Modelling the non-Hydrostatic Pressure in σ-coordinate Ocean Models

Master of Science Thesis in Applied Mathematics

Eirik Keilegavlen

Department of Mathematics
University of Bergen

22nd September 2006

Preface

Now that I have finished this thesis, my time as a master student has come to an end. First I would like to thank my supervisor, Jarle Berntsen, for always being enthusiastic and willing to help, and for allowing me to work on my own when I have wanted to do so. I will also express my gratitude to all the other persons from whom I have learned mathematics during my time here at the university.

Of the present and former graduate students at the Department of Mathematics, I will especially thank two people. Kristin Rygg has, in addition to being a good friend, helped me a lot on the finishing touches of the layout. Kristine Selvikvåg deserves thanks for all the company, for always being willing to listen, and for her ability to see the glass as half full instead of half empty. As for the rest of the graduate students, I am grateful for help with my thesis, and for you distracting me from my work whenever I needed a break.

At last, I will thank my family for always being there when I need help and support.

Eirik Keilegavlen
Bergen, September 2006

Contents

Introduction

Numerical ocean modelling is based upon solving equations which govern the motion of fluids. The equations are hard to solve, and approximations must be done. In ocean modelling, one has traditionally made a hydrostatic assumption, which makes the computations less expensive. This was not problematic if the coarse grid resolution in the early days of ocean modelling is taken into account.

Large scale processes in the ocean are to a good approximation hydrostatic. This means that the pressure at a point is caused by the weight of the fluid above the point. The non-hydrostatic pressure due to motion of the fluid is neglected. However, for processes on a horizontal scale smaller than 10 km, non-hydrostatic effects are important (see Marshall et al. (1997b) for a schematic diagram of hydrostatic and non-hydrostatic regimes). With the increasing computer power available, it is possible to use a fine enough spatial resolution to resolve non-hydrostatic processes also for realistic problems in ocean modelling.

Non-hydrostatic processes are important for many phenomena. One example of this is mixing of water masses. Without mixing, most of the motion in the ocean would stop within a few thousand years (Munk and Wunsch 1998). Non-hydrostatic effects can also be important in industrial applications. For instance, offshore oil production in deep waters can be affected by internal waves (Osborne and Burch 1980). Also for biological processes on a length scale of centimetres, non-hydrostatic processes should be taken into account.

An ocean model with non-hydrostatic capabilities using a Cartesian z-coordinate in the vertical direction was described in Marshall et al. (1997a), and Marshall et al. (1997b). The non-hydrostatic pressure is included by a linear system of partial differential equations. This set of equations is combined into an elliptic differential equation, which is discretized by finite differences. The resulting linear system has a sparse structure which is well studied.

Ocean models which use Cartesian coordinates in the vertical direction

have problems resolving processes near the bottom. Small scale processes steered by local topography tend to be highly non-hydrostatic. There have been attempts to improve the ability of z-coordinate models to resolve the bottom layer, see e.g. Adcroft et al. (1997). However, it can be useful to use the terrain following σ-coordinates to model processes near the seabed.

It has been harder to include non-hydrostatic effects in σ-coordinate models than in z-coordinates. One model was described by Lin and Li (2002). However, this model does not allow for splitting of the baroclinic and barotropic mode, leading to a computationally expensive method.

Kanarska and Maderich (2003) designed a numerical model with both mode splitting, terrain following vertical coordinates and non-hydrostatic capabilities. The equations for the non-hydrostatic pressure are transformed from a Cartesian to a σ-coordinate system. This approach also gives a linear system to be solved for the non-hydrostatic pressure. However, the equations are complicated from a computational point of view. The resulting linear system has a structure which is not as well studied as the system of Marshall et al. (1997a). This makes the equations harder to solve than the corresponding system in z-coordinate models. Kanarska and Maderich (2003) also compare the model results with results from laboratory experiments and other numerical models for some chosen test cases. The results look promising.

Proposals of improvements of the methods by Kanarska and Maderich (2003) were given by Heggelund et al. (2004). Their approach gives a linear system with the same structure as in z-coordinates. However, the numerical scheme is rather complicated. From the numerical experiments done by Heggelund et al. (2004), the model seems to capture non-hydrostatic effects well.

Berntsen and Furnes (2005) proposed to model the non-hydrostatic pressure directly in σ-coordinates instead of transforming the equations from Cartesian coordinates. The resulting linear system has the same structure as in z-coordinate models. The method was used by Berntsen et al. (2006). The results are in good agreement with the results obtained by the model due to Marshall et al. (1997a). However, there are no mathematical proofs that the set of equations proposed by Berntsen and Furnes (2005) is the same as the one used by Kanarska and Maderich (2003).

There is also disagreement about which boundary condition should be applied on the surface for the non-hydrostatic pressure. A homogeneous Neumann condition was proposed by Marshall et al. (1997b). This was adopted by e.g. Casulli (1999) and Heggelund et al. (2004). Kanarska and Maderich (2003) used a homogeneous Dirichlet condition on the surface. From a physical point of view, there are reasons to apply both kinds of

boundary conditions.

The purpose of this thesis is to investigate different methods to compute the non-hydrostatic pressure in mode-split, σ-coordinate ocean models. We will examine the sets of equations proposed and the boundary conditions on the surface. Both physical and numerical aspects will be investigated.

In Chapter 1 we give the governing equations which our numerical model is based upon, and discuss different sources of pressure in the ocean. Different sets of equations and boundary conditions proposed to model the non-hydrostatic pressure are derived and discussed in Chapter 2. Chapter 3 gives an introduction to some mathematical concepts which we will need later. The real test of the different methods is their ability to model non-hydrostatic phenomena. In order to investigate this, we will do numerical experiments. The setup for the test case is described in Chapter 4. When we discretize the equations for the non-hydrostatic pressure correction by finite differences, we get a linear system of equations. Some properties of the corresponding matrices for the numerical experiments are investigated in Chapter 5. In Chapter 6 results from the simulations are presented, and the differences in the solutions we obtain are highlighted. In Chapter 7 we consider the simulations from a numerical point of view, and discuss if some of the linear systems are easier to solve than others. In the final chapter we draw conclusions and point out further work which should be done.

Chapter 1

The Basic Equations

In this chapter, we describe the numerical model used in this thesis, and give the governing equations on which it is based. In the end, different sources of pressure in the ocean are discussed. The variables are listed and explained in Appendix A.

1.1 Bergen Ocean Model

The simulations in this thesis are done by using Bergen Ocean Model (BOM). BOM is a hydrostatic model which has been modified to include non-hydrostatic effects. The discretizations are done by finite differences. The staggered Arakawa C-grid is used for the spatial coordinates (Arakawa 1966). See Berntsen (2004) for a further description of the model.

We will focus more on comparing different models than on comparing the model results with laboratory experiments in this thesis. We therefore trade the ability to capture 3D-effects which are important for the phenomenon for the higher resolution of 2D-models. The 2D-version of BOM allows for rotation and constant velocities in a third direction. However, the simulations are done on a short time scale so the Coriolis force plays no significant role. Motivated by this, the simulations are done without rotation, and without any velocities or nonzero derivatives in the third direction.

1.2 The Governing Equations

1.2.1 The Momentum Equations

In a Cartesian coordinate system with x and z as horizontal and vertical coordinates, respectively, the momentum equations are (Berntsen and Furnes

2005)

$$\frac{\partial U}{\partial t}+\frac{\partial U^2}{\partial x}+\frac{\partial UW}{\partial z}=-\frac{1}{\rho_0}\frac{\partial P}{\partial x}+\frac{\partial}{\partial x}\left(A_M\frac{\partial U}{\partial x}\right)+\frac{\partial}{\partial z}\left(K_M\frac{\partial U}{\partial z}\right), \quad (1.1)$$

$$\frac{\partial W}{\partial t}+\frac{\partial UW}{\partial x}+\frac{\partial W^2}{\partial z}=-\frac{1}{\rho_0}\frac{\partial P}{\partial z}+\frac{\rho g}{\rho_0}+\frac{\partial}{\partial x}\left(A_M\frac{\partial W}{\partial x}\right)+\frac{\partial}{\partial z}\left(K_M\frac{\partial W}{\partial z}\right). \quad (1.2)$$

Here U is the horizontal velocity and W is the vertical velocity. The pressure is denoted by P, ρ is the density, A_M and K_M is the horizontal and vertical viscosity coefficient, respectively, and g is the acceleration of gravity. We have applied the Boussinesq approximation. This implies that the density is assumed equal to a reference density, ρ_0, unless it is multiplied by the gravity.

If the hydrostatic approximation is used, Equation (1.2) simplifies to

$$\rho g=-\frac{\partial P}{\partial z}. \quad (1.3)$$

1.2.2 The Equation of Continuity

In the Boussinesq approximation, the equation of continuity,

$$\frac{\partial \rho}{\partial t}+\nabla\cdot(\rho\mathbf{U})=0,$$

is replaced by the incompressible form

$$\nabla\cdot\mathbf{U}=0, \quad (1.4)$$

where $\mathbf{U}=(U,W)$ (Kundu and Cohen 2004).

1.2.3 The Equation for Density

In this thesis , we apply an equation for conservation of density

$$\frac{\partial \rho}{\partial t}+\frac{\partial U\rho}{\partial x}+\frac{\partial W\rho}{\partial z}=A_H\frac{\partial^2\rho}{\partial x^2}+K_H\frac{\partial^2\rho}{\partial z^2}, \quad (1.5)$$

where A_H and K_H is the horizontal and the vertical diffusivity coefficients, respectively.

1.2.4 A Closed Set of Equations

In order to get a closed set of equations, the coefficients A_H, A_M, K_H, and K_M must be specified. Choosing values for these coefficients is a major problem in ocean modelling, see for instance Haidvogel and Beckmann (1999). We will use constant values for the viscosity and diffusivity coefficients. When these are specified the Equations (1.1), (1.2), (1.4), and (1.5) form a closed set, i.e. there are as many equations as there are unknowns. However, we will decompose the pressure into pressure from different sources. Therefore, we will also need more equations. Which equations should be used will be discussed later.

1.3 The Boundary Conditions

The kinematic boundary conditions at the free surface and at the bottom are given by, respectively

$$\begin{aligned} W &= U\frac{\partial \eta}{\partial x} + \frac{\partial \eta}{\partial t}\,, \qquad && z = \eta(x,t)\,, \\ W &= -U\frac{\partial H}{\partial x}\,, && z = -H(x)\,. \end{aligned} \tag{1.6}$$

At lateral boundaries, we have a no-flow condition,

$$\mathbf{U}\cdot\mathbf{n} = 0\,,$$

where $\mathbf{n}$ is the outer normal unit vector. At the bottom, the effect of friction is described by

$$\rho_0 K_M \frac{\partial U}{\partial z} = \tau\,.$$

The bottom drag, τ, is assumed to be given by

$$\tau = \rho_0 C_D |U_b| U_b\,,$$

where U_b is the velocity at the bottom, and the bottom drag coefficient, C_D, is given by

$$C_D = \max\left(0.0025\,, \frac{\kappa^2}{(\ln(\frac{z_b}{z_0})^2)}\right).$$

Here, κ is the von Karman constant which is set to 0.4, z_b is the distance from the bottom to the nearest grid point, and z_0 is the bottom roughness parameter.

1.4 The σ-Coordinate System

Ocean models based on Cartesian coordinates have problems with resolving processes near the bottom and the surface. We therefore transform the equations into a terrain following σ-coordinate system (Phillips 1957). The σ-coordinate of the bottom is always -1, while σ at the free surface always equals 0.

The relationships between the old variables (x, z, t) and the new ones (x^*, σ, t^*) are

$$x^* = x\,, \qquad \sigma = \frac{z-\eta}{D}\,, \text{ and } \qquad t^* = t\,,$$

where

$$D = H + \eta\,,$$

is the dynamic depth. If A is any dependent variable such that

$$A(x, z, t) = A^*(x^*, \sigma, t^*)\,, \tag{1.7}$$

the relationship between the old and the new derivatives is found by applying the chain rule. We get (Blumberg and Mellor 1987)

$$\begin{aligned}
\frac{\partial A}{\partial x} &= \frac{\partial A^*}{\partial x^*} - \frac{1}{D}\frac{\partial A^*}{\partial \sigma}\left(\sigma\frac{\partial D}{\partial x^*} + \frac{\partial \eta}{\partial x^*}\right)\,,\\
\frac{\partial A}{\partial z} &= \frac{1}{D}\frac{\partial A^*}{\partial \sigma}\,,\\
\frac{\partial A}{\partial t} &= \frac{\partial A^*}{\partial t^*} - \frac{1}{D}\frac{\partial A^*}{\partial \sigma}\left(\sigma\frac{\partial D}{\partial t^*} + \frac{\partial \eta}{\partial t^*}\right)\,.
\end{aligned} \tag{1.8}$$

We define a new vertical velocity perpendicular to the iso-σ surfaces,

$$\omega = W - U\left(\sigma\frac{\partial D}{\partial x^*} + \frac{\partial \eta}{\partial x^*}\right) - \left(\sigma\frac{\partial D}{\partial t^*} + \frac{\partial \eta}{\partial t^*}\right)\,. \tag{1.9}$$

We also get a new horizontal velocity U^* along the iso-σ surfaces.

After deleting the asterisks, the momentum equations become

$$\frac{\partial UD}{\partial t} + \frac{\partial U^2 D}{\partial x} + \frac{\partial U\omega}{\partial \sigma} = -\frac{D}{\rho_0}\frac{\partial P}{\partial x} + \frac{1}{\rho_0}\frac{\partial P}{\partial \sigma}\left(\sigma\frac{\partial D}{\partial x} + \frac{\partial \eta}{\partial x}\right) + DF_x\,,$$

$$\frac{\partial WD}{\partial t} + \frac{\partial UWD}{\partial x} + \frac{\partial W\omega}{\partial \sigma} = -\frac{1}{\rho_0}\frac{\partial P}{\partial \sigma} + DF_\sigma\,.$$

The first equation is taken from Blumberg and Mellor (1987), the second can be derived from Equation (1.2) by using the Equations (1.8). The terms DF_x and DF_σ represents viscosity terms. These are rather complicated for the σ-coordinates. See for instance Blumberg and Mellor (1987) for a description of DF_x. The terms in DF_σ can be derived from the Equations (1.8).

The new equation of continuity becomes

$$\frac{\partial UD}{\partial x} + \frac{\partial \omega}{\partial \sigma} + \frac{\partial \eta}{\partial t} = 0 \,.$$

The new boundary conditions are

$$\omega(-1) = \omega(0) = 0 \,.$$

The conservation equation of density becomes

$$\frac{\partial \rho D}{\partial t} + \frac{\partial \rho UD}{\partial x} + \frac{\partial \rho \omega}{\partial \sigma} = \frac{\partial}{\partial x}\left(DA_H \frac{\partial \rho}{\partial x}\right) + \frac{\partial}{\partial \sigma}\left(\frac{K_H}{D}\frac{\partial \rho}{\partial \sigma}\right) .$$

1.5 Important Concepts in Numerical Ocean Modelling

Here, we give a short overview of some concepts in ocean modelling which will be needed later.

1.5.1 Mode Splitting

For coastal waters both fast moving barotropic waves and slower baroclinic waves are important for the ocean dynamics (Blumberg and Mellor 1987). It is computationally expensive to solve the full 3D-equations. We therefore split the equations into a depth averaged 2D-part and the 3-dimensional deviations from the average. Each of the fields are then propagated forward in time. The fast barotropic or external modes will be contained in the 2D-part. Therefore the 2D time step must be relatively small in order to fulfil the CFL-criterion. The 3D-part contains the slower baroclinic or internal modes. Thus the computational expensive 3D-time step can be taken larger.

There are numerous ways of doing the mode splitting. The basic idea is to propagate the barotropic part first and compute averaged values of the velocities and the surface elevation. These values are then used to propagate the baroclinic part forward in time. For details on how this is done in

different models, see for instance Berntsen (2004) or Mellor (2004). A detailed description of the operator splitting used in BOM can also be found in Ådlandsvik and Engedahl (1991) or Kowalik and Murty (1993).

We will say much about how the equations are solved in BOM. For the numerical method used for time stepping, advection schemes etc. see Berntsen et al. (2006). We will, however, mention a few details which we will need later on. In the depth averaged computations there is a coefficient for 2D-viscosity denoted $A_{M_{2D}}$. The value of this must be specified to get a closed system of equations. The time stepping is done with a predictor-corrector method. This implies that for each time step, the mode splitting is done twice.

1.5.2 Conservation of Mass

One of the most important properties a numerical ocean model should have is to conserve mass. That is, the equation of continuity must be fulfilled. If water masses for some reason appears or disappears in the model, this will soon lead to numerical instabilities. If the model uses mode splitting, there are different ways of enforcing the principle of conservation of mass. It is of great importance that after a time step is finished, the final velocity and density fields must fulfil the equation of continuity.

1.5.3 The non-Hydrostatic Pressure

Most numerical ocean models use the hydrostatic approximation. That is Equation (1.2) is replaced by Equation (1.3). This makes each time step computationally much cheaper. However, with the increasing computer power available, questions have been raised whether the hydrostatic approximation is the best way of spending computational resources.

According to Marshall et al. (1997b), the hydrostatic approximation breaks down for processes on a length scale somewhere between 1 and 10 km. When the vertical and the horizontal length scales are of the same order, non-hydrostatic effects will be important. This can easily be seen from the aspect ratio defined in Gill (1982). Examples of non-hydrostatic processes are high-frequency internal waves, wave breaking, shear instabilities, and the entrainment in gravity currents (Bourgault and Kelley 2004), (Legg et al. 2006). If the spatial resolution of the model is high enough to resolve such processes, the model should have non-hydrostatic capabilities to capture the physics correctly.

It is important to notice that the main pressure field in the ocean is hydrostatic. The non-hydrostatic pressure only contributes with a small

part of the forces which drives the motion, but this small part is necessary for some phenomena to take place.

One way of including the non-hydrostatic effects is to assume that the pressure can be decomposed into (Marshall et al. 1997a)

$$P = P_{Atm} + P_{\eta} + P_{Int} + Q \, . \tag{1.10}$$

Here, P_{Atm} is the atmospheric pressure and

$$P_{\eta} = g\rho_0\eta \, ,$$

is the pressure due to surface elevation. The internal pressure is represented by

$$P_{Int} = \int_z^0 \rho(x, z', t)dz' \, ,$$

and Q is the non-hydrostatic pressure. Only the first three terms in Equation (1.10) are included in hydrostatic models. For such models, the basic equations given in this chapter form a closed set of equations.

If mode splitting is not applied, the non-hydrostatic pressure is included in the governing equations together with the other terms in Equation (1.10). This was done by for instance Casulli (1999) in z-coordinates and Lin and Li (2002) with terrain following σ-coordinates.

If we will combine mode splitting and non-hydrostatic pressure, we need more equations in addition to those given in this chapter in order to get a closed set. Proposals of how this can be done in σ-coordinates were given by Kanarska and Maderich (2003), Heggelund et al. (2004), and Berntsen and Furnes (2005). In this thesis we will compare the methods by Kanarska and Maderich (2003) and Berntsen and Furnes (2005).

Chapter 2

Equations for the non-Hydrostatic Pressure

Different sets of equations have been proposed to compute the non-hydrostatic pressure corrections in σ-coordinate models. In this chapter we give two systems of equations, and discuss different boundary conditions for the non-hydrostatic pressure.

2.1 The Full System of Equations

The non-hydrostatic pressure correction in Cartesian coordinates is assumed to be given by

$$\begin{aligned} \frac{\partial UD}{\partial t} &= -\frac{D}{\rho_0}\frac{\partial Q}{\partial x}\,, \\ \frac{\partial WD}{\partial t} &= -\frac{D}{\rho_0}\frac{\partial Q}{\partial z}\,. \end{aligned} \tag{2.1}$$

There is no gravity term in the vertical direction, this is included in the hydrostatic part of the calculations. We have also neglected the non-linear terms in the equations of motion, see Patankar (1980) for a discussion of this. The equation of continuity,

$$\frac{\partial UD}{\partial x} + \frac{\partial WD}{\partial z} = 0\,, \tag{2.2}$$

must also be fulfilled. We transform the equations to σ-coordinates by the Equations (1.8) and get (Heggelund et al. 2004)

$$\begin{aligned}\frac{\partial UD}{\partial t} &= -\frac{D}{\rho_0}\frac{\partial Q}{\partial x} + \frac{1}{\rho_0}\frac{\partial Q}{\partial \sigma}\left(\sigma\frac{\partial D}{\partial x} + \frac{\partial \eta}{\partial x}\right), \\ \frac{\partial WD}{\partial t} &= -\frac{1}{\rho_0}\frac{\partial Q}{\partial \sigma}. \end{aligned} \tag{2.3}$$

The equation of continuity becomes

$$\frac{\partial UD}{\partial x} + \frac{\partial \omega}{\partial \sigma} + \frac{\partial \eta}{\partial t} = 0. \tag{2.4}$$

Together with the relationship between W and ω given by Equation (1.9) this is a closed set of equations which can be used to compute the velocity corrections (Heggelund et al. 2004). If combined with the equations given in Chapter 1, the Equations (1.9), (2.3), and (2.4) form a closed set which governs the motion in the ocean. As we will see, the terms of the form $(\sigma\frac{\partial D}{\partial x} + \frac{\partial \eta}{\partial x})$ complicate the computations considerably.

The computation of the external and the internal mode were described in Section 1.5.1. This gives us provisional velocities $\tilde{U}$ and $\tilde{W}$. The non-hydrostatic pressure and velocity corrections are then assumed to be given by semi-discrete equations of the form (Kanarska and Maderich 2003)

$$\begin{aligned}\frac{(U-\tilde{U})D}{\Delta t} &= -\frac{D}{\rho_0}\frac{\partial Q}{\partial x} + \frac{1}{\rho_0}\frac{\partial Q}{\partial \sigma}\left(\sigma\frac{\partial D}{\partial x} + \frac{\partial \eta}{\partial x}\right), \\ \frac{(W-\tilde{W})D}{\Delta t} &= -\frac{1}{\rho_0}\frac{\partial Q}{\partial \sigma}, \\ \frac{\partial UD}{\partial x} + \frac{\partial \omega}{\partial \sigma} + \frac{\partial \eta}{\partial t} &= 0. \end{aligned} \tag{2.5}$$

The Equations (2.5) can differ from model to model, depending on which method is used for the time stepping.

We get the following expressions for the velocities

$$UD = -\frac{\Delta t}{\rho_0}D\frac{\partial Q}{\partial x} + \frac{\Delta t}{\rho_0}\frac{\partial Q}{\partial \sigma}\left(\sigma\frac{\partial D}{\partial x} + \frac{\partial \eta}{\partial x}\right) + \tilde{U}D, \tag{2.6}$$

and by using Equation (1.9),

$$\omega = -\frac{\Delta t}{\rho_0}\frac{1}{D}\frac{\partial Q}{\partial \sigma} - U\left(\sigma\frac{\partial D}{\partial x} + \frac{\partial \eta}{\partial x}\right) + \tilde{W}. \tag{2.7}$$

We differentiate Equation (2.6) with respect to x and Equation (2.7) with respect to σ and get

$$\frac{\partial UD}{\partial x} = -\frac{\Delta t}{\rho_0}\frac{\partial}{\partial x}\left(D\frac{\partial Q}{\partial x}\right) + \frac{\Delta t}{\rho_0}\frac{\partial}{\partial x}\left(\frac{\partial Q}{\partial \sigma}\left(\sigma\frac{\partial D}{\partial x} + \frac{\partial \eta}{\partial x}\right)\right) + \frac{\partial \tilde{U}D}{\partial x}\,, \quad (2.8)$$

and

$$\begin{aligned}\frac{\partial \omega}{\partial \sigma} &= -\frac{\Delta t}{\rho_0}\frac{\partial}{\partial \sigma}\left(\frac{1}{D}\frac{\partial Q}{\partial \sigma}\right) - \frac{\partial}{\partial \sigma}\left(U\left(\sigma\frac{\partial D}{\partial x} + \frac{\partial \eta}{\partial x}\right)\right) + \frac{\partial \tilde{W}}{\partial \sigma} \\ &= -\frac{\Delta t}{\rho_0}\frac{\partial}{\partial \sigma}\left(\frac{1}{D}\frac{\partial Q}{\partial \sigma}\right) + \frac{\Delta t}{\rho_0}\frac{\partial}{\partial \sigma}\left(\frac{\partial Q}{\partial x}\left(\sigma\frac{\partial D}{\partial x} + \frac{\partial \eta}{\partial x}\right)\right) \\ &\quad -\frac{\Delta t}{\rho_0}\frac{1}{D}\frac{\partial}{\partial \sigma}\left(\frac{\partial Q}{\partial \sigma}\left(\sigma\frac{\partial D}{\partial x} + \frac{\partial \eta}{\partial x}\right)^2\right) + \frac{\partial}{\partial \sigma}\left(\tilde{W} - \tilde{U}\sigma\frac{\partial D}{\partial x}\right), \quad (2.9)\end{aligned}$$

where we have used Equation (2.6). The last term on the right hand side of Equation (2.9) equals $\frac{\partial \tilde{\omega}}{\partial \sigma}$. We insert Equation (2.8) and Equation (2.9) into the equation of continuity (Equation (2.4)) and get

$$\begin{aligned}&\frac{\partial}{\partial x}\left(D\frac{\partial Q}{\partial x}\right) - \frac{\partial}{\partial x}\left(\frac{\partial Q}{\partial \sigma}\left(\sigma\frac{\partial D}{\partial x} + \frac{\partial \eta}{\partial x}\right)\right) + \frac{1}{D}\frac{\partial^2 Q}{\partial \sigma^2} - \frac{\partial}{\partial \sigma}\left(\frac{\partial Q}{\partial x}\left(\sigma\frac{\partial D}{\partial x} + \frac{\partial \eta}{\partial x}\right)\right) \\ &\qquad + \frac{1}{D}\frac{\partial}{\partial \sigma}\left(\frac{\partial Q}{\partial \sigma}\left(\sigma\frac{\partial D}{\partial x} + \frac{\partial \eta}{\partial x}\right)^2\right) = \frac{\rho_0}{\Delta t}\left(\frac{\partial \tilde{U}D}{\partial x} + \frac{\partial \tilde{\omega}}{\partial \sigma} + \frac{\partial \eta}{\partial t}\right).\end{aligned} \quad (2.10)$$

A finite difference approximation of Equation (2.10) is given in Appendix B. The discretization gives a linear system with 9 nonzero diagonals. When the non-hydrostatic pressure is known, discrete versions of the first two of the Equations (2.5) are used to find the corrected velocities.

2.2 A Simplified System of Equations

Berntsen and Furnes (2005) proposed to consider the non-hydrostatic pressure directly as a function of (x, σ, t) instead of first using z as vertical coordinate, and then do a transformation. The pressure is caused by convergences or divergences in the cells. The equations are based on the same

physical principle as is used for the full set of equations in Cartesian coordinates (the Equations (2.1)), but we avoid the mathematical complications connected to the change of coordinate system. This point of view gives us the following set of equations for the non-hydrostatic pressure correction

$$\begin{aligned} \frac{(U-\tilde{U})D}{\Delta t} &= -\frac{D}{\rho_0}\frac{\partial Q}{\partial x}\,, \\ \frac{(\omega-\tilde{\omega})D}{\Delta t} &= -\frac{1}{\rho_0}\frac{\partial Q}{\partial \sigma}\,, \\ \frac{\partial UD}{\partial x}+\frac{\partial \omega}{\partial \sigma}+\frac{\partial \eta}{\partial t} &= 0\,. \end{aligned} \tag{2.11}$$

This gives us

$$UD = \tilde{U}D - \frac{\Delta t}{\rho_0} D\frac{\partial Q}{\partial x}\,,$$

and

$$\omega = \tilde{\omega} - \frac{\Delta t}{\rho_0}\frac{1}{D}\frac{\partial Q}{\partial \sigma}\,,$$

and by inserting into the equation of continuity, we get

$$\frac{\partial}{\partial x}\left(D\frac{\partial Q}{\partial x}\right)+\frac{1}{D}\frac{\partial^2 Q}{\partial \sigma^2} = \frac{\rho_0}{\Delta t}\left(\frac{\partial \tilde{U}D}{\partial x}+\frac{\partial \tilde{\omega}}{\partial \sigma}+\frac{\partial \eta}{\partial t}\right). \tag{2.12}$$

The finite difference approximation used in this thesis is given in Appendix B. When the non-hydrostatic pressure is known, the Equations (2.11) can be used to find the corresponding velocity corrections.

2.3 Boundary Conditions

For the Equations (2.10) and (2.12) to be solvable, boundary values must be defined. There is no flow through the bottom or through the closed lateral boundaries, hence $\frac{\partial Q}{\partial \sigma} = 0$ there. For open lateral boundaries, several suggestions have been given, see for instance Marshall et al. (1997b), Kanarska and Maderich (2003), or Berntsen and Furnes (2005). Boundary conditions for open boundaries will not be discussed in this thesis, we will instead focus on the surface.

2.3.1 A Neumann Condition on the Surface

Marshall et al. (1997b) proposed to use the homogeneous Neumann condition $\frac{\partial Q}{\partial \sigma} = 0$ also on the surface. The underlying idea is that the main ocean circulation is taken care of in the hydrostatic part, while the small non-hydrostatic correction does not contribute to changes in the surface elevation. Gravity waves will for instance be neglected in the non-hydrostatic calculation.

Many non-hydrostatic models use a Neumann condition, see for instance Casulli (1999) or Berntsen and Furnes (2005).

The Neumann condition can also put the solvability of the sets of Equations (2.2) and (2.3), or (2.11) in jeopardy, as the principle of conservation of mass comes into play. Both systems are on the form

$$\begin{aligned} \frac{(\mathbf{U}-\tilde{\mathbf{U}})D}{\Delta t} &= \frac{1}{\rho_0}\nabla Q\,, \\ \nabla\cdot(\mathbf{U}D) + \frac{\partial \eta}{\partial t} &= 0\,. \end{aligned}$$

These equations can be combined into

$$\nabla\cdot(\tilde{\mathbf{U}}D) - \frac{\Delta t}{\rho_0}\nabla\cdot(D\nabla Q) + \frac{\partial \eta}{\partial t} = \nabla\cdot(\mathbf{U}D) + \frac{\partial \eta}{\partial t} = 0\,,$$

which gives us the elliptic equation for Q

$$\nabla\cdot(\nabla Q) = \frac{\rho_0}{\Delta t}\Big(\nabla\cdot(\tilde{\mathbf{U}}D) + \frac{\partial \eta}{\partial t}\Big)\,. \tag{2.13}$$

The homogeneous Neumann condition is

$$\mathbf{n}\cdot(\nabla Q) = 0\,,$$

where $\mathbf{n}$ is the outer normal vector. By integrating over the boundary $\partial\Omega$ and using the divergence theorem we get

$$0 = \int_{\partial\Omega}\mathbf{n}\cdot(\nabla Q)\,d\sigma = \int_{\Omega}\nabla\cdot(\nabla Q)\,d\tau = \int_{\Omega}\frac{\rho_0}{\Delta t}\Big(\nabla\cdot(\tilde{\mathbf{U}}D) + \frac{\partial \eta}{\partial t}\Big)\,d\tau\,, \tag{2.14}$$

where we have used Equation (2.13). If the boundary condition for the pressure condition is of the Neumann type for the entire boundary, the integral over the domain of the right hand side of Equation (2.13) must be zero. If this is not fulfilled, the equation will not be solvable.

2.3.2 A Dirichlet Condition on the Surface

There is general agreement that the correct boundary value for the hydrostatic pressure at the surface is

$$P_{Hydrostatic} = P_{Atm} \ ,$$

if we neglect the surface tension. The surface elevation is adjusted so that the pressure at the surface is equal to the atmospheric presure. By the same reasons it may be argued that the non-hydrostatic pressure should have a homogeneous Dirichlet condition at the surface too. This will imply that non-hydrostatic effects can not occur near the surface. This was proposed by Kanarska and Maderich (2003).

2.4 What Is the Right Choice?

The two sets of equations described in this chapter are based on the same physical reasoning; the non-hydrostatic pressure is caused by convergence of water masses into a cell. The difference between the methods is whether we do the physical reasoning in z-coordinates and then transform the equations to σ-coordinates, or if we do it the other way around. The two boundary conditions, on the other hand, model different physical processes.

When we are to choose of a system of equations and a boundary condition on the surface to model the non-hydrostatic pressure, we should be guided by two main principles:

1) The method chosen should model non-hydrostatic phenomena accurately.

2) It should give us a problem which is solvable, and not too computationally expensive.

Of these two criteria, the first will be considered as the most important one. We will investigate if the methods give different solutions, that is if they govern different physical processes. In addition, we will point out advantages and disadvantages of both methods.

Differences in the results obtained by using the distinct sets of equations and boundary conditions will be investigated in Chapter 6. Numerical aspects of the equations and the boundary conditions will be considered in the Chapters 5 and 7.

Chapter 3

Mathematical Background

This chapter contains an overview of different mathematical concepts which we will need later on.

3.1 Matrix Properties

Here we give some results from the theory of matrix analysis. The results will be given without proofs, readers are referred to consult some of the books in the bibliography, e.g. Strang (1988), Golub and van Loan (1996), or Trefethen and Bau (1997). We restrict ourselves to real and quadratic matrices, even though some of the results are valid for complex and non-quadratic matrices too.

3.1.1 Definiteness

A matrix $\mathbf{A} \in \mathbf{R}^{n \times n}$ is said to be positive semidefinite if (Strang 1988)

$$\mathbf{x^T A x} \geq \mathbf{0}\,, \qquad \forall\, \mathbf{x} \in \boldsymbol{R^n}\,,\, \mathbf{x} \neq \mathbf{0}\,.$$

If the inequality is strict, the matrix is positive definite. Negative definiteness and semidefiniteness is defined in the obvious way. A definite matrix is either positive or negative definite.

For an arbitrary matrix, it is necessary that the real part of all the eigenvalues is positive for the matrix to be positive definite. However, unless the matrix is symmetric, it is not sufficient. If some eigenvalues of a symmetric matrix are zero, but all are non-negative, the matrix is positive semidefinite.

3.1.2 The Singular Value Decomposition

Any matrix $\mathbf{A} \in \boldsymbol{R}^{n \times n}$ of arbritrary size and rank can be decomposed into a product

$$\mathbf{A} = \mathbf{E}\boldsymbol{\Sigma}\mathbf{F}^{\mathbf{T}} .$$

This is called the singular value decomposition of the matrix $\mathbf{A}$ (Trefethen and Bau 1997). The matrices $\mathbf{E}$ and $\mathbf{F}$ are unitary and quadratic of size n, and $\boldsymbol{\Sigma} \in \boldsymbol{R}^{n \times n}$ is a diagonal matrix. The diagonal elements in $\boldsymbol{\Sigma}$ are called the singular values of $\mathbf{A}$. The singular values will all be non-negative. It is customary to number them in descending order.

For a symmetric matrix, the singular values of the matrix will be equal to the absolute value of the eigenvalues.

It is possible to show that

$$\|\mathbf{A}\|_2 = \sigma_{\max} .$$

The singular values also give a measure of how close the matrix is from being rank deficient. The number of positive singular values will be equal to the rank of the matrix. If

$$s < r = rank(\mathbf{A}) ,$$

then

$$\min_{rank(\mathbf{B})=s} \|\mathbf{A} - \mathbf{B}\| = \sigma_{s+1} , \tag{3.1}$$

where σ_{s+1} is the s+1st largest singular value (Golub and van Loan 1996).

3.1.3 Condition Numbers

Computations done in finite precision will be affected by rounding errors. A problem is said to be ill-conditioned if the effects of rounding errors are large, and the problem has a large condition number κ (for a precise definition of condition numbers, see Trefethen and Bau (1997)). The condition number of a matrix is defined by

$$\kappa(\mathbf{A}) = \|\mathbf{A}\|\|\mathbf{A}^{-1}\| . \tag{3.2}$$

If the matrix $\mathbf{A}$ is singular, we write $\kappa(\mathbf{A}) = \infty$. If the norm in Equation (3.2) is the 2-norm, the condition number will be given by

$$\kappa(\mathbf{A}) = \frac{\sigma_{\max}}{\sigma_{\min}} , \tag{3.3}$$

where $\sigma_{\max}$ and $\sigma_{\min}$ is the largest and the smallest singular value of $\mathbf{A}$, respectively. We will use 2-norms in this thesis.

3.2 Iterative Methods

We have a linear system of equations $\mathbf{Ax} = \mathbf{b}$ where $\mathbf{A} \in \boldsymbol{R}^{n\times n}$ and $\mathbf{b} \in \boldsymbol{R}^n$ are known while $\mathbf{x} \in \boldsymbol{R}^n$ is unknown. If we discretise a partial differential equation by finite differences, the coefficient matrix $\mathbf{A}$ will be a sparse matrix, with nonzero elements only on some diagonals. For realistic problems in ocean modelling, the equations we must solve can have $n \gg 100000$ (Marshall et al. 1997a). If we are to solve the system directly e.g. by LU factorisation we need of the order $\frac{2}{3}n^3$ operations, and we must store n^2 elements. This is unrealistic; both the size and the sparsity of the matrix calls for an iterative method.

3.2.1 Successive Over-Relaxation

Let $\mathbf{D}$ be the main diagonal of $\mathbf{A}$, let $\mathbf{T}$ and $\mathbf{B}$ be the upper and lower triangular part of $\mathbf{A}$, respectively. If we set

$$\mathbf{G} = (\mathbf{D} + \lambda\mathbf{B})^{-\mathbf{1}}((\mathbf{1} - \lambda)\mathbf{D} + \lambda\mathbf{T}) \,, \tag{3.4}$$

and

$$\mathbf{c} = (\mathbf{D} + \lambda\mathbf{B})^{-\mathbf{1}}\mathbf{b} \,,$$

an iterative method can be written

$$\mathbf{x}_{\mathrm{k}+1} = \mathbf{G}\mathbf{x}_{\mathrm{k}} + \mathbf{c} \,,$$

where $\mathbf{G}$ is a matrix, $\mathbf{c}$ is a vector, and $\mathbf{x}_{\mathrm{k}}$ is a sequence of approximations to the solution vector. This is the method of successive over-relaxation (SOR). A necessary and sufficient condition for convergence of the iterations is that the spectral radius $\rho(\mathbf{G}) < 1$ (Espelid 2003).

The parameter λ is chosen to speed up the convergence. There is a huge literature on how to find the optimal λ, see for instance Young (1971).

3.2.2 The Method of Preconditioned Conjugate Gradients

The basic iterative methods are robust, but their convergence can be slow. In addition, they depend upon parameters that can be difficult to choose properly, for instance the λ in SOR (Golub and van Loan 1996). Other iterative methods are based on Krylov subspaces, that is the subspace of $\boldsymbol{R}^n$ spanned by $\{\mathbf{Ab}, \mathbf{A^2b}, \mathbf{A^3b}, \ldots\}$. The methods search for the best approximation of the solution in the Krylov subspace. If these methods converge, the convergence is usually fast.

We will consider the method of conjugate gradients (CG). This method can be applied when the matrix $\mathbf{A}$ is symmetric and positive definite. The pure method of conjugate gradients works well when the matrix is either well conditioned, or has just a few distinct eigenvalues (Golub and van Loan 1996). In order to speed up the convergence, some preconditioning of the problem is usually done. The derivation below follows Marshall et al. (1997a), though the notation is changed to be in agreement with the common notation for the method of conjugate gradients.

We will solve the linear system of equations

$$\mathbf{Ax} = \mathbf{b} \, .$$

We premultiply with $\mathbf{M}$, which is an approximate inverse of $\mathbf{A}$. The matrix $\mathbf{M}$ is called a preconditioner. This gives us

$$(\mathbf{I} - \mathbf{C})\mathbf{x} = \mathbf{Mb} \, ,$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{C} = \mathbf{I} - \mathbf{MA}$ is close to the zero matrix if $\mathbf{M} \approx \mathbf{A^{-1}}$. An iterative method can then be written

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{Cx}_k + \mathbf{Mb} \\ &= (\mathbf{I} - \mathbf{MA})\mathbf{x}_k + \mathbf{Mb} \\ &= \mathbf{x}_k + \mathbf{M}(\mathbf{b} - \mathbf{Ax}_k) \\ &= \mathbf{x}_k + \mathbf{Mr}_k \, . \end{aligned} \tag{3.5}$$

Here the residual, $\mathbf{r}_k$, is defined by

$$\mathbf{r}_k = (\mathbf{b} - \mathbf{Ax}_k) \, .$$

We also define the vector $\mathbf{p}_k$ by

$$\mathbf{p}_k = \mathbf{Mr}_k \, .$$

This is called the search direction. If the updates of the solution vector and the search directions are chosen as good as possible, we get the following algorithm (Golub and van Loan 1996)

Preconditioned Conjugate Gradients
$\mathbf{x}_0 = \text{initial guess}$
$\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$
$\mathrm{k} = 0$
While $\|\mathbf{r}_\mathrm{k}\| > \epsilon\|\mathbf{b}\|$ **and** $\mathrm{k} <$ maxiter
    $\mathrm{k} = \mathrm{k} + 1$
    if $\mathrm{k} = 1$
        $\mathbf{p}_1 = \mathbf{M}\mathbf{r}_\mathbf{0}$
    else
        $\beta_\mathrm{k} = (\mathbf{r}_{\mathrm{k}-1}{}^T\mathbf{M}\mathbf{r}_{\mathrm{k}-1})/(\mathbf{r}_{\mathrm{k}-2}{}^T\mathbf{M}\mathbf{r}_{\mathrm{k}-2})$
        $\mathbf{p}_\mathrm{k} = \mathbf{M}\mathbf{r}_{\mathrm{k}-1} + \beta_\mathrm{k}\mathbf{p}_{\mathrm{k}-1}$
    end
    $\alpha_\mathrm{k} = (\mathbf{r}_{\mathrm{k}-1}{}^T\mathbf{M}\mathbf{r}_{\mathrm{k}-1})/(\mathbf{p}_{\mathrm{k}-2}{}^T\mathbf{A}\mathbf{p}_{\mathrm{k}-2})$
    $\mathbf{x}_\mathrm{k} = \mathbf{x}_{\mathrm{k}-1} + \alpha_\mathrm{k}\mathbf{p}_\mathrm{k}$
    $\mathbf{r}_\mathrm{k} = \mathbf{r}_{\mathrm{k}-1} - \alpha_\mathrm{k}\mathbf{A}\mathbf{p}_\mathrm{k}$
end
$\mathbf{x} = \mathbf{x}_\mathrm{k}$

3.2.3 The Method of Preconditioned Bi-Conjugate Gradients

The method of conjugate gradients assumes that the matrix $\mathbf{A}$ is symmetric, and is not guaranteed to work if this is not fulfilled (Faber and Manteuffel 1984). As we will see, this will cause problems in some of our simulations. Even if CG is designed for symmetric problems, it is known to work also for mildly asymmetric problems (Avlesen 2006). We will therefore use CG to solve linear systems of equations despite of the lack of symmetry. Because of the possible problems, we will also consider a version of the method of bi-conjugate gradients (Bi-CG). This method works for non-symmetric as well as symmetric matrices. However, the pure Bi-CG algorithm has problems with irregular convergence (Trefethen and Bau 1997). This can in the extreme lead to breakdown of the iteration process. We will instead use a stabilised version of bi-conjugate gradients (Bi-CGSTAB) (van der Vorst 1992). Bi-CGSTAB has a smoother convergence rate than Bi-CG, but it can still suffer from breakdowns. In order to speed up the convergence, we apply preconditioning.

Like CG, Bi-CGSTAB uses an iterative scheme based on Equation (3.5). The algorithm can be found in van der Vorst (1992).

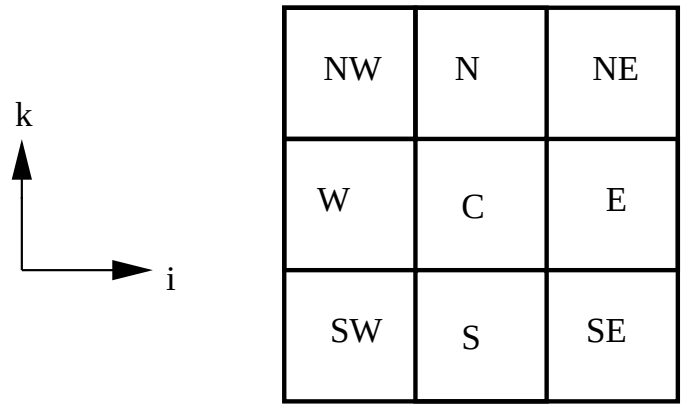


Figure 3.1: *The local index of the point* (i, j).

3.2.4 Preconditioning

A Preconditioner for the Simplified System of Equations

What is left for the iterative methods CG and Bi-CGSTAB is to specify the matrix **M**. The discretization of the simplified system of equations gives us a 5-diagonal matrix. The corresponding linear system for a point with indexes (i, k) can be written

$$a^C_{i,k}x_{i,k} + a^E_{i,k}x_{i+1,k} + a^S_{i,k}x_{i,k+1} + a^W_{i,k}x_{i-1,k} + a^N_{i,k}x_{i,k-1} = b_{i,k}\ , \tag{3.6}$$

where it is understood that $a_{i,k}$ is zero if it is outside the domain. The superscripts are explained in Figure 3.1. For our problem, the matrix in the linear system will be diagonally dominant. Therefore, a first approximation of the solution is

$$x_{i,k} = \frac{b_{i,k}}{a^C_{i,k}}\ .$$

This will hold for all the grid points. Hence, a better approximation to the solution of Equation (3.6) is

$$x_{i,k} = \frac{b_{i,k}}{a^C_{i,k}} - \frac{1}{a^C_{i,k}}\Big(\frac{a^E_{i,k}b_{i+1,k}}{a^C_{i+1,k}} + \frac{a^S_{i,k}b_{i,k+1}}{a^C_{i,k+1}} + \frac{a^W_{i,k}b_{i-1,k}}{a^C_{i-1,k}} + \frac{a^N_{i,k}b_{i,k-1}}{a^C_{i,k-1}}\Big)\ .$$

The product **Mr** is therefore computed by

$$(\mathbf{Mr})_{i,k} = \frac{r_{i,k}}{a^C_{i,k}} - \frac{1}{a^C_{i,k}}\Big(\frac{a^E_{i,k}r_{i+1,k}}{a^C_{i+1,k}} + \frac{a^S_{i,k}r_{i,k+1}}{a^C_{i,k+1}} + \frac{a^W_{i,k}r_{i-1,k}}{a^C_{i-1,k}} + \frac{a^N_{i,k}r_{i,k-1}}{a^C_{i,k-1}}\Big)\ .$$

To be used in the method of conjugate gradients, the preconditioner must be symmetric. Therefore, the product **Mr** in CG is computed by

$$(\mathbf{Mr})_{i,k} = \frac{r_{i,k}}{a^C_{i,k}} - \frac{2}{a^C_{i,k}}\Big(\frac{a^E_{i,k}r_{i+1,k}}{a^C_{i+1,k}+a^C_{i,k}} + \frac{a^S_{i,k}r_{i,k+1}}{a^C_{i,k+1}+a^C_{i,k}} + \frac{a^W_{i,k}r_{i-1,k}}{a^C_{i-1,k}+a^C_{i,k}} + \frac{a^N_{i,k}r_{i,k-1}}{a^C_{i,k-1}+a^C_{i,k}}\Big).$$

This is the preconditioner used in the z-coordinate MIT general circulation model, see Marshall et al. (1997a).

A Preconditioner for the Full Set of Equations

The full set of equations for the non-hydrostatic pressure correction gives us a linear system with a 9-diagonal matrix. This matrix will also be diagonally dominant. Therefore, we design a preconditioner for this linear system by following the work by Marshall et al. (1997a) for a the 5-diagonal system. The linear system in a point is given by

$$\begin{aligned} a^C_{i,k}x_{i,k} + a^{NE}_{i,k}x_{i+1,k-1} + a^E_{i,k}x_{i+1,k} + a^{SE}_{i,k}x_{i+1,k+1} + a^S_{i,k}x_{i,k+1} & \\ + a^{SW}_{i,k}x_{i-1,k+1} + a^W_{i,k}x_{i-1,k} + a^{NW}_{i,k}x_{i-1,k-1} + a^N_{i,k}x_{i,k-1} &= b_{i,k}\,. \end{aligned}$$

Motivated by the same reasoning as for the simplified system, we compute the product **Mr** by

$$\begin{aligned} (\mathbf{Mr})_{i,k} = \frac{r_{i,k}}{a^C_{i,k}} - \frac{1}{a^C_{i,k}}\Big(&\frac{a^{NE}_{i,k}r_{i+1,k-1}}{a^C_{i,k}} + \frac{a^E_{i,k}r_{i+1,k}}{a^C_{i,k}} + \frac{a^{SE}_{i,k}r_{i+1,k+1}}{a^C_{i,k}} + \frac{a^S_{i,k}r_{i,k+1}}{a^C_{i,k}} \\ &+ \frac{a^{SW}_{i,k}r_{i-1,k+1}}{a^C_{i,k}} + \frac{a^W_{i,k}r_{i-1,k}}{a^C_{i,k}} + \frac{a^{NW}_{i,k}r_{i-1,k-1}}{a^C_{i,k}} + \frac{a^N_{i,k}r_{i,k-1}}{a^C_{i,k}}\Big)\,, \end{aligned}$$

for the full set of equations.

Again, **M** must symmetric to be used in CG. Therefore the preconditioning in CG is done by

$$\begin{aligned} (\mathbf{Mr})_{i,k} = \frac{r_{i,k}}{a^C_{i,k}} - \frac{2}{a^C_{i,k}}\Big(&\frac{a^{NE}_{i,k}r_{i+1,k-1}}{a^C_{i+1,k-1}+a^C_{i,k}} + \frac{a^E_{i,k}r_{i+1,k}}{a^C_{i+1,k}+a^C_{i,k}} + \frac{a^{SE}_{i,k}r_{i+1,k+1}}{a^C_{i+1,k+1}+a^C_{i,k}} \\ &+ \frac{a^S_{i,k}r_{i,k+1}}{a^C_{i,k+1}+a^C_{i,k}} + \frac{a^{SW}_{i,k}r_{i-1,k+1}}{a^C_{i-1,k+1}+a^C_{i,k}} + \frac{a^W_{i,k}r_{i-1,k}}{a^C_{i-1,k}+a^C_{i,k}} \\ &+ \frac{a^{NW}_{i,k}r_{i-1,k-1}}{a^C_{i-1,k-1}+a^C_{i,k}} + \frac{a^N_{i,k}r_{i,k-1}}{a^C_{i,k-1}+a^C_{i,k}}\Big)\,. \end{aligned}$$

3.2.5 A Stopping Criterion

For the iterative methods we need a stopping criterion. For SOR we define the residual by

$$\mathbf{r}_k = \mathbf{b} - \mathbf{A}\mathbf{x}_k \, . \tag{3.7}$$

This is the same as the vector $\mathbf{r}_k$ in CG and Bi-CGSTAB. The iterations are stopped when

$$\frac{\|\mathbf{r}_k\|}{\|\mathbf{b}\|} < \epsilon \, , \tag{3.8}$$

where ϵ is some tolerated value of the error. In order to avoid infinite loops if convergence problems occurs, we only allow a limited number of iterations.

3.3 Discretizations of Elliptic Equations

When we compute the non-hydrostatic pressure correction, we end up with an elliptic differential equation. For this problem to be well posed, we must have boundary conditions, where at least some part of the boundary has a Dirichlet condition. That is the value of the dependent variable is specified there. If the entire boundary has conditions of the Neumann type where the normal derivative is specified, the solution will in general not be unique (Gilbarg and Trudinger 1998). To any solution, we can add an arbitrary constant and get another solution. This extra degree of freedom can be removed by imposing an additional constraint on the system. We will force the mean non-hydrostatic pressure to be equal to zero. We are only interested in the derivatives of the pressure, which translates to differences between the pressure in the cell centres for the discrete solution. Therefore, the additional constraint has no effect on the solution.

However, the non-uniqueness of the solution to the continuous problem leads to a singular coefficient matrix for the discrete problem. One eigenvalue will be equal to zero. It might still be possible to get an approximate solution of the corresponding linear system. The condition number of the matrix will be infinite, though. Motivated by the Equations (3.1) and (3.3) we define a modified condition number by

$$\kappa_m(\mathbf{A}) = \frac{\sigma_{\max}}{\sigma_{\min-1}} \, , \tag{3.9}$$

where $\sigma_{\min-1}$ is the second smallest singular value of $\mathbf{A}$. We will discuss the singularity of the coefficient matrices further in Chapter 5.

3.4 Energy

3.4.1 Computation of Energy

The energy in the ocean consists of kinetic, potential and internal energy. The kinetic energy, E_K, is given by

$$E_K = \int_{x=x_1}^{x=x_2} \int_{z=-H(x)}^{z=\eta(x)} \frac{1}{2}\rho(U^2 + W^2)\, dzdx\,, \tag{3.10}$$

where x_1 and x_2 is the left and the right boundary of the domain, respectively. The potential energy, E_P, is computed by

$$E_P = \int_{x=x_1}^{x=x_2} \int_{z=-H(x)}^{z=\eta(x)} \rho g z\, dzdx + E_{P_0}\,. \tag{3.11}$$

Here, E_{P_0} is a reference level for the potential energy. In this thesis we set

$$E_{P_0} = -\int_{x=x_1}^{x=x_2} \int_{z=-H(x)}^{z=\eta(x)} \bar{\rho} g z\, dzdx\,,$$

where $\bar{\rho}$ is the average density. If the integrals with respect to z in Equation (3.11) are taken from 0 to η, we get the potential energy due to the surface elevation.

The potential energy can be divided into available potential energy (APE) and background potential energy (BPE) (Peltier and Caufield 2003). The BPE is defined as the energy of the system at rest with the density layers sorted so that heavier fluid particles are located below lighter particles. The BPE will change as the water masses are mixed. The APE is the potential energy minus the background potential energy. See e.g. Gill (1982) for more on available potential energy.

The kinetic and potential energy are computed by the Equations (3.10) and (3.11), respectively. The integrals are replaced by summations. The summations are taken over the same cells for all time steps. This gives a smooth time evolution for both the kinetic and the potential energy.

The background potential energy is more complicated to compute because we must sort the water masses with respect to density. First, we construct a grid which allow us to compute the volume of the domain we are interested in with an error less than some tolerance. Next, the water masses are sorted, and the BPE is computed on the grid constructed for the integration. We also get an estimate of the error. If the error is larger than a given tolerance the accuracy of the sorting is increased. The resulting integration method is automatic and adaptive. Because of the adaptivity, the grid points used for

the integration are not necessarily the same from one time step to the next, nor are the densities used in the computations. This may lead to noisy data if the BPE is plotted as a function of time. Because of this, we will smooth the data by using time averaging.

3.4.2 Transfer of Energy

According to Peltier and Caufield (2003), there are several sources of changes in the potential energy. There can be an exchange between available potential energy and kinetic energy because of buoyancy fluxes. Fluid motion can also increase the potential energy by stirring and mixing. Stirring is defined as lifting of dense water up to lighter water without any diffusion taking place. The diffusion caused by stirring is called mixing. There will also be diffusion that will take place without any macroscopic motion. Stirring is a reversible process which will increase the available potential energy. Diffusion and mixing are irreversible, hence they increase the background potential energy.

Chapter 4

The Model Problem

In Chapter 2 we discussed systems of equations and boundary conditions to model the non-hydrostatic pressure in σ-coordinate models. In this chapter we will describe the setup of a numerical simulation, which we will use later to explore the differences in the solutions we get from the distinct methods. We will consider a solitary wave hitting a slope and propagating up the incline. Non-hydrostatic effects are important for the small scale dynamics in this process (Legg and Adcroft 2003). Also the wave approaches the surface as it propagates up the slope. Thus the impact of the boundary condition on the surface will hopefully be seen. The model setup is based on an experiment described in Berntsen et al. (2006). Their experiment is again based on experiment number 12 in Michallet and Ivey (1999).

4.1 The Geometry

The experiments are done in a tank with length $L = 1.72$ m. The depth is $H = 0.15$ m in the left part of the tank. In the right part, the depth is decreasing. At the very right end of the tank, the depth is zero, see Figure 4.1.

For the simulations done in Chapter 6, the slope starts 0.70 m from the right end of the tank. This gives a slope factor of 0.214.

There are 100 σ-layers vertically. In the horizontal there are 800 interior grid cells. The spacing is equidistant both in the horizontal and in the vertical direction. In order to avoid too thin σ-layers, points with a depth of less than 0.01 m are defined as land. Thus the very right part of Figure (4.1) is land.

We apply no-flow boundary conditions on the bottom and on the lateral boundaries.

For the analysis of the linear systems done in Chapter 5, both the starting

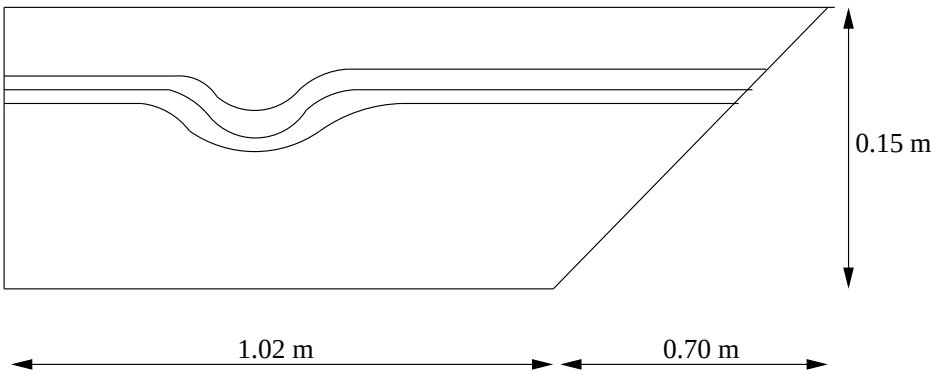


Figure 4.1: *The tank. The shape of the density interface is indicated.*

point of the slope and the number of grid cells are varying.

4.2 The Initial Conditions

Initially both the surface elevation and the velocities are zero. The initial density is calculated by

$$\rho(x,z) = \frac{\Delta\rho}{2}\Big(1 + \tanh\Big(\frac{z - z_i - \zeta}{\Delta h}\Big)\Big) \,. \tag{4.1}$$

Here the density difference $\Delta\rho = 12$ kg m^{-3} , the interface depth $z_i = 0.024$ m , and the thickness of the interface Δh equals 0.014 m. The interface displacement near the left wall is calculated by

$$\zeta = 2a_0 \mathrm{sech}^2\Big(\frac{x}{2W_w}\Big) \,,$$

where $a_0 = 0.026$ m and x is the distance from the left wall. The width of the wave, W_w, is computed from

$$a_1 W_w^2 = \frac{4(h_1 h_2)^2}{3(h_2 - h_1)} \,,$$

where $h_1 = 0.024$ m, $h_2 = 0.146$ m, and $a_1 = 0.031$ m (Berntsen et al. 2006). The values are based on KdV theory, and chosen in order to minimise the initial stirring and mixing.

The reference density used in the Equations (1.1) and (1.2) is set to $\rho_0 = 1024.8$ kg m^{-3}. The bottom roughness parameter z_0 is set to 0.0005.

The time step in the simulations is set to $\Delta t = 0.0025$ seconds.

4.3 The Internal Pressure Errors

When steep topographies are combined with density variations, false velocities appears in σ-coordinate models (Haidvogel and Beckmann 1999). In order to quantify these velocities, the model was set up with the parameter ζ in Equation (4.1) equal to zero. This created a flat density profile, hence there should be no motion at all. The model was run for 60 s. The viscosities and diffusivities were all set to 10^{-6} m s^{-1} , except from $A_{M_{2D}}$, which was $5 \cdot 10^{-5}$ m s^{-1}. The maximal horizontal velocity in this simulation was 0.00158 m s^{-1} , the minimal horizontal velocity was -0.00073 m s^{-1}. These values were obtained after 13.0 seconds and 15.2 seconds, respectively. 35 seconds after the simulation was started, the maximal horizontal velocity was 0.00137 m s^{-1} , while the minimal horizontal velocity was -0.00071 m s^{-1}.

Chapter 5

Analysis of the Matrices

In this chapter we analyse some properties of the coefficient matrices derived in Chapter 2 and Appendix B. We will see if the matrices are symmetric, and give estimates of the qualitative behaviour of their eigenvalues and singular values. See Section 3.1 for definitions of matrix properties.

5.1 Motivation

When we discretise the elliptic Equations (2.10) or (2.12), we get a quadratic coefficient matrix. The number of rows and columns is the number of grid points in the x-direction multiplied by the number of grid points in the z-direction. For the model problem described in Chapter 4, the number of rows and columns are nearly 80.000. For such a large matrix, it is impossible to compute eigenvalues and condition numbers by traditional methods. In order to get some information, we will instead make a coarse grid for the tank described in Chapter 4. If we do simulations using this grid, the results will be inaccurate. However, we hope that the properties we find for the coarse resolution are also valid for finer grids.

We will also see if the coefficient matrices are symmetric. This is computationally cheap, and can be done with the same resolution as we will use in Chapter 6.

Because of the coarse resolution, any result of a simulation will be inaccurate. Therefore the matrices will be based on the situation when the simulation is started. Thus we will not see the effects of spatial changes in the surface elevation in Equation (2.10). As will be shown in Chapter 6, the surface elevation is very small in this simulation. Therefore, the surface elevation hopefully has negligible impact on the coefficient matrix.

5.2 Symmetry

Lin and Li (2002) claims the coefficient matrix to be symmetric. In order to check this, we compute $\mathbf{A} - \mathbf{A}^T$ for the coefficient matrices in the first time step with 800 and 100 grid points in the horizontal and the vertical direction, respectively. If $\mathbf{A}$ is symmetric, this should give us zero matrices. However, the maximal absolute value of the difference is $1.40 \cdot 10^{-5}$ for the simplified system and $2.43 \cdot 10^{-2}$ for the full set of equations. Both these numbers are valid for both boundary conditions. This contradicts the findings of Lin and Li (2002). Our results are, on the other hand, in accordance with Marshall et al. (1997a).

The reason that the matrices are non-symmetric is the changing depth. The full set of equations has more terms which contain derivatives of the depth than the simplified system. This can explain why the asymmetry is larger for the full system of equations. The values given here are from the first time step, when the surface elevation is zero. Hence, we can expect the matrices to be more asymmetric later in the simulation, when we also get contributions from $\frac{\partial \eta}{\partial x}$.

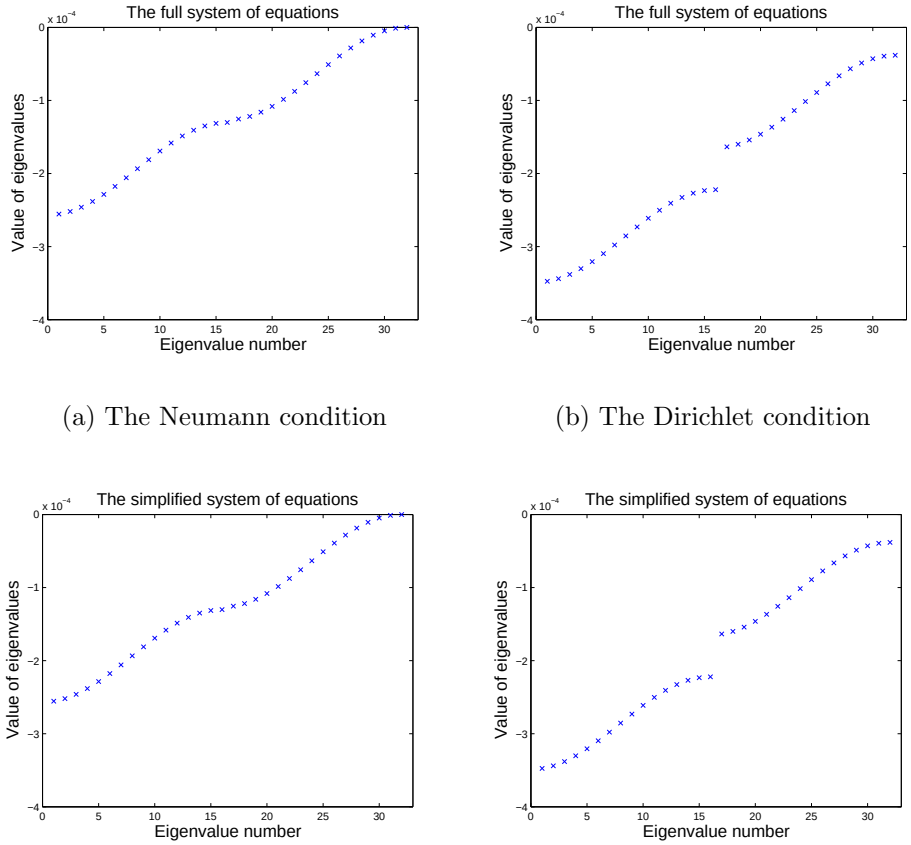
The lack of symmetry can give problems if we solve the linear systems with CG. However, as mentioned in Chapter 3 the method might work, despite $\mathbf{A}$ being non-symmetric. The price we pay is a slow convergence rate compared to the performance of CG for symmetric matrices. SOR and Bi-CGSTAB, on the other hand, are designed for non-symmetric matrices. Therefore, the methods should work well.

We should remember that the discretization is not unique. It is possible that other approaches give different symmetry properties of the matrices regarding symmetry.

5.3 Eigenvalues

We will find the eigenvalues of the coefficient matrices in four different cases. We will consider both the case of a flat bottom, and a slope like the one in the model problem described in Chapter 4. For the flat bottom, the σ-coordinates reduces to Cartesian coordinates, and the two sets of equations for the non-hydrostatic pressure are equal. We can then focus on the differences caused by the boundary conditions. The case with a slope equal to the one used in the model problem will give us insight to the matrix properties for the real test case.

We will use two different grids for the analysis. One grid is very coarse, with 16 grid cells in the horizontal direction and 2 layers in the vertical



(a) The Neumann condition

(b) The Dirichlet condition

(c) The Neumann condition

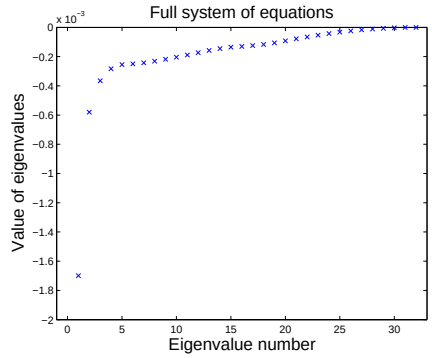
(d) The Dirichlet condition

Figure 5.1: *The eigenvalues of the matrices for a coarse grid and a flat bottom.*

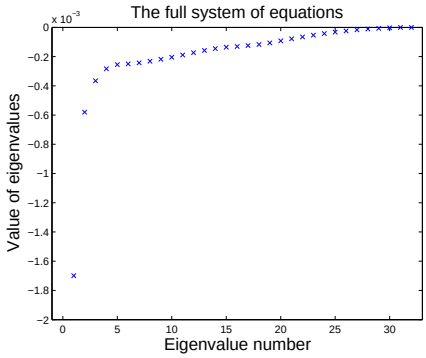
direction. The second grid has 160 points in the x-direction, and 20 points in the z-direction. Both grids have 8 times as many points in the horizontal direction as in the vertical direction. This is the same as is used in the real simulations.

5.3.1 Flat Bottom and Coarse Grid

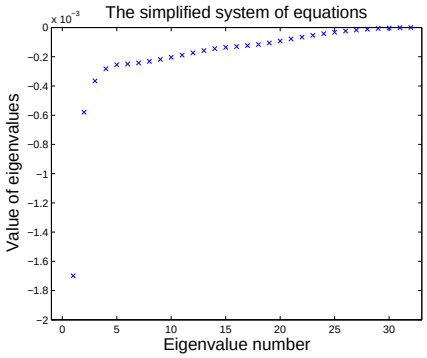
The eigenvalues for the coefficient matrices are shown in Figure 5.1. We first note that all the eigenvalues are real and non-positive. The differences between the eigenvalues from the discretizations of the Equations (2.10) and (2.12) are numerically zero, as they should be. However, the difference between the distinct boundary conditions is striking. When we use a Neumann condition, one eigenvalue is identically zero. If we instead use a



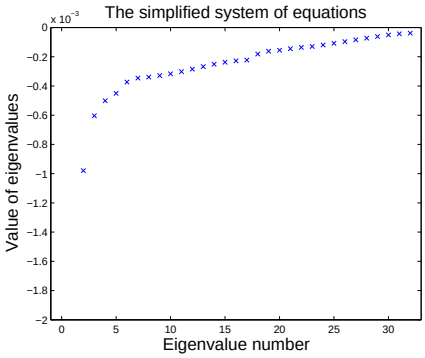
(a) The Neumann condition



(b) The Dirichlet condition



(c) The Neumann condition



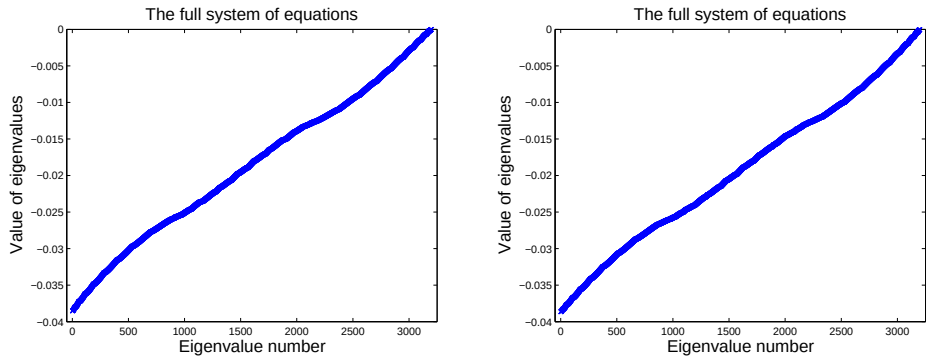
(d) The Dirichlet condition

Figure 5.2: *The eigenvalues of the matrices for a coarse grid and a slope of 0.214.*

Dirichlet condition, all the eigenvalues are negative, and their absolute value is larger than the eigenvalues for the Neumann condition.

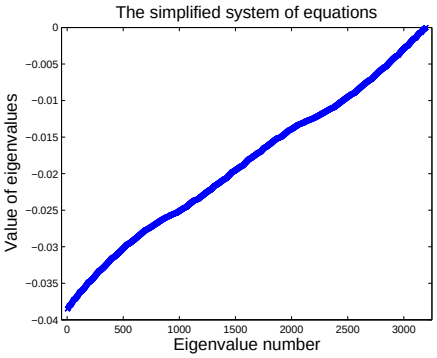
5.3.2 Non-zero Slope and Coarse Grid

Next we consider a slope of 0.214, that is the same as is used in the model problem. The eigenvalues of the coefficient-matrices are shown in Figure 5.2. If we sort the eigenvalues with respect to their value, the largest difference between the full system and the simplified version for the Neumann condition is $4.4 \cdot 10^{-7}$. We still get one eigenvalue equal to zero when we apply the Neumann condition. For the Dirichlet condition, the largest difference is $1.9 \cdot 10^{-6}$. The largest negative eigenvalue is closer to zero for the Dirichlet condition than for the Neumann condition.

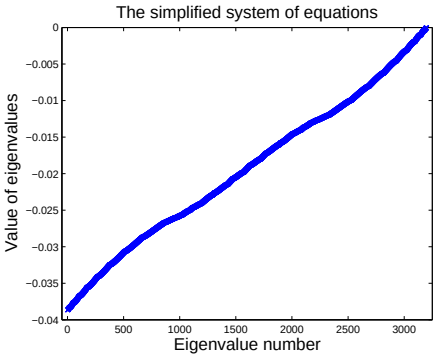


(a) The Neumann condition

(b) The Dirichlet condition



(c) The Neumann condition



(d) The Dirichlet condition

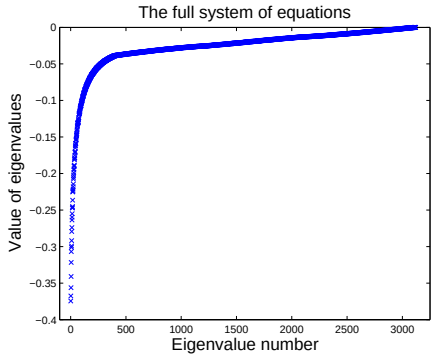
Figure 5.3: *The eigenvalues of the matrices for a finer resolution and a flat bottom.*

5.3.3 Finer Grid, Flat Bottom

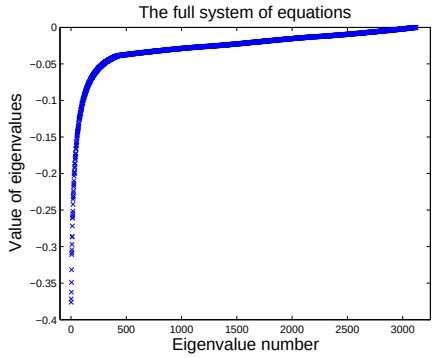
The results for the finer resolution with flat bottom are shown in Figure 5.3. The pattern is the same as for the coarse grid above. The difference between the full system and the simplified one is numerically zero. With the Neumann condition, we get one eigenvalue equal to zero. The Dirichlet condition only has negative eigenvalues.

5.3.4 Finer Grid, nonzero Slope

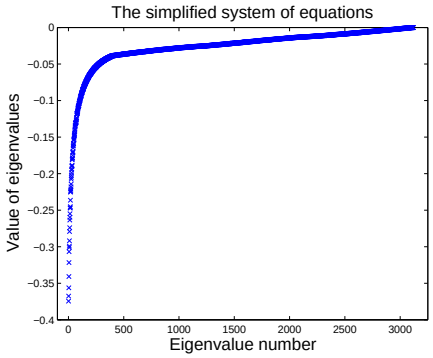
The results for the finer resolution with a slope are shown in Figure 5.4. All the eigenvalues are real and non-positive. The largest negative eigenvalues for both the full system of equations and the simplified version are approx-



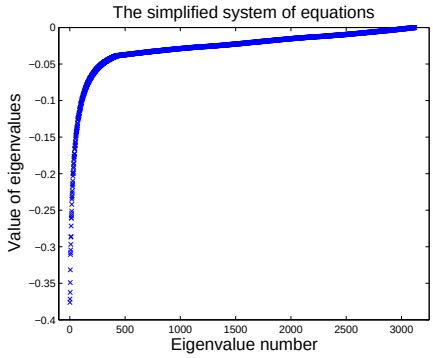
(a) The Neumann condition



(b) The Dirichlet condition



(c) The Neumann condition



(d) The Dirichlet condition

Figure 5.4: *The eigenvalues of the matrices for a finer resolution and a slope of 0.214.*

imately -0.3762 and -0.3745 for the Dirichlet and the Neumann condition, respectively. For both systems of equations, all the eigenvalues for the Neumann condition are slightly closer to zero than for the Dirichlet condition.

5.3.5 Discussion

According to Lin and Li (2002), the coefficient matrices we get from discretizing the equations for the non-hydrostatic pressure equation is definite. Since they claim the coefficient matrices to be symmetric, this implies that all the eigenvalues must have the same sign. We have found that all the eigenvalues are negative for the Dirichlet surface condition, and non-positive for the Neumann case. Thus the eigenvalues of our matrices have the same properties

as the eigenvalues of Lin and Li (2002). However, since the matrices are not symmetric, we cannot tell if the matrices are definite.

Whether the matrices are definite or not is important for our choice of solver for the linear systems. It the matrices have this property, we can solve the linear systems by CG and Bi-CGSTAB. Thus, it seems that we can use these methods if we apply a Dirichlet condition on the surface, but not if we use a Neumann condition. However, the method of conjugate gradients is applied for the non-hydrostatic elliptic equation, also with the Neumann condition on the surface, see e.g. Marshall et al. (1997a) or Casulli (1999). In an eigenvalue expansion of the true solution, the eigenvector corresponding to the zero eigenvalue will in general be present. This part of the solution will, however, not be captured by the iterative method. Thus if the eigenvector corresponding to the zero eigenvalue gives a large contribution to the solution, we cannot expect the solution obtained by the iterative method to be accurate. In addition, there is a risk of breakdown or slow convergence due to rounding errors if the product $\mathbf{p}_{k-2}{}^T\mathbf{A}\mathbf{p}_{k-2}$ is close to zero. For the same reason, Bi-CGSTAB can have problems when the matrix $\mathbf{A}$ is not definite. We will investigate this further in Chapter 7.

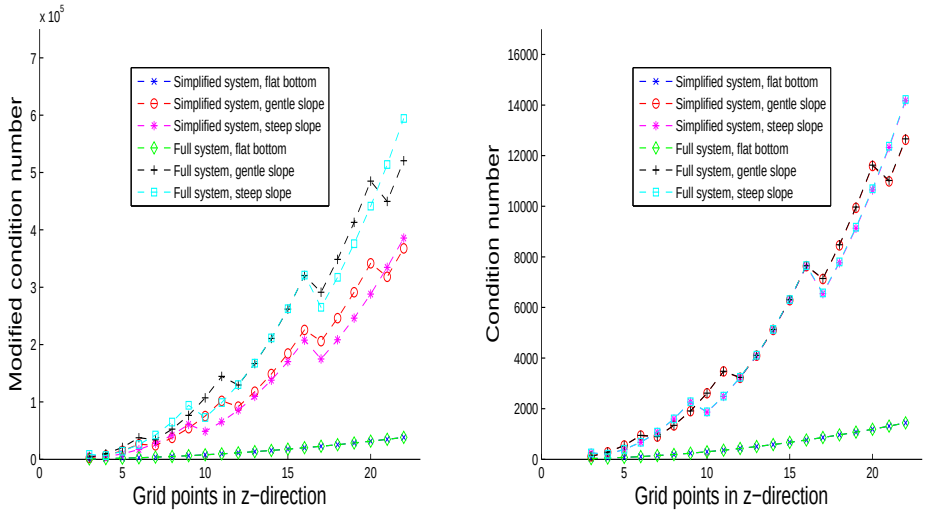
As explained in Section 3.3, it is the gradients of the pressure field which are important to us. Thus if we do succeed in finding a solution with an iterative method, the zero eigenvalue will not affect the density or the velocity fields.

We should keep in mind that the discretizations by finite differences are not unique. It is possible that other ways of discretizing the equations give matrices with different properties. Also, the results are only valid for a flat bottom, or a flat bottom combined with a constant slope as shown in Figure 4.1. Other bottom topographies can give different results.

5.4 Condition Numbers

In order to investigate the condition numbers of the matrices, we use a grid with 8 times as many points in the x-direction as in the z-direction. We start with 3 grid points in the z-direction, and increase the resolution until we have 22 points in the vertical direction. Three different bottom topographies are considered: One with flat bottom, one with a gentle slope of 0.214, and one with a really steep slope of 0.3. These slopes are steeper than what is common in the ocean. However, we choose to focus on these inclines in order to highlight the differences between the coefficient matrices.

When we apply the Neumann condition on the surface, the matrices becomes non-invertible, and the condition numbers are infinite. Therefore, we



(a) The Neumann condition

(b) The Dirichlet condition

Figure 5.5: *The condition numbers of the matrices. For the Neumann surface condition, a modified condition number is used. The number of points in the x-direction is 8 times the number of points in the z-direction.*

use the modified condition number defined in Equation (3.9).

5.4.1 Results

The condition number of the coefficient matrices for the refined grids are shown in Figure 5.5.

For all the matrices, the condition numbers usually increases as the number of grid points increases. However, for the bottom topographies with a slope the condition number sometimes decreases when the grid is refined. This is because the refined grid has grid points with a depth less than 0.01 m. These are removed, and the dimension of the resulting matrix is smaller than for the previous coarser grid. This gives a smaller condition number.

For the flat bottom, the matrices with equal boundary conditions are equal, thus the condition numbers are the same.

For the cases with a slope, we get some interesting results. The condition number for the Neumann condition is one to two orders of magnitude larger than for the Dirichlet condition.

The condition numbers for the Dirichlet condition are almost equal for the full and simplified systems of equations. For the Neumann condition,

the simplified system gives a better posed problem than the full system of equations. The difference between the systems increases as the number of grid points increases. It is probably much larger for the grid used in the real simulations.

The condition numbers increase with increasing slope.

5.4.2 Discussion

We obtain larger condition numbers for the Neumann than for the Dirichlet boundary condition. This was to be expected, since the smallest eigenvalues for the Neumann condition are closer to zero than for the Dirichlet condition. Also, when we apply the Dirichlet condition on the surface, we get matrices which have larger values on the main diagonal. The matrices are more diagonally dominant, hence they are better posed.

The simplified system gives 5-diagonal matrices, whereas the full system leads matrices with 9 nonzero diagonals. Hence, the simplified system can be expected to give more diagonally dominant matrices, and therefore better posed problems. This is what we see for the Neumann condition on the surface. We would expect the same results for the Dirichlet condition. However, this is surprisingly not the case. The reason might be that the singularity of the matrices due to the Neumann condition makes the conditioning much more sensitive to the degree of diagonal dominance. When the matrices are invertible, as it is in the Dirichlet case, the diagonal dominance is not that important.

The increase in the condition number with larger slopes can be explained by larger derivatives of the depth, which gives larger off-diagonal elements in the matrices. Also, the discontinuity in the derivative of the depth where the slope starts will be larger for a steeper slope. This may give a worse posed problem.

5.5 Summary

The eigenvalues of the coefficient matrices are real and non-positive for all four methods. If a Neumann condition is applied on the surface, one eigenvalue equals zero, and the small eigenvalues are closer to zero than for the Dirichlet condition. The zero eigenvalue can give problems if the linear system is solved by the method of conjugate gradients and bi-conjugate gradients. Also, the method of conjugate gradients might get problems because of the weak asymmetry of the matrices.

From our analysis of condition numbers, we can expect the linear system from the simplified set of equations to be easier to solve numerically than the full set. We can also expect better posed problems when we apply a Dirichlet boundary condition on the surface, than when we use a Neumann condition.

Chapter 6

Numerical Experiments

In the previous chapters, we have presented four methods for modelling the non-hydrostatic pressure correction. Here we will investigate the differences in the results obtained by using the four methods on a test case. We will use both the full set of equations and the simplified version. On the surface we apply both a Dirichlet and a Neumann condition. To compare the different methods, we will use both the density and the velocity fields and energetics.

6.1 The Numerical methods

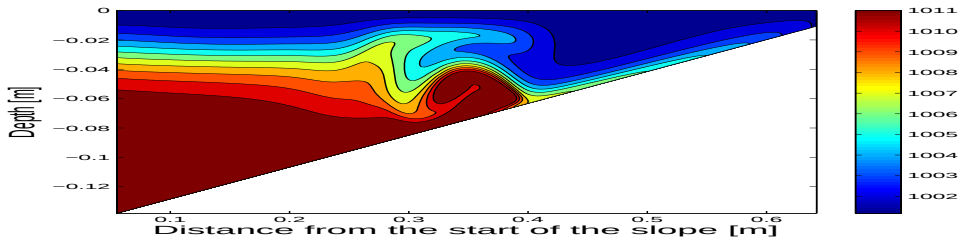
The linear system for the non-hydrostatic pressure correction is solved by using SOR. The solution from the previous sequence of iterations is used as an initial guess. For convergence, we demand that the relative error defined in Equation 3.8 is less than $5 \cdot 10^{-6}$. For more details on the numerical methods, see Section 1.5.1 and the references there.

6.2 The Primary variables

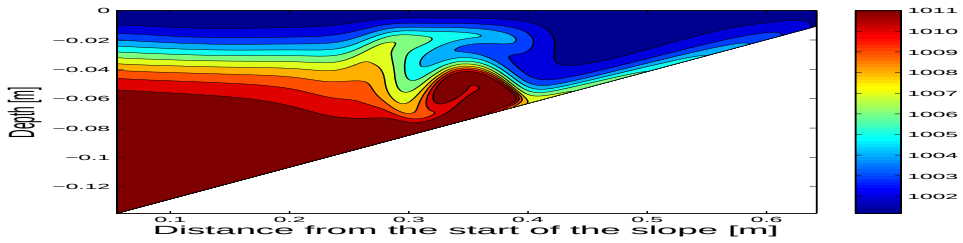
The simulations are done with the coefficients A_M, K_M, A_H, and K_H all equal to $2 \cdot 10^{-6}$ m^2 s^{-1}. The viscosity in the depth averaged calculation, $A_{M_{2D}}$, is set to $5 \cdot 10^{-5}$ m^2 s^{-1}.

The density and velocity profiles were taken at 32, 35, and 37 seconds. The differences between the profiles from the distinct ways to model the non-hydrostatic pressure were largest at 35 seconds. Therefore, all the plots shown were taken 35 seconds after the simulation started.

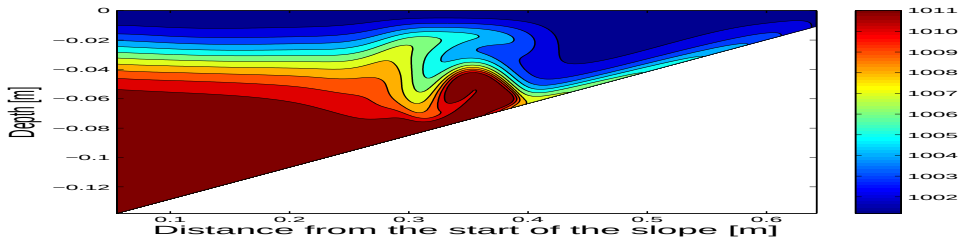
When the wave hits the slope, it starts to move upwards in a bolus. As the bolus propagates upwards, the water in front of it moves downwards. At the front of the bolus, this water separates from the bottom and flows over



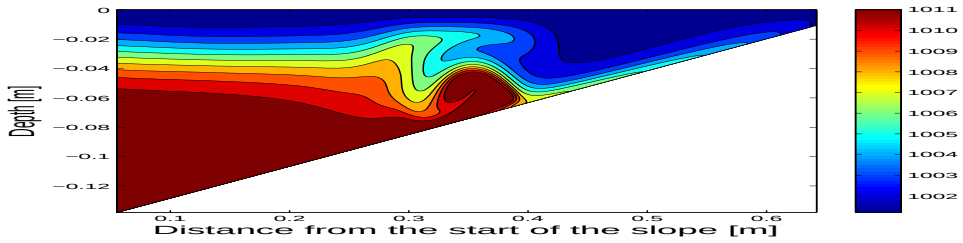
(a) The full system with a Neumann condition on the surface



(b) The full system with a Dirichlet condition on the surface

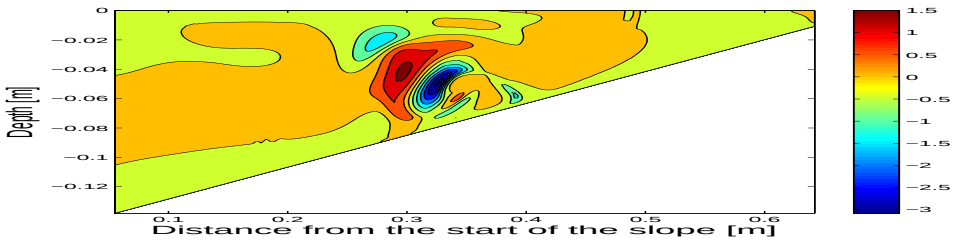


(c) The simplified system with a Neumann condition on the surface

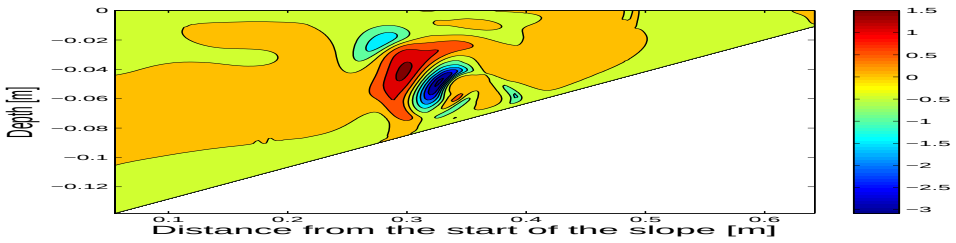


(d) The simplified system with a Dirichlet condition on the surface

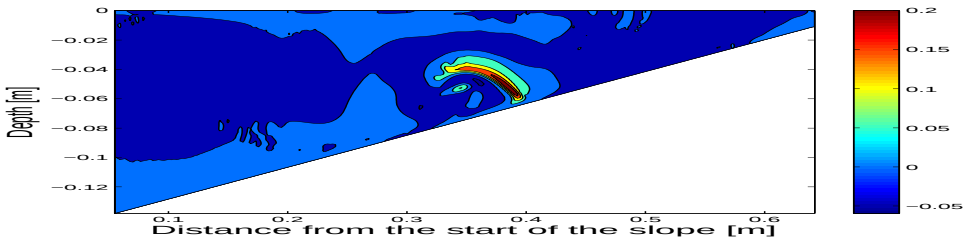
Figure 6.1: *The density distributions after 35 seconds.*



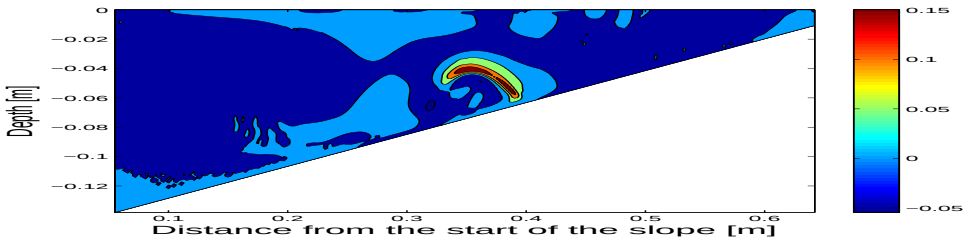
(a) The simplified minus the full system with a Neumann condition



(b) The simplified minus the full system with a Dirichlet condition



(c) The Neumann minus the Dirichlet condition for the full system



(d) The Neumann minus the Dirichlet condition for the simplified system

Figure 6.2: *Differences in the density distributions after 35 seconds.*

the bolus. The water masses are mixed behind the head front of the wave. Eventually, the wave breaks, and the mixing increases. See Bourgault et al. (2005) for plots of the time evolution of the breaking of an internal solitary wave.

6.2.1 The Density

The density distributions after 35 seconds are shown in Figure 6.1. From these figures, there are apparently no large differences between the distinct methods. The separation point is located at the same place in all the figures. The bolus has a somewhat smaller lateral extention for the simplified system than for the full system. For the full set of equations, the wave seems closer to breaking than for the simplified system.

In the Figures 6.2(a) and 6.2(b), the density distribution for the simplified set of equations minus the distribution for the full system are shown for the Neumann and the Dirichlet condition on the surface, respectively. The largest absolute value of the difference in density is about 3 kg m^{-3} both for the Neumann and the Dirichlet condition on the surface.

The Figures 6.2(c) and 6.2(d) show the density distribution obtained by using the Neumann condition minus the density for the Dirichlet condition for the full and the simplified set of equations, respectively. The front of the dense water has propagated longer for the Neumann boundary condition than for the Dirichlet condition. However, the differences due to the boundary conditions are much smaller than the differences due to the systems of equations. The maximal absolute value of the difference is about 0.15 kg m^{-3} and 0.2 kg m^{-3} for the simplified and the full set of equations, respectively.

6.2.2 The Horizontal Velocity

The horizontal velocity fields are shown in Figure 6.3. The Figures 6.4(a) and 6.4(b) show the velocity fields for the simplified minus the full set of

Set of equations	Boundary condition	Maximal velocity [m s^{-1}]	Maximal negative velocity [m s^{-1}]
Full	Neumann	0.0443	-0.0350
Full	Dirichlet	0.0440	-0.0305
Simplified	Neumann	0.0435	-0.0285
Simplified	Dirichlet	0.0435	-0.0285

Table 6.1: *The maximal horizontal velocities after 35 seconds.*

Set of equations	Boundary condition	Position after 32 s [m]	Position after 37 s [m]
Full	Neumann	0.3560	0.4592
Full	Dirichlet	0.3560	0.4571
Simplified	Neumann	0.3560	0.4571
Simplified	Dirichlet	0.3560	0.4571

Table 6.2: *The location of the separation point after 32s and 37s. The distance is measured from the start of the slope.*

equations for the Neumann and Dirichlet boundary condition, respectively. The full set of equations leads to larger velocities than the simplified one. The absolute value of the maximal difference is about 0.0012 $\mathrm{m\,s^{-1}}$ for both boundary conditions. The differences due to the boundary conditions are much smaller than the differences due to the set of equations, as can be seen in the Figures 6.4(c) and 6.4(d).

In Table 6.1 the maximal positive and negative velocities are given. The maximal positive velocities are almost the same, while the maximal negative velocities are larger for the full system of equations.

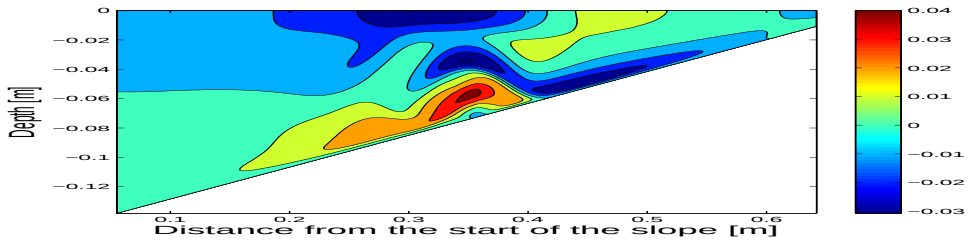
In Table 6.2 the location of the separation point after 32 seconds and 37 seconds are given. There are no differences at all after 32 seconds. After 37 seconds the separation point has propagated one grid cell longer up the slope with the full set of equations and a Neumann boundary condition than for the other methods. The average speed of the separation point between 32 and 37 seconds is 0.021 $\mathrm{m\,s^{-1}}$ with the full set of equations and a Neumann condition. For the other methods, the average speed is 0.020 $\mathrm{m\,s^{-1}}$. Thus both the location and the speed of the separation point are robust to the different methods to model the non-hydrostatic pressure.

6.2.3 The Vertical Velocity

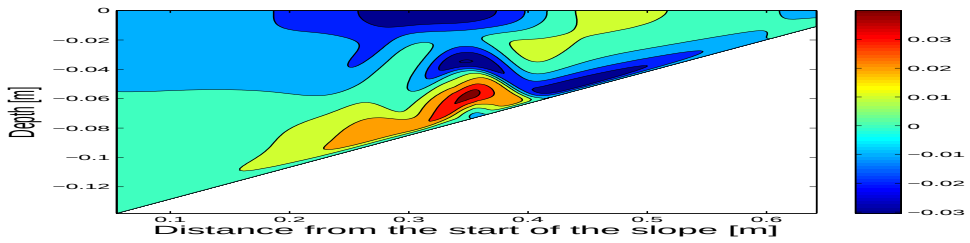
The vertical velocity fields after 35 seconds are shown in Figure 6.5. Overall, the full system of equations gives somewhat higher velocities than the simplified system, although the differences are small.

The vertical velocity field for the simplified set of equations minus the field for the full set is shown in Figure 6.6(a) for the Neumann condition, and in Figure 6.6(b) for the Dirichlet boundary condition. The largest absolute value of the difference is about 0.01 $\mathrm{m\,s^{-1}}$ for both boundary conditions.

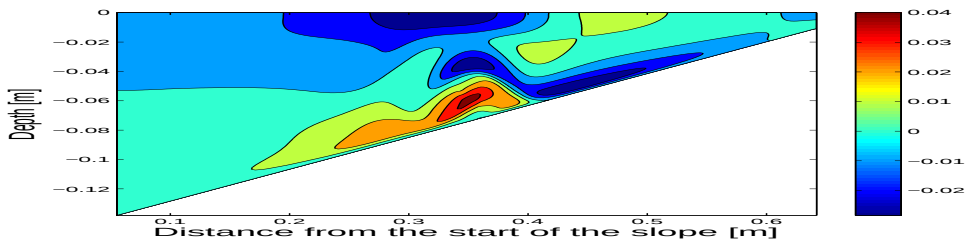
Like the results for the density and horizontal velocity, the differences in vertical velocity due to the boundary conditions are much smaller. The maximal values are $1.2 \cdot 10^{-3}\,\mathrm{m\,s^{-1}}$ for the full system, and $4 \cdot 10^{-4}\,\mathrm{m\,s^{-1}}$



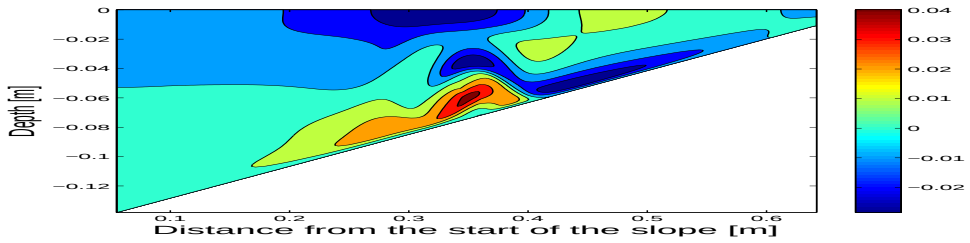
(a) The full system with a Neumann condition on the surface



(b) The full system with a Dirichlet condition on the surface

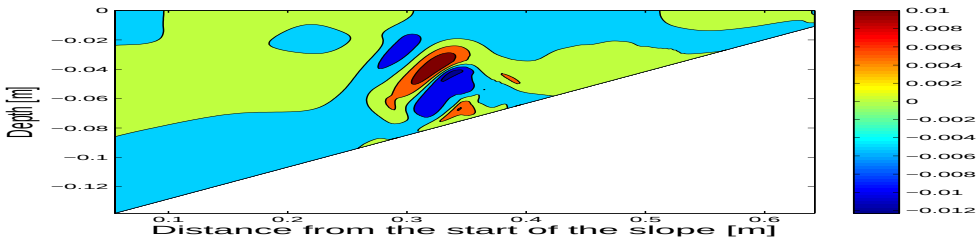


(c) The simplified system with a Neumann condition on the surface

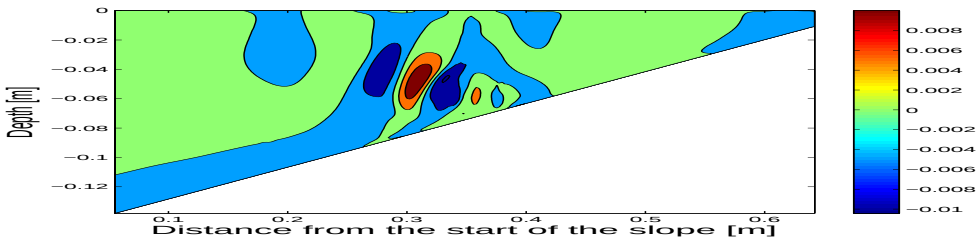


(d) The simplified system with a Dirichlet condition on the surface

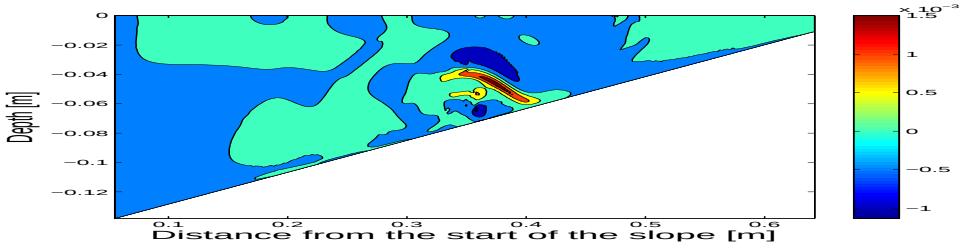
Figure 6.3: *The horizontal velocities after 35 seconds.*



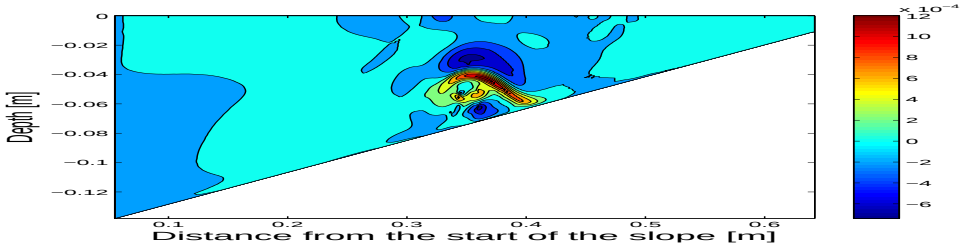
(a) The simplified minus the full set of equations with a Neumann condition



(b) The simplified minus the full set of equations with a Dirichlet condition

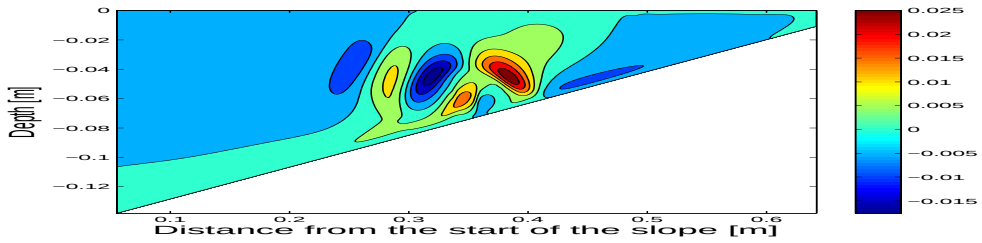


(c) The Neumann minus the Dirichlet condition for the full system

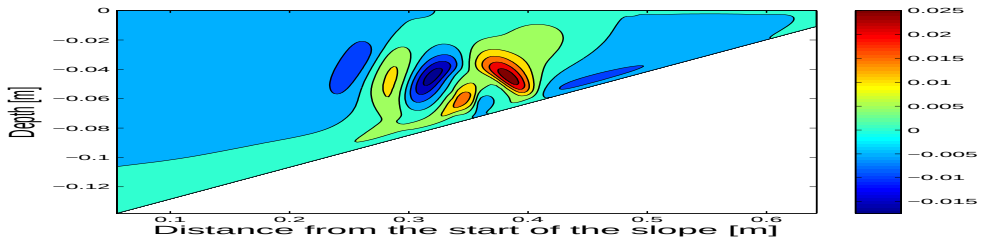


(d) The Neumann minus the Dirichlet condition for the simplified system

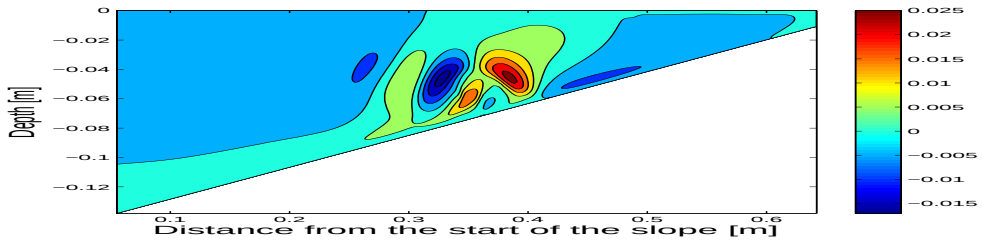
Figure 6.4: *Differences in the horizontal velocities after 35 seconds.*



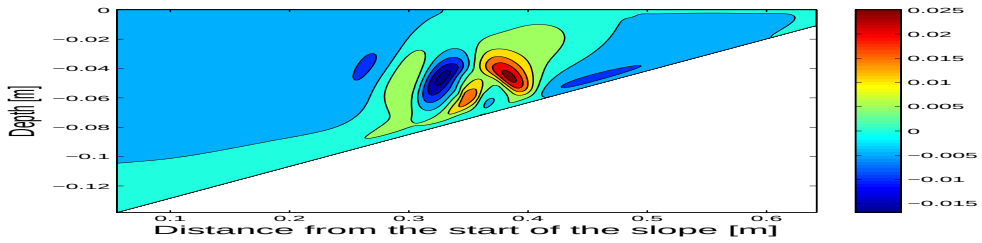
(a) The full system with a Neumann condition on the surface



(b) The full system with a Dirichlet condition on the surface

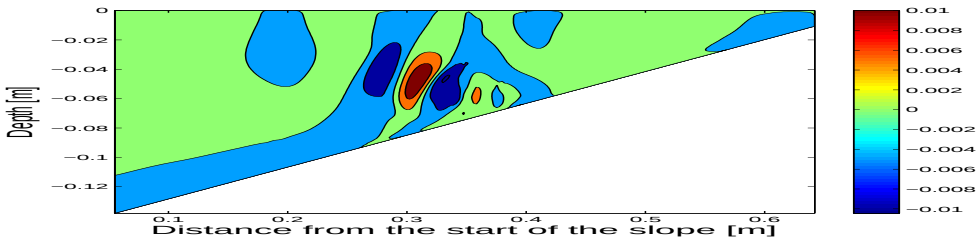


(c) The simplified system with a Neumann condition on the surface

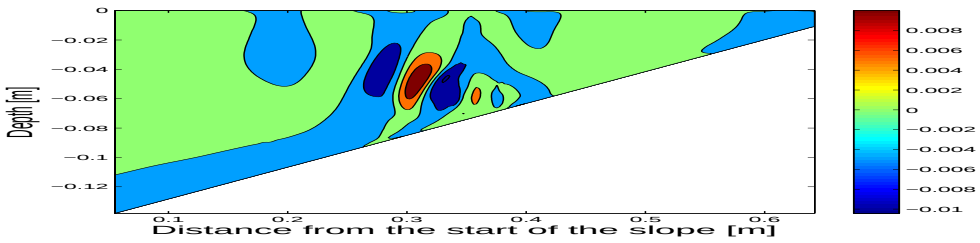


(d) The simplified system with a Dirichlet condition on the surface

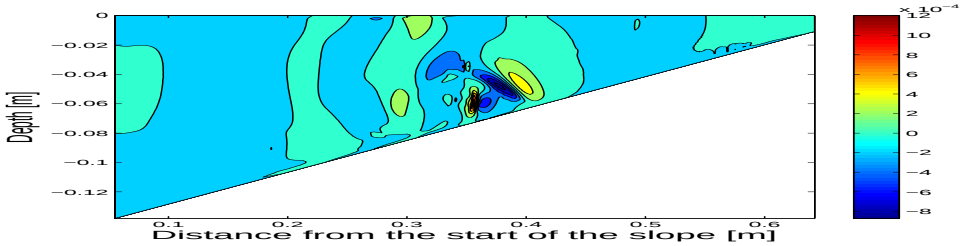
Figure 6.5: *The vertical velocities after 35 seconds.*



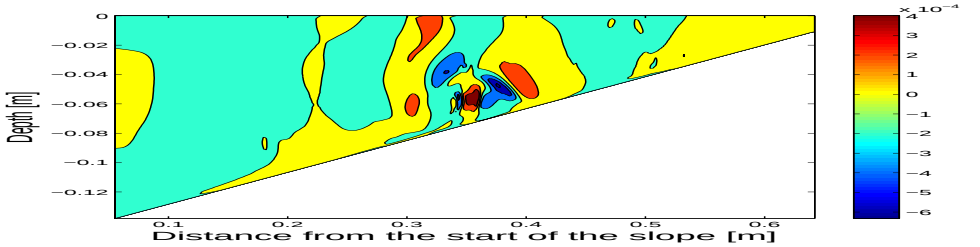
(a) The simplified minus the full set of equations with a Neumann condition



(b) The simplified minus the full set of equations with a Dirichlet condition



(c) The Neumann minus the Dirichlet condition for the full system



(d) The Neumann minus the Dirichlet condition for the simplified system

Figure 6.6: *Differences in the vertical velocities after 35 seconds.*

Set of equations	Boundary condition	Maximal velocity [m s^{-1}]	Maximal negative velocity [m s^{-1}]
Full	Neumann	0.0309	-0.0164
Full	Dirichlet	0.0308	-0.0161
Simplified	Neumann	0.0287	-0.0163
Simplified	Dirichlet	0.0286	-0.0160

Table 6.3: *The maximal vertical velocities after 35 seconds.*

for the simplified system, see the Figures 6.6(c) and 6.6(d), respectively.

Table 6.3 contains the maximal positive and negative vertical velocities for the four methods. The full set of equations gives the largest maximal velocities both in the positive and the negative direction. The maximal velocities are fairly robust with respect to the boundary condition used on the surface.

6.2.4 Discussion

The four methods give density profiles after 35 seconds which looks nearly the same. The separation point is located at the same place. However, the lateral extention of the bolus is larger for the full set of equations than for the simplified system. Thus positive velocities inside the bolus for the full system of equations are located at the same place as the negative velocities in the front and above the bolus for the simplified set. This gives large differences between the velocity fields. Hence, the small phase error in the density distributions gives large errors in the velocity fields.

Although the relative differences are large, about $\frac{1}{4}$ for the horizontal velocities, and $\frac{1}{3}$ for the vertical velocities, the velocity profiles looks much the same, they are just slightly translated. The speed of the separation point is about 0.02 m s^{-1} in all the simulations, while it is about 0.03 m s^{-1} in Figure 4 in Michallet and Ivey (1999). Thus the differences between the results given here are much smaller than the differences between the simulations and laboratory experiments.

It should be mentioned that the simulations are done in 2D. Solitary waves are well modeled as a 2D-phenomenon as long as they propagate without developing instabilities. When instabilities occur, 3D effects become important (Fringer and Street 2003). This can be one reason for the diversities between the simulation results and laboratory experiments.

It does make a difference which set of equations we use to model the non-hydrostatic pressure correction. However, it is not evident which system gives the best approximation to nature. Further investigations are needed to

decide which is the best one, but none of them is much better than the other one.

The results are robust with respect to a change of boundary conditions on the surface. The Neumann condition allows the bolus to propagate a bit faster than the Dirichlet condition. However, the difference is very small, and well within the differences compared to laboratory experiments.

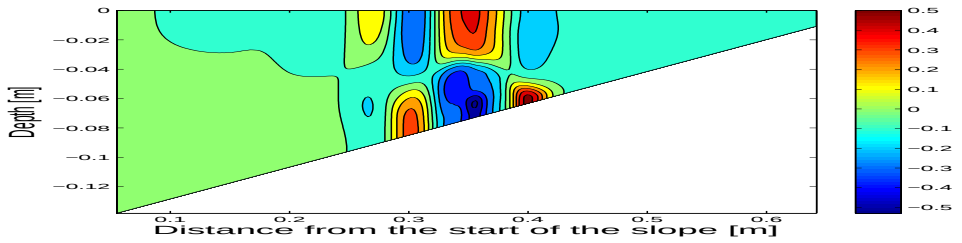
6.3 The Pressure Field

As we saw in Section 6.2, the density and velocity fields obtained by using the different methods are almost the same. However, if we consider the pressure fields, there are considerable differences. When a homogeneous Dirichlet condition is applied on the surface, the non-hydrostatic pressure there is zero. This is not the case with a Neumann condition, as can be seen in Figure 6.7. The pressure fields obtained by using the distinct boundary conditions are not even qualitatively the same. Nevertheless, the differences in the resulting density and velocity fields are not very large.

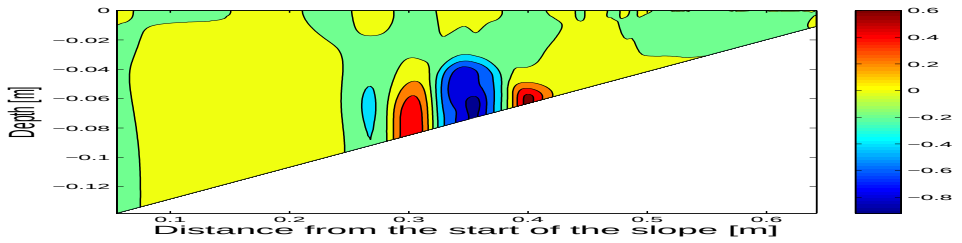
There are probably several reasons why the differences in the pressure fields do not have larger impact on the primary variables. First of all, we should remember that it is not the velocities and the pressure which are dependent on each other, but their time and spatial derivatives, respectively. The velocity field after 35 seconds is not the result of the instantaneous pressure field, but an accumulated result of the pressure field since the start of the simulation. Moreover, it is the spatial gradients in the pressure field that change the velocities.

The spatial difference along the slope located at the same place as the bolus are qualitatively the same for all the methods, although the values of the gradients are largest when we use the Dirichlet boundary condition. Also, the gradients are larger for the full set of equations than for the simplified system. We saw in Figure 6.3 that the horizontal velocities are higher when we use the full set of equations than if we use the simplified system. This must be caused by larger gradients in the velocity field for the full system. This leads us to belive that the pressure fields near the bottom as the bolus propagates up the incline look the same as we see in Figure 6.7 for the different models.

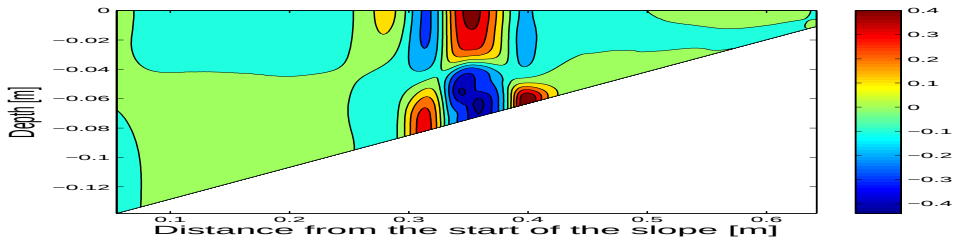
It is harder to explain why the differences in the pressure field above the bolus between the two boundary conditions do not have larger impact on the solutions. We should remember that the main circulation is caused by the hydrostatic pressure. The non-hydrostatic pressure only contributes with a small correction to the velocity fields. This is necessary to set up the bolus



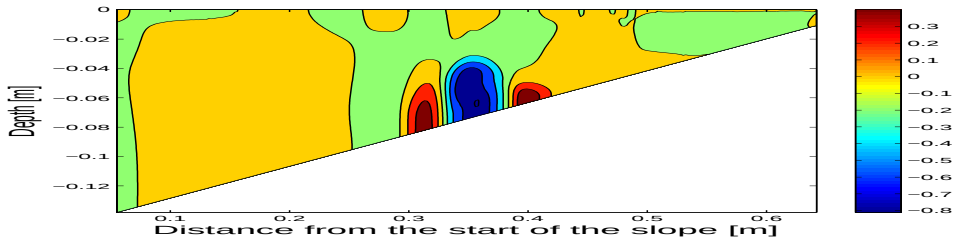
(a) The full system with a Neumann condition on the surface



(b) The full system with a Dirichlet condition on the surface



(c) The simplified system with a Neumann condition on the surface



(d) The simplified system with a Dirichlet condition on the surface

Figure 6.7: *Non-hydrostatic pressure fields after 35 seconds.*

when the wave hits the slope. Except from this, the motion is driven by the hydrostatic pressure. Thus, we must include non-hydrostatic effects to model nature accurately for this specific case, but the results are qualitatively the same with the four methods considered in this thesis.

6.4 Lower Viscosities and Diffusivities

The simulations described in the last sections were done with all viscosity and diffusivity coefficients equal to $2 \cdot 10^{-6}$ m^2 s^{-1} , except from $A_{M_{2D}}$, which was $5 \cdot 10^{-5}$ m^2 s^{-1}. If the coefficients are reduced to 10^{-6} m^2 s^{-1} , the simulations using the full set of equations become unstable. This happens after about 35.6 seconds for both types of boundary conditions on the surface. In order to check the effect of a reduction, the simulation with the simplified system and Neumann boundary condition is done again. Now, the viscosity and diffusivity coefficients are, except from $A_{M_{2D}}$, set to 10^{-6} m^2 s^{-1}. The value of $A_{M_{2D}}$ is unchanged. The results are compared to the corresponding simulation in Section 6.2.

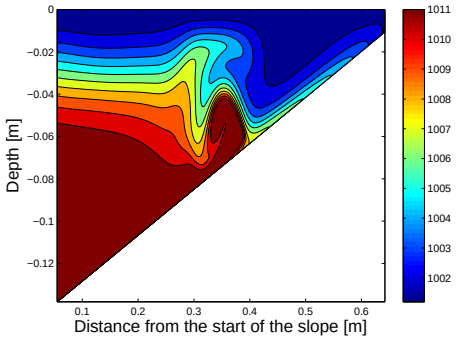
6.4.1 Results

Figure 6.8 shows the density distributions and the velocity fields for the simulations with high and low viscosities. With higher values of the coefficients, the fields we obtain are smoother. The mixing of water masses decreases with larger viscosities. Also, the maximal velocities are smaller, both in the horizontal and in the vertical direction. Compared to the laboratory results from Michallet and Ivey (1999), the high viscosity simulation gives too smooth fields, while the simulation with the lower viscosities is in better agreement with nature. The speed of the separation point is the same, as was reported by Berntsen et al. (2006).

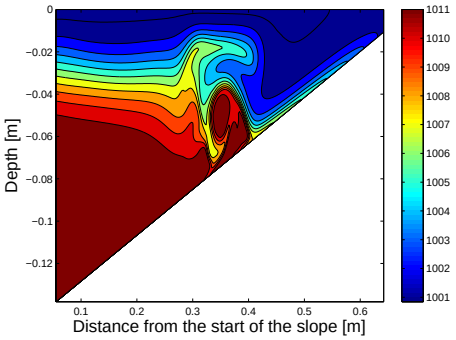
6.4.2 Discussion

With higher values of viscosities, the system looses more energy, causing smaller velocities. This gives less mixing. The breaking of internal waves is considered to be a major source of mixing in the ocean (Bourgault and Kelley 2003). To model this process accurately it is important to capture the dynamics in the phenomenon.

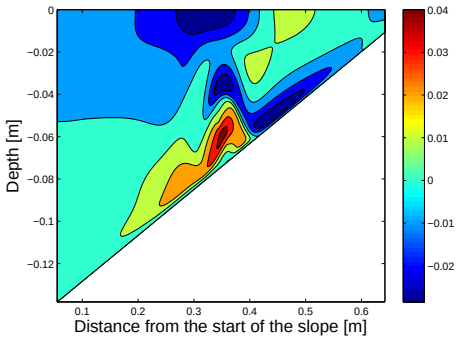
When both sets of equations are used with the same viscosity and diffusivity coefficients, the results obtained by using the full system are in best agreement with laboratory experiments. The simplified system seems to give



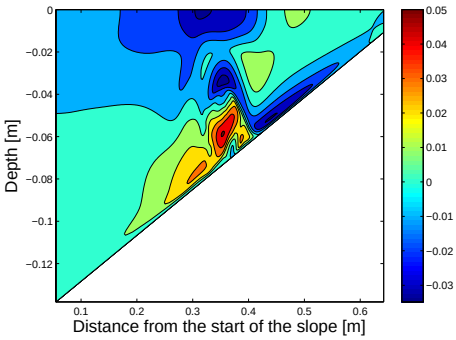
(a) The densities with high viscosities



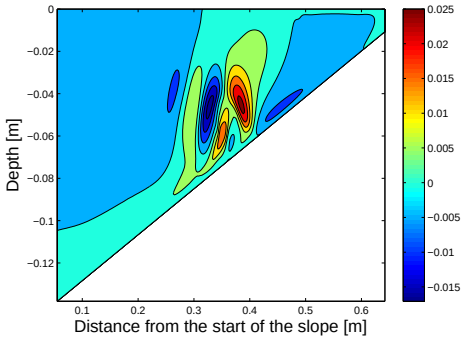
(b) The densities with low viscosities



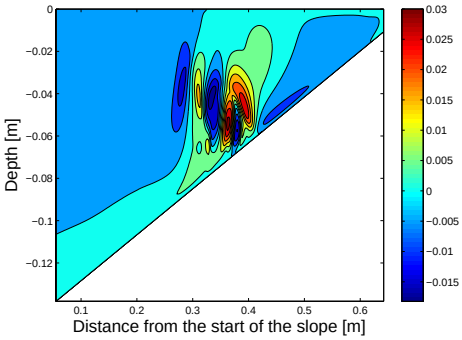
(c) The horizontal velocities with high viscosities



(d) The horizontal velocities with low viscosities

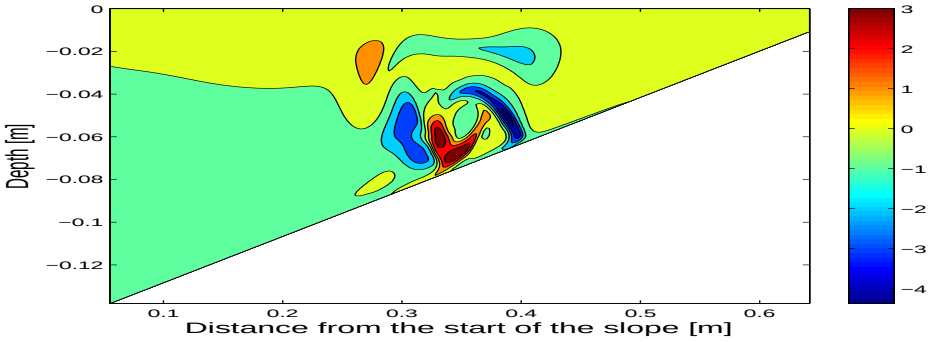


(e) The vertical velocities with high viscosities

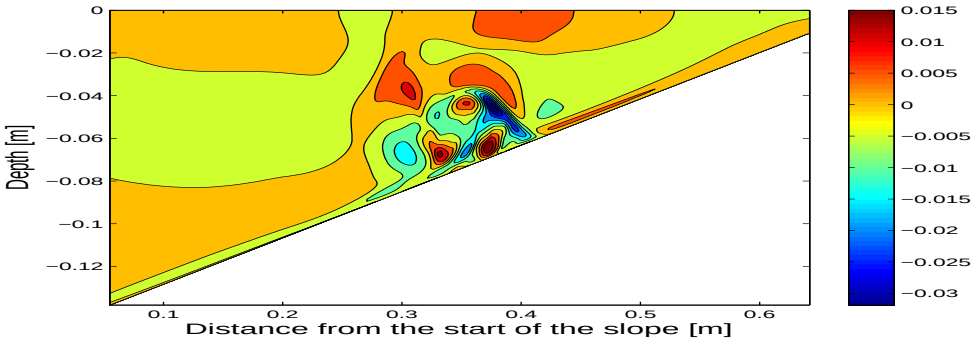


(f) The vertical velocities with low viscosities

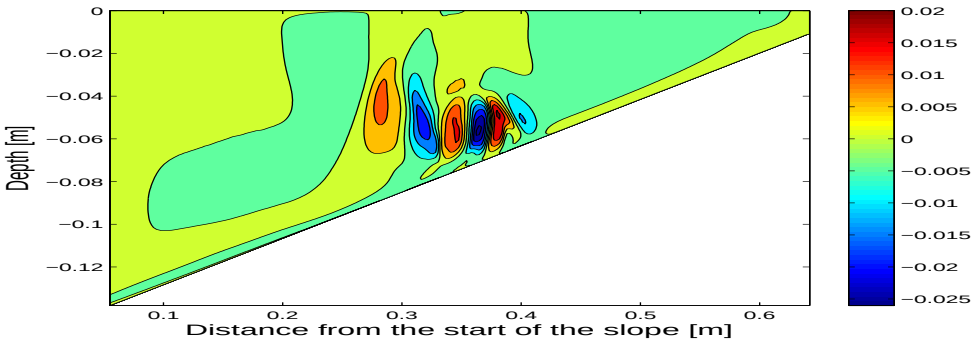
Figure 6.8: *The density and velocity distributions after 35 seconds with high and low viscosities.*



(a) Differences in the densities



(b) Differences in the horizontal velocities



(c) Differences in the vertical velocities

Figure 6.9: *Differences in the density and the velocity distributions between simulations with high and low viscosities after 35 seconds.*

more implicit viscosity and diffusion than the full set of equations, hence we can apply lower explicit values. Therefore, it is possible to get closer to what we find in nature with the simplified set of equations. We can, however, not be sure if this also holds if we use another numerical model.

6.5 Energetics

6.5.1 Results

The computed energy of the simulations done in Section 6.2 are shown in Figure 6.10. The energy is categorised as explained in Section 3.4.

The kinetic energy is zero at the start It increases when the wave starts to move, as can be seen in Figure 6.10(a). The kinetic energy decreases as the wave gets closer to the slope. The decrease is larger for the simplified set than for the full system of equations. Near the end of the simulation, the kinetic energy increases again, and the difference between the set of equations gets smaller.

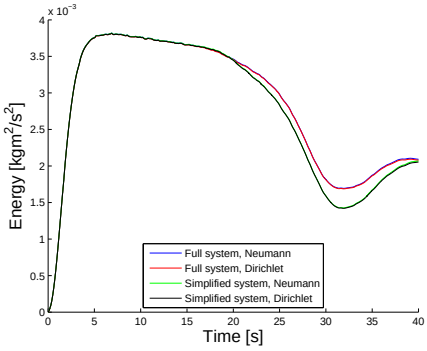
The available potential energy drops quickly as the wave is set in motion, as shown in Figure 6.10(b). After the first few seconds, the APE is almost constant until the wave approaches the slope, at which it increases. The APE is larger for the full system of equations than for the simplified system. The difference gets smaller at the end of the simulation, but it does not seem to go to zero.

In Figure 6.10(c) the background potential energy is shown. It increases linearly, and the BPE is equal for both sets of equations and both boundary conditions. In the end, the BPE for the full system of equations gets somewhat larger than for the simplified system, as we see in Figure 6.10(d). The difference is, however, only about one percent of the total increase in the BPE.

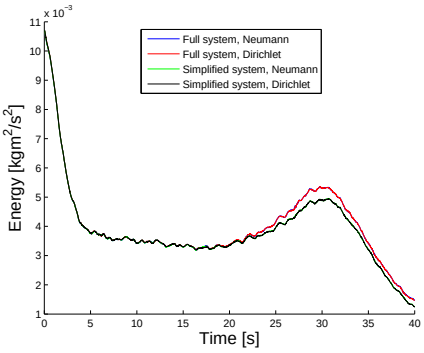
The energy due to surface elevation is shown in Figure 6.10(e). Only the results for the simplified set is shown, the full system of equations gave the same results. What is worth to notice here is the small scale on the y-axis. For instance, the energy due to the surface elevation is two orders of magnitude less than the kinetic energy. Also, the differences due to the boundary condition on the surface are small.

The potential energy can be seen in Figure 6.10(f). The energy is equal for all the methods until the wave hits the slope, then the full set of equations leads to somewhat more energy.

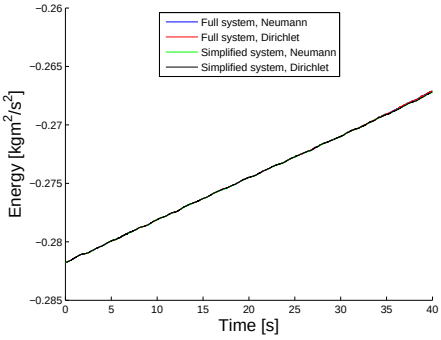
In Figure 6.11 the kinetic and the available potential energy for the simplified and full system with high viscosity are plotted together with the energy



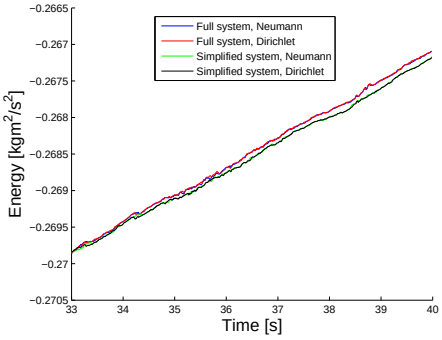
(a) The kinetic energy



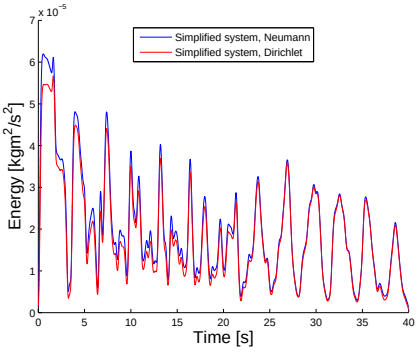
(b) The available potential energy



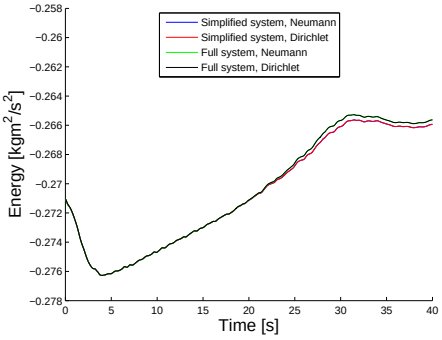
(c) The background potential energy



(d) The background potential energy in the last 7 seconds of the simulations

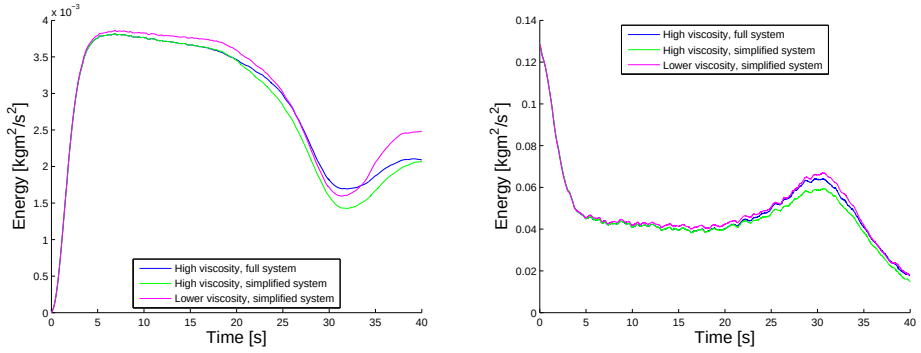


(e) The energy due to the surface elevation



(f) The potential energy

Figure 6.10: *The different kinds of energy in the simulations.*



(a) The kinetic energy

(b) The available potential energy

Figure 6.11: *The energy of the simplified and the full system with high viscosity, and for the simplified system with lower viscosity. The Neumann boundary condition is used for all the simulations.*

of the low-viscosity simulation described in Section 6.4. The APE with low viscosity is higher than for both sets of equations with high viscosity. Except from a small period when the wave reaches the slope, we obtain higher kinetic energy by using the simplified system with low viscosity than we get by using the full set of equations and higher viscosity.

6.5.2 Discussion

First, we notice that the choice of boundary condition makes no difference to the energy.

We saw in Section 6.2 that we get higher overall velocities by using the full set of equations. Hence, the kinetic energy obtained by the full system is higher than what is obtained by using the simplified set of equations. This leads to more stirring, and higher APE. Because of the stirring, the surfaces over which diffusion can take place get larger, hence the background potential energy increases more with the full set of equations than with the simplified set.

The constant growth in the background potential energy is due to diffusion which takes place independent of the motion.

The increase due to mixing contributes with only a small amount compared to this constant growth. However, the contribution from mixing is important for the energy budget in the ocean (Munk and Wunsch 1998). Although the difference in the change in BPE between the sets of equations is

small compared to the total increase, it can be important when, for instance, mixing efficiency is computed (see Peltier and Caufield (2003) for a definition of mixing efficency).

The low energy due to the surface elevation justifies the neglection of surface elevation done in Chapter 5.

The difference in kinetic energy for the simplified system with high and low viscosity is about 25 % after 35.6 seconds, when the low-viscosity simulation with the full set of equations gets unstable. If the effect of reducing the viscosity is the same for the full set of equations, the velocities for this simulations with low viscosities will be considerably higher. Since the full set gives higher maximal velocities than the simplified system, we need higher viscosities to avoid numerical instabilities.

6.6 Summary

In this chapter, we have compared different systems of equations and boundary conditions proposed to model the non-hydrostatic pressure. The shape of the bolus as it propagates up the slope is almost the same for all the methods. This is the case, even though the pressure fields changes dramatically when we change the boundary condition on the surface. However, using the full set of equations makes the lateral extention of the bolus somewhat larger. Because of large local changes in the velocity fields, this small phase difference gives large differences in the velocity fields. The full system also gives somewhat larger overall velocities.

The differences between the sets of equations are best seen when we consider the energetics. Although the energy behaves in qualitatively the same manner, the full set of equations gives solutions which have higher kinetic energy. This cause the simulation to become unstable if low viscosities are used. The instability of the full set of equations is a major drawback of this method. The higher viscosities needed make the system less capable of reproducing results from laboratory experiments, and probably also from nature.

From the results of the simulations in this chapter, it is hard to tell which set of equations is the best one. Further investigations and comparison to laboratory experiments and other numerical models are needed to resolve this issue. However, this also means that the consequences of using one system instead of another are small. Thus, if the results in this chapter also holds for other problems, we are, in some sense, free to choose which set of equations we will use to model the non-hydrostatic pressure. It seems that the most crucial differences can be found in the energy budget and the

numerical instabilities of the full set of equations.

From the experiments done here, we can not tell if it makes any difference which boundary condition we use on the surface.

Chapter 7

Comparison of the Iterative Methods

Our choice regarding set of equations and boundary condition to model the non-hydrostatic pressure was to be guided by the solutions obtained by the systems. In Chapter 6 we compared the density and the velocity fields for a test case. We will also investigate the numerical solvability of the resulting linear systems. In the previous chapter, the linear systems were solved by using SOR. Here, we will evaluate the results by doing the same simulations using the method of conjugate gradients and stabilised bi-conjugate gradients. We will also compare the number of iterations for the iterative solvers for the different methods to model non-hydrostatic pressure. Additionally, we investigate the impact on the solution of the tolerance for the error in the iterative methods.

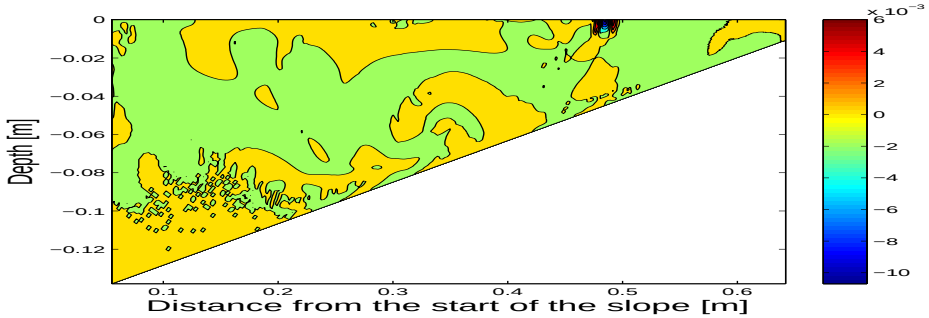
7.1 Solving the Linear System with CG and Bi-CGSTAB

The method of successive over-relaxations works well as long as the spectral radius of the matrix $\mathbf{G}$ defined in Equation (3.4) is less than 1. However, to check if this is fulfilled we must compute the inverses of matrices, which takes of the order n^3 computations. Also, finding the eigenvalues of $\mathbf{G}$ will be computational expensive, since the matrix in general not will be sparse. Thus, it is in practise not possible to find the spectral radius of $\mathbf{G}$ for the simulations done in Chapter 6. The results we obtained are in fairly good agreement with laboratory experiments. This indicates that SOR is well suited for the problem.

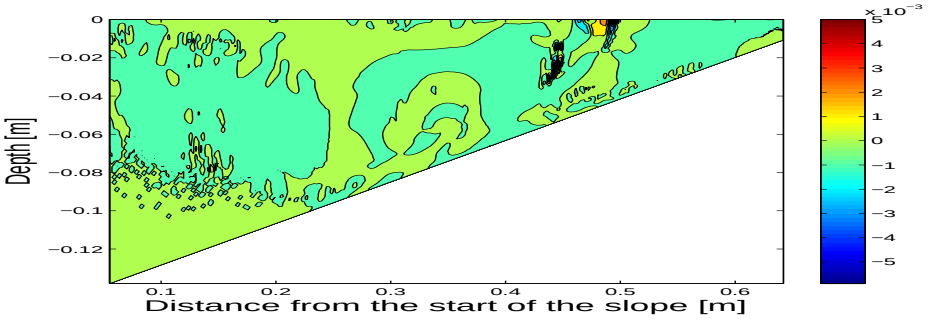
However, there might be an eigenvalue of $\mathbf{G}$ which is larger than 1. If the

contribution of the corresponding eigenvector in an eigenvector expansion of the right hand side is small, the method will appear to be convergent. However, if the number of iterations gets high enough, the eigenvector will start to dominate the solution, showing that the method really is divergent. If this is what happens, the solution we obtain by using a limited number of iterations or a weak convergence criterion cannot be trusted. This can explain the numerical instabilities when we use the full set of equations and low values for the viscosity and diffusivity coefficients.

Because of this, we do the simulation with the full set of equations and a Dirichlet condition on the surface again. This time we use the method of conjugate gradients to solve the linear system. We use the full set of equations because this is where the the convergence problems occurred. The coefficients for the viscosities and diffusivities have the same values as in



(a) The method of conjugate gradients



(b) The method of stabilised bi-conjugate gradients

Figure 7.1: *The difference in the density distributions after 35 seconds between the solutions obtained by SOR and CG and Bi-CGSTAB, respectively.*

Section 6.2. The problem is the same as in the previous chapter, hence the solution should also be the same.

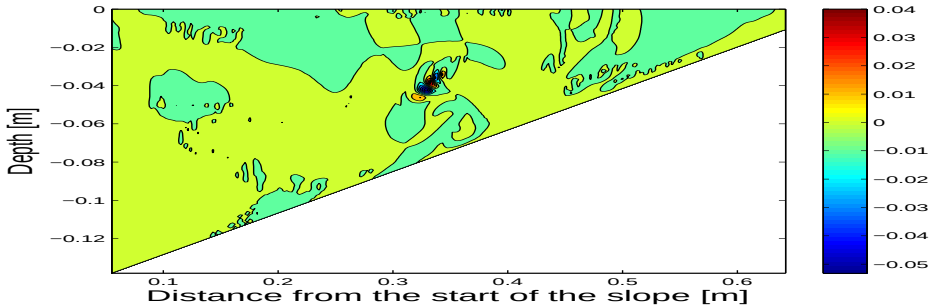
In Figure 7.1(a), the density distribution obtained by CG minus the distribution obtained by SOR is shown. The largest difference is of the order 10^{-2} kg m^{-3}. For the simplified system, the differences due to a change in the solver of the linear system are of the same order of magnitude for both boundary conditions. If we try to do a low viscosity simulation with the full set of equations and solve the linear system with CG, the simulation become unstable, just as it did in Section 6.4. Thus, the results obtained in Chapter 6 are robust to a change of linear solver.

We also do the same experiment with Bi-CGSTAB in order to check the implementation of this method. The result is shown in Figure 7.1(b). The small differences gives us trust in the implementation of Bi-CGSTAB.

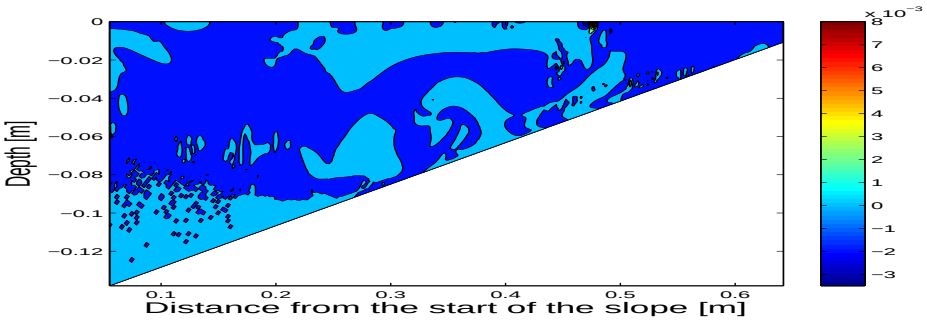
7.2 The Dependency Upon the Convergence Criterion

The simulations in Chapter 6 were done with the tolerance in Equation (3.8) set to $5 \cdot 10^{-6}$. This is a fairly strong convergence criterion. The reason for not allowing larger errors was to make sure that the differences we obtained were due to the equations, and not to the linear solver.

By only allowing small errors, much of the computation time is used by the linear solver. If we can allow larger residuals without making large changes in the solution, the simulations can be done much faster. When the convergence criterion is relaxed, the error in the computed pressure field increases. However, we are interested in the gradients of the pressure field. The initial error will be largest where the pressure gradients are largest, since this is where the field has changed most since the last sequence of iterations. If the accuracy of the solution is too low when the iteration process is stopped, we can get false pressure gradients where the non-hydrostatic pressure is most important. Also, the solution of an elliptic equation is smooth (Evans 1998), hence the true pressure field will not be noisy. The sequence of solutions obtained by the iterative method will also be smoothed as the residual gets smaller. If the iteration process is stopped too early, the false pressure gradients can create a noisy velocity field for the velocity corrections due to the non-hydrostatic pressure. This will be the opposite property of what the physical correct solution has. To conclude, we would prefer not to use more iterations than necessary, but the accuracy of the solution must be high enough for the solution to be trustable.



(a) The full system of equations



(b) The simplified system of equations

Figure 7.2: *The difference in the density fields after 35 seconds with SOR with the convergence criterion set equal to* $5 \cdot 10^{-6}$ *and* 10^{-3}*. The Neumann condition is used for both methods.*

In Figure 7.2 the differences in the density fields after 35 seconds using two distinct convergence criterions are shown for the full and the simplified system. The Neumann condition is applied on the surface for both sets of equations. For the simplified system, the maximal difference in the solution above the slope is $8 \cdot 10^{-3}\,\mathrm{kg\ m^{-3}}$. The maximal difference is located near the surface. For the full set of equations, the maximal error is $4 \cdot 10^{-2}\,\mathrm{kg\ m^{-3}}$, and it is located inside the bolus. Thus a relaxation of the convergence criterion has larger effect on the bolus for the full set of equations than for the simplified one.

The iterations are stopped when the norm of the residual is sufficiently small. When two iterative methods are applied with the same convergence criterion, the sum of the final errors over the entire domain will be of the

same size. Thus, there are probably larger errors in other parts of the tank in Figure 7.2(b) than in Figure 7.2(a). However, we want to model the solitary wave accurately. It is better for us that the errors are located away from the wave than near it. Therefore, we prefer the situation in Figure 7.2(b) to what we see in Figure 7.2(a). Also in Figure 7.1, there are probably larger errors away from the slope.

If we apply a Dirichlet condition on the surface, the differences due to the tolerated error are of the same order as for the simplified system with a Neumann condition. This is valid for both sets of equations.

7.3 Solvability by Different Iterative Methods

As shown in Section 7.1 we get approximately the same solution with SOR, CG and Bi-CGSTAB. We here present the convergence of all the three methods.

7.3.1 Description of the Experiments

As explained in Section 1.5.1, the linear system is solved twice for each time step. As an initial guess we use the solution from the previous time step for the predictor step. The solution for the predictor step is used as initial guess for the corrector step. In order to start the iterations with CG and Bi-CGSTAB with a good initial guess, SOR is used for the first 10 time steps.

We do the simulations with two distinct convergence criterions. The error ϵ in Equation (3.8) is set to 10^{-3} and $5 \cdot 10^{-6}$, respectively.

To avoid infinite loops, we only allow a limited number of iterations. the wave is set into motion, non-hydrostatic effects are important.s Additionally, the initial guess is not good. Therefore, we allow up to 1000 iterations for SOR and 200 for CG and Bi-CGSTAB. If the maximal number of iterations is reached without convergence of the solution, the hydrostatic approximation is used for this time step. Thus if convergence problems often occurs, we will in practice use a hydrostatic model. This is unacceptable for us. If we are to obtain the correct solution, we must use a model which capture non-hydrostatic effects.

The parameter λ in SOR is set to 1.8816 in all the iterations. This value is based on previous experience with the linear solver from the experiments done in Berntsen et al. (2006). Thus the value of λ is probably a good one for the simplified system, since both the equations and the initial conditions are the same. However, the value is not optimised for the full set of equations.

It is possible that the performance of SOR for this system could have been somewhat better with a more carefully chosen λ. Also, the preconditioner used for CG, although efficient is not very advanced. It might be possible to do this better.

The simulations are done with 400 time steps per second. We only consider the average number of iterations per second.

7.3.2 Expectations

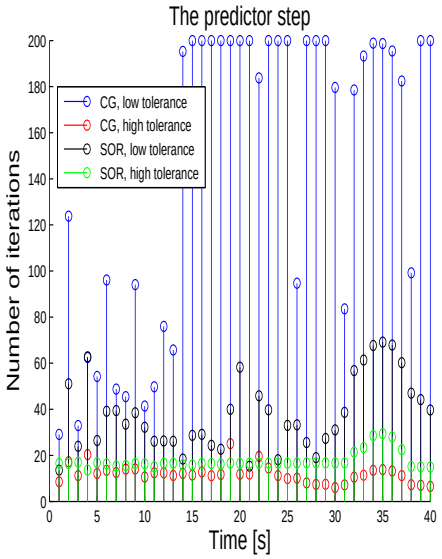
From the analysis of the matrices done in Chapter 5, we got some information of properties which might affect the linear solvers. Problems can arise for both the method of conjugate gradients and the method of stabilised bi-conjugate gradients when they are applied to a Neumann condition because of the zero eigenvalue. The conditioning of the simplified set of equations is better than that of the full set for a Neumann condition on the surface. For the Dirichlet boundary condition, the conditioning of the matrices is the same. Also, the condition numbers are much lower when we apply a Dirichlet condition on the surface than the modified condition number we get by using a Neumann condition. The gradients of the non-hydrostatic pressure in the vertical direction are smaller with a Dirichlet condition than with a Neumann condition, as can be seen in Figure 6.7. Even though the gradients are larger along the slope, the solution is more homogenous. Therefore, it should be easier to find by an iterative method. Finally, the matrices are not symmetric. This can cause problems for CG.

7.3.3 Results

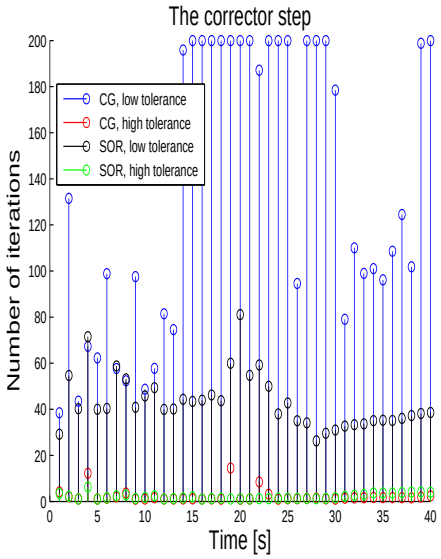
In Figure 7.3 the number of iterations for the Neumann condition is shown, while the average number of iterations for the Dirichlet condition is plotted in Figure 7.4. In order to make it easier to compare the results, the performance of the linear solver for SOR and CG are shown in the Figures 7.5 and 7.6, respectively. Figure 7.7 shows the average number of iterations for Bi-CGSTAB with a Dirichlet condition on the surface.

The Neumann Condition

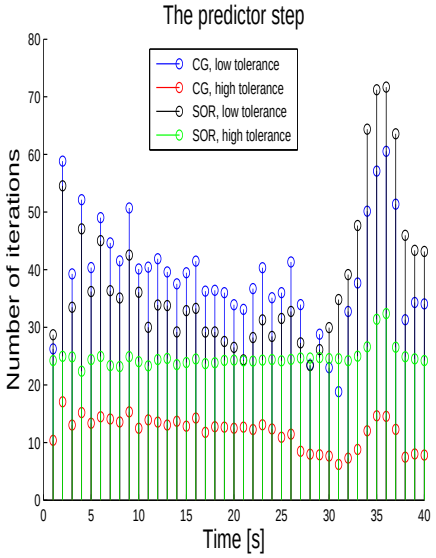
The method of conjugate gradients has problems when we use the full set of equations and apply a strong convergence criterion. The average number of iterations reaches the maximal allowed number in about half of the time intervals. During the first 30 seconds, both CG and SOR need about as many iterations in the predictor step as in the corrector step. In the remaining 10



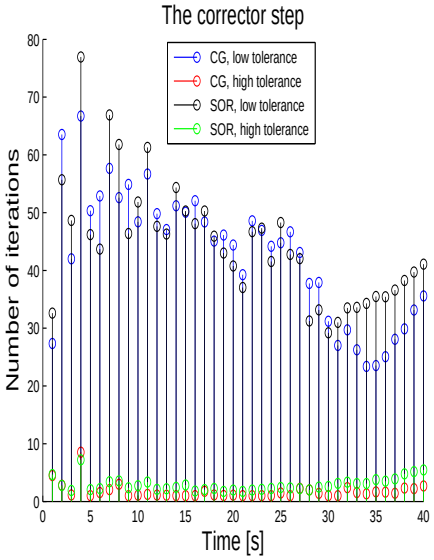
(a) The full set of equations



(b) The full set of equations

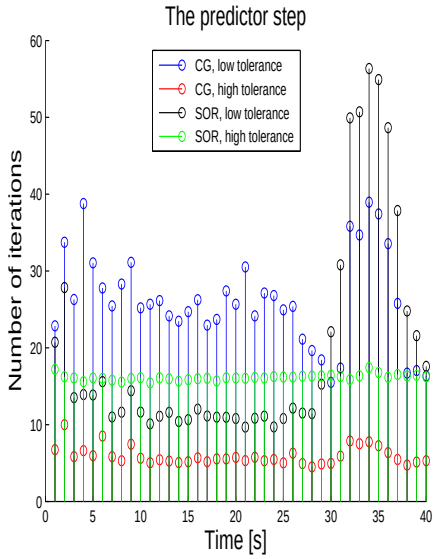


(c) The simplified set of equations

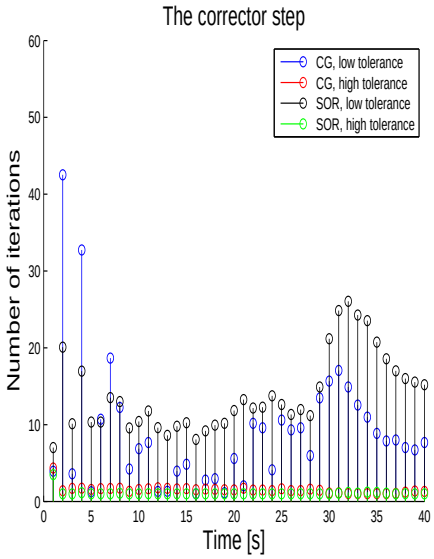


(d) The simplified set of equations

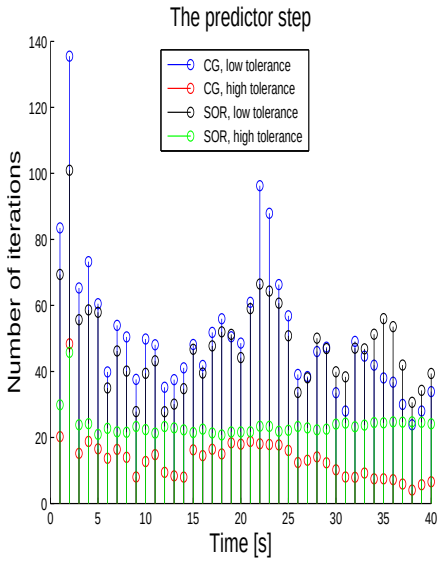
Figure 7.3: *Average number of iterations with a Neumann boundary condition.*



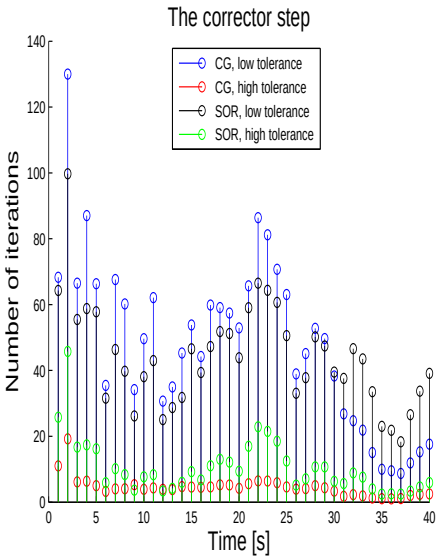
(a) The full set of equations



(b) The full set of equations

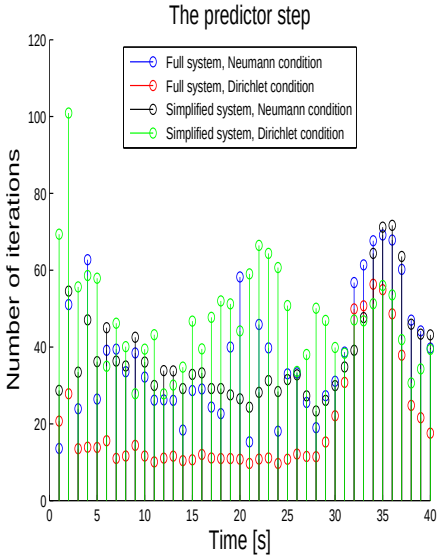


(c) The simplified set of equations

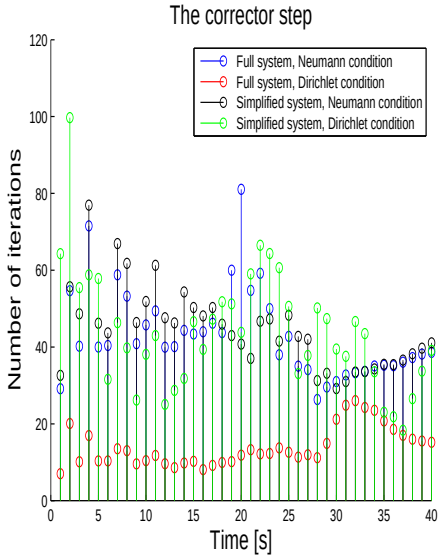


(d) The simplified set of equations

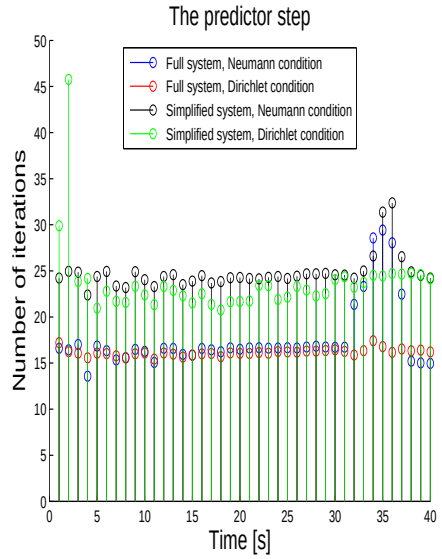
Figure 7.4: *Average number of iterations with a Dirichlet boundary condition.*



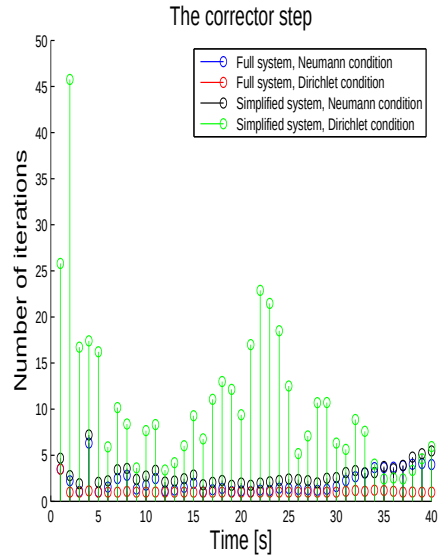
(a) Strong convergence criterion



(b) Strong convergence criterion

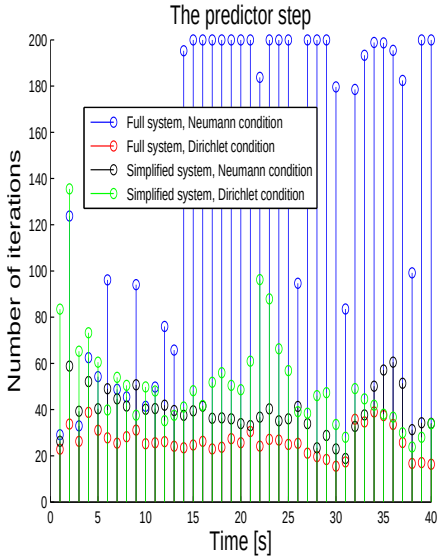


(c) Weak convergence criterion

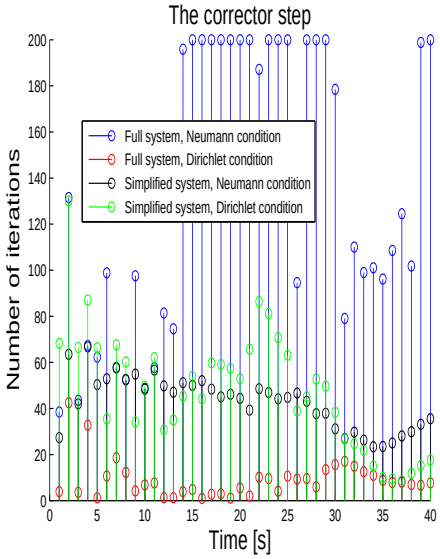


(d) Weak convergence criterion

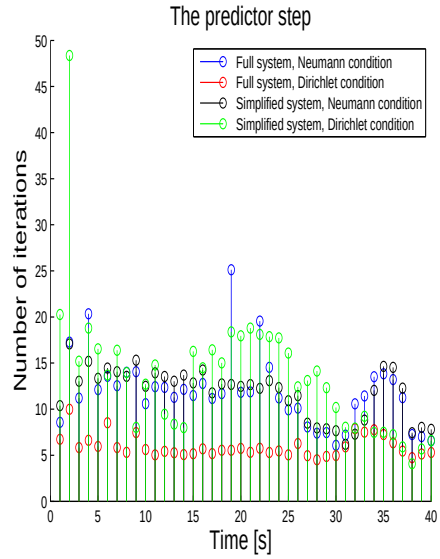
Figure 7.5: *Average number of iterations with SOR for the different sets of equations and boundary conditions. The maximal allowed number of iterations is 1000.*



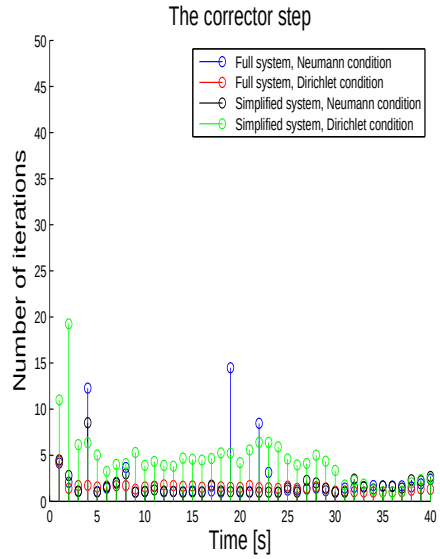
(a) Strong convergence criterion



(b) Strong convergence criterion

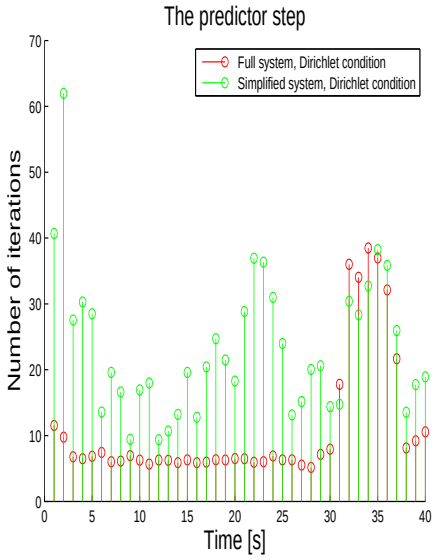


(c) Weak convergence criterion

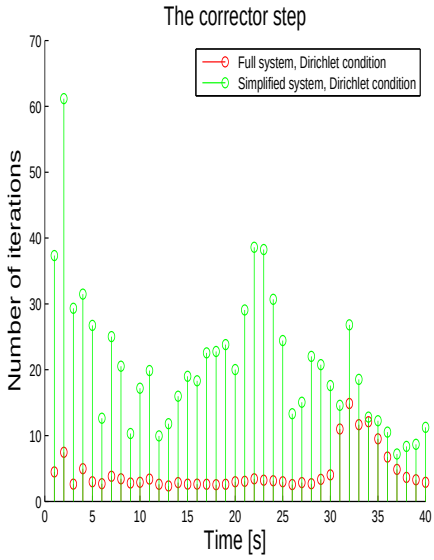


(d) Weak convergence criterion

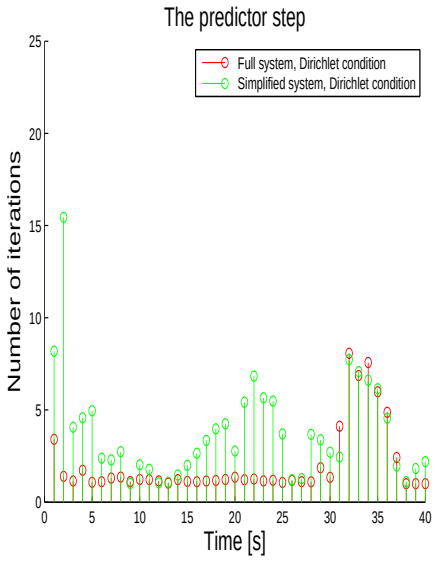
Figure 7.6: *Average number of iterations with CG for the different sets of equations and boundary conditions. The maximal allowed number of iterations is 200.*



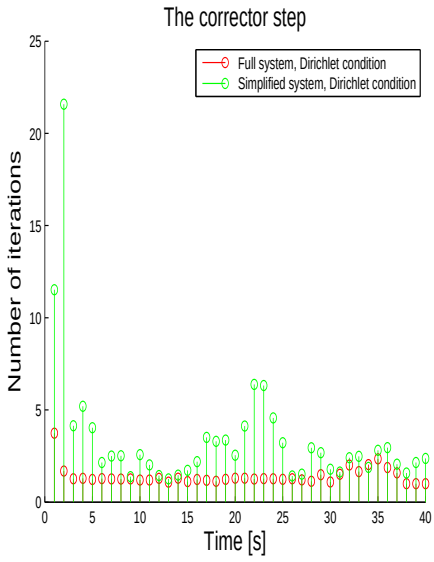
(a) Strong convergence criterion



(b) Strong convergence criterion



(c) Weak convergence criterion



(d) Weak convergence criterion

Figure 7.7: *Average number of iterations with Bi-CGSTAB for the different sets of equations and boundary conditions. The maximal allowed number of iterations is 200.*

seconds, more iterations are needed for the predictor step. If we relax the convergence criterion, the performance of CG is somewhat better than of SOR in the predictor step. As long as the we tolerate large errors, only a few iterations are needed in the corrector step.

For the simplified system and a low tolerated error, the number of iterations for CG is slightly higher than for SOR during the first 30 seconds. In the remaining 10 seconds, CG needs less iterations. This can be observed both for the predictor and the corrector step. If the convergence criterion is relaxed, the method of conjugate gradients uses only half as many iterations as SOR in the predictor step. For the corrector step, the methods are nearly equal in performance.

When Bi-CGSTAB is applied to a system with a Neumann condition on the surface, the method has severe convergence problems. The number of iterations reaches the maximal allowed value so often that it is clear that the method does not work for this problem. The problems occurs during the entire simulation, also for a weak convergence criterion. Therefore, the results are not shown here.

The Dirichlet Condition

If we use a Dirichlet condition, CG works well for the full set of equations. This holds also when we use a strong convergence criterion. However, SOR is better than CG except from the last 10 seconds of the simulation. Remarkably, the number of iterations in the prediction step of SOR is higher if we allow a high value of the error than for a low tolerance in the first 30 seconds of the simulation. This is not the case in the last 10 seconds, as the number of iterations needed increases for the simulations with strong convergence criterion.

For the simplified system, we get much of the same behaviour as with a Neumann condition. When large errors are allowed, CG is still better than SOR in the predictor step. The difference is less than with a Neumann condition, though. However, SOR needs more iterations in the corrector step. With a strong convergence criterion, the performance of the two methods are nearly equal in the first 30 seconds. In the last 10 seconds, SOR needs more iterations than CG, but again the difference is not as significant as for the Neumann condition.

The performance of Bi-CGSTAB is good, also with a Dirichlet condition on the surface. The performance of Bi-CGSTAB with the full set of equations is very good. It needs less than 10 iterations with a strong convergence criterion in the predictor step for the first 30 seconds. If larger errors are tolerated, less than 5 iterations are needed. As for the corrector step, both

the convergence criterions are met in less than 5 iterations. The average number of iterations for the simplified system is also considerably lower than for both SOR and CG. This is valid for both convergence criterions.

7.3.4 Discussion

The Convergence Criterion

The number of iterations needed to find the non-hydrostatic pressure field is fairly high, also when the movement is to a good approximation hydrostatic. The residual is computed according to Equation (3.7). The right hand side of the linear system is the divergence of the provisional velocity field. In the first 30 seconds of the simulation, these values are low. This leads to a strong convergence criterion, and therefore many iterations. Thus, with the convergence criterion used for the iterative methods, we spend much computation time to find a pressure field which does not contribute much to the result. As the wave propagates up the slope in the last 10 seconds of the simulation, the changes in the non-hydrostatic pressure from one time step to the next are larger. That is, the nominator in Equation (3.8) is larger for the initial guess. However, also the denominator will increase because of higher values of the divergence of the velocity field. This put together we do not need much more iterations in the last period of the simulation, even though non-hydrostatic effects are more important than in the beginning.

It would probably have been better to have a convergence criterion based on the velocity field itself, instead of it's divergence. However, it is not clear how this can be done in a way that gives a strong convergence criterion when non-hydrostatic effects are important.

Convergence Problems

Both CG and Bi-CGSTAB have problems with the convergence when they are applied to the full set of equations with a Neumann boundary condition. Since also Bi-CGSTAB has problems, the reason cannot only be the asymmetry of the matrices. The reason is probably the zero eigenvalue. We notice that the problems are worst when non-hydrostatic effects are not important for CG. This means that the denominator in Equation (3.8) is small. Therefore, the residual in the nominator must also be small to fulfil the convergence criterion. Because one eigenvalue is zero, the range of the matrix is not the entire $\boldsymbol{R}^n$. Thus, unless the right hand side is entirely in the range of the matrix, the solution obtained by the iterative method will never equal the true solution. Hence the residual can never be equal to zero, regardless

of how many iterations is used. Some iterative methods will be able to get the residual as small as possible. For other methods, the part of the solution which is not in the range of the matrix is necessary for the iteration process to converge. This can explain why it is possible to fulfil the convergence criterion by SOR, but not by CG or Bi-CGSTAB.

Apparently, the zero eigenvalue does not affect the convergence of CG for the simplified system as much as for the full set of equations. Indeed, in the first part of the simulation, the number of iterations is lower with a Neumann condition than when a Dirichlet condition is applied. The structure of the matrix is different from the full set of equations, in that it has 5 nonzero diagonals instead of 9. This can give different convergence properties for the linear solver. When non-hydrostatic effects are important in the last part of the simulations, the convergence is slower for the Neumann condition, as expected. However, the iterations does converge. Thus the possible problems with CG due to the Neumann boundary condition foreseen in Chapter 5 do not appear, even though it was not obvious that the method would work.

When the iterations do not converge, the simulations are done hydrostatic. For CG and Bi-CGSTAB this happens so often that it is doubtfull that the solution obtained can be trusted. When the convergence criterion is relaxed, CG works well, also for the full set of equations and a Neumann boundary condition. However, as we saw in Figure 7.2(a), the relaxation of the convergence criterion changes the density field near the bolus much more for this set of equations and boundary conditions than for the other combinations. Hence, it seems that CG should not be applied to the full system of equations with a Neumann condition on the surface.

Bi-CGSTAB is severely affected by the zero eigenvalue also for the simplified system. Because of convergence problems, the method should not be applied for any of the systems as long as we use a Neumann condition on the surface.

Condition Numbers

The expected slower convergence for the Neumann condition occurs for the full set of equation for both CG and SOR. For the simplified set of equations, however, the number of iterations is less for the Dirichlet condition only when the wave propagates up the slope. That is, the expected behaviour appears only when non-hydrostatic effects are important for these to methods. The behaviour of Bi-CGSTAB is as expected. It works with a Dirichlet boundary condition, but not with a Neumann condition.

Predictor and Corrector Steps

Because the initial guess for the corrector step comes from the present time step, we would expect the number of iterations to be less for the corrector step than for the prediction. This holds for all methods to compute the non-hydrostatic pressure if we tolerate large errors. If the convergence criterion is strong, the convergence is somewhat faster in the corrector step, especially as the wave propagates up the slope.

When we solve the full set of equations with a Dirichlet condition, the number of iterations for the prediction increases for the first 30 seconds if we relax the convergence criterion. This is contrary to what we would have expected. However, if we sum the iterations in the predictor and corrector step, we get a slightly larger value with a strong convergence criterion. Also, the initial guess both for the prediction and the correction is better because the residual of the solution in the last sequence of iterations was smaller.

Conclusions

All the four methods to model non-hydrostatic pressure gives a linear system which is solvable by using SOR. The Krylov subspace methods are in general considered to be better than SOR. Therefore, it is remarkable that the difference in performance between SOR and CG is not larger. We should, however, have the asymmetry of the matrices in mind. This probably slows down the convergence of CG, and might explain why CG does not outperforms SOR. Bi-CGSTAB does not have such problems, but it suffers from severe problems when a Neumann condition is used on the surface.

The number of iterations needed is lowest if we use the full set of equations and a Dirichlet boundary condition.

It should be mentioned that to compare the linear solvers by the number of iterations is somewhat naive. What really matters is the time needed to solve the linear system. This is dependent on how the program is written, which computer is used, and so on. Hence, it is rather complicated to give a general analysis of the real computational time for each method. The most expensive part of the algorithms is to compute a matrix-vector product. Both CG and SOR need one such multiplication for each iteration. In addition, the preconditioning in CG means to compute another matrix-vector product. Hence, the computational burden for SOR for each iteration is less than what it is for CG. Bi-CGSTAB has five matrix-vector multiplications for each iteration, and is therefore much more expensive than both CG and SOR.

7.4 Summary

The results of the simulations are independent of the choice of linear solver.

It is not straightforward to draw conclusions from the comparison of convergence of the linear solvers. The results are not in very good agreement with what we expected from the analysis done in Chapter 5. This shows that convergence of iterative methods is not only a matter of condition numbers and eigenvalues.

The full set of equations with the Dirichlet boundary condition needed least iterations for both SOR, CG, and Bi-CGSTAB. However, this set has 9 nonzero diagonals compared to the 5 nonzero diagonals of the simplified system. Hence, each iteration needs more computations, so the overall performance is not that much better. Nevertheless, the full set of equations with a Dirichlet boundary condition on the surface is the best choice from a purely computational point of view.

Chapter 8

Conclusions and Further Work

In this thesis, we have investigated four different ways to model the non-hydrostatic pressure in σ-coordinate ocean models. We have considered the methods both from a physical and a numerical point of view. In this final chapter, we will summarise the results, and draw conclusions.

8.1 Summary and Conclusions

8.1.1 The Set of Equations

There are some differences in the primary variables depending on which set of equations we use to model the non-hydrostatic pressure. The velocities are higher when we apply the full set of equations. However, much of the differences are due to a small difference in the lateral extention of the bolus. The speed of the separation point and the shape of the bolus are the same for both systems of equations. Thus the sets of equations leads to qualitatively the same velocity and density fields for a solitary wave which propagates up a slope.

For the energetics, there are larger differences. This can be important, because the energy budget is an indication of what is the result after the motion has died out. As we saw in Section 6.5, the distinct sets of equations gave different values of kinetic and potential energy when the wave approached the slope and propagated upwards. The difference in the kinetic energy tended to zero at the end of the simulation. Also for the available potential energy the difference got smaller. It did not, however, tend to zero. If we model a real ocean with many occurrences of non-hydrostatic phenomena, the energy budget we obtain by the two methods might differ a lot. This can have consequences for the prognostic of the long time motion in the ocean.

The instability of the full set of equations when low viscosities and diffusivities are applied is a major problem with this method. The low viscosity simulation with the simplified system done in Section 6.4 is in better agreement with laboratory experiments than the results obtained by the full set of equations in Section 6.2. If we in general need higher viscosities for the full system of equations to avoid numerical instabilities, this makes the simplified set more attractive. However, it the simulations might be stable if the spatial resolution is increased.

It should also be mentioned that the finite difference approximation for the simplified system (Appendix B.1) is easier to implement correctly than the approximation for the full system (given in Appendix B.2). Additionally, the linear solver is more complicated for the full set of equations.

All this together, we have not found any reasons to say that one set of equations is the right choice. It should be possible to use both systems to model the non-hydrostatic pressure. The numerical instability of the full set of equations is the largest drawback of the two methods. Hence, if we are to recommend one of the methods, the simplified system seems to be the best one. However, we have not done enough experiments to draw any strong conclusions.

8.1.2 The Boundary Condition

We saw in Chapter 6 that the boundary condition on the surface hardly has any impact on the solution of the problem. This was valid both for the density and velocity fields, and for the energetics. Thus it seems that the choice of boundary condition on the surface is not important to the solution.

However, there might be a weakness with the model problem we used in Chapter 6. Internal waves are baroclinic, their impact on the surface elevation is small. Thus the surface elevation in Chapter 6 is close to be constantly equal to zero. For a barotropic process, the changes in the surface elevation will be much larger. It is reasonable to expect that the effects of the boundary condition at the surface will be larger for processes where the surface is more important.

One underlying assumption for our numerical scheme is that the pressure can be splitted as in Equation (1.10). That is, we must apply a homogenous Dirichlet condition for the non-hydrostatic pressure on the surface.

On the other hand, in nature the non-hydrostatic pressure can be nonzero also at the surface. One example is when to surface currents meet. For the equation for the non-hydrostatic pressure to be well posed, boundary values must be specified. Since we do not know the pressure at the surface, it does not seem natural to use a Dirichlet condition there. A condition of the

Neumann type seems better suited. This also has the effect of filtering high frequency waves at the surface (Marshall et al. 1997b).

However, a nonzero non-hydrostatic pressure at the surface contradicts the splitting of the pressure described by Equation (1.10). If we do not split the pressure, we also have problems with doing mode splitting. We are then left with a model like the one of Lin and Li (2002), where the time step is limited by the fast travelling surface waves. One possible way to combine Equation (1.10) with mode splitting is to claim the equation to be valid only in the interior of the ocean.

Many of todays ocean models are hydrostatic. These can be extended to include non-hydrostatic effects by adding velocity corrections at the end of each time step. See for instance Kanarska and Maderich (2003) or Heggelund et al. (2004) for two proposals of how this can be done. If the Neumann condition is applied on the surface, the restriction derived in Section 2.3.1 should be kept in mind. This means we should take the approach of Heggelund et al. (2004) instead of Kanarska and Maderich (2003). If we instead use a Dirichlet condition, there are no such restrictions.

8.1.3 Numerical Considerations

When we choose which set of equations and which boundary condition to use in order to find the non-hydrostatic pressure field, we should also keep the computational cost of the methods in mind. It may sound strange to make a choice of a physical model based on the solvability of the resulting mathematical problem. Also, this contradicts the priority of the two criterions we established in the end of Chapter 2. To justify this, we must take a broader view than to just model the non-hydrostatic pressure. The main goal of ocean modelling should be to give an as accurate description of nature as possible. The non-hydrostatic part of the simulations in Chapter 6 takes about $\frac{2}{3}$s of the total computation time when we use SOR, which is the only method that works for all the linear systems. If we can reduce the number of iterations in the linear solver by solving a simpler problem, and spend some of this time on other parts of the numerical model, this may give a better overall solution. Of course this must be done carefully, to get a physical correct solution is the most important issue.

For instance, in Chapter 6, we saw the boundary condition on the surface hardly makes any difference to the density or velocity fields. This holds for both sets of equations. To use a Dirichlet condition on the surface gives some advantages when we solve the linear system to find the non-hydrostatic pressure. The equations are then solvable with Bi-CGSTAB, and just a few iterations are needed. Also SOR and CG seem to converge faster with a

Dirichlet condition as long as non-hydrostatic effects are important. From this point of view, the Dirichlet boundary condition is preferable. However, we should remember that we have only used the linear solvers for this one test case. Their performance can be different for other problems.

A solver for the linear system of equations based on geometric multigrid (GMG) and algebraic multigrid (AMG) methods is being developed now by Helge Avlesen (Avlesen 2006). The results are promissing, with only a few iterations needed to meet strong convergence criterions. Even though the computational cost for each iteration is large, the time it takes to solve the linear system is in some cases better better than SOR by a factor 5. In the worst cases, the SOR is better than the advanced GMG and AMG methods. However, the performance of SOR can be expected to be worse for larger systems of equations. With the increasing computer power available, the multigrid methods will probably be more important in the future.

8.2 Evaluation of the Results

The comparison of the results obtained by the four methods in Chapter 6 gives us no information of how well the methods model the non-hydrostatic pressure. What we do get to know is whether the choice of model makes a difference to the solution. In order to evaluate how well the models approximate the non-hydrostatic pressure, another approach should be used.

In Chapter 6, we compared the model results with laboratory experiments done by Michallet and Ivey (1999), and showed that the model results differed a lot from nature. However, this might be a too optimistic comparison. The numerical model is based on the equations given in Chapter 1. These are just a mathematical model of what happens in nature. The best result we can possibly get from them is an exact solution of the equations, not an exact modelling of the real-life phenomenon. In addition, the numerical solution will be affected by errors from discretizations, operator splitting and so on. Thus, the large errors when numerical results are compared to laboratory experiments should not come as a surprise to us. Because the differences due to the choice of model for the non-hydrostatic pressure are much smaller than the differences between model results and laboratory experiments, it seems more important to make efforts to improve other parts of the numerical model. However, we should keep in mind that the simulations are done in 2 dimensions. The results will probably be in better agreement with nature in a 3D-simulation. This can make the relative differences between the models larger when compared to nature.

A better way to evaluate the results in this thesis might be to do the

same simulations as in Chapter 6 with another numerical model based on the same basic equations. Then, the answers should ideally be the same, at least if the grid size and the time steps are sufficiently small. However, distinct models use different ways of splitting of the operators, the equations are solved by different numerical methods and so on. Therefore, we might find that the differences between the results from distinct models are larger than the differences due to the equations and the boundary conditions for the non-hydrostatic pressure. Despite of these possible problems, a comparison with a distinct numerical model should be done.

8.3 Further Work

All the simulations done in the thesis have been of a solitary wave propagating up an incline. Although this is an important process in the ocean, for instance for the mixing of water masses (Bourgault and Kelley 2003), there are other important non-hydrostatic phenomena too. Some examples of this were mentioned in Section 1.5.3. It is not obvious that the results in this thesis will be valid also for other processes in the ocean. This should be investigated further.

Internal waves are baroclinic phenomena, they have small impact on the surface elevation. However, there are other processes which are barotropic and non-hydrostatic. It is possible that the boundary condition on the surface is more important for a barotropic phenomenon. Additionally, the full set of equations give a linear system with coefficients which depends on the surface elevation. This may lead to a more asymmetric coefficient matrix, and thus problems for some of our linear solvers.

All the derivations and simulations are done in two dimensions. From a mathematical point of view it should not be difficult to extend the results to three dimensions. The transformation to σ-coordinates for the third spatial variable, y, is equal to the transformation for x, given in Equation (1.8). The resulting linear systems will have 7 and 19 nonzero diagonals for the simplified and the full set of equations, respectively. From a physical point of view, 3D-effects are important for the breaking of internal waves (Fringer and Street 2003). It is possible that the introduction of a third dimension allows for larger variability between the solutions from the four methods to compute non-hydrostatic pressure.

All the simulations in this thesis are done with a fixed spatial resolution. It could have been interesting to do simulations with an increasing number of grid points, as was done by Berntsen et al. (2006). In this way we can check if the results really converge as the spatial resolution increases. Also, we can

investigate if the instability of the full set of equations with low viscosity is affected by the resolution.

The results in this thesis are also dependent on the numerical methods used in BOM. The simulations should be checked by another model.

Appendix A

Nomenclator

Here we give the parameters used in the thesis. The first column gives the symbol, the second gives a description, and the unit is given in the third column. The parameters which are dimensionless have the unit 1.

Symbol	Description	Unit
$\mathbf{A}$	A matrix in a linear system	1
A	Dummy symbol used in the definition of transformations to σ-coordinates	1
A_H	Horizontal diffusivity	$\mathrm{m^2\,s^{-1}}$
A_M	Horizontal viscosity	$\mathrm{m^2\,s^{-1}}$
$A_{M_{2D}}$	Horizontal viscosity in the depth averaged calculation	$\mathrm{m^2\,s^{-1}}$
a_0	Parameter used for the initial density distribution	m
a_1	Parameter used for the initial density distribution	m
$a_{i,k}$	Element number (i,k) in the matrix $\mathbf{A}$	1
$\mathbf{B}$	Lower triangular part of a matrix	1
$\mathbf{b}$	A vector	1
$b_{i,k}$	Element of a right hand side vector	1
$\mathbf{C}$	Matrix in the derivation of the method of conjugate gradients	1
C_D	Bottom drag coefficient	1
$\mathbf{c}$	Vector in SOR	1
$\mathbf{D}$	The main diagonal of a matrix	1
$D = H + \eta$	Dynamic depth	m
D_i	Discrete representation of the dynamic depth in the middle of a cell	m
DF_x	Viscosity terms in the horizontal momentum equation in σ-coordinates	$\mathrm{m^2\,s^{-2}}$

Symbol	Description	Unit
DF_σ	Viscosity terms in the vertical momentum equation in σ-coordinates	$\mathrm{m^2\,s^{-2}}$
DZ	Thickness in σ-coordinates of cells	1
DZZ	Distance in σ-coordinates between cell centres	1
$\mathbf{E}$	Matrix consisting of left singular vectors	1
E_K	Kinetic energy	$\mathrm{kg\,m^2\,s^{-2}}$
E_P	Potential energy	$\mathrm{kg\,m^2\,s^{-2}}$
E_{P_0}	Reference potential energy	$\mathrm{kg\,m^2\,s^{-2}}$
$\mathbf{F}$	Matrix consisting of right singular vectors	1
g	Acceleration of gravity	$\mathrm{m\,s^{-2}}$
H	Static depth	m
h_1	Parameter in the initial condition	m
h_2	Parameter in the initial condition	m
$\mathbf{I}$	The identity matrix	1
i	Index of a matrix element in the horizontal direction	1
KB	Number of cell interfaces in the vertical direction	1
K_H	Vertical diffusivity	$\mathrm{m^2\,s^{-1}}$
K_M	Vertical viscosity	$\mathrm{m^2\,s^{-1}}$
k	Index of a matrix element in the vertical direction	1
L	Length of the tank	m
$\mathbf{M}$	Preconditioning matrix	1
$\mathbf{n}$	Outer normal vector	1
n	Size of a quadratic matrix	1
P	Pressure	$\mathrm{kg\,m^{-1}\,s^{-2}}$
P_{Atm}	Atmospheric pressure	$\mathrm{kg\,m^{-1}\,s^{-2}}$
$P_{Hydrostatic}$	Hydrostatic pressure	$\mathrm{kg\,m^{-1}\,s^{-2}}$
P_{Int}	Internal pressure	$\mathrm{kg\,m^{-1}\,s^{-2}}$
P_η	Pressure due to surface elevation	$\mathrm{kg\,m^{-1}\,s^{-2}}$
$\mathbf{p}$	Search direction for CG	1
Q	Non-hydrostatic pressure	$\mathrm{kg\,m^{-1}\,s^{-2}}$
$\boldsymbol{R}^{n\times n}$	The space of matrices of dimension $n \times n$	1
$\boldsymbol{R}^n$	The space of n-dimensional vectors	1
r	The rank of a matrix	1
$\mathbf{r}$	Residual vector for a linear solver	1

Symbol	Description	Unit
s	Number smaller than the rank of $\mathbf{A}$	1
$\mathbf{T}$	Upper triangular part of a matrix	1
t	Time	s
t^*	Time after change of variables	s
U	Horizontal velocity	$\mathrm{m\,s^{-1}}$
U^*	Velocity along iso-σ lines	$\mathrm{m\,s^{-1}}$
$\tilde{U}$	Provisional horizontal velocity before non-hydrostatic correction is added	$\mathrm{m\,s^{-1}}$
U_b	Velocity at the bottom	$\mathrm{m\,s^{-1}}$
$\mathbf{U}$	Velocity vector	$\mathrm{m\,s^{-1}}$
$\tilde{\mathbf{U}}$	Provisional velocity vector	$\mathrm{m\,s^{-1}}$
W	Vertical velocity in z-coordinates	$\mathrm{m\,s^{-1}}$
$\tilde{W}$	Provisional vertical velocity in z-coordinates before non-hydrostatic correction is added	$\mathrm{m\,s^{-1}}$
W_w	Width of the solitary wave in the initial density distribution	m
$\mathbf{x}$	A vector	1
x	Horizontal coordinate	m
x^*	Horizontal coordinate after change of variables	m
x_1	Limit of integration	1
x_2	Limit of integration	1
$x_{i,k}$	Element of a solution vector	1
y	Horizontal coordinate	m
Z	The σ-coordinate of cell interfaces	1
ZZ	The σ-coordinate of cell centres	1
z	Vertical coordinate in a Cartesian coordinate system	m
z'	Dummy variable for integration	m
z_0	Bottom roughness parameter	m
z_b	Distance from the bottom to the nearest grid point	m
z_i	Interface depth for the initial condition	m
α_k	Parameter in CG	1
β_k	Parameter in CG	1
Δh	Thickness of interface in initial condition	m
Δt	Time step	s
Δx	The grid spacing in x-direction	m
$\Delta\rho$	Density difference in the initial condition	$\mathrm{kg\,m^{-3}}$
ϵ	Tolerated error in iterative methods	1

Symbol	Description	Unit
ζ	Initial interface displacement for the density distribution	m
η	Surface elevation	m
η_i	Discrete representation of the surface elevation in the middle of a cell	m
η^n	The surface elevation after time step n	m
κ	The von Karman constant	1
$\kappa(\mathbf{A})$	Condition number of the matrix $\mathbf{A}$	1
$\kappa_m(\mathbf{A})$	Modified condition number of a matrix	1
λ	Relaxation parameter in SOR	1
ρ	Density	$\mathrm{kg\,m^{-3}}$
$\bar{\rho}$	Average density	$\mathrm{kg\,m^{-3}}$
ρ_0	Reference density	$\mathrm{kg\,m^{-3}}$
$\rho(\mathbf{G})$	The spectral radius of the matrix $\mathbf{G}$	1
$\mathbf{\Sigma}$	Diagonal matrix of singular values	1
σ	Vertical coordinate	1
σ_s	The s'th largest singular value of a matrix	1
$\sigma_{\max}$	The largest singular value of a matrix	1
$\sigma_{\min}$	The smallest singular value of a matrix	1
τ	Bottom drag	$\mathrm{kg\,m^{-1}\,s^{-2}}$
Ω	Domain of integration	1
$\partial\Omega$	The boundary of Ω	1
ω	Vertical velocity in σ-coordinates	$\mathrm{m\,s^{-1}}$
ω^*	Provisional vertical velocity in σ-coordinates before non-hydrostatic correction is added	$\mathrm{m\,s^{-1}}$

Appendix B

Discretizations of the Elliptic Equations

In this appendix we give details on the discretization of the Equations (2.10) and (2.12).

B.1 The Simplified System

In Chapter 2, the equations in the simplified system is combined into the elliptic Equation (2.12)

$$\frac{\partial}{\partial x}\left(D\frac{\partial Q}{\partial x}\right)+\frac{1}{D}\frac{\partial^2 Q}{\partial \sigma^2}=\frac{\rho_0}{\Delta t}\left(\frac{\partial \tilde{U}D}{\partial x}+\frac{\partial \tilde{\omega}}{\partial \sigma}+\frac{\partial \eta}{\partial t}\right).$$

This can be discretised by using finite differences. As mentioned in Chapter 5, the discretisation is not unique. However, one natural way of doing it is

$$\begin{aligned}&\frac{1}{\Delta x}\left(H_{i+1,k}\Big(\frac{Q_{i+1,k}-Q_{i,k}}{\Delta x}\Big)-H_{i,k}\Big(\frac{Q_{i,k}-Q_{i-1,k}}{\Delta x}\Big)\right)\\&\qquad+\frac{1}{H_i}\frac{1}{DZ_k}\left(\frac{Q_{i,k-1}-Q_{i,k}}{DZZ_{k-1}}-\frac{Q_{i,k}-Q_{i,k+1}}{DZZ_k}\right)\\&=\frac{\rho_0}{\Delta t}\left(\frac{U^*_{i+1,k}DU_{i+1}-U^*_{i,k}DU_i}{\Delta x}+\frac{\omega^*_{i,k-1}-\omega^*_{i,k+1}}{DZ_k}+\frac{\eta^n_i-\eta^{n-1}_i}{\Delta t}\right).\end{aligned}\tag{B.1}$$

Here η^n and η^{n-1} are the surface elevation from the present and previous time step, respectively. If other methods of mode splitting are applied, averaged values of η can be used instead, see e.g. Kanarska and Maderich (2003).

B.2 The Full System of Equations

The full system of equations was combined into Equation (2.10)

$$\frac{\partial}{\partial x}\left(D\frac{\partial Q}{\partial x}\right) - \frac{\partial}{\partial x}\left(\frac{\partial Q}{\partial \sigma}\left(\sigma\frac{\partial D}{\partial x} + \frac{\partial \eta}{\partial x}\right)\right) + \frac{1}{D}\frac{\partial^2 Q}{\partial \sigma^2} - \frac{\partial}{\partial \sigma}\left(\frac{\partial Q}{\partial x}\left(\sigma\frac{\partial D}{\partial x} + \frac{\partial \eta}{\partial x}\right)\right) + \frac{1}{D}\frac{\partial}{\partial \sigma}\left(\frac{\partial Q}{\partial \sigma}\left(\sigma\frac{\partial D}{\partial x} + \frac{\partial \eta}{\partial x}\right)^2\right) = \frac{\rho_0}{\Delta t}\left(\frac{\partial \tilde{U} D}{\partial x} + \frac{\partial \tilde{\omega}}{\partial \sigma} + \frac{\partial \eta}{\partial t}\right).$$

For this equation, there are several natural ways of discretising. As mentioned in Chapter 5, the properties of the resulting matrix may be dependent on the discretization. In this thesis, the following discretization has been used:

$$\begin{aligned}
\left.\frac{\partial}{\partial x}\left(D\frac{\partial Q}{\partial x}\right)\right|_{i,k} &\approx \frac{1}{\Delta x}\left[DU_{i+1}\frac{(Q_{i+1,k}-Q_{i,k})}{\Delta x} - DU_i\frac{(Q_{i,k}-Q_{i-1,k})}{\Delta x}\right], \\
\left.\frac{\partial}{\partial x}\left(\sigma\frac{\partial Q}{\partial \sigma}\frac{\partial D}{\partial x}\right)\right|_{i,k} &\approx \frac{ZZ_k}{4\Delta x}\frac{(D_{i+1}-D_i)}{\Delta x}\Big[\frac{(Q_{i+1,k-1}+Q_{i,k-1})}{DZZ_{k-1}} \\
&\quad -\frac{(Q_{i+1,k}+Q_{i,k})}{DZZ_{k-1}} + \Big(\frac{(Q_{i+1,k}+Q_{i,k})}{DZZ_k} \\
&\quad -\frac{(Q_{i+1,k+1}+Q_{i,k+1})}{DZZ_k}\Big)\Big] \\
&\quad -\frac{ZZ_k}{4\Delta x}\frac{(D_i-D_{i-1})}{\Delta x}\Big[\frac{(Q_{i,k-1}+Q_{i-1,k-1})}{DZZ_{k-1}} \\
&\quad -\frac{(Q_{i,k}+Q_{i-1,k})}{DZZ_{k-1}} + \Big(\frac{(Q_{i,k}+Q_{i-1,k})}{DZZ_k} \\
&\quad -\frac{(Q_{i,k+1}+Q_{i-1,k+1})}{DZZ_k}\Big)\Big],
\end{aligned}$$

$$\begin{aligned}
\left.\frac{\partial}{\partial x}\left(\frac{\partial Q}{\partial \sigma}\frac{\partial \eta}{\partial x}\right)\right|_{i,k} &\approx \frac{1}{4\Delta x}\frac{(\eta_{i+1}-\eta_i)}{\Delta x}\Big[\frac{(Q_{i+1,k-1}+Q_{i,k-1})-(Q_{i+1,k}+Q_{i,k})}{DZZ_{k-1}} \\
&\quad +\frac{(Q_{i+1,k}+Q_{i,k})-(Q_{i+1,k+1}+Q_{i,k+1})}{DZZ_k}\Big] \\
&\quad -\frac{1}{4\Delta x}\frac{(\eta_i-\eta_{i-1})}{\Delta x}\Big[\frac{(Q_{i,k-1}+Q_{i-1,k-1})-(Q_{i,k}+Q_{i-1,k})}{DZZ_{k-1}} \\
&\quad +\frac{(Q_{i,k}+Q_{i-1,k})-(Q_{i,k+1}+Q_{i-1,k+1})}{DZZ_k}\Big],
\end{aligned}$$

$$
\begin{aligned}
\frac{1}{D}\frac{\partial^2 Q}{\partial \sigma^2}\bigg|_{i,k} &\approx \frac{1}{D_i}\frac{1}{DZ_k}\Big(\frac{Q_{i,k-1}-Q_{i,k}}{DZZ_{k-1}} - \frac{Q_{i,k}-Q_{i,k+1}}{DZZ_k}\Big),\\
\frac{\partial}{\partial \sigma}\left(\sigma\frac{\partial Q}{\partial x}\frac{\partial D}{\partial x}\right)\bigg|_{i,k} &\approx \frac{1}{4}\frac{D_{i+1}-D_i}{\Delta x}\frac{1}{DZ_k}\Big[\frac{ZZ_{k-1}(Q_{i+1,k-1}-Q_{i,k-1})}{\Delta x}\\
&\quad -\frac{ZZ_k(Q_{i+1,k}-Q_{i,k})}{\Delta x} - \Big(\frac{ZZ_k(Q_{i+1,k}-Q_{i,k})}{\Delta x}\\
&\quad -\frac{ZZ_{k+1}(Q_{i+1,k+1}-Q_{i,k+1})}{\Delta x}\Big)\Big]\\
&\quad +\frac{1}{4}\frac{D_i-D_{i-1}}{\Delta x}\frac{1}{DZ_k}\Big[\frac{ZZ_{k-1}(Q_{i,k-1}-Q_{i-1,k-1})}{\Delta x}\\
&\quad -\frac{ZZ_k(Q_{i,k}-Q_{i-1,k})}{\Delta x} - \Big(\frac{ZZ_k(Q_{i,k}-Q_{i-1,k})}{\Delta x}\\
&\quad -\frac{ZZ_{k+1}(Q_{i,k+1}-Q_{i-1,k+1})}{\Delta x}\Big)\Big],
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial}{\partial \sigma}\left(\frac{\partial Q}{\partial x}\frac{\partial \eta}{\partial x}\right)\bigg|_{i,k} &\approx \frac{\eta_{i+1}-\eta_i}{\Delta x}\Big[\frac{(Q_{i+1,k-1}-Q_{i,k-1})-(Q_{i+1,k}-Q_{i,k})}{DZZ_{k-1}\,\Delta x}\\
&\quad -\frac{(Q_{i+1,k}-Q_{i,k})-(Q_{i+1,k+1}-Q_{i,k+1})}{DZZ_k\,\Delta x}\Big]\\
&\quad +\frac{\eta_i-\eta_{i-1}}{\Delta x}\Big[\frac{(Q_{i,k-1}-Q_{i-1,k-1})-(Q_{i,k}-Q_{i-1,k})}{DZZ_{k-1}\,\Delta x}\\
&\quad -\frac{(Q_{i,k}-Q_{i-1,k})-(Q_{i,k+1}-Q_{i-1,k+1})}{DZZ_k\,\Delta x}\Big],
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial}{\partial \sigma}\left(\frac{\partial Q}{\partial \sigma}\Big(\sigma\frac{\partial D}{\partial x}+\frac{\partial \eta}{\partial x}\Big)^2\right)\bigg|_{i,k} &\approx \frac{1}{DZ_k}\Big[\frac{(Z_k((D_{i+1}-D_{i-1}))+(\eta_{i+1}-\eta_{i-1}))^2}{(\Delta x)^2}\frac{Q_{i,k-1}}{DZZ_{k-1}}\\
&\quad -\frac{(Z_k((D_{i+1}-D_{i-1}))+(\eta_{i+1}-\eta_{i-1}))^2}{(\Delta x)^2}\frac{Q_{i,k}}{DZZ_{k-1}}\\
&\quad -\frac{(Z_{k+1}((D_{i+1}-D_{i-1}))+(\eta_{i+1}-\eta_{i-1}))^2}{(\Delta x)^2}\frac{Q_{i,k}}{DZZ_k}\\
&\quad +\frac{(Z_{k+1}((D_{i+1}-D_{i-1}))+(\eta_{i+1}-\eta_{i-1}))^2}{(\Delta x)^2}\frac{Q_{i,k+1}}{DZZ_k}\Big].
\end{aligned}
$$

The discretization of the right hand side is done as in Equation (B.1). The discretization is done such that the accuracy is of second order, also if a

neighbourhood cell is a land cell. Masking arrays are used to remove land cells.

B.3 The Boundary Conditions

The very upper and lower cells needs special treatment due to the boundary conditions.

B.3.1 Finite Difference Approximations at the Bottom

At the bottom, $\frac{\partial Q}{\partial \sigma} = 0$. This gives us the following finite difference approximations for the lowermost cells

$$\begin{aligned}\left.\frac{\partial}{\partial x}\left(\sigma\frac{\partial Q}{\partial \sigma}\frac{\partial D}{\partial x}\right)\right|_{i,KB-1} \approx\ & \frac{ZZ_{KB-1}}{4\Delta x}\frac{(D_{i+1}-D_i)}{\Delta x}\Big[\frac{(Q_{i+1,KB-2}+Q_{i,KB-2})}{DZZ_{KB-2}} \\ & -\frac{(Q_{i+1,KB-1}+Q_{i,KB-1})}{DZZ_{KB-2}}\Big] \\ & -\frac{ZZ_{KB-1}}{4\Delta x}\frac{(D_i-D_{i-1})}{\Delta x}\Big[\frac{(Q_{i,KB-2}+Q_{i-1,KB-2})}{DZZ_{KB-2}} \\ & -\frac{(Q_{i,KB-1}+Q_{i-1,KB-1})}{DZZ_{KB-2}}\Big],\end{aligned}$$

$$\begin{aligned}\left.\frac{\partial}{\partial x}\left(\frac{\partial Q}{\partial \sigma}\frac{\partial \eta}{\partial x}\right)\right|_{i,KB-1} \approx\ & \frac{1}{4\Delta x}\frac{(\eta_{i+1}-\eta_i)}{\Delta x}\Big[\frac{(Q_{i+1,KB-2}+Q_{i,KB-2})}{DZZ_{KB-2}} \\ & -\frac{(Q_{i+1,KB-1}+Q_{i,KB-1})}{DZZ_{KB-2}}\Big] \\ & -\frac{1}{4\Delta x}\frac{(\eta_i-\eta_{i-1})}{\Delta x}\Big[\frac{(Q_{i,KB-2}+Q_{i-1,KB-2})}{DZZ_{KB-2}} \\ & -\frac{(Q_{i,KB-1}+Q_{i-1,KB-1})}{DZZ_{KB-2}}\Big],\end{aligned}$$

$$\left.\frac{1}{D}\frac{\partial^2 Q}{\partial \sigma^2}\right|_{i,KB-1} \approx \frac{1}{D_i}\frac{1}{DZ_{KB-1}}\frac{Q_{i,KB-2}-Q_{i,KB-1}}{DZZ_{KB-2}},$$

$$\begin{aligned}\left.\frac{\partial}{\partial\sigma}\left(\sigma\frac{\partial Q}{\partial x}\frac{\partial D}{\partial x}\right)\right|_{i,KB-1} \approx\ & \frac{D_{i+1}-D_i}{\Delta x}\Big[\frac{ZZ_{KB-2}(Q_{i+1,KB-2}-Q_{i,KB-2})}{DZZ_{KB-2}\,\Delta x} \\ & -\frac{ZZ_{KB-1}(Q_{i+1,KB-1}-Q_{i,KB-1})}{DZZ_{KB-2}\,\Delta x}\Big] \\ & +\frac{D_i-D_{i-1}}{\Delta x}\Big[\frac{ZZ_{KB-2}(Q_{i,KB-2}-Q_{i-1,KB-2})}{DZZ_{KB-2}\,\Delta x} \\ & -\frac{ZZ_{KB-1}(Q_{i,KB-1}-Q_{i-1,KB-1})}{DZZ_{KB-2}\,\Delta x}\Big] \\ & +\frac{1}{2}\Big(\frac{Q_{i+1,KB-1}-Q_{i,KB-1}}{\Delta x}\frac{D_{i+1}-D_i}{\Delta x} \\ & +\frac{Q_{i,KB-1}-Q_{i-1,KB-1}}{\Delta x}\frac{D_i-D_{i-1}}{\Delta x}\Big)\,, \end{aligned} \tag{B.2}$$

$$\begin{aligned}\left.\frac{\partial}{\partial\sigma}\left(\frac{\partial Q}{\partial x}\frac{\partial \eta}{\partial x}\right)\right|_{i,KB-1} \approx\ & \frac{1}{4\Delta x}\frac{\eta_{i+1}-\eta_i}{\Delta x}\Big[\frac{(Q_{i+1,KB-2}-Q_{i,KB-2})}{DZZ_{KB-2}} \\ & -\frac{(Q_{i+1,KB-1}-Q_{i,KB-1})}{DZZ_{KB-2}}\Big] \\ & +\frac{1}{4\Delta x}\frac{\eta_i-\eta_{i-1}}{\Delta x}\Big[\frac{(Q_{i,KB-2}-Q_{i-1,KB-2})}{DZZ_{KB-2}} \\ & -\frac{(Q_{i,KB-1}-Q_{i-1,KB-1})}{DZZ_{KB-2}}\Big]\,, \end{aligned}$$

$$\begin{aligned}\left.\frac{\partial}{\partial\sigma}\left(\frac{\partial Q}{\partial\sigma}\left(\sigma\frac{\partial D}{\partial x}+\frac{\partial\eta}{\partial x}\right)^2\right)\right|_{i,KB-1} \approx\ & \frac{1}{4(\Delta x)^2}\Big[(ZZ_{KB-2}((D_{i+1}-D_{i-1})) \\ & +(\eta_{i+1}-\eta_{i-1}))\Big]^2\frac{Q_{i,KB-2}}{DZZ_{KB-2}} \\ & -\frac{1}{4D_i(\Delta x)^2}\Big[(ZZ_{KB-2}((D_{i+1}-D_{i-1})) \\ & +(\eta_{i+1}-\eta_{i-1}))^2\frac{Q_{i,KB-1}}{DZZ_{KB-2}}\Big]\,. \end{aligned}$$

The term $\frac{\partial}{\partial x}(D\frac{\partial Q}{\partial x})$ is treated in the same way as for inner points.

B.3.2 Finite Difference Approximations at the Surface

If we apply a Neumann condition on the surface, we get difference approximations similar to the ones used at the bottom, just replace $KB-2$ and $KB-1$ with 1 and 2, respectively. Also, ZZ_1 is the σ-value at the surface, that is zero. Hence, the expressions get somewhat simplified.

For the Dirichlet condition, we have $Q = 0$ at the surface. We will get the same contributions as for the Neumann condition, except from the discretization of the term $\frac{\partial Q}{\partial x}\frac{\partial \eta}{\partial x}$ in Equation (B.2), which disappears as a consequence of σ being equal to zero. In addition, we get some extra terms:

$$\begin{aligned}
\frac{\partial}{\partial x}\left(\frac{\partial Q}{\partial \sigma}\frac{\partial \eta}{\partial x}\right)\bigg|_{i,1} &\rightarrow \frac{1}{DZ_1\Delta x}\Big[\Big(\frac{\eta_{i+1}-\eta_i}{\Delta x}\frac{Q_{i+1,1}+Q_{i,1}}{2}\Big) \\
&\qquad -\Big(\frac{\eta_i-\eta_{i-1}}{\Delta x}\frac{Q_{i,1}+Q_{i-1,1}}{2}\Big)\Big], \\
\frac{1}{D}\frac{\partial^2 Q}{\partial \sigma^2}\bigg|_{i,1} &\rightarrow -\frac{1}{D_i DZ_1}\frac{Q_{i,1}}{\frac{1}{2}DZ_1}, \\
\frac{1}{D}\frac{\partial}{\partial \sigma}\left(\frac{\partial Q}{\partial \sigma}\left(\frac{\partial \eta}{\partial x}\right)^2\right)\bigg|_{i,1} &\rightarrow \frac{1}{DDZ_1}\frac{Q_{i,1}}{\frac{1}{2}DZ_1}\Big(\frac{\eta_{i+1}-\eta_{i-1}}{2\Delta x}\Big)^2.
\end{aligned}$$

B.4 Limitations

There are more ways to do the interpolations needed to estimate the derivatives. The discretization given here is only valid for equidistant σ-layers. For a non-equidistant grid, the accuracy of the approximations given here will be lower. Weights should then be introduced to keep second order accuracy.

References

Adcroft, A., C. Hill, and J. Marshall (1997). Representation of topography by shaved cells in a height coordinate ocean model. *Monthly Weather Review* (125), 2293–2315.

Arakawa, A. (1966). Computational design for long-term numerical integration of the equations of fluid motion: Two-dimensional incompressible flow, part i. *Journal of Computational Physics* (1), 119–143.

Avlesen, H. (2006). Multigrid methods to solve the elliptic equation used to model non-hydrostatic pressure. Personal Communication, Helge.Avlesen@bccs.uib.no.

Berntsen, J. (2004). *USERS GUIDE for a modesplit σ-coordinate numerical Ocean model* (4.1 ed.). Johs. Bruns gt.12, N-5008 BERGEN: University of Bergen. www.mi.uib.no/BOM.

Berntsen, J. and G. Furnes (2005). Internal pressure errors in σ-coordinate ocean models - sensitivity of the growth of the flow to the time stepping method and possible non-hydrostatic effects. *Continental Shelf Research 25*, 829–848.

Berntsen, J., J. Xing, and G. Alendal (2006). Assessment of non-hydrostatic ocean models using laboratory scale problems. *Continental Shelf Research 26*, 1433–1447.

Blumberg, A. and G. Mellor (1987). *Three-Dimensional Coastal ocean Models*, Volume 4 of *Coastal and Estuarine Series*, Chapter A Description of a Three-Dimensional Coastal Ocean Circulation Model, pp. 1–16. American Geophysical Union. ISBN: 0-87590-253-7.

Bourgault, D. and D. Kelley (2003). Wave-induced boundary mixing in a partially mixed estuary. *Journal of Marine Research 63*, 553–576.

Bourgault, D. and D. Kelley (2004). A laterally averaged nonhydrostatic ocean model. *Journal of Atmospheric and Oceanic Technology 21*(12), 1910–1924.

Bourgault, D., D. Kelley, and P. Galbraith (2005). Interfacial solitary wave run-up in the St. Lawrence Estuary. *Journal of Marine Research 65*, 1001–1015.

Casulli, V. (1999). A semi-implicit finite difference method for non-hydrostatic, free-surface flows. *International Journal of Numerical Methods in Fluids 30*, 425–440.

Espelid, T. (2003). Iterative metoder for lineære likningssystemer ++. Lecture note in MAT160, Scientific Computing 1, Course at the University of Bergen.

Evans, L. (1998). *Partial Differential Equations*, Volume 19 of *Graduate Studies in Mathematics*. American Mathematical Society. ISBN: 0-8218-0772-2.

Faber, V. and T. Manteuffel (1984). Necessary and sufficient conditions for the existence of a conjugate gradient method. *SIAM Journal of Numerical Analysis 21*(2), 352–362.

Fringer, O. and R. Street (2003). The dynamics of breaking progressive interfacial waves. *Journal of Fluid Mechanics 494*, 319–353.

Gilbarg, D. and N. Trudinger (1998). *Elliptic Partial Differential Equations of Second Order*. Springer. ISBN 3-540-13025-X.

Gill, A. (1982). *Atmosphere-Ocean Dynamics*. International Geophysics Series. Academic Press. ISBN: 0-12-283522.

Golub, G. and C. van Loan (1996). *Matrix Computations* (3 ed.). John Hopkins. ISBN 0-8018-5414-8.

Haidvogel, D. and A. Beckmann (1999). *Numerical Ocean Circulation Modelling*, Volume 2 of *Series on environmental science and management*. Imperial College Press.

Heggelund, Y., F. Vikebø, J. Berntsen, and G. Furnes (2004). Hydrostatic and non-hydrostatic studies of gravitational adjustment over a slope. *Continental Shelf Research 24*, 2133–2148.

Kanarska, Y. and V. Maderich (2003). A non-hydrostatic numerical model for calculating free-surface stratified flows. *Ocean Dynamics 53*, 176–185.

Kowalik, Z. and T. Murty (1993). *Numerical Modelling of Ocean Dynamics*. World Scientific. ISBN 981-02-1333-6.

Kundu, P. and I. Cohen (2004). *Fluid Mechanics*. Elsevier Academic Press. ISBN 0-12-178253-0.

Legg, S. and A. Adcroft (2003). Internal wave breaking at concave and convex continental slopes. *Journal of Physical Oceanography 33*(11), 2224–2246.

Legg, S., R. Hallberg, and J. Girton (2006). Comparison of entrainment in overflows simulated by z-coordinate, isopycnal and non-hydrostatic models. *Ocean Modelling*. In Press.

Lin, P. and C. Li (2002). A σ-coordinate three-dimensional numerical model for surface wave propagation. *International Journal for Numerical Methods in Fluids 38*, 1045–1068.

Marshall, J., A. Adcroft, C. Hill, L. Perelman, and C. Heisey (1997a). A finite-volume, incompressible Navier Stokes model for studies of the ocean on parallel computers. *Journal of Geophysical Research 102*(C3), 5753–5766.

Marshall, J., C. Hill, L. Perelman, and A. Adcroft (1997b). Hydrostatic, quasi-hydrostatic and nonhydrostatic ocean modelling. *Journal of Geophysical Research 102*(C3), 5733–5752.

Mellor, G. (2004). *USERS GUIDE for a Three-Dimensional, Primitive Equation, Numerical Ocean Model*. Princeton University, Princeton, NJ 08544-0710: Program in Atmospheric and Oceanic Sciences.

Michallet, H. and G. Ivey (1999). Experiments on mixing due to internal solitary waves breaking on uniform slopes. *Journal of Geophysical Research 104*(C6), 13467–13477.

Munk, W. and C. Wunsch (1998). Abyssal recipes II: energetics of tidal and wind mixing. *Deep-Sea Research 45*, 1977–2010.

Osborne, A. and T. Burch (1980). Internal solitones in the Adaman sea. *Science* (208), 451–460.

Patankar, S. (1980). *Numerical Heat Transfer and Fluid Flow*. Taylor & Francis. ISBN 0-89116-522-3.

Peltier, W. and C. Caufield (2003). Mixing efficiency in stratified shear flows. *Annual Review of Fluid Mechanics* (35), 135–167.

Phillips, N. (1957). A coordinate system having some special advantages for numerical forecasting. *Journal of Meteorology 14*, 184–185.

Strang, G. (1988). *Linear algebra and its applications* (3 ed.). Brooks Cole. ISBN 0-15-551005-3.

Trefethen, L. and D. Bau (1997). *Numerical Linear Algebra*. SIAM. ISBN 0-89871-361-7.

van der Vorst, H. (1992). Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM Journal of Scientific and Statistical Computation 13*(2), 631–644.

Young, D. (1971). *Iterative Solution of Large Linear Systems*. Academic Press. ASIN B000H8ZI02.

Ådlandsvik, B. and H. Engedahl (1991). Documentation of a Three-Dimensional Baroclinic Sea Model. Technical report, Bergen, Norway.