

## Chapter 3

# Numerical methods

### 3.1 Introduction

Several finite element methods have been used, here, to solve the basic equations (2.1-2.4) numerically. General characteristics of these methods are i) subdivision of the model domain into finite elements ii) construction of basis functions using these elements iii) construction of the best approximation to the solution within the function space spanned by these basis functions. The solution found in this manner satisfies the equations in an average sense. This is the weak formulation of the problem, as opposed to the classical solution, that satisfies the equation in each point of the domain. Here the weak formulation known as Galerkin's principle is used. The general application of this method to boundary value problems, incorporation of natural and essential boundary conditions, and the use of the finite element method to construct the basis functions, are described by Cuvelier et al. [1986]. This chapter gives information on the methods that have been employed to solve the basic equations. First, a description is given of the solution of the time-dependent equations of temperature and composition. Next, the different formulations and methods to solve the equations of motion and of mass conservation (2.1-2.2) are considered. The chapter is concluded with some benchmark results.

### 3.2 Solution of the temperature equation

The heat equation is solved using a linear triangular element with 3 nodal points in the vertices of the elements. The temperature is approximated within an element by

$$T(x, y) = \sum_{j=1}^3 T_j \phi_j(x, y) \quad (3.1)$$

where  $T_j$  is the temperature in nodal point  $j$  and  $\phi_j$  is the linear shape function connected with this point. Application of the method of Galerkin to (2.3) leads to a system of ordinary differential equations

$$\mathbf{M}\dot{\mathbf{T}} + \mathbf{S}\mathbf{T} = \mathbf{f} \quad (3.2)$$

where  $\mathbf{T}$  is the vector of unknowns  $T_i$ , the dot denotes time derivative,  $\mathbf{M}$  is the heat capacity mass matrix,  $\mathbf{S}$  the stiffness matrix, associated with the advection and diffusion of heat, and  $\mathbf{f}$  the load vector representing the influence of volumetric heat generation and the inhomogeneous boundary conditions. The matrices and load vector are assembled elementwise using element matrices of the form

$$\mathbf{M}_{ij}^e = \int_e \int \phi_i \phi_j \, dx dy \quad (3.3a)$$

$$\mathbf{S}_{ij}^e = \int_e \int \kappa \nabla \phi_i \cdot \nabla \phi_j \, dx dy + \int_e \int \phi_j (\mathbf{u} \cdot \nabla) \phi_i \, dx dy \quad (3.3b)$$

$$\mathbf{f}_j^e = \int_e \int \frac{Q}{\rho c_p} \phi_j \, dx dy \quad (3.3c)$$

The effects of the inhomogeneous boundary conditions are later added to the load vector  $\mathbf{f}$ . The velocity components are obtained from a solution of the equations of motion and numerical values of  $\mathbf{u}^T = (v, w)$  are given in the nodal points. The integrals are calculated using Newton-Cotes quadrature, after expanding the velocity components in the linear shape functions

$$v(x, y) = \sum_{k=1}^3 v_k \phi_k(x, y) \quad , \quad w(x, y) = \sum_{k=1}^3 w_k \phi_k(x, y) \quad (3.4)$$

where  $k$  corresponds to the local numbering of the nodal points in the element. Newton-Cotes integration of (3.3a) reduces  $\mathbf{M}$  to a diagonal matrix. This process of lumping of the mass matrix is computationally convenient as it reduces the work necessary to solve (3.2) and it often increases the accuracy [Zienkiewicz, 1977].

In most of the calculations presented in this thesis, stable solutions to the heat equation could be obtained with the standard (Bubnov-)Galerkin method described above. Grid refinements were used in the few cases that oscillations were found in the temperature solution. Note that for the discretization of the temperature equation no upwinding techniques have been employed. Upwinding can be implemented in these finite element methods by application of the Petrov-Galerkin method [Heinrich et al., 1977]. This suffers from the disadvantage that artificial diffusivity is introduced and the solution may be inaccurate [Cuvelier et al., 1986]. It has been argued that streamline upwinding [Hughes & Brooks, 1982] greatly reduces cross-wind diffusivity. Upwind techniques have found standard application in mantle convection modelling to stabilize solutions of the advection dominated heat equation [Christensen, 1984a; Blankenbach et al., 1989]. A different, and more natural, approach to find stable solutions is the method of characteristics

[Pironneau, 1982], which is introduced in mantle convection modelling by Malevsky & Yuen [1991]. Disadvantage of this method is that high order shape-functions have to be used to reduce the amount of artificial diffusivity, introduced by the interpolation that is inherent in this method.

### 3.3 Transport equation for composition

There are several ways to represent the composition function  $\Gamma$ . In the field approach, the compositional differences are represented by a (continuous) function  $\Gamma$  and the transport equation (2-4) can be solved for  $\Gamma$  by the method of Galerkin as described above for the temperature. In this case the diffusion connected terms in the stiffness matrix (3.3b) are set to zero ( $\kappa = 0$ ). A disadvantage of the field approach is, that the applicability of the method of Galerkin -*sec*- to the hyperbolic equation (2-4) is not well defined. For particular cases it can be shown that convergence to the correct solution of hyperbolic equations cannot be obtained [Strang & Fix, 1973], and artificial diffusivity needs to be introduced, e.g. through discontinuous elements, streamline upwinding or the method of characteristics. Another disadvantage of the field approach is, that a discontinuous composition function, as is often encountered in geodynamical problems (e.g. the salt-sediment interface in modelling salt dynamics), can not be modelled accurately. The finite element approximation will smear out the interface over at least the length of one element by its linear approximation to the function. Transporting this approximation over a stationary grid will cause distortion and oscillations of the composition function near the interface. One way to circumvent this problem is to define a smoothing of the interface, for example through a hyperbolic tangent function, that transforms the discontinuity into a shape where the function drops smoothly from one discrete value to the other over the length of a few elements [e.g., Christensen, 1992].

A more popular way of representing compositional differences is the tracer method, in which discrete tracers are initially homogeneously distributed through the fluid layer it represents [Gurnis, 1986; Christensen, 1991; Zaleski & Julien, 1992; Weinberg & Schmeling, 1992]. One of the disadvantages of the method is that often a large number of tracer particles is necessary to discretize the composition function  $\Gamma$ . In situations where no large scale mixing of the fluids occurs, the interface between the fluids can efficiently be represented by a chain of markers (marker chain method). In both cases, the positions of the markers or tracers are determined by the initial distribution and the hyperbolic equations

$$\begin{aligned}\dot{x}_m(t) &= v(x_m(t), y_m(t), t) \\ \dot{y}_m(t) &= w(x_m(t), y_m(t), t)\end{aligned}\tag{3.5}$$

[Christensen & Yuen, 1984].

### 3.4 Time integration

The equations of motion and mass conservation (2.1-2.2) contain no explicit time-dependence because inertial forces can be neglected. The velocity and pressure fields are determined by the rheology, boundary conditions and the driving forces, which in the general geodynamical context are governed by the distribution of temperature and composition. Time-dependence is introduced in the equations through the heat and composition equations (2.3-2.4) and possibly the boundary conditions. This allows for following predictor-corrector scheme to solve the governing equations [Christensen, 1984a; Hansen & Ebel, 1988]:

*Specify initial distribution of  $T$  and  $\Gamma$*   
*Determine velocity field  $\mathbf{u}^0$  at time  $t$  by solving (2.1-2.2)*  
 $n = 0$   
*Repeat until end of time integration*  
     *Determine discrete time step  $\Delta t$*   
     *Predict temperature and composition distributions,  $T^{n+1(0)}$*   
     *and  $\Gamma^{n+1(0)}$ , at time  $t + \Delta t$*   
     *Predict velocity field  $\mathbf{u}^{n+1(0)}$  at time  $t + \Delta t$*   
  
 $i = 1$   
     *Repeat for each corrector step  $i$*   
         *Correct the temperature and composition distributions*  
          *$T^{n+1(i)}$  and  $\Gamma^{n+1(i)}$*   
         *Correct the velocity field  $\mathbf{u}^{n+1(i)}$*   
      $i = i + 1$   
 $n = n + 1$   
 $t = t + \Delta t$

To predict the temperature a semi-implicit Euler scheme is used to solve the system of equations (3.2)

$$\mathbf{M} \cdot \frac{\mathbf{T}^{n+1(0)} - \mathbf{T}^n}{\Delta t} + \mathbf{S}(\mathbf{u}^n) \cdot \mathbf{T}^{n+1(0)} = \mathbf{f}^{n+1} \quad (3.6)$$

after which the correction is calculated by a Crank-Nicholson step

$$\mathbf{M} \cdot \frac{\mathbf{T}^{n+1(i)} - \mathbf{T}^n}{\Delta t} + \frac{1}{2} \mathbf{S}(\mathbf{u}^n) \cdot \mathbf{T}^n + \frac{1}{2} \mathbf{S}(\mathbf{u}^{n+1(i-1)}) \cdot \mathbf{T}^{n+1(i)} = \mathbf{f}^{n+1} \quad (3.7)$$

for  $i = 1, 2, \dots$  [Hansen & Ebel, 1988]. A similar scheme can be used for  $\Gamma$  in the field approach. In the marker chain (or tracer) method the coordinates of the

markers can be updated by solving (3.5) using an explicit predictor

$$\begin{aligned} x_m^{(0)}(t + \Delta t) &= x_m(t) + v(t) \cdot \Delta t \\ y_m^{(0)}(t + \Delta t) &= y_m(t) + w(t) \cdot \Delta t \end{aligned} \quad (3.8)$$

(where  $v$  and  $w$  are the horizontal and vertical components of velocity  $\mathbf{u} = (v, w)^T$ ), and, after solution of the Stokes equation, a Crank-Nicholson corrector,

$$\begin{aligned} x_m^{(1)}(t + \Delta t) &= x_m(t) + \frac{\Delta t}{2} (v^{(0)}(t + \Delta t) + v(t)) \\ y_m^{(1)}(t + \Delta t) &= y_m(t) + \frac{\Delta t}{2} (w^{(0)}(t + \Delta t) + w(t)) \end{aligned} \quad (3.9)$$

The corrector step can be repeated to obtain higher accuracy. Ideally, that is in cases where the evolution of velocity  $\mathbf{u}$  is exactly known, the scheme is second order correct in time. The choice of time step  $\Delta t$  is based on the Courant criterion

$$\|v\Delta t/\Delta x, w\Delta t/\Delta y\|_\infty \leq 1 \quad (3.10)$$

where  $\Delta x$  and  $\Delta y$  are the (local) grid spacings in horizontal and vertical direction, resp. The choice of the explicit predictor in (3.8) requires this condition for stability. The accuracy of the time integration scheme depends strongly on the spatial structure of the velocity field and the coupling between the temperature and velocity field. For a strongly temperature dependent viscosity, errors in the temperature field will have a stronger effect than in isoviscous flows, which makes a smaller time step necessary. For a number of applications (both Rayleigh-Taylor instabilities and thermal convection experiments), it was found that limiting the time step to be half the Courant step (3.10) and performing only one corrector step gave a satisfactory trade-off between accuracy and computational cost.

Stationary results for purely thermal convection are obtained by explicitly neglecting time-dependence in (2.3) ( $\partial T/\partial t \equiv 0$ ), yielding

$$(\mathbf{u} \cdot \nabla)T = \kappa \nabla^2 T + Q/\rho c_p \quad (3.11)$$

In this case, the system of equations (2.1-2.2,3.11) is time independent but non-linearly coupled through velocity and temperature. Picard iteration, or successive substitution, is used to solve the system, starting from an initial temperature field. Using temperature dependent and/or non-Newtonian viscosity introduces additional non-linearities in the equations of motion and underrelaxation is needed to obtain convergence of the Picard iteration.

### 3.5 The equations of motion

For solving the equations of motion (2.2), combined with the continuity equation for an incompressible fluid (2.1), three different finite element methods will be

discussed. Two are based on the stream function formulation (see equation 2.22), that requires the use of higher order elements, as a consequence of the occurring fourth order derivatives. The first method to be discussed is the penalty function method, in which pressure is eliminated from the equations of motion (2.2) by a perturbation of the continuity equation. Following section summarizes some important points of chapters 7 and 8 of Cuvelier et al. [1986].

### 3.5.1 Penalty function method

First the solution of (2.1-2.2) for an isoviscous fluid will be considered. Equation (2.2) reduces to

$$\nabla p = \eta \nabla^2 \mathbf{u} + \rho g \hat{\mathbf{z}} \quad (3.12)$$

Pressure can only be defined up to an additive constant. The three equations (3.12) and (2.1) can be solved directly for  $\mathbf{u}$  and  $p$  using the method of Galerkin, but this is computationally inefficient. The discretized system of equations that has to be solved is large as a consequence of the three unknowns ( $v, w, p$ ) per nodal point. The pressure does not enter the incompressibility constraint and this introduces zeroes on the main diagonal of the stiffness matrix, which complicates the solution of the system considerably.

A way to circumvent these problems is to eliminate the pressure from (3.12) by perturbing the incompressibility constraint

$$\nabla \cdot \mathbf{u} = -\varepsilon p \quad (3.13)$$

where  $\varepsilon$  is small. This is the penalty function method;  $\varepsilon$  is called the penalty function parameter. Temam [1977] gives a proof for general boundary conditions that the solution to (3.12-3.13) approaches the solution to (2.1-2.2) as  $\varepsilon \rightarrow 0$ . After discretising the domain, approximating the velocity and pressure by

$$\bar{v}(x, y) = \sum_{i=1}^N v_i \phi_i, \quad \bar{w}(x, y) = \sum_{i=1}^N w_i \phi_i, \quad \bar{p}(x, y) = \sum_{i=1}^M p_i \psi_i \quad (3.14)$$

and applying the method of Galerkin, the pressure can be eliminated from the discretized equations of motion, yielding a system of equations

$$\mathbf{S}\mathbf{v} + \tau \mathbf{L}_x^T \mathbf{D}^{-1} \mathbf{L}_x \mathbf{v} = \mathbf{0} \quad (3.15a)$$

$$\mathbf{S}\mathbf{w} + \tau \mathbf{L}_y^T \mathbf{D}^{-1} \mathbf{L}_y \mathbf{w} = \mathbf{f} \quad (3.15b)$$

$$\mathbf{p} = \tau \mathbf{D}^{-1} [ \mathbf{L}_x \mathbf{v} + \mathbf{L}_y \mathbf{w} ] \quad (3.15c)$$

where  $\tau = \varepsilon^{-1}$ ,  $\mathbf{v}$  is the vector of unknowns  $v_i$ ,  $\mathbf{w}$  the vector of unknowns  $w_i$ ,  $\mathbf{p}$  the vector of unknowns  $p_i$ , and  $\mathbf{f}$  the load vector, defined by

$$\mathbf{f}_i = \int_{\Omega} \rho g \phi_i d\Omega \tag{3.15d}$$

The matrix components are given by

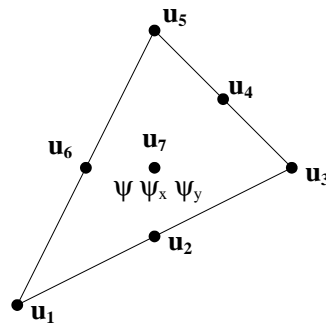
$$S_{ij} = \eta \int_{\Omega} \nabla \phi_i \cdot \nabla \phi_j d\Omega \tag{3.15e}$$

$$L_{z,ij} = - \int_{\Omega} \psi_j \frac{\partial \phi_i}{\partial z} d\Omega \quad z = x, y \tag{3.15f}$$

$$D_{ij} = \int_{\Omega} \psi_i \psi_j d\Omega \tag{3.15g}$$

The above equations are given for homogeneous boundary conditions. Inhomogeneous boundary conditions give a contribution to the load vector  $\mathbf{f}$ .

The shape functions  $\phi_i$  and  $\psi_i$  are constructed using the Crouzeix-Raviart ( $P_2^+ - P_1$ ) element (figure 3.1). Necessary conditions for an element are that basis functions for velocity are continuous over element boundaries and piecewise continuously differentiable. Pressure basis functions should be continuous within the element, and the order of these basis functions should be at least one lower than that of the velocity basis functions. The Crouzeix-Raviart ( $P_2^+ - P_1$ ) element approximates velocity as an extended quadratic function (using the velocity components in the seven nodal points depicted in figure 3.1 as degrees of freedom) and pressure as a linear function using the pressure and its derivatives to  $x$  and  $y$  in the barycenter. This element has pleasant properties as to the assembly of the matrix



**Figure 3.1** The Crouzeix-Raviart  $P_2^+ - P_1$  element.

$\mathbf{S} + \tau \mathbf{L}^T \mathbf{D}^{-1} \mathbf{L}$ : i) the matrix is symmetric and positive definite. ii) construction of the matrix can be carried out elementwise. iii) the velocity components in the barycenter and the pressure derivatives can be eliminated from the system of equations (i.e. they can be expressed as functions of the velocity components from the boundary nodal points only), reducing the number of degrees of freedom with 4. The resulting element has 6 nodal points for velocity and 1 nodal point for pressure (modified Crouzeix-Raviart element). iv) this elimination process makes the pressure matrix  $D$  diagonal, with simple inverse  $D^{-1}$ .

A more detailed discussion of this element can be found in [Cuvelier et al., 1986].

#### *Choice of penalty function parameter*

In the penalty function method, the incompressibility constraint is relaxed to make a more efficient solution of the equations of motion and the conservation of mass possible. To minimize the effects of finite compressibility it is desirable to choose  $\tau$  as high as possible. However, the penalty function matrix  $\mathbf{L}^T \mathbf{D}^{-1} \mathbf{L}$  is singular, and for large  $\tau$  the total matrix  $\mathbf{S} + \tau \mathbf{L}^T \mathbf{D}^{-1} \mathbf{L}$  will be singular. Cuvelier et al. [1986] suggest to choose  $\tau$  in the range

$$10^3 \leq \tau \leq 10^9 \quad (3.16)$$

Throughout the calculations  $\tau = 10^6$  has been used.

The choice of the penalty function parameter is less constrained than in the case of incompressible elasticity problems that are solved with the penalty function method [Hughes, 1987]. For compressible elasticity, the constitutive equation is given by

$$\underline{\sigma} = \lambda \nabla \cdot \underline{\mathbf{s}} \mathbf{I} + 2\mu \underline{\epsilon} \quad (3.17)$$

where  $\mu$  is the shear modulus,  $\underline{\mathbf{s}}$  is the displacement vector,  $\underline{\epsilon}$  the strain tensor and  $\mathbf{I}$  the Kronecker tensor. The Lamé parameter  $\lambda$  is given by

$$\lambda = \frac{2\nu\mu}{1-2\nu} \quad (3.18)$$

where  $\nu$  is Poisson's ratio. In the incompressible case ( $\nu = \frac{1}{2}$ ),  $\lambda$  is unbounded and a different formulation is necessary. The constitutive equation is then written

$$\underline{\sigma} = -p \mathbf{I} + 2\mu \underline{\epsilon} \quad (3.19)$$

where the pressure  $p$  is introduced as unknown. The additional equation to be solved is just the incompressibility constraint ( $\nabla \cdot \underline{\mathbf{s}} = 0$ ). In the mixed formulation, (3.19) is used as constitutive equation, where the parameter  $p$  is defined by

$$0 = \nabla \cdot \mathbf{s} + \frac{p}{\lambda} \quad (3.20)$$

This leads directly to the penalty function formulation for incompressible elasticity. For the compressible case the average normal stress equals  $-\sigma_{kk}/3 = -(\lambda + 2\mu/3)\nabla \cdot \mathbf{s}$ . Then, the parameter  $p$  from (3.20),  $p = -\lambda\nabla \cdot \mathbf{s}$ , can only be interpreted as pressure when  $\mu \ll \lambda$ . Therefore, in the penalty formulation  $\lambda$  has to be chosen finite, but large with respect to  $\mu$ . Numerical problems arise if  $\lambda$  is chosen too large. Hughes [1987] finds the range

$$10^7 \leq \frac{\lambda}{\mu} \leq 10^9 \quad (3.21)$$

to be effective.

The equations for (isotropic) incompressible elasticity and Stokes flow are identical, except for a different meaning of the parameters (the displacement  $\mathbf{s}$  is replaced by the velocity  $\mathbf{u}$  and  $\mu$  by the dynamic viscosity  $\eta$ ). This would suggest a choice of the penalty parameter  $\tau$  as a local function of the dynamic viscosity, which would have important consequences for the solution of non-isoviscous flows with the penalty function method. However, it can be shown for a fluid without bulk viscosity,  $\zeta = \lambda + 2\eta/3 = 0$ , that the average normal stress  $-\sigma_{kk}/3$  is always equal to the thermodynamical pressure [Malvern, 1969]. This relaxes the extra constraint on the choice of the penalty function parameter with respect to the dynamic viscosity. The bulk viscosity is related to rapid changes in volume (compared to molecular relaxation processes) and can be neglected in problems with slow deformation [Jarvis & McKenzie, 1980].

It has to be stressed that the viscosity should be scaled properly in a way that the average viscosity is around one. If the average viscosity is much less than one the total matrix  $\mathbf{S} + \tau \mathbf{L}^T \mathbf{D}^{-1} \mathbf{L}$  can be singular.

### *Practical implementation*

The Stokes equation, combined with the incompressibility constraint, and the temperature equation (2.1-2.3) are solved using the finite element package Sepran [Segal & Praagman, 1984]. Appendix A contains a description of some technical details of the implementation.

### *Generalized Newtonian fluids*

The rheological model of generalized Newtonian fluid has been used, where the viscosity depends on strain rate, temperature, pressure and composition. The coefficients of the viscous stress matrix (3.15e) have the form

$$S_{ij} = \int_{\Omega} \eta \nabla \phi_i \cdot \nabla \phi_j \, dx dy \quad (3.22)$$

The effective strain rate is a function of gradients of velocity which, in the finite element approximation, are continuous within an element, but in general discontinuous across element boundaries. Pressure is continuous over element boundaries (as it translates to depth in the Boussinesq approximation) and the same holds for the finite element approximation of temperature. Then, the approximation of the viscosity is in general continuous within the elements, but discontinuous across the element boundaries. The integral (3.22) is still integrable and simple tests using a power law creep model have correct convergence behaviour (error in velocity is a function of  $h^3$ , error in pressure a function of  $h^2$ ;  $h$  is the element size) and the element shows to be applicable to problems with generalized Newtonian fluids [Jaap van der Zanden, pers. comm., 1992]. A discussion on the effects of composition dependent viscosity is given in chapter 4.

For strain rate dependent viscosity the viscous stress matrix is a function of velocity and the discretized system (3.15) is non-linear. In time-dependent problems Picard sub-iteration is used to solve (3.15), keeping the buoyancy forces and boundary conditions fixed. The velocity field obtained at the previous step is used as initial estimate for the next step.

### 3.5.2 Stream function formulation

In the stream function formulation, the non-dimensional equation of motion for incompressible thermochemical convection is given by (2.22). The introduction of the stream function reduces the three equations for velocity and pressure to one scalar equation at the cost of fourth order derivatives. Conformity of element shape functions is often imposed to satisfy the basic convergence requirements [Hughes, 1987]. In the stream function formulation, the equation of motion is a fourth order partial differential equation and, for a finite element approximation to be conform, the basis functions should be twice continuously differentiable in each element and at least continuously differentiable throughout the computational domain  $\Omega$  [Cuvellier et al., 1986]. We consider the implementation of two methods that have been used to solve the equation of motion in the stream function formulation: the first uses a non-conforming type of element [Hansen & Ebel, 1984], the second uses bicubic splines as shape functions [Woidt, 1978; Kopitzke, 1979]. The notation will be used

$$Ra_0 = \frac{\rho_0 \alpha g \Delta T h^3}{\kappa_0 \eta_0}, \quad Ra(\Gamma) = \left[ \frac{\rho_0 + \Gamma \Delta \rho}{\rho_0} \right] Ra_0 \quad (3.23)$$

and a model of two homogeneous fluid layers with different compositional densities will be considered. The function  $\Gamma$  is a step function.

### *Non-conforming element*

The condition of continuity of the slope of shape functions across element boundaries ( $C_1$  continuity) is difficult to impose for conforming elements [Zienkewicz, 1977]. Here a non-conforming element is used, that, although it does not fulfil condition of  $C_1$  continuity, has found successful application in plate bending problems [Zienkewicz, 1977] and mantle convection modelling [Hansen & Ebel, 1984].

The model region  $\Omega$  is subdivided into rectangular elements with four nodal points in the vertices of the element. In each nodal point a vector of unknowns is defined with three components

$$\mathbf{a}_i = \left[ \bar{\psi}_i, \quad \left( \frac{\partial \bar{\psi}}{\partial y} \right)_i, \quad - \left( \frac{\partial \bar{\psi}}{\partial x} \right)_i \right]^T \quad (3.24)$$

The stream function in each element is approximated by

$$\bar{\psi} = \sum_{i=1}^4 \mathbf{N}_i \mathbf{a}_i \quad (3.25)$$

where  $\mathbf{N}_i$  is a vector of shape functions. Application of the method of Galerkin leads to a system of equations

$$\mathbf{S} \mathbf{a} = \mathbf{f} \quad (3.26)$$

where  $\mathbf{S}$  is the stiffness matrix resulting from the discretization of the differential operator and  $\mathbf{f}$  is the load vector, containing the contribution of the right hand side of (2.22) and of the non-homogeneous boundary conditions. The system is solved for the unknowns  $\mathbf{a}$ . The global matrix and vector are assembled elementwise after calculation of the element matrix

$$\begin{aligned} \mathbf{S}_{ij}^e = & \int_e \int \eta \left[ \frac{\partial^2 \mathbf{N}_i}{\partial x^2} - \frac{\partial^2 \mathbf{N}_i}{\partial y^2} \right] \left[ \frac{\partial^2 \mathbf{N}_j}{\partial x^2} - \frac{\partial^2 \mathbf{N}_j}{\partial y^2} \right] dx dy \\ & + \int_e \int 4\eta \frac{\partial^2 \mathbf{N}_i}{\partial x \partial y} \frac{\partial^2 \mathbf{N}_j}{\partial x \partial y} dx dy \end{aligned} \quad (3.27)$$

and element load vector

$$\mathbf{f}_j^e = \int_e \int Ra(\Gamma) \mathbf{N}_j \frac{\partial T}{\partial x} dx dy - Rb \int_e \int \mathbf{N}_j \frac{\partial \Gamma}{\partial x} dx dy \quad (3.28)$$

We thus have 12 degrees of freedom in the element and the unknown  $\bar{\psi}$  can be expanded by retaining 12 terms of a fourth order polynomial in  $x$  and  $y$ . Appendix B gives the shape functions for this element. The volume integrals in the coefficients of the stiffness matrix are calculated with the  $3 \times 3$  Gaussian quadrature rule. The same integration rule is applied in the calculation of the coefficients in the first term of the element load vector

$$\int_e \int Ra(\Gamma) \mathbf{N}_j \frac{\partial T}{\partial x} dx dy \quad (3.29)$$

where the temperature is given in the nodal points. The temperature derivative in the Gaussian points is calculated in a way that is consistent with the linear approximation of the temperature within the element (refer appendix B).

The interface between the two fluids is represented by a marker chain with coordinates  $(x_m, y_m)$ ,  $m = 1, \dots, N_m$ . The contribution of  $\Gamma$  to the element load vector is then calculated by the approximation

$$\int_e \int \mathbf{N}_j \frac{\partial \Gamma}{\partial x} dx dy \approx \sum_{m=2}^{N_m-1} \mathbf{N}_j(x_m, y_m) \frac{(y_{m+1} - y_{m-1})}{2} \quad (3.30)$$

[Christensen & Yuen, 1984].

### *Bicubic splines*

In this section the use of splines in the solution of the equation of motion in the stream function formulation is considered. Bicubic splines are used to define shape functions for the method of Galerkin on a rectangular grid. A rigorous definition of the use of splines in variational methods can be found in Schultz [1973]. Kopitzke [1977] gives a technical description of the use of the method in mantle convection modelling and gives expressions for spline functions defined on non-equidistant grids. Woidt [1980] describes the use of bicubic splines for Rayleigh-Taylor instabilities. Following is a summary of the implementation of the method to solve (2.22), with  $Ra \equiv 0$  and free-slip boundaries on equidistant grids. An approximate solution  $\bar{\psi}$  of (2.22) can be constructed by

$$\bar{\psi}(\mathbf{x}) = \sum_{j=1}^N \beta_j \Psi_j(\mathbf{x}) \quad (3.31)$$

where  $\beta_j$  are called the spline coefficients. Main advantage of using this type of

approximate solution is that the continuity requirement of the first order derivative is satisfied (refer appendix B) at a minimum cost (= low number of degrees of freedom). Disadvantage is that the shape functions are not local (each shape function is non-zero on 16 elements), in contrast to Lagrangian types of shape functions. Another complication arises in the implementation of the boundary conditions, which cannot be treated by elimination of local shape functions, as is common in the Lagrangian approach. Instead, a space spanned by the spline shape functions that satisfies the boundary conditions must explicitly be constructed.

The rectangular geometry  $\Omega = [0, L] \times [0, 1]$  can be discretized using a mesh of  $N_x \times N_y$  equidistant elements. Each element has dimensions  $\Delta x \times \Delta y$ . An approximate solution is sought of the form (3.31) where the spline functions  $\Psi_j$  satisfy the boundary conditions. The construction of these spline functions is described in appendix B. Application of the method of Galerkin yields a matrix equation

$$\mathbf{A} \mathbf{x} = \mathbf{f} \quad (3.32)$$

where the matrix coefficients  $A_{ij}$  are given by

$$A_{ij} = \iint_{\Omega} \eta \left[ \frac{\partial^2 \Psi_i}{\partial x^2} - \frac{\partial^2 \Psi_i}{\partial y^2} \right] \left[ \frac{\partial^2 \Psi_j}{\partial x^2} - \frac{\partial^2 \Psi_j}{\partial y^2} \right] dx dy + \iint_{\Omega} 4\eta \frac{\partial^2 \Psi_i}{\partial x \partial y} \frac{\partial^2 \Psi_j}{\partial x \partial y} dx dy \quad (3.33)$$

The integrals in (3.33) are calculated using a  $3 \times 3$  Gaussian rule. The load vector  $\mathbf{f}$  is calculated by an approximation similar to (3.30)

### 3.6 Benchmark results

The accuracy and the efficiency of the implementations have been compared against analytical solutions and numerical results obtained by different codes for a variety of cases. The three different methods are denoted by

- $\varepsilon$  penalty function method
- C1 stream function approach: non-conforming element
- $\psi$  stream function approach: bicubic splines

The quoted CPU times are in seconds and hold for the Iris Indigo workstation with a risc processor (MIPS R3000 @ 33 MHz). Inversion of a  $1000 \times 1000$  matrix using the Linpack benchmark [Dongara, 1986] is performed at this machine with

3.1 Mfbps (64 bit accuracy). The grid specification, as is used throughout this thesis, reflects the number of elements in an indirect way. The actual number of elements depends on the type of approximating shape functions: a grid specified as  $10 \times 10$  contains  $21 \times 21$  nodal points,  $2 \times 10 \times 10$  quadratic triangles (penalty function method),  $2 \times 20 \times 20$  linear triangles (temperature equation) and  $20 \times 20$  rectangles (non-conforming element, spline method). The equation(s) of motion (in either of the three formulations described above) and the temperature equation are solved on meshes with the same distribution of nodal points. The coupling between the equations is described in appendices A and B.

### 3.6.1 Thermal convection benchmark

Both the penalty function method and the stream function method with the non-conforming element have been used to study thermal convection. The benchmark comparison of Blankenbach et al. [1989] defines a number of cases and best estimates for the numerical values of some derived quantities, obtained from a comparison of several finite difference, finite element, and spectral methods. Two dimensional thermal convection of a Newtonian Boussinesq fluid at infinite Prandtl number is considered. The non-dimensional equations (2.15-2.17) are solved on a rectangular domain  $\Omega = [0, \lambda] \times [0, 1]$ , where  $\lambda$  is the aspect ratio. The quantities to be derived are

i) the Nusselt number

$$Nu = - \int_0^\lambda \frac{\partial T}{\partial y}(x, y = 1) dx / \int_0^\lambda T(x, y = 0) dx \quad (3.34)$$

ii) the non-dimensional root-mean-squared velocity

$$v_{rms} = \frac{1}{\lambda} \left[ \int_{\Omega} (v_2^2 + w^2) d\Omega \right]^{\frac{1}{2}} \quad (3.35)$$

and iii) the non-dimensional temperature gradient in the corners of the region  $\Omega$

$$q = - \frac{\partial T}{\partial y} \quad (3.36)$$

with  $q_1 = q(0, 1)$ ,  $q_2 = q(\lambda, 1)$ ,  $q_3 = q(\lambda, 0)$  and  $q_4 = q(0, 0)$ . Grid refinements are used to improve the resolution in the boundary layers of the convecting cell.

Case 1

Stationary convection with constant viscosity in a square region ( $\lambda = 1$ ), free-slip boundaries, constant temperature at top ( $T = 0$ ) and bottom ( $T = 1$ ) and reflective vertical boundaries ( $\partial T/\partial x = 0$ ). Rayleigh number is varied between  $10^4$  and  $10^6$ .

Table 3.1a Blankenbach et al. [1989], Case 1a  $Ra = 10^4$

Code	Grid	$Nu$	$v_{rms}$	$q_1$	$q_2$	cpu †
$\epsilon$	$20 \times 20$	4.8891	42.896	8.0551	0.5888	37
	$30 \times 30$	4.8880	42.878	8.0589	0.5908	118
	$40 \times 40$	4.8880	42.887	8.0587	0.5911	282
C1	$20 \times 20$	4.8900	42.923	8.0563	0.5886	58
	$30 \times 30$	4.8885	42.894	8.0596	0.5907	211
	$35 \times 35$	4.8885	42.897	8.0590	0.5910	355
best estimate		4.8844	42.865	8.0594	0.5888	

† Cpu-time in seconds on Iris Indigo R3000 workstation (see text)

Table 3.1b Blankenbach et al. [1989], Case 1b  $Ra = 10^5$

Code	Grid	$Nu$	$v_{rms}$	$q_1$	$q_2$	cpu
$\epsilon$	$20 \times 20$	10.532	193.262	18.993	0.7181	62
	$30 \times 30$	10.541	193.153	19.076	0.7251	162
	$40 \times 40$	10.539	193.191	19.077	0.7264	386
C1	$20 \times 20$	10.5351	193.407	18.990	0.71691	81
	$30 \times 30$	10.5425	193.233	19.074	0.72438	251
	$35 \times 35$	10.5412	193.272	19.075	0.72548	424
best estimate		10.534	193.215	19.079	0.72275	

Table 3.1c Blankenbach et al. [1989], Case 1c  $Ra = 10^6$

Code	Grid	$Nu$	$v_{rms}$	$q_1$	$q_2$	cpu
$\epsilon$	$20 \times 20$	21.852	835.331	44.419	0.8412	64
	$30 \times 30$	21.986	834.030	45.902	0.8791	309
	$40 \times 40$	21.986	833.985	45.948	0.8771	469
C1	$20 \times 20$	21.861	835.703	44.561	0.83619	65
	$30 \times 30$	21.997	834.100	45.852	0.8732	237
	$35 \times 35$	21.998	834.316	45.902	0.8773	403
best estimate		21.972	833.990	45.964	0.8772	

## Case 2

Stationary convection with temperature and depth dependent viscosity  $\eta$

$$\eta = \eta_0 \exp[ -bT + c(1 - y) ] \quad (3.37)$$

The Rayleigh number  $Ra_0$  is based on the viscosity  $\eta_0$

$$Ra_0 = \frac{\rho\alpha g\Delta Th^3}{\kappa\eta_0} \quad (3.38)$$

Table 3.2a Blankenbach et al. [1989], Case 2a:  $\lambda = 1$ ,  $Ra_0 = 10^4$ ,  $b = \ln(1000)$ ,  $c = 0$

Code	Grid	$Nu$	$v_{rms}$	$q_1$	$q_2$	$q_3$	$q_4$	cpu
$\epsilon$	$20 \times 20$	10.040	476.87	17.393	1.0046	25.182	0.4301	150
	$30 \times 30$	10.061	478.44	17.504	1.0089	26.731	0.4955	517
	$40 \times 40$	10.065	478.77	17.520	1.0109	26.905	0.5001	1379
C1	$20 \times 20$	10.049	477.32	17.406	1.0025	25.284	0.4310	512
	$30 \times 30$	10.065	478.49	17.509	1.0075	26.771	0.4962	2167
best estimate		10.066	480.43	17.531	1.0085	25.809	0.4974	

Table 3.2b Blankenbach et al. [1989], Case 2b:  $\lambda = 2.5$ ,  $Ra_0 = 10^4$ ,  $b = \ln(16384)$ ,  $c = \ln(64)$

Code	Grid	$Nu$	$v_{rms}$	$q_1$	$q_2$	$q_3$	$q_4$	cpu
$\epsilon$	$32 \times 16$	6.9011	171.013	18.187	0.1791	13.756	0.5972	229
	$40 \times 20$	6.9269	171.442	18.401	0.1799	14.040	0.6115	452
	$48 \times 24$	6.9276	171.407	18.456	0.1799	14.113	0.6168	809
	$56 \times 28$	6.9276	171.397	18.472	0.1798	14.139	0.6176	1265
C1	$32 \times 16$	6.9156	171.592	18.128	0.1796	13.872	0.5984	544
	$40 \times 20$	6.9396	171.828	18.372	0.1803	14.076	0.6122	1236
	$48 \times 24$	6.9357	171.650	18.428	0.1802	14.122	0.6173	2422
best estimate		6.9299	171.755	18.484	0.1774	14.168	0.6177	

## Case 3

Time-dependent convection with constant viscosity and internal heating. Rigid top and bottom boundary. Constant temperature at top ( $T = 0$ ), insulated bottom ( $\partial T/\partial y = 0$ ) and side ( $\partial T/\partial x = 0$ ) boundaries. Aspect ratio  $\lambda = 1.5$ , constant heat production rate  $Q$ , and Rayleigh number

$$Ra = \frac{\alpha g Q h^5}{\kappa_0^2 \rho_0 c_p \eta_0} = 216000 \quad (3.39)$$

As initial condition the result from the stationary model with lower Rayleigh number  $Ra = 21600$  is chosen. After integration of the time-dependent equations over a sufficiently long time interval, the solution becomes periodic in time with two distinct periods (P2). From the time-series of  $Nu$  and  $v_{rms}$ , the position of the extrema and the values are obtained by quadratic Lagrangian interpolation, using the three data points around each extremum.

Table 3.3 Blankenbach et al. [1989], Case 3

Code	Grid	Period	$Nu$				$v_{rms}$			
			max	min	max	min	max	min	max	min
C1	$30 \times 20$	0.0489	7.367	6.44	7.19	6.77	60.8	32.2	57.6	30.2
	$35 \times 25$	0.0486	7.374	6.45	7.18	6.78	60.7	32.1	57.2	30.9
best	estimate	0.0480	7.379	6.47	7.20	6.80	60.4	32.0	57.4	30.3

The penalty function method gives very similar results. The global accuracy of the method is governed by the accuracy in solving the time-dependent heat equation (2.3), which is identical for both methods.

### 3.6.2 Rayleigh-Taylor instability: linear stability analysis

The stream function codes have further been tested for Rayleigh-Taylor instabilities. From linear stability analysis [Chandrasekhar, 1961] it is found that the amplitude of harmonic perturbations between originally horizontal interfaces grows exponentially in time with a specific growth rate  $\kappa$ . The growth rate depends a.o. on the viscosity and densities of the fluids and the boundary conditions, and can be determined analytically for simple models. Consider the situation of two superimposed homogeneous fluid layers of equal thickness, contained in a rectangular geometry  $\Omega = [0, 1] \times [0, 1]$  with free-slip boundaries. The fluids are isoviscous ( $\eta = 1$ ) and have density difference  $\Delta\rho = 1$ . The interface between the fluids at  $y = 0.5$  is perturbed by a cosinusoidal perturbation with wavelength 2 and amplitude 0.001. Table 3.4 shows the non-dimensional growth factor for this case. The numerical values are obtained using a discretization of the interface with 101 markers.

Table 3.4 Rayleigh-Taylor instability

Code	Grid	Growth rate	cpu
C1	10 × 10	0.05304	1.9
	20 × 20	0.05310	19.5
	30 × 30	0.05312	85
$\Psi$	2 × 2	0.05295	0.14
	5 × 5	0.053118	2.4
	10 × 10	0.053121	12.8
Analytical		0.0531307	

### 3.7 Conclusions

From the above results we can conclude that the codes for thermal convection are very accurate: all derived quantities, even for the coarse grids, fall within 0.5 % of the 'best estimates' quoted in Blankenbach et al. [1989]. Results obtained with the non-conforming element and with the penalty function method compare very well. The non-conforming element is about 50 % more expensive with regard to memory and cpu requirements. The spline method shows to be accurate even at very coarse grids. For a given accuracy, the use of splines is much more efficient than using the non-conforming element. A further comparison of the methods is given in the next chapter, in which the application of the methods to Rayleigh-Taylor instabilities is discussed.