
lq=2
rg=0.006d0
sigp=0.329d0
sigs=sigp+rg
sig=sigp-rg
dett(1)=0.0d0
ltt=1000000
lflag=2
etol=1.0d-4
delta=0.001
delta1=delta
il=100
omg=7.2921d-5
rcm=3480.0d0/6371

    call UMACH (2, nout)

    ! Set limits of integration
xi= 0.0d0
xf= 3480.0d0/6371
! Set error tolerances
errabs= 1.0d-8
! errest= 1.0d-8
errrel= 0.0d0
! Parameter for non-oscillatory function
irule = 6

    call PREM(x, PVM, RO, g, gro, beta)
bt=beta

    OPEN(unit=10, file=flname, access= 'append')
```

```
write(10,13) bt
write(10,11) LM,N,lq, m
write(*,11) LM,N,lq, m
close(10)
do j2=1, ltt
do li=1, LDA
do lj=1, LDA
CM(li,lj)=0.d0
end do
end do
do t=1, N
if (lq.eq.1 .and. m.eq.0) then
q=2*t-1 else if (lq.eq.1 .and. m.eq.1) then
q=2*t-1
else if (lq.eq.2 .and. m.eq.0) then
q=2*(t-1)
else if (lq.eq.2 .and. m.eq.1) then
q=2*t
end if
do k=1, LM
lx=LM*(t-1)+k ! for momentum equation
lx1=LM*(N+t-1)+k ! for possion equation
lx2=LM*(2*N+t-1)+k ! for entropy equation
do l=1, LM
ly=LM*(t-1)+l
ly1=LM*(N+t-1)+l
ly2=LM*(2*N+t-1)+l
lz=LM*(t)+l
lz1=LM*(N+t)+l
lz2=LM*(2*N+t)+l
lw=LM*(t-2)+l
lw1=LM*(N+t-2)+l
lw2=LM*(2*N+t-2)+l

     !coefficients of matrix
if (ly.le.0.or.ly.gt.(LM*N)) goto 101
call DQDAG (M1, xi, xf, errabs, errrel, irule, resultM1, errest)
CM(lx,ly)=resultM1
call DQDAG (P1, xi, xf, errabs, errrel, irule, resultP1, errest)
CM(lx1,ly)=resultP1
if (ne.eq.3)then
call DQDAG (E1, xi, xf, errabs, errrel, irule, resultE1, errest)
CM(lx2,ly)=resultE1
end if
101 continue
```

```
    if (ly1.lt.(LM*N).or.ly1.gt.(2*LM*N)) goto 102
call DQDAG (M2, xi, xf, errabs, errrel, irule, resultM2, errest)
CM(lx,ly1)=resultM2
call DQDAG (P2, xi, xf, errabs, errrel, irule, resultP2, errest)
ds=(q+1)*FX(l-1,0,rcm)*(FX(k-1,0,rcm)*rcm)
CM(lx1,ly1)=resultP2+ds
call DQDAG (E2, xi, xf, errabs, errrel, irule, resultE2, errest)
CM(lx2,ly1)=resultE2
102 continue
    if (ly2.lt.(2*LM*N).or.ly2.gt.(3*LM*N)) goto 103
call DQDAG (M3, xi, xf, errabs, errrel, irule, resultM3, errest)
CM(lx,ly2)=resultM3
call DQDAG (P3, xi, xf, errabs, errrel, irule, resultP3, errest)
CM(lx1,ly2)=resultP3
call DQDAG (E3, xi, xf, errabs, errrel, irule, resultE3, errest)
CM(lx2,ly2)=resultE3
103 continue

    if (lz.gt.(LM*N)) goto 104
call DQDAG (M4, xi, xf, errabs, errrel, irule, resultM4, errest)
CM(lx,lz)=resultM4
CM(lx1,lz)=0.0d0
call DQDAG (E4, xi, xf, errabs, errrel, irule, resultE4, errest)
CM(lx2,lz)=resultE4
104 continue

    if (lz1.gt.(2*LM*N)) goto 105
call DQDAG (M5, xi, xf, errabs, errrel, irule, resultM5, errest)
CM(lx,lz1)=resultM5
CM(lx1,lz1)=0.0d0
call DQDAG (E5, xi, xf, errabs, errrel, irule, resultE5, errest)
CM(lx2,lz1)=resultE5
105 continue

    if (lz2.gt.(3*LM*N)) goto 106
call DQDAG (M6, xi, xf, errabs, errrel, irule, resultM6, errest)
CM(lx,lz2)=resultM6
CM(lx1,lz2)=0.0d0
call DQDAG (E6, xi, xf, errabs, errrel, irule, resultE6, errest)
CM(lx2,lz2)=resultE6
106 continue

    if (lw.le.0) goto 107
call DQDAG (M7, xi, xf, errabs, errrel, irule, resultM7, errest)
CM(lx,lw)=resultM7
```

```
CM(lx1,lw)=0.0d0
if (ne.eq.3)then
call DQDAG (E7, xi, xf, errabs, errrel, irule, resultE7, errest)
CM(lx2,lw)=resultE7
107 continue

    if (lw1.le.(LM*N).or.lw1.gt.(2*LM*N)) goto 108
call DQDAG (M8, xi, xf, errabs, errrel, irule, resultM8, errest)
CM(lx,lw1)=resultM8
CM(lx1,lw1)=0.0d0
call DQDAG (E8, xi, xf, errabs, errrel, irule, resultE8, errest)
CM(lx2,lw1)=resultE8
108 continue

    if (lw2.le.(2*LM*N).or.lw2.gt.(3*LM*N)) goto 109
call DQDAG (M9, xi, xf, errabs, errrel, irule, resultM9, errest)
CM(lx,lw2)=resultM9
CM(lx1,lw2)=0.0d0
call DQDAG (E9, xi, xf, errabs, errrel, irule, resultE9, errest)
CM(lx2,lw2)=resultE9
109 continue
end do
end do
end do

    if (m.eq.0.and.lq.eq.2) CM(1,1)=1.0d0
call DLFTRG(mv, CM, LDA, fact, ldfact, ipvt)
call DLFDRG(mv, fact, ldfact, ipvt, det1, det2)
call UMACH (2, nout)
dett(2)=det1
det=dett(1)*dett(2)

    if (lflag.eq.2) det=1.d0
if (det.lt.0) then
delta=-delta/3
if(dabs(delta).le.(etol)) then
OPEN(unit=10, file=flname, access= 'append')
write (10,12) sig
write (*,12) sig
close(10)
delta=delta1
sig=sig+delta/7
lflag=2
if (il.lt.60) goto 15
goto 33
```

```
end if
end if

     lflag=1
33 continue
dett(1)=dett(2)
115 continue
sig=sig+delta
print*, sig,det if ((sig).gt.sigs) return
end do
15 continue
11 format(' LM=', i3,' ', ' ','N=', i3, ' ','lq=',i2, ' ','m=',i3,' ', 'ne=',i2)
12 format (' ',,'sig=', f16.9)
13 format (' ',,'bt=', f16.9)
end subroutine poincare
!-------------------------------------------------------
!momentum equation
!-------------------------------------------------------
double precision function M1(x) !chi
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, dfk, a, sig, omg
double precision :: s1, s2, s3,FX,dFX,BX
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)
dfk= a*dFX(k-1,0,x)
s1=0.d0
s2=0.d0
s3=0.d0
M1=0.0d0
s1=(sig**2-1.0d0/3)*dfl*dfk*x*x+m*sig*(fk*dfl+fl*dfk)*x
s2=fl*fk*(m*m-q*(q+1)*(2.0d0/3-sig**2)+m*sig)
s3=-BX(m,q)*(2*dfl*dfk*x*x/3+(fk*dfl+fl*dfk)*x-(dfloat(2*q*(q+1))/3-2.0d0)*fl*fk)
M1=s1+s2+s3
end function M1
!-------------------------------------------------------
double precision function M2(x) !V1
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, dfk, a, sig, omg
```

```
double precision :: s1, s2, s3,FX,dFX,BX
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)
dfk= a*dFX(k-1,0,x)
s1=0.d0
s2=0.d0
s3=0.d0
M2=0.0d0
s1=(sig**2-1.0d0/3)*dfl*dfk*x*x+m*sig*(fk*dfl+fl*dfk)*x
s2=fl*fk*(m*m-q*(q+1)*(2.0d0/3-sig**2)+m*sig)
s3=-BX(m,q)*(2*dfl*dfk*x*x/3+(fk*dfl+fl*dfk)*x-(dfloat(2*q*(q+1))/3-2.0d0)*fl*fk)
M2=-(s1+s2+s3)
end function M2
!—————————————————————————————
double precision function M3(x) !zeta
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, re, PVM, RO, g, gro, beta, sig, omg
double precision ::a, dfk, s1, s2, s3, g0, gr,FX,dFX, BX
common/xxx/m,q,l,k,sig,omg
re= 6371.0d3
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfk= a*dFX(k-1,0,x)
s1=0.d0
s2=0.d0
s3=0.d0
M3=0.0d0
call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
s1=beta*fl*(-sig**2*gr+gr/3)*dfk*x*x
s2=-(m*sig*beta*gr*x+sig**2*(sig**2-1.0d0)*x*x)*fl*fk
s3=BX(m,q)*beta*gr*(2*fl*dfk*x*x/3+fl*fk*x)
M3=-(s1+s2+s3)
end function M3
!—————————————————————————————
double precision function M4(x) !chi
implicit none
double precision, intent(in)::x
```

```
integer :: m, q, l, k
double precision ::fl, fk, sig, omg ,FX,dFX,AX
double precision ::s1, s2, dfl, dfk, a
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)
dfk= a*dFX(k-1,0,x)
s1=0.d0
s2=0.d0
M4=0.0d0
s1=dfl*dfk*x**2+(q*(fl*dfk-fk*dfl)+3*fl*dfk)*x
s2=-q*(q+3)*fl*fk
M4=-2*AX(m,q+2)*(s1+s2)/3
end function M4
!-------------------------------------------------------
double precision function M5(x) !V1
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, sig, omg,FX,dFX,AX
double precision :: s1, s2, dfl, dfk, a
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)
dfk= a*dFX(k-1,0,x)
s1=0.d0
s2=0.d0
s1=dfl*dfk*x**2+(q*(fl*dfk-fk*dfl)+3*fl*dfk)*x
s2=-q*(q+3)*fl*fk
M5=2*AX(m,q+2)*(s1+s2)/3
end function M5
!-------------------------------------------------------
double precision function M6(x) !zeta
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, sig, omg, PVM, RO, g, gro, beta, re
double precision ::a, dfk,FX,dFX,g0,gr,AX
common/xxx/m,q,l,k,sig,omg
re= 6371.0d3
a=2*6371.0d0/3480
```

```
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfk= a*dFX(k-1,0,x)
M6=0.0d0

     call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
M6=-AX(m,q+2)*beta*gr*(2*fl*dfk*x*x/3+(2.d0-2*dfloat(q+3)/3)*x*fl*fk)
end function M6
!———————————————————————————————————
double precision function M7(x) !chi
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, sig, omg,FX,dFX,CX
double precision :: s1, s2, dfl, dfk, a
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)
dfk= a*dFX(k-1,0,x)
s1=0.d0
s2=0.d0
M7=0.0d0
s1=dfl*dfk*x**2-((q-2)*(fl*dfk-fk*dfl)-3*fk*dfl)*x
s2=-(q+1)*(q-2)*fl*fk
M7=-2*CX(m,q-2)*(s1+s2)/3
end function M7
!———————————————————————————————————
double precision function M8(x) !V1
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, sig ,FX,dFX,CX
double precision :: s1, s2, dfl, dfk, a, omg
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)
dfk= a*dFX(k-1,0,x)
s1=0.d0
s2=0.d0
```

```
M8=0.0d0
s1=dfl*dfk*x**2-((q-2)*(fl*dfk-fk*dfl)-3*fk*dfl)*x
s2=-(q+1)*(q-2)*fl*fk
M8=2*CX(m,q-2)*(s1+s2)/3
end function M8
!'-------------------------------------------------
double precision function M9(x) !zeta
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, sig, omg, PVM, RO, g, gro, beta, re
double precision ::a, dfk,FX,dFX,CX ,g0 ,gr
common/xxx/m,q,l,k,sig,omg
re= 6371.0d3
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfk= a*dFX(k-1,0,x)

    call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
M9=0.0d0
M9=-CX(m,q-2)*beta*gr*(2*fl*dfk*x*x/3+(2.d0+2*dfloat(q-2)/3)*x*fl*fk)
end function M9
!-------------------------------------------------
!Possion equation
!-------------------------------------------------
double precision function P1(x) !chi
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, pi, GO, PVM, RO, g, gro, beta, sig, omg, re
double precision ::V, FX
common/xxx/m,q,l,k,sig,omg
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
pi= dacos(-1.0d0)
GO = 6.6690941D-11
re= 6371.0d3
P1=0.0d0

    call PREM(x, PVM, RO, g, gro, beta)
V=PVM/(2*re*omg)
P1= -4*pi*(GO/4/omg/omg)*RO*(1.0d0-beta)*fl*fk*x*x/V/V
```

```
end function P1
!———————————————————————————————
double precision function P2(x) !V1
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, dfk, pi, a, sig, omg, re,FX,dFX
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)
dfk= a*dFX(k-1,0,x)
pi= dacos(-1.0d0)
re= 6371.0d3
P2=0.0d0
P2=dfk*dfl*x*x+(q*(q+1))*fl*fk
end function P2
!'———————————————————————————————
double precision function P3(x) !zeta
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, pi, GO, PVM, RO, g, gro, beta, sig, omg, re
double precision ::FX,dFX
common/xxx/m,q,l,k,sig,omg
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
pi= dacos(-1.0d0)
GO = 6.6690941D-11
re= 6371.0d3
P3=0.0d0

     call PREM(x, PVM, RO, g, gro, beta)
P3= 4*pi*(GO/4/omg/omg)*RO*beta*fk*fl*x*x
end function P3
!———————————————————————————————
!entropy equation
!———————————————————————————————
double precision function E1(x) !chi
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, a, sig, omg, PVM, RO, g, gro, beta,re, g0,gr
double precision :: s1, s2, s3,FX,dFX,BX
```

```
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)

    call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
s1=0.0d0
s2=0.0d0
s3=0.0d0
E1=0.0d0
s1=fk*(-sig**2*gr+gr/3)*dfl*x*x
s2=-(m*sig*gr*x+sig**2*(sig**2-1.0d0)*x*x)*fl*fk
s3=BX(m,q)*(2*gr*fk*dfl*x*x/3+fl*fk*gr*x)
E1=(s1+s2+s3)
end function E1
!-----------------------------------------------------
double precision function E2(x) !V1
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, a, sig, omg, PVM, RO, g, gro, beta, g0,gr, re
double precision :: s1, s2, s3,FX,dFX,BX
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)
s1=0.d0
s2=0.d0
s3=0.d0
E2=0.0d0

    call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
s1=fk*(-sig**2*gr+gr/3)*dfl*x*x
s2=-m*sig*gr*fl*fk*x
s3=BX(m,q)*(2*gr*fk*dfl*x*x/3+fl*fk*gr*x)
E2=-(s1+s2+s3)
end function E2
```

```
!———————————————————————————————————
double precision function E3(x) !zeta
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, re, PVM, RO, g, gro, beta, sig, omg, g0, gr1, V
double precision FX,BX
common/xxx/m,q,l,k,sig,omg
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)

    call PREM(x, PVM, RO, g, gro, beta)
V=PVM/2/re/omg
g0=g/(4*re*omg**2)
gr1= g0**2-g0*x/3
E3=0.0d0
E3=-fl*fk*(V*V*sig**2*(sig**2-1.0d0)+beta*(sig**2*g0**2-gr1/3-2*BX(m,q)*gr1/3))*x*x
end function E3
!———————————————————————————————————
double precision function E4(x) !chi
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, a, sig, omg, PVM, RO, g, gro, beta, g0,gr, re
double precision:: FX,dFX,AX
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)

    call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
E4=0.0d0
E4=AX(m,q+2)*(2*gr/3*(fk*dfl*x*x+(q+3)*fl*fk*x))
end function E4
!———————————————————————————————————
double precision function E5(x) !V1
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, a, sig, omg, PVM, RO, g, gro, beta, g0,gr, re
```

```
double precision :: FX,dFX,AX
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)

     call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
E5=0.0d0
E5=-AX(m,q+2)*(2*gr/3*(fk*dfl*x*x+(q+3)*fl*fk*x))
end function E5
!———————————————————————————————————
double precision function E6(x) !zeta
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, sig, omg, PVM, RO, g, gro, beta,re, g0, gr1
double precision :: FX,AX
common/xxx/m,q,l,k,sig,omg
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)

     call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr1= g0**2-g0*x/3
E6=0.0d0
E6=AX(m,q+2)*2*beta*gr1*fl*fk*x*x/3
end function E6
!———————————————————————————————————
double precision function E7(x) !chi
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, a, sig, omg, PVM, RO, g, gro, beta, g0,gr , re
double precision:: FX,dFX,CX
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)
```

```
    call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
E7=0.0d0
E7=CX(m,q-2)*(2*gr/3*(fk*dfl*x*x-(q-2)*fl*fk*x))
end function E7
!-----------------------------------------------------
double precision function E8(x) !V1
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, a, sig, omg, PVM, RO, g, gro, beta, g0,gr, re
double precision :: FX,dFX,CX
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)

    call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
E8=0.0d0
E8=-CX(m,q-2)*(2*gr/3*(fk*dfl*x*x-(q-2)*fl*fk*x))
end function E8
!-----------------------------------------------------
double precision function E9(x) !zeta
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, sig, omg, PVM, RO, g, gro, beta,re, g0, gr1
double precision :: FX,dFX,CX
common/xxx/m,q,l,k,sig,omg
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)

    call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr1= g0**2-g0*x/3
E9=0.0d0
E9=CX(m,q-2)*2*beta*gr1*fl*fk*x*x/3
end function E9
```

```
!———————————————————————————-
double precision function AX(m, q)
implicit none
integer, intent(in)::m, q
AX=dfloat(3*(q+m)*(q+m-1))/2/(2*q+1)/(2*q-1)
end function AX
!———————————————————————————————————-
double precision function BX(m, q)
implicit none
integer, intent(in)::m, q
BX=dfloat(q*(q+1)- 3*m*m)/(2*q+3)/(2*q-1)
end function BX
!———————————————————————————————————-
double precision function CX(m, q)
implicit none
integer, intent(in)::m, q
CX=dfloat(3*(q+2-m)*(q+1-m))/2/(2*q+3)/(2*q+1)
end function CX
!———————————————————————————————————-
double precision function FX(l,m,y) ! Associated Legendre Polynomials
implicit none
double precision, intent(in)::y
integer, intent(in)::l,m
double precision :: fact, pll, pmm, pmmp1, somx2, x
integer :: i, ll
x=2*y*6371/3480 -1.0d0
if (m.lt.0.or.m.gt.1.or.dabs(x).gt.1.) pause 'bad arguments of dFX'
pmm=1.0d0
if (m.gt.0) then
somx2=dsqrt((1.0d0-x)*(1.0d0+x))
fact= 1.0d0
do i=1, m
pmm=-pmm*fact*somx2 ! for P(m,m)
fact=fact+2.0d0
end do
end if

     if (l.eq.m) then
FX=pmm
else
pmmp1= x*(2*m+1)*pmm ! for P(m,m+1)
if (l.eq.m+1)then
FX= pmmp1
else
do ll=m+2, l
```

```
pll= (x*(2*ll-1)*pmmp1-(ll+m-1)*pmm)/(ll-m)
pmm=pmmp1
pmmp1=pll
end do
FX= pll
end if
end if
end function FX
!———————————————————————————————
double precision function dFX(l,m,y) ! Derivative of Legendre Polynomials
implicit none
double precision, intent(in)::y
integer, intent(in)::l,m
double precision :: fact, pll, pmm, pmmp1, somx2, x
double precision :: dpll, dpmm, dpmmp1, dsomx2, FX
integer :: i, ll
x=2*y*6371.d0/3480 -1.0d0
if (m.lt.0.or.m.gt.1.or.dabs(x).gt.1.) pause 'bad arguments of FX'
pmm=1.0d0
dpmm=0.0d0
if (m.gt.0) then
somx2=dsqrt((1.0d0-x)*(1.0d0+x))
dsomx2=-x/dsqrt((1.0d0-x)*(1.0d0+x))
fact= 1.0d0
do i=1, m
pmm=-pmm*fact*somx2 ! for P(m,m)
dpmm=-dpmm*fact*somx2-pmm*fact*dsomx2
fact=fact+2.0d0
end do
end if

    if (l.eq.m) then
FX=pmm
dFX=dpmm
else
pmmp1= x*(2*m+1)*pmm ! for P(m,m+1)
dpmmp1=(2*m+1)*pmm+x*(2*m+1)*dpmm
if (l.eq.m+1)then
FX= pmmp
dFX= dpmmp1
else
do ll=m+2, l
pll= (x*(2*ll-1)*pmmp1-(ll+m-1)*pmm)/(ll-m)
dpll= ((2*ll-1)*pmmp1+x*(2*ll-1)*dpmmp1-(ll+m-1)*dpmm)/(ll-m)
pmm=pmmp1
```

```
dpmm=dpmmp1
pmmp1=pll
dpmmp1=dpll
end do
FX= pll
dFX= dpll
end if
end if
end function dFX
!————————————————————————————————
subroutine PREM (y,PVM, RO, g, gro, beta) !modified model
implicit none
integer, parameter::n=12, b=3
double precision ::c(b), d(n)
double precision ::y, re, PVM, RO, g, pi, GO, gro, beta, g1
integer ::j, i , imode
common/xix/imode
i=imode
!dbeta.= 0.000000000
if (i.eq.1) then
d(1)= 0.12476650D+05 !beta=1.d-6
d(2)= 0.00000000D+00
d(3)= -0.77408717D+04
d(4)= 0.51131860D-01
d(5)= -0.25078047D+04
d(6)= 0.16271224D+02
d(7)= -0.98071386D+03
d(8)= 0.54545089D+03
d(9)= -0.18448425D+04
d(10)= 0.27433988D+04
d(11)= -0.28131212D+04
d(12)= 0.12135027D+04
end if

    ! dbeta.= -0.00100000
if (i.eq.2) then
d(1)= 0.12478347D+05 !beta=-0.001
d(2)= 0.00000000D+00
d(3)= -0.77507123D+04
d(4)= 0.42523291D-01
d(5)= -0.25064629D+04
d(6)= 0.14671828D+02
d(7)= -0.96946550D+03
d(8)= 0.51130929D+03
d(9)= -0.17655030D+04
```

```
d(10)= 0.26346862D+04
d(11)= -0.27285598D+04
d(12)= 0.11866606D+04
end if

     !dbeta.= -0.00200000
if (i.eq.3)then
d(1)= 0.12480046D+05 !beta = -0.002
d(2)= 0.00000000D+00
d(3)= -0.77605710D+04
d(4)= 0.15008436D+00
d(5)= -0.25071639D+04
d(6)= 0.32641905D+02
d(7)= -0.10695994D+04
d(8)= 0.87488428D+03
d(9)= -0.25853357D+04
d(10)= 0.37762356D+04
d(11)= -0.36198489D+04
d(12)= 0.14871349D+04
end if

     !dbeta.= -0.00300000
if (i.eq.4) then
d(1)= 0.12482092D+05
d(2)= 0.00000000D+00
d(3)= -0.77743565D+04
d(4)= 0.10355821D+00
d(5)= -0.25000200D+04
d(6)= 0.24898556D+02
d(7)= -0.10199516D+04
d(8)= 0.71891891D+03
d(9)= -0.22307860D+04
d(10)= 0.32905272D+04
d(11)= -0.32417501D+04
d(12)= 0.13630128D+04
end if

     !dbeta.= -0.004000000
if (i.eq.5) then
d(1)= 0.12483793D+05
d(2)= 0.00000000D+00
d(3)= -0.77842298D+04
d(4)= 0.17363608D+00
d(5)= -0.25000255D+04
d(6)= 0.36508076D+02
```

```
d(7)= -0.10837720D+04
d(8)= 0.95261127D+03
d(9)= -0.27564340D+04
d(10)= 0.40222879D+04
d(11)= -0.38126688D+04
d(12)= 0.15558631D+04
end if

     !dbeta.= -0.005000000
if (i.eq.6) then
d(1)= 0.12485145D+05 ! beta=-0.005
d(2)= 0.00000000D+00
d(3)= -0.77901663D+04
d(4)= 0.50952918D-01
d(5)= -0.25017904D+04
d(6)= 0.16296668D+02
d(7)= -0.97174782D+03
d(8)= 0.54864426D+03
d(9)= -0.18507424D+04
d(10)= 0.27708191D+04
d(11)= -0.28435132D+04
d(12)= 0.12334900D+04
end if

     ! dbeta.= 0.001000000
if (i.eq.7) then
d(1)= 0.12474794D+05
d(2)= 0.00000000D+00
d(3)= -0.77308248D+04
d(4)= 0.13254268D+00
d(5)= -0.25103793D+04
d(6)= 0.29604811D+02
d(7)= -0.10576489D+04
d(8)= 0.81070115D+03
d(9)= -0.24401762D+04
d(10)= 0.35617495D+04
d(11)= -0.34462410D+04
d(12)= 0.14227414D+04
end if

     !dbeta.= 0.002000000
if (i.eq.8) then
d(1)= 0.12473097D+05
d(2)= 0.00000000D+00
d(3)= -0.77209895D+04
```

```
d(4)= 0.60311281D-01
d(5)= -0.25102243D+04
d(6)= 0.17672653D+02
d(7)= -0.99227547D+03
d(8)= 0.57153108D+03
d(9)= -0.19031905D+04
d(10)= 0.28157263D+04
d(11)= -0.28643778D+04
d(12)= 0.12258271D+04
end if

     !dbeta.= 0.005000000
if (i.eq.9) then
d(1)= 0.12468007D+05
d(2)= 0.00000000D+00
d(3)= -0.76914939D+04
d(4)= 0.11308380D+00
d(5)= -0.25146109D+04
d(6)= 0.26302777D+02
d(7)= -0.10461213D+04
d(8)= 0.74177777D+03
d(9)= -0.22856964D+04
d(10)= 0.33318854D+04
d(11)= -0.32598077D+04
d(12)= 0.13523472D+04
end if
c(1)= 10.6776d3
c(2)= 0.0d0
c(3)= -8.7572d3
PVM = 0.00d0
RO = 0.00d0
g= 0.00d0
g1=0.00d0
gro= 0.00d0
beta= 0.00d0
re= 6371.0d3 ! earth radius
pi= dacos(-1.0d0)
GO= 6.6690941D-11 !gravitational constant (PREM)

     do j=1, b
PVM = PVM + c(j)*(y**(j-1)) !speed
end do

     do 100 j=1, n
RO = RO + d(j)*(y**(j-1)) ! density
```

```
g = g+ 4*pi*GO*re*d(j)/(j+2)*y**j !gravity
g1 = g1+ 4*pi*GO*re*d(j)/(j+2)*y**(j-1) ! g=graviry/y
if (j .le. 2) goto 100
gro= gro +(j-1)*d(j)*(y**(j-3))/re !gro=gradient of density/y
100 continue
beta = 1.0d0+ (PVM*PVM/(RO*g1))*gro
end subroutine PREM
!————————————————————————————————————
```

C.3 The Code for the Eigenfunctions of Inertial Modes

After finding the non-dimensional frequencies of the inertial modes for a compressible and modified spherical core models, this FORTRAN code generates the data for the eigenfunctions of these modes as below:

```
    program trPD
implicit none
integer :: LM,N,imode
integer:: ne, LDA, ldfact
common/xix/imode

    ne=3
write(*,*) 'Enter LM'
read(*,*) LM
write(*,*) 'Enter N'
read(*,*) N
write(*,*) 'Enter imode, 1 for b=-0.0, 2 for b=-0.001... 6 for b=-0.005'
read(*,*) imode
LDA=ne*LM*N
ldfact=LDA
call poincare(LM,N,LDA,ldfact,ne)

    end program trPD
!————————————————————————————————————
subroutine poincare(LM,N,LDA,ldfact,ne)
implicit none
integer :: LM, N, ne, LDA, ldfact, ndel
double precision :: CM(LDA,LDA), B(LDA,LDA), C(ldfact), D(ldfact)
double precision :: sig, omg, pi, del1, del2, phi, tht, r1, r2, rc, ri, ik, r
double precision :: x, y, ux, uy, v, ur, ut, up, chi, V1, zeta, phi1
character*40 :: flname
integer :: m, q, k, l, lq, i, j, ipath
common/xxx/m,q,l,k,sig,omg
common/yy/lq
```

```
      m=1
lq=1
sig=0.776d0
omg=7.2921d-5
ndel=200
ri=0.0d0
rc=3480.0d0
pi=dacos(-1.0d0)
del1=pi/ndel
del2=1.0d0/ndel
r1=ri/6371
r2=rc/6371
phi=0.0d0

      call matrix(CM, LM, N, ne, LDA)
do i=1, LDA
do j=1, LDA
B(i,j)=CM(i,j)
end do
C(i)=1.0d-15
end do

      ipath=1 !means the system BX=C is solved
call DLSARG(ldfact, B, LDA, C, ipath, D)

      OPEN(unit=20, file=flname)
do 1 ik=1, 2
if (ik.eq.1) phi=0.0d0
if (ik.eq.2) phi=pi
do 2 i=3, ndel
r=i*del2
if (r.gt.r2) goto 2
if (r.lt.r1) goto 2
do 3 j=1, ndel+1
tht=(j-1)*del1
if (tht.lt.(1.0d-6).or.tht.gt.(pi)) goto 3

      call displ(LM, N, LDA, D, ur, ut, up, tht, phi, r, chi, V1, zeta, phi1)
x=r*dsin(tht)*dcos(phi)
y=r*dcos(tht)
if (dabs(x).le.(1.0d-8).or.dabs(y).le.(1.0d-8)) goto 4
ux=(ur*dsin(tht)+ut*dcos(tht))*dcos(phi)
uy=(ur*dcos(tht)-ut*dsin(tht))
v=ux*ux+uy*uy
```

```
      print*, x*6371.0d0/3480,y*6371.0d0/3480, ux, uy, chi, V1, zeta, phi1

      write(20, 12) x*6371.0d0/3480,y*6371.0d0/3480, ux, uy, chi, V1, zeta, phi1
4 continue
3 continue
2 continue
1 continue
close (20)
12 format (8e16.6)
end subroutine poincare
!—————————————————————————————————
subroutine displ(LM, N, LDA, D, ur, ut, up, tht, phi, x, chi, V1, zeta, phi1)
implicit none
integer :: LM, N, LDA
double precision :: D(LDA), ur, ut, up, tht, phi, x1, x, sig, omg, f, f3, f2, chi, zeta, V1, phi1
double precision ::a, fl, fl1, dfl, z1, dl, y, s, PVM, RO, g, gro, beta, g0, gr, re, FX1, dFX1
integer ::m, q, k, l, lq, t, ly, ly1, ly2
external::FX1, dFX1
common/xxx/m,q,l,k,sig,omg
common/yy/lq

      re=6371.0d3
a=2*6371.0d0/3480
y=dcos(tht)
s=dsin(tht)
ur=0.0d0
ut=0.0d0
up=0.0d0
chi=0.0d0
V1=0.0d0
zeta=0.0d0
phi1=0.0d0
x1=2*x*6371.d0/3480 -1.0d0

      do t=1, N
if (lq.eq.1 .and. m.eq.0) then
q=2*t-1
else if (lq.eq.1 .and. m.eq.1) then
q=2*t-1
else if (lq.eq.2 .and. m.eq.0) then
q=2*(t-1)
else if (lq.eq.2 .and. m.eq.1) then
q=2*t
end if
```

```
    call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
z1=FX1(q,m,y)
dl=-s*dFX1(q,m,y)
f=0.0d0
f3=0.0d0
f2=0.0d0
fl=0.0d0
fl1=0.0d0
dfl=0.0d0

    do l=1, LM
ly=LM*(t-1)+l
ly1=LM*(N+t-1)+l
ly2=LM*(2*N+t-1)+l
f= f+D(ly)*FX1(l-1,0,x1)
f3= f3+D(ly1)*FX1(l-1,0,x1)
f2= f2+D(ly2)*FX1(l-1,0,x1)
fl= fl+(D(ly)-D(ly1))*FX1(l-1,0,x1)
fl1=fl1+D(ly2)*FX1(l-1,0,x1)
dfl=dfl+a*(D(ly)-D(ly1))*dFX1(l-1,0,x1)
end do

    ur=ur+(sig*sig-y*y)*dfl*z1+(y*s*dl+m*sig*z1)*fl/x-fl1*beta*gr*(-sig*sig+y*y)*z1
ut=ut+s*y*dfl*z1+((sig*sig-s*s)*dl+m*sig*y*z1/s)*fl/x+fl1*beta*gr*s*y*z1
chi=chi+f*z1*dcos(m*phi)
V1=V1+f3*z1*dcos(m*phi)
zeta=zeta+f2*z1*dcos(m*phi)
phi1=chi-V1
up=up
end do
ur=ur/sig/sig/(sig*sig-1.0d0)*dcos(m*phi)
ut=ut/sig/sig/(sig*sig-1.0d0)*dcos(m*phi)
end subroutine displ
!———————————————————————————— double precision function FX1(l,m,x)
! Associated Legendre Polynomials
implicit none
double precision, intent(in)::x
integer, intent(in)::l
double precision :: fact, pll, pmm, pmmp1, somx2
integer :: i, ll,m

    if (m.lt.0.or.m.gt.1.or.dabs(x).gt.1.) pause 'bad arguments of FX1'
```

```
pmm=1.0d0
if (m.gt.0) then
somx2=dsqrt((1.0d0-x)*(1.0d0+x))
fact= 1.0d0
do i=1, m
pmm=-pmm*fact*somx2 ! for P(m,m)
fact=fact+2.0d0
end do
end if

    if (l.eq.m) then
FX1=pmm
else
pmmp1= x*(2*m+1)*pmm ! for P(m,m+1)
if (l.eq.m+1)then
FX1= pmmp1
else
do ll=m+2, l
pll= (x*(2*ll-1)*pmmp1-(ll+m-1)*pmm)/(ll-m)
pmm=pmmp1
pmmp1=pll
end do
FX1= pll
end if
end if
end function FX1
!———————————————————————————————
double precision function dFX1(l,m,x) ! Derivative of Legendre Polynomials
implicit none
double precision, intent(in)::x
integer, intent(in)::l
double precision :: fact, pll, pmm, pmmp1, somx2, FX1
double precision :: dpll, dpmm, dpmmp1, dsomx2
integer :: i, ll ,m

    if (m.lt.0.or.m.gt.1.or.dabs(x).gt.1.) pause 'bad arguments of dFX1'
pmm=1.0d0
dpmm=0.0d0
if (m.gt.0) then
somx2=dsqrt((1.0d0-x)*(1.0d0+x))
dsomx2=-x/dsqrt((1.0d0-x)*(1.0d0+x))
fact= 1.0d0
do i=1, m
dpmm=-dpmm*fact*somx2-pmm*fact*dsomx2
pmm=-pmm*fact*somx2 ! for P(m,m)
```

```
fact=fact+2.0d0
end do
end if

     if (l.eq.m) then
FX1=pmm
dFX1=dpmm
else
pmmp1= x*(2*m+1)*pmm ! for P(m,m+1)
dpmmp1=(2*m+1)*pmm+x*(2*m+1)*dpmm
if (l.eq.m+1)then
FX1= pmmp1
dFX1= dpmmp1
else
do ll=m+2, l
dpll= ((2*ll-1)*pmmp1+x*(2*ll-1)*dpmmp1-(ll+m-1)*dpmm)/(ll-m)
pll= (x*(2*ll-1)*pmmp1-(ll+m-1)*pmm)/(ll-m)
pmm=pmmp1
dpmm=dpmmp1
pmmp1=pll
dpmmp1=dpll
end do
FX1= pll
dFX1= dpll
end if
end if
end function dFX1
!———————————————————————————————————
subroutine matrix(CM, LM, N, ne, LDA)
implicit none
integer :: LM, N, ne, LDA
double precision ::AX, BX, CX, FX, dFX, E1, E2, E3, E4, E5, E6, E7, E8, E9
double precision ::M1, M2, M3, M4, M5, M6, M7, M8, M9, P1, P2, P3
double precision ::xi, xf, resultE1,resultE2,resultE3,errabs, errest, errrel
double precision ::resultE4,resultE5,resultE6, resultE7,resultE8, resultE9, resultP1, resultP2,
resultP3
double precision ::resultM1,resultM2,resultM3,resultM4,resultM5, resultM6, resultM7, re-
sultM8, resultM9
double precision ::sig, omg, rcm, ds
double precision ::CM(LDA,LDA)
integer :: irule, nout
integer :: m, q, k, l, lq,lx,ly,lz,lw , t, lx1, lx2, ly1, ly2, lw1, lw2, lz1,lz2
integer :: li, lj
external:: AX, BX, CX, FX, dFX, E1, E2, E3, E4, E5, E6, E7, E8, E9
external:: M1, M2, M3, M4, M5, M6, M7, M8, M9, P1, P2, P3
```

```
common/xxx/m,q,l,k,sig,omg
common/yy/lq

     rcm=3480.0d0/6371
call UMACH (2, nout)
! Set limits of integration
xi= 0.0d0
xf= 3480.0d0/6371
! Set error tolerances
errabs= 1.0d-8
errrel= 0.0d0
! Parameter for non-oscillatory function
irule = 6

     !Initialization of matrix
do li=1, LDA
do lj=1, LDA
CM(li,lj)=0.d0
end do
end do

     if (m.eq.0.and.lq.eq.2) CM(1,1)=1.0d0
do t=1, N
if (lq.eq.1 .and. m.eq.0) then
q=2*t-1
else if (lq.eq.1 .and. m.eq.1) then
q=2*t-1
else if (lq.eq.2 .and. m.eq.0) then
q=2*(t-1)
else if (lq.eq.2 .and. m.eq.1) then
q=2*t
end if

     do k=1, LM
lx=LM*(t-1)+k ! for momentum equation
lx1=LM*(N+t-1)+k ! for possion equation
lx2=LM*(2*N+t-1)+k ! for entropy equation
do l=1, LM
ly=LM*(t-1)+l
ly1=LM*(N+t-1)+l
ly2=LM*(2*N+t-1)+l
lz=LM*(t)+l
lz1=LM*(N+t)+l
lz2=LM*(2*N+t)+l
lw=LM*(t-2)+l
```

```
lw1=LM*(N+t-2)+l
lw2=LM*(2*N+t-2)+l

     !coefficients of matrix
if (m.eq.0.and.lq.eq.2) CM(1,1)=1.0d0
if (ly.le.0.or.ly.gt.(LM*N)) goto 101
call DQDAG (M1, xi, xf, errabs, errrel, irule, resultM1, errest)
CM(lx,ly)=resultM1
call DQDAG (P1, xi, xf, errabs, errrel, irule, resultP1, errest)
CM(lx1,ly)=resultP1
if (ne.eq.3)then
call DQDAG (E1, xi, xf, errabs, errrel, irule, resultE1, errest)
CM(lx2,ly)=resultE1
end if
101 continue

     if (ly1.lt.(LM*N).or.ly1.gt.(2*LM*N)) goto 102
call DQDAG (M2, xi, xf, errabs, errrel, irule, resultM2, errest)
CM(lx,ly1)=resultM2
call DQDAG (P2, xi, xf, errabs, errrel, irule, resultP2, errest)
ds=(q+1)*FX(l-1,0,rcm)*(FX(k-1,0,rcm)*rcm)
CM(lx1,ly1)=resultP2+ds
if (ne.eq.3)then
call DQDAG (E2, xi, xf, errabs, errrel, irule, resultE2, errest)
CM(lx2,ly1)=resultE2
end if
102 continue

     if (ly2.lt.(2*LM*N).or.ly2.gt.(3*LM*N)) goto 103
call DQDAG (M3, xi, xf, errabs, errrel, irule, resultM3, errest)
CM(lx,ly2)=resultM3
call DQDAG (P3, xi, xf, errabs, errrel, irule, resultP3, errest)
CM(lx1,ly2)=resultP3
call DQDAG (E3, xi, xf, errabs, errrel, irule, resultE3, errest)
CM(lx2,ly2)=resultE3
103 continue

     if (lz.gt.(LM*N)) goto 104
call DQDAG (M4, xi, xf, errabs, errrel, irule, resultM4, errest)
CM(lx,lz)=resultM4
CM(lx1,lz)=0.0d0
if (ne.eq.3)then
call DQDAG (E4, xi, xf, errabs, errrel, irule, resultE4, errest)
CM(lx2,lz)=resultE4
end if
```

```
104 continue

    if (lz1.gt.(2*LM*N)) goto 105
call DQDAG (M5, xi, xf, errabs, errrel, irule, resultM5, errest)
CM(lx,lz1)=resultM5
CM(lx1,lz1)=0.0d0
call DQDAG (E5, xi, xf, errabs, errrel, irule, resultE5, errest)
CM(lx2,lz1)=resultE5
105 continue

    if (lz2.gt.(3*LM*N)) goto 106
call DQDAG (M6, xi, xf, errabs, errrel, irule, resultM6, errest)
CM(lx,lz2)=resultM6
CM(lx1,lz2)=0.0d0
call DQDAG (E6, xi, xf, errabs, errrel, irule, resultE6, errest)
CM(lx2,lz2)=resultE6
106 continue

    if (lw.le.0) goto 107
call DQDAG (M7, xi, xf, errabs, errrel, irule, resultM7, errest)
CM(lx,lw)=resultM7
CM(lx1,lw)=0.0d0
call DQDAG (E7, xi, xf, errabs, errrel, irule, resultE7, errest)
CM(lx2,lw)=resultE7
107 continue

    if (lw1.le.(LM*N).or.lw1.gt.(2*LM*N)) goto 108
call DQDAG (M8, xi, xf, errabs, errrel, irule, resultM8, errest)
CM(lx,lw1)=resultM8
CM(lx1,lw1)=0.0d0
call DQDAG (E8, xi, xf, errabs, errrel, irule, resultE8, errest)
CM(lx2,lw1)=resultE8
108 continue

    if (lw2.le.(2*LM*N).or.lw2.gt.(3*LM*N)) goto 109
call DQDAG (M9, xi, xf, errabs, errrel, irule, resultM9, errest)
CM(lx,lw2)=resultM9
CM(lx1,lw2)=0.0d0
call DQDAG (E9, xi, xf, errabs, errrel, irule, resultE9, errest)
CM(lx2,lw2)=resultE9
109 continue
end do
end do
end do
end subroutine matrix
```

```
!———————————————————————— ——————————-
!momentum equation
!———————————————————————————————————
double precision function M1(x) !chi
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, dfk, a, sig, omg
double precision :: s1, s2, s3,FX,dFX,BX
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)
dfk= a*dFX(k-1,0,x)
s1=0.d0
s2=0.d0
s3=0.d0
M1=0.0d0
s1=(sig**2-1.0d0/3)*dfl*dfk*x*x+m*sig*(fk*dfl+fl*dfk)*x
s2=fl*fk*(m*m-q*(q+1)*(2.0d0/3-sig**2)+m*sig)
s3=-BX(m,q)*(2*dfl*dfk*x*x/3+(fk*dfl+fl*dfk)*x-(dfloat(2*q*(q+1))/3-2.0d0)*fl*fk)
M1=s1+s2+s3
end function M1
!———————————————————————————————————-
double precision function M2(x) !V1
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, dfk, a, sig, omg
double precision :: s1, s2, s3,FX,dFX,BX
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)
dfk= a*dFX(k-1,0,x)
s1=0.d0
s2=0.d0
s3=0.d0
M2=0.0d0
s1=(sig**2-1.0d0/3)*dfl*dfk*x*x+m*sig*(fk*dfl+fl*dfk)*x
s2=fl*fk*(m*m-q*(q+1)*(2.0d0/3-sig**2)+m*sig)
s3=-BX(m,q)*(2*dfl*dfk*x*x/3+(fk*dfl+fl*dfk)*x-(dfloat(2*q*(q+1))/3-2.0d0)*fl*fk)
M2=-(s1+s2+s3)
```

```
end function M2
!———————————————————————————————
double precision function M3(x) !zeta
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, re, PVM, RO, g, gro, beta, sig, omg
double precision ::a, dfk, s1, s2, s3, g0, gr,FX,dFX, BX
common/xxx/m,q,l,k,sig,omg
re= 6371.0d3
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfk= a*dFX(k-1,0,x)
s1=0.d0
s2=0.d0
s3=0.d0
M3=0.0d0
call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
s1=beta*fl*(-sig**2*gr+gr/3)*dfk*x*x
s2=-(m*sig*beta*gr*x+sig**2*(sig**2-1.0d0)*x*x)*fl*fk
s3=BX(m,q)*beta*gr*(2*fl*dfk*x*x/3+fl*fk*x)
M3=-(s1+s2+s3)
end function M3
!———————————————————————————————
double precision function M4(x) !chi
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, sig, omg ,FX,dFX,AX
double precision ::s1, s2, dfl, dfk, a
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)
dfk= a*dFX(k-1,0,x)
s1=0.d0
s2=0.d0
M4=0.0d0
s1=dfl*dfk*x**2+(q*(fl*dfk-fk*dfl)+3*fl*dfk)*x
s2=-q*(q+3)*fl*fk
M4=-2*AX(m,q+2)*(s1+s2)/3
```

```
end function M4
!—————————————————————————————
double precision function M5(x) !V1
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, sig, omg,FX,dFX,AX
double precision :: s1, s2, dfl, dfk, a
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)
dfk= a*dFX(k-1,0,x)
s1=0.d0
s2=0.d0
s1=dfl*dfk*x**2+(q*(fl*dfk-fk*dfl)+3*fl*dfk)*x
s2=-q*(q+3)*fl*fk
M5=2*AX(m,q+2)*(s1+s2)/3
end function M5
!—————————————————————————————
double precision function M6(x) !zeta
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, sig, omg, PVM, RO, g, gro, beta, re
double precision ::a, dfk,FX,dFX,g0,gr,AX
common/xxx/m,q,l,k,sig,omg
re= 6371.0d3
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfk= a*dFX(k-1,0,x)
M6=0.0d0

    call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
M6=-AX(m,q+2)*beta*gr*(2*fl*dfk*x*x/3+(2.d0-2*dfloat(q+3)/3)*x*fl*fk)
end function M6
!—————————————————————————————
double precision function M7(x) !chi
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
```

```
double precision ::fl, fk, sig, omg,FX,dFX,CX
double precision :: s1, s2, dfl, dfk, a
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)
dfk= a*dFX(k-1,0,x)
s1=0.d0
s2=0.d0
M7=0.0d0
s1=dfl*dfk*x**2-((q-2)*(fl*dfk-fk*dfl)-3*fk*dfl)*x
s2=-(q+1)*(q-2)*fl*fk
M7=-2*CX(m,q-2)*(s1+s2)/3
end function M7
!———————————————————————————————
double precision function M8(x) !V1
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, sig ,FX,dFX,CX
double precision :: s1, s2, dfl, dfk, a, omg
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)
dfk= a*dFX(k-1,0,x)
s1=0.d0
s2=0.d0
M8=0.0d0
s1=dfl*dfk*x**2-((q-2)*(fl*dfk-fk*dfl)-3*fk*dfl)*x
s2=-(q+1)*(q-2)*fl*fk
M8=2*CX(m,q-2)*(s1+s2)/3
end function M8
!’———————————————————————————————
double precision function M9(x) !zeta
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, sig, omg, PVM, RO, g, gro, beta, re
double precision ::a, dfk,FX,dFX,CX ,g0 ,gr
common/xxx/m,q,l,k,sig,omg
re= 6371.0d3
a=2*6371.0d0/3480
```

```
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfk= a*dFX(k-1,0,x)

    call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
M9=0.0d0
M9=-CX(m,q-2)*beta*gr*(2*fl*dfk*x*x/3+(2.d0+2*dfloat(q-2)/3)*x*fl*fk)
end function M9
!———————————————————————————————
!Possion equation
!———————————————————————————————
double precision function P1(x) !chi
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, pi, GO, PVM, RO, g, gro, beta, sig, omg, re
double precision ::V, FX
common/xxx/m,q,l,k,sig,omg
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
pi= dacos(-1.0d0)
GO = 6.6690941D-11
re= 6371.0d3
P1=0.0d0

    call PREM(x, PVM, RO, g, gro, beta)
V=PVM/(2*re*omg)
P1= -4*pi*(GO/4/omg/omg)*RO*(1.0d0-beta)*fl*fk*x*x/V/V
end function P1
!———————————————————————————————
double precision function P2(x) !V1
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, dfk, pi, a, sig, omg, re,FX,dFX
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)
dfk= a*dFX(k-1,0,x)
pi= dacos(-1.0d0)
re= 6371.0d3
```

```
P2=0.0d0
P2=dfk*dfl*x*x+(q*(q+1))*fl*fk
end function P2
!'————————————————————————————————————
double precision function P3(x) !zeta
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, pi, GO, PVM, RO, g, gro, beta, sig, omg, re
double precision ::FX,dFX
common/xxx/m,q,l,k,sig,omg
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
pi= dacos(-1.0d0)
GO = 6.6690941D-11
re= 6371.0d3
P3=0.0d0

    call PREM(x, PVM, RO, g, gro, beta)
P3= 4*pi*(GO/4/omg/omg)*RO*beta*fk*fl*x*x
end function P3
!————————————————————————————————————
!entropy equation
!————————————————————————————————————
double precision function E1(x) !chi
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, a, sig, omg, PVM, RO, g, gro, beta,re, g0,gr
double precision :: s1, s2, s3,FX,dFX,BX
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)

    call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
s1=0.0d0
s2=0.0d0
s3=0.0d0
E1=0.0d0
s1=fk*(-sig**2*gr+gr/3)*dfl*x*x
```

```
s2=-(m*sig*gr*x+sig**2*(sig**2-1.0d0)*x*x)*fl*fk
s3=BX(m,q)*(2*gr*fk*dfl*x*x/3+fl*fk*gr*x)
E1=(s1+s2+s3)
end function E1
!-------------------------------------------------------
double precision function E2(x) !V1
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, a, sig, omg, PVM, RO, g, gro, beta, g0,gr, re
double precision :: s1, s2, s3,FX,dFX,BX
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)
s1=0.d0
s2=0.d0
s3=0.d0
E2=0.0d0

    call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
s1=fk*(-sig**2*gr+gr/3)*dfl*x*x
s2=-m*sig*gr*fl*fk*x
s3=BX(m,q)*(2*gr*fk*dfl*x*x/3+fl*fk*gr*x)
E2=-(s1+s2+s3)
end function E2
!-------------------------------------------------------
double precision function E3(x) !zeta
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, re, PVM, RO, g, gro, beta, sig, omg, g0, gr1, V
double precision FX,BX
common/xxx/m,q,l,k,sig,omg
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)

    call PREM(x, PVM, RO, g, gro, beta)
V=PVM/2/re/omg
g0=g/(4*re*omg**2)
```

```
gr1= g0**2-g0*x/3
E3=0.0d0
E3=-fl*fk*(V*V*sig**2*(sig**2-1.0d0)+beta*(sig**2*g0**2-gr1/3-2*BX(m,q)*gr1/3))*x*x
end function E3
!-------------------------------------------------------
double precision function E4(x) !chi
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, a, sig, omg, PVM, RO, g, gro, beta, g0,gr, re
double precision:: FX,dFX,AX
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)

    call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
E4=0.0d0
E4=AX(m,q+2)*(2*gr/3*(fk*dfl*x*x+(q+3)*fl*fk*x))
end function E4
!-------------------------------------------------------
double precision function E5(x) !V1
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, a, sig, omg, PVM, RO, g, gro, beta, g0,gr, re
double precision :: FX,dFX,AX
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)

    call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
E5=0.0d0
E5=-AX(m,q+2)*(2*gr/3*(fk*dfl*x*x+(q+3)*fl*fk*x))
end function E5
!-------------------------------------------------------
```

```
double precision function E6(x) !zeta
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, sig, omg, PVM, RO, g, gro, beta,re, g0, gr1
double precision :: FX,AX
common/xxx/m,q,l,k,sig,omg
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)

    call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr1= g0**2-g0*x/3
E6=0.0d0
E6=AX(m,q+2)*2*beta*gr1*fl*fk*x*x/3
end function E6
!———————————————————————————————
double precision function E7(x) !chi
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, a, sig, omg, PVM, RO, g, gro, beta, g0,gr , re
double precision:: FX,dFX,CX
common/xxx/m,q,l,k,sig,omg
a=2*6371.0d0/3480
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)

    call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
E7=0.0d0
E7=CX(m,q-2)*(2*gr/3*(fk*dfl*x*x-(q-2)*fl*fk*x))
end function E7
!———————————————————————————————
double precision function E8(x) !V1
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, dfl, a, sig, omg, PVM, RO, g, gro, beta, g0,gr, re
double precision :: FX,dFX,CX
common/xxx/m,q,l,k,sig,omg
```

```
a=2*6371.0d0/3480
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)
dfl= a*dFX(l-1,0,x)

     call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr=-g0+x/6
E8=0.0d0
E8=-CX(m,q-2)*(2*gr/3*(fk*dfl*x*x-(q-2)*fl*fk*x))
end function E8
!———————————————————————————————————
double precision function E9(x) !zeta
implicit none
double precision, intent(in)::x
integer :: m, q, l, k
double precision ::fl, fk, sig, omg, PVM, RO, g, gro, beta,re, g0, gr1
double precision :: FX,dFX,CX
common/xxx/m,q,l,k,sig,omg
re= 6371.0d3
fl= FX(l-1,0,x)
fk= FX(k-1,0,x)

     call PREM(x, PVM, RO, g, gro, beta)
g0=g/(4*re*omg**2)
gr1= g0**2-g0*x/3
E9=0.0d0
E9=CX(m,q-2)*2*beta*gr1*fl*fk*x*x/3
end function E9
!———————————————————————————————————-
double precision function AX(m, q)
implicit none
integer, intent(in)::m, q
AX=dfloat(3*(q+m)*(q+m-1))/2/(2*q+1)/(2*q-1)
end function AX
!———————————————————————————————————-
double precision function BX(m, q)
implicit none
integer, intent(in)::m, q
BX=dfloat(q*(q+1)- 3*m*m)/(2*q+3)/(2*q-1)
end function BX
!———————————————————————————————————-
double precision function CX(m, q)
implicit none
```

```
integer, intent(in)::m, q
CX=dfloat(3*(q+2-m)*(q+1-m))/2/(2*q+3)/(2*q+1)
end function CX
!———————————————————————————————————-
double precision function FX(l,m,y) ! Associated Legendre Polynomials
implicit none
double precision, intent(in)::y
integer, intent(in)::l,m
double precision :: fact, pll, pmm, pmmp1, somx2, x
integer :: i, ll
x=2*y*6371/3480 -1.0d0
if (m.lt.0.or.m.gt.1.or.dabs(x).gt.1.) pause 'bad arguments of dFX'
pmm=1.0d0
if (m.gt.0) then
somx2=dsqrt((1.0d0-x)*(1.0d0+x))
fact= 1.0d0
do i=1, m
pmm=-pmm*fact*somx2 ! for P(m,m)
fact=fact+2.0d0
end do
end if

     if (l.eq.m) then
FX=pmm
else
pmmp1= x*(2*m+1)*pmm ! for P(m,m+1)
if (l.eq.m+1)then
FX= pmmp1
else
do ll=m+2, l
pll= (x*(2*ll-1)*pmmp1-(ll+m-1)*pmm)/(ll-m)
pmm=pmmp1
pmmp1=pll
end do
FX= pll
end if
end if
end function FX
!———————————————————————————————————
double precision function dFX(l,m,y) ! Derivative of Legendre Polynomials
implicit none
double precision, intent(in)::y
integer, intent(in)::l,m
double precision :: fact, pll, pmm, pmmp1, somx2, x
double precision :: dpll, dpmm, dpmmp1, dsomx2, FX
```

```
integer :: i, ll
x=2*y*6371.d0/3480 -1.0d0
if (m.lt.0.or.m.gt.1.or.dabs(x).gt.1.) pause 'bad arguments of FX'
pmm=1.0d0
dpmm=0.0d0
if (m.gt.0) then
somx2=dsqrt((1.0d0-x)*(1.0d0+x))
dsomx2=-x/dsqrt((1.0d0-x)*(1.0d0+x))
fact= 1.0d0
do i=1, m
pmm=-pmm*fact*somx2 ! for P(m,m)
dpmm=-dpmm*fact*somx2-pmm*fact*dsomx2
fact=fact+2.0d0
end do
end if

     if (l.eq.m) then
FX=pmm
dFX=dpmm
else
pmmp1= x*(2*m+1)*pmm ! for P(m,m+1)
dpmmp1=(2*m+1)*pmm+x*(2*m+1)*dpmm
if (l.eq.m+1)then
FX= pmmp
dFX= dpmmp1
else
do ll=m+2, l
pll= (x*(2*ll-1)*pmmp1-(ll+m-1)*pmm)/(ll-m)
dpll= ((2*ll-1)*pmmp1+x*(2*ll-1)*dpmmp1-(ll+m-1)*dpmm)/(ll-m)
pmm=pmmp1
dpmm=dpmmp1
pmmp1=pll
dpmmp1=dpll
end do
FX= pll
dFX= dpll
end if
end if
end function dFX
!-------------------------------------------------------
subroutine PREM (y,PVM, RO, g, gro, beta) !modified model
implicit none
integer, parameter::n=12, b=3
double precision ::c(b), d(n)
double precision ::y, re, PVM, RO, g, pi, GO, gro, beta, g1
```

```
integer ::j, i , imode
common/xix/imode
i=imode
!dbeta.= 0.000000000
if (i.eq.1) then
d(1)= 0.12476650D+05 !beta=1.d-6
d(2)= 0.00000000D+00
d(3)= -0.77408717D+04
d(4)= 0.51131860D-01
d(5)= -0.25078047D+04
d(6)= 0.16271224D+02
d(7)= -0.98071386D+03
d(8)= 0.54545089D+03
d(9)= -0.18448425D+04
d(10)= 0.27433988D+04
d(11)= -0.28131212D+04
d(12)= 0.12135027D+04
end if

    ! dbeta.= -0.00100000
if (i.eq.2) then
d(1)= 0.12478347D+05 !beta=-0.001
d(2)= 0.00000000D+00
d(3)= -0.77507123D+04
d(4)= 0.42523291D-01
d(5)= -0.25064629D+04
d(6)= 0.14671828D+02
d(7)= -0.96946550D+03
d(8)= 0.51130929D+03
d(9)= -0.17655030D+04
d(10)= 0.26346862D+04
d(11)= -0.27285598D+04
d(12)= 0.11866606D+04
end if

    !dbeta.= -0.00200000
if (i.eq.3)then
d(1)= 0.12480046D+05 !beta = -0.002
d(2)= 0.00000000D+00
d(3)= -0.77605710D+04
d(4)= 0.15008436D+00
d(5)= -0.25071639D+04
d(6)= 0.32641905D+02
d(7)= -0.10695994D+04
d(8)= 0.87488428D+03
```

```
d(9)= -0.25853357D+04
d(10)= 0.37762356D+04
d(11)= -0.36198489D+04
d(12)= 0.14871349D+04
end if

     !dbeta.= -0.00300000
if (i.eq.4) then
d(1)= 0.12482092D+05
d(2)= 0.00000000D+00
d(3)= -0.77743565D+04
d(4)= 0.10355821D+00
d(5)= -0.25000200D+04
d(6)= 0.24898556D+02
d(7)= -0.10199516D+04
d(8)= 0.71891891D+03
d(9)= -0.22307860D+04
d(10)= 0.32905272D+04
d(11)= -0.32417501D+04
d(12)= 0.13630128D+04
end if

     !dbeta.= -0.004000000
if (i.eq.5) then
d(1)= 0.12483793D+05
d(2)= 0.00000000D+00
d(3)= -0.77842298D+04
d(4)= 0.17363608D+00
d(5)= -0.25000255D+04
d(6)= 0.36508076D+02
d(7)= -0.10837720D+04
d(8)= 0.95261127D+03
d(9)= -0.27564340D+04
d(10)= 0.40222879D+04
d(11)= -0.38126688D+04
d(12)= 0.15558631D+04
end if

     !dbeta.= -0.005000000
if (i.eq.6) then
d(1)= 0.12485145D+05 ! beta=-0.005
d(2)= 0.00000000D+00
d(3)= -0.77901663D+04
d(4)= 0.50952918D-01
d(5)= -0.25017904D+04
```

```
d(6)= 0.16296668D+02
d(7)= -0.97174782D+03
d(8)= 0.54864426D+03
d(9)= -0.18507424D+04
d(10)= 0.27708191D+04
d(11)= -0.28435132D+04
d(12)= 0.12334900D+04
end if

     ! dbeta.= 0.001000000
if (i.eq.7) then
d(1)= 0.12474794D+05
d(2)= 0.00000000D+00
d(3)= -0.77308248D+04
d(4)= 0.13254268D+00
d(5)= -0.25103793D+04
d(6)= 0.29604811D+02
d(7)= -0.10576489D+04
d(8)= 0.81070115D+03
d(9)= -0.24401762D+04
d(10)= 0.35617495D+04
d(11)= -0.34462410D+04
d(12)= 0.14227414D+04
end if

     !dbeta.= 0.002000000
if (i.eq.8) then
d(1)= 0.12473097D+05
d(2)= 0.00000000D+00
d(3)= -0.77209895D+04
d(4)= 0.60311281D-01
d(5)= -0.25102243D+04
d(6)= 0.17672653D+02
d(7)= -0.99227547D+03
d(8)= 0.57153108D+03
d(9)= -0.19031905D+04
d(10)= 0.28157263D+04
d(11)= -0.28643778D+04
d(12)= 0.12258271D+04
end if

     !dbeta.= 0.005000000
if (i.eq.9) then
d(1)= 0.12468007D+05
d(2)= 0.00000000D+00
```

```
d(3)= -0.76914939D+04
d(4)= 0.11308380D+00
d(5)= -0.25146109D+04
d(6)= 0.26302777D+02
d(7)= -0.10461213D+04
d(8)= 0.74177777D+03
d(9)= -0.22856964D+04
d(10)= 0.33318854D+04
d(11)= -0.32598077D+04
d(12)= 0.13523472D+04
end if
c(1)= 10.6776d3
c(2)= 0.0d0
c(3)= -8.7572d3
PVM = 0.00d0
RO = 0.00d0
g= 0.00d0
g1=0.00d0
gro= 0.00d0
beta= 0.00d0
re= 6371.0d3 ! earth radius
pi= dacos(-1.0d0)
GO= 6.6690941D-11 !gravitational constant (PREM)

    do j=1, b
PVM = PVM + c(j)*(y**(j-1)) !speed
end do

    do 100 j=1, n
RO = RO + d(j)*(y**(j-1)) ! density
g = g+ 4*pi*GO*re*d(j)/(j+2)*y**j !gravity
g1 = g1+ 4*pi*GO*re*d(j)/(j+2)*y**(j-1) ! g=graviry/y
if (j .le. 2) goto 100
gro= gro +(j-1)*d(j)*(y**(j-3))/re !gro=gradient of density/y
100 continue
beta = 1.0d0+ (PVM*PVM/(RO*g1))*gro
end subroutine PREM
```