On techniques for modelling coastal and ocean flow with unstructured meshes

David Ham

On techniques for modelling coastal and ocean flow with unstructured meshes

David Ham

On techniques for modelling coastal and ocean flow with unstructured meshes

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. dr ir J.T. Fokkema,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op maandag 27 november 2006 om 12.30 uur

door

David Anthony HAM
Bachelor of Science with first class honours in mathematics and
Bachelor of Laws, the Australian National University
geboren te Canberra, Australië

Dit proefschrift is goedgekeurd door de promotor:
Prof. dr ir G.S. Stelling

Toegevoegd promotor:
Dr J.D. Pietrzak

Samenstelling promotiecommissie:

Rector Magnificus	voorzitter
Prof. dr ir G.S. Stelling	Technische Universiteit Delft, promotor
Dr J.D. Pietrzak	Technische Universiteit Delft, toegevoegd promotor
Prof. dr ir A.E. Mynett	Technische Universiteit Delft
Prof. dr ir P. Wesseling	Technische Universiteit Delft
Prof. dr ir A. Heemink	Technische Universiteit Delft
Prof. dr A.E.P. Veldman	Rijksuniversiteit Groningen
Dr Roy A. Walters	National Institute of Water and Atmospheric Research, Nieuw Zeeland

© 2006 David Ham. Permission is granted for the verbatim reproduction in whole or part of this work for any purpose provided that the origin of the work is acknowledged.

ISBN-10: 90-78723-01-7
ISBN-13: 978-90-78723-01-1

Abstract

Computer models of fluid flow play an important role in the study of the world's oceans and seas. Previous generations of models have generally employed Cartesian structured grids. That is to say, the equations of motion have generally been discretised onto grids composed of rows and columns of points arranged in the manner of a chess board. While this can greatly simplify the numerics of such models and can contribute to the computational efficiency of simulations, such grids have difficultly in representing regions with complex geometry, such as the coastlines, and impose limits on the extent to which model resolution, and hence computational effort, can be focused on regions of interest or complexity.

In this thesis, a new unstructured mesh model for coastal and ocean flow is described. In an unstructured mesh, the nodes at which the equations of motion are discretised are placed so as to best represent the geometry of the domain and the flow features of interest to the modeller. The cells which are formed from these nodes do not have a regular chess board structure but rather form an unordered decomposition of the domain. Unstructured meshes provide the flexibility which is lacking in structured grids to focus resolution on features of interest and avoid the unfortunate staircase representation of coastlines which is typical of structured meshes.

Delfin, the model developed in this work and presented here, is a member of the class of C grid staggered variable models which includes many of the existing structured grid models and several new unstructured mesh codes. Three new contributions to the development of this class of model which are implemented in Delfin are described. First, the expanded stencil path integral pressure gradient formulation is applied for the first time in a coastal or ocean model. The advantage of this approach is that poorer quality meshes can be tolerated than is possible with the previously published circumcentre based schemes. The cost of this generality is that the symmetry properties of the continuous pressure gradient operator are not preserved by the discontinuous operator. The implications of this for the stability of the resulting scheme are analysed. The second new development is a trajectory tracking algorithm for semi-Lagrangian advection schemes which is based on analytic rather than numerical integration of the velocity field. This approach prevents the unphysical errors of trajectories crossing or leaving the computational domain. The

analytic integration scheme is derived here and its advantage in preserving the physical properties of trajectories are demonstrated. Finally, an analysis is undertaken of the symmetry properties of the discrete Coriolis operator in the unstructured C grid discretisation. An instability caused by a failure to preserve the symmetry of the continuous Coriolis and pressure gradient operators is identified and a symmetry preserving Coriolis scheme which removes this instability for models with a suitable pressure gradient discretisation is presented.

The value of model development lies in delivering tools which are useful for the study of flow phenomena. As a demonstration of the utility of this work, results of the application of Delfin to the study of the Indian Ocean tsunami are presented as are tests of the model against the problem of calculating the propagation of tides in the North Sea. The Indian Ocean tsunami of 26 December 2004 was caused by ocean bed displacements resulting from an earthquake. For the first time, data from fixed GPS stations were used to determine the motion of the Earth's crust and hence of the sea bed in this event. Delfin was used to model the tsunami wave predicted by the GPS based model of the ocean bed uplift and has been used to validate and differentiate different bed displacement scenarios. The results of a comparison between Delfin and the Dutch Continental Shelf Model (DCSM98) in computing the M2 tide in the North Sea are presented. It is demonstrated that the untuned Delfin simulation produces results which are in general poorer than the highly tuned DCSM98 results but that particular coastal results suggest that the superior representation of coastlines made possible by the unstructured mesh has tangible benefits. It is therefore argued that a fully tuned unstructured mesh simulation of this problem might be expected to exceed the performance of the current structured grid model.

Samenvatting

Numerieke stromingsmodellen spelen een belangrijke rol bij het bestuderen van de oceanen en zeeën op aarde. De vroegere modelgeneraties zijn over het algemeen gebaseerd op Cartesiaanse gestructureerde roosters. Met andere woorden, de bewegingsvergelijkingen worden normaal gesproken gediscretiseerd op roosters, die uit rijen en kolommen zijn samengesteld net zoals op een schaakbord. Hoewel dit de numerieke oplossing van dergelijke modellen zeer kan vereenvoudigen en tot efficiënte computersimulaties kan bijdragen, hebben dergelijke roosters moeite gebieden met complexe geometrieën, zoals de kustlijnen, te representeren en leggen ze grenzen aan de modelresolutie en dus de computerinspanning op, die op ingewikkelde of van belang zijnde gebieden geconcentreerd kan worden.

In dit proefschrift wordt een nieuw model voor kust- en oceaanstromingen beschreven, dat op ongestructureerd roosters gebaseerd is. In een ongestructureerd rooster worden de knopen, waar de bewegingsvergelijkingen gediscretiseerd zijn zo geplaatst, dat ze de vorm van het domein en de voor de modelleur van belang zijnde stromingskenmerken zo goed mogelijk weergeven. De cellen die met deze knopen worden gevormd, hebben geen regelmatige schaakbordstructuur, maar vormen een ongeordende weergave van het domein. Het ongestructureerde rooster geeft in tegenstelling tot het gestructureerde rooster de flexibiliteit om de resolutie op de van belang zijnde kenmerken te concentreren. De ongelukkige trapvormige discretisatie van de kustlijn, die voor gestructureerd roosters kenmerkend is, wordt vermeden.

Delfin, het model dat in dit werk ontwikkeld en gepresenteerd wordt, is één van het staggered C-rooster model type, waartoe veel van de bestaande gestructureerde rooster modellen en verscheidene nieuwe ongestructureerde rooster codes behoren. Drie nieuwe bijdragen aan de ontwikkeling van dit modeltype, die in Delfin geïmplementeerd zijn, worden beschreven. Ten eerste wordt de uitgebreide stencil path integraal formulering voor de drukgradiënt voor het eerst in een kust- of oceaanmodel toegepast. Het voordeel van deze methode is dat een slechtere kwaliteit rooster gebruikt kan worden dan met eerder gepubliceerde, op de circumcenter gebaseerde formuleringen mogelijk is. Deze generalisatie gaat ten kostte van het behoud van de symmetrie-eigenschappen van de continue vorm van de drukgradiënt bij de discretisatie ervan. De implicaties hiervan voor de stabiliteit van het resulterende schema

worden geanalyseerd. De tweede nieuwe ontwikkeling is een stroomlijn algoritme voor een semi-Lagrangiaans advectieschema, dat op de analytische in plaats van de numerieke integratie van het snelheidsveld gebaseerd is. Deze benadering verhindert het onfysische gedrag, dat stroomlijnen elkaar kruisen of het computerdomein verlaten. Het analytische integratieschema wordt afgeleid en het voordeel van het bewaren van de fysische eigenschappen van stroomlijnen wordt aangetoond. Ten slotte wordt een analyse uitgevoerd van de symmetrie-eigenschappen van de gediscretiseerde Coriolis term in een ongestructureerd C-rooster. Een instabiliteit, die veroorzaakt wordt door het verlies van de symmetrie-eigenschappen bij de discretisatie van de drukgradiënt en de Coriolis term wordt geïdentificeerd en een discretisatie, die de symmetrie van de Coriolis term bewaart en daardoor deze instabiliteit bij modellen met een geschikte discretisatie van de drukgradiënt verwijdert, wordt voorgesteld.

Het belang van modelontwikkeling ligt in het leveren van hulpmiddelen, die voor het bestuderen van stromingsfenomenen bruikbaar zijn. Ter demonstratie van het nut van dit werk, worden de resultaten van het toepassen van Delfin op de tsunami in de Indische Oceaan en van getijvoortplantingsberekeningen in de Noordzee gepresenteerd. De tsunami van 26 December 2004 in de Indische Oceaan werd veroorzaakt door verplaatsingen van de oceaanbodem ten gevolge van een aardbeving. Voor het eerst werden de gegevens van vaste GPS posten gebruikt om de beweging van de aardkorst en daardoor die van de zeebodem in deze gebeurtenis te bepalen. Delfin is gebruikt om de tsunamigolf te modelleren die door het GPS gebaseerde model van de oceaanbodemverplaatsing wordt voorspeld. Het is bovendien gebruikt om de verschillende scenario's voor de bodemverplaatsing te valideren en te onderscheiden. De resultaten van een vergelijking tussen Delfin en het Dutch Continental Shelf Model (DCSM98) worden gepresenteerd voor een berekening van het M2 getij in de Noordzee. Er wordt aangetoond dat een ongekalibreerde Delfin simulatie resultaten geeft, die in het algemeen slechter zijn dan de sterk gekalibreerde resultaten van DCSM98, maar dat bepaalde kustresultaten suggereren dat de superieure representatie van de kustlijn, die door het ongestructureerd rooster mogelijk wordt gemaakt, tastbare voordelen biedt. Het is daarom waarschijnlijk, dat een volledig gekalibreerde simulatie van dit probleem met een ongestructureerde rooster, de prestaties van het huidige gestructureerde rooster model zal overtreffen.

Contents

1 Introduction **1**
1.1 C grid models . . . 2
1.2 Structure of the thesis . . . 3

2 Equations and boundaries **5**
2.1 The shallow water equations . . . 5
2.2 Boundary conditions . . . 6
2.2.1 Kinematic boundaries at the surface and bed . . . 6
2.2.2 Wind stress . . . 6
2.2.3 Bottom stress . . . 7
2.3 The depth integrated continuity equation . . . 7
2.4 The linear shallow water equations . . . 8
2.5 Properties of the shallow water equations . . . 8

3 Unstructured Meshes and Variable Placement **11**
3.1 Meshes . . . 11
3.2 Unstructured Triangular Meshes . . . 13
3.2.1 Variable resolution . . . 13
3.2.2 Element Shape . . . 14
3.2.3 Layers in the vertical . . . 14
3.2.4 Mesh quality . . . 16
3.3 Mesh notation . . . 16

4 Discretisation of the equations **19**
4.1 Courant numbers . . . 19
4.2 The continuity equation . . . 20
4.3 The momentum equation . . . 21
4.4 Matrix structure and the free surface equation . . . 22
4.5 Scaling behaviour . . . 23
4.6 An explicit time stepping scheme . . . 24
4.7 A C-D grid discretisation . . . 25
4.8 Discretisation of the linearised equations . . . 26
4.9 A conventional semi-implicit scheme . . . 26

5 The pressure gradient term **29**
5.1 Definition of the wave Courant number . . . 30
5.2 Solution algorithm . . . 31
5.3 The gradient in the tangential direction . . . 32
5.4 Boundary conditions . . . 32
5.5 Stability considerations . . . 34
5.5.1 Checkerboard modes . . . 34
5.5.2 A linear algebra approach to stability . . . 34
5.5.3 A stability proof for the orthogonal mesh case . . . 35
5.5.4 Stability in the non-orthogonal case . . . 36
5.5.5 Eigenvalues . . . 37
5.6 Numerical tests . . . 38
5.6.1 Truncation error . . . 38
5.6.2 Wave calculations . . . 40
5.6.3 Long term wave calculations . . . 41

6 A semi-Lagrangian advection scheme **47**
6.1 Introduction . . . 47
6.2 Trajectories . . . 48
6.3 Integration techniques . . . 49
6.4 Crossing Trajectories . . . 50
6.5 The continuity equation . . . 51
6.6 Closure of streamlines . . . 52
6.7 Increased Accuracy . . . 53
6.8 Results . . . 54
6.9 Numerical issues . . . 55
6.9.1 Finding the cell face . . . 55
6.9.2 Sign uncertainty in the normal velocity . . . 56
6.9.3 The airborne trajectory problem . . . 56
6.10 Interpolation of the velocity . . . 57
6.11 Flow in a river bend . . . 58
6.11.1 Two dimensional test . . . 58
6.11.2 Three dimensional test . . . 60

7 The Coriolis term **63**
7.1 Reconstructing the tangent velocities in a cell . . . 64
7.2 Discretising Coriolis . . . 66
7.3 Physical properties of the reconstruction . . . 68
7.4 Eigenvalues of the systems . . . 68
7.5 Kelvin wave test case . . . 70
7.5.1 Results . . . 70
7.6 Alternative time stepping schemes . . . 72
7.7 Conclusion . . . 73

8 The Indian Ocean Tsunami **75**
8.1 The unstructured mesh . . . 76
8.1.1 Coastlines . . . 76
8.1.2 Prescribed resolution . . . 76
8.1.3 Mesh Generation . . . 77
8.2 Bathymetry . . . 79
8.3 Boundary and parameter values . . . 80
8.4 Observations of tsunami propagation . . . 80
8.4.1 Satellite data . . . 80
8.4.2 Coastal arrival data . . . 82
8.5 Initial conditions . . . 83
8.5.1 Treatment of initial conditions . . . 83
8.5.2 Calculation of the initial fields . . . 84
8.5.3 Incorporating the rupture velocity . . . 86
8.5.4 Simulations run . . . 86
8.6 Performance comparison with the Hansen scheme . . . 87
8.7 Results . . . 87
8.8 Discussion . . . 92
8.9 Conclusion . . . 92

9 Tides in the North Sea **95**
9.1 The mesh . . . 96
9.2 Model parameters and boundary settings . . . 98
9.3 Results and discussion . . . 99
9.4 Conclusion . . . 100

10 Conclusions **105**
10.1 Numerical Developments . . . 105
10.2 Delfin as a tool for ocean science . . . 106
10.3 Recommendations for future work . . . 107

A Some vector calculus **109**
A.1 Product operators . . . 109
A.2 The Nabla operator . . . 110
A.3 The chain rule . . . 110
A.4 The material derivative . . . 111
A.5 The Leibniz integral rule . . . 112

Notation

Differential equation notation

$\mathbf{u}$	velocity vector.
η	surface elevation.
ν^v	vertical viscosity.
ν^h	horizontal viscosity.
ρ	water density.
h	bed elevation.
$\mathbf{\Omega}$	the Earth's rotation vector.
f	the Coriolis parameter.
$\mathbf{u}_\mathrm{a}$	wind velocity.
ρ_a	density of air.
C_a	wind drag coefficient.
C_D	bottom drag coefficient.
κ	the von Karmen constant.

Mesh notation

G	a mesh.
(V, E)	the vertices and the edges of the graph of G.
i	an index in V.
j	an index in E.
A_i	water column cross section area.
A_i	column side area.
$A_{*,k}$	cell face area.
L_*	set of layer indices of a water column or column side.
$\mathbf{n}_{i,*,[,k]}$	cell or column face outward normal vector.
$\mathbf{N}_{*[,k]}$	cell or column face normal vector.
S_i	set of column side indices of water column i.
$u_{*,k}$	cell face normal velocity component.
$v_{*,k}$	cell face tangent velocity component.
$\mathbf{u}_{*,k}$	velocity vector.
$\mathbf{x}_{*[,k]}$	position.
δ_j	$\left\|\mathbf{x}_{i(j,1)} - \mathbf{x}_{i(j,2)}\right\|$.
$\Delta z_{*,k}$	$\left\|\mathbf{x}_{*,k+\frac{1}{2}} - \mathbf{x}_{*,k-\frac{1}{2}}\right\|$.
η_*	free surface elevation.
h_*	bed elevation.
λ_j	length of polygon edge.

Discrete operators

H_j	column face normal pressure gradient.
I_j	column face tangent pressure gradient.
M_j	mass matrix.
$F(u)_{j,k}$	cell face normal advection and Coriolis.

Chapter 1

Introduction

The flow of the world's seas and oceans plays a constant and important part in the lives of the billions of people around the world who live on or near the coast. Recently this has been illustrated most graphically and tragically in the Indian Ocean Tsunami and the flooding of New Orleans by Hurricane Katrina. However, the ongoing processes of transport, erosion, deposition and flooding are omnipresent in the coastal and marine environment. Understanding these processes is an essential part of coastal engineering and of all aspects of oceanography. The importance of these fields and of this understanding will become all the more significant in the coming decades as we attempt to understand and live with the effects on oceans and coastal areas of climate change and sea level rise.

Computer flow models implementing approximate mathematical solutions to the equations of fluid flow have been an important tool in understanding flow processes for decades. Over that period, the amount of processing power available has increased and the numerical techniques used to discretise the equations have become more sophisticated. Nonetheless, computing power is still limited and every algorithm still suffers from limitations whether they be of efficiency, accuracy or applicability. This thesis is an attempt to increase our understanding of one class of models, the staggered variable unstructured mesh finite difference/finite volume models, and to improve the efficiency, accuracy and applicability of those techniques.

A further important goal of this research project was to build a software framework, a toolkit which can be used both by numericists as a test bed for new techniques and by oceanographers and engineers to research flow processes. With the goal of a versatile and useful tool base in mind, the discretisations presented here have been formulated in three dimensions. In particular, it is hoped that a future version of the model will be used to investigate density driven currents in coastal regions. For this purpose, a three dimensional model incorporating baroclinic forcing will be essential. However, the numerical research carried out in this project and presented here has been primarily

directed at the important two dimensional terms in the equations. Fully exploiting the three dimensional nature of the model will be the subject of future research projects.

1.1 C grid models

Various flow models are in use in the Netherlands in research, engineering and operational contexts. Particularly prominent models in current use are the Delft3D package developed and maintained by WL|Delft Hydraulics and Simona developed and maintained by Rijkswaterstaat, the Dutch Directorate for Public Works and Water Management. These models share important features. They are based on structured curvilinear grids and employ the variable staggering scheme known as the C grid.

Structured grids have limited flexibility as a mechanism for describing problem domains so it seems natural to attempt to generalise this C grid approach to unstructured meshes. Nicolaides (1993) developed the idea of a staggered finite volume scheme on an unstructured mesh with his covolume method but it was in Casulli and Walters (2000) that a scheme directly analogous to the structured C grid was published. The existing C grid models benefit from the computational efficiency of the alternating direction implicit (ADI) scheme, a scheme which relies on the rows and columns of cells present in a structured grid and which is therefore totally inapplicable to an unstructured mesh. To keep the computational costs within reasonable limits, Casulli and Walters devised as scheme in which certain terms were dealt with explicitly. This in turn produces a matrix which is partially block diagonal which in turn facilitates the partial solution of the ultimate difference equation with fast direct methods. An implementation of Casulli and Walters (2000) by the authors of that paper is available under the name UnTRIM.

Finite element schemes for computational fluid dynamics on unstructured meshes related to the C grid have also been developed. In particular, the Casulli and Walters (2000) scheme is a close relative of the schemes proposed in Walters and Casulli (1998) and Miglio et al. (1999). Among finite difference/finite volume models, there are also at least three descendants of the Casulli and Walters scheme. The ELCIRC model (Zhang and Baptista, 2005) differs from the other schemes in that full velocity vectors rather than individual velocity components are solved for. This arrangement is known as the C-D grid. Most recently Stanford University have announced their SUNTANS model (Fringer et al., 2006).

The model developed in this project and whose numerics are described in this thesis, Delfin, forms a further member of this growing class of models. In developing Delfin three key aspects of the model have been the focus of work which differs from, and arguably extends, the contributions made by the other models in this class. First, a different pressure gradient formulation enables the use of a wider class of unstructured mesh than is suitable for the other models. Semi-Lagrangian advection schemes are a common feature of the class of models, however the semi-Lagrangian scheme employed in Delfin is based

on an innovative trajectory tracking algorithm which addresses some of the limitations of a semi-Lagrangian approach. Finally, the development of Delfin has lead to the derivation of a symmetry preserving Coriolis scheme which removes an instability observed in other Coriolis discretisations on this form of mesh.

1.2 Structure of the thesis

The thesis can be divided into three groups of chapters. In chapters 2 to 4 the shallow water equations are introduced and the core structure of the discretisations including the mesh structure and variable placement are discussed. In the following three chapters three important horizontal terms in are discussed. In chapter 5 the path integral pressure formulation presented in Wenneker (2002) is compared with the orthogonal formulation of Casulli and Walters (2000). The former scheme is found to facilitate the use of a wider variety of meshes at the cost of the symmetry properties of the discrete pressure gradient operator. The path integral pressure gradient formulation is clearly not original, however this is believed to be the first demonstration of a model in the Casulli class which employs an extended pressure stencil to avoid the mesh orthogonality requirement. Many of the core ideas in these chapters are published in Ham et al. (2005) while the linear algebra stability analysis is also found in Ham et al. (In press).

Chapter 6 is primarily concerned with the trajectory tracking algorithm necessary for the semi-Lagrangian advection scheme. The concept of analytically solving a velocity field for the trajectory was introduced for a structured grid in Dunsbergen (1994) but its generalisation to an unstructured mesh based on triangles is believed to be new. The concepts covered in this chapter have been published in Ham et al. (2006).

In chapter 7 the discretisation of the Coriolis term is addressed. The linear algebra approach to stability employed in chapter 5 is employed to derive a velocity interpolation with favourable symmetry properties. Nonetheless, it is discovered that the lack of symmetry of the path integral pressure gradient scheme results in a much more severe instability in the presence of Coriolis. Stability problems associated with operators which fail to preserve symmetry are well known and well studied, however this is believed to be one of the first investigations of the symmetry properties of the Coriolis term on unstructured C grids. This work has been accepted for publication as Ham et al. (In press).

In chapters 8 and 9 Delfin is applied to two actual flow problems of scientific interest. In the first of these chapters, the 2004 Indian Ocean tsunami is simulated. This is believed to be the first set of simulations conducted based on GPS rather than seismic data. A paper documenting this approach to understanding tsunamis is in preparation (Pietrzak et al., 2006). Chapter 9 is a verification or proof of concept for the use of Delfin to simulate tidal processes in the North Sea. It is found that Delfin performs as well as may be expected given the limitations in the input data and the lack of calibration.

Finally, appendix A provides the vector calculus which is used in the main body of the thesis.

Chapter 2

Equations and boundaries

2.1 The shallow water equations

Coastal flows may be modelled using the shallow water equations, a well known simplification of the Navier-Stokes equations. The shallow water equations may be derived by making the following assumptions about the flow:

1. Vertical accellerations are small compared with gravity. This is equivalent to assuming that the pressure is hydrostatic ($\frac{\partial p}{\partial z} = -\rho g$)

2. Density differences are small compared to the reference density ρ_0 ($\frac{\Delta\rho}{\rho_0} \ll 1$). This is known as the Boussinesq approximation. The result of this is that the density may be assumed to be constant except in the pressure gradient term.

Let $\mathbf{d}$ be any unit vector in the horizontal plane and let $u_{\mathbf{d}} = \mathbf{u} \cdot \mathbf{d}$. Then the shallow water equations may be presented as follows:

$$\nabla \cdot \mathbf{u} = 0 \tag{2.1}$$

$$\begin{aligned}\frac{\mathrm{D}u_{\mathbf{d}}}{\mathrm{D}t} = &\left(\frac{1}{\rho_0}\nabla_{xy}\int_{z'=z}^{z'=\eta}\rho g \mathrm{d}z'\right)\cdot\mathbf{d} + \nabla_{xy}\cdot\nu^h\nabla_{xy}u_{\mathbf{d}} \\ &+ \frac{\partial}{\partial z}\nu^v\frac{\partial u_{\mathbf{d}}}{\partial z} + 2\boldsymbol{\Omega}\cdot\mathbf{k}\,(\mathbf{k}\times\mathbf{u})\cdot\mathbf{d}\end{aligned} \tag{2.2}$$

Here, η is the free surface height, ν^h and ν^v are the horizontal and vertical turbulence viscosities, $\boldsymbol{\Omega}$ is the Earth's rotation vector, $\mathbf{k}$ is the upward unit normal vector and g is acceleration due to gravity.

The Lagrangian or material derivative in the momentum equation:

$$\frac{\mathrm{D}}{\mathrm{D}t} = \frac{\partial}{\partial t} + \mathbf{u}\cdot\nabla \tag{2.3}$$

is deliberately chosen rather than an explicit advection term. For reasons which will be explained in chapter 6, we will adopt a semi-Lagrangian discretisation for the advection of momentum. A derivation of the material derivative and further explanation of the vector calculus notation used in this thesis is presented in appendix A.

We choose to present equation (2.2), the momentum equation, in this form to demonstrate its rotational invariance: it is not necessary to solve for horizontal momentum components in the directions of the conventional basis vectors, and we do not propose to do so. As is usual for the shallow water equations, the vertical velocity may be obtained from equation (2.1), the continuity equation.

In this work, we have not considered the effects of density differences so the density cancels out of the pressure gradient term. In common with similar models (Casulli and Walters, 2000; Zhang and Baptista, 2005), we also disregard horizontal mixing. The actual flow equations solved by Delfin are therefore:

$$\nabla \cdot \mathbf{u} = 0 \tag{2.4}$$

$$\frac{\mathrm{D}u_{\mathbf{d}}}{\mathrm{D}t} = \nabla_{xy}\eta \cdot \mathbf{d} + \frac{\partial}{\partial z}\nu^v \frac{\partial u_{\mathbf{d}}}{\partial z} + 2\boldsymbol{\Omega} \cdot \mathbf{k}\,(\mathbf{k} \times \mathbf{u}) \cdot \mathbf{d} \tag{2.5}$$

2.2 Boundary conditions

2.2.1 Kinematic boundaries at the surface and bed

The water surface and bed are presumed to be impermeable. The result of this is that the Lagrangian surface movement must be equal to the vertical velocity of the fluid at that point. If the surface elevation, η is a suitably smooth function of position in the x, y plane:

$$\begin{aligned} w|_\eta &= \frac{\mathrm{D}\eta}{\mathrm{D}t} \\ &= \frac{\partial \eta}{\partial t} + \mathbf{u}_{xy}|_{z=\eta} \cdot \nabla_{xy}\eta \end{aligned} \tag{2.6}$$

Similarly, assuming that the bed elevation, h, is a smooth function of (x, y) and that the bed is stationary in time we have:

$$\begin{aligned} w|_h &= \frac{\mathrm{D}h}{\mathrm{D}t} \\ &= \mathbf{u}_{xy}|_{z=h} \cdot \nabla_{xy}\eta \end{aligned} \tag{2.7}$$

2.2.2 Wind stress

The wind stress at the surface is given by a quadratic formulation:

$$\nu^v \frac{\partial \mathbf{u}_{xy}}{\partial t} = \frac{\rho_\mathrm{a}}{\rho_0} C_\mathrm{a} \|\mathbf{u}_\mathrm{a}\| \mathbf{u}_\mathrm{a} \tag{2.8}$$

where ρ_{a} is the air density, $\mathbf{u}_{\mathrm{a}}$ is the wind speed at some reference height and C_{a} is a dimensionless drag coefficient. This formulation assumes that the wind velocity is large with respect to the surface water velocity. For situations in which this assumption is not valid, alternative formulations which account for the difference between the wind velocity and the surface water velocity must be employed.

The default air density is set to $1.25\mathrm{kg/m}^3$ while a typical value for C_{a} would be 0.0016.

2.2.3 Bottom stress

The friction with the bed is likewise represented with a quadratic dependence on velocity:

$$\nu^v \frac{\partial \mathbf{u}_{\mathrm{b}}}{\partial t} = C_{\mathrm{D}} \|\mathbf{u}_{\mathrm{b}}\| \mathbf{u}_{\mathrm{b}} \tag{2.9}$$

Where $\mathbf{u}_{\mathrm{b}}$ is the horizontal fluid velocity at the cell nearest the bed. The dimensionless drag coefficient C_{D} may either be specified as a constant (typically 0.0025 for a single layer, depth averaged simulation) or it may be given by:

$$C_{\mathrm{D}} = \left(\kappa \Big/ \ln\left(\frac{z_{\mathrm{b}} + z_0}{z_0} \right) \right)^2 \tag{2.10}$$

where z_{b} is the distance from the bed to the point at which $\mathbf{u}_{\mathrm{b}}$ is evaluated and the von Karmen constant κ has the value 0.4. z_0, the characteristic roughness length of the bed, is a system parameter.

2.3 The depth integrated continuity equation

The kinematic boundary conditions presented in section 2.2.1 may be used in combination with the continuity equation integrated over the water column to produce an expression for the free surface. In the first instance, the divergence of the velocity is integrated over z using the Leibniz integral rule in its divergence operator form:

$$\int_h^\eta \nabla \cdot \mathbf{u} \mathrm{d}z = w|_{z=\eta} - w|_{z=h} + \nabla_{xy} \cdot \int_h^\eta \mathbf{u}_{xy} \mathrm{d}z - \mathbf{u}_{xy} \cdot \nabla \eta + \mathbf{u}_{xy} \cdot \nabla h \tag{2.11}$$

Substituting in the kinematic boundary conditions and using equation (2.4), this becomes:

$$\frac{\partial \eta}{\partial t} + \nabla_{xy} \cdot \int_h^\eta \mathbf{u}_{xy} \mathrm{d}z = 0 \tag{2.12}$$

2.4 The linear shallow water equations

For the purpose of testing individual components of numeric schemes, it is often useful to investigate a simplified problem. In particular, the study of linear differential equations is frequently more straightforward than the nonlinear case. The linear shallow water equations are a two dimensional linear simplification of the shallow water equations. Advection and viscosity are both neglected and the height of the water column is assumed to be almost constant. Representing the water column height as a fixed component, d and a small varying component η and disregarding the varying component in the application of the kinematic boundary conditions, we arrive at:

$$\frac{\partial \eta}{\partial t} + d\nabla \cdot \bar{\mathbf{u}} = 0 \tag{2.13}$$

$$\frac{\partial \bar{\mathbf{u}}}{\partial t} + 2\boldsymbol{\Omega} \times \bar{\mathbf{u}} + g\nabla\eta = 0 \tag{2.14}$$

where $\bar{\mathbf{u}}$ is the depth averaged horizontal velocity vector. Note that d may be allowed to vary in space, so that variable bathymetry may be represented. The linearisation merely demands that d be constant in time. When the effect of dissipative terms is important, a bottom friction term may be added:

$$\frac{\partial \bar{\mathbf{u}}}{\partial t} + 2\boldsymbol{\Omega} \times \bar{\mathbf{u}} + g\nabla\eta - C_{\mathrm{D}}\bar{\mathbf{u}}\|\bar{\mathbf{u}}\|_2 = 0 \tag{2.15}$$

where C_{D} is a dimensionless drag coefficient and $\|\cdot\|_2$ is the Euclidean norm, also referred to as the 2-norm.

2.5 Properties of the shallow water equations

Consider equations (2.13) and (2.14) restricted to one horizontal dimension and in the absence of rotation. Then we have:

$$\frac{\partial \eta}{\partial t} + d\frac{\partial u}{\partial x} = 0 \tag{2.16}$$

$$\frac{\partial \bar{\mathbf{u}}}{\partial t} + g\frac{\partial \eta}{\partial x} = 0 \tag{2.17}$$

This is a linear wave equation whose solutions are disturbances travelling at speed $\pm\sqrt{gd}$ relative to the water. These solutions also admit a an arbitrary background velocity, U, so long as it is constant in space and time. The total propagation speed of the disturbances in a stationary frame of reference is then $U \pm \sqrt{gd}$. The traveling wave solutions with velocity $\sqrt{gd}$ (often also written $\sqrt{gh}$ with h now standing for total water depth) are driven by the gravitation force and are therefore known as gravity waves. In two spatial directions, the same solution types exist but can clearly travel in any direction. From the gravity wave celerity, $\sqrt{gd}$ and the background velocity U it is possible to construct

a dimensionless quantity known as the Froude number:

$$\mathrm{Fr} = |U|/\sqrt{gd} \tag{2.18}$$

A number of important characteristics of fluid flow are related to the Froude number. For example, where the Froude number is greater than 1, gravity waves cannot propagate in the upstream direction. This flow regime is known as supercritical flow. The assumption in the previous section that advection could be ignored is, in fact, valid precisely in a very low Froude number ($\mathrm{Fr} \ll 1$) regime. For a more extensive exploration of the wave solutions of the linear shallow water equations, see Wesseling (2001) chapter 8.

Chapter 3

Unstructured Meshes and Variable Placement

3.1 Meshes

Discrete methods for the approximate solution of partial differential equations rely on approximating the continuous fields of state variables by the values of those state variables at a finite set of points. The grid or mesh used describes the adjacency relationships of that finite set of points and this in turn determines which values directly affect which other values. Associated with a mesh is a graph whose vertices coincide with the mesh elements and whose edges describe the adjacency relationships of the mesh. To complete the topological description of a mesh, we allow for boundary edges which indicate the adjacency of a vertex with a boundary of the problem domain. Notationally, if G is a mesh then we will write an ordered pair (V, E) for its graph where V is the set of vertices of G and E is the set of edges. For further details of graph theory applicable to the understanding of meshes, the reader is referred to Bollobas (1979).

Historically, most environmental flow models have been based on regular quadrilateral or, in three dimensions, hexahedral structured meshes. The term grid is usually used for such a mesh and it is adopted here. Being structured is a graph property of the mesh:

Definition 3.1 *A mesh (V, E) is structured if there exists an integer n, an injective function $i : V \to \mathbb{N}$ and a set of integer offsets $\{m_1, \ldots, m_n\}$ such that:*

1. *Every vertex in V is associated with exactly n edges.*
2. *If $v_1, v_2 \in V$ are adjacent then $i(v_1) = i(v_2) + m_j$ for some $1 \leq j \leq n$.*

Note that the direction of the implication in condition two is important, it is not true that $i(v_1) = i(v_2) + m_j$ implies that v_1 and v_2 are adjacent. Informally,

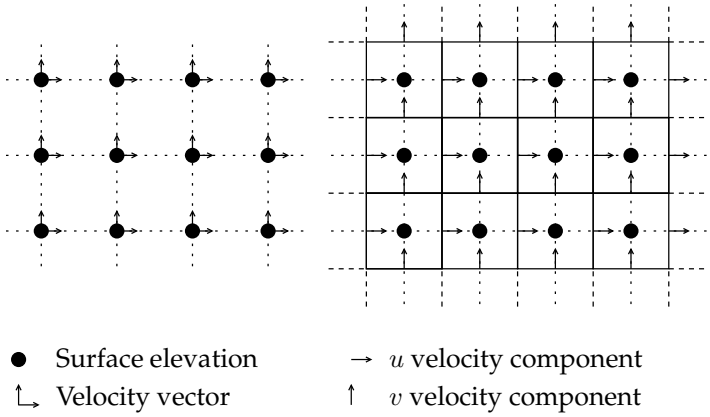

Figure 3.1: Variable placement on an A grid (left) and C grid (right). Dotted lines represent the edges of the associated graph.

a quadrilateral structured mesh is topologically similar to a chess board with rows and columns of cells. The advantage of such structure is that it is not necessary to explicitly store the adjacency relationships between vertices. This dramatically simplifies the task of implementing a discretisation as a computer program and significantly reduces the memory required to store the mesh information. If an implicit time stepping scheme is used, structured meshes usually produce system matrices with at most kn non-zero diagonals where k is a fixed positive integer (and is 5 for many low order schemes in two dimensions). If direct linear algebra solvers are to be used, it is also possible to choose a node enumeration which minimises the matrix bandwidth thereby decreasing the solution time. Indeed, time stepping schemes such as the alternating direction implicit (ADI) scheme have been developed which depend on the row and column structure of quadrilateral structured grids (Peaceman and Rachford, 1955). An unstructured mesh, conversely, is simply any mesh which is not structured. The term unstructured grid is also in use and is synonymous with unstructured mesh.

To this point, we have identified all of the flow variables with the mesh vertices. This approach, dubbed the A grid in the canonical taxonomy of variable placements presented in Arakawa and Lamb (1977) and referred to generally as the collocated grid, is not the only one. Arrangements of the variables which place the discrete velocity points at different locations in space to the discrete surface elevations are referred to as staggered schemes. Of particular importance in many models of free surface flow is the C grid. In this approach, the surface elevation nodes are placed at the centre of rectangular (in three dimensions, hexahedral) cells and only the water elevation or pressure is associated with these points. The full velocity vector is not explicitly represented at any point in space. Rather, the flux from one cell to the next is solved for at the

centre of each cell boundary. Figure 3.1 illustrates the difference between these approaches. From this figure it is apparent that a C grid actually has two graphs associated with it. First there is the mesh which is usually drawn and whose edges make up the boundaries of the cells and whose vertices are the intersections of these edges. Secondly there is a graph whose vertices are the surface elevation points associated with the cells and whose edges are associated with the velocity components stored at the cell boundaries. In the terminology of graph theory, each of these two meshes is the dual of the other. In a C grid discretisation, the former defines geometric properties such as the cell volume, but the latter determines the adjacency relationships which are used to formulate discrete differential operators.

The key advantage of the C grid arises directly out of this staggered placement of variables: inspection of equations (2.5) and (2.12) reveals that the evolution of the free surface elevation in the continuity equation is governed by the divergence of the velocity while the evolution of velocity in the momentum equation is driven by the gradient of the surface elevation. The C grid therefore facilitates the use of central difference type schemes without producing odd-even or checkerboard type instability. The result of this is that, in an inertial frame of reference, the C grid is free of spurious velocity or pressure modes. In the presence of Coriolis acceleration, the C grid suffers from a single Coriolis mode due to the necessity of interpolating the component of the velocity tangent to the cell face. These features of the C grid have been known for decades (see, for example, Walters and Carey (1984)) although an interesting new demonstration based on the linear algebra of the discretised system was presented in Le Roux et al. (2005). It is therefore unsurprising that several unstructured mesh flow models have emerged based on either triangular analogues of the C grid (Casulli and Walters, 2000) or the finite element equivalent, the Raviert-Thomas element (Walters and Casulli, 1998; Miglio et al., 1999).

3.2 Unstructured Triangular Meshes

3.2.1 Variable resolution

In order to achieve accurate results in particular regions of interest, in this case the coastal region, it is necessary to employ a grid with a small mesh size. However, using a very fine grid in regions whose detailed structure is of less interest is a highly inefficient use of computational resources. To be able to resolve the coast accurately while modelling the sea at acceptable cost, it is therefore necessary to vary the grid resolution in space. In addition, since coastal flows are strongly influenced by the shape of the coastline, it is important that flow artifacts introduced by the presence of grid corners on the coast be avoided. While some grid variation is possible using curvilinear structured grids, to achieve highly variable resolution in complex geometries, it is necessary to employ unstructured meshes.

In previous work similar to this, a requirement was made that the grid used

be orthogonal (Casulli and Walters, 2000). That is, the line joining two adjacent column centres must be orthogonal to the column face between them. On a triangular grid, this can be achieved by using the circumcentres as the centres of the triangles. However, unless all of the angles in a triangle are acute, its circumcentre will lie outside the triangle. Generating triangular grids in complex geometries which contain no obtuse angles is a formidable challenge. We therefore choose not to impose an orthogonality requirement. This choice allows the use of a much wider range of grids but has implications for the treatment of the pressure gradient term. More generally, the variability of mesh size and cell angles on an unstructured C grid has implications for the complexity, stability and accuracy of differential operators and mechanisms for overcoming these challenges are a theme of several parts of this thesis.

3.2.2 Element Shape

The generation of unstructured grids based on simplices is a well developed field. A simplex is an n-dimensional polytope (generalised polygon) with exactly $n+1$ vertices. This is in large part due to the long history of unstructured finite element codes. In the shallow water case, the geometry of the problem presents particular challenges to the use of 3 dimensional simplices (tetrahedra) in staggered grid finite volume calculations. Since the horizontal cell dimensions may be orders of magnitude larger than the vertical dimension, the cell face normal vectors of a tetrahedron will be approximately vertical making accurate resolution of horizontal velocities difficult. By instead using prisms as cells we can ensure vertical cell faces thereby obtaining horizontal normal vectors and hence accurate horizontal velocity resolution. Figure 3.2 illustrates this distinction. Some other authors have derived schemes similar to that presented here but which are also applicable to unstructured meshes composed of a mix of triangular and quadrilateral prisms (Casulli and Walters, 2000; Zhang and Baptista, 2005). This approach has not been followed here for two reasons. First, the overwhelming majority of mesh generators available generate triangular meshes. Second, the trajectory tracking algorithm developed in chapter 6 is best suited to triangular meshes. The pressure gradient and Coriolis discretisations presented in chapters 5 and 7 respectively are readily generalisable to the quadrilateral case.

3.2.3 Layers in the vertical

In a shallow water simulation, the vertical dimension differs from the horizontal dimension in important ways. In the previous section the aspect ratio of the flow domain was used to justify the use of prismatic rather than tetrahedral elements. The use of prismatic elements also facilitates the use of a structured mesh in the vertical direction. This results in vertical mixing and pressure gradient operators which are simpler, more efficient and more accurate than would be the case were the mesh unstructured in the vertical.

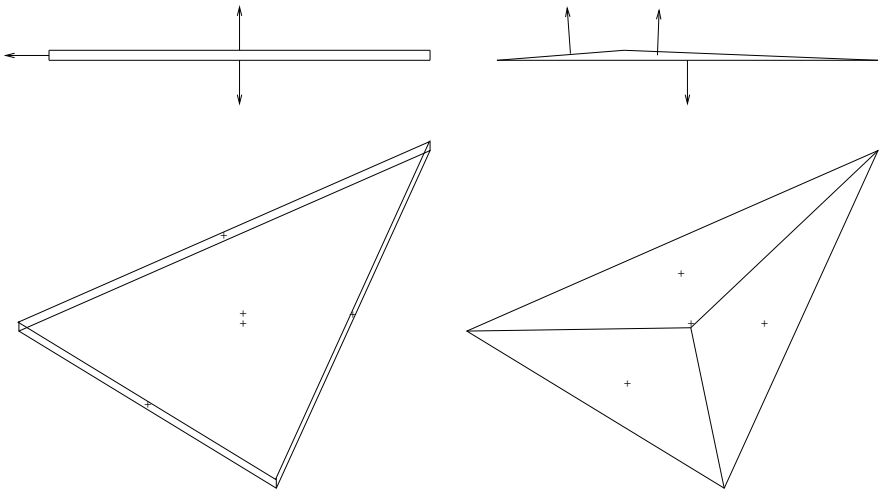

Figure 3.2: Elevation and perspective views of prismatic and tetrahedral elements showing face centres and a representative selection of face normal vectors. It is evident in the case of the tetrahedron that the horizontal components of those vectors are small.

This approach also simplifies the task of mesh generation. First a two dimensional grid is generated for the domain. This mesh defines a set of water columns stretching from the free surface down to the bed. Each water column is then divided into a series of cells. This is currently implemented as a z-level scheme, although other vertical discretisations are in principle possible (Pietrzak et al., 2002). Figure 3.3 shows a slice through part of a mesh demonstrating the vertical structure.

Figure 3.3 also shows the representation of the bed and free surface. Movement in the free surface may cause the number of z-layers in a water column to change resulting in the addition or subtraction of cells at the free surface. At the bed, the exact bathymetry at the water column centre is used to specify the bed elevation for the column. This almost always results in a partial cell at the bottom of the water column. This partial cell representation reduces the extent to which a variable bathymetry is represented as a staircase thereby reducing the likelihood of significant spurious currents occurring due to the errors in the representation of the bathymetry.

An important benefit of this mesh structure is that the momentum evaluation points are aligned vertically. This reduces the evaluation of the vertical viscosity to a one dimensional problem at each water column face. In addition, it means that the pressure gradient term is the same for each momentum point on a water column face. These factors facilitate the creation of an algorithm for the discrete momentum and continuity equations which has advantageous scaling properties. We will return to this matter in section 4.5.

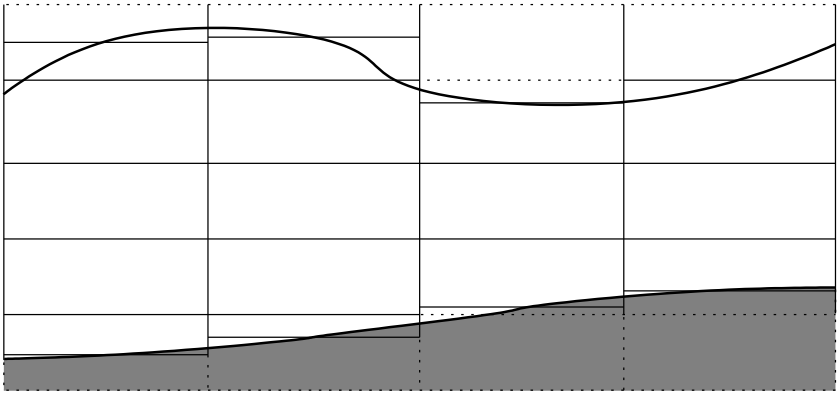

Figure 3.3: Slice through a domain showing the regular grid structure in the vertical and the partial cells at the bed and free surface.

3.2.4 Mesh quality

The accuracy and even stability of many numerical schemes is known to degrade on meshes and grids on which, for example, the resolution of the mesh changes rapidly in space. On an unstructured triangular mesh, the concept of grid quality is the extent to which the mesh deviates from a mesh composed entirely of equilateral triangles. There are a number of related metrics which apply to mesh quality. In particular, the aspect ratio of a triangle or the minimum angle in a triangle are measures of the mesh quality. Causes of mesh quality loss fall into two categories. The first is that meshing algorithms are imperfect and may not result in an optimal mesh. The second is that externally imposed constraints limit the capacity of any meshing algorithm. The key external constraints are the boundary shape and the mesh resolution. Generally speaking, the better resolved a domain, the greater the extent to which the meshing algorithm can produce high quality meshes. An illustration of this is the successively refined circular meshes presented in section 5.6.1.

3.3 Mesh notation

To facilitate discussion of the discretised problem, we adopt the following conventions based loosely on those found in Casulli and Walters (2000). Let G be an unstructured mesh and (V, E) its graph. We adopt the convention that i will always indicate an index in V while j will always indicate an index in E. For example $\mathbf{x}_i$ refers to the position of the ith column centre while $\mathbf{x}_j$ refers to the position of the jth column face. Let S_i be the set of the indices of the column faces adjacent to column i. Then define the function j such that $j(i, l), 1 \leq l \leq |S_i|$ is the index of the lth column face adjacent to column i. Similarly, we define $i(j, l), l \in \{1, 2\}$ to be the column(s) adjacent to face j. Figure 3.4 illustrates this

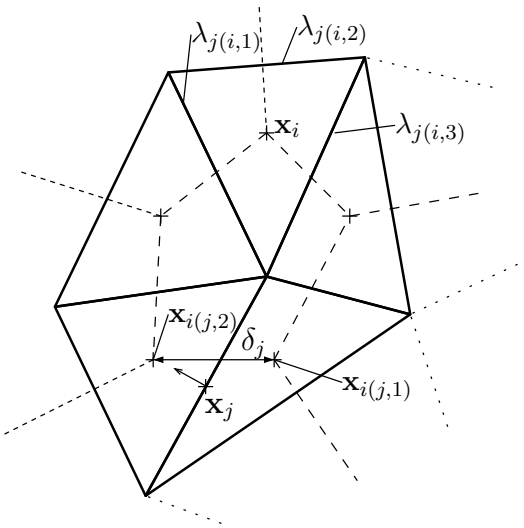

Figure 3.4: Mesh fragment showing indexing conventions for water columns and column faces.

naming scheme. Finally, we will reserve the index k to refer to the layer of a given cell and define L and a function k in an analogous manner to S and the function j. The layer of a horizontal face above level k is $k + \frac{1}{2}$ and the layer below is naturally $k - \frac{1}{2}$. Observe that this numbering is physically consistent in that $(k + 1) - \frac{1}{2} = k + \frac{1}{2}$.

In addition to the index numbers for water columns and column sides, some notation will be used to consistently refer to the data associated with those objects. These are displayed in table 3.1. The notation $\mathbf{n}$ and $\mathbf{N}$ require a little explanation. $\mathbf{N}_j$ is the fixed normal vector associated with column side j. $\mathbf{n}_{i,j}$ on the other hand is the normal vector at side j which points outwards relative to the adjacent column cell i. This is most often used in the case of the quantity $(\mathbf{n} \cdot \mathbf{N})_{i,j}$ which has value 1 if $\mathbf{N}_j$ is outward pointing and -1 if it is inward pointing relative to column i. Note, then, that

$$(\mathbf{n} \cdot \mathbf{N})_{i(j,1),j} = -(\mathbf{n} \cdot \mathbf{N})_{i(j,2),j} \tag{3.1}$$

In the case of horizontal cell faces, we remove some notational clutter by specifying, without loss of generality, that all normal vectors point upwards. That is, we require:

$$\forall i \in V \, \forall k \in L_i \mathbf{N}_{i,k\pm\frac{1}{2}} = (0, 0, 1)$$

A_i	water column cross section area.
A_j	column side area.
$A_{*,k}$	cell face area.
L_*	set of layer indices of a water column or column side.
$\mathbf{n}_{i,*,[,k]}$	cell or column face outward normal vector.
$\mathbf{N}_{*[,k]}$	cell or column face normal vector.
S_i	set of column side indices of water column i.
$u_{*,k}$	cell face normal velocity component.
$\bar{u}_j$	depth averaged column face normal velocity component.
$\mathbf{u}_{*,k}$	velocity vector.
$\bar{\mathbf{u}}_*$	depth averaged velocity vector.
$\mathbf{x}_{*[,k]}$	position.
δ_j	$\left\|\mathbf{x}_{i(j,1)} - \mathbf{x}_{i(j,2)}\right\|$.
$\Delta z_{*,k}$	$\left\|\mathbf{x}_{*,k+\frac{1}{2}} - \mathbf{x}_{*,k-\frac{1}{2}}\right\|$.
η_*	free surface elevation.
h_*	bed elevation.
λ_j	length of polygon edge.

Table 3.1: Notation for data associated with water columns and column sides. An asterisk is a wildcard for a member of $\{i, j\}$ while square brackets ([]) indicate an optional argument. For temporally variable data, time step will be indicated with a superscript.

Chapter 4

Discretisation of the equations

In Casulli and Walters (2000), the authors developed an unstructured generalisation of the C grid discretisation which employs a clever mix of explicit and implicit Eulerian discretisations with a semi-Lagrangian advection scheme to produce an algorithm which is both efficient and stable at large Courant numbers. We will adopt their approach as a point of departure and develop improved discretisations of the pressure gradient, advection and Coriolis terms.

Other variants on the Casulli and Walters (2000) scheme are possible. Zhang and Baptista (2005) supplement the normal velocity components at the vertical cell faces with tangential components, a variable placement scheme dubbed a C-D grid by its original proponents on structured grids (Adcroft et al., 1999). We will also show that a fully explicit time stepping scheme may be applied to the Casulli and Walters (2000) structure. The relative merits of all of these schemes will be investigated.

4.1 Courant numbers

The discretisation choices made here are largely driven by stability requirements of the form first identified by Courant, Friedrichs and Lewy in 1928 (Courant et al., 1928) which are a stability requirement for explicit time stepping schemes. These stability requirements can be stated in terms of dimensionless numbers, called Courant numbers, which describe the rate at which flow disturbances pass through the mesh in comparison with the length of a time step.

The fastest propagating solutions to the shallow water equations are long waves which, as observed in section 2.5 have the velocity $U \pm \sqrt{gh}$. Since coastal and ocean flows typically have flow speeds U of at most a few metres per second while $\sqrt{gh}$ varies from a tens of metres per second near the coast

to hundreds of metres per second in the deap ocean, it is reasonable to neglect the flow speed from the wave speed calculation. The wave Courant number is therefore given by:

$$\mathrm{Cw} := \sqrt{gh}\frac{\Delta t}{\Delta x} \tag{4.1}$$

where Δt is the time step and Δx is the grid spacing. The relevant definition of grid spacing on an unstructured C grid depends on the pressure gradient discretisation adopted. The definition relevant to Delfin will therefore be given in section 5.1.

The stability of explicit advection operators is, in turn, dependant on the rate at which water flows through the mesh. This may be described by the flow Courant number:

$$\mathrm{Cr} := U\frac{\Delta t}{\Delta x} \tag{4.2}$$

where U is the local flow speed.

4.2 The continuity equation

Conservation of fluid volume may be ensured by adopting a finite volume discretisation of the depth integrated momentum equation (2.12):

$$A_i\frac{\partial \eta_i}{\partial t} + \sum_{j\in S_i}\left((\mathbf{n}\cdot\mathbf{N})_{i,j}\sum_{k\in L_j}A_{j,k}u_{j,k}\right) = 0 \tag{4.3}$$

Since coastal and ocean flows typically have very low Froude numbers, the stability of the scheme will therefore depend in large part on the stability of long wave solutions. In particular, it would be advantageous if the stability of long wave solutions were independent of the wave Courant number. This is particularly true on an unstructured mesh since the stability of a Courant limited scheme, as will be demonstrated in section 5.6.2, is dependent on the maximum Courant number in the mesh (see Wesseling, 2001, chaper 8 for derivations of the Courant limitations of various schemes). With the variable mesh resolution which is an intrinsic feature of unstructured meshes, it is exceptionally likely that in a given mesh there will be cells in which the wave Courant number, $\sqrt{gh}\Delta t/\Delta x$, takes a value much greater than that which is characteristic of the mesh as a whole. We therefore choose semi-implicit θ-scheme as the temporal discretisation of the free surface equation. Adopting vector notation to replace the sums over the vertical, we have:

$$\begin{aligned} A_i\eta_i^{n+1} = A_i\eta_i^n &- \theta\Delta t\sum_{j\in S_i}\left((\mathbf{n}\cdot\mathbf{N})_{i,j}\mathrm{A}_j^n\cdot\mathrm{U}_j^{n+1}\right) \\ &- (1-\theta)\Delta t\sum_{j\in S_i}\left((\mathbf{n}\cdot\mathbf{N})_{i,j}\mathrm{A}_j^n\cdot\mathrm{U}_j^{n}\right) \end{aligned} \tag{4.4}$$

Where:

$$\mathrm{A}_j = \begin{bmatrix} A_{j,k(j,1)} \\ \vdots \\ A_{j,k(j,|L_j|)} \end{bmatrix}, \qquad \mathrm{U}_j = \begin{bmatrix} u_{j,k(j,1)} \\ \vdots \\ u_{j,k(j,|L_j|)} \end{bmatrix} \tag{4.5}$$

4.3 The momentum equation

Equation (2.5) is solved for the velocity component normal to each cell face. The terms of the momentum equation have different characteristics which demand different approaches. In particular, the advection and Coriolis operators are both important and troublesome and will be treated in chapters 6 and 7 respectively. For the structure of the solver, it is sufficient at this stage to note that these terms will be dealt with explicitly (at least relative to the pressure term). This choice of explicit and implicit terms results in the advantagious matrix structure described in section 4.4. At this stage, let F be an explicit finite difference operator representing the Coriolis and advection terms. Were horizontal viscosity terms to be included in the equations to be solved, they would also be handled explicity as a part of F.

The discretisation of the pressure gradient term is also non-trivial and will be considered in chapter 5. At this stage we simply define a set of linear operators $H_j : \{\eta_i | i \in G\} \rightarrow \mathbb{R}$ such that H_j approximates $\nabla\eta|_j \cdot \mathbf{N}_j$. For the time evolution of the pressure gradient, we adopt the same θ method chosen for equation (4.4) and for the same reason: to ensure that the propagation of long waves is not subject to a Courant restriction.

The remaining term in equation (2.5) is the vertical viscosity term. On each water column side, the vertical viscosity problem is one-dimensional. It may therefore be discretised using a straightforward central differencing scheme. The viscosity term is a second order operator whose stability is therefore governed by the quantity $\nu^v \Delta t/(\Delta z^2)$ and the typical layer separation, Δz, may be very small in comparison with the other length scales of the problem. We therefore adopt a fully implicit backwards Euler temporal discretisation. We accordingly arrive at a discrete momentum equation:

$$\begin{aligned} u_{j,k}^{n+1} =& F(u)_{j,k}^n - g\Delta t(\theta H_j^{n+1} + (1-\theta)H_j^n) \\ & - \frac{\Delta t}{\Delta z_{j,k}^n}\left(\nu^v_{j,k+\frac{1}{2}} \frac{u_{j,k+1}^{n+1} - u_{j,k}^{n+1}}{\Delta z_{j,k+\frac{1}{2}}^n} - \nu^v_{j,k+\frac{1}{2}} \frac{u_{j,k}^{n+1} - u_{j,k-1}^{n+1}}{\Delta z_{j,k-\frac{1}{2}}^n}\right) \end{aligned} \tag{4.6}$$

Using a vector notation similar to that employed in equation (4.4), we arrive at the following formulation:

$$\mathbf{M}_j^n \mathrm{U}_j^{n+1} = \mathrm{B}_j^n - \theta g \Delta t H_j^{n+1} \Delta \mathrm{Z}_j^n \tag{4.7}$$

Where:

$$\mathbf{M}_j^n = \begin{bmatrix} \Delta z^n_{j,k(j,1)} + a^n_{j,1-\frac{1}{2}} & -a^n_{j,1-\frac{1}{2}} & & \\ -a^n_{j,2+\frac{1}{2}} & \Delta z^n_{j,k(j,2)} + a^n_{j,2+\frac{1}{2}} + a^n_{j,2-\frac{1}{2}} & -a^n_{j,2-\frac{1}{2}} & \\ & \ddots & \ddots & \ddots \\ & & -a^n_{j,|L_j|+\frac{1}{2}} & \Delta z^n_{j,k(j,|L_j|)} + a^n_{j,|L_j|+\frac{1}{2}} \end{bmatrix}$$

with $a^n_{j,l\pm\frac{1}{2}} = \nu^v_{j,k(j,l)\pm\frac{1}{2}} \frac{\Delta t}{\Delta z^n_{j,k(j,l)\pm\frac{1}{2}}}$ and:

$$\mathrm{B}_j^n = \begin{bmatrix} \Delta z^n_j \left(F(u)^n_{j,k(j,1)} - (1-\theta) g \Delta t H^n_j \right) \\ \vdots \\ \Delta z^n_j \left(F(u)^n_{j,k(j,|L_j|)} - (1-\theta) g \Delta t H^n_j \right) \end{bmatrix}, \qquad \mathrm{Z}_j^n = \begin{bmatrix} \Delta z^n_{j,k(j,1)} \\ \vdots \\ \Delta z^n_{j,k(j,|L_j|)} \end{bmatrix}$$

This equation lacks expressions for the wind drag and bed friction boundary conditions. The wind drag term, as it does not depend on the fluid velocity, can be added to the first element of the vector B_j^n. It may be written:

$$\frac{\rho_\mathrm{a}}{\rho_0} C_\mathrm{a} \Delta t \|\mathbf{u}^n_\mathrm{a}\| \mathbf{u}^n_\mathrm{a} \cdot \mathbf{N}_j \tag{4.8}$$

The bed friction is a function if the flow velocity and must be added to the final diagonal element of $\mathbf{M}_j^n$. Its value is:

$$C_\mathrm{D} \Delta t \|\mathbf{u}^n_{j,k(j,|L_j|)}\| \tag{4.9}$$

4.4 Matrix structure and the free surface equation

A particular advantage of the choice of explicit and implicit terms made in Casulli and Walters (2000) and adopted here becomes apparent if we consider the structure of the matrix for the implicit part of the system. Choose a numbering, m, of the velocity unknowns so that $m(j,k+1) = m(j,k)+1$ then the matrix has the form shown in figure 4.1. The upper left quarter of the matrix is block diagonal and is made up purely of the tridiagonal mass matrices $\mathbf{M}_j^n$. These matrices can be inverted using fast direct methods in $\mathcal{O}(M)$ operations where M is the total number of velocity unknowns. Doing so amounts to solving equation (4.7) for U_j^{n+1} and allows the U^{n+1} terms to be eliminated from equation (4.4). This results in a linear implicit system for the free surface elevation:

$$\begin{aligned} A_i \eta_i^{n+1} = A_i \eta_i^n &- \theta \Delta t \sum_{j \in S_i} \left((\mathbf{n}\cdot\mathbf{N})_{i,j} \mathrm{A}^n_j \cdot (\mathbf{M}^n_j)^{-1} \left(\mathrm{B}^n_j - \theta g \Delta t H^{n+1}_j \Delta \mathrm{Z}_j \right) \right) \\ &- (1-\theta) \Delta t \sum_{j \in S_i} \left((\mathbf{n}\cdot\mathbf{N})_{i,j} \mathrm{A}^n_j \cdot \mathrm{U}^n_j \right) \end{aligned} \tag{4.10}$$

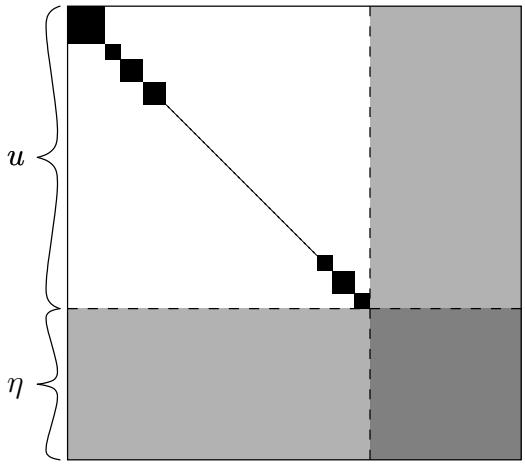

Figure 4.1: The matrix structure of the discrete system. The form of the upper left hand submatrix allows the velocity variables to be eliminated from the equation efficiently.

This equation has the form of a discrete wave equation in η. The θ method temporal discretisation of the free surface terms therefore leads to a system in which, for $\theta \geq 0.5$ the stability of linear long wave solutions is not limited by the wave Courant number (Wesseling, 2001).

The block diagonal structure of the velocity submatrix results directly from the explicit treatment of the advection and Coriolis terms. Any implicit treatment would remove this feature and hence the efficient solution algorithm presented here.

4.5 Scaling behaviour

Equation (4.10) has only $N = |V|$ unknowns: one free surface height for each water column. In contrast, M, the number of velocity unknowns, is equal to the sum over all the column sides of the number of layers at each column side:

$$M = \sum_{j \in E} |L_j| \tag{4.11}$$

The number of column sides in a mesh is proportional to the number of water columns (in a triangular mesh, $|E| \approx 3/2|V|$). If $\bar{L}$ is the average number of column layers then $M \propto \bar{L}N$. The largest implicit system in the algorithm presented here is equation (4.10) with N unknowns. Were all the terms of the equations to be discretised implicitly then then the resulting implicit system would have $M + N \propto \bar{L}N$ unknowns. Since the evaluation of explicit terms scales linearly with the number of unknowns while the solution of implicit systems scales faster than linearly (Trefethen and Bau, 1997), for sufficiently large

problems, the computational work will be dominated by the solution of the linear system. In other words, for sufficiently large problems, the algorithm allows a three dimensional problem to be solved for close to the cost of a two dimensional problem.

4.6 An explicit time stepping scheme

The time stepping scheme detailed above is minimally implicit in the sense that only the fastest modes are solved for implicitly. However, it is still necessary to solve equation (4.10) at each time step and the time taken to do so will increase superlinearly. It is therefore worthwhile considering whether it might not, in fact, be faster to employ an explicit scheme and accept the associated time step limitations.

A well known scheme which may be easily applied to the discretisation here is the scheme presented in Hansen (1956) but reproduced in many sources such as Wesseling (2001). The Hansen scheme is a variant of the even more well known and venerable leapfrog scheme. While in the the leapfrog scheme the velocity and surface elevation are evaluated at the same points in time, the Hansen scheme is distinguished by the evaluation of velocity at half time steps while the free surface elevation remains evaluated at the whole time steps. This algorithm removes the need to store the previous value of the system state. This also effectively removes one initial condition from the scheme which in turn removes a spurious mode to which the leapfrog scheme is subject.

Starting from equation (4.3) and using the velocity at the half time step, we arrive at:

$$A_i\eta_i^{n+1} = A_i\eta_i^n - \Delta t \sum_{j\in S_i}\left((\mathbf{n}\cdot\mathbf{N})_{i,j}\mathrm{A}_j^n \cdot \mathrm{U}_j^{n+\frac{1}{2}}\right) \tag{4.12}$$

where the vectors A_j and U_j have the meanings given above in equation (4.5). This provides the values for η^{n+1} which are required to evaluate $u^{n+3/2}$ using equation (4.6) with $\theta = 1$ and the time steps staggered:

$$\begin{aligned} u_{j,k}^{n+\frac{3}{2}} =& F(u)_{j,k}^{n+\frac{1}{2}} + g\Delta t H_j^{n+1} \\ &+ \frac{\Delta t}{\Delta z_{j,k}^{n+1}}\left(\nu^v_{j,k+\frac{1}{2}}\frac{u_{j,k+1}^{n+\frac{3}{2}} - u_{j,k}^{n+\frac{3}{2}}}{\Delta z_{j,k+\frac{1}{2}}^{n+1}} - \nu^v_{j,k+\frac{1}{2}}\frac{u_{j,k}^{n+\frac{3}{2}} - u_{j,k-1}^{n+\frac{3}{2}}}{\Delta z_{j,k-\frac{1}{2}}^{n+1}}\right) \end{aligned} \tag{4.13}$$

Observe that the vertical viscosity is still dealt with implicitly. The arguments about the special structure of the implicit viscosity matrix still apply in this situation so the stability and time step benefits of using an implicit formulation may be enjoyed without excessive cost. Once again we may write a vector formulation of the equation at one column side:

$$\mathbf{M}_j^{n+1}\mathrm{U}_j^{n+\frac{3}{2}} = \mathrm{B}_j^{n+\frac{1}{2}} - g\Delta t H_j^{n+1}\Delta\mathrm{Z}_j^{n+1} \tag{4.14}$$

all the terms given have the same meanings as in equation (4.6) except for B which becomes:

$$\mathrm{B}_j^{n+\frac{1}{2}} = \begin{bmatrix} \Delta z_j^{n+1} F(u)_{j,k(j,1)}^{n+\frac{1}{2}} \\ \vdots \\ \Delta z_j^{n+1} F(u)_{j,k(j,|L_j|)}^{n+\frac{1}{2}} \end{bmatrix} \tag{4.15}$$

In fact, the evaluation of the momentum is exactly the same as in equation (4.6) with $\theta = 1$, a fact that makes the implementation of the Hansen scheme almost trivial. Note also that since η^{n+1} and therefore Δz^{n+1} are known from equation (4.12), the right hand side of equation 4.14 can indeed be evaluated explicitly to solve for $u^{n+3/2}$.

Since initial conditions for u may be expected to be provided at the same point in time as initial conditions for η, it is necessary to calculate the values $u^{1/2}$. This may be done by simply applying equation 4.14 with a time step of $\Delta t/2$ and all velocities and evaluating surface elevations on the right hand side at $t = 0$.

4.7 A C-D grid discretisation

The Coriolis term presents a particular challenge to the C grid discretisation. The C grid usually ensures that the quantities required for a calculation are available at the point at which that calculation occurs. So, for example, the free surface gradient is required at the velocity points which are between the surface elevation points facilitating a central difference discretisation. However, the evaluation of the Coriolis acceleration in the evolution of a velocity component u requires the evaluation of the face tangent velocity component v, a quantity which is not immediately available. Various interpolation schemes are possible and will be explored in chapter 7 but an alternative possibility was suggested for the structured C grid in Adcroft et al. (1999) and adapted for an unstructured discretisation very similar to that presented here in Zhang and Baptista (2005). The basic idea is that the momentum equation (2.5) can be used to solve for the velocity component tangent to the cell face as well as the normal component solved for in equation (4.7) or (4.14).

We will write $v_{j,k}$ for the velocity component tangential to cell face i, j. If we define C and I in an analogous manner to B and H then the momentum equation for the tangential velocity component becomes:

$$\mathbf{M}_j^n \mathrm{V}_j^{n+1} = \mathrm{C}_j^n - \theta g \Delta t I_j^{n+1} \Delta \mathrm{Z}_j \tag{4.16}$$

It is worth noting that the friction matrix $\mathbf{M}_j$ is rotationally invariant in the horizontal plane so the same matrix may be used for the calculation of U_j and V_j. It should also be noted that V^{n+1} is not used in the calculation of η^{n+1}. The right hand side of equation (4.16) may therefore be calculated explicitly. This also means that V is not coupled to the surface height in the same way

as U. This has the consequence that the full discrete system does not preserve the skew symmetric nature of the original partial differential equation. The importance of this form of symmetry presevation is discussed in chapter 7.

4.8 Discretisation of the linearised equations

Just as equations (2.13) and (2.15) are a simplification of the shallow water equations, the discrete linearised equations are simplifications of equations (4.10) and (4.7). The free surface equation becomes:

$$\begin{aligned} A_i\eta_i^{n+1} =& A_i\eta_i^n \\ &- \theta\Delta t \sum_{j\in S_i} d_j(\mathbf{n}\cdot\mathbf{N})_{i,j}A_j \cdot (M_j^n)^{-1}\left(f(u)_j^n - \Delta tg\left(\theta H^{n+1} + (1-\theta)H^n\right)\right) \\ &- (1-\theta)\Delta t \sum_{j\in S_i} (\mathbf{n}\cdot\mathbf{N})_{i,j}A_j u_j^n \end{aligned} \tag{4.17}$$

Observe that the water column face areas, A_j, and water depths at the faces, d_j, are independent of time.

Similarly, the momentum equation becomes:

$$u_j^{n+1} = (M_j^n)^{-1}\left(f(u)_j^n - \theta\Delta tg\nabla H_j^{n+1} - (1-\theta)\Delta tgH_j^n\right) \tag{4.18}$$

In each case, M_j^n is $d_j + \Delta tC_d\|\mathbf{u}_j^n\|^2$ and $f(u)$ is a linear operator which applies the Coriolis term to u_j^n. In fact, equations (4.17) and (4.18) are simply equations (4.10) and (4.7) subject to the constraints that there be only one layer, that advection is disregarded and that the water depth is held constant in time. The Hansen and C-D grid schemes may be applied *mutatis mutandi* to the linearised problem.

4.9 A conventional semi-implicit scheme

For the purpose of conducting stability analyses, it can be convenient to express the whole system of equations in a semidiscrete form and then to solve the resulting system of ordinary differential equations in time. Due to the complexity of stability analysis and the increased computational cost of this method as opposed to those described earlier in this chapter, our consideration of this mechanism will be limited to the two dimensional linear version of the equations. The semidiscrete form of equations (2.13) and (2.14) may be written as a single matrix equation:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\mathbf{u}\\ \eta\end{bmatrix} = \begin{bmatrix}\mathbf{F} & \mathrm{P}\\ \mathrm{C} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{u}\\ \eta\end{bmatrix} \tag{4.19}$$

where $\mathbf{u}$ is the vector of all face normal depth averaged velocity components and η is the vector of all free surface elevations. In this equation, C C is the

discrete divergence operator, that is the right hand side of the linear two dimensional form of (2.12):

$$\frac{\partial \eta_i}{\partial t} = -\frac{1}{A_i} \sum_{j \in S_i} ((\mathbf{n} \cdot \mathbf{N})_{i,j} \lambda_j d_j \bar{u}_j) \tag{4.20}$$

F contains the contributions due to the Coriolis term and P contains the discrete pressure gradient term. This vector ODE can be integrated in time using a θ scheme applied directly to the whole equation. If we write Q for the system matrix:

$$\mathrm{Q} = \begin{bmatrix} \mathbf{F} & \mathrm{P} \\ \mathrm{C} & \mathbf{0} \end{bmatrix} \tag{4.21}$$

and $\mathbf{v}$ for a system state vector in $\mathbf{u}$ and η then we have:

$$(\mathrm{I} - \theta \Delta t \mathrm{Q}) \mathbf{v}^{n+1} = (\mathrm{I} + (1 - \theta) \Delta t \mathrm{Q}) \mathbf{v}^n \tag{4.22}$$

Chapter 5

The pressure gradient term

On an orthogonal staggered mesh, the pressure gradient can be discretised in a straightforward manner with a central difference scheme:

$$H_j = (\mathbf{n}\cdot\mathbf{N})_{i(j,1),j}\frac{\eta_{i(j,2)} - \eta_{i(j,1)}}{\delta_j} \tag{5.1}$$

In the non-orthogonal case, the interval joining two adjacent column centres is not parallel to the normal of the face between those water columns. We may therefore choose use more information, and hence a larger stencil, to calculate the projection of the discretised pressure gradient onto the column face normal. We adopt here the path integral method introduced for curvilinear structured grids in van Beek et al. (1995) and generalised to unstructured meshes in Wenneker et al. (2002). The essence of this approach is that the positions of nearby column centres are used to construct a basis for $\mathbb{R}^2$ and the gradient of the pressure field is evaluated with respect to that basis by applying a central difference approximation on the surface elevations at those column centres. The gradient vector so constructed is then projected onto the water column face normal.

For some arbitrary $e_j \in E$ we will construct $\{\mathbf{t}_{j,1}, \mathbf{t}_{j,2}\}$, a unit (although not necessarily orthogonal) basis for $\mathbb{R}^2$. We choose:

$$\mathbf{t}_{j,1} = (\mathbf{n}\cdot\mathbf{N})_{i(j,1),j}\frac{\mathbf{x}_{i(j,2)} - \mathbf{x}_{i(j,1)}}{\delta_j}$$

so that:

$$\nabla\eta \cdot \mathbf{t}_{j,1} \approx \frac{\eta_{i(j,2)} - \eta_{i(j,1)}}{\delta_j}$$

For numerical reasons, it is important that $|\mathbf{t}_{j,1} \cdot \mathbf{t}_{j,2}| \ll 1$. That is, the basis vectors should be far from collinear. This can be achieved by using the four water column centres $\{\alpha, \beta, \gamma, \delta\}$ indicated in figure 5.1 to interpolate the surface

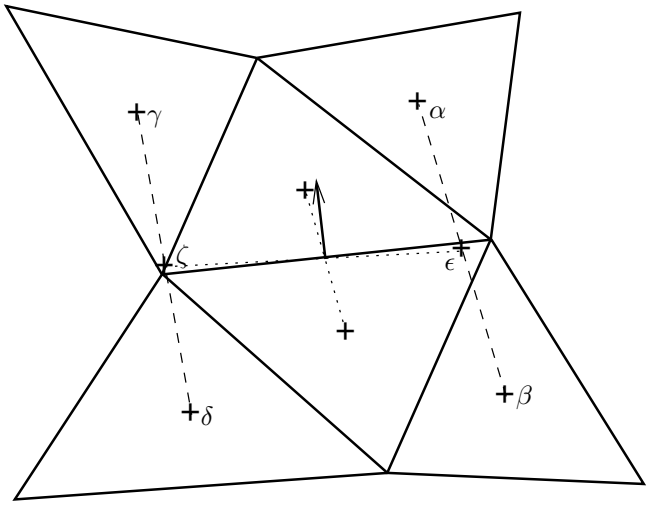

Figure 5.1: The stencil of the path integral discretisation for the pressure gradient term

elevations at ϵ and ζ. We then choose:

$$\begin{aligned}\mathbf{t}_{j,2} &= \frac{\mathbf{x}_\epsilon - \mathbf{x}_\zeta}{\|\mathbf{x}_\epsilon - \mathbf{x}_\zeta\|} \\ &= \frac{\mathbf{x}_\alpha + \mathbf{x}_\beta - \mathbf{x}_\gamma - \mathbf{x}_\delta}{\|\mathbf{x}_\alpha + \mathbf{x}_\beta - \mathbf{x}_\gamma - \mathbf{x}_\delta\|}\end{aligned}$$

The corresponding component of the pressure gradient vector is then given by:

$$\nabla\eta \cdot \mathbf{t}_{j,2} \approx \frac{\eta_\alpha + \eta_\beta - \eta_\gamma - \eta_\delta}{\|\mathbf{x}_\alpha + \mathbf{x}_\beta - \mathbf{x}_\gamma - \mathbf{x}_\delta\|}$$

After some rearrangement, this yields an expression for the water column face normal component of the pressure:

$$H_j = \left(\begin{bmatrix} \mathbf{t}_{j,1}^T \\ \mathbf{t}_{j,2}^T \end{bmatrix}^{-1} \mathbf{N}_j \right) \cdot \begin{bmatrix} (\mathbf{n}\cdot\mathbf{N})_{i(j,1),j} \frac{\eta_{i(j,2)} - \eta_{i(j,1)}}{\delta_j} \\ \frac{\eta_\alpha + \eta_\beta - \eta_\gamma - \eta_\delta}{\|\mathbf{x}_\alpha + \mathbf{x}_\beta - \mathbf{x}_\gamma - \mathbf{x}_\delta\|} \end{bmatrix} \tag{5.2}$$

By construction of $\mathbf{t}_{j,1}$, in the special case where $\mathbf{t}_{j,1} = \mathbf{N}_j$, H_j reduces to (5.1). It should be noted that, for each j, H_j is a linear operator on $\{\eta_i\}$ with coefficients constant in time. It is therefore only necessary to calculate these coefficients once for each water column face.

5.1 Definition of the wave Courant number

For the low Froude number regimes which are typical for coastal and ocean flows, the stability of an explicit time stepping scheme is dependent on the

wave Courant number:

$$\mathrm{Cw} := \sqrt{gh}\frac{\Delta t}{\Delta x} \tag{5.3}$$

Clearly h varies in space in any model which incorporates variable bathymetry or a moving free surface. However on an irregular mesh, Δx is also a function of position. To determine precisely what Δx means in this context, we return to the original analysis presented in Courant et al. (1928) and recall that the Courant number is the ratio of the distance travelled in one time step by a characteristic solution to the differential equation to the width of the domain of influence of the discrete equation. In the case of long wave propagation, the surface elevations are coupled by the pressure gradient operator so the width of the domain of influence is that of the stencil of the discrete pressure gradient operator. The pressure gradient operator is associated with a water column side so it makes sense to define Δx and hence Cw at those points. By the construction of H, there are in effect two finite difference operations each over a separate interval in $\mathbf{x}$. Since the essence of the CFL condition is to establish whether the domain of influence is broad enough compared with the propagation speed of the characteristics, it is appropriate to determine Δx by taking the minimum of the candidate values:

$$\Delta x_j = \min\left\{\Delta x_{i(j,1)i(j,2)}, \Delta x_{\alpha\delta}, \Delta x_{\alpha\gamma}, \Delta x_{\beta\delta}, \Delta x_{\beta\gamma}\right\} \tag{5.4}$$

where $\Delta x_{ab} = \|\mathbf{x}_a - \mathbf{x}_b\|$. However unless the mesh is of particularly poor quality, equation (5.4) reduces to $dx_j = \Delta x_{i(j,1)i(j,2)}$, a more convenient quantity to evaluate.

On the assumption that h does not vary significantly over the stencil of H, we may define:

$$\mathrm{Cw}_j := \sqrt{gh_j}\frac{\Delta t}{\Delta x_j} \tag{5.5}$$

When determining stability questions, it is the maximum wave Courant number which is applicable:

$$\mathrm{Cw}_{\max} = \max_{j\in E}\{\mathrm{Cw}_j\} \tag{5.6}$$

However the Courant number also serves as an indication of the relation between the spatial and the temporal resolution. In this context, the mean wave Courant number ($\mathrm{Cw}_{\mathrm{mean}}$, derived in the obvious way) is a more meaningful measure.

5.2 Solution algorithm

An advantage of pressure gradient discretisation on orthogonal meshes is that equation (5.1) represents a symmetric operator and the resulting free surface matrix is symmetric positive definite (Casulli and Walters, 2000). This enables the use of the conjugent gradient method which is efficient and guaranteed to converge. The path integral formulation does not share this property and the

resulting matrix must therefore be solved using a more general technique such as GMRES or BiCGStab. It should be noted, however, that on a high quality mesh, that is one in which the cells are close to regular in shape, the path integral formulation is dominated by $\mathbf{t}_{j,1}$ so that the resulting matrix may be considered a perturbed symmetric matrix. This makes the generalised conjugate gradient method of Concus and Golub (1976) applicable, and this algorithm has been successfully applied in Delfin for large problems (see chapters 8 and 9). The additional cost of the matrix solution compared with the conjugate gradient method is not excessive although further improvements in the solvers used have the potential to significantly reduce execution times. Similarly, no significant convergence problems have been encountered.

5.3 The gradient in the tangential direction

For calculations using the C-D grid discretisation presented in section 4.7 it is also necessary to derive an expression for I_j, the column side tangential pressure gradient component in (4.16). In the orthogonal case, (5.1) provides no information concerning the tangential component of the pressure gradient so Zhang and Baptista (2005) resort to interpolating the surface heights at the corners of the water columns. The path integral formulation adopted here, however, yields a direct expression for the full pressure gradient vector so that an expression for the tangential component follows immediately:

$$I_j = \left(\begin{bmatrix} \mathbf{t}_{j,1}^T \\ \mathbf{t}_{j,2}^T \end{bmatrix}^{-1} \mathbf{T}_j \right) \cdot \begin{bmatrix} (\mathbf{n}\cdot\mathbf{N})_{i(j,1),j} \frac{\eta_{i(j,2)} - \eta_{i(j,1)}}{\delta_j} \\ \frac{\eta_\alpha + \eta_\beta - \eta_\gamma - \eta_\delta}{\|\mathbf{x}_\alpha + \mathbf{x}_\beta - \mathbf{x}_\gamma - \mathbf{x}_\delta\|} \end{bmatrix} \tag{5.7}$$

where $\mathbf{T}_j$ is the horizontal column face tangent vector. It should be noted that (5.7) is not required for the solution of the surface elevation field so that (4.16) is, in fact, explicit in η.

5.4 Boundary conditions

Near boundaries, the stencil of the discrete pressure gradient operator crosses the boundary of the domain with the result that one or more of the column centres used in (5.2) is absent. Wenneker (2002) advocates replacing the missing column centres with the adjacent centre. This approach may result in local basis vectors $\{\mathbf{t}_{j,1}, \mathbf{t}_{j,2}\}$ which are less orthogonal than in the interior case. In some cases, such as that presented in figure 5.2, the basis vectors even become collinear and the algorithm breaks down.

If instead we replace any missing elevation points with points on the boundary, we remove the breakdown circumstances and further provide a structure in which different forms of boundary condition can be incorporated through the same structure. Assume without loss of generality that column centre α is absent. Then, as shown in figure 5.3, we choose $\mathbf{x}_{\hat{\alpha}}$ to be the nearest point to

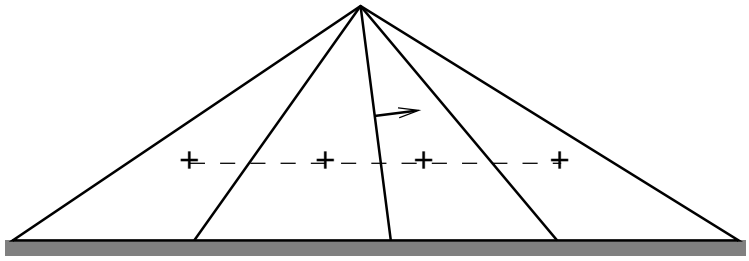

Figure 5.2: A boundary case in which replacing absent column centres with the adjacent column centre causes the pressure gradient algorithm to break down.

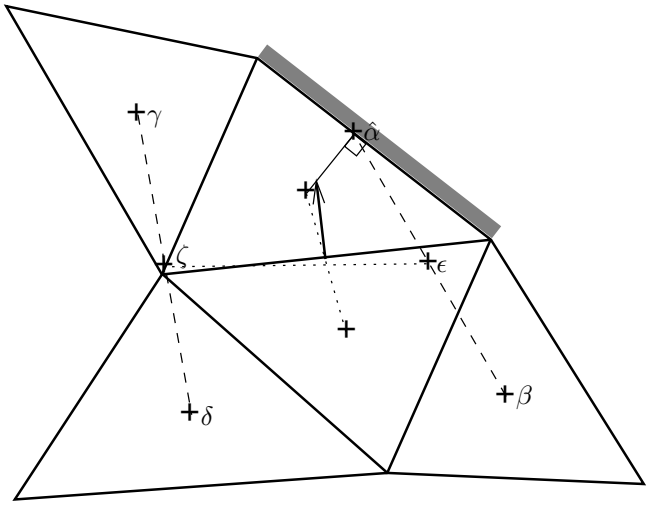

Figure 5.3: Where a boundary falls in the stencil of the pressure gradient operator, the nearest point on the boundary is used as the surface elevation point.

the remaining centre, $i(j,1)$ on the boundary side. The value of the water level at this side then depends on the boundary conditions in force. For example, if the boundary is closed then the approximation $\eta_{\hat{\alpha}} = \eta_{i(j,1)}$ is appropriate while a boundary condition which prescribes the surface elevation at the boundary will directly specify the value at $\mathbf{x}_{\hat{\alpha}}$.

A boundary-like issue which affects the pressure gradient term occurs when a water column face dries. The water column face at the centre of the stencil is no issue since this face will be excluded from the calculation when it dries. However when one of the other four faces is dry then a particular issue arises. Assume without loss of generality that the edge between column α and $i(j,1)$ is dry. The actual value η_α cannot be used. Not only does this correspond to the theoretically objectionable concept of conveying information over a dry boundary, in practice the η_α will frequently correspond to the elevation of a dry

shoreline which is higher than the elevation $\eta_{i(j,1)}$. If this is the case then using η_α will result in a spurious current in the offshore direction. The spurious pressure gradient which causes this current may be avoided by the simple expedient of assuming, for the purpose of the pressure gradient calculation, that $\eta_\alpha = \eta_{i(j,1)}$.

5.5 Stability considerations

5.5.1 Checkerboard modes

One of the reasons for choosing a staggered variable placement scheme is that the checkerboard instability associated with central difference operators is avoided. If we revisit briefly the causes of this instability then some insight into the operation of the present discrete gradient operator may be gleaned. Recall that in 1 dimension, the central difference pressure gradient operator on a uniform collocated grid is:

$$\left.\frac{\partial \eta}{\partial x}\right|_{x_i} = \frac{x_{i+1} - x_{i-1}}{2\Delta x} \tag{5.8}$$

The checkerboard wiggles associated with this operator result from the fact that this is a second order difference equation. On a poor quality mesh, a similar phenomenon may be exhibited by the scheme presented here. In the special case of a regular mesh of equilateral triangles the difference equation is first order. However, as the mesh quality decreases, the weights associated with the higher order terms of the difference equation increase. Figure 5.4 shows an extreme case in which the surface elevation points closest to the relevant water column face are excluded completely from the evaluation of the pressure gradient. In practice, it is unlikely that this situation will occur on a generated mesh, however on a poor quality mesh there are likely to be instances where the coefficients of the adjacent surface elevation points become sufficiently small so that partial decoupling from these values occurs. The occurrence of checkerboard modes on poor quality meshes will be investigated further in section 5.6.2.

5.5.2 A linear algebra approach to stability

Further analysis of the stability implications of the choice of pressure gradient operator may be observed if we examine the properties of the semidiscrete system matrix for the two dimensional linear shallow water equations presented in section 4.9. In the absence of Coriolis, (4.19) becomes:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\mathbf{u}\\ \eta\end{bmatrix} = \begin{bmatrix}\mathbf{0} & \mathrm{P}\\ \mathrm{C} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{u}\\ \eta\end{bmatrix} \tag{5.9}$$

To show that a scheme is both stable and non-dissipative, it is sufficient to show that this matrix has only imaginary eigenvalues. This in turn may be proven by

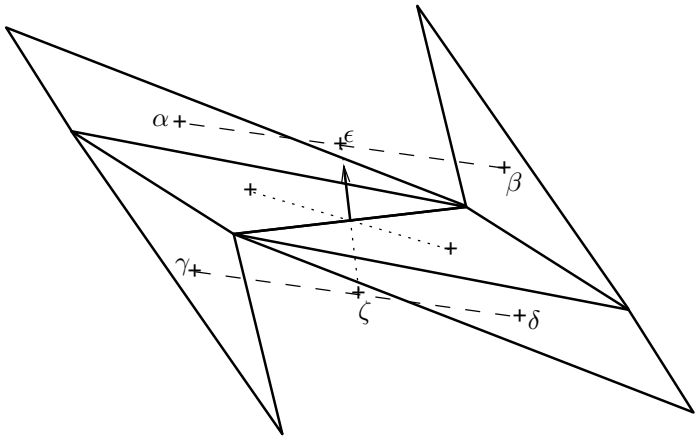

Figure 5.4: On this extremely poor quality mesh fragment the adjacent surface elevation points are completely absent from the pressure gradient stencil.

showing that the matrix, which we shall refer to as A, is skew symmetric or is skew-symmetric under multiplication with some diagonal scaling matrix D.

5.5.3 A stability proof for the orthogonal mesh case

On an orthogonal mesh, with discrete pressure gradient operator given by equation (5.1), the scheme has a property which Espelid et al. (2000) termed 'sign skew symmetry'. That is, for any i, j, either $a_{ij} = a_{ji} = 0$ or $a_{ij}a_{ji} < 0$. To see this, note that if j is an edge of triangle i then $a_{ij} = \pm d_j\lambda_j/A_i$ and $a_{ji} = \mp g/\delta_j$. On the other hand, if edge j is not an edge of triangle i then $a_{ij} = a_{ji} = 0$. It is now straightforward to construct the diagonal scaling matrix D. If we write:

$$\mathrm{D} = \begin{bmatrix} \mathrm{D_P} & \mathbf{0} \\ \mathbf{0} & \mathrm{D_C} \end{bmatrix} \tag{5.10}$$

such that the blocks of D are the same size as those of A then for $\mathrm{D_P}$, $d_{jj} = \delta_j d_j \lambda_j$ and for $\mathrm{D_C}$, $d_{ii} = gA_i$. With this choice of D, DA is skew symmetric and hence the eigenvalues of A are purely imaginary. Espelid et al. (2000) adopt an alternative approach in which the matrix A is made skew-symmetric through the use of a similarity transform $\bar{\mathrm{D}}^{-1}\mathrm{A}\bar{\mathrm{D}}$. This approach produces an equivalent result to that given here and the two diagonal matrices are related by $\mathrm{D} = \bar{\mathrm{D}}^{\mathrm{T}}\bar{\mathrm{D}}$.

Having shown that the orthogonal mesh scheme is scaled skew-symmetric, we can also gain some insight into the energy conservation properties of this scheme.It is worth recalling at this point that the water depth d is assumed to be constant in time. Further, assume that the boundaries of the domain are closed and that no forcing is applied and let $\mathbf{v}$ be any real state vector in u and η. Because DA is a real skew symmetric matrix we know that $\mathrm{sym}(\mathrm{DA}) = 0$,

where $\mathrm{sym}(\mathrm{DA}) = 1/2(\mathrm{DA} + (\mathrm{DA})^{\mathrm{T}})$ is the symmetric part of DA. Therefore

$$\frac{d}{dt}\tfrac{1}{2}\langle \mathbf{v}|\mathrm{D}|\mathbf{v}\rangle = \langle \mathbf{v}|\,\mathrm{sym}(\mathrm{DA})|\mathbf{v}\rangle = 0. \tag{5.11}$$

Indeed this can be seen as conservation of energy, as

$$\tfrac{1}{2}\langle \mathbf{v}|\mathrm{D}|\mathbf{v}\rangle = \sum_{i\in V} \tfrac{1}{2}A_i\ g\eta_i^2 + \sum_{j\in E} \tfrac{1}{2}\Delta x_j d_j \lambda_j\ u_j^2 \tag{5.12}$$

$$= \sum_{i\in V} \tfrac{1}{2}A_i\ g\eta_i^2 + \sum_{i\in V}\sum_{j\in S_i} \tfrac{1}{2}\Delta x_{ij}^c d_j \lambda_j\ u_j^2 \tag{5.13}$$

Where once again V is the set of water column indices and E is the set of column faces. Δx_{ij}^c is the orthogonal distance of the centre of water column i from face i. This can be seen as analogous to the proof of conservation of energy in the continuous equations:

$$\frac{\partial}{\partial t}\left(\tfrac{1}{2}g\eta^2 + \tfrac{1}{2}d\mathbf{u}\cdot\mathbf{u}\right) = g\eta\frac{\partial \eta}{\partial t} + d\mathbf{u}\cdot\frac{\partial \mathbf{u}}{\partial t} \tag{5.14}$$

$$= -g\eta\nabla\cdot d\mathbf{u} + gd\mathbf{u}\cdot\nabla\eta \tag{5.15}$$

$$= -\nabla\cdot\eta d\mathbf{u} \tag{5.16}$$

$$= 0 \tag{5.17}$$

The final step in this proof requires the application Gauss's theorem to the right hand side of equation (5.16). Since the boundaries of the domain were assumed to be closed, the normal velocity in the resulting loop integral is uniformly zero yielding the result given.

5.5.4 Stability in the non-orthogonal case

On a non-orthogonal mesh, with pressure gradient operator given by equation (5.2), the system matrix is not sign skew symmetric. It is true that if i is a triangle and j is one of its edges, $a_{ij}a_{ji} < 0$. However, if j is an edge in the mesh and i is a non-adjacent triangle which participates in the pressure stencil of j, then $a_{ji} \neq 0$ and $a_{ij} = 0$. This indicates that no scaling transformation is available which will make the system matrix skew symmetric in this case. Further, note that for any positive definite diagonal scaling matrix H, if we calculate the rate of change of $\mathbf{v}$ in the H-norm we have:

$$\frac{\mathrm{d}}{\mathrm{d}t}\tfrac{1}{2}\langle \mathbf{v}|\mathrm{H}|\mathbf{v}\rangle = \tfrac{1}{2}\left(\mathbf{v}^{\mathrm{T}}(\mathrm{HA}))^{\mathrm{T}}\mathbf{v} + \mathbf{v}^{\mathrm{T}}(\mathrm{HA}))\mathbf{v}\right) \tag{5.18}$$

$$= \langle \mathbf{v}|\,\mathrm{sym}(\mathrm{HA})|\mathbf{v}\rangle \tag{5.19}$$

Now, since HA cannot be skew symmetric there must exist state vectors $\mathbf{v}$ for which $\langle \mathbf{v}|\,\mathrm{sym}(\mathrm{HA})|\mathbf{v}\rangle$ is nonzero. If, as we have shown, there is no L^2 equivalent norm which is preserved by A, then we could still show stability by finding

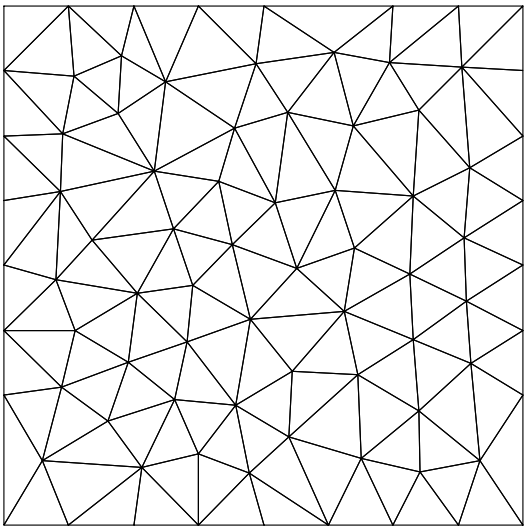

Figure 5.5: Irregular discretisation of a square used in the eigenvalue calculations. There are 133 triangles and 185 interior faces.

H such that $\langle \mathbf{v}| \operatorname{sym}(\mathrm{HA})|\mathbf{v}\rangle \leq 0$ for all $\mathbf{v}$. This is precisely the requirement that $\operatorname{sym}(\mathrm{HA})$ be negative semidefinite. However, once again the lack of sign skew symmetry in A combined with its zero diagonal pose a difficulty. It is easy to show that a necessary requirement for negative semidefiniteness of a symmetric matrix is that the element of largest modulus lies on the main diagonal. However, the main diagonal of A, and hence of $\operatorname{sym}(\mathrm{HA})$ is known to be zero while the absence of sign skew-symmetry in A proves that $\operatorname{sym}(\mathrm{HA})$ has non-zero entries in non-diagonal positions. This proves that $\operatorname{sym}(\mathrm{HA})$ does not have purely negative eigenvalues. Since $\operatorname{sym}(\mathrm{HA})$ is symmetric, it has real eigenvalues so we have in fact proven the existence of positive, real eigenvalues.

The result of this analysis is that this centroid based scheme, and indeed any similar scheme in which the pressure stencil is expanded to cover more surface elevation points, is neither conservative nor uniformly dissipative in any L^2 norm. This therefore leaves open the possibility that the scheme will exhibit growing modes which will manifest for suitable initial conditions and suitably long time integrations.

5.5.5 Eigenvalues

As an illustrative example of the symmetry characteristics of this method, the system matrix for each scheme was calculated on the small but irregular mesh shown in figure 5.5. A uniform unit depth was specified and g was set to 9.8. The domain side length is 2 units.

Figure 5.6 shows the eigenvalues of the semidiscrete matrix for both the or-

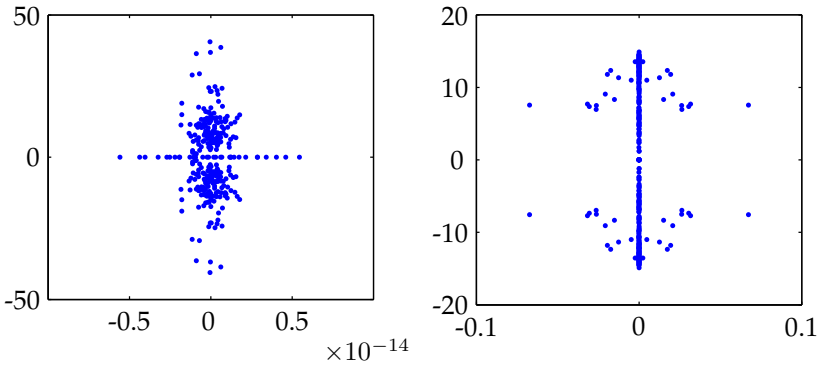

Figure 5.6: Eigenvalues of the discrete systems for the circumcentre based scheme (left) and the centroid based scheme (right). Note that the axis limits in both the real (horizontal) and imaginary (vertical) directions differ. In particular, the eigenvalues of the circumcentre scheme are zero to machine precision.

thogonal circumcentre based and the non-orthogonal centroid based schemes. It is apparent that the eigenvalues of the orthogonal scheme are, up to rounding error, purely imaginary as was predicted by the derivations above. Similarly, the growing modes, the possibility of whose existence in the non-orthogonal case was predicted above are evident in the eigenvalues of that scheme with a small positive real part.

5.6 Numerical tests

5.6.1 Truncation error

As an initial test of the convergence behaviour of the path integral pressure gradient formulation adopted here, we examine the gradient of a smooth function in a circular domain. In polar coordinates, the function chosen was:

$$\eta(r,\theta) = \cos\left(\frac{r}{R}\pi\right) \tag{5.20}$$

so that:

$$\frac{\partial \eta}{\partial r} = -\frac{\pi}{R}\sin\left(\frac{r}{R}\pi\right) \tag{5.21}$$

where R, the radius of the domain. The domain was then discretised with successively finer meshes examples of which are presented in figure 5.7. For the purpose of comparison, the same tests were conducted on a hexagonal domain discretised with equilateral triangles. To eliminate errors due to the boundary conditions, only water columns with centres lying within $R/2$ of the origin were included in the calculation.

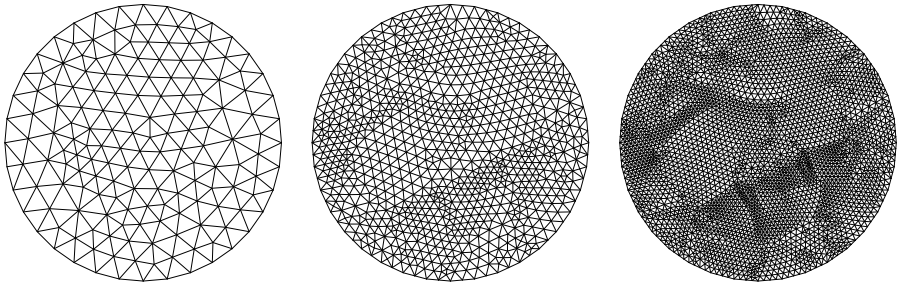

Figure 5.7: Successively refined triangulations of a circle. From the left, the values of $\Delta x/R$ are 0.184, 0.085 and 0.037.

The length scale relevant to the discrete pressure operator is the distance between adjacent water column centres. A useful measure of mesh refinement is therefore the mean water column separation:

$$\Delta x = \frac{\sum_{j\in E} \|\mathbf{x}_{j(i,1)} - \mathbf{x}_{j(i,2)}\|}{|V|} \tag{5.22}$$

$\Delta x/R$ is therefore the number of surface elevation points per wavelength. The error in the pressure gradient can be quantified by the area weighted average of the error at each column face:

$$e = \frac{\sum_{j\in E} \hat{A}_j \left\| \nabla_{xy}\eta|_{\mathbf{x}_j} \cdot [\mathbf{N}_j, \mathbf{T}_j] - [H_j, I_j] \right\|}{\sum_{j\in E} \hat{A}_j} \tag{5.23}$$

where $\hat{A}_j = \left(A_{j(i,1)} + A_{j(i,2)}\right)$ is the surface area of the columns adjacent to column face j.

Figure 5.8 shows the scaling of the error in the pressure gradient. The least squares fit to the curve indicates that the error scales as $\Delta x^{1.96}$ on the regular meshes but $\Delta x^{1.53}$ on the irregular meshes. Recall that for a regular mesh, the path integral formulation reduces to equation (5.1) which is a one dimensional central difference scheme. Such schemes are expected to be second order accurate and this is reflected in the result observed. On an irregular mesh, the pressure gradient discretisation is not completely centred in space. Non-centred difference schemes are typically first order in space, however the additional truncation error vanishes as the mesh approaches regularity. The successively refined meshes shown in figure 5.7 exhibit regions of quasi-regular mesh joined by less regular elements. As the mesh is refined, the proportion of the elements which make up the quasi-regular patches increases and hence the mesh quality improves. The $\mathcal{O}(1.53)$ convergence observed may therefore be attributable to the increase in mesh quality as the mesh is refined.

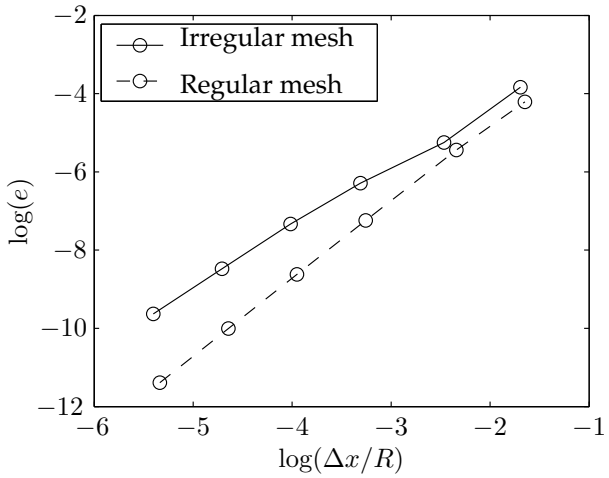

Figure 5.8: Scaling of the error in the pressure gradient operator. The least squares fit to the data indicates that the operator is $\mathcal{O}\left(\Delta x^{1.53}\right)$ on the irregular mesh and $\mathcal{O}\left(\Delta x^{1.96}\right)$ on the regular mesh.

5.6.2 Wave calculations

We now move on to test the long wave solutions of the pressure gradient operator by simulating a standing wave in a long, shallow basin. The basin used is 100km long and 10km wide with a mean surface elevation of 10.197m. This last value was chosen to produce a long wave celerity ($\sqrt{gh}$) of 10m/s. Wind drag and bed friction and viscosity are disregarded. In the absence of viscous terms, the vertical structure of the flow is uniform so we chose to conduct the simulation in only one layer. To establish an essentially linear wave, a sinusoidal initial surface displacement was imposed with an amplitude of 5mm and a wavelength of 200km. In every case the simulation was run for 200 000s which is 10 periods of the longest mode of the basin.

Two different meshes were generated. In each case, the nominal side length provided to the mesh generator was 1km however in one case the mesh is of high quality with 78% of triangles having an aspect ratio of less than 1.5 and 93% less than 2. The other mesh is of lower quality with only 49% having an aspect ratio of less than 1.5 and 84% less than 2. Furthermore it is obvious from a visual inspection of figure 5.9 that the high quality mesh is much closer to a regular mesh in large regions of the domain than is the low quality mesh.

Several comparisons may be made using this test case. In the first instance, the same simulation was run using each mesh. The time step was set to 25s resulting in a mean wave Courant number, $\mathrm{Cw}_{\mathrm{mean}}$, of 0.56 and a maximum wave Courant number, $\mathrm{Cw}_{\mathrm{max}}$, of 0.88 on the low quality mesh and a $\mathrm{Cw}_{\mathrm{mean}}$ of 0.50 and $\mathrm{Cw}_{\mathrm{max}}$ of 0.87 on the high quality mesh. That is to say, on each mesh the wave Courant number is less than one everywhere in each mesh. The

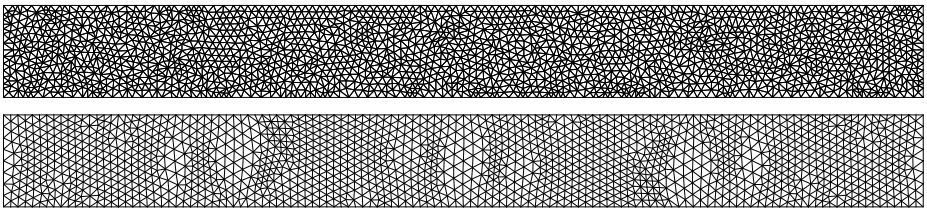

Figure 5.9: Two meshes of a rectangular basin. The upper mesh is of poor quality with many small angles while the lower mesh is of much higher quality.

θ scheme was used in time with $\theta = 0.5$ to eliminate any damping effect of the time integration scheme. By 70 000s (3.5 periods of the oscillation) spurious surface elevations were visible in the low quality mesh at mesh triangle scale and by the end of the simulation these had grown to amplitudes of up to 2m. The alternating high-low elevations characteristic of a checkerboard instability were exhibited at several locations. No spurious elevations were exhibited by the simulation on the high quality mesh. This phenomenon appears to be consistent with the expected behaviour on poor quality meshes detailed in section 5.5.1. The calculation on the poor quality mesh was rerun with $\theta = 0.55$ and the spurious modes were absent indicating that this instability is weak and may easily be controlled with a very mildly dissipative scheme.

The response of the schemes to increasing time step was investigated for both the θ and Hansen schemes. Once again θ was set to 0.5 and successively increasing time steps were employed. The Hansen scheme, being explicit, is subject to the CFL criterion (Courant et al., 1928) and, indeed, was stable and non-dissipative with a time step of 25s ($\mathrm{Cw}_{\mathrm{mean}} = 0.50$, $\mathrm{Cw}_{\mathrm{max}} = 0.88$) but with a time step of 30s ($\mathrm{Cw}_{\mathrm{mean}} = 0.60$, $\mathrm{Cw}_{\mathrm{max}} = 1.06$) the scheme was immediately and catastrophically unstable. The θ scheme is unconditionally stable for linear problems and was tested for mean wave Courant numbers up to 10. No loss of amplitude was observable for any Courant number however a slight loss of phase velocity was observable as the time step increased. Figure 5.10 illustrates this phase lag. Notable features of this graph are that even at a wave Courant number of 10 the phase lag less than 4 parts per thousand and that the Hansen scheme exhibits the same phase lag as the θ scheme for the time step for which the former scheme is stable.

5.6.3 Long term wave calculations

To further test the circumstances in which the checkerboard modes observed previously are exhibited, we look to a larger scale wave test case which is run over a longer period. The domain chosen was 420km by 100km and a uniform depth of 5m was specified. The simulation was started from rest with an initial surface elevation field:

$$\eta(x, y) = 0.05 \cos(\pi x / L) \tag{5.24}$$

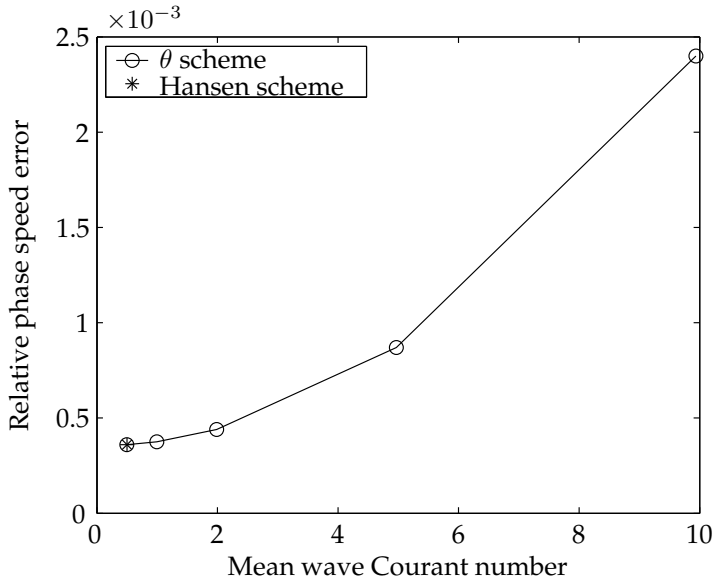

Figure 5.10: The phase speed error in the oscillating basin for various Courant numbers.

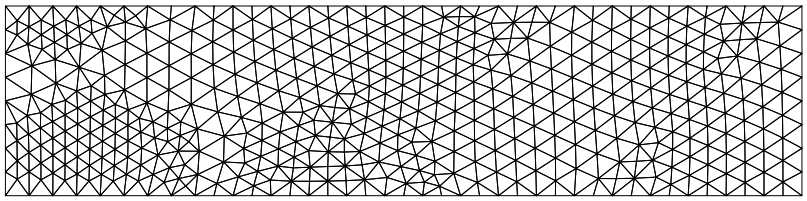

Figure 5.11: The mesh used for the standing wave test case. The channel length is 420km and the nominal triangle edge length is 10km

where $L = 420\text{km}$ is the length of the domain. This configuration corresponds to a standing wave of wavelength $2L$ and period 60000s. The Crank-Nicholson time discretisation (that is, $\theta = 0.5$) was applied with a time step of 1000 seconds. A triangular mesh with a typical edge length of 10km was generated using an algorithm by Legrand et al. (2000).

Since this experiment is concerned with the stability of the scheme, the metric which is used to evaluate the performance of the orthogonal and non-orthogonal schemes is the energy norm $\langle \mathbf{v}|\mathrm{D}|\mathbf{v}\rangle$ introduced in section 7.3. The results are shown in figure 5.12. It is apparent from the figure that the centroid scheme does not conserve this norm and eventually becomes unstable. In fact, spurious surface elevations are visible at 170 days simulation time and take the form shown in figure 5.13. In the case of the circumcentre scheme, however, no change in the energy norm is observed in the graph and, in fact, no the energy norm was found to be conserved to all 6 significant figures output. However, once again, when θ was increased to 0.55, the instability was not observed although the energy in the system slowly dissipated. From this we may conclude that the non-orthogonal centroid based discretisation proposed here is subject to instabilities related to the structure of the stencil but that these instabilities are weak and appear to be suppressed by the presence of even a small amount of dissipation in the scheme.

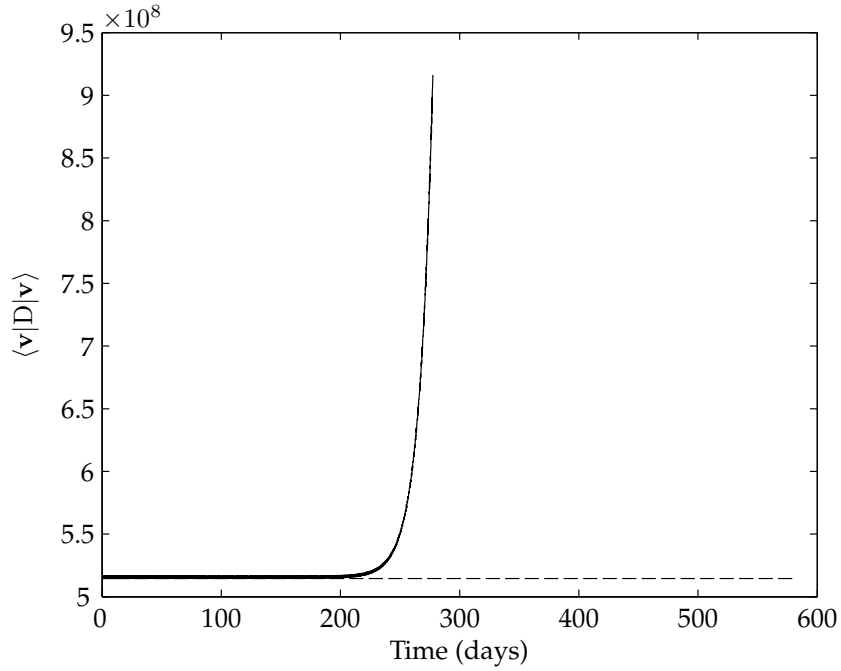

Figure 5.12: Energy norm against time for the standing wave test case. The solid line gives the result for the centroid scheme while the broken line is the result for the circumcentre scheme.

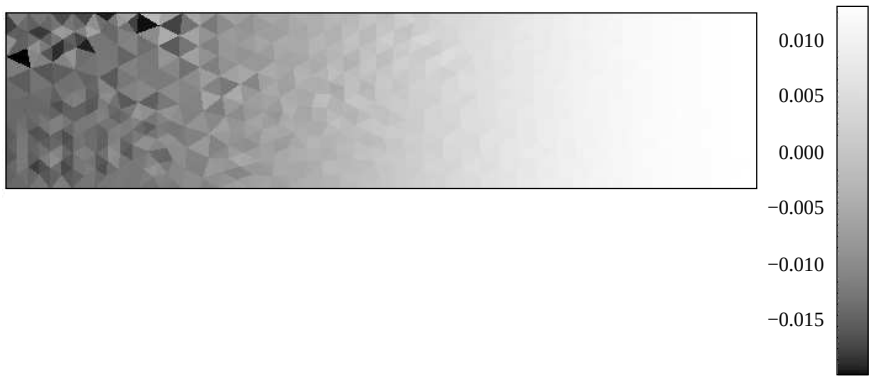

Figure 5.13: Surface elevation for the non-orthogonal centroid-based scheme after 207 days. Growing instabilities are visible at the Western end of the channel.

Chapter 6

A semi-Lagrangian advection scheme

6.1 Introduction

In chapter 4, an semi-implicit scheme was adopted for the discretisation of the pressure gradient term on the basis that an explicit scheme would make the stability of long wave solutions dependent on the maximum wave Courant number, $\mathrm{Cw}_{\max}$. A similar problem presents itself for the discretisation of the advection term. The flow characteristics which determine the stability of the advection operator are typically much slower than the long waves which are determinative for the pressure gradient operator. However, the flow Courant number $|\mathbf{u}|/\Delta x$ is still dependent on the mesh size. An explicit Eulerian scheme would therefore exhibit a time step dependence on the smallest cells in the mesh. On the other hand an implicit scheme would create an implicit coupling between the velocity vectors on different water column faces. This in turn would destroy the special structure of the system matrix presented in section 4.4 resulting in a much larger implicit system to be solved at each time step.

As an alternative, Casulli and Walters (2000) propose the use of a semi-Lagrangian advection scheme. Such a scheme is explicit but may be shown to be stable at any flow Courant number. While such stability at high flow Courant numbers may not hold for general non-linear problems, the key criteria of being able to exceed the Courant limit in particular regions is met.

The concept behind a semi-Lagrangian scheme is that the material derivative of the velocity may be discretised by tracing the trajectory of a fluid parcel back through one time step to calculate where the fluid that will arrive at a given velocity node at the end of a time step has come from.

This discretisation may be written as follows:

$$\frac{\mathrm{D}\mathbf{u}(\mathbf{x})}{\mathrm{D}t} \approx \frac{\mathbf{u}^{n+1}(\mathbf{x}) - \mathbf{u}^{n}(\mathbf{x}^{*})}{\Delta t} \tag{6.1}$$

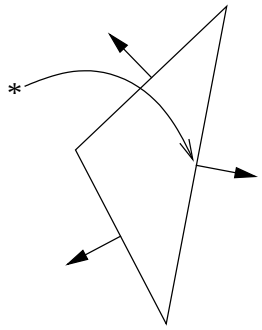

Figure 6.1: Under a semi-Lagrangian formulation, the advected velocity is interpolated at a point $\mathbf{x}^*$ found by following the flow backwards through the time interval Δt.

where $\mathbf{x}^*$ is position of the end of a trajectory tracked back through the time interval Δt from $\mathbf{x}$. Figure 6.1 illustrates this process. A semi-Lagrangian discretisation is therefore a two step process: first the trajectory ending at the point of interest must be traced backwards in space. The value of the velocity at that point must then be interpolated from the discretised velocity field solved for at the preceding time step. The focus of the new work presented here is on a method for constructing the trajectories necessary for the first of these steps rather than on the interpolation of the advected velocity. The importance of the quality of trajectory tracking has been recognised elsewhere as significant in the results of semi-Lagrangian advection algorithms Oliveira and Baptista (1998).

In the notation of equation (4.6) and neglecting the Coriolis term, the semi-Lagrangian advection scheme produces the following formulation:

$$F(u)^n_{j,k} = \mathbf{u}^n(\mathbf{x}^*) \cdot \mathbf{N}_j \tag{6.2}$$

6.2 Trajectories

The trajectories of a velocity field are obtained by solving the ordinary differential equation:

$$\frac{\mathbf{dx}}{\mathbf{d}t} = \mathbf{u}(\mathbf{x}) \tag{6.3}$$

Since a flow model does not generally result in an analytic solution at every point in space, but rather produces a discretised velocity field, it is necessary to reconstruct some velocity field which may then be integrated to discover the trajectories of the flow.

It is an obvious requirement that the continuous velocity field used to calculate the trajectories be consistent with the discretised velocity field produced by the flow model. Since the flow model is intended to model physical behaviour it is desirable that the trajectories produced should exhibit physically

realistic behaviour. In particular, we shall require that the trajectories produced not cross each other or a closed boundary (the no crossing condition).

6.3 Integration techniques

The conventional way to integrate the velocity field would be using a numerical ODE solver (see, for example, Oliveira and Baptista (1998)). However, due to the errors inherent in numerical integration techniques, it is difficult or impossible to guarantee the no crossing condition. At a minimum, very small time steps are required when calculating trajectories near closed boundaries. Instead, we generalise the approach first developed in Dunsbergen (1994) to our unstructured mesh. The basis for this approach is the construction of a continuous velocity field which is then integrated analytically to produce an analytic expression for the trajectory starting at a given point. Since the integration is analytic, no errors (up to machine precision) are introduced by the integration process.

The analytic integration approach imposes another constraint on the continuous velocity field: it must be analytically integrable at a reasonable computational cost. The obvious candidate is a cell-wise linear field. Assume $\mathbf{x}$ lies in cell i then:

$$\mathbf{u}(\mathbf{x}) = \mathrm{A}_i\mathbf{x} + b_i \tag{6.4}$$

Where A_i is a constant matrix and $\mathbf{b}_i$ is a constant vector. The integration problem reduces in this case to the solution of a three dimensional system of linear ordinary differential equations. Dunsbergen (1994) constructed his scheme in the transformed space in which his curvilinear rectangular mesh became strictly rectangular. In combination with the simplifying approximation that within each cell the u component of the velocity does not vary in y and v does not vary with x this allows for the decomposition of the resulting system into three linear scalar ordinary differential equations. The cost is that the velocity field used is distorted by the change of coordinates and made less accurate by the simplifying approximations. This necessitated the development in Dunsbergen (1994) of a complex velocity correction scheme.

On a triangular mesh, there is no continuous coordinate transformation which produces face normals which are mutually orthogonal. The refuge of decoupled velocity fields is therefore unavailable to us, at least in the horizontal directions. This results in a somewhat more complex set of differential equations and, in particular, requires the use of numerical techniques to solve for cell face crossings. However, the velocity field used need not be distorted by a change of coordinates.

To determine the unknown matrix and vector in equation (6.4), we first impose the constraints given by the discretised flow field. For each face j of cell i, if $\mathbf{n}_j$ is the normal to that face and u_j is the normal velocity component then:

$$(\mathrm{A}_i\mathbf{x}_j + b_i) \cdot \mathbf{n}_j = u_j \tag{6.5}$$

This is a single linear scalar constraint which therefore costs one degree of freedom per face. A linear vector field in three dimensions has 12 degrees of freedom so this constraint does not fully determine the velocity field. Similarly, if we examine the problem restricted to two dimensions, this constraint fixes 3 of the 6 available degrees of freedom. In both two and three dimensions this leaves equation (6.4) underdetermined. It will therefore be necessary to find further constraints in order to uniquely specify a velocity field.

6.4 Crossing Trajectories

In addition to failing to produce a well posed velocity field problem, the constraints so far imposed fail to guarantee that trajectories will not meet. In other words, the existence and uniqueness of solutions to (6.3) is not guaranteed at cell boundaries.

A two dimensional example which illustrates this problem may be constructed by considering the case where the flux through an interior face is zero. If the flux through the other faces is not zero then solutions exist for which the normal velocity component to the zero flux face is not uniformly zero. This raises the possibility of the mirror image case presented in figure 6.2 in which trajectories meet. Not only does this pose severe conservation problems for a semi-Lagrangian scheme, it makes the tracking of trajectories generally impossible since they may be followed into dead end situations. However, by specifying the normal component of velocity at every point on the cell face, this difficulty may be avoided. Clearly this requirement will also ensure that the inviolability of closed boundaries is preserved.

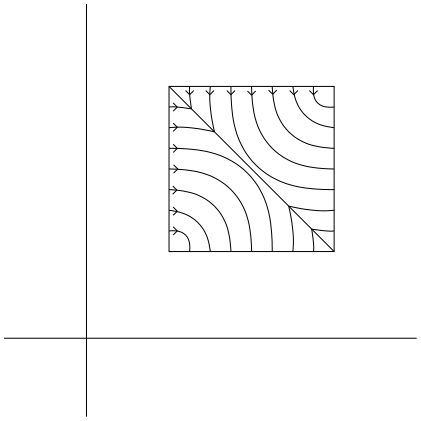

Figure 6.2: Sketch of trajectories in two mirror image cells illustrating the possibility of crossing trajectories. The flux through the diagonal cell face is 0.

The most obvious manner in which this new constraint may be applied is by requiring that the face normal velocity component be everywhere equal to the average flux through the face (the flux constraint):

$$\frac{\partial u_i}{\partial z} = 0 \qquad \frac{\partial v_i}{\partial z} = 0 \tag{6.6}$$

$$\frac{\partial w_i}{\partial x} = 0 \qquad \frac{\partial w_i}{\partial y} = 0 \tag{6.7}$$

where u_i, v_i and w_i are the components of $\mathbf{u}_i$. In addition, if $\mathbf{t}_j$ is the horizontal tangent vector to side face j of cell i then the imposition of constant normal

velocity along the line $\mathbf{x}_y + k\mathbf{t}_j$ amounts to a further constraint on each cell side. In the two dimensional case this last constraint defines the remaining 3 degrees of freedom while in the three dimensional case, these 3 and the 4 above in addition to the 5 imposed by (6.5) provide the twelve required constraints. In addition, constraints (6.6) and (6.7) make the matrix A_i block diagonal. This reduces the integration problem to a two dimensional linear first order ODE and a one dimensional linear first order ODE.

6.5 The continuity equation

Given that the discretised flow field is mass conservative, it is known that the normal flux integrated over the surface of one three dimensional cell is zero. In the two dimensional case, this is true if the flow is steady. It will be shown that the velocity field constructed above satisfies the continuity equation in the weak sense everywhere. We start with a simple proof concerning the situation within each cell.

Proposition 6.1 *If a linear velocity field satisfies the continuity equation integrated over some region of non-zero volume then the velocity field satisfies the continuity equation at every point.*

Proof: Let $\mathbf{u} = \mathrm{A}\mathbf{x} + \mathbf{b}$ be a velocity field such that on some region V with $\mathrm{vol}(V) > 0$:

$$\iint_{\Gamma V} \mathbf{u} \cdot \mathbf{n} d\mathrm{A} = 0 \tag{6.8}$$

Observe that:

$$\forall \mathbf{x} \in \mathbb{R}^3, \nabla \cdot \mathbf{u} = \mathrm{trace}(\mathrm{A}) \tag{6.9}$$

which is constant. Then by Gauss' theorem:

$$\begin{aligned} \iint_{\Gamma V} \mathbf{u} \cdot \mathbf{n} d\mathrm{A} &= \iiint_V \nabla \cdot \mathbf{u} dV \\ &= \iiint_V \mathrm{trace}(\mathrm{A}) dV \\ &= \mathrm{vol}(V)\mathrm{trace}(\mathrm{A}) \end{aligned}$$

Hence from (6.8) and (6.9) we conclude that $\forall \mathbf{x} \in \mathbb{R}^3, \nabla \cdot \mathbf{u} = 0$ □

This result applies in the two dimensional case with the obvious modifications. Having proved that the continuity equation is satisfied at every point interior to a cell, we now wish to prove continuity everywhere. However, the velocity field we have constructed is not smooth (or even continuous) at cell boundaries so its divergence is undefined at those points. The strongest result we can therefore hope to prove is that continuity is satisfied when integrated over any region (possibly about such a point).

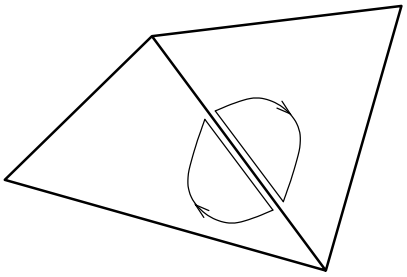

Figure 6.3: Decomposition of a loop integral so that each sub-loop intersects only one cell.

Proposition 6.2 *The integral over any non-trivial region of the divergence of a velocity field defined as specified in sections 6.3 and 6.4 is 0. In two dimensions the velocity field is also assumed to be steady.*

Proof: Let $\mathbf{u}(\mathbf{x})$ be a two dimensional velocity field satisfying the conditions above and let V be some region with $\mathrm{vol}(V) > 0$.

If V lies wholly within one cell, then $\mathbf{u}$ is linear throughout V and the result follows from proposition 6.1. If V intersects more than one cell then we decompose V into a number of sub-regions $V_1 \ldots V_n$ such that each sub-region intersects only one cell. Figure 6.3 illustrates this form of decomposition. Under Gauss' theorem we must show that

$$\iint_{\Gamma V} \mathbf{u} \cdot \mathbf{n} d\Gamma = 0 \tag{6.10}$$

We know from the previous case that this statement holds if we replace V with V_i for any i in $1 \ldots n$. Since the V_i are a decomposition of V, this is equivalent to requiring that the integral of the flux through the boundary shared between V_i and V_j is the same when evaluated on either side of that boundary. Since these boundaries follow the cell boundaries, this follows immediately from the flux constraint. □

6.6 Closure of streamlines

A further physically significant property of trajectories in steady two dimensional flow is that (except in unimportant degenerate cases) they form streamlines of the flow and describe closed loops. The usual proof of this result (see, for example, Batchelor (1967, pp75–77)) assumes that the velocity field is continuous and differentiable, however the results above will allow us to generalise it to this case.

Proposition 6.3 *The streamlines of a steady two dimensional velocity field defined as specified in sections 6.3 and 6.4 are closed.*

Proof: To prove that the streamlines are closed, it is sufficient to demonstrate the existence of a stream function. The streamlines are then the contours of that function and are therefore closed. If $\mathbf{u} := (u, v)$ is the velocity field then the stream function is the scalar potential of the field $\mathbf{F} := (-v, u)$. That is, the stream function is defined (up to a constant) by:

$$\Psi(\mathbf{x}) - \Psi_0 = \int_0^{\mathbf{x}} \mathbf{F} \cdot d\mathbf{r} \tag{6.11}$$

Clearly Ψ is only well defined if this integral is path independent. This in turn is equivalent to requiring that:

$$\oint_C \mathbf{F} \cdot d\mathbf{r} = 0 \tag{6.12}$$

for any loop C. However, by construction of $\mathbf{F}$:

$$\oint_C \mathbf{F} \cdot d\mathbf{r} = \oint_C \mathbf{u} \cdot \mathbf{n} ds \tag{6.13}$$

The right hand side of this equation is simply the two dimensional version of (6.10) so this result follows from proposition 6.2. □

6.7 Increased Accuracy

The assumption that the normal velocity component is constant over each cell face is first order in space. In the two dimensional case, we may improve this by specifying that the normal velocity may vary linearly along each cell side. The appropriate gradient for this variation may be obtained using finite differences on nearby velocity points. This constraint incurs the same cost in degrees of freedom as specifying a constant normal velocity along each face however the resulting velocity field is now exact for linear velocity fields rather than only for constant fields. As with the pressure gradient operator, the accuracy of which was demonstrated in section 5.6.1, this process is exactly centred in space and therefore second order accurate on meshes composed entirely of equilateral triangles. Consequently, the field generated will be second order accurate in the special case of a regular triangular mesh but the accuracy will decrease on lower quality meshes.

For three dimensional models based on tetrahedral discretisations, this approach can be directly generalised. In this case, the tangent space to each face is two-dimensional so the total flux and two flux gradients on each of the four triangular faces together fix the twelve degrees of freedom of the general linear three-dimensional velocity field. Solving equation (6.3), then, primarily consists of solving a three dimensional eigenproblem.

In a mesh made up of layers of triangular prisms, each cell has 5 faces so the linear system obtained by directly generalising the two dimensional case

will be overdetermined. It is therefore necessary to reduce the number of constraints applied to the velocity. In a shallow water model, the vertical velocity is typically much less important than the horizontal velocities. With this in mind, we may choose to forgo higher accuracy in the representation of this velocity. In other words, we continue to impose constraint (6.7). This results in the expression:

$$w_i = az + b \tag{6.14}$$

and hence:

$$z_i(t) = -b/a + (z(0) + b/a)\, e^{at} \tag{6.15}$$

Now, just as the mesh itself consists of layers of prisms, we may construct a two-dimensional field in the plane through the centre of each layer and then interpolate linearly between those layers. The horizontal velocity at a point between layers α and β is then given by:

$$\mathbf{u}_{xy}(t) = \frac{z - z_\alpha}{z_\beta - z_\alpha} \left(\mathrm{A}_\alpha \mathbf{x}_{xy} + \mathbf{b}_\alpha\right) + \frac{z - z_\beta}{z_\alpha - z_\beta} \left(\mathrm{A}_\beta \mathbf{x}_{xy} + \mathbf{b}_\beta\right) \tag{6.16}$$

This is not a linear expression in $\mathbf{x}$, however when (6.15) is substituted into (6.16) then the resulting ODE for position is linear, albeit with variable coefficients. The form of the equation is:

$$\mathbf{u}_{xy}(t) = \left(\mathrm{A}\mathbf{x}_{xy} + \mathbf{b}\right)\left(c + e^{dt}\right) \tag{6.17}$$

If $\{\mathbf{e}_1, \mathbf{e}_2\}$ are the generalised eigenvectors of A and $\{\lambda_1, \lambda_2\}$ are the corresponding eigenvalues, and $Z(t) = e^{ct} + cdt$ then a fundamental matrix for (6.17) is:

$$\mathrm{X}(t) = \left[\mathbf{e}_1 e^{Z(t)\lambda_1} | \mathbf{e}_2 e^{Z(t)\lambda_2}\right] \tag{6.18}$$

The scheme currently implemented in Delfin is a somewhat simplified version of this approach in which the face normal velocities are allowed to vary in the horizontal but not the vertical dimension.

6.8 Results

The streamline closure property proven in the preceding sections may be demonstrated by tracking streamlines in a simple rotating flow:

$$\mathbf{u}(\mathbf{x}) = \begin{bmatrix} -y \\ x \end{bmatrix}$$

The flow domain was 30m in diameter and a typical mesh side length of 1m was specified. 21 equally spaced trajectories at radiuses from 5m to 25m were tracked for a full revolution (60πs) using the two dimensional second order scheme described in section 6.7. Since the prescribed velocity field is linear, the scheme was exact up to machine precision with the relative error (connection error divided by trajectory length) being between 4×10^{-8} and 6×10^{-8} in each case. Figure 6.4 illustrates one of the streamlines generated.

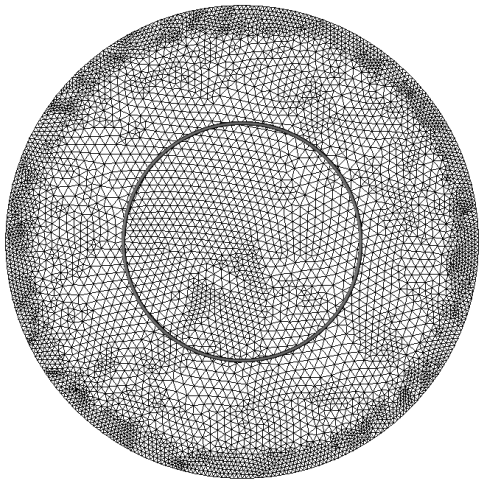

Figure 6.4: Tracking of a closed streamline in a rotating flow. The error in the connection of the loop is on the order of machine precision and is not visible at this scale.

6.9 Numerical issues

As everywhere in computational mathematics, the algorithm presented here is affected by the finite precision of floating point arithmetic and the limited convergence rates of iterative algorithms. In this case, the problematic areas are uncertainty about which cell a point lies in and uncertainty in the sign of the normal velocity when the flux through a face is small.

6.9.1 Finding the cell face

Solving the ODE at the heart of the trajectory tracking algorithm produces an expression for the position of a tracked point as a function of time. However, in cases where the flow Courant number is greater than one the trajectory will be tracked through more than one cell. Since the flow field, and hence the analytic solution for the trajectory, changes at each cell boundary, it is necessary to determine the point at which the trajectory leaves the cell in question. Determining where an expression in transcendental functions first crosses a face (i.e. a plane) is difficult and complex. We therefore resort to numerical methods for determining the point of intersection. In the first instance, Newton's method is used to find the cell exit point. Where Newton's method breaks down, bisection is used as a failsafe fall back. In either case, it is not possible to determine the exact point of crossing. Instead, the algorithm exits when a point lying on the curve is within some $\epsilon > 0$ of the line.

The uncertainty in the crossing point means that the starting point of the

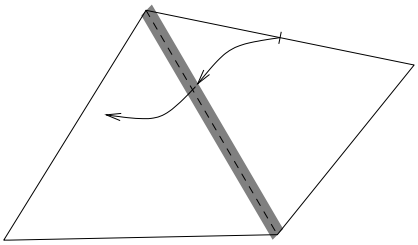

Figure 6.5: Difficulties associated with crossing a cell face. The trajectory through the right hand cell ends inside the grey tolerance are of the left hand cell. The first crossing of the broken cell face must be ignored for the trajectory to be successfully tracked.

trajectory in the next cell may not actually lie in that cell so it also necessary to apply the same tolerance when testing whether the point lies in the cell. In addition, where the starting point for a cell lies outside that cell, it is necessary to exclude the first crossing of that face from consideration. Figure 6.5 demonstrates this situation.

6.9.2 Sign uncertainty in the normal velocity

Another effect of limited precision arithmetic is that the value of the velocity field is only known to a finite precision. That is to say if $\mathbf{x}$ lies on the interface of two cells then there may be a small variation in the face normal velocity depending on which cell is used to calculate that velocity. Where the face normal velocity is very small in magnitude, this may result in the trajectory tracking algorithm breaking down. Where this occurs, the behaviour currently implemented is that the tracking of trajectories is abandoned and the interpolation of velocity occurs at the point of the breakdown. This typically occurs in a fraction of a percent or less of cases so the accuracy cost is minimal.

6.9.3 The airborne trajectory problem

The representation of the surface presented in figure 3.3 produces cell side faces which are partially or fully exposed. However, the velocities through these faces are not, in general, zero. This raises the possibility that a trajectory may be followed through such a face and out of the flow domain thereby causing a breakdown in the algorithm. These "airborne" trajectories are a consequence of the discontinuous representation of the free surface and are obviously spurious. This difficulty is overcome by mapping the water column on one side of the column side onto the other. However, we have just constructed a trajectory integration mechanism with the feature that the cell face normal velocity component at any point is the same when evaluated on either side of the face. If the trajectory is moved vertically every time it crosses a face, then the possibility

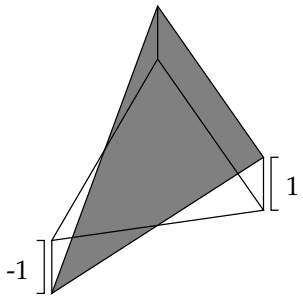

Figure 6.6: The non-conforming basis function used in the interpolation of velocity.

arises that the velocity calculated on the two sides of the face may differ in sign resulting once again in the breakdown of the algorithm. This subsequent difficulty may be minimised by adjusting only the top layers on each side. To avoid a badly conditioned problem in the case where the top layer on the lower side of a water column face is very thin, the adjustment applies down to the level of the second from top cell on the low side of each interface. In the tiny minority of cases in which this is still insufficient to prevent the no crossing condition being violated, the trajectory tracking is abandoned and, as in other breakdown situations, the interpolation happens at that point.

6.10 Interpolation of the velocity

Once the end of the trajectory has been found, the advected velocity $\mathbf{u}^n(\mathbf{x}^*)$ must be interpolated. Since the momentum equation (2.5) is only solved for horizontal the components of the velocity vector, it is not necessary to interpolate the value of the vertical velocity.

The interpolation of the velocity is itself a two stage process. First, the face tangent velocity at each of the vertical sides of the cell is interpolated. Next, the velocity at the point is interpolated using linear non-conforming elements in the horizontal and continuous linear elements in the vertical.

The subject of the interpolation of the cell face tangent velocity is the key question of chapter 7 so it need not be investigated here. It should simply be noted that the velocity is interpolated in two dimensions using the four other sides of the two triangular cells attached to the current face.

Using the full face centre velocity vectors so determined, the velocity at any position in two dimensional space may be calculated using the basis function shown in figure 6.6. To evaluate the velocity at a point $\mathbf{x}$ in $\mathbb{R}^3$, a linear interpolation is carried out between the velocity calculated in the plane of the cell centres above $\mathbf{x}$ and that below $\mathbf{x}$. The scheme so constructed is a linear upwinded scheme and is therefore first order accurate and diffusive.

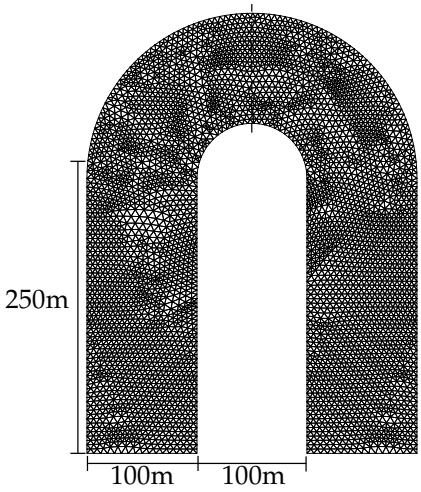

Figure 6.7: The mesh used in the river bend simulations. The slice used for measurements is indicated by the broken line.

6.11 Flow in a river bend

The flow in a tight river bend is chosen as a test of the advection term. In a rotating frame of reference, the inertia which the advection term represents produces an apparent centrifugal force which must be opposed by some other force if the flow is to be circular. In the case of a frictionless two dimensional flow field, the apparent centrifugal force is balanced by an increase in surface elevation towards the outside of the curve resulting in a pressure gradient force radially inwards. If $u(r)$ is the velocity component tangential to a circular rotating flow at radius r from the centre of rotation. Then the centripetal acceleration is:

$$f_c = -u^2/r \tag{6.19}$$

The balance of accelerations for which tests were made was therefore:

$$g(\nabla\eta \cdot \mathbf{r}) = u^2/r \tag{6.20}$$

Where $\mathbf{r}$ is the radial unit vector.

6.11.1 Two dimensional test

The model geometry is shown in figure 6.7. The nominal mesh side length is 5m. A constant inflow velocity of 0.5m/s is prescribed at the right hand arm of the U. At the left hand end, a constant surface elevation of 10m is prescribed to match the initial surface elevation throughout the interior of the domain. The remaining boundaries are closed. To induce convergence to the steady state solution as rapidly as possible, θ was set to 1. A time step of 10s was employed

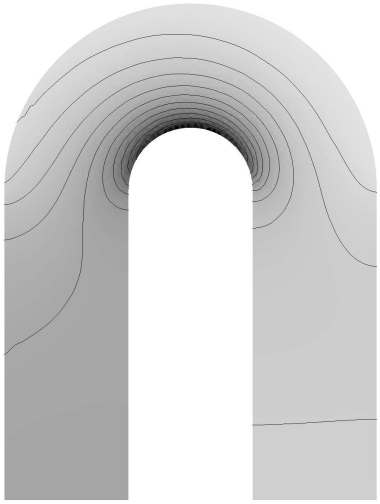

Figure 6.8: The surface elevation of the channel bend simulation after the steady state has been achieved. The maximum surface elevation is 10.0157m and the minimum is 9.9819m. There are 10 equally spaced contours.

and free slip boundaries were employed at all of the closed boundaries as well as at the bed. Convergence to the steady state was observed after approximately 200 time steps (33 minutes of simulation time) but the simulation was allowed to run for 1200 time steps before measurements were taken.

For comparison purposes, an identical simulation was conducted using a cut cell model (Kramer and Stelling, 2006) with a conservative advection scheme based on that presented in (Stelling and Duinmeijer, 2003). The cut cell approach employs a structured rectangular grid in the interior of the domain with partial cells at the boundaries to match the shape of the domain. In this simulation, the grid was made up of 5m squares.

Figure 6.8 shows the surface elevation of the Delfin simulation after 1200 time steps. The banking up of water at the outside of the bend is evident as is a certain amount of head loss over the length of the channel. The head loss is a consequence of the artificial dissipation inherent in a first order upwind scheme, which this is. The cross-stream surface gradient, on the other hand, is the predicted effect of the advection term. Figure 6.9 shows the equivalent result using the conservative, cut cell technique. Superficially, the results are similar, however it is noticeable that the head loss, and hence the asymmetry in the surface elevation contours, is less in the conservative scheme. The difference between the minimum and the maximum surface elevation in the bend is around 10% less in the Delfin simulation which is further evidence that more energy is lost than in the cut cell simulation.

To make a more quantitative statement it is necessary to note that the flow in

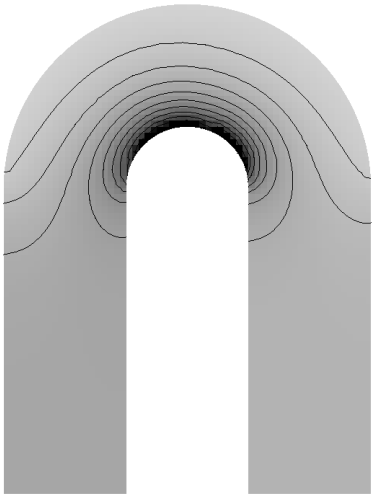

Figure 6.9: The surface elevation of the cut cell channel bend simulation in steady state. The maximum surface elevation is 10.0114m and the minimum is 9.9719m. There are 10 evenly spaced contours and the colour map is the same as for the previous image.

such a bend is not totally rotational since there must be some adjustment from the straight channel to the curve and back again. To minimise the impact of these effects, measurements were taken in a 10° wide slice around the broken line shown in 6.7. To exclude any effect of the modified pressure stencil near the boundaries, results within 10m of the edge of the domain were disregarded. The velocity and pressure gradient were extracted at each water column face and the ratio:

$$\frac{u^2/r}{g(\nabla\eta\cdot\mathbf{r})} \tag{6.21}$$

was calculated. The average value of this quantity over the slice chosen was 1.0015 which indicates that the apparent centrifugal acceleration is around 0.15% greater than that which is balanced by the pressure gradient. In other words, this amount of velocity is lost in the scheme.

6.11.2 Three dimensional test

If the vertical structure of the flow is taken into account and bed friction and vertical viscosity imposed, then the situation becomes more complex. The velocity increases from the bed to the surface, however the acceleration due to the pressure gradient is constant over the vertical. The result of this is that the centrifugal acceleration at the surface is greater than the acceleration due to the pressure gradient while near the bed, the pressure gradient term is the greater.

The result is the so called secondary flow, a circulation in the radial direction which is outwards at the surface and inwards near the bed. A realistic calculation of the secondary flow, of the sort carried out in Booij (2002) would require a more sophisticated vertical viscosity treatment than the constant value considered here, nonetheless, this is a useful test of the model's capability to conduct qualitatively correct simulations of three dimensional flow.

The simulation described in the previous section was conducted with 10 layers of height 0.1m. In accordance with the z-layer approach currently implemented in Delfin, the top layer varies in height as the free surface moves. All model parameters remained the same except that a vertical viscosity of $0.02\mathrm{m}^2/\mathrm{s}$ and a bottom drag coefficient of 0.0025 were imposed. Figure 6.10 shows the velocities in the slice through the domain indicated in figure 6.7. It is clear that the expected secondary circulation is present. More complex dynamics, such as an additional counter-rotating vortex at the outside of the bend as described in Booij (2002) would require a more sophisticated viscosity model and, possibly, a non-hydrostatic pressure term.

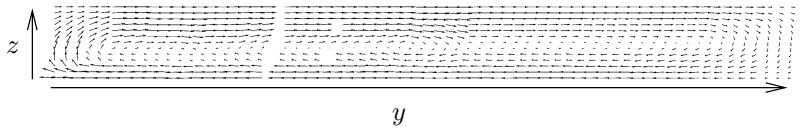

Figure 6.10: The cross stream component of the flow in the middle of a tight bend viewed in the streamwise direction. The velocity field is interpolated on a regular grid. The absence of some vectors indicates an absence of data and is a consequence of the unstructured mesh and the thin sample region.

Chapter 7

The Coriolis term

The distinctive feature of the C grid is that no full velocity vectors are solved for or stored. Instead, the prognostic velocity variables are the normal component of the velocity at each cell face. Where the full velocity vector is needed, such as in the calculation of the Coriolis acceleration, the component of velocity tangential to the cell face must be interpolated at that cell face. Any interpolation procedure has the potential to introduce accuracy and stability problems. Indeed Zhang and Baptista (2005) abandoned the C grid in favour of a C-D grid in which full velocities are solved for on the grounds that the C grid was introducing inaccuracies in the Coriolis term. While the C-D grid introduced by Adcroft et al. (1999) offers the advantage that velocity component interpolation becomes unnecessary, it has been shown that it suffers from spurious modes caused by the imbalance between the number of discrete momentum and continuity equations generated Le Roux (2001).

In preparing Ham et al. (2005) the authors encountered instabilities in Coriolis dominated simulations and eventually followed Zhang and Baptista (2005) in adopting a C-D grid. However, as stated in that paper, the C-D grid approach also exhibited noise after several months of simulation time. On the basis of this observation and the theoretical result by Le Roux (2001) above, we now return to a deeper analysis of the discretisation of the Coriolis term on unstructured C grid analogues.

Espelid et al. (2000) established that geometric irregularity may give rise to instabilities in a structured C-grid. In that paper, the authors were concerned with the stability of the C-grid in the presence of varying bathymetry. They studied the linear inertia-gravity wave equations and considered the eigenvalues of the propagation matrices of the semi-discrete (in space) partial differential equations. In the case of hyperbolic problems, these should only have imaginary eigenvalues. Skew-symmetric matrices or matrices similar to skew symmetric matrices have this property (see, for example, Strang, 1988). They established that an energy conserving C-grid discretisation is achieved when the interpolation weights of the Coriolis term are chosen in such a way that the

semidiscrete system matrix is similar to a skew-symmetric matrix. Perot (2000) also considered the problem of irregularity in C-grid discretisations. He was concerned with unstructured C-grid schemes for the two dimensional Navier-Stokes equations in an inertial (that is, non-rotating) frame of reference for domains of constant depth. He demonstrated that energy conservative discretisations were possible given appropriate choices of discrete operator.

Le Roux et al. (2005) demonstrated the value of applying linear algebra approaches, that is, approaches based on the properties of the discretised linear systems, to stability problems. By analysing the null spaces of the semi-discrete system of equations, those authors were able to demonstrate the existence of a spurious Coriolis mode associated with a C grid discretisation on a regular grid. In addition, Verstappen and Veldman (Veldman and Rinzema, 1992; Verstappen and Veldman, 1998, 2003) have demonstrated the importance of the spectral properties of the discrete convection and diffusion operators. Their analysis, rather than focusing on the null space, centred on unstable solutions of the equations. Their significant contribution was that the stability of a discretised system may depend on whether the symmetry properties of the original differential operators are preserved by the discretisation.

The symmetry preserving properties of the convection operator on a triangular C grid were investigated by Wenneker (2002, pp159-166). He showed that the operator was only symmetry preserving on structured grids. That is, grids formed of a regular tessellation of congruent triangles. In this chapter we apply the principle of symmetry analysis to unstructured triangular C grid discretisations of the Coriolis term. We will show that the most straightforward discretisations are subject to an instability quite distinct from the spurious mode discovered by Le Roux (2001). We will then derive a symmetry preserving interpolation scheme which does not suffer from this instability.

7.1 Reconstructing the tangent velocities in a cell

The tangent velocity at the centre of a cell face must be interpolated as the linear sum of the nearby face normal velocity components. We will approach this as a two stage process: for each cell, the tangential velocity at each adjacent cell face centre is reconstructed. This results in two values for the tangential velocity component of each cell face centre. The final interpolated velocity component is a linear combination of these two values. In this section we will focus on developing an appropriate interpolation scheme within each cell .

To simplify the analysis, we will examine the two dimensional linearised shallow water equations introduced in section 2.4. Since the evaluation of the Coriolis term requires only the reconstruction of the horizontal tangent velocity components, this two dimensional restriction of the problem retains many of the important features of the interpolation problem.

The stability of the scheme is an important consideration in developing a suitable interpolation. If we return to the linear algebra approach to stability

which was pursued in section 5.5.2 then the semidiscrete linear equations can be written in the form introduced as equation (4.19):

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\mathbf{u}\\ \eta\end{bmatrix} = \begin{bmatrix}\mathrm{F} & \mathrm{P}\\ \mathrm{C} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{u}\\ \eta\end{bmatrix} \tag{7.1}$$

As before, the submatrix P encodes the discrete pressure term while C contains the discrete continuity operator. The new addition is F, the submatrix pertaining to the Coriolis operator. Once again we wish to ensure that the Coriolis term is stable and non-dissipative. We should therefore like to ensure that the submatrix F is skew-symmetric under some suitable diagonal scaling matrix D. In particular this means that the velocity interpolation should be chosen such that F has a zero diagonal.

Linearly interpolating the tangent velocities at the centres of the faces of a triangle is equivalent to fitting a linear velocity field to that triangle. As we observed in section 6.3, such fields have six degrees of freedom in two dimensions and, if $\mathbf{u}(\mathbf{x})$ is the interpolating velocity field, the basic consistency requirement that for each side j of triangle i:

$$\mathbf{u}(\mathbf{x}_j)\cdot\mathbf{n}_j = u_j \tag{7.2}$$

consumes three of these degrees of freedom. It is easy to show that all possibilities for such a consistent reconstruction of the full velocity vector are given by taking the approximation of the velocity field by a polynomial from the lowest order Raviart-Thomas space and evaluating it at some point in the triangle. This approximation of the vector field has the property that along any line the vector component normal to that line does not change. In any vertex of a triangle the velocity vector is constructed out of the two adjacent face normal components only. Along an altitude line (the line between a vertex and its opposite face that intersects that face orthogonally) the projection of the approximated vector field in the tangential direction of the face is constant. It follows that there is a unique consistent tangential velocity reconstruction with no contribution from the normal velocity of the face itself. That is to say, that there is a family of interpolations which correspond to the points along the altitude to face j which results in an interpolation of the velocity component in the direction tangent to face j which is both constant and retains the zero diagonal of the matrix F. It is therefore obvious that this reconstruction can be evaluated by projecting the vector field approximation at any point along this altitude line in the face tangent direction.

One such velocity reconstruction in a triangle is given by Perot (2000, equation 96). It can be compactly expressed using the position of the circumcentre. Let $\Delta x_{i,j_1}$ be the orthogonal distance of the circumcentre to face j_1, where j_1, j_2 and j_3 are the faces of the triangular cell i. Perot's velocity reconstruction is given by:

$$A_i\mathbf{u} = u_{j_1}\, l_{j_1}\Delta x_{i,j_1}\mathbf{n}_{j_1} + u_{j_2}\, l_{j_2}\Delta x_{i,j_2}\mathbf{n}_{j_2} + u_{j_3}\, l_{j_3}\Delta x_{i,j_3}\mathbf{n}_{j_3}, \tag{7.3}$$

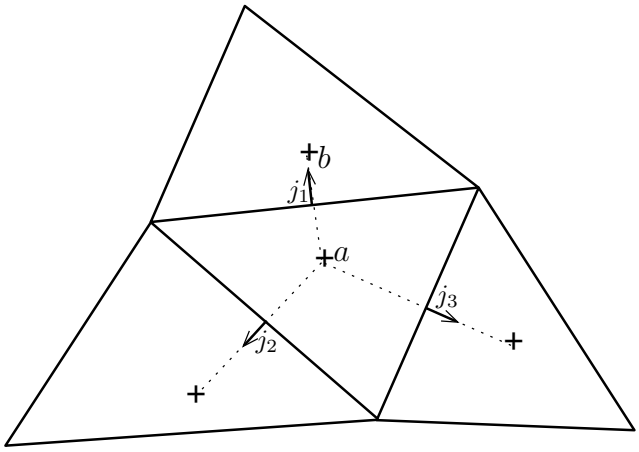

Figure 7.1: The mesh labelling used in the discretisation of the Coriolis term.

where l_{j_1} is the length of face j_1, and A_i is the area of the triangle. The fact that this reconstruction fulfils the consistency condition follows from the following geometrical identity for arbitrary triangles:

$$l_{j_1}\Delta x_{i,j_1}\mathbf{n}_{i,j_1}\mathbf{n}_{i,j_1}^{\mathrm{T}} + l_{j_2}\Delta x_{i,j_2}\mathbf{n}_{i,j_2}\mathbf{n}_{i,j_2}^{\mathrm{T}} + l_{j_3}\Delta x_{i,j_3}\mathbf{n}_{i,j_3}\mathbf{n}_{i,j_3}^{\mathrm{T}} = A_i\mathbf{I}. \tag{7.4}$$

If $\mathbf{n}_{i,j_1}$ is the outward pointing normal to triangle i on face j_1 then let $\mathbf{t}_{i,j_1}$ be the face tangent vector in anticlockwise direction along the triangle boundary. Thus the reconstructed tangential velocity is given by

$$v_{j_1} = \frac{\Delta x_{i,j_2} l_{j_2}}{A_i}\, u_{j_2}\mathbf{n}_{i,j_2}\cdot\mathbf{t}_{i,j_1} + \frac{\Delta x_{i,j_3} l_{j_3}}{A_i}\, u_{j_3}\mathbf{n}_{i,j_3}\cdot\mathbf{t}_{i,j_1} \tag{7.5}$$

It is directly clear that the face normal component u_{j_1} does not contribute. Thus we have calculated the unique consistent reconstruction of the tangential velocity out of the other two faces only. Alternatively we could first reconstruct the full vector out of the two other faces for each face individually and then project in each respective tangential direction. Using expression (7.3) however we only need one full vector reconstruction per triangle. Moreover the fact that it is expressed in distances between the circumcentre and the faces will be convenient in our later analysis.

7.2 Discretising Coriolis

Having developed a consistent velocity reconstruction within one triangle which will not contribute to the main diagonal, we proceed to discretise the Coriolis term on the circumcentre mesh. Consider the mesh fragment displayed in figure 7.1. If we write v_{a,j_1} for the tangent velocity to face j_1 reconstructed in

triangle a and v_{b,j_1} for the reconstruction from triangle b then we may construct a linear combination of these:

$$v_{j_1} = \gamma_{a,j_1} v_{a,j_1} + \gamma_{b,j_1} v_{b,j_1} \tag{7.6}$$

This is the tangent velocity used to calculate the Coriolis term $-fv_{j_1}$ at face j_1.

Recall that the Coriolis terms occur only in the matrix F. In section 5.5.3 we established that the use of an orthogonal mesh and the pressure gradient formulation given in equation (5.1) results in a semidiscrete system which is skew symmetric under scaling by the diagonal matrix D, the entries of which are derived in section 5.5.3. It was further established in section 5.5.4 that no similar scaling exists for the scheme for non-orthogonal meshes with the path integral pressure gradient formulation given by (5.2) or, indeed, with any other scheme based on an enlarged pressure stencil. Accordingly, to devise a pressure gradient formulation which is stable and non-dissipative, we shall attempt to find a discretisation which preserves the signed skew symmetry of the orthogonal mesh based scheme.

Since we have chosen velocity discretisations for v_{j_1} which contain no contribution from u_{j_1}, the zero diagonal required for scaled skew symmetry has been maintained. By substituting (7.5) into (7.6), the coefficients of v_{a,j_1}, v_{a,j_2} and v_{a,j_3} can be written as:

$$\begin{bmatrix} 0 & \gamma_{a,j_1}\frac{\Delta x_{a,j_2} l_{j_2}}{A_a}\mathbf{n}_{i,j_2}\cdot\mathbf{t}_{i,j_1} & \gamma_{a,j_1}\frac{\Delta x_{a,j_3} l_{j_3}}{A_a}\mathbf{n}_{i,j_3}\cdot\mathbf{t}_{i,j_1} \\ \gamma_{a,j_2}\frac{\Delta x_{a,j_1} l_{j_1}}{A_a}\mathbf{n}_{i,j_1}\cdot\mathbf{t}_{i,j_2} & 0 & \gamma_{a,j_2}\frac{\Delta x_{a,j_3} l_{j_3}}{A_a}\mathbf{n}_{i,j_3}\cdot\mathbf{t}_{i,j_2} \\ \gamma_{a,j_3}\frac{\Delta x_{a,j_1} l_{j_1}}{A_a}\mathbf{n}_{i,j_1}\cdot\mathbf{t}_{i,j_3} & \gamma_{a,j_3}\frac{\Delta x_{a,j_2} l_{j_2}}{A_a}\mathbf{n}_{i,j_2}\cdot\mathbf{t}_{i,j_3} & 0 \end{bmatrix}$$

When multiplied by the Coriolis parameter f, this is a (discontinuous) submatrix of F.

Note that the scaling matrix D is fully determined by the requirement that the system be skew symmetric in an inertial frame of reference. This allows us to write the following relationships between opposing elements of F:

$$\Delta x_{j_1} d_{j_1} l_{j_1} \gamma_{a,j_1} \frac{\Delta x_{a,j_2} l_{j_2}}{A_a}\mathbf{n}_{i,j_2}\cdot\mathbf{t}_{i,j_1} = -\Delta x_{j_2} d_{j_2} l_{j_2} \gamma_{a,j_2} \frac{\Delta x_{a,j_1} l_{j_1}}{A_a}\mathbf{n}_{i,j_1}\cdot\mathbf{t}_{i,j_2} \tag{7.7}$$

$$\Delta x_{j_1} d_{j_1} l_{j_1} \gamma_{a,j_1} \frac{\Delta x_{a,j_3} l_{j_3}}{A_a}\mathbf{n}_{i,j_3}\cdot\mathbf{t}_{i,j_1} = -\Delta x_{j_3} d_{j_3} l_{j_3} \gamma_{a,j_3} \frac{\Delta x_{a,j_1} l_{j_1}}{A_a}\mathbf{n}_{i,j_1}\cdot\mathbf{t}_{i,j_3} \tag{7.8}$$

$$\Delta x_{j_3} d_{j_3} l_{j_3} \gamma_{a,j_3} \frac{\Delta x_{a,j_2} l_{j_2}}{A_a}\mathbf{n}_{i,j_2}\cdot\mathbf{t}_{i,j_3} = -\Delta x_{j_2} d_{j_2} l_{j_2} \gamma_{a,j_2} \frac{\Delta x_{a,j_3} l_{j_3}}{A_a}\mathbf{n}_{i,j_3}\cdot\mathbf{t}_{i,j_2} \tag{7.9}$$

Noting that $\mathbf{n}_{i,j_2}\cdot\mathbf{t}_{i,j_1} = -\mathbf{n}_{i,j_1}\cdot\mathbf{t}_{i,j_2}$ and solving for the weights γ, we find that:

$$\gamma_{a,j_1} = c\frac{\Delta x_{a,j_1}}{\Delta x_{j_1} d_{j_1}} \tag{7.10}$$

Since the linear combination weights, γ, must be dimensionless, it is clear that the free parameter c has the dimension of a depth. Note also that the same c must apply across $\gamma_{a,*}$, that is, across all the weights associated with triangle a. The natural choice for this parameter is therefore d_a, the water depth at the circumcentre of the triangle. This is the one depth which is shared by all the faces of that triangle.

In the analysis presented here, the Coriolis parameter, f has been taken to be constant. In fact, it is possible to construct a similar analysis without making this assumption in which case equation (7.10) becomes:

$$\gamma_{a,j_1} = \frac{\Delta x_{a,j_1} d_a f_a}{\Delta x_{j_1} d_{j_1} f_{j_1}} \tag{7.11}$$

where f_a is the value of the Coriolis parameter in triangle a and f_{j_1} is the value of that parameter at the centre of face j_1.

7.3 Physical properties of the reconstruction

That this interpolation preserves the stability energy norm conservation properties of the orthogonal mesh based scheme derived in section 5.5.3 has been shown by construction. Of course no such stability proof exists for the path integral scheme on non-orthogonal meshes. Although the Coriolis term itself preserves the energy norm $\langle \mathbf{v}|\mathrm{D}|\mathbf{v}\rangle$, this does not of itself provide insight into the behaviour of the system as a whole.

As an interesting aside, if the γ we have constructed are substituted into equation (7.6) then the following relationship is apparent:

$$d_{j_1} v_{j_1} = \frac{\Delta x_{a,j_1}}{\Delta x_{j_1}} d_a v_{a,j_1} + \frac{\Delta x_{b,j_1}}{\Delta x_{j_1}} d_b v_{b,j_1}. \tag{7.12}$$

Since $\Delta x_{j_1} = \Delta x_{a,j_1} + \Delta x_{b,j_1}$, the v_{j_1} may be interpreted as an interpolation of the transverse component of the depth-integrated velocity.

7.4 Eigenvalues of the systems

To illustrate the effect of the Coriolis discretisation on the symmetry properties of the two schemes, the eigenvalue calculation presented in section 5.5.5 was repeated with the inclusion of a Coriolis force with a value of f equivalent to a latitude of 90° North. The results of this test are shown in figure 7.2. As would be expected, adding the energy conserving Coriolis scheme developed for the centroid case makes no noticeable difference to the eigenvalues. As before, the orthogonal mesh scheme has purely imaginary eigenvalues to machine precision and therefore preserves the symmetry properties of the original equation. The path integral non-orthogonal scheme has eigenvalues with real part but the magnitude of these are unaffected, to machine precision, by the introduction of the Coriolis term.

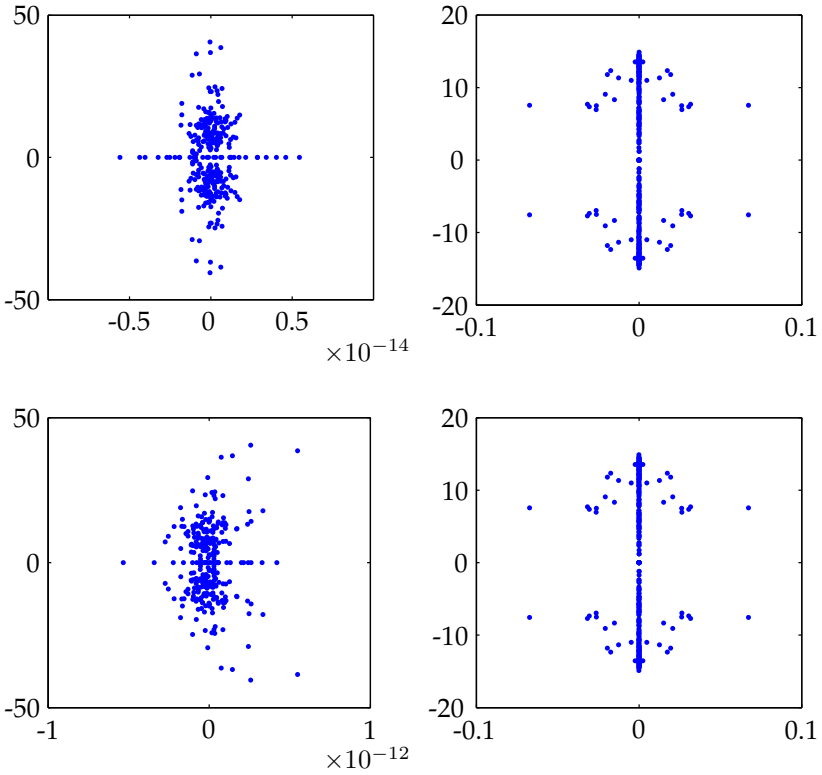

Figure 7.2: Eigenvalues of the discrete systems for the orthogonal scheme (left) and the path integral scheme (right). The plots in the first row are calculated without Coriolis while Coriolis was included in the second row. Note that the axis limits in both the real (horizontal) and imaginary (vertical) directions differ between the plots. In particular, the eigenvalues of the circumcentre scheme are zero to machine precision in each case.

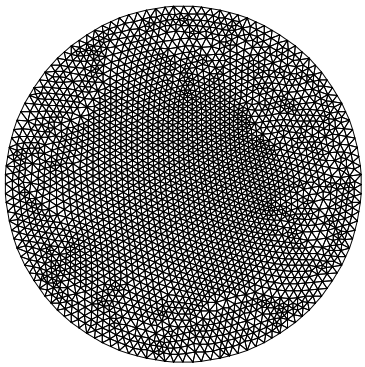

Figure 7.3: The mesh used for the Kelvin wave test case. The basin radius is 250km and the nominal triangle edge length is 10km

7.5 Kelvin wave test case

As a test case which stresses the balance between the pressure gradient and the Coriolis terms, we choose to investigate the propagation of a Kelvin wave in a shallow circular basin. A similar test case presented in Ham et al. (2005) prompted the current analysis of the Coriolis term.

The basin was given a uniform depth of 5 m and a radius of 250km. A mesh with a nominal edge length of 10km was generated using the algorithm presented in Legrand et al. (2000) (see figure 7.3) and the Crank-Nicholson time discretisation was applied. This simulation was conducted at a specified latitude of 45°, the latitude at which the f-plane approximation used in this scheme is most accurate.

The initial state chosen was:

$$\eta(r,\theta) = 0.05\mathbf{e}^{(r-r_0)/L_D}\cos\theta \tag{7.13}$$

$$u_\theta(r,\theta) = 0.05\sqrt{g/h}\mathbf{e}^{(r-r_0)/L_D}\cos\theta \tag{7.14}$$

$$u_r(r,\theta) = 0 \tag{7.15}$$

where L_D is the Rossby radius, in this case approximately 68km, and r_0 is the basin width. In the limiting case of an infinitely large basin, this is the expression for a Kelvin wave of amplitude 5cm (Csanady, 1982). The specified simulation time was 10 years.

7.5.1 Results

Figure 7.4 shows the evolution of the energy norm for each scheme. It is apparent that the path integral scheme exhibits unstable modes very rapidly: within two weeks of the start of the simulation. It is noteworthy how much faster these instabilities grow than those observed in the standing wave calculations

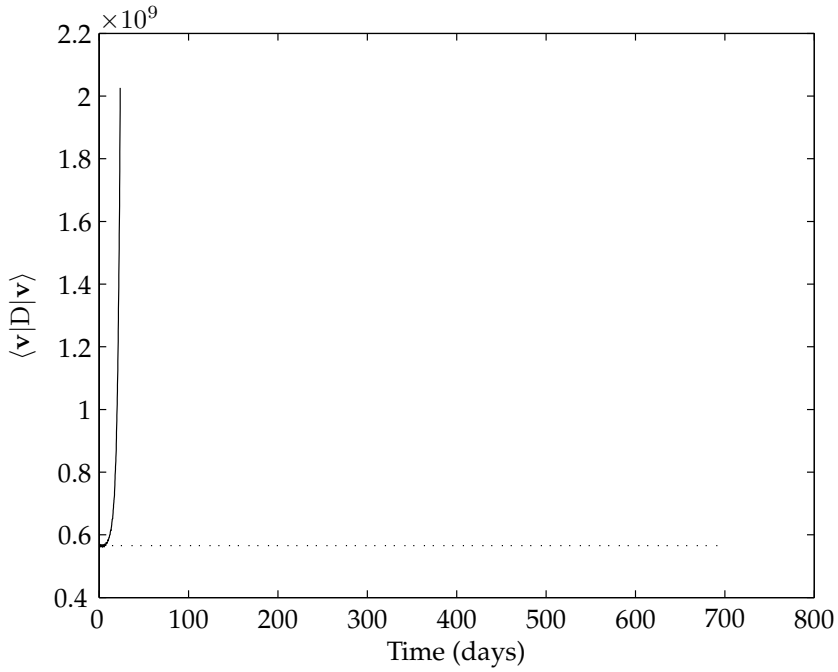

Figure 7.4: Energy norm against time for the Kelvin wave test case for the first two years of the simulation. The solid line gives the result for the non-orthogonal path integral scheme while the broken line is the result for the orthogonal mesh scheme. The latter scheme was run for 10 years in total and no change to the energy norm occurred.

presented in section 5.6.3. The orthogonal scheme, however, once again conserves the energy norm exactly to the observable precision for the full 10 years of the simulation. Figure 7.4 is limited to two years to make the result in the unstable case easier to observe.

The instabilities of the path integral scheme are distinctive and take the form shown in figure 7.5. Unstable vortices form around regions of increased or depressed surface elevation. These instabilities are clearly qualitatively different from those observed in the previous test case. The direction of rotation is consistent with geostrophic flow and the phenomenon is consistent with the source of the instability in the equations. That is, the figure is consistent with a surface elevation gradient generating spurious velocities via the non-symmetry preserving pressure gradient operator. These velocities flow under the influence of Coriolis forming vortices around elevations or depressions. This then forms a feedback loop in which the elevation or depression at the centre of the vortex generates further spuriously elevated velocities which cause the elevation or depression to grow still further. As noted before, the Coriolis term preserves

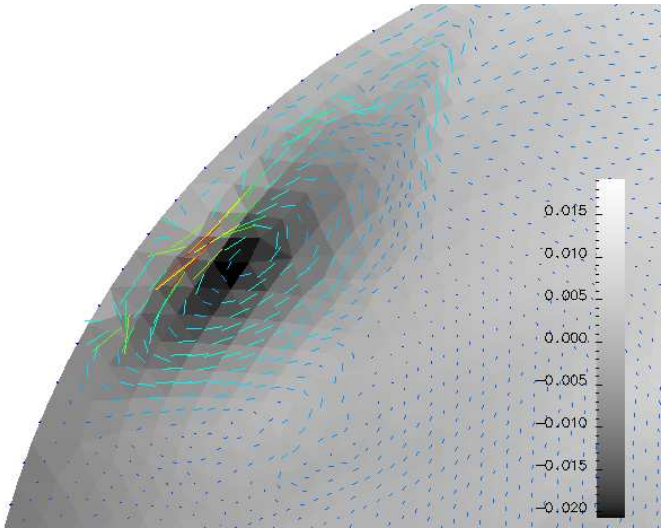

Figure 7.5: Unstable rotating mode generated by the centroid scheme. The maximum velocity shown is approximately 0.4m/s

the discrete energy analogue but this does not prevent it inducing system states in which other instabilities are fed.

The significance of the Coriolis term is most evident in the difference in timescales over which this test case and the standing wave test case presented in section 5.6.3 became unstable. Note that the two test problems have similar length and timescales and the long wave celerity and Courant numbers for the two problems are also approximately equal. It is therefore apparent that instability occurs very much more rapidly in the presence of the Coriolis term. Furthermore, increasing the implicitness parameter, θ, to 0.55 did not have the stabilising effect which was observed in the standing wave test cases.

7.6 Alternative time stepping schemes

The instabilities demonstrated in this chapter result from the spectral properties of the semidiscrete system rather than from the time marching algorithm. When the full shallow water equations are to be solved, the maintenance of the matrix structure described in section 4.4 demands that the time stepping of the Coriolis term be fractional (that is, explicit) with respect to the other terms of the momentum equation but it is not necessary that the θ scheme be used. Indeed, for large or three dimensional problems, the θ scheme may be inconvenient and expensive. For this reason Delfin uses an extrapolation scheme in time so that the operator $F(u)$ in equation (4.6) becomes:

$$F(u)^n_{j,k} = \mathbf{u}^*_{j,k} \cdot \mathbf{n}_j + f\left(v^n_{j,k} + \gamma(v^{n-1}_{j,k} - v^n_{j,k})\right) \tag{7.16}$$

where $\mathbf{u}^*_{j,k}$ is the advected velocity calculated using the semi-Lagrangian scheme described in chapter 6. If γ is set to 0.5 then this is the Adams-Bashforth scheme for the Coriolis term.

7.7 Conclusion

It has been demonstrated here that the different symmetry and stability characteristics of the two discrete pressure gradient operators introduced in chapter 5 are observable to a much greater extent in the presence of Coriolis.

Where the surface elevations are located at the circumcentres of the triangles it has been shown that the resulting system matrix may be rescaled to be symmetric and that the scheme exhibits a form of discretised energy conservation. A Coriolis discretisation which preserves these symmetry properties has been derived and the stability of the resulting scheme has been both proven and demonstrated in a test case. This discretisation uses only the properties of the circumcentre so it is also applicable to schemes such as Casulli and Walters (2000) in which cyclic polygons are also permitted in the mesh. This study has been restricted to finite volume/finite difference schemes but it seems likely that this form of analysis could also be applied to finite element schemes based on the lowest order Raviart-Thomas element.

In the path integral scheme considered here, the pressure gradient operator has a larger stencil than in the circumcentre case. As a result, the semidiscrete matrix is not sign skew symmetric and hence not equivalent to a scaled skew symmetric system. In this case, whether or not Coriolis is introduced, there exist unstable spurious modes which will be excited by suitable system states. It should be noted that the existence of spurious modes associated with an operator is no guarantee that they will be observed in any given simulation, however we have demonstrated here that they are particularly evident in the case of a long running non-dissipative Coriolis dominated problem. From this we can conclude that, especially for large scale problems in which Coriolis is important and where there is little natural dissipation, the stability properties of the skew symmetric circumcentre scheme are superior to those of the other mechanism we have studied, and additionally provide desirable energy conservation properties.

Chapter 8

The Indian Ocean Tsunami

The Indian Ocean Tsunami of 26 December 2004 was a human tragedy of colossal scale. It has provoked significant scientific interest both due to the desire to create more effective warning systems and due to the opportunity that such a large and prominent event presents to improve our understanding both of the physical processes involved and of the techniques available for studying them.

Tsunamis are usually long surface gravity waves generated by rapid displacement of the sea bed. This tsunami was driven by a massive earthquake along a stretch of fault stretching from Sumatra through the Nicobar Islands to the Andaman Islands. There are therefore two distinct processes of interest to science. The first is the earthquake dislocation and resulting sea bed movements while the second is the formation and propagation of the resulting tsunami. The conventional mechanism for determining the surface uplift due to an earthquake is to invert the seismographic signals received at many different recording stations (see, for example, Ammon et al., 2005). However, Vigny et al. (2005) presented a method for determining the earth movement associated with the Sumatra-Andaman earthquake using the GPS data of the ASEAN/EU SEAMERGES project.

Two distinct types of observation of the propagation of the tsunami wave are available. First, the arrival time of the first, and sometimes of subsequent, waves is known at a number of points around the Bay of Bengal. Second, and probably more significantly, the Jason-1 oceanography satellite passed over the Bay of Bengal approximately 2 hours after the earthquake occurred. The sea surface height data returned by the satellite provides a track of the major direct and reflected tsunami waves in the Indian Ocean.

The challenge for the modeller is to fill the gap between the sea bed movement generated from the GPS data and the observed arrival times and surface elevations. Can Delfin reproduce reasonable agreement with the observed data? Is it sensitive and accurate enough to distinguish between different GPS inversions and even to suggest where errors in the initial fields may lie? We shall see that to each of these questions the answer is a qualified "yes".

8.1 The unstructured mesh

The Indian Ocean Tsunami has two features which make an unstructured mesh particularly appealing. First, the region in which the tsunami is generated is an Island chain. The shape of the coastlines and the bathymetry around them will influence the reflections generated and the steering of the waves. It is therefore important the the islands be well resolved and accurately represented. Second, the length scale of wave features will be shortest near coasts and in shallow water and will be larger in deeper water as the tsunami spreads out. It will therefore be advantageous to concentrate the highest resolution in coastal areas close to the generation of the tsunami wave and to conserve the available computing power by reducing the resolution in the deep ocean. All geographic data expressed in spherical coordinates was converted to metres using a Mercator projection centred at 95° E, 0° N.

8.1.1 Coastlines

In a high quality mesh, the resolution at a coastline is represented will be equal to the resolution of the mesh immediately adjacent to that coastline. Since a mesh generator will insert coast points, if at all, directly on the intervals joining existing coastal points, accuracy in the representation of the coastline representation is maximised if the coastline is initially sampled at the resolution which is to apply. In fact, where the representation of the coastline is of particular importance, the coastline can be sampled at very high resolution and minimum quality requirements passed to the mesh generator to ensure that the adjacent mesh resolution is suitable.

The coastlines were based on the World Vector Shoreline produced by the US National Imagery and Mapping agency and published in GEBCO (2003). In areas of particular interest such as the Andaman and Nicobar Islands, the coastline was sampled at 500m resolution falling back to 1km on the Sumatran coast and parts of Thailand through 10km around Sri Lanka to 20km on Northern parts of the Indian coast. The seaward boundaries of the domain to the South and West were sampled at 40km.

8.1.2 Prescribed resolution

The mesh generator was provided with a nominal triangle area at each point in the domain. While it is possible to arrive at mesh resolution criteria mechanically based on domain features such as long wave speed (that is, depth) and bathymetry gradients, in this case the areas of interest are known in advance and do not necessarily correspond to bathymetric features. The nominal side lengths, and hence triangle areas, of the mesh were therefore prescribed manually as follows.

A base minimum resolution of 40km was established. North and East of 75° E, 3° S this increases to 20km and past 78° E, 0° N it is approximately 12km. In

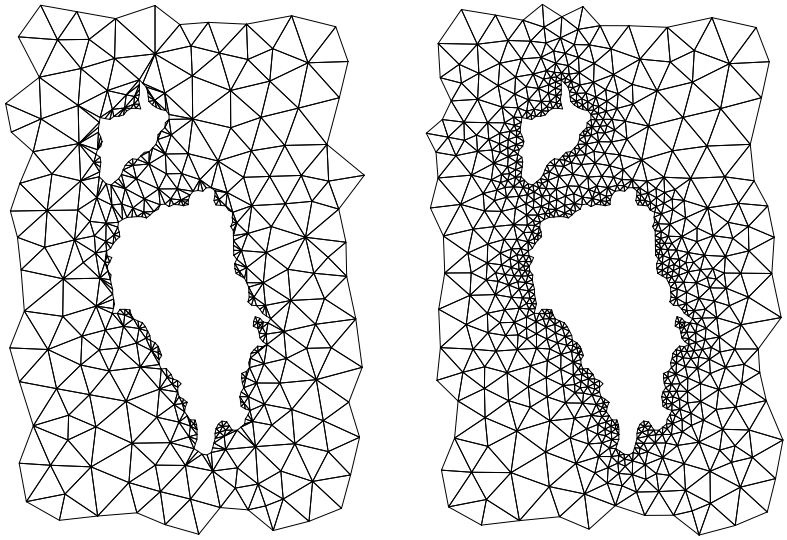

Figure 8.1: The mesh around Great Nicobar before and after the imposition of a minimum angle.

addition a high resolution region with a nominal mesh edge length of 5km is established in the tsunami generation region between Sumatra and the Andaman Islands and extending Eastwards to Phuket in Thailand.

8.1.3 Mesh Generation

The mesh was generated using the Resolute mesh generator (Chaffey and Greenberg, 2003) which builds on the Triangle (Shewchuk, 1996) mesh generator by employing a node position averaging algorithm to improve mesh quality. Resolute employs an iterative approach to improving mesh quality while Triangle employs a direct node insertion algorithm based on prescribed triangle area and minimum angle requirements. A limitation of the Triangle minimum angle algorithm is that, when applied to improve the quality of a low quality mesh, it tends to result in clumps of very small triangles in areas where the input mesh is of low quality. To overcome this, no minimum angle was specified for the first several iterations of refinement and quality improvement. This resulted in a mesh which was of a high quality in the interior of the domain but with much poorer elements close to highly resolved coastlines. When a minimum angle requirement of 30° was applied to the final refinement iteration, the transition to high resolution along the costs became smoother and the associated elements were of higher quality. Figure 8.1 illustrates this process.

The full mesh is shown in figure 8.2. It is immediately apparent that the

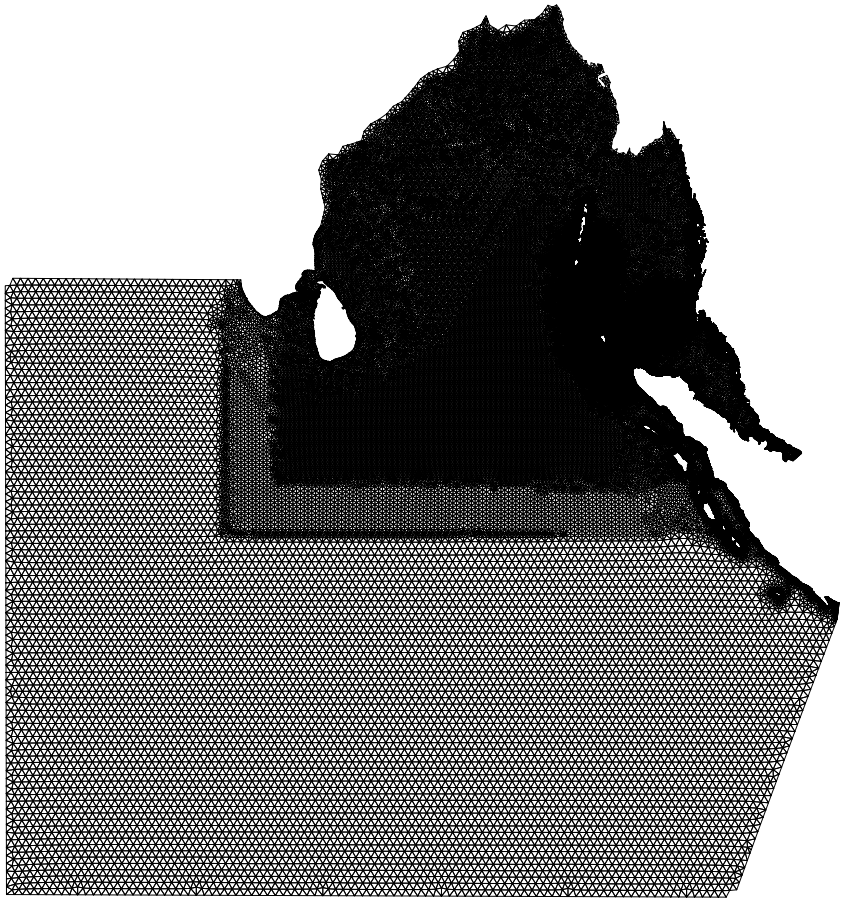

Figure 8.2: The mesh used to simulate the Indian Ocean Tsunami

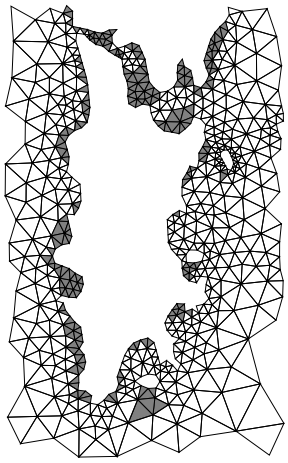

Figure 8.3: The mesh around Phuket showing triangles with bathymetry above mean sea level in grey.

unstructured mesh allows very significant concentration of the resolution in areas of interest while allowing a relatively large computational domain. The final mesh has 161746 triangular elements.

8.2 Bathymetry

The bathymetry used was drawn from the General Bathymetric Chart of the Oceans (GEBCO, 2003). This resource is not entirely satisfactory. For example, even though the World Vector Shoreline is published alongside the GEBCO data, it fails to match the zero contour in that bathymetry, often quite significantly. As an illustration, figure 8.3 shows the mesh around Phuket. The mesh boundary is drawn according to the World Vector Shoreline but the elements shown in grey are exposed according to the GEBCO bathymetry. Some of the dry elements away from the coast may be attributable to unresolved islands, however those adjacent to the coast indicate data errors. Actual bathymetry measurements are known to be sparse over much of the globe and the GEBCO data set relies extensively on interpolation. The bathymetry used must therefore be regarded as an approximation, especially in shallow coastal regions in which any errors may comprise a significant proportion of the water depth. It is doubtful whether better bathymetry data is available easily or at all for large parts of the domain.

8.3 Boundary and parameter values

One of the objectives of including the large low resolution region in the South and West of the domain is to ensure the water boundaries are removed from the area of interest of the simulation. For this reason it was considered appropriate to treat all boundaries as closed. Wind effects are unlikely to be observable on the time and length scales of the tsunami so the wind is ignored. Since the region of interest is in the region of the equator, the Coriolis acceleration is also uniformly set to 0. Since the only hydrodynamic feature of interest is the propagation of long waves, the simulation was conducted with a single layer. The bed roughness length was set uniformly to 5cm. The time step was set to 30s; this value being chosen to adequately resolve the time scale of the initial rupture and subsequent wave propagation rather than by any stability limitation. To avoid artificially damping the tsunami wave, the simulations were conducted with an implicitness, θ, of 0.5.

8.4 Observations of tsunami propagation

8.4.1 Satellite data

Two oceanographic satellites passed over the propagating tsunami approximately two hours after the Earthquake. The two satellites - Jason-1 and TOPEX/-Poseidon follow almost identical orbits and repeat their tracks every 10 days (Ménard et al., 2003). The satellite data which is relevant to this investigation is the sea surface anomaly. This data potentially provides a cross section through the propagating tsunami including both direct and reflected waves. Furthermore, the satellite altimetry primarily reflects the propagation of the tsunami in deep water thereby reducing the probability that any inconsistencies are attributable to the dubious accuracy of the bathymetry in shallow water.

The processed satellite data was obtained from the Department of Earth Observation and Space Systems at Delft University of Technology. The data from the newer and more accurate Jason-1 is substantially complete over the area of interest whereas the TOPEX/Poseidon data includes substantial gaps in key regions. In an attempt to remove long term background signals, the data was compared to that for the previous pass of each satellite 10 days before. In the case of TOPEX/Poseidon, the comparison data was missing for the Northern part of the track and the total length over which data from both passes was available was short and fragmented. The data record for Jason-1 is once again substantially complete and several features are well correlated indicating that they do not form part of the tsunami signal. Figure 8.4 shows the track of Jason-1 over the region of interest while figure 8.5 shows the data from the two passes and the candidate tsunami signal obtained by subtracting the previous signal from the current one.

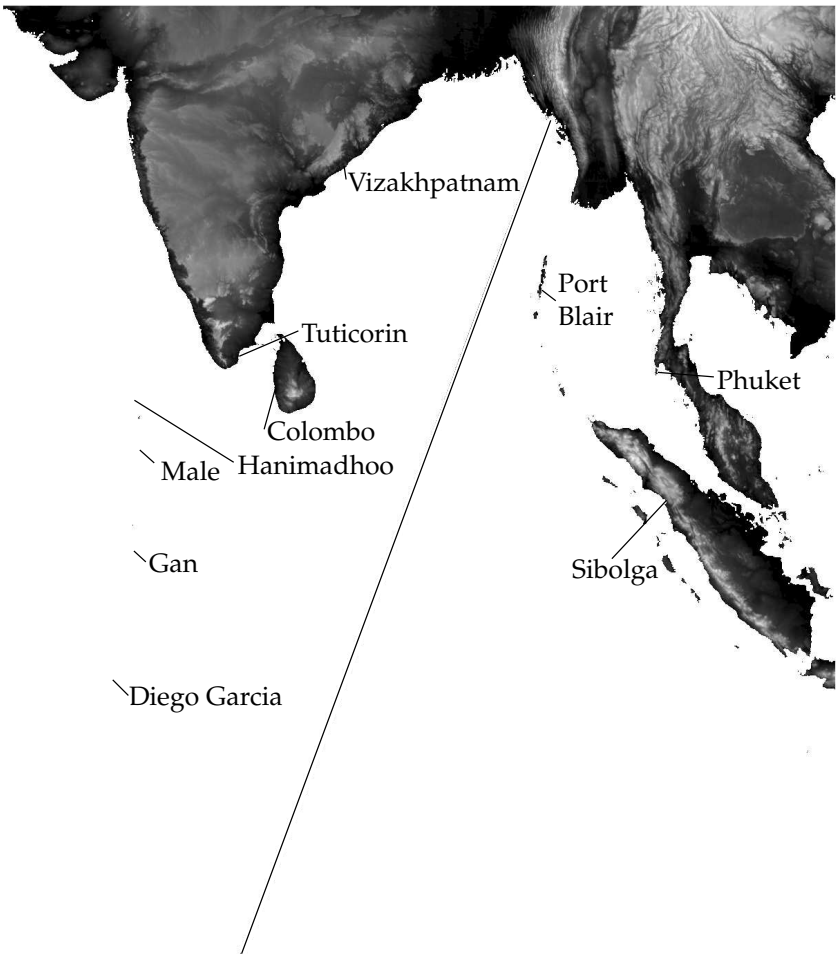

Figure 8.4: The Bay of Bengal and part of the Indian Ocean showing the path followed by the satellite Jason-1 and locations at which the arrival time of the tsunami is known.

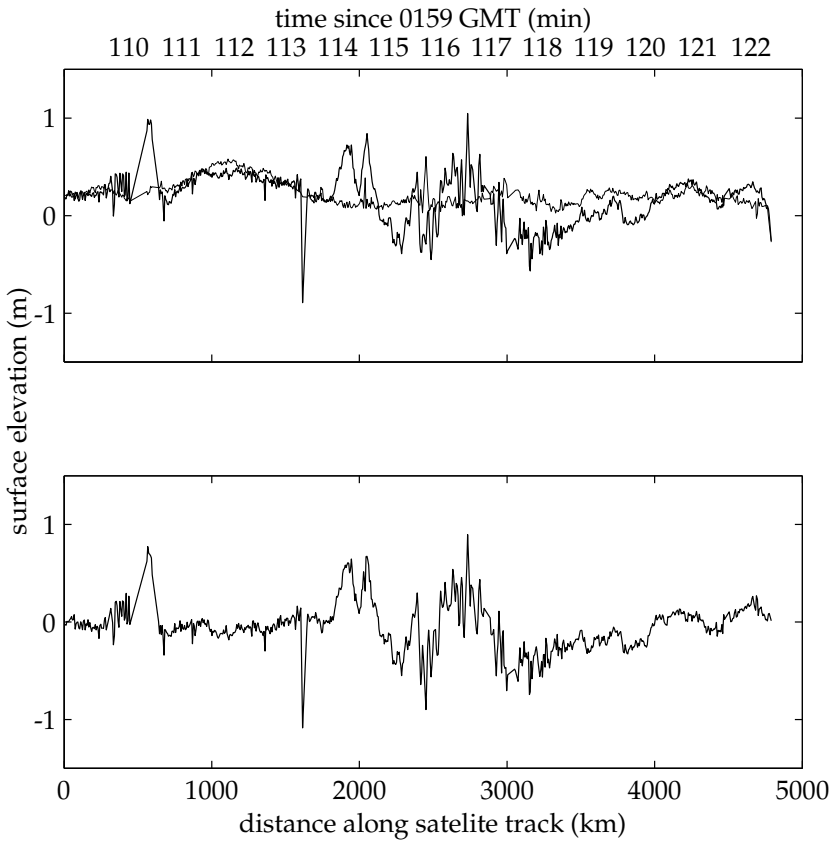

Figure 8.5: Above: processed Jason-1 satellite data from 16 December 2004 (light line) and 26 December 2004 (heavy line). Below: the difference of the two signals. The double x axes show the movement of the satellite in time and space relative to the point at which the satellite enters the domain.

8.4.2 Coastal arrival data

The arrival times of one or more tsunami peaks were recorded by tide gauges at various points around the Indian Ocean. The advantage of this data is that it is available at a wider geographic range than is the satellite data. However, the wave height observed at point on the coast is strongly influenced by relatively small scale bathymetric and coastal features which may be underresolved by the computational mesh or poorly represented in the bathymetry and coastline data used. Where the sea adjacent to the measurement point is shallow, inaccuracies in the representation of bathymetry may also affect the arrival times. In addition, underresolved or inaccurate local data raises the probability that local reflections will be incorrectly modelled so data after the initial incident wave is

Site name	Location	Arrival time (min)	Data source
Sibolga	1° 45′ N 98° 46′ E	161	Bakosurtanal
Tuticorin	8° 45′ N 78° 13′ E	208	NIO India
Vizakhapatnam	17° 42′ N 83° 18′ E	155	NIO India
Colombo	6° 59′ N 79° 51′ E	180	UHSLC
Male	4° 11′ N 73° 31′ E	190	UHSLC
Diego Garcia	7° 14′ S 72° 26′ E	230	UHSLC
Hanimadhoo	6° 46′ N 73° 1′ E	220	UHSLC
Gan	0° 41′ S 73° 9′ E	198	UHSLC
Port Blair	11° 40′ N 92° 30′ E	30	

Table 8.1: Arrival times of peaks and troughs in minutes after the commencement of the earthquake (0159 GMT). NIO is the Indian National Institute for Oceanography, Bakosurtanal is the Indonesian National Coordinating Agency for Surveys and Mapping, UHSLC is University of Hawaii Sea Level Center. Data compiled by Deepak Vatvani of WL|Delft Hydraulics except for Port Blair which was published in Neetu et al. (2005).

unlikely to be reproduced by the model. However, despite these caveats, the coastal observations of arrival time still provide a resource which can be used to test and validate the model and its initial conditions. Table 8.1 shows the arrival times of the initial tsunami wave at various points around the Indian Ocean while figure 8.4 shows the locations of these points. With the exception of the data at Port Blair, the arrival times were determined from tide gauge data and are accurate to within 5 or 10 minutes. The arrival time of the initial large wave is shown in each case. A smaller wave is apparent in the Sibolga tide gauge at 120 minutes. The Port Blair result is the reconstructed value given in Neetu et al. (2005).

8.5 Initial conditions

8.5.1 Treatment of initial conditions

The sea surface displacement which initiates the tsunami is a result of the displacement of the sea floor. In this investigation this is modelled by simply displacing the sea surface instantaneously by the bed uplift predicted by the GPS model. A more accurate approach might be to displace the bed and allow the resulting forces to displace the water however those forces are not representable in a hydrostatic model. Furthermore, the shallowness of the water is such that, even if it is assumed that the water is displaced not upwards but radially outward in all directions, the surface displacement will occur within a few kilometres of that predicted by simply moving the sea surface with the bed. Compared with the length scale of the long shallow tsunami waves this

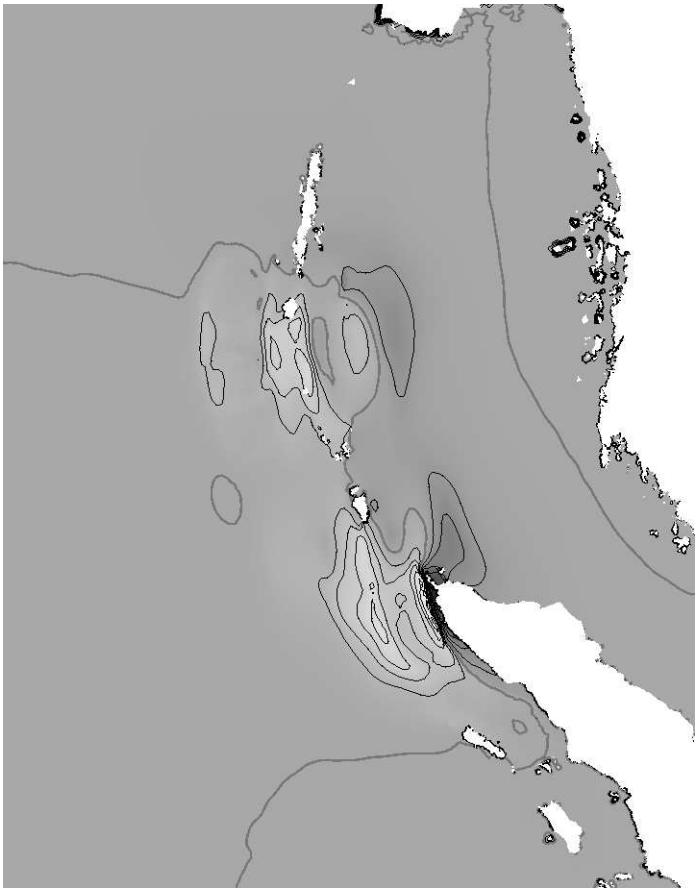

Figure 8.6: Modelled sea bed displacements caused by the Andaman-Sumatra earthquake. This model is based on the existing USGS fault geometry. The contour interval is 0.5m. The broader contour is at 0m.

error is unlikely to be significant. The Finel non-hydrostatic model (Labeur and Pietrzak, 2005) has been used to model the initial stages of tsunami propagation driven directly by the motion of the sea floor. The results indicate that the assumptions made in this paragraph are valid.

8.5.2 Calculation of the initial fields

The techniques used to calculate the initial fields are presented in Pietrzak et al. (2006) and the details are beyond the scope of this work. Two different uplift fields were received differing in the number of GPS data points used to constrain the modelled fault movement and in the geometry of the fault used in that model. The first data set (figure 8.6) was based on known USGS data while

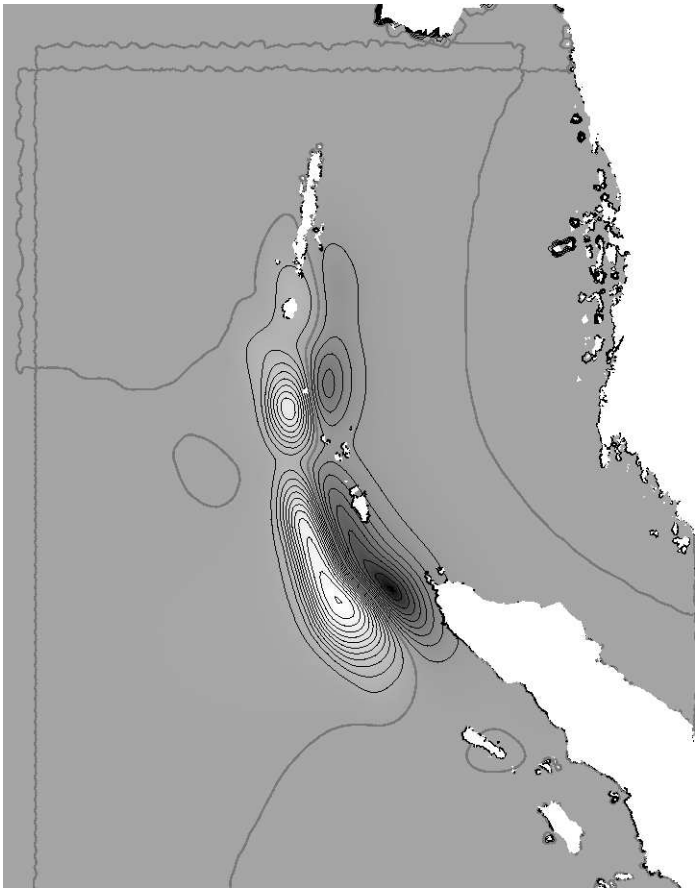

Figure 8.7: Modelled sea bed displacements with fault geometry based on aftershock data. The contour interval is 0.5m. The broader contour is at 0m.

for the second set (figure 8.7) aftershock data from the 2004 earthquake was used to construct a new fault geometry. A notable feature of both data sets is the region of no displacement coincident with the Nicobar islands. This gap is not present in the reconstructed slip fields based on seismic data (Bilham, 2005). The two sets of data differ in significant ways:

1. The maximum magnitude of the uplift increases from approximately 3.5m in the first field to around 6m in the second field.

2. The peak of the Northern part of the displacement is further South in field two than in field one.

3. The secondary peak to the East of the Northern displacement is absent in the second field.

4. The Southern peak is located much further West and the second peak to the East is absent in the second field.

8.5.3 Incorporating the rupture velocity

A feature of the initial surface displacement which does have a significant impact on the propagation of the tsunami waves is the differing times at which the rupture occurs at different locations along the fault. In the Indian Ocean Tsunami the region of significant displacement is over 1000km long and the rupture propagated from South to North at speeds in the range 2.2-3.7km/s (Krüger and Ohrnberger, 2005; Vigny et al., 2005). A consequence of this is that the extreme Northern end of the rupture lags the earthquake by approximately 8 minutes. This has consequences for the arrival times of the Northern portion of the wave.

This is implemented in the model by applying the surface displacement progressively. Vigny et al. (2005) lists a set of points giving the position of the rupture front every thirty seconds from the beginning of the earthquake at 0159 GMT. This data can be used to construct a decomposition of the prescribed surface displacement such that the surface can be progressively perturbed at each time step.

let $R = \{\mathbf{r}_0..\mathbf{r}_{17}\}$ be the set of points indicating the maximum extent of the rupture along the fault in 30 second increments. Then for each $i < 17$ define:

$$\phi_i(\mathbf{x}) = \max\left(0, \min\left(1, \frac{(\mathbf{x}-\mathbf{r}_i)\cdot(\mathbf{r}_{i+1}-\mathbf{r}_i)}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}\right)\right) - \sum_{j<i}\phi_j(\mathbf{x}) \tag{8.1}$$

For the special case of $i = 17$ we have the residual weighting:

$$\phi_{17}(\mathbf{x}) = 1 - \sum_{j<17}\phi_j(\mathbf{x}) \tag{8.2}$$

This decomposition reduces to a set of one dimensional P_1 basis functions along the fault line itself. The surface elevation displacement to be applied at time step i is therefore given by:

$$\Delta\eta_i(\mathbf{x}) = \phi_i(\mathbf{x})\Delta\eta_{\mathrm{GPS}}(\mathbf{x}) \tag{8.3}$$

where $\Delta\eta_{\mathrm{GPS}}(\mathbf{x})$ is the prescribed surface displacement based on the GPS data.

8.5.4 Simulations run

To determine the sensitivity of the model to the differences in the initial fields provided and to attempt to evaluate the significance of the features highlighted in sections 8.5.2 and 8.5.3, several simulations were conducted. Simulations were undertaken using each of the two initial fields and incorporating the rupture information from the preceding section.

Based on the results of these initial simulations, the second field was chosen as the source field for simulations designed to test various features of the initial conditions. To investigate which observed data constrained which parts of the initial field, simulations were conducted using only the Northern and only the Southern parts of the input field. The field was split in the region of minimal displacement and the inactive part of the displacement was set to 0.

To determine the importance of including consideration of the rupture velocity, an additional simulation was carried out using the field applied instantly at 0159 GMT.

8.6 Performance comparison with the Hansen scheme

In section 5.6.2 the Hansen scheme was shown to give similar results for long wave propagation to the θ scheme. In this series of simulations, the time step is chosen to resolve flow features rather than on the basis of a maximal allowable Courant number. This raises the question of whether, for such a problem, the Hansen scheme might be more efficient. The initial maximum wave Courant number with a 30s time step was 4.64 while the mean wave Courant number was just 0.494. To ensure that the maximum wave Courant number stayed below 1, the time step for the explicit simulation was set to 5s. This resulted in an initial maximum wave Courant number of 0.773. The average time taken to simulate 30s of simulation time using he Hansen scheme was 29.25s. This compares with 31.01s for the θ scheme for the same problem. The Hansen scheme is therefore not substantially faster for this problem than the θ scheme so the robustness of the θ scheme at larger time steps makes it the better choice.

8.7 Results

Figure 8.8 shows the surface elevations along the satellite track from the initial simulations using each of the initial data fields. The surface elevation was interpolated in time and space to correspond with each of the measurement events in the satellite data.

Figure 8.9 shows the propagation of the wave fronts for the second field at four instants in time from shortly after the beginning of the simulation until shortly before the satellite pass. The initial wave fronts in both the North and the South are captured to some extent by all of the simulations. The subsequent waves, corresponding to the region from 2500km to 4000km along the satellite track are less well represented.

If the Southern wave fronts at around 2000km on the satellite track are considered then it is immediately apparent that the results from the first initial field are the least satisfactory. The waves pass the satellite track hundreds of kilometres to the North of the observed waves and have only around half the amplitude. It is not clear that the third peak, that between 2500km and 3000km, is resolved at all.

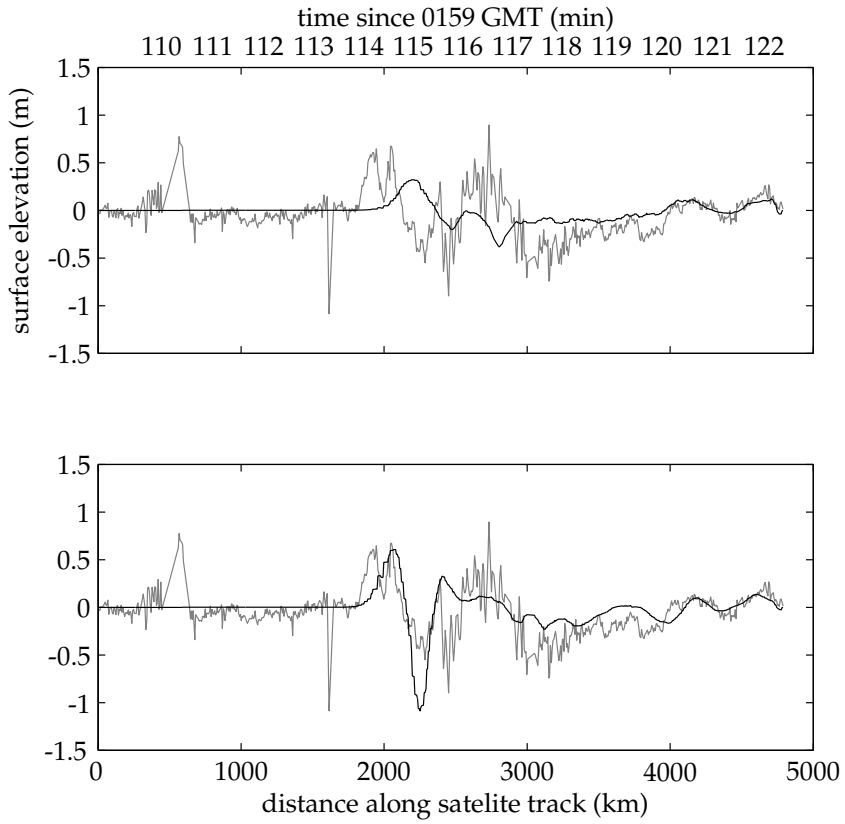

Figure 8.8: Surface elevation along the satellite track. The upper plot is from initial field one while the lower plot results from initial field two. The satellite signal is included for comparison.

The result of the second simulation in this region is significantly better. The second of the double peaks at 2000km is captured and both the amplitude and the position are correct. The second peak, at 2400km, is also captured and there is some indication that the third peak is also represented. Unfortunately the first peak is completely absent. In addition the second trough (that at around 2300km) is far too deep and the third trough (at around 2500lm) is completely absent.

At the Northern end of the track, the differences in the results are less significant. In each cases the two peaks are present in the correct location and with approximately the correct amplitude. However, the second simulation is missing part of the peak at around 4000km and may be lagging slightly at the extreme Northern end of the track.

Figure 8.10 shows the results of simulations based on variants of the second input field. The first two plots make it clear that the most important waves are

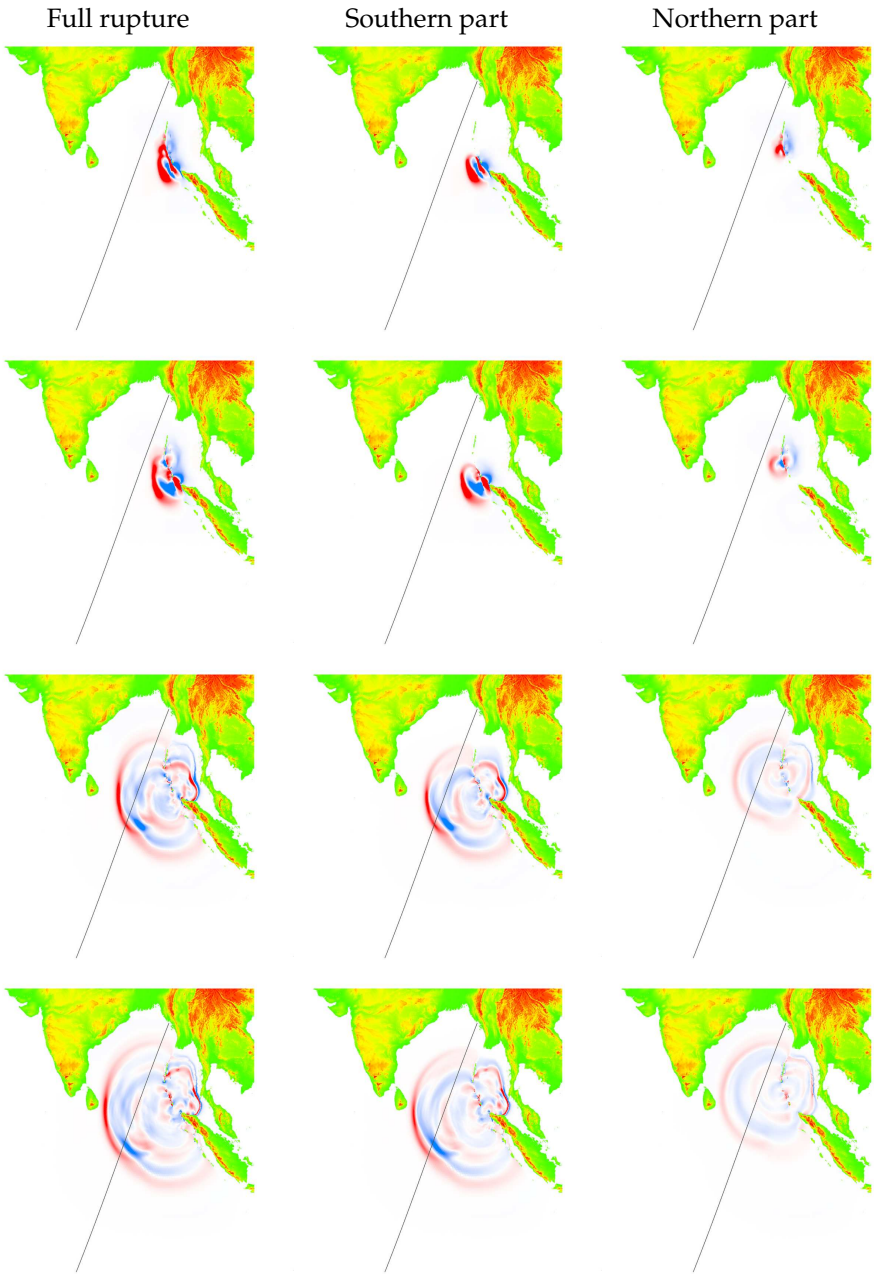

Figure 8.9: The propagation of the tsunami wave simulated using initial field two. From top to bottom, the images are taken at 750s (12.5min), 1350s (22.5min), 4800s (80min) and 6000s (100min). The satellite pass occurs shortly after the last image.

Site	Observed	v1	v2	v2(S)	v2(N)	v2(Instant)
Sibolga	161	141	134	134	189	133
Tuticorin	208	-	227	226	234	225
Vizakhapatnam	155	154	166	179	164	160
Colombo	180	185	175	175	180	173
Male	190	222	205	204	212	202
Diego Garcia	230	239	225	225	-	223
Hanimadhoo	220	-	230	228	238	228
Gan	198	217	203	203	214	201
Port Blair	30	31	34	105	34	26

Table 8.2: Tsunami arrival times in minutes after the commencement of the earthquake (0159 GMT). From the left, the data are the observed arrival time and the modelled arrival times using the first field, the second field, the Northern portion of the second field, the Southern portion of the second field and the second field assuming that the entire fault ruptured at the same instant. The blanks indicate that no peak arrived at that point during the simulation.

those caused by the Southern part of the rupture. In particular, the third peak is much better resolved when only the Southern part of the rupture is used to drive the simulations. This result also illustrates the fact (which is confirmed by animations of the simulation) that, of the Southern peaks, only the first is an incident wave. The subsequent peaks are reflections from Sumatra. It is important to note, however, that the Northernmost peak is only recovered if the Northern rupture data is included.

The bottom plot in figure 8.10 shows the effect of including the rupture propagation data in the simulations. Interestingly, the signal in the South is affected much more than that in the North.

Table 8.2 shows the arrival times of the first tsunami wave at various observation points. While recalling the limitations of such comparisons raised in section 8.4.2, there is some important information which can be recovered from this data. First, very few of the arrival times are the result of the Northern part of the initial fields. The most obvious exception is Port Blair, however at this point all of the model results are close to the observation. The eight minute difference in the arrival time which results from the inclusion of rupture speed information in the initial fields illustrates the significance of this data. The results at Vizakhapatnam are also dominated by the Northern part of the field but here again the relative differences between the first and the second field are minor.

In the West of the domain, the arrival times at locations other than Colombo are generally later than the observed time. This is consistent with the missing leading peak in the input data. These results are also significantly better for the second data set than for the first.

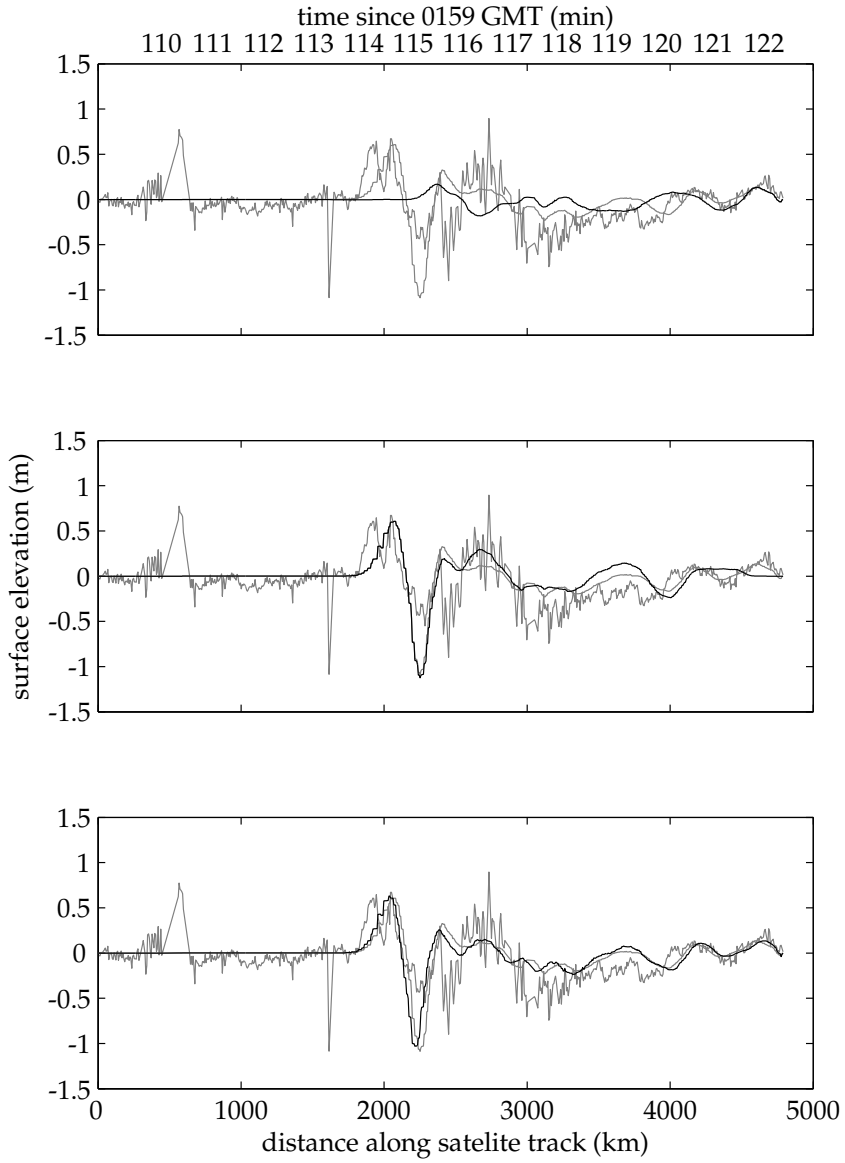

Figure 8.10: Surface elevation along the satellite track. The first two plots are the result of applying only the Northern and only the Southern parts of the second initial field respectively. The lower figure shows the results for the second field if the rupture is assumed to be instantaneous. The satellite signal and the model result for the full second data set are included for comparison.

8.8 Discussion

The simulations presented in this chapter are subject to two unknown variables: the accuracy of the initial conditions and the accuracy of the model. The model errors which are likely to be manifest in this problem are loss of amplitude due to excessive dissipation and phase speed errors in waves. The amplitude of the wave peaks resulting from the second initial field appears to be approximately correct however the troughs in the model output overshoot significantly. There are two possible causes of this. Either the model is essentially accurate and the inaccuracies in the troughs are due to the initial fields or the magnitude of the initial uplift is excessive in a manner diabolically correlated with the dissipation of the model. The former seems more likely.

Much more can be said about the propagation speeds exhibited by the model. For the second initial field, the peaks and troughs match at several points. Most notably, in the interval between 2000 and 3000 metres. This very strongly suggests that the propagation speeds are correct.

If it is assumed that the propagation speeds and dissipation exhibited by the model are essentially correct, then this information may be used to examine accuracy of the initial conditions used. The correlation of the model results with the Satellite data suggests that Southern part of the initial displacements is similar in form to that presented in the second data field and that the maximum in the first field is both too low and too far to the North and East. However, it seems apparent that there must be further displacement activity to the South or West of that presented to account for the peak at approximately 1900km on the Satellite track. The existence of significant displacement to the South of that present in the fields used here is reported in the fields based on seismic data presented in Bilham (2005) and Lay et al. (2005) (see especially supplementary figure S8). The hypothesis that the displacement must extend further is also supported by the general delay in arrival times at points in the South-West of the domain.

In the North there are fewer conclusions which may be drawn. The results at Port Blair and Vizakhapatnam agree with the observed data to within the accuracy of the tide gauges. The general picture is therefore that the lack of data makes it difficult to form strong conclusions concerning the relative merits of the different initial fields for the Northern part of the rupture. In particular, the differences between the initial fields have a minor impact on the correlation with the satellite track.

8.9 Conclusion

The coincidence in amplitude and arrival times of the various peaks in the tsunami wave between the model, especially when driven by the second initial field, and the observed data suggest that Delfin is sufficiently accurate to provide useful insights into tsunami propagation.

It is further possible to draw some conclusions about the relative merits of the initial fields generated by the GPS based fault model. In particular, it is possible to conclude that the Southern part of the second initial field is significantly more accurate that the corresponding part of the first initial field. Given the additional information which was used in constructing this field, this is reassuring. The arrival time and satellite data misfit in the South of the region also supports the hypothesis that there is a missing region of bed displacement to the South-West of the fields resolved by GPS. This coincides with the seismic data. Despite this limitation, even using the first version of the initial data, which was based on information available before the tsunami, the Delfin simulations conducted here indicate that the GPS signals do predict the fault movement sufficiently well that the direction and approximate amplitude of the tsunami can be predicted. With sufficiently fast reporting mechanisms and efficient inversion and simulation software, this indicates that GPS data could be used as part of an operational tsunami warning system.

Chapter 9

Tides in the North Sea

The simulation of flow and transport processes in the North Sea is an ultimate goal of the research program of which this project is a part. In addition, the tidal processes in the North Sea provide a real world case study which compliments that of the Indian Ocean Tsunami. While the Tsunami problem is driven by its initial conditions, the tides of the North Sea are driven by incoming tidal waves on the Atlantic boundary of the model domain. The Tsunami occurs very near the equator so the Coriolis acceleration is negligible, while the tides in the North Sea are primarily Kelvin waves in which Coriolis is significant and essential. A discussion of Kelvin waves and their role in tidal propagation can be found in Gill (1982) chapter 10. The tides of the North Sea have different scales to the tsunami: the domain is much smaller but the time scale is much longer.

Understanding and predicting tidal processes in the North Sea is an important task of Rijkswaterstaat, the Dutch Directorate for Public Works and Water Management. The current generation of models used for this task, Waqua and Triwaq are of structured curvilinear C grid type. In particular, the Dutch Continental Shelf Model, the current version of which is referred to as DCSM98, is used to simulate tidal processes over the whole North Sea (Gebraad and Philippart, 1998).

Fully calibrating a continental shelf model requires extensive tuning preferably using data assimilation techniques which are not yet a part of Delfin. Production of a full continental shelf model is therefore beyond the scope of this project. On the other hand, it is a useful test of the basic formulations presented here to conduct untuned simulations of the North Sea in the hope of producing results that are sufficiently correct that results which match or exceed the existing operational results are feasible with a tuned and extended future version of the model. To this end, simulations were conducted using the bathymetry and boundary data available for DCSM98.

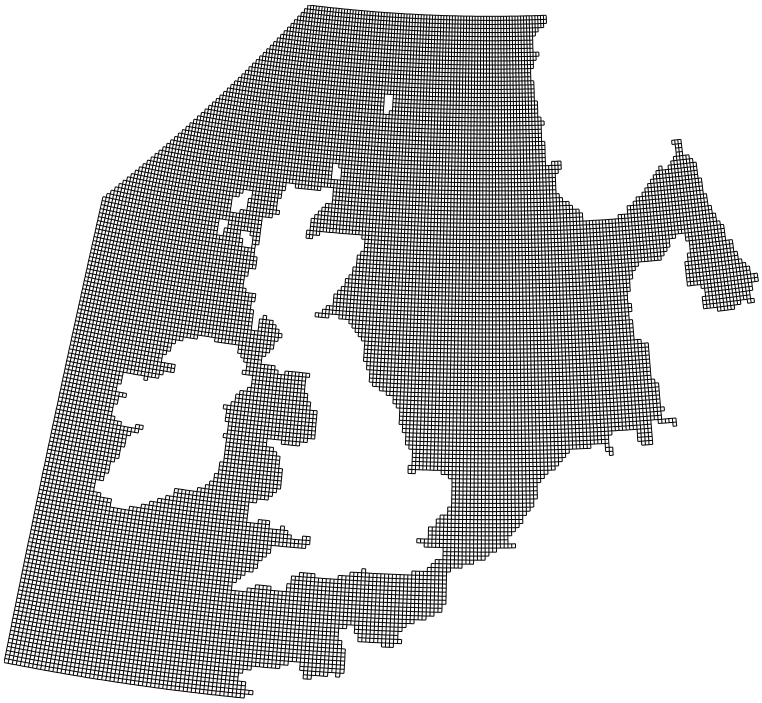

Figure 9.1: The mesh used in DCSM98. The mesh resolution is approximately 8km

9.1 The mesh

To provide modelling of processes from the continental shelf scale down to the coastal flow along the Dutch coast, the current operational system employs a hierarchy of nested models. The continental shelf model is run at approximately 8km resolution while the nested models have successively higher resolutions. Delfin, on the other hand, can accommodate meshes in which a very large resolution at the edge of the continental shelf is smoothly refined in space so that coastal regions of interest are represented at far higher resolution. Figure 9.1 shows the grid used in the current operational model DCSM8 while figure 9.2 shows the mesh employed in the Delfin simulations presented here. The longest mesh sides present in the mesh are approximately 20km long and the resolution increases as a function of distance from the coast so that much of the coastal zone is resolved with a resolution of approximately 5km and the coastal resolution is approximately 1km. The meshes were once again generated using a modified version of the Resolute mesh generator (Chaffey and Greenberg, 2003). There is a significant difference in the number of degrees of freedom

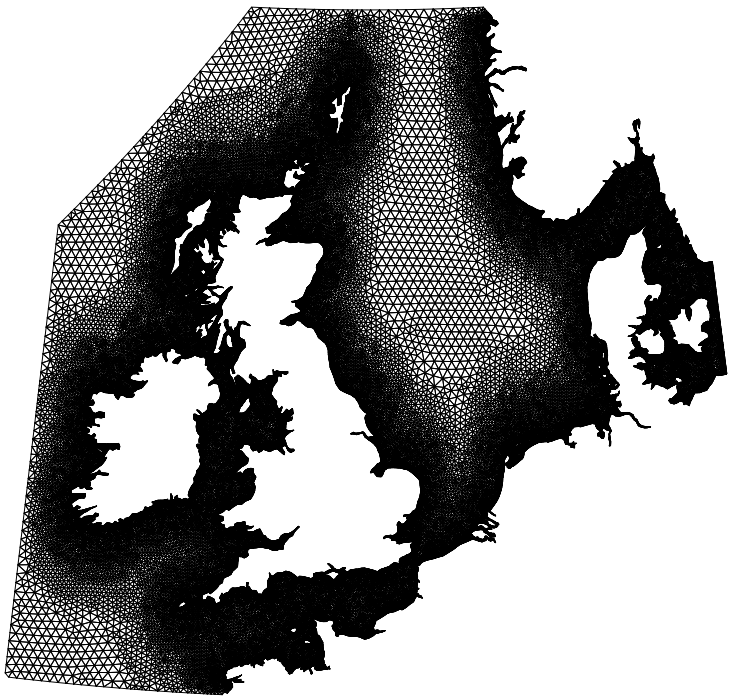

Figure 9.2: The unstructured mesh used in the simulations. The mesh resolution varies from 20km to 1km.

in the meshes. The unstructured mesh presented here consists of just under 120 000 triangular elements while the DCSM8 mesh notionally consists of just under 35000 cells, although a significant proportion are inactive.

The justification for choosing this mesh configuration is that the tidal wave is a Kelvin wave trapped against the coast so the most significant meshing priority is that the coast and coastal region are well resolved. At this stage the mesh is non-discriminatory in that all coastal regions are resolved with the same resolution. There is a case, as a more complete continental shelf model is constructed, for increased resolution in areas of interest, such as the Dutch coastal zone. The boundary placement is also less than optimal. The boundaries are collocated with those of DCSM98 to facilitate the use of the same boundary conditions. However, an advantage of modelling the whole continental shelf is that the boundaries may be placed in deep water where long wave velocities are low and boundary effects are therefore minimal. In the case of the DCSM98 mesh, and hence of the unstructured mesh presented here, the mesh boundary off Ireland remains on the continental shelf and the North-Western inclined boundary is on the shelf around the Faeroe Islands (which

are just outside the mesh). Since the cost of expanding the unstructured mesh in regions of low resolution is minimal, a future improved mesh could extend significantly further into the North Atlantic.

Despite the limitations of this domain, it is also immediately obvious that the unstructured mesh enjoys significant advantages over the existing structured grid. The staircase representation of coastlines which is evident in the structured grid is absent from the unstructured mesh and many coastal features which are absent in the DCSM98 grid are present in the structured mesh. Indeed, there are several islands which are either much reduced, or absent in the DCSM98 grid. Finally, important straights are much better resolved in the unstructured mesh. In particular, the Straits of Dover, which are only 3 grid cells across at the narrowest point, are around 20 mesh triangles across and this could easily be increased. Even taking into account the larger number of degrees of freedom in the unstructured mesh, it is clearly a better representation of the model domain.

9.2 Model parameters and boundary settings

The simulation is driven solely by the surface elevation on the Atlantic boundaries. The data used was that used in DCSM98 although for simplicity of analysis, only the diurnal lunar ("M2") component of the tide was specified. This component of the tide is one for which observations as well as the DCSM98 results are available. The bathymetry used was that of the existing mesh. This has the drawback that the bathymetry has already been down-sampled to the 8km resolution. This is particularly an issue in the coastal region where the additional coastal features resolved by the model may be recorded as dry or with bathymetries with a very large relative error. On the other hand, the limitations of publicly available bathymetry data were discussed in section 8.2. In addition, the bathymetry used has been heavily tuned to the DCSM8 model including to correct for the poor resolution of the Straits of Dover (Gebraad and Philippart, 1998). A future closer collaboration with Rijkswaterstaat including access to a more complete bathymetry and coastline database will remedy this limitation.

Modifications of the bottom friction parameters form the most important part of the tuning of DCSM98. Since this option is not currently available for Delfin, a single relatively low roughness length of 1cm was specified for the entire domain. The implicitness parameter θ was set to 0.55 and a time step of 10 minutes was chosen. At almost 75 time steps per tidal cycle, the M2 tide is thus well resolved in time. The simulation was run for 10 days of simulation time. The latitude used in the calculation of the Coriolis parameter was set to 55° which is a reasonable value for the Southern North Sea although a variable value would clearly be preferable. The path integral pressure gradient formulation was employed along with an earlier version of the tangent velocity interpolation scheme than that discussed in chapter 7. The simulation was, like the DCSM98 model, conducted with a single layer in the vertical.

9.3 Results and discussion

To evaluate the outcome of the simulation conducted, part of the data presented in Gebraad and Philippart (1998) is used. The observed M2 tide at the points shown in figure 9.3 is compared with the available results for the two models. Figure 9.4 shows a visual representation of the tidal phase and amplitude at each point and for each data source while tables 9.1 and 9.2 give the relative performance of the models in amplitude and phase respectively. The basic characteristics of the tidal wave are that it is driven by a periodic surface displacement in the North Atlantic which progresses from South to North along the domain boundary. Inside the domain, this causes a wave to progress up the English Channel and back each tidal cycle. More significantly, a Kelvin wave is generated which passes in an anticlockwise direction down the British coast and along the Dutch coast towards Denmark. No instabilities of the sort discussed in chapter 7 were observed, although it must be noted that these simulations were conducted for a relatively short period of 10 days model time.

The first observation which must be made is that the Delfin results are not as accurate as those of the DCSM98 model. However, the degree of the inaccuracy varies in space. Generally, away from the North Sea coast, both the phase and amplitude of the Delfin results are close to those of DCSM98. There is a structural lagging phase error evident in the Delfin results which, at least in the deeper areas, is a little surprising since the M2 tidal wave should be well resolved in space and time by the parameters given. As the tidal wave passes down the East coast of the United Kingdom there is an immediate amplitude loss which generally increases as the wave progresses.

Despite the general result that the Delfin simulation is less accurate than the DCSM98 simulation, there are individual points at which Delfin is more accurate than DCSM98. For example, at STATIONK13A Delfin is the better performing model for both variables. At the stations TEXWGT and DENHDR the Delfin result is also generally the better one. It is significant that these sites are near or on the Dutch coast and are therefore in the region in which the model generally performs relatively poorly. In these cases, and especially that of DENHDR (Den Helder), the higher resolution around the coast and islands may also play a part since the Wadden Sea and Wadden Islands are absent in the DCSM98 mesh.

The loss of amplitude and phase occurs largely in the shallowest areas traversed by the tidal wave. This suggests that the effective dissipation in these areas is excessive. Given the good results for long waves in deep water presented in chapter 8, this suggests that some combination of tuning the bed friction parameters and better bathymetry and coastal data might produce substantially better results.

9.4 Conclusion

The objective of this simulation was to demonstrate at a proof of concept level that Delfin is a suitable model for tidal simulations in the North Sea. The results presented in the previous section indicate that, while the untuned Delfin results are generally less accurate than the highly tuned DCSM98 results, the results are sufficiently promising that it is possible to conclude that a properly tuned Delfin simulation would be likely to produce very good results indeed. Furthermore, at certain locations there is some evidence that the superior representation of coastlines provided by Delfin leads to results which surpass those achieved with the current generation of structured grid models.

Station	Observed amplitude	DCSM98 Error (%)	Delfin Error (%)
J76-57	24.5	-22.6	-25.5
J76-55	50	-25	-27.4
J76-54	66.6	-5.3	-14.7
WICK	101.1	7.8	-11.1
ABERDEEN	131.096	-2.8	-18.5
NORTHSHIELDS	158.1	-0.6	-18.6
LOWESTOFT	70.361	1.4	2.4
DOVER	225.323	6.9	-21.3
CHERBOURG	187	0.6	-16.5
DIEPPE	313	-1.5	-14.4
OOSTENDE	179.56	4.6	-14.7
HOEKVANHOLLAND	77.97	4.2	-15.2
IJMUIDEN	67.5	-2.3	-21.2
DEN HELDER	65.082	12.3	-4.2
HELGOLAND	108.6	-10.1	-27.6
FOULAISLAND	54.3	5	-6.5
PORT-MAGEE	117.5	-0.7	0.5
EKOFISK	28	-2.4	-6.2
WESTGAT	65.425	9.1	-6.9
S3	92.7	-2.2	-6.6
S1	108	-2.4	-4.4
S8	55.3	-6.3	-9.7
S14	112.4	-1	2.7
C3	115.7	-3.5	-2.7
C5	111.6	-4.3	-5.5
STMARYS	176.5	0.3	-8.2
MALINHEAD	107.5	7.6	17.5
HANSTHOLM	12.2	59.5	100.0
STATIONK13A	53.225	11.2	0.5
STATIONEURO0	72.937	-2.1	-21.4
ROSCOFF	269	-7.9	-12.3
HOLYHEAD	181	3.5	-5.1
HARLINGEN	82.984	-8.2	-5.4
DELFZIJL	134.715	-3	-33.1
VLISSINGEN	174.212	2.3	-23.1
WEST-TERSCHELLING	79.797	-7	-8.8

Table 9.1: M2 tidal amplitude observed at stations around the North Sea and the relative error for DCSM98 and Delfin.

Station	Observed phase	DCSM98 Error	Delfin Error
J76-57	286	-5.1	-2.3
J76-55	318	-0.9	0.2
J76-54	323	-3.8	-1.4
WICK	322.2	-0.7	7.2
ABERDEEN	25.1	1.2	-0.7
NORTHSHIELDS	89	-0.4	-2.4
LOWESTOFT	259.9	-6.2	-24
DOVER	331.9	0.7	-11.6
CHERBOURG	230	1.9	-1.6
DIEPPE	310	0.6	-5.6
OOSTENDE	5.3	-2.5	-22.3
HOEKVANHOLLAND	56.4	2.2	-10.9
IJMUIDEN	100.3	1.6	-10.9
DEN HELDER	162.4	-6.3	-9.8
HELGOLAND	312	-1.4	-18.7
FOULAISLAND	264	-5	-6.5
PORT-MAGEE	123	-1.3	-1.3
EKOFISK	85	-5.2	-10.9
WESTGAT	148.6	0.2	-2
S3	205	-0.6	-1.1
S1	179	-2.9	-1.9
S8	283	-0.6	-2.4
S14	163	-2.2	-2.1
C3	107	-0.4	0.5
C5	116	-1.9	-1.8
STMARYS	130.1	1.2	2.2
MALINHEAD	177.7	-7.8	-14.4
HANSTHOLM	105	-17	-34.7
STATIONK13A	176.6	4	-1.7
STATIONEURO0	25.7	-0.9	-17
ROSCOFF	142	-5.2	-3.7
HOLYHEAD	291.9	4.4	-1.1
HARLINGEN	249.6	-2.1	-37.1
DELFZIJL	305	-0.5	-38.9
VLISSINGEN	30.5	5.4	-25.8
WEST-TERSCHELLING	219.1	15.3	-20.9

Table 9.2: M2 tidal phase observed at stations around the North Sea and the error in degrees for DCSM98 and Delfin.

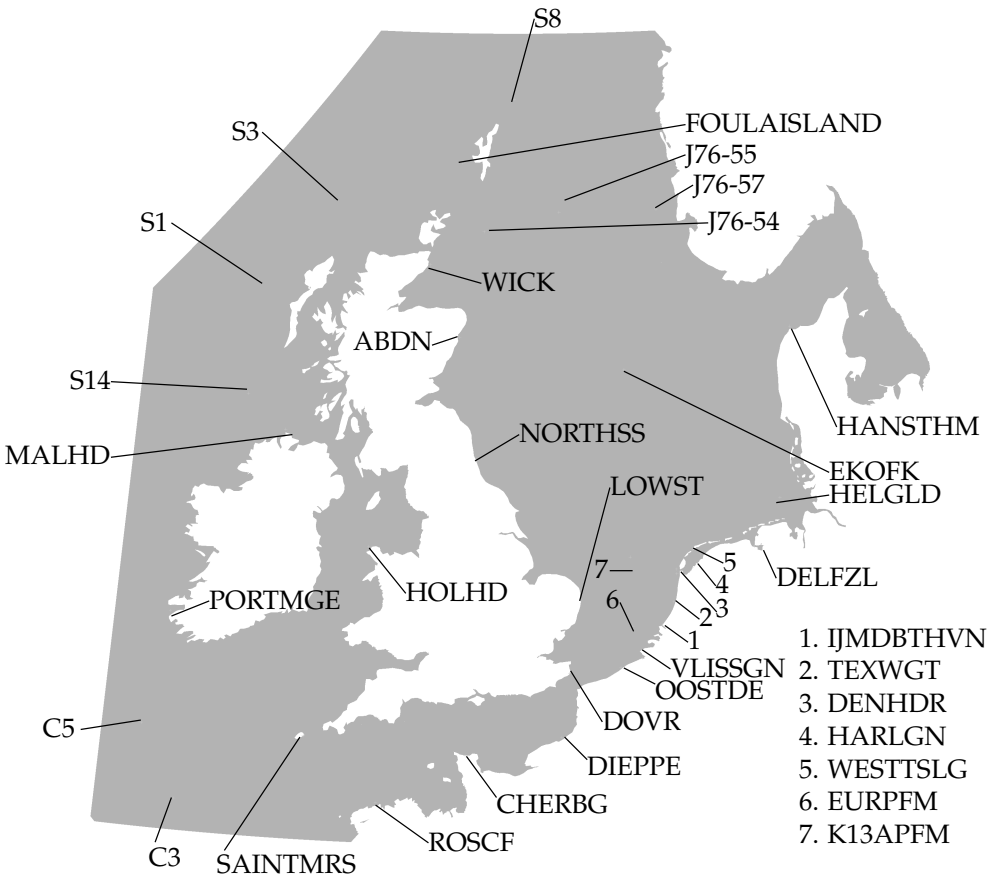

Figure 9.3: Approximate locations of the stations at which tidal observations were available for comparison.

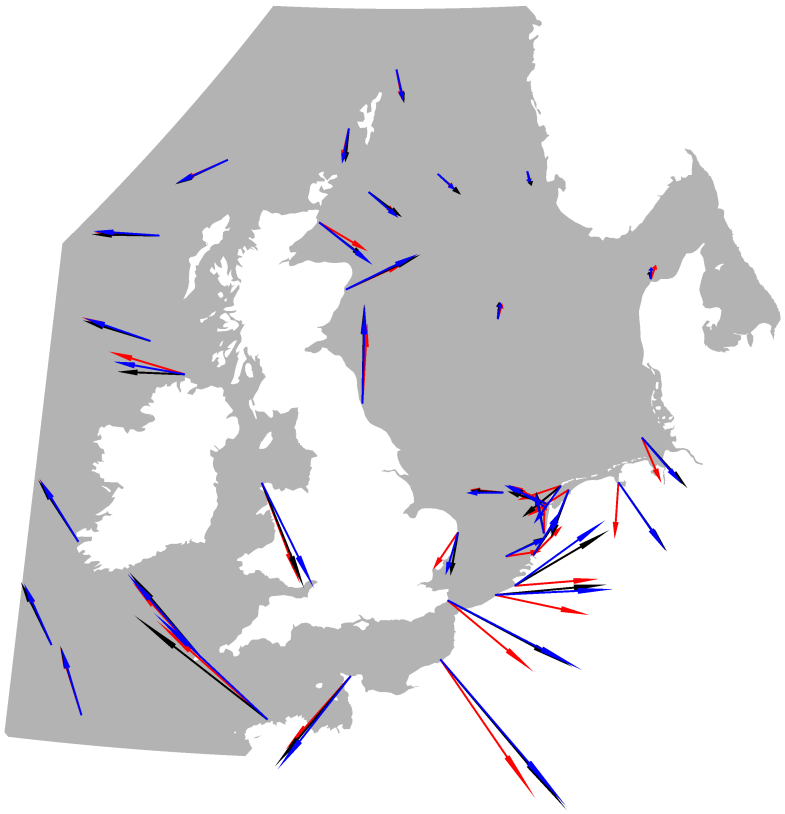

Figure 9.4: The phase and amplitude of the observed M2 tide (black), DCSM98 (blue) and Delfin (red). The length of the arrows gives the amplitude while the phase is the angle measured anticlockwise from the x axis.

Chapter 10

Conclusions

We now return to the dual purposes of this research project as laid out in the introduction to this thesis. In the first instance, this project is an attempt to produce an unstructured C grid scheme with improved numerics. The other important goal was to produce a model, Delfin, which can act as a base both for further numerical research and as a tool for investigating the physics of coastal and ocean flows.

10.1 Numerical Developments

Delfin differs from other Casulli and Walters class models in three significant numerical details. In each of these cases, it is suggested that the Delfin approach marks an improvement on what has gone before.

The path integral pressure gradient term was adopted from (Wenneker, 2002) and applied here to this class of models for the first time in chapter 5. It has been demonstrated here that this is an effective way of removing the mesh orthogonality constraint which applies to other similar models. This in effect reduces the demands which the model places on grid generation software. this is not only a practical advantage, it is also philosophically pleasing in that it moves the model a step closer to answering the question posed rather than specifying which questions may be posed. However, it is accepted that this modification does result in a discretised system with less desirable properties. Indeed, in the presence of significant Coriolis forcing, this limitation may become overwhelming. It was also shown that, even in the absence of Coriolis, the risk of the generation of spurious pressure modes remains, although the presence of any dissipation in the scheme appears to damp this instability.

Throughout this work, the θ scheme has been used as the primary timestepping scheme. In sompe places this has been compared with the Hansen scheme, an explicit timestepping method. The key result here is that for the large scale but short duration tsunami problem addressed in chapter 8, the Hansen scheme was not significantly faster than the θ scheme. Taking into account the well

know stability limitation of explicit schemes, which was demonstrated for Delfin in section 5.6.2, it may therefore be concluded that the choice of a semi-implicit pressure scheme is the preferable one for this sort of unstructured mesh model.

In chapter 6, the problem of tracking trajectories in order to construct a semi-Lagrangian advection scheme was considered. The solution proposed was to construct an interpolating velocity field which readily admits analytic integration in time. By doing so, the difficulties caused by the inherent error in a numeric integration scheme are avoided. in particular, the scheme constructed is free of trajectories which cross each other or closed boundaries. The numerical tests of the advection scheme indicate that, as with all first order upwind schemes, there is a degree of artificial dissipation caused by the scheme. On the other hand, the results of chapters 8 and 9 appear to indicate that, for long wave problems at least, the advection scheme does not significantly impair the solution. Nonetheless, it is not possible on the basis of the work conducted here to exclude the possibility that this sort of scheme will prove unsuitable for some problems for which Delfin would otherwise be a good choice of model.

The instability which was observed in the Coriolis term illustrates the difficulties which can occur when one strays outside the relative paradise of symmetry and orthogonality offered by a structured grid. The primary new contributions that this work makes are the rigorous derivation of a symmetry preserving and stable discrete Coriolis operator for schemes utilising the orthogonal mesh pressure gradient formulation, and the demonstration that the lack of symmetry in the path integral pressure gradient operator results in significant instabilities in the presence of Coriolis.

10.2 Delfin as a tool for ocean science

The two application chapters, chapters 8 and 9 represent the first attempts to use Delfin to answer questions about ocean and coastal dynamics. In particular, the results documented in chapter 8 indicate that Delfin was able to reproduce the important observed features of the Indian Ocean tsunami. Furthermore, the level of reliability and precision which was achieved enables useful distinctions to be drawn between the initial fields supplied but also supports the validity of that initial data. In this way, Delfin both adds weight to the proposition that GPS data may be an effective alternative to conventional seismic data in recovering tsunami-generating ocean bed displacements and also provides a constructive critique of the GPS solutions provided.

The results of the simulations of the North Sea presented in chapter 9 are much more preliminary than the tsunami results so it is much more difficult to draw definitive conclusions. Nonetheless, the ability to conduct tidal simulations in coastal seas such as the North Sea is important and it is therefore significant that Delfin was able to capture the essential characteristics of the M2 tide without calibration. This holds out the hope for the future that a fully calibrated simulation of the North Sea out to the edge of the continental shelf will provide very accurate results. The North Sea simulation also provides the plainest of

evidence in favour of an unstructured mesh approach. The difference between the unstructured mesh used in the Delfin simulations and the structured mesh used in DCSM98 is stark and the superiority of the unstructured mesh in representing the complex coastal geometry is clear.

10.3 Recommendations for future work

The possible future directions for this research fall into two broad categories. First, there are potential improvements which might be made in the schemes presented here and second, there are additional terms and techniques for which schemes might be chosen and implemented in order to increase the capabilities of the model.

The analysis of Coriolis discretisations presented here at the very least raises serious questions about the viability of the path integral formulation of the pressure gradient operator. The question of whether a centroid based finite difference operator is preferable to, for example, a finite element scheme based on Raviart-Thomas elements remains open.

As noted above, it is not clear that a streamline upwinded advection scheme is always the best choice. It would therefore be useful to investigate the performance of Delfin in problems where the advection is of particular importance. Constructing alternative advection schemes which preserve the block diagonal momentum matrix is a particularly challenging opportunity.

Where the path integral pressure gradient term is suitable, it produces a model which is more general than it would otherwise be but at the cost of a linear system which is, at least in some senses, harder to solve. Although success has been had with the generalised conjugate gradients method, the question remains as to whether a more modern scheme and/or an appropriate preconditioner might be more efficient.

As an alternative or as a supplement to a more efficient matrix solver, additional modelling capability would also be realised by the creation of a parallel version of the model. That the existing sparse matrix library already has a parallel version reduces the amount of non-research programming which would be necessary to achieve this goal. Parallelisation will particularly aid the running of the model with a large number of layers in the vertical since the addition of layers is an almost embarrassingly parallel problem.

A key missing feature of Delfin is scalar transport. This is a prerequisite for any sort of density current simulation as well as for many forms of turbulence model. Once an effective transport scheme was in place, either a pressure gradient formulation which takes density into account or the incorporation of turbulence modelling would be logical next steps.

Along with density currents, non-hydrostatic effects form an important class of three dimensional phenomena which may occur in a coastal or ocean context. It may be possible to incorporate non-hydrostatic correction terms, possibly in the manner proposed in Stelling and Zijlema (2003), and hence to simulate these phenomena.

Currently, Delfin incorporates a z-layer vertical discretisation. There is no inherent reason why a σ-, isopycnal or hybrid coordinate system should not be used instead, and the layer update mechanism in Delfin has been designed to facilitate this. There are clearly problems for which another vertical coordinate system would be more appropriate and the model could be expanded to accommodate this.

A research project quite distinct from the scheme expansion and improvement activity suggested above would be to development of a data assimilation capability for Delfin. This capability is particularly important if Delfin is to be calibrated to produce highly accurate results for regions such as the North Sea.

Finally, and most radically, it would be arrogant and foolhardy in the extreme to suggest that this scheme is necessarily the best one which may be implemented on a mesh with this or a similar placement of variables. Delfin has therefore been designed to facilitate the gutting of the numerics partially or completely while leaving the supporting data management routines, which comprise most of the lines of code, intact. A future research project which developed and implemented a very different approach to that presented here would therefore still stand to benefit from this work.

Appendix A

Some vector calculus

In this appendix the vector calculus notation used in the main body of the thesis is introduced. The Eulerian representation of the material derivative is derived and a proof of the Leibniz integration rule used in section 2.3 is given. It is assumed that the reader is familiar with single variable calculus and the definition and basic properties of the partial derivative.

The operators in this appendix are given in terms of the Cartesian coordinate system. It is important, however, to note that the operators remain unchanged in any orthonormal coordinate system, although the scalar partial derivatives which are their components will change with the basis.

A.1 Product operators

Throughout this thesis the Euclidean inner product, also known as the dot product or scalar product is denoted by the symbol $\cdot$. In Cartesian coordinates it is written:

$$\mathbf{a} \cdot \mathbf{b} = \mathbf{a}_x \mathbf{b}_x + \mathbf{a}_y \mathbf{b}_y + \mathbf{a}_z \mathbf{b}_z \tag{A.1}$$

where the subscripts x, y and z denote the relevant components of the vectors. The cross, or vector product is written $\times$ and is written:

$$\mathbf{a} \times \mathbf{b} = [\mathbf{a}_y \mathbf{b}_z - \mathbf{a}_z \mathbf{b}_y,\ \mathbf{a}_z \mathbf{b}_x - \mathbf{a}_x \mathbf{b}_z,\ \mathbf{a}_x \mathbf{b}_y - \mathbf{a}_y \mathbf{b}_x] \tag{A.2}$$

While the dot product is applicable in any number of dimensions, the cross product is peculiar to $\mathbb{R}^3$. There is, however a somewhat similar operator on $\mathbb{R}^2$ which is used from time to time:

$$\mathbf{a} \times \mathbf{b} = \mathbf{a}_x \mathbf{b}_y - \mathbf{a}_y \mathbf{b}_x \tag{A.3}$$

A.2 The Nabla operator

The notation of vector calculus revolves around the differential operator ∇. In Cartesian coordinates, this is written as follows:

$$\nabla = \left[\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}\right] \tag{A.4}$$

Some of the fields encountered in fluid dynamics are two dimensional. In this case we write ∇_{xy} to indicate the two dimensional gradient operator whose decomposition in Cartesian coordinates is:

$$\nabla_{xy} = \left[\frac{\partial}{\partial x}, \frac{\partial}{\partial y}\right] \tag{A.5}$$

Using the ∇ operator we may define the basic forms of vector derivative. If f is a scalar field over $\mathbb{R}^3$ then its *gradient* is defined to be:

$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}\right] \tag{A.6}$$

If $\mathbf{f}$ is a vector field over $\mathbb{R}^3$ then its *divergence* is:

$$\nabla \cdot \mathbf{f} = \frac{\partial \mathbf{f}_x}{\partial x} + \frac{\partial \mathbf{f}_y}{\partial y} + \frac{\partial \mathbf{f}_z}{\partial z} \tag{A.7}$$

and its *curl* is defined to be:

$$\nabla \times \mathbf{f} = \left[\frac{\partial \mathbf{f}_z}{\partial y} - \frac{\partial \mathbf{f}_z}{\partial y}, \frac{\partial \mathbf{f}_x}{\partial z} - \frac{\partial \mathbf{f}_z}{\partial x}, \frac{\partial \mathbf{f}_y}{\partial x} - \frac{\partial \mathbf{f}_x}{\partial y}\right] \tag{A.8}$$

A.3 The chain rule

The reader will be familiar with the chain rule in one dimension:

$$\frac{\mathrm{d}f(g(x))}{\mathrm{d}x} = \frac{\mathrm{d}f}{\mathrm{d}g}\frac{\mathrm{d}g}{\mathrm{d}x} \tag{A.9}$$

In higher dimensions there are two complications which arise. The first is that there may be two or more function arguments ($f(a(x), b(x))$). Second, a function argument may depend on one or more independent variables which also appear in the main function ($f(a(x), x)$). This latter case also introduces a difficulty of notation: If the partial derivative of f with respect to x is calculated, is a allowed to vary or is its value held constant? To resolve this ambiguity, the following convention is adopted:

$$\frac{\partial f(a(x), x)}{\partial x} = \lim_{\Delta x \to 0} \frac{f(a(x + \Delta x), x + \Delta x) - f(a(x), x)}{\Delta x} \tag{A.10}$$

$$\left.\frac{\partial f(a(x), x)}{\partial x}\right|_a = \lim_{\Delta x \to 0} \frac{f(a(x), x + \Delta x) - f(a(x), x)}{\Delta x} \tag{A.11}$$

That is to say, an unadorned partial derivative is evaluated by allowing the relevant variable to vary wherever it appears in the expression while a subscript indicates that the specified function(s) are to be held constant while calculating the derivative. Note that this notation differs from that used elsewhere in the thesis where a subscript indicates the point at which a function is to be evaluated.

We will now attempt to differentiate $f(a(x), b(x))$ with respect to x. Let $f : \mathbb{R}^2 \to r$ be continuous and differentiable in both variables and let $a, b : \mathbb{R} \to \mathbb{R}$ be continuous and differentiable. Then using the definition of the partial derivative:

$$\begin{aligned}\frac{\partial f(a(x),b(x))}{\partial x} &= \lim_{\Delta x \to 0} \frac{f(a(x+\Delta x), b(x+\Delta x)) - f(a(x), b(x))}{\Delta x} \\ &= \lim_{\Delta x \to 0} \frac{f(a(x+\Delta x), b(x+\Delta x)) - f(a(x), b(x+\Delta x))}{\Delta x} \\ &\quad + \lim_{\Delta x \to 0} \frac{f(a(x), b(x+\Delta x)) - f(a(x), b(x))}{\Delta x}\end{aligned} \tag{A.12}$$

Then by applying the one dimensional chain rule to each of the final terms we find:

$$\frac{\partial f(a(x),b(x))}{\partial x} = \frac{\partial f}{\partial a}\frac{\partial a}{\partial x} + \frac{\partial f}{\partial b}\frac{\partial b}{\partial x} \tag{A.13}$$

If we choose $b(x) = x$ then it is apparent that:

$$\frac{\partial f(a(x),x)}{\partial x} = \frac{\partial f}{\partial a}\frac{\partial a}{\partial x} + \left.\frac{\partial f}{\partial x}\right|_a \tag{A.14}$$

Finally as an illustration of the power of the chain rule in more dimensions, let $\mathbf{f} : \mathbb{R}^4 \to \mathbb{R}^2$ and $a, b : \mathbb{R}^2 \to \mathbb{R}$ be continuous and differentiable in all variables. Then by combining the previous results:

$$\nabla \cdot \mathbf{f}(a(x,y), b(x,y), x, y) = \nabla \cdot \mathbf{f}|_{a,b} + \frac{\partial \mathbf{f}}{\partial a} \cdot \nabla a + \frac{\partial \mathbf{f}}{\partial b} \cdot \nabla b \tag{A.15}$$

A.4 The material derivative

A function defined over a time varying flow field may be specified with respect either to a fixed, or Eulerian, frame of reference or a flow following, or Lagrangian frame of reference. If $f(\mathbf{x}, t)$ is a function in the Eulerian reference frame and $\mathbf{q}(t) = [x(t), y(t), z(t)]$ gives the position of a Lagrangian fluid element then $f(\mathbf{q}(t), t)$ is the same function in the Lagrangian frame of reference. The velocity of the Lagrangian element is given by:

$$\begin{aligned}\mathbf{u}(t) &= \frac{\partial \mathbf{q}}{\partial t} \\ &= \left[\frac{\partial x}{\partial t}, \frac{\partial y}{\partial t}, \frac{\partial z}{\partial t}\right]\end{aligned} \tag{A.16}$$

The material derivative, also known as the total derivative or Lagrangian derivative, is the rate of change of a quantity at a given Lagrangian fluid element. Using D to designate this derivative, and applying the chain rule for differentiation we find:

$$\begin{aligned}\frac{\mathrm{D}f}{\mathrm{D}t} &= \frac{\partial f}{\partial t} + \frac{\partial f}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial f}{\partial y}\frac{\partial y}{\partial t} + \frac{\partial f}{\partial z}\frac{\partial z}{\partial t} \\ &= \frac{\partial f}{\partial t} + \mathbf{u}\cdot\nabla f\end{aligned} \tag{A.17}$$

A.5 The Leibniz integral rule

The derivation of the depth integrated equations presented in section 2.3 requires the evaluation of a definite integral whose limits very in space. The Leibniz integral rule provides a mechanism for the evaluation of such integrals. Here we present a proof of that theorem along similar lines to the proof presented in Franklin (1955). The rule is stated in the form in which it is used in equation (2.11).

Let $\mathbf{u}_{xy} : \mathbb{R}^3 \to \mathbb{R}^2$ and $a, b : \mathbb{R}^2 \to \mathbb{R}$ be continuous and differentiable in all arguments. Then define:

$$\mathrm{F} = \int_{a(x,y)}^{b(x,y)} \mathbf{u}_{xy}(x,y,z)dz \tag{A.18}$$

Then by the chain rule:

$$\nabla_{xy}\mathrm{F} = \nabla_{xy}\mathrm{F}|_{a,b} + \frac{\partial \mathrm{F}}{\partial a}\cdot\nabla a + \frac{\partial \mathrm{F}}{\partial b}\cdot\nabla b \tag{A.19}$$

Since the limits of integration are now fixed:

$$\nabla_{xy}\mathrm{F}|_{a,b} = \int_a^b \nabla_{xy}\cdot\mathbf{u}_{xy}dz \tag{A.20}$$

Furthermore, by the fundamental theorem of calculus:

$$\frac{\partial \mathrm{F}}{\partial a} = -\mathbf{u}_{xy}(x,y,a) \tag{A.21}$$

$$\frac{\partial \mathrm{F}}{\partial b} = \mathbf{u}_{xy}(x,y,b) \tag{A.22}$$

Therefore:

$$\nabla_{xy}\cdot\int_a^b \mathbf{u}_{xy}(x,y,z)dz = \int_a^b \nabla_{xy}\cdot\mathbf{u}_{xy}dz - \mathbf{u}_{xy}(x,y,a) + \mathbf{u}_{xy}(x,y,b) \tag{A.23}$$

which is the Leibniz integral rule in two dimensional divergence form.

Bibliography

Adcroft, A., Hill, C., Marshall, J., 1999. A new treatment of the Coriolis terms in C-grid models at both high and low resolution. Monthly Weather Review 127, 1928–1936.

Ammon, C. J., Ji, C., Thio, H.-K., Robinson, D., Ni, S., Hjorleifsdottir, V., Kanamori, H., Lay, T., Das, S., Helmberger, D., Ichinose, G., Polet, J., Wald, D., 2005. Rupture process of the 2004 Sumatra-Andaman earthquake. Science 308, 1133–1139.

Arakawa, A., Lamb, V., 1977. Computational design of the basic dynamical processes of the UCLA general circulation model. Methods in Computational Physics 17, 174–267.

Batchelor, G. K., 1967. An introduction to fluid dynamics. University Press, Cambridge.

van Beek, P., van Nooyen, R., Wesseling, P., 1995. Accurate discretisation of gradients on non-uniform curvilinear staggered grids. Journal of Computational Physics 117, 364–367.

Bilham, R., May 2005. A flying start, then a slow slip. Science 308, 1126–1127.

Bollobas, B., 1979. Graph theory : an introductory course. Springer Verlag, New York, graduate texts in mathematics ; 63.

Booij, R., 2002. Modelling of the secondary flow structure in river bends. In: Proceedings of the 5th International Conference on Fluvial Hydrodynamics. Louvain-la-Neuve, Belgium, pp. 127–133.

Casulli, V., Walters, R. A., 2000. An unstructured grid, three-dimensional model based on the shallow water equations. Interational Journal for Numerical Methods in Fluids 32, 331–348.

Chaffey, J., Greenberg, D., 2003. Resolute: a semi-automated finite element mesh generation routine. Canadian Technical Report of Hydrography and Ocean Sciences 225. Bedford Institute of Oceanography.

Concus, P., Golub, G., 1976. Computing Methods in Applied Science and Engineering Second International Symposium. Vol. 134 of Lecture Notes in Economics and Mathematical Sciences. Springer, Ch. A generalized conjugate gradient method for nonsymmetric systems of linear equations.

Courant, R., Friedrichs, K., Lewy, H., 1928. Über die partiellen Differenzengleichungen der mathematischen Physik (on the partial difference equations of mathematical physics). Mathematische Annalan 100, 32–74, English translation: IBM Journal March 1967, 215-234.

Csanady, G., 1982. Circulation in the coastal ocean. Reidel, Dordrecht.

Dunsbergen, D. W., 1994. Particle models for transport in three-dimensional flow. Ph.D. thesis, Delft University of Technology.

Espelid, T. O., Berntsen, J., Barthel, K., 2000. Conservation of energy for schemes applied to the propagation of shallow-water inertia-gravity waves in regions with varying depth. International journal for numerical methods in engineering 159, 1521–1545.

Franklin, P., 1955. A treatise on advanced calculus. Wiley, London.

Fringer, O. B., Gerritsen, M., Street, R. L., 2006. An unstructured-grid, finite-volume, nonhydrostatic, parallel coastal ocean simulator. Ocean Modelling In Press, Corrected Proof.
URL `http://dx.doi.org/10.1016/j.ocemod.2006.03.006`

GEBCO, 2003. Centenary Edition of the GEBCO Digital Atlas. published on CD-ROM on behalf of the Intergovernmental Oceanographic Commission and the International Hydrographic Organization as part of the General Bathymetric Chart of the Oceans; British Oceanographic Data Centre, Liverpool.

Gebraad, A., Philippart, M., 1998. The Dutch continental shelf model, dcsm98: calibration using altimeter data. Tech. Rep. RIKZ/OS-98.121x, Rijksinstituut voor Kust en Zee.

Gill, A. E., 1982. Atmosphere-Ocean Dynamics. Vol. 30 of International Geophysics Series. Academic Press.

Ham, D. A., Kramer, S. C., Stelling, G. S., Pietrzak, J., In press. The symmetry and stability of unstructured mesh C-grid shallow water models under the influence of Coriolis. Ocean Modelling.

Ham, D. A., Pietrzak, J., Stelling, G. S., 2005. A scalable unstructured grid 3-dimensional finite volume model for the shallow water equations. Ocean Modelling 10, 153–169.

Ham, D. A., Pietrzak, J., Stelling, G. S., 2006. A streamline tracking algorithm for semi-lagrangian advection schemes based on the analytic integration of the velocity field. Journal of Computational and Applied Mathematics 192, 168–174.

Hansen, W., 1956. Theorie zur Errechnung des Waterstandes und der Strömingen in Randmeeren nebst Anwendungen. Tellus 8, 289–300.

Kramer, S., Stelling, G., 2006. A conservative unstructured scheme for rapidly varied shallow water flows. In preparation.

Krüger, F., Ohrnberger, M., 2005. Tracking the rupture of the $m_{\mathrm{w}} = 9.3$ Sumatra earthquake over 1,150km at teleseismic distance. Nature 435, 937–939.

Labeur, R. J., Pietrzak, J. D., 2005. A fully three dimensional unstructured grid non-hydrostatic finite element coastal model. Ocean Modelling 10, 51–67.

Lay, T., Kanamori, H., Ammon, C. J., Nettles, M., Ward, S. N., Aster, R. C., Beck, S. L., Bilek, S. L., Brudzinski, M. R., Butler, R., DeShon, H. R., Ekström, G., Satake, K., Sipkin, S., May 2005. The great Sumatra-Andaman earthquake of 26 december 2004. Science 308, 1127–1133.

Le Roux, D., Séne, A., Rostand, V., Hanert, E., 2005. On some spurious mode issues in shallow-water models using a linear algebra approach. Ocean Modelling 10, 83–94.

Le Roux, D. Y., 2001. A new triangular finite-element with optimum constraint ratio for compressible fluids. SIAM Journal on Scientific Computing 23 (1), 66–80.

Legrand, S., Legat, V., Deleersnijder, E., 2000. Delaunay mesh generation for an unstructured-grid ocean general circulation model. Ocean Modelling 2, 17–28.

Ménard, Y., Fu, L.-L., Escudier, P., Parisot, F., Perbos, J., Vincent, P., Desai, S., Haines, B., Kunstmann, G., 2003. The jason-1 mission. Marine Geodesy 26, 131–146.

Miglio, E., Quarteroni, A., Saleri, F., 1999. Finite element approximation of quasi-3d shallow water equations. Computer methods in applied science and engineering 174, 355–369.

Neetu, S., Suresh, I., Shankar, R., Shankar, D., Shenoi, S., Shetye, S., Sundar, D., Nagarajan, B., 2005. Comment on "the great sumatra-andaman earthquake of 26 december 2004". Science 310, 1431a.

Nicolaides, R., 1993. The covolume approach to computing incompressible flows. In: Gunzberger, M., Nicolaides, R. (Eds.), Incompressible Computational Fluid Dynamics. Cambridge University Press, Cambridge, pp. 295–333.

Oliveira, A., Baptista, A. M., 1998. On the role of tracking on Eulerian–Lagrangian solutions of the transport equation. Advances in Water Resources 21, 539–554.

Peaceman, D., Rachford, H.H., J., March 1955. The numerical solution of parabolic and elliptic differential equations. Journal of the Society for Industrial and Applied Mathematics 3 (1), 28–41.

Perot, B., 2000. Conservation properties of unstructured staggered mesh schemes. Journal of Computational Physics 159, 58–89.

Pietrzak, J., Jakobson, J. B., Buchard, H., Vested, H. J., Petersen, O., 2002. A three-dimensional hydrostatic model for coastal and ocean modelling using a generalised topography following co-ordinate system. Ocean Modelling 4, 173–205.

Pietrzak, J., Socquet, A., Ham, D., Simons, W., Vigny, C., Labeur, R. J., Schrama, E., Stelling, G., Vatvani, D., 2006. Insights into the indian ocean tsunami from gps, altimeters and tsunami models. Submitted to Earth and Planetary Science Letters.

Shewchuk, J. R., May 1996. Triangle: engineering a 2D quality mesh generator and Delaunay triangulator. In: Lin, M. C., Manocha, D. (Eds.), Applied Computational Geometry: Towards Geometric Engineering. Vol. 1148 of Lecture Notes in Computer Science. Springer-Verlag, pp. 203–222, from the First ACM Workshop on Applied Computational Geometry.

Stelling, G., Duinmeijer, S., 2003. A staggered conservative scheme for every froude number in rapidly varied shallow water flows. International Journal for Numerical Methods in Fluids 43 (1329-1354).

Stelling, G., Zijlema, M., 2003. An accurate and efficient finite-difference algorithm for non-hydrostatic free-surface flow with application to wave propagation. International Journal for Numerical Methods in Fluids 43, 1–23.

Strang, G., 1988. Linear Algebra and its Applications. Harcourt Brace Jovanovich College Publishers.

Trefethen, L. N., Bau, D., 1997. Numerical linear algebra. Society for Industrial and Applied Mathematics, Philadelphia.

Veldman, A., Rinzema, K., 1992. Playing with nonuniform grids. Journal of Engineering Mathematics 26, 119–130.

Verstappen, R., Veldman, A., 1998. Spectro-consistent discretization of Navier-Stokes: a challenge to RANS and LES. Journal of Engineering Mathematics 34, 163–179.

Verstappen, R., Veldman, A., 2003. Symmetry-preserving discretization of turbulent flow. Journal of Computational Physics 187, 343–368.

Vigny, C., Simons, W., Abu, S., Bamphenyu, R., Satirapod, C., Choosakul, N., Subarya, C., Socquet, A., Omar, K., Abidin, H., Ambrosius, B., July 2005. Insight into the 2004 Sumatra-Andaman earthquake from GPS measurements in southeast Asia. Nature 436, 201–206.

Walters, R., Carey, G., 1984. Numerical noise in ocean and estuarine models. Advances in water resources 7, 15–20.

Walters, R., Casulli, V., 1998. A robust finite element model for hydrostatic surface water flows. Communications in Numerical Methods in Engineering 14, 931–940.

Wenneker, I., 2002. Computation of flows using unstructured staggered grids. Ph.D. thesis, Delft University of Technology.

Wenneker, I., Segal, A., Wesseling, P., 2002. A mach-uniform unstructured staggered grid method. International Journal for Numerical Methods in Fluids 40, 1209–1235.

Wesseling, P., 2001. Principles of Computational Fluid Dynamics. Springer.

Zhang, Y., Baptista, A., 2005. A cross-scale model for 3D baroclinic circulation in estuary-plume-shelf systems: I. formulation and skill assessment. Continental Shelf Research 24, 2187–2214.

Acknowledgements

In undertaking this work over the last five years, I have been privileged to be associated with and supported by a great number of important people in my professional and personal life. The first and greatest thanks must go to my supervisor Dr Julie Pietrzak. Julie introduced me not just to the science of ocean modelling but also to many of the fascinating and brilliant people who work in this area. Even though she always seems to pay me the unearned compliment of believing that I know a bit more oceanography than I do, it is her deep professional knowledge of ocean physics which informs the numerics choices I made in undertaking this work. I would also like to acknowledge the input of my promotor, Prof. Guus Stelling, whose knowledge of areas of numerical mathematics I didn't even know existed has been invaluable.

I firmly believe that the support of ones fellow PhD students is as important in completing a doctorate as is the quality of the supervision one receives. Fortunately, from the day that Bas, Bram, Ankie and Federico collected me from Schiphol Airport back in 2001, I was a member of a tight-knit group of PhD students whose academic and social support is one of the main reasons this thesis was completed. So thank you to Alexander, André, Ankie, Bas, Bram, Elmar, Federico, Francesca, Gerben, Harmen, Ketut, Olga, Martijn, Michel, Petra, Sander, Stephan, Tim, Volker, Walter and Wim. For most of my time in Delft I shared an office with Bas Hofland. He was a pleasure to work with and the office was a less enjoyable place to be when he had left and I had the room to myself. Stephan Kramer was not only my fellow PhD student but also my collaborator on several aspects of our research, and remains so at Imperial College. He deserves particular mention for the substantial contribution he made to the work on the Coriolis term which is presented in chapter 7 and which we have published together. In addition to TU Delft's PhD students, some of the more senior staff who really ought to have known better were a welcome and lively addition to the life of the group. Wim and Rob, your attitude and involvement are much appreciated!

I would also like to acknowledge the input of the Department of Mathematics at the Australian National University who provided an excellent education in mathematics, some of which I absorbed, and who got me sufficiently hooked that I wanted to go on to my PhD. At ANU I would like to particularly thank my honours supervisor Dr Brian Davies who unwittingly introduced me to

this fluids lark when he suggested I might be interested in a course on computational astrophysics and Dr Stephen Roberts who invited me back to visit the department when he learned that he and I were working on quite similar shallow water codes. His input into the semi-Lagrangian trajectory tracking algorithm in chapter 6 is particularly appreciated.

Outside of work I derived a great deal of pleasure in my time in Delft from my involvement in singing and debating so my thanks go out to the successive boards of Krashna Musika and to the many organisers of debates that I have attended around the world.

Finally I would like to mention some significant people in my personal life. My thanks go to my parents and sister for their ongoing support. I do not get to see them as much as I would like and I contact them less frequently than I ought. I greatly appreciate the repeated hospitality of my aunt, Monica, and Gebina's mother, Ingrid. Finally I express my great thanks to my girlfriend Gebina Harren who has made the stress of finishing so much easier to bear and whose work on the design of the gorgeous Delfin logo deserves a very special mention.

Curriculum Vitae

David Ham was born in Canberra, Australia in 1976. He lived in Guildford in England during his primary school years but attended secondary school at Canberra High School and Lake Ginninderra College in Canberra. Beside his studies David played trumpet, violin and viola in school bands and youth orchestras, debated for his school and was a keen scout. In 1994 he received the Queen's Scout Award.

After a gap year in which he worked as an outdoor activity instructor in England he studied mathematics and law at the Australian National University. In 2001 he graduated as a bachelor of science with first class honours in mathematics with a thesis entitled "Control of chaotic systems and fractal basin boundaries". In 2003 he graduated as a bachelor of laws having completed his studies part time while working on his PhD. He spent 1999 studying mathematics and comparative law at Uppsala University in Sweden.

In 2001 David commenced work on his PhD in numerical flow model development in the Fluid Mechanics Section of the Faculty of Civil Engineering and Geosciences of Delft University of Technology. The results of his work are this thesis and four refereed publications, three of which have already been accepted or published. In October 2005 he was invited to present the results of his doctoral work at the fourth Physical Oceanography Dissertation Seminar in Hawaii. He is currently a research associate at Imperial College London where he is a member of the Imperial College Ocean Model development team.

Alongside his scientific work, David is a choral singer and has sung in university choral societies in four different countries. He debated and judged on the international university circuit for several years and has served as the registrar of the European and World University Debating Councils.

6. De klacht van overheid en bedrijfsleven over de afname van het aantal studenten in exacte vakken is hypocriet, als men te weinig doet om het rendement van een exacte studie meer in overeenstemming te brengen met de inspanning.

7. De huidige betaalwijze van "pay to read" voor artikelen van elektronisch verspreide wetenschappelijke tijdschriften heeft meer nadelen dan voordelen.

8. Hoe hoog het niveau van de vakgroep waar men afstudeert ook is, het is altijd een voordeel om ergens anders te promoveren.

9. De bepaling dat houders van buitenlandse academische graden slechts in bijzondere gevallen tot promotie aan een Nederlandse universiteit worden toegelaten is een blijk van onaanvaardbaar chauvinisme aan de kant van de universiteiten en de overheid.

   — WHW Art.7.18, Promotiereglement TU Delft Art.2

10. Het fundamentele verschil tussen geloof in de wetenschap en geloof in een religie is dat er voor de eerste geen geloofsaanname nodig is.

Deze stellingen worden opponeerbaar en verdedigbaar geacht en zijn als zodanig goedgekeurd door de promotor prof. dr ir G.S. Stelling

PROPOSITIONS
accompanying the thesis

On techniques for modelling coastal and ocean flow with unstructured meshes

by David Ham

1. The ability to focus resolution on objects of interest will make unstructured meshes the technology of choice for future models of ocean and coastal flow.

   — This thesis and C.Pain et al. Three-dimensional unstructured mesh ocean modelling, Ocean Modelling 10 (2005) 5-33.

2. Unstructured C grid flow models are subject to an instability caused by the discretisation of the pressure gradient and Coriolis terms which may be removed by appropriate symmetry-preserving discretisations of those terms.

   — This thesis and D. Ham et al. The symmetry and stability of unstructured mesh C-grid shallow water models under the influence of Coriolis, Ocean Modelling in press.

3. The trajectories employed in a semi-Lagrangian advection scheme have advantageous physical properties if they are generated by the analytic integration of a divergence free velocity field.

   — This thesis and D. Ham et al. A streamline tracking algorithm for semi-Lagrangian advection schemes based on the analytic integration of the velocity field, Journal of Computational and applied mathematics 192 (2006) 168-174.

4. Unstructured C grid shallow water simulations constitute an effective method for testing the efficacy of fixed GPS stations as a mechanism for deriving the ocean bed displacements which generate tsunamis.

   — This thesis and Pietrzak et al. Insights into the Indian Ocean Tsunami from GPS, altimeters and tsunami models, submitted to Earth and Planetary Science Letters.

5. The goals of science are best served by the release of software developed in the course of research as free software.

6. The complaints voiced by government and business about the falling numbers of science students are hypocritical if they are not matched by action to make the returns on studying science better match the effort involved.

7. With the advent of electronic distribution of journal articles, the current pay-to-read business model does more harm than good.

8. No matter how high the quality of the department from which one graduates, it is always advantageous to move elsewhere to undertake a PhD.

9. The stipulation that holders of foreign degrees may be admitted as doctoral candidates at Dutch universities only in exceptional circumstances is an act of unacceptable chauvinism on the part of the universities and the government.

   — WHW Art.7.18, TU Delft Doctorate Regulations Art.2

10. The fundamental difference between belief in science and belief in religion is that the former requires nothing to be taken on faith.

These propositions are considered opposable and defendable and as such have been approved by the supervisor prof. dr ir G.S. Stelling

STELLINGEN
behorende bij het proefschrift

On techniques for modelling coastal and ocean flow with unstructured meshes

van David Ham

1. Het voordeel van hoge resolutie, uitsluitend in het aandachtsgebied, maakt dat vooral ongestructureerde roosters gekozen zullen worden voor toekomstige kust- en oceaanstromingsmodellen.

   — Dit proefschrift en C.Pain et al. Three-dimensional unstructured mesh ocean modelling, Ocean Modelling 10 (2005) 5-33.

2. De instabiliteit, bij versprongen roosters, van de ruimtelijke discretisatie van de drukgradiënt in combinatie met de Coriolisterm, kan opgelost worden door op de juiste wijze deze termen symmetrisch te discretiseren.

   — Dit proefschrift en D. Ham et al. The symmetry and stability of unstructured mesh C-grid shallow water models under the influence of Coriolis, Ocean Modelling in press.

3. De stroomlijnen toegepast in een semi-Lagrangiaans advectieschema hebben voordelige fysische eigenschappen als ze zijn gegenereerd door middel van analytische integratie van een divergentievrij snelheidsveld.

   — Dit proefschrift en D. Ham et al. A streamline tracking algorithm for semi-Lagrangian advection schemes based on the analytic integration of the velocity field, Journal of Computational and applied mathematics 192 (2006) 168-174.

4. Ondiepwatersimulaties van tsunami's met een ongestructureerd C-rooster zijn een effectieve methode om de doeltreffendheid te testen waarmee vaste GPS installaties de onderliggende bodemverplaatsingen bepalen.

   — Dit proefschrift en Pietrzak et al. Insights into the Indian Ocean Tsunami from GPS, altimeters and tsunami models, submitted to Earth and Planetary Science Letters.

5. De vooruitgang in de wetenschap is er bij gebaat als software die ontwikkeld is in het kader van openbaar wetenschappelijk onderzoek, als vrije software beschikbaar is.