A Data Assimilation Scheme for the One-dimensional Shallow Water Equations

A Data Assimilation Scheme for the One-dimensional Shallow Water Equations

By Ramsha Khan,

A Thesis Submitted to the School of Graduate Studies in the Partial Fulfillment of the Requirements for the Degree Masters of Science

McMaster University

Masters of Science (2016)

Hamilton, Ontario (Department of Mathematics and Statistics)

TITLE: A Data Assimilation Scheme for the One-dimensional Shallow Water Equations

AUTHOR: Ramsha Khan (McMaster University)

SUPERVISOR: Dr. Nicholas Kevlahan

NUMBER OF PAGES: x, 96

Lay Abstract

In ocean wave modelling, information on the system dynamics and full initial and/or boundary data is required. When the latter is not fully available the primary objective is to find an optimal estimate of these conditions, using available information. Data Assimilation is a methodology used to optimally integrate observed measurements into a mathematical model, to generate a better estimate of some control parameter, such as the initial condition of the wave, or the sea floor bathymetry. In this study, we considered the shallow water equations in both linear and non-linear form as an approximation for ocean wave propagation, and derived a data assimilation scheme to optimise some distorted form of the initial condition to generate predictions converging to the exact initial data. The error between measurements and observation data was sufficiently minimised across all cases. A relationship was found between the number of measurement points and the error, dependent on the choice of where measurements were taken.

Abstract

For accurate prediction of tsunami wave propagation, information on the system of PDEs modelling its evolution and full initial and/or boundary data is required. However the latter is not generally fully available, and so the primary objective becomes to find an optimal estimate of these conditions, using available information. Data Assimilation is a methodology used to optimally integrate observed measurements into a mathematical model, to generate a better estimate of some control parameter, such as the initial condition of the wave, or the sea floor bathymetry. In this study, we considered the shallow water equations in both linear and non-linear form as an approximation for ocean wave propagation, and derived a data assimilation scheme based on the calculus of variations, the purpose of which is to optimise some distorted form of the initial condition to give a prediction closer to the exact initial data. We considered two possible forms of distortion, by adding noise to our initial wave, and by rescaling the wave amplitude. Multiple cases were analysed, with observations measured at different points in our spatial domain, as well as variations in the number of observation points. We found that the error between measurements and observation data was sufficiently minimised across all cases. A relationship was found between the number of measurement points and the error, dependent on the choice of where measurements were taken. In the linear case, since the wave form simply translates a fixed form, multiple measurement points did not necessarily provide more information. In the nonlinear case, because the waveform changes shape as it translates, adding more measurement points provides more information about the dynamics and the wave shape. This is reflected in the fact that in the nonlinear case adding more points gave a bigger decrease in error, and much closer convergence of the optimised guess for our initial condition to the exact initial wave profile.

Acknowledgements

Fore-mostly I would like to thank my supervisor, Dr. Nicholas Kevlahan, for going above and beyond in his support during the completion of this research, and for taking the time to make sure I understood even the most basic of concepts. I would like to thank my committee members Dr. Stanley Alama and Dr. Bartosz Protas for their constructive feedback, and especially to Dr. Protas for feedback which upon addressing, greatly improved the results of this study. I would like to give a special thanks to Tyler Meadows, Alexander Chernyavsky, Niky Hristov and Lee Van Brussels for all their help in helping me wrestle with maths these last two years. To all the friends who helped me find that work-play balance so necessary for graduate school, I owe a lot. And last, but assuredly not least, to my parents without whom I categorically would not have made it this far.

Contents

Lay Abstract iii

Abstract iv

Acknowledgements v

Declaration of Authorship x

1 Introduction 1

2 Derivation of the Shallow Water Equations 6
2.1 Shallow Water Equations . . . 9
2.1.1 Conservation of Mass from First Principles . . . 9
2.1.2 Balance of Momentum . . . 12
2.2 Dissipative Forms of the Shallow Water Equations . . . 14
2.2.1 Viscosity . . . 14
2.2.2 Constant Density Flow (Homogeneous Flows) . . . 15
2.2.3 Rotation . . . 16
2.3 Hydrostatic & Geostrophic Balance . . . 17
2.3.1 Hydrostatic Balance . . . 17
2.3.2 Geostrophic Balance . . . 19
2.4 Waves in Shallow Water . . . 20
2.4.1 Gravity Waves . . . 21
2.4.2 Inertio-Gravity (Poincaré) Waves . . . 23
2.4.3 Kelvin Waves . . . 25
2.4.4 Rossby Waves . . . 28
2.4.5 Wave Propagation Summary . . . 31
2.5 Boussinesq Approximation . . . 32
2.6 Comparison with Shallow Water System . . . 33
2.6.1 Applications of the Boussinesq Equations . . . 35

3 Solving the One-Dimensional Shallow Water Equations Forward in Time 38
3.1 Finite Volume Approximation in Space . . . 38
3.1.1 Non-Linear System . . . 38

3.1.2 Linearised System . . . 42
3.2 Runge-Kutta Time Integration Scheme . . . 43
3.2.1 Stability . . . 44
3.2.2 Courant-Friedrichs-Lewy Condition . . . 45
3.3 Comparison with Analytical Solution . . . 46

4 Optimal Control and Data Assimilation 49
4.1 Introduction . . . 49
4.2 Optimal Control . . . 50
4.2.1 Optimal Control Example[11] . . . 52
4.3 Data Assimilation . . . 55

5 Data Assimilation for the Shallow Water Equations 59
5.1 Initial Condition as Control Variable . . . 59
5.1.1 Linear Case . . . 59
5.1.2 Non-linear Case . . . 62
5.1.3 Algorithm Summary . . . 64
5.1.4 Line Minimisation . . . 65
5.1.5 Smoothing the Cost Function Gradient . . . 66

6 Kappa Test for Gradient used in Data Assimilation Variational Scheme 67
6.1 Kappa Test . . . 67
6.2 Numerical Results . . . 68

7 Results 76
7.1 Data Assimilation for Linear Case . . . 76
7.2 Data Assimilation - Non-Linear Case . . . 82

8 Conclusions and Future Considerations 88
8.1 Conclusions . . . 88
8.2 Future Considerations . . . 90
8.2.1 Model Parameters & Grid Refinement . . . 90
8.2.2 Inverse Problem with Bathymetry Optimisation . . . 90

List of Figures

2.1 Single Fluid Layer, where $h = \eta + H$ . . . 7
2.2 Column of fluid in shallow water . . . 9
2.3 Dispersion relation for gravity waves, where $\omega = \pm\sqrt{gHK}$, where the different lines represent different values of H . . . 23
2.4 Dispersion relation for Poincaré waves: $\omega \approx f_0$ for small k, and $\omega \approx \sqrt{gh}k$ for large k . . . 25
2.5 Boundary at $y = 0$.[4] . . . 25
2.6 Dispersion relation for Kelvin waves: $\omega = \sqrt{gH}k$, with boundary at $y = 0$. 27
2.7 β-plane for varying latitude.[4] . . . 28
2.8 Dispersion relation for Rossby waves: $\omega = -\frac{\beta k}{k^2+\frac{1}{L^2{}_D}}$ . . . 30

3.1 Analytically derived height (blue) vs the numerical height calculated using forward solver (red) at x_0 over time period 0 to 2. . . . 47

6.1 $\log \kappa(\varepsilon)$ vs $\log \varepsilon$. As we can see, as ε goes to zero, κ converges to approximately 1, indicating the different gradient formulations are equivalent to a high degree, in both the linear and non-linear systems . . . 69
6.2 $|\kappa(\varepsilon) - 1|$ vs $\log \varepsilon$. This allows us to see how closely κ converges to 1: to $\approx 10^{-3}$ for the linear case, and to $\approx 10^{-1}$ for the non-linear case. . . . 70
6.3 Linear Case: Analytical vs numerical height at x_0 given distorted IC. Results for analytically derived height and the output of the numerical scheme match almost exactly. . . . 71
6.4 Linear Case: Analytical vs numerical adjoint height at t_0. Indicates our backward solver coincides very closely with the analytical solution for η^* over time. . . . 71
6.5 Difference in analytical vs numerical height at x_0 given distorted IC . . . 74
6.6 Difference in analytical vs numerical adjoint height at t_0 . . . 75

7.1 Linear Case: Cost Function normalised by number of observation points with amplitude distortion. Increasing the number of observation points (where points are chosen at any point in our domain) beyond a single point increases the error, but marginally with increasing points after that. 79

7.2 Linear Case:Cost Function normalised by number of observation points with noise distortion. Increasing the number of observation points (where points are chosen at any point in our domain) beyond a single point increases the error, but marginally with increasing points after that. . . . 79
7.3 Linear Case: Cost Function normalised by number of observation points with Amplitude Distortion, for points in full observation range. 80
7.4 Linear Case: Cost Function normalised by number of observation points with Noise Distortion, for points in full observation range. 80
7.5 The optimised initial condition with amplitude distortion has an additional smaller wave to the right, in order to minimise the error at the assimilation points when it propagates. 81
7.6 The optimised initial condition with noise has some additional noise to the right of the wave, in order to minimise the error at the assimilation points when it propagates. 81
7.7 Non-linear case: Cost function with amplitude distortion. Aside from a single assimilation point case, adding more points seems to decrease the error (where points are chosen at any point in our domain). 83
7.8 Non-linear case: Cost function with noise distortion. Adding more points decreases the error in the cost function (where points are chosen at any point in our domain). 84
7.9 Non-linear case: Cost function with amplitude distortion. Increasing the number of observation points in full observation range reduces the cost function as more points are added. 84
7.10 Non-linear case: Cost function with noise distortion. Increasing the number of observation points in full observation range reduces the cost function as more points are added. 85
7.11 The optimised and exact initial conditions match almost exactly. 86
7.12 The noise distortion has been smoothed out in the optimised initial condition to a high degree. 86
7.13 Comparison of wave profiles in linear and non-linear case. As can be seen, the non-linear wave is more steep, and does not translate a fixed form over time as in the linear case, hence adding more points decreases our overall error in the cost function in the non-linear case, whereas it does not prove to be significant in the linear case. 87

Declaration of Authorship

I, Ramsha KHAN, declare that this thesis titled, 'A Data Assimilation Scheme for the One-dimensional Shallow Water Equations ' and the work presented in it are my own.

Chapter 1

Introduction

The primary objective of this thesis is to develop data assimilation techniques to improve numerical forecasting of tsunami waves. The purpose of tsunami modelling is to accurately predict the wave height and velocity at coastal areas given initial conditions from seismic data and bathymetry. The 2004 Indian ocean tsunami as well as the 2011 Japanese tsunami have poignantly highlighted the need for a better forecasting model that can be used to create and implement evacuation and emergency protocols effectively in a limited amount of time. In the past, existing methods for prediction faced issues such as high errors in numerical approximations, extremely high dimensions for estimating various parameters, lack of accurately measured observation data as well as insufficient processing power to find results in a small time window.[14]

In this thesis, we present a concise description of the physical system we are attempting to predict and its mathematical formulation, and an overview of the methodology we utilise to do so (data assimilation). In particular, this thesis presents and implements a new variational method for improving estimates of the initial conditions, and derives the mathematical form of the assimilation problem for bathymetry. These two concepts are then combined and the various issues, considerations and results are discussed. Our

model for wave propagation uses the shallow water equations. These are a coupled system of equations for non-dispersive travelling waves where the wavelength λ is much larger than the ocean depth h from the seabed, allowing us to average over the depth and our height variable becomes $h = H + \eta$, where H is the average depth and η is the pertubation of surface. Our objective is to develop a data assimilation variational scheme for the two-dimensional problem in some subdomain of (x, t), in order to optimally estimate the initial condition for our model and so improve the accuracy of our numerical model solving the shallow water equations.

In the past, numerical simulation of tsunami waves as well as other geophyscial dynamical systems was done independently of the data used to validate the model prediction. There are shortcomings in both of these processes; the numerical simulations being subject to inaccuracies and approximation errors, and the availability of observational data suffering from technical, budgetary and physical restrictions [14]. For example, we may not have complete measurements across all variables at observation points, and the measuring process itself is subject to errors, resulting in noisy data. And so the logical solution is to take the incompleteness presented by both, and try and minimise its negative affect by combining the process of using the observed data as a correction for the model prediction dynamically, within the model itself. Data assimilation is the optimal integration of observed data into a numerical predictive model.

The outline of this study is as follows. Chapter two introduces the shallow water equations, giving their derivation from the Euler equations and the principles of conservation of mass and momentum.The assumptions we make such as hydrostatic and geostrophic balance are discussed. The various parameters and factors that can be included in different forms of the system are also introduced, such as viscosity, varying pressure and

density, rotation and the Coriolis effect. We also look at the different forms of the solutions and outline where each model is appropriate, as well as computing the dispersion relation between the wavenumber k and frequency ω. The topic of frequency dispersion also brings to light the suitability of the shallow water equations, which neglect frequency dispersion effects on wave propagation, versus the Boussinesq approximation which accounts for it. We mathematically derive the Boussinesq equations, and present a comparison with the shallow water system.

Chapter three introduces the numerical scheme for solving our systems in forward time from an initial time t_0 to a final time t_T, on a spatial grid of discrete points. Our discretisation in space employs a second-order finite difference scheme on a staggered grid, where the velocity u is computed at the edges of our cells and height h at the centres. To find our trends with respect to time, we use a third-order four stage Runge Kutta scheme. We discuss the stability of our numerical solver and the criteria necessary for it, such as the Courant-Friedrich-Lewy condition. We then show the accuracy of the numerical solver by comparing results to the analytical solution.

In Chapter four we give a brief overview of optimal control theory and the calculus of variations employed in our data assimilation model.We state the form of a typical one-dimensional optimisation problem and its components, as well as the necessary conditions required for an optimal state (a full derivation of these is not given). An optimal control example from Kirk (2004) is presented in order to illustrate the nature of the problem form, and this lays the foundation for variational data assimilation methods, introducing formulations like the cost function, first variation and conditions for minimisation, and the adjoint method for finding our cost function gradient.

Chapter five then derives our new data assimilation variational algorithm for the shallow water equations, where the initial condition is optimised at every iteration by minimising a cost function measuring the difference between the model prediction and the observational data. We begin with the linear equations, but then extend the discussion to the nonlinear case as well. We give the derivation for the cost function gradient, the corresponding adjoint equations for our shallow water model, and the necessary criteria for optimisation. Concepts like smoothing and optimal step size calculation within our algorithm are also outlined.

The aim of chapter six is to test the algorithm derived in chapter five. We do this by conducting a *Kappa test*, where the different numerical approximations for our gradient forms (such as the Gateaux derivative and the Sobolev gradient) are measured relative to each other, as well as against their formulation using the analytical solution for our shallow water and adjoint models.

The preceding chapters provide the foundation for chapter seven, which presents the numerical implementation, issues encountered and results based on differing cases. We begin by creating an 'exact' initial condition and solving our model in forward time, generating values for height at some point x_0. These serve as our observations. We then distort this exact condition, either by adding noise, or changing the amplitude or removing some points. We then run the data assimilation scheme on this distorted initial condition. Our objective is to have a final output that is as close to the 'exact' initial condition as possible. Different variations such as the choice or number of x_0 measurement points are considered, as well as the differences in outcomes between the linear and non-linear shallow water systems.

The final chapter summarises the main results of the thesis and outlines possible future research topics, that were not in this simple model. This includes numerical implementation of our bathymetry and Boussinesq models, as well as extensions to higher dimensions.

Chapter 2

Derivation of the Shallow Water Equations

Introduction

The Shallow Water Equations describe the motion of fluids and are derived from the conservation laws of mass and momentum. More specifically, they are used to study the propagation of water waves where the length scale of horizontal motion is much larger than the fluid depth ($\lambda >> h$), and the wave height is much less than the fluid depth. This allows us to average over the depth to get rid of the vertical dimension, and instead consider the displacement of the free surface as a function of the horizontal dimensions and time. The shallow water equations are useful in modelling tsunami propagation in coastal waters. Figure 1 represents our single fluid layer, where $\eta(x, y, t)$ is the displacement of the free surface, h is the total thickness of the fluid where $h = H + \eta$, and H is the averaged height from the flat bottom.

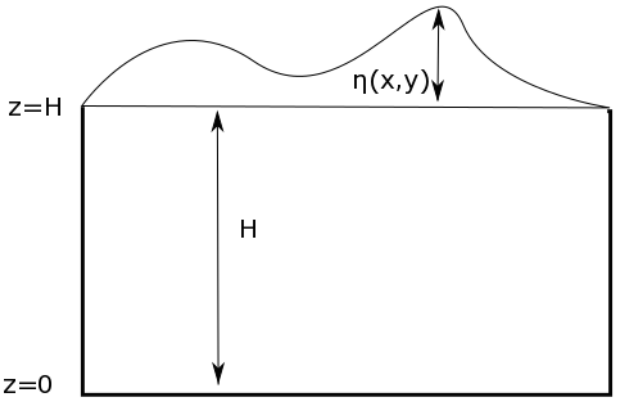

FIGURE 2.1: Single Fluid Layer, where $h = \eta + H$

The reason the condition $\lambda >> h$ is important, is that this aspect ratio satisfies the hydrostatic approximation, and hence allow us to use the hydrostatic equation $\frac{\partial p}{\partial z} = -\rho g$, where $p(x, y, z)$ is the pressure. We discuss why this approximation is appropriate in shallow water in Section 4.

To derive our shallow water equations, we begin with the Euler equations[18]. The Euler Equations describe the motion of a fluid, where $\mathbf{v}(x, y, z, t)$ is the velocity of fluid particles. These equations are derived from the conservation laws for mass and momentum. The implications of this are that no force, mass or energy is added or removed from the domain, so the rate of change of each conserved quantity (such as mass) with respect to time should equal the amount crossing the boundary of our domain Ω over time. To eventually derive the Shallow Water Equations, we need the Euler equations for mass and momentum. These are

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) \ = 0, \tag{2.1}$$

$$\frac{D\mathbf{v}}{Dt} + \ \frac{1}{\rho}\nabla(p) = 0. \tag{2.2}$$

When the fluid is incompressible, the density ρ is constant. Subsequently (1) reduces down to

$$\text{div } \mathbf{v} = \ 0 \ . \tag{2.3}$$

We use hydrostatic balance to derive the momentum equation for shallow water, and first principles to derive the conservation of mass. We then discuss the various dissipative forms for the shallow water equations, and consider the effects of viscosity, rotation and compressible flow. We then discuss the assumptions based on hydrostatic and geostrophic balance in more detail. We also introduce the Boussinesq approximation, which gives us a set of equations utilising the fact that density variation in the oceans is very small. We end with a discussion on the different wave solutions of the shallow water equations, such as Gravity waves, Inertio-gravity waves, Kelvin waves and Rossby waves.

2.1 Shallow Water Equations

2.1.1 Conservation of Mass from First Principles

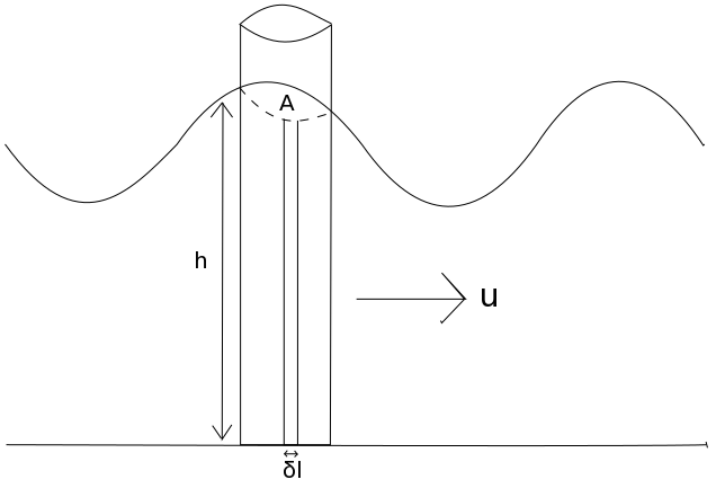

FIGURE 2.2: Column of fluid in shallow water

Consider a sub-column of our fluid in shallow water, which has a cross-sectional surface area A. Let us assume a flat-bottomed domain, in which case $h = \eta$. Then

$$\text{Mass of fluid Column} = \int_A \rho \, dAdz. \tag{2.4}$$

Where ρ is the density (mass per unit volume), which here is constant as we assume incompressibility. And so integrating with respect to both the area **A** and the vertical dimension we get mass of the fluid column. To conserve the balance of mass, the flow of fluid across the boundary of the column must be balanced by an increase in the mass

of A. If we let S be the surface area of the column boundary, then the mass flux across S in the inward direction is

$$-\int_S \rho \mathbf{u} \cdot \mathbf{n} \; dS, \tag{2.5}$$

where $\mathbf{n}$ is the unit vector perpendicular to the boundary, and $\mathbf{u} = \quad (u \; v)^T$ is the horizontal velocity. Let us consider the case when the boundary surface S is broken down into sub-elements of area $h\delta l\mathbf{n}$. Then if we consider the flux in the inward direction along the unit vector of each subsection along δl, (5) becomes the

$$-\oint \rho h \; \mathbf{u} \cdot \mathbf{n} \; dl, \tag{2.6}$$

integrating all subsections over the closed curve A. Using the Divergence Theorem,

$$-\oint \rho h \; \mathbf{u} \cdot \mathbf{n} \; dl = -\int_A \nabla(\rho h \; \mathbf{u}) \; dA. \tag{2.7}$$

$$\tag{2.8}$$

If mass $= \int \rho \; dV$, the change in mass over time is

$$\frac{d}{dt}\int \rho \; dV = \frac{d}{dt}\int_A h\rho \; dA = \int_A \rho \frac{\partial h}{dt} \; dA. \tag{2.9}$$

And so as the net increase in mass equals the flux across the boundary S,

$$-\int_A \nabla(\rho \mathbf{u} h)\ dA = \int_A \rho \frac{\partial h}{\partial t}\ dA. \tag{2.10}$$

and we have

$$\int_A \rho \frac{\partial h}{\partial t} + \nabla \cdot (\mathbf{u} h)\ dA\ =\ 0\ . \tag{2.11}$$

As this holds for any A, we have the continuity equation for shallow water

$$\frac{\partial h}{\partial t} + \nabla \cdot (\mathbf{u} h) = 0. \tag{2.12}$$

2.1.2 Balance of Momentum

The Balance of Momentum equation for the Shallow Water Equations is derived using the the assumption of hydrostatic balance and Newton's second law of motion, implying that since $\lambda >> H$, vertical velocity $<<$ horizontal velocity. The hydrostatic pressure equation is

$$\frac{\partial p}{\partial z} = -\rho g, \tag{2.13}$$

and so integrating with respect to the height z,

$$p = -\rho g z + p_0. \tag{2.14}$$

We assume that the atmosphere exerts negligible pressure on the fluid. Then $p = 0$ on the free surface ($z = h = \eta(x, y, t)$). Using this fact we get

$$p_0 = \rho g \eta. \tag{2.15}$$

Substituting back into (14) we get

$$\begin{aligned} p = & -\rho g z + \rho g \eta, \\ & = \rho g \ (\eta - z), \\ & = \rho g \ \big(h(x, y, t) - z\big) \end{aligned}$$

since $\eta(x, y, t)$ and $h(x, y, t)$ have the same z-value. Note, that the horizontal pressure gradient ∇p is independent of height

$$\nabla p = \rho g \nabla \eta. \tag{2.16}$$

And so when we substitute this expression back into our balance of momentum equation (2) we get

$$\begin{aligned} \frac{D\mathbf{u}}{Dt} &= -\frac{1}{\rho}\nabla p, \\ &= -g\nabla\eta. \end{aligned}$$

This tells us that the horizontal flow of the fluid over time as represented by the material derivative is independent of height z, and will stay so. Therefore the momemtum equation in shallow water is

$$\frac{D\mathbf{u}}{Dt} = -g\nabla\eta, \tag{2.17}$$

Combining the continuity and momentum equations, we have **the shallow water equations**.

$$\begin{aligned} \frac{\partial h}{\partial t} + \nabla \cdot (h\mathbf{u}) &= 0, \\ \frac{D\mathbf{u}}{Dt} + g\nabla\eta &= 0. \end{aligned}$$

2.2 Dissipative Forms of the Shallow Water Equations

2.2.1 Viscosity

To account for viscosity, we use the Navier-Stokes equations to derive the shallow water equations in the same way, from the conservation of mass and momentum. However, there is an additional viscosity term. In our derivation of the laws of mass and momentum for the Euler equations, we neglected viscosity in the sense that we assumed forces across the fluid surface were normal to that surface. This is not always a good approximation. And so as a consequence, the stress forces on the surface of the fluid are no longer simply $p \cdot \mathbf{n}$, and we add another term $\sigma(x,t) \cdot \mathbf{n}$ where σ is the stress tensor matrix, and $\sigma(x,t) \cdot \mathbf{n}$ is not necessarily orthogonal to the surface. σ is related linearly to the velocity gradients $\nabla \mathbf{u}$. Specifically, it depends on the deformation of $\nabla \mathbf{u}$, (the symmetric matrix $\mathbf{D}$). σ and $\nabla \mathbf{u}$ are related by the following

$$\sigma = \lambda(\nabla \cdot \mathbf{u})\mathbf{I} + 2\mu\mathbf{D}, \tag{2.18}$$

where μ and $\zeta = \lambda + \frac{2}{3}\mu$ are known as the **coefficients of viscosity**.

Deriving the balance of momentum equation as previously, but this time including the stress tensor, we get the **Navier Stokes Equations for Incompressible Flow**

$$\rho \frac{D\mathbf{v}}{Dt} + \nabla p - \mu \Delta \mathbf{v} = 0,$$
$$\nabla \cdot \mathbf{v} = 0,$$

where ρ is a constant (homogeneous incompressible flow), and $\nu = \frac{\mu}{\rho}$ is the coefficient of **kinematic viscosity**. Therefore when we derive the momentum equation for shallow water, we also include the viscous term in our calculations. Using the hydrostatic equation as before to derive the shallow water equation, we have the **Viscous Shallow Water Equations**

$$\frac{\partial h}{\partial t} + \nabla \cdot \ (h\mathbf{u}) = 0, \tag{2.19}$$

$$\frac{D\mathbf{u}}{Dt} + g\nabla\eta - \nu\Delta\mathbf{u} = 0. \tag{2.20}$$

2.2.2 Constant Density Flow (Homogeneous Flows)

In the previous section we assume that the fluid density is constant, and does not vary; $\rho(x, y, t) = \rho_0 =$ constant. With constant density ρ, the flow is necessarily incompressible. And so the continuity equation is automatically satisfied (due to incompressibility) and we get

$$\nabla\mathbf{u} \ = \ \mathbf{0}. \tag{2.21}$$

However, incompressibility does not necessarily imply constant density. There are incompressible flows with variable density, for example salt concentration is variable in the oceans, but the equation $\nabla\mathbf{u} \ = \ \mathbf{0}$ still holds. Note that we can only assume incompressibility if salt does not diffuse.

2.2.3 Rotation

The presence of Coriolis force takes into account the Earth's rotation. In our derivation of the balance of momentum, the effect of the Earth's rotation on the motion of the fluid was neglected. If $\mathbf{v}$ is the velocity of the fluid and $\mathbf{\Omega}$ is the angular velocity vector pointing out of the northern hemisphere, then the Coriolis force is $-2\mathbf{\Omega}\sin\phi \times \mathbf{v}$, where ϕ is the latitude. It will deflect particles to the right in the northern hemisphere, and to the left in the hemisphere, where the angular deflection is $\mathbf{\Omega}t$, which represents the Earth's rotation at time t. We call $-2\mathbf{\Omega}\sin\phi$ the Coriolis parameter, and denote it by $\mathbf{f}$. In terms of our balance of momentum equation, we need to add in the Coriolis force into our equation:

$$\frac{D\mathbf{v}}{Dt} + \mathbf{f} \times \mathbf{v} = -\frac{1}{\rho}\,\nabla p. \tag{2.22}$$

To account for this in the Shallow Water equations, we precede as previously, using the hydrostatic equation to derive an expression for pressure in terms of gravity and density. Keeping our Coriolis force parameter in the momentum equation, and integrating over height in our continuity equation as before, our inviscid Shallow Water equations become:

$$\frac{D\mathbf{u}}{Dt} + \mathbf{f} \times \mathbf{u} + g\nabla\eta = 0, \tag{2.23}$$

$$\frac{\partial h}{\partial t} + \nabla \cdot (h\mathbf{u}) = 0. \tag{2.24}$$

2.3 Hydrostatic & Geostrophic Balance

2.3.1 Hydrostatic Balance

In our derivation of the shallow water momentum equation, we assumed the system was in hydrostatic balance, and thus were able to proceed using the hydrostatic equation. Let us now consider what makes that assumption valid. We stipulated that the hydrostatic assumption is met because of the aspect ratio of our fluid layer, namely that $\lambda >> h$. We can prove this assertion thus.

Consider for a moment the momentum equation in the z-direction

$$\frac{\partial w}{\partial t} + w \cdot \nabla \mathbf{v} + 2(\Omega_x v - \Omega_y u) = \frac{1}{\rho}\frac{\partial p}{\partial z} - g, \tag{2.25}$$

where $\mathbf{v} = (u\ v\ w)^T$ is the three-dimensional velocity of the fluid. We now consider the components of this equilibrium in scalar non-dimensional form

$$\frac{UW}{T} + \frac{UW}{L} + \frac{W^2}{H} + \Omega U \approx \frac{\Phi}{H} - g \tag{2.26}$$

where the U represents the magnitude of the horizontal velocity and L the magnitude of horizontal length , W is the magnitude of the vertical velocity and H the magnitude of the vertical scale, Ω is the magnitude of the coriolis effect, and Φ of the change in $\phi = \frac{\delta p}{\rho_0}$.

Comparing these components, we can see that the magnitudes of the left-hand-size are much smaller relative to those on the right. And so for the equation to be in equilibrium,

the pressure gradient and the gravitational term must be approximately equal, and to balance it out we get

$$\frac{\partial p}{\partial z} = -\rho g \tag{2.27}$$

Let us consider an incompressible barotropic flow (with constant density), and rewrite the pressure as $p = p_0 + p'$, where $\frac{\partial p_0}{\partial z} = -\rho_0 g$ are in hydrostatic balance. Substituting this expression for p back into the vertical momentum equation (without viscosity) we get

$$\frac{Dw}{Dt} = -\frac{1}{\rho_0}\frac{\partial p'}{\partial z} \tag{2.28}$$

where we can see that there are no further effects by the gravitational force that have not been balanced out by the vertical pressure gradient. Subsequently, we can replace p by p' in the horizontal momentum equations as well. It should be noted however, that this result is by no means universal and in stratified fluid layers, we will need to account for the additional pressure forces associated with each layer. However, the implications of this for a barotropic non-rotating fluid in the absence of viscosity, is that we get the following Boussinesq momentum equation

$$\frac{D\mathbf{v}}{Dt} = -\nabla\phi, \tag{2.29}$$

Now we consider how this impacts the aspect ratio of our λ-h scale. Considering the horizontal non-rotating Boussinesq momemtum equation, we have

$$\frac{D\mathbf{u}}{Dt} = -\nabla\phi, \tag{2.30}$$

which implies the scaling $\Phi \sim U^2$. Using the continuity equation $\nabla \cdot \mathbf{v} = 0$, we can say that $\frac{U}{L} \sim \frac{W}{H}$. And if we consider the magnitudes of the horizontal (advective) terms of the vertical momentum equation, we have $\frac{Dw}{Dt} \sim \frac{UW}{L} = \frac{U^2 H}{L^2}$.

And so, if we consider the ratio

$$\frac{|\frac{Dw}{Dt}|}{|\frac{1}{\rho}\frac{\partial p}{\partial z}|} \sim \frac{\frac{U^2 H}{L^2}}{\frac{\Phi}{H}} \sim \frac{\frac{U^2 H}{L^2}}{\frac{U^2}{H}} = \frac{H}{L}^2, \tag{2.31}$$

since we know that for hydrostatic balance the magnitude of the pressure gradient is much higher than the material derivative of the fluid for large scale motions, we can say that $\frac{H}{L}^2 = \alpha^2 << 1$, and so

$$\alpha^2 = \frac{H}{L} << 1 \tag{2.32}$$

gives us an aspect ratio condition for hydrostatic balance.

2.3.2 Geostrophic Balance

Another assumption we make use of (for example, in finding the dispersion relation for Rossby waves) is geostrophic balance. The general idea is similar to that of hydrostatic balance, in that it concerns the relationships between magnitudes of the components

of the horizontal momentum equations; we say a flow is in geostrophic balance if the relation $\Phi \sim fUL$ holds, where U is the magnitude of horizontal velocity, L is the horizontal scale over which it varies, f is the Coriolis parameter and Φ is the pressure deviation. Using this, we obtain

$$\frac{|\frac{Dw}{Dt}|}{|\frac{1}{\rho}\frac{\partial p}{\partial z}|} \sim \frac{\frac{UW}{L}}{\frac{fUL}{H}} \sim \frac{U}{fL}\frac{WH}{UL} = R_0\frac{H}{L} \tag{2.33}$$

where the ratio $R_0 = \frac{U}{fL}$ is called the **Rossby Number**. If $R_0 << 1$, then this means that the scale of the Coriolis force in the horizontal momentum equation is much greater than that of $|\frac{D\mathbf{u}}{Dt}|$, and so we can say that the Coriolis force is balanced by the horizontal pressure gradient. Our shallow water horizontal momentum equations then are reduced to

$$-f_0 v = -g\frac{\partial \eta}{\partial x} \tag{2.34}$$

$$f_0 u = -g\frac{\partial \eta}{\partial y}. \tag{2.35}$$

These are the Geostrophic balance first order momentum equations. We use these assumptions to rewrite our horizontal velocity $\mathbf{u}$ as stream functions when looking at Rossby waves.

2.4 Waves in Shallow Water

There are different wave solutions to the shallow water equations, depending on the parameters we consider, such as the presence of rotation, boundaries or vorticity. These

solutions are classified as Gravity waves, Inertio-Gravity (or Poincaré) waves, Kelvin waves or Rossby waves. For the following analysis of shallow water wave solutions[4], let us consider the linearised shallow water equations with a flat bottom (so that $h = \eta + H$). Let $\mathbf{u} = (u\ v)^T$ be the horizontal velocity. We linearise about a state of rest, and assume u and η are small, with $\frac{\eta}{H} = O(\epsilon)$. Then using the fact that the nonlinear terms dependent on these variables have even smaller magnitudes, we can neglect them, and our momentum and continuity equations become

$$\frac{\partial u}{\partial t} - fv = -g\frac{\partial \eta}{\partial x}, \tag{2.36}$$

$$\frac{\partial v}{\partial t} + fu = -g\frac{\partial \eta}{\partial y}, \tag{2.37}$$

$$\frac{\partial \eta}{\partial t} + H\Big(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\Big) = 0. \tag{2.38}$$

where we use $\eta(x, y, t) = h = H + O(\epsilon)$ in the continuity equation, as we have a flat bottom assumption.

2.4.1 Gravity Waves

When we neglect the Coriolis effect in the shallow water equation, the solutions obtained are known as Gravity waves. They are found using cross differentiation of the horizontal momentum equations, i.e. differentiating the momentum equation in the meridional (north-south direction on the globe) direction with respect to x, and the equation in the zonal (west-east direction on the globe) direction with respect to y, and substituting into the partial derivative of the continuity equation with respect to t. The resulting

expression is

$$\frac{\partial^2 \eta}{\partial t^2} - gh(\nabla^2 \eta) \;=\; 0, \tag{2.39}$$

and this can be solved to find a solution for η of the form $\eta = \eta_0 e^{i(kx+ly-\omega t)}$, where k is the zonal wavenumber, the wave speed is $\sqrt{gH}$, l is the meridional wavenumber and ω is the wave frequency. Substituting this back into our wave equation gives us the following dispersion relation

$$\omega = \pm\sqrt{gH}K \text{ where } K = k^2 + l^2. \tag{2.40}$$

These solutions are called shallow water gravity waves, since the motion of the wave is dependent on gravity. Figure 2.3 illustrates the dispersion relation. These solutions only hold in shallow water, i.e. when the hydrostatic approximation based on the aspect ration of wavelength and fluid height is maintained.

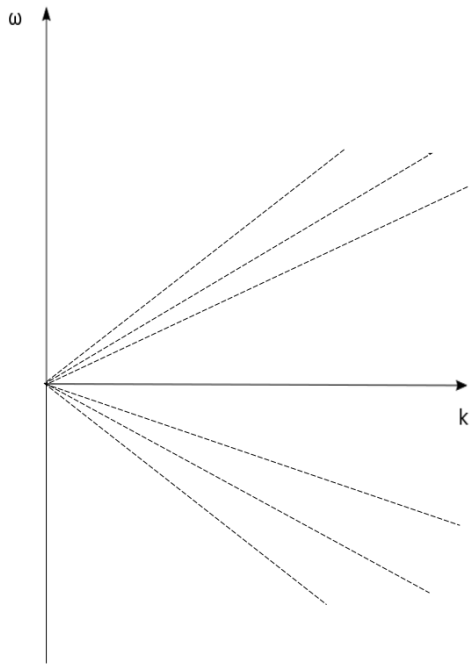

FIGURE 2.3: Dispersion relation for gravity waves, where $\omega = \pm\sqrt{gH}K$, where the different lines represent different values of H

2.4.2 Inertio-Gravity (Poincaré) Waves

Inertio-gravity waves are similar to gravity waves, except that this time Coriolis force is not omitted in our derivation. However, we assume that the coriolis parameter f is some constant, $f_0 = 2\Omega \sin \phi_0$. Proceeding as before (cross differentiation of the momentum equations, and substitution into the continuity equation), we get the following wave equation

$$\frac{\partial}{\partial t}\Big(\frac{\partial^2}{\partial t^2} + {f_0}^2 - gh\nabla^2\Big)\eta = \ 0. \tag{2.41}$$

We get the same solution for η as for gravity waves, and substituting back into our wave equation, we see our dispersion relation is

$$-i\omega(-\omega^2 + gH(K^2) + {f^2}_0) = \ 0. \tag{2.42}$$

where $K^2 = \ k^2 + l^2$. This gives us two relations

$$\omega = \ 0 \ \text{ or } \ \omega^2 = \ {f_0}^2 + gHK^2. \tag{2.43}$$

The zero case represents the time independent flow satisfying geostrophic balance, and the other case represents inertio-gravity waves in shallow water, and inertio-gravity waves are dependent on both gravity and rotational effects.

The propagation of inertio-gravity waves is dependent on the wavelength λ (and consequently the wavenumber K). Let us consider the dispersion relation

$$\omega^2 = \ {f_0}^2 + gHK^2. \tag{2.44}$$

If $K^2 >> \frac{{f_0}^2}{gH}$, then this relation can be approximated by $\omega^2 = \ gHK$. As we can see, this is the same relation as for gravity waves, and the effects of rotation are no longer significant. Since $\lambda = \ \frac{1}{k}$, this is equivalent to stating $\lambda << \frac{\sqrt{gH}}{f_0}$ and implies that small wavelengths will not be affected by the Earth's rotation.

Conversely, if $K^2 << \frac{{f_0}^2}{gH}$ (or $\lambda >> \frac{\sqrt{gH}}{f_0}$), then the dispersion relation can be approximated by $\omega^2 = \ {f_0}^2$, and describe waves where the primary influence is rotation. And so we can say that the ratio $\frac{\sqrt{gH}}{f_0}$ plays a significant role in determining the dominant effects

on the waves, whether of gravity or rotation. This is known as the **Rossby radius of deformation** and is denoted by L_D. As can be seen in figure 2.4, for small k (big λ) the primary effect is rotation, but as k gets bigger, beyond L_D, wave propagation is the same as of gravity waves in shallow water, in the absence of rotation.

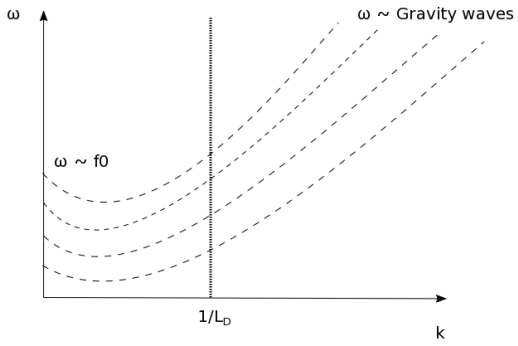

FIGURE 2.4: Dispersion relation for Poincaré waves: $\omega \approx f_0$ for small k, and $\omega \approx \sqrt{ghk}$ for large k

2.4.3 Kelvin Waves

Kelvin waves are solutions of the shallow water equations either at the equator where $f = 0$, or in the presence of some boundary (such as the coastline). Consider figure 2.5:

FIGURE 2.5: Boundary at $y = 0$.[4]

If we have a pressure gradient in the x-direction, then the Coriolis force will induce a velocity v in the y-direction. However, if we consider a similar pressure gradient along the boundary, then we cannot have a velocity in the y-direction, as we cannot have a flux perpendicular to the boundary. Therefore, when solving the equations we now also consider the boundary condition $v = 0$ at $y = 0$. It is reasonable then, to consider solutions where $v = 0$ everywhere, i.e. there is no flow in the y-direction. Substituting this into our momentum and continuity shallow water equations, and combining them as we did for Gravity and Poincaré waves, we get the following wave equation

$$\frac{\partial^2 \eta}{\partial t^2} + gh\,\frac{\partial^2 \eta}{\partial x^2} = 0. \tag{2.45}$$

Solving this gives us the following solution for η

$$\eta = A(y)\Big(e^{i(x+\sqrt{gH}t)} + e^{i(x-\sqrt{gH}t)}\Big). \tag{2.46}$$

We consider (46) in two parts, for $A(y)\Big(e^{i(x+\sqrt{gH}t)}\Big)$ and $A(y)\Big(e^{i(x-\sqrt{gH}t)}\Big)$ respectively.

For $A(y)\Big(e^{i(x+\sqrt{gH}t)}\Big)$ (westward propagating waves), substituting this into the momentum equation to find the displacement velocity in the x-direction and solving for $A(y)$ gives us $A(y) = e^{\frac{y}{L_D}} + C$. This implies the displacement grows exponentially in the y-direction, which is physically not possible, and so we disregard this. For $A(y)\Big(e^{i(x-\sqrt{gH}t)}\Big)$ (eastward propagating waves), we get $A(y) = e^{-\frac{y}{L_D}} + C$. This means that as we get further away from the boundary $y = 0$ and cover a distance greater than the Rossby Radius L_D, the amplitude of the dispersion η goes to zero exponentially.

And so we find the Kelvin solution to be

$$\eta = A(y)e^{-\frac{y}{L_D}}\left(e^{i(x-\sqrt{gH}t)}\right) \tag{2.47}$$

This solution for eastward propagating waves gives us the following one-dimensional dispersion relation

$$\omega = \sqrt{gH}k. \tag{2.48}$$

This is similar to a gravity wave, and Kelvin waves travel eastward at the same phase velocity regardless of wavenumber. They are essentially trapped to the boundary, and their amplitude goes to zero away from it. We can use a similar approach to solve the equations for any boundary, in the zonal or meridional directions. Figure 2.6 illustrates the dispersion relation for Kelvin waves.

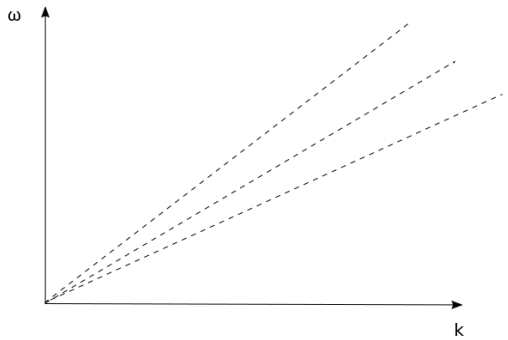

FIGURE 2.6: Dispersion relation for Kelvin waves: $\omega = \sqrt{gH}k$, with boundary at $y = 0$.

2.4.4 Rossby Waves

Previous when considering Inertio-Gravity waves, we assumed that the Coriolis parameter was constant, meaning that we considered waves along a constant latitude ϕ_0. However, now we consider wave propagation in shallow water when the latitude is allowed to vary. Consider the following diagram:

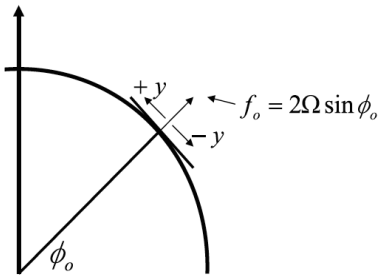

FIGURE 2.7: β-plane for varying latitude.[4]

As we can see, the Coriolis parameter is allowed to vary along the β-plane (tangent line at f_0 to the Earth), and we use Taylor expansions around $y = 0$ (where we have latitude ϕ_0). This means f varies linearly with the latitude.

$$f(y) = f_0 + \beta y \tag{2.49}$$

where $\beta = \frac{2\Omega \sin \phi_0}{a}$, and a is the earth's radius.

This change in Coriolis parameter with latitude is significant, because if there is any vorticity in the fluid, due to Potential Vorticity conservation, the relative vorticity of the fluid will be affected by variations in the background planetary vorticity (Coriolis effect).

The concept of potential vorticity provides us with a quantity related to vorticity, that is materially conserved;

$$\frac{Dq}{Dt} = 0 \ , \ q = \frac{1}{h}\bigg((\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}) + f_0\bigg). \tag{2.50}$$

Due to PV conservation, angular momentum is transferred and the relative vorticity will change as latitude varies.

And so to find a Rossby wave equation, we substitute our $f(y)$ into our shallow water momentum equations. And since we are assuming geostrophic balance, we can find approximate expressions for u and v, using the Geostrophic balance equations (42) and (43), and subsequently substitute these into the momentum equations, deriving formulae for u and v. Finally, substituting into the continuity equation gives us

$$\frac{1}{L_D}\frac{\partial \eta}{\partial t} - \frac{\partial}{\partial t}\nabla^2\eta + \beta_0\frac{\partial \eta}{\partial x} = 0 \tag{2.51}$$

where L_D is the Rossby radius of deformation.

Because we are assuming geostrophic balance, if we cross-differentiate the horizontal momentum equations (42) and (43), we eliminate the gravitional terms and see that div $\mathbf{u} = 0$. This then allows us to express the components of $\mathbf{u}$ as the derivatives of the stream function ψ, which represents the trajectory of fluid particles in a uniform flow. We then get

$$\mathbf{u} = \big(-\frac{\partial \psi}{\partial y} \ , \ \frac{\partial \psi}{\partial x}\big). \tag{2.52}$$

And so rewriting (50) in terms of ψ ,we have the following non-linear wave equation for Rossby waves

$$\frac{\partial}{\partial t}\big(\nabla^2\psi - \frac{1}{{L_D}^2}\psi\big) + \beta\frac{\partial\psi}{\partial x} = 0. \tag{2.53}$$

Solving this wave equation for ψ gives us the solution $\psi = \psi_0 e^{i(kx+ly-\omega t)}$, and substituting this back into our equation yields the following one-dimensional dispersion relation (neglecting the meridional wavenumber) for barotropic shallow water Rossby waves

$$\omega = -\frac{\beta k}{k^2 + \frac{1}{{L^2}_D}}. \tag{2.54}$$

As we can see this is always negative, and so these waves always propagate westward with respect to the zonal flow.

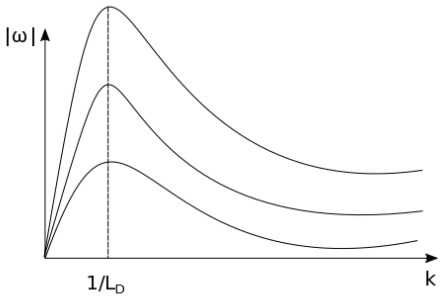

FIGURE 2.8: Dispersion relation for Rossby waves: $\omega = -\frac{\beta k}{k^2+\frac{1}{{L^2}_D}}$.

2.4.5 Wave Propagation Summary

Type	**Wave Equation**	**Solution**	**Dispersion Relation**	**Propagation**
Gravity	$\frac{\partial^2\eta}{\partial t^2} - gH\frac{\partial^2\eta}{\partial x^2} = 0$	$\eta_0 e^{i(kx+ly-\omega t)}$	$\omega = \pm\sqrt{gH}K$	Waves travelling at the same phase speed independent of wavenumber K.
Poincaré	$\frac{\partial^2\eta}{\partial t^2} + {f_0}^2\eta - gH\frac{\partial^2\eta}{\partial x^2} = 0$	$\eta_0 e^{i(kx+ly-\omega t)}$	$\omega = 0$ or $\omega^2 = {f_0}^2 + gHK^2$	Dispersive or non-dispersive if wavelength greater or smaller than Rossby Radius of Deformation.
Kelvin	$\frac{\partial^2\eta}{\partial t^2} + gH\frac{\partial^2\eta}{\partial x^2} = 0$	$A(y)e^{-\frac{y}{L_D}}\left(e^{i(x-\sqrt{gH}t)}\right)$	$\omega = \sqrt{gH}k$	Exhibit behaviour of gravity waves at vertical plane of boundary and any parallel vertical plane; in NH propagate with boundary on right-hand-side and vice versa in the SH.
Rossby	$\frac{\partial}{\partial t}\left(\nabla^2\psi - \frac{1}{{L_D}^2}\psi\right) + \beta\frac{\partial\psi}{\partial x} = 0$	$\psi_0 e^{i(kx+ly-\omega t)}$	$\omega = -\frac{\beta k}{K^2+\frac{1}{L^2{}_D}}$	Dispersive waves as ω dependent on wave number.

2.5 Boussinesq Approximation

The topic of frequency dispersion brings to light the suitability of the shallow water equations which neglect frequency dispersion effects on wave propagation, versus another approximation in shallow water which accounts for it; the Boussinesq approximation .

The following are the one-dimensional Boussinesq equations for depth averaged veleocity[20]

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + g\frac{\partial \eta}{\partial x} - \frac{H^2}{3}\frac{\partial^3 u}{\partial x^2 \partial t} = 0, \tag{2.55}$$

$$\frac{\partial \eta}{\partial t} + \frac{\partial}{\partial x}\Big((H+\eta)u\Big) = 0. \tag{2.56}$$

2.6 Comparison with Shallow Water System

When we look at the the Boussinesq momentum equation for constant depth H, we see that the only difference between the shallow water equations under the same conditions is that the term $\frac{H^2}{3}\frac{\partial^3 \bar{u}}{\partial x^2 \partial t}$ is neglected in the shallow water approximation but retained in Boussinesq. To consider this, let us look at the scaled non-dimensionalised form of the momentum equation.

Non-dimensionalisation is a useful tool, as it allows us to remove additional parameters and variables from the system while conserving its dynamics, thus making it simpler to solve. We create a new set of dimensionless variables representing ratios of the values with some fundamental units of the same dimensionality. In this case, our variables are non-dimensionalised with respect to ρ, g, and H and we have

$$\tilde{x} = \frac{x}{H},\ \tilde{z} = \frac{z}{H},\ \tilde{h} = \frac{h}{H},\ \tilde{\eta} = \frac{\eta}{H},\ \tilde{t} = \sqrt{\frac{g}{H}}t,\ \tilde{U} = \frac{U}{\sqrt{gH}},\ \tilde{W} = \frac{W}{\sqrt{gH}},\ \tilde{p} = \frac{p}{\rho g H}. \quad (2.57)$$

This choice is not unique, and other expressions can be used to nondimensionalize with the same resulting system. However these particular transformations allow for a simplified derivation, when we consider the order of the terms retained. Before we proceed to scale our system so that we can compare the relative magnitudes between the terms to the first order, let us introduce some fundamental parameters; let λ be the typical wavelength, H the typical depth and a the typical amplitude of the surface displacement. We use these to introduce the following parameters

$$\varepsilon = \frac{a}{H}, \tag{2.58}$$

$$\sigma = \frac{H}{\lambda}. \tag{2.59}$$

where the ratios ε and σ parametrise the nonlinearity and dispersion present in the system respectively. We already know that shallow water approximation requires the typical wavelength λ to be very large compared to the typical depth H, and so we require that $\sigma << 1$. Additionally, we assume that the amplitude of the surface displacement be small compared to the depth, requiring that $\sigma << 1$.

A condition of the Boussinesq approximation is that the ratio $\frac{\varepsilon}{\sigma^2}$ known as the Ursell Number Ur, be $\mathcal{O}(1)$. Because of this, the final representation of the system contains terms up to and including $\mathcal{O}(\varepsilon, \sigma^2)$, and terms of higher order are discarded.

We now scale our variables, so that the scaled x-coordinate changes $\mathcal{O}(1)$ over one wavelength λ, and the free surface displacement changes $\mathcal{O}(1)$ compared to a. Time is scaled such that horizontal velocity is $\mathcal{O}(1)$. And so we have

$$\hat{x} = \sigma\tilde{x},\ \hat{t} = \sigma\tilde{t},\ \hat{\eta} = \frac{\tilde{\eta}}{\varepsilon} \tag{2.60}$$

Substituting these scales into the free surface boundary condition and the continuity equation gives us the following scaled velocities

$$\hat{U} = \frac{\tilde{U}}{\varepsilon},\ \hat{W} = \frac{\tilde{W}}{\varepsilon\sigma}\ . \tag{2.61}$$

And so we have the non-dimensional form of the momentum equation

$$\frac{\partial \hat{\tilde{u}}}{\partial \hat{t}} + \varepsilon \hat{\tilde{u}} \frac{\partial \hat{\tilde{u}}}{\partial \hat{x}} + \frac{\partial \hat{\eta}}{\partial \hat{x}} + \sigma^2 \Big(\frac{\tilde{h}^2}{6} \frac{\partial^2}{\partial \hat{x}^2} \Big(\frac{\partial \hat{\tilde{u}}}{\partial \hat{t}} \Big) - \frac{\tilde{h}}{2} \frac{\partial^2}{\partial \hat{x}^2} \Big(\tilde{h} \frac{\partial \hat{\tilde{u}}}{\partial \hat{t}} \Big) \Big) \; = \; \mathcal{O}(\varepsilon\sigma^2, \sigma^4) \; . \tag{2.62}$$

As we can see, in order to discard the term with $\mathcal{O}(\sigma^2)$, we require that the magnitude of the term of $\mathcal{O}(\varepsilon)$ be much larger, such that the former is considered negligible. Subsequently we need the scale ε to be much larger than σ^2, and so we can say that for the shallow water equations we need $Ur >> 1$. And therefore evaluating the suitability of applying the Boussinesq water wave equations and the shallow water equations depends on the ratio between ε and σ^2; if it is of magnitude $\mathcal{O}(1)$ then Boussinesq is the better theoretical choice, and if $\sigma^2 << \varepsilon$, then the shallow water equations are more suitable.

Let us consider an example to illustrate the different requirements on the wavelength λ in determining the suitable approximation. Let the depth average height of the water column H equal 1, and let the amplitude a of the wave be 10^{-2}. Then the Ursell number Ur is $\lambda^2/10^2$. As the Boussinesq approximation requires that $Ur \approx 1$, we need $\lambda \approx 10$ to use this model. However, as the shallow water approximation requires $Ur >> 1$, we require $\lambda >> 10$. And so we can see that the wavelength needs to be much larger to accurately fit a shallow water model, than is required to fit a $\mathcal{O}(\sigma^2)$ Boussinesq model.

2.6.1 Applications of the Boussinesq Equations

As outlined in [9], the Boussinesq equations are most often used to model near shore hydrodynamic behaviour, and can include additional effects accounting for wave breaking, mixing and propagation. This is because the wavelength-to-water-depth ratio as

described by $Ur \approx 1$ is most closely exhibited in the region of breaking waves, and directly seaward of that point. Aditionally, for the linearised problem the phase velocity is described by

$$c^2 = gH\frac{\tanh(kH)}{kH} \ , \tag{2.63}$$

where k is the spatial wavenumber, and for $\sigma << 1$ as required for the long wave approximation, this reduces to $c^2 \approx gH$, and the resulting waves are nearly non-dispersive, with the leading order correction

$$c^2 = gH(1 + \mathcal{O}(kH))^2 \ . \tag{2.64}$$

Obviously, this results in restrictions imposed on the range of water depths considered in the approximation, and it is the goal of many modern Boussinesq models to approximate the ratio imposed by (51), no longer restricted by the requirement that $\sigma << 1$ [9]. Examples of such restrictions of the $\mathcal{O}(\sigma^2)$ Boussinesq model include the breakdown of predictions of the horizontal velocity near the sea bed for values of σ far smaller than than required by Ur of $\mathcal{O}(1)$; [5] showed that in their model for $\sigma^2 = 10$, the horizontal velocity fell to zero and became negative relative to the surface velocity in deeper waters, exhibiting the drawbacks of the model for near bed kinematics, at depths outside of the required ratios between water depth and wavelength. In contrast, the shallow water equations are commonly used to describe rivers and lakes, and to model flooding and dam breakages, as well as multilayer flows. Both Boussinesq and Shallow Water aproximations are widely used for analysing solitary wave propagation, and studies such as [5] have compared the two processes, finding that in certain run up processes the two approximations are identical, however for breaking waves, the Boussinesq equations

provide a better model of the wave evolution up till breaking point.

In conclusion, we can say that while this study focuses on the shallow water equations, applying the 4D-VAR algorithm to the Boussinesq equations for water waves and then comparing the accuracy of initial condition prediction with the shallow water model would be a logical next step for future research.

Chapter 3

Solving the One-Dimensional Shallow Water Equations Forward in Time

3.1 Finite Volume Approximation in Space

3.1.1 Non-Linear System

We begin with the coupled shallow water one-dimensional equations in conservation form

$$\frac{\partial \eta}{\partial t} + \frac{\partial}{\partial x}\Big((H+\eta)u\Big) = 0, \tag{3.1}$$

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\Big(\frac{1}{2}u^2 + g\eta\Big) = 0. \tag{3.2}$$

We take our domain to be $[a, -a]$ for some real number a , and begin with initial conditions for η and u at time $t = 0$. These are of the form

$$\eta_0(x) = \ e^{\frac{-x^2}{\delta^2}} \ , \ u_0(x) = \ 0.$$

Where η_0 is a Gaussian curve centred at zero, and δ is the standard deviation parameter. We proceed by discretising our domain in space, so that we have $N-1$ control volumes of size $\Delta x = \ x_{i+1} - x_i$, for $x = 1, ..., N$.

We can now use a second order finite difference-finite volume method to calculate approximations for the flux integrals across the edges of each control volume. We consider the integral form of our equations, and use the Divergence Theorem to express the volume integral as surface integrals representing the fluxes across our boundaries. We then use the midpoint rule to approximate the volume integral as the product of our approximation of (hu) at the cell centre (so at $x_{i+\frac{1}{2}}$), and the magnitude of the control volume, in this case Δx. For our equation (1), this will look like

$$\begin{aligned}
\int_{V_i} \frac{\partial}{\partial x}\Big((H+\eta)u\Big)_{i+\frac{1}{2}} &= \int_S \Big((H+\eta)u\Big) \cdot \ \mathbf{n} \ dS \\
\frac{\partial}{\partial x}\Big((H+\eta)u\Big)_{i+\frac{1}{2}} \times \Delta x &= \int_S \Big((H+\eta)u\Big) \cdot \ \mathbf{n} \ dS \\
\frac{\partial}{\partial x}\Big((H+\eta)u\Big)_{i+\frac{1}{2}} &= \frac{1}{\Delta x}\int_S \Big((H+\eta)u\Big) \cdot \ \mathbf{n} \ dS
\end{aligned}$$

And subsequently we can express $\frac{\partial \eta}{\partial t}$ in (1) at the centre of the cell in terms of the flux integrals at the boundary of each discrete control volume. The next step is to find an approximation for the flux integrals. For each control volume, these can be split into two integrals for each cell,

$$\int_S \Big((H+\eta)u\Big)\cdot\ \mathbf{n}\ dS = \int_{S_w} \Big((H+\eta)u\Big)_i \cdot\ \mathbf{n}\ dS + \int_{S_e} \Big((H+\eta)u\Big)_{i+1} \cdot\ \mathbf{n}\ dS,$$

Where S_w and S_e represent the flux at the western and eastern edges of the control volume respectively. As the unit normal in the western direction is -1, and in the eastern direction is 1, these become

$$\int_{S_e} \Big((H+\eta)u\Big)_{i+1}\ dS\ - \int_{S_w} \Big((H+\eta)u\Big)_i\ dS,$$

We now approximate these integrals, and get values for $\Big((H+\eta)u\Big)_i$ and $\Big((H+\eta)u\Big)_{i+1}$.

After time t_0, u and η are generated using a numerical discretisation scheme in time to integrate $\frac{\partial u}{\partial t}$ and $\frac{\partial \eta}{\partial t}$. To illustrate this, for now let us suppose we move forward in time using a simple first order Euler Explicit Scheme of the form

$$\eta_{n+1} = \ \eta_n - \Delta t\Big(\Big((H+\eta_n)u_n\Big)\Big)_x$$
$$u_{n+1} = \ u_n - \Delta t\Big(\frac{1}{2}{u_n}^2 + g\eta_n\Big)_x$$

We discretize so that at each time step we find u at the edges of each spacial control volume cells, so at i, $i+1$ etc, and find η at the centre points of each cell, so that at each time step, we have values for u_i, u_{i+1}, u_{i+2}, and $\eta_{i-\frac{1}{2}}$, $\eta_{i+\frac{1}{2}}$, $\eta_{i+\frac{3}{2}}$, and so on. (We use this staggered grid method, since if we were to evaluate u and η at the same points directly our system would become uncoupled.) With these values, for each time step we

can find an approximation for the flux integrals at the control volume edges using the following centre difference formula to find h

$$\eta_{i+1} = \frac{1}{2}\Big(\eta_{i+\frac{1}{2}} + \eta_{i+\frac{3}{2}}\Big).$$

At the control volume edge x_{i+1}, we have

$$\Big((H+\eta)u\Big)_{i+1} \approx \frac{1}{2}\Big(H + \eta_{i+\frac{1}{2}} + \eta_{i+\frac{3}{2}}\Big)(u_{i+1})$$

Using this, our finite volume approximation for $\frac{\partial}{\partial x}\Big((H+\eta)u\Big)_{i+\frac{1}{2}}$ at the control volume centre becomes

$$\begin{aligned}
\frac{\partial}{\partial x}\Big((H+\eta)u\Big)_{i+\frac{1}{2}} &= \frac{1}{\Delta x}\int_S \Big((H+\eta)u\Big)_{i+\frac{1}{2}} \cdot \mathbf{n}\, dS \\
&= \frac{1}{\Delta x}\int_{S_e} \Big((H+\eta)u\Big)_{i+1} dS \; - \int_{S_w} \Big((H+\eta)u\Big)_i \, dS \\
&\approx \frac{1}{\Delta x}\frac{1}{2}\Big(\Big(\eta_{i+\frac{1}{2}} + \eta_{i+\frac{3}{2}}\Big)u_{i+1} - \; \Big(\eta_{i-\frac{1}{2}} + \eta_{i+\frac{1}{2}}\Big)u_i\Big)
\end{aligned}$$

And so, since $\frac{\partial \eta}{\partial t} = \; -\frac{\partial}{\partial x}\Big((H+\eta)u\Big)$, we can say that

$$\frac{\partial \eta}{\partial t} \approx \; -\frac{1}{\Delta x}\frac{1}{2}\Big(\Big(\eta_{i+\frac{1}{2}} + \eta_{i+\frac{3}{2}}\Big)u_{i+1} - \; \Big(\eta_{i-\frac{1}{2}} + \eta_{i+\frac{1}{2}}\Big)u_i\Big)$$

Now that we have an expression for $\frac{\partial \eta}{\partial t}$ (and likewise can compute a similar expression for $\frac{\partial u}{\partial t}$) , we proceed using a third order four stage Runge-Kutta numerical scheme (we discuss later why this and not the Euler scheme is implemented), where our inputs at the nth time step $\Big(\frac{\partial \eta}{\partial t}\Big)^n$ and $\Big(\frac{\partial u}{\partial t}\Big)^n$ in terms of u^n, η^n, Δt, Δx will give us the outputs

$\left(\frac{\partial \eta}{\partial t}\right)^{n+1}$ and $\left(\frac{\partial u}{\partial t}\right)^{n+1}$. This will allow us to find the trends in the wave height and velocity over time.

We now derive our approximation for $\frac{\partial u}{\partial t}$ at the control volume boundaries in a similar fashion.

$$\begin{aligned}
\left(\frac{\partial u}{\partial t}\right)_i &= -\frac{\partial}{\partial x}\left(\frac{1}{2}u^2 + g\eta\right)_i \\
&\approx -\frac{1}{\Delta x}\left[\left(\frac{1}{2}u^2 + g\eta\right)_{i+1} - \left(\frac{1}{2}u^2 + g\eta\right)_i\right] \\
&= -\frac{1}{\Delta x}\left[\left(\frac{1}{2}(u_{i+1})^2 + g(\eta_{i+1})\right) - \left(\frac{1}{2}(u_i)^2 + g(\eta_i)\right)\right] \\
&= -\frac{1}{\Delta x}\left[\left(\frac{1}{2}(u_{i+1})^2 + \frac{1}{2}g\left(\eta_{i+\frac{1}{2}} + \eta_{i+\frac{3}{2}}\right)\right) - \left(\frac{1}{2}(u_i)^2 + \frac{1}{2}g\left(\eta_{i-\frac{1}{2}} + \eta_{i+\frac{1}{2}}\right)^2\right)\right]
\end{aligned}$$

3.1.2 Linearised System

To solve our the shallow water equations, we take a step back and begin by considering the simpler, linearised system. We linearise about the average depth H and zero velocity u so that $h = H + \eta$, where $\eta << 1$ is the perturbation of the surface.

$$\begin{aligned}
0 &= \frac{\partial h}{\partial t} + \frac{\partial}{\partial x}\left(hu\right) \\
&= \frac{\partial (H+\eta)}{\partial t} + \frac{\partial}{\partial x}\left((H+\eta)u\right) \\
&= \frac{\partial \eta}{\partial t} + \frac{\partial}{\partial x}(Hu).
\end{aligned}$$

Where the non-linear terms have been discarded. Similarly,

$$\begin{aligned}\frac{\partial u}{\partial t} &= -\frac{\partial}{\partial x}\Big(\frac{1}{2}u^2 + g\eta\Big) \\ &= -\frac{\partial}{\partial x}\Big(g\eta\Big).\end{aligned}$$

For simplicity, we non-dimensionalise by taking $g = H = 1$, and so we have

$$\frac{\partial \eta}{\partial t} = -\frac{\partial u}{\partial x}, \tag{3.3}$$

$$\frac{\partial u}{\partial t} = -\frac{\partial \eta}{\partial x} \tag{3.4}$$

We discretise in space as before, and our approximation for the linear system in space becomes

$$\Big(\frac{\partial u}{\partial t}\Big)_i = -\frac{1}{\Delta x}\Big(\eta_{i+\frac{1}{2}} - \eta_{i-\frac{1}{2}}\Big) \tag{3.5}$$

$$\Big(\frac{\partial \eta}{\partial t}\Big)_{i+\frac{1}{2}} = -\frac{1}{\Delta x}\Big(u_{i+1} - u_i\Big) \tag{3.6}$$

$$\tag{3.7}$$

3.2 Runge-Kutta Time Integration Scheme

Now that we have a numerical approximation for the partial derivatives in space, we use a four stage third order Runge-Kutta scheme [16] to approximate the trends in time for height and velocity, and so find numerical solutions for both at each time step between the initial time t_0 and final time t_T. We consider this scheme, as it preserves stability

strongly, and it was found that simpler numerical schemes such as the explicit Euler scheme did not yield sufficiently stable solutions. The solution at the $n+1$th time step is calculated thus:

$$
\begin{aligned}
U_0 &= y_n, \\
U_1 &= U_0 + \Delta t \frac{1}{2} f\Big(t, U_0, \frac{\partial y}{\partial x}\Big), \\
U_2 &= U_1 + \Delta t \frac{1}{2} f\Big(t, U_1, \frac{\partial y}{\partial x}\Big), \\
U_3 &= \frac{2}{3} U_0 + \frac{1}{2} U_2 + \Delta t \frac{1}{6} f\Big(t, U_2, \frac{\partial y}{\partial x}\Big), \\
U_4 &= U_3 + \Delta t \frac{1}{2} f\Big(t, U_3, \frac{\partial y}{\partial x}\Big), \\
y_{n+1} &= U_4, \\
t_{n+1} &= t_n + \Delta t.
\end{aligned}
$$

y_n represents the solution of the system at time n with y_1 being the initial condition at t_0, and $f\Big(t, U, \frac{\partial y}{\partial x}\Big)$ is the approximation of the right-hand-side in space. And so using the finite volume and Runge-Kutta approximations, we can solve our linearised system in forward time from $t = 0$ to final time $t = T$ for some integer T, and thus obtain solutions for $u(x,t)$ and $\eta(x,t)$ across our domain.

3.2.1 Stability

In a general sense, a numerical scheme is stable in time if it does not amplify numerical errors. A lack of stability in our scheme can cause exponential decay or growth in our solution. The Euler explicit scheme example we gave earlier,

$$\eta^{n+1} = \eta^n - \Delta t\Big(u^n\Big)_x, \tag{3.8}$$

$$u^{n+1} = u^n - \Delta t\Big(\eta^n\Big)_x, \tag{3.9}$$

is not a stable numerical scheme for our system. This is because a forward Euler explicit scheme in time and a centred difference method in space (FTCS) is unstable for the advection equation, which is the form of our PDEs for the linearised shallow water equations. We can check this using Von Neumann stability analysis, which stipulates that the solution for h^n should be

$$\eta^n = \xi^n e^{ik\Delta x} \tag{3.10}$$

where k is our wave number, and the amplication factor $\xi = \xi(k)$ varies with k. The scheme is unstable if $|\xi(k)| > 1$. We can find our amplication factor for our scheme by substituting the form in (10) into (8). We get

$$\xi(k) = 1 - \frac{i\Delta t}{\Delta x}\sin(k\Delta x) \tag{3.11}$$

and so $|\xi(k)| > 1\ \forall p$, and hence our scheme utilising an explicit Euler forward method in time is unstable.

3.2.2 Courant-Friedrichs-Lewy Condition

Stability in the Von Neumann sense is not the only condition required for our scheme to generate a stable solution. In order to obtain a correct solution for our system, we

need to ensure that the Courant-Friedrichs-Lewy (CLF) condition is met. The CFL condition stipulates that in our time integration scheme, the time step must be less than a certain value; if our time step is too large, then it may be larger than the time taken for information to travel to adjacent grid points in space where the solution is calculated. The one-dimensional form for the CFL condition in an explicit scheme is :

$$c\Delta t \leq \Delta x \tag{3.12}$$

Where c is the wave speed, and the quotient $c\Delta t/\Delta x$ is called the CFL number . As $c = \sqrt{gH} = 1$, we need $c\Delta t \leq \Delta x$. The third order four stage Runge-Kutta scheme we implement is stable for CFL ≤ 2 [16], and so in our solver to make sure this condition is met, we take

$$\Delta t = \frac{\Delta x}{3}, \tag{3.13}$$

and thus ensure our scheme is numerically stable.

3.3 Comparison with Analytical Solution

We now compare our solution with the analytical solution. The parameters for our simulation are

$$N = 2000,$$
$$x \in [-8, 8],$$
$$t \in [0, 2].$$

The analytical solution for the linearised system with initial conditions $\eta(x,0) = \eta_0(x), u(x,0) = 0$ is

$$\eta(x,t) = \frac{1}{2}\eta_0(x+t) + \frac{1}{2}\eta_0(x-t),$$
$$u(x,t) = -\frac{1}{2}\eta_0(x+t) + \frac{1}{2}\eta_0(x-t),$$

To check our numerical solution against the analytical, we fix a point $x_0 = 1.5$, and then compare $\eta_{numerical}(x_0,t)$ and $\eta_{analytical}(x_0,t)$. Taking $h_0 = e^{\frac{-x^2}{0.2^2}}$ and $t \in [0,2]$, the following graph represents the change in height at x_0 over time for both solutions.

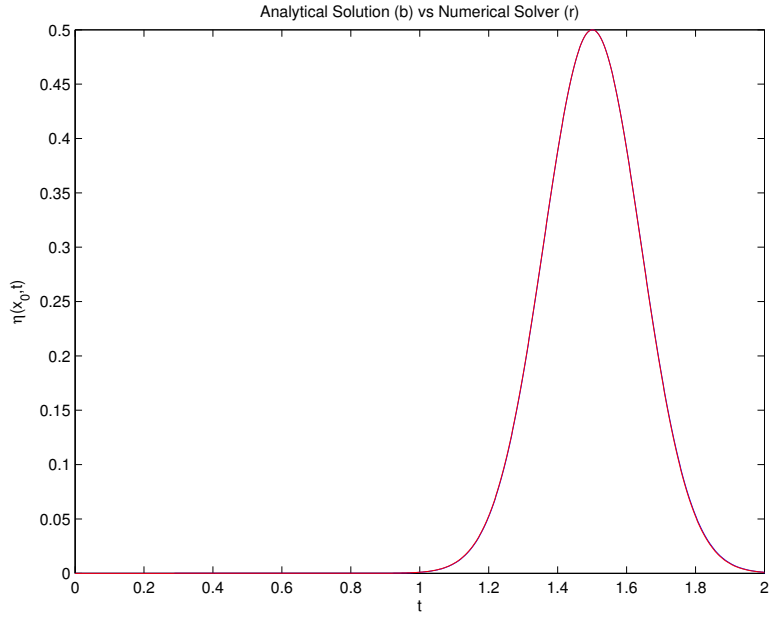

FIGURE 3.1: Analytically derived height (blue) vs the numerical height calculated using forward solver (red) at x_0 over time period 0 to 2.

As we can see, the numerical and analytical solutions coincide almost exactly, and any error that results is due to the numerical approximation. We therefore have a functioning

algorithm for the first stage of our data assimilation scheme, to be fully discussed in chapters four and five, and further tested in chapter six.

Chapter 4

Optimal Control and Data Assimilation

4.1 Introduction

Data Assimilation is the study of finding the most suitable way to integrate actual data into a computational model in order to achieve the most accurate representation of a dynamical system possible. Simply replacing model values with observed data is not always appropriate as they may be inconsistent with values produced by the model in neighbouring grid points. There are many ways of assimilating observations into a model in such a way that the observed data remains consistent. This can include statistical techniques, like Kalman filtering, or weighting the difference between the model output and observation. In our model, we consider a variational approach which utilises optimal control theory to find a best fit. The first part of this chapter introduces the key elements of Optimal Control theory along with a simple numerical example. The latter section outlines how this is integrated into a Data Assimilation approach, which is then applied to the linearised Shallow Water Equations in the next chapter.

4.2 Optimal Control

In this study, we use a variational scheme where we express the difference as a 'cost functional', a measure of the difference between the model and observations, which is then minimised in order to get the closest fit. This utilises optimal control theory, where we find the function (called our 'control' variable) from a set of permissible functions that minimises our cost function subject to some constraint. The control variable is a parameter which affects the output of our system as it changes, such as initial condition data, or bathymetry data. Subsequently, minimising the cost functional would mean picking appropriate values for these such that the output generated by the model is as close to the observed data as possible, where the constraint is our system of equations. Therefore the control out of the admissable set of all possible controls that minimises the cost functional, is considered 'optimal'.

Consider a dynamical system of the form

$$\frac{d\boldsymbol{x}}{dt} = \boldsymbol{a}(\boldsymbol{x}(t), \boldsymbol{u}(t), t), \tag{4.1}$$

subject to some state constraints, such as initial conditions

$$\boldsymbol{x}(t_0) = k, \tag{4.2}$$

Where t_0 and t_T represent the initial and final times. Then the cost functional J can be defined as

$$J = \int_{t_0}^{t_T} g(\boldsymbol{x}(t), \boldsymbol{u}(t), t)dt \tag{4.3}$$

where g is a scalar function. If J is defined for x and $x + \delta x$, where δx is called the variation in x, then the *increment* of J is

$$\Delta J = J(x + \delta x) - J(x) \tag{4.4}$$

The first variation of J, δJ, is expressed as

$$J'(x, \delta x) = \lim_{\varepsilon \to 0} \frac{J(x + \varepsilon \delta x) - J(x)}{\varepsilon}$$

As we can see, this is the Gateaux derivative for the functional J. Therefore the optimal control for the system, is the 'extremal' of the system (1), u^*, such that $J'(x, \delta x) = 0$. Our goal is to find such a u^*.

In order to include the constraints of our state equation, we form an *Augmented Functional* J_a, where the integrand is the Lagrangian of g subject to the constraint $\dot{\boldsymbol{x}}(t) - \boldsymbol{a} = 0$,

$$J_a = \int_{t_0}^{t_T} \Big(g(\boldsymbol{x}(t), \boldsymbol{u}(t), t) + \boldsymbol{p}(\frac{d\boldsymbol{x}}{dt} - \boldsymbol{a}(\boldsymbol{x}(t), \boldsymbol{u}(t), t)) \Big) dt, \tag{4.5}$$

where p is a Lagrange multiplier, also known as the *adjoint variable*. It is often useful to express the integrand $g + \boldsymbol{p}(\dot{\boldsymbol{x}}(t) - \boldsymbol{a})$ as $\mathcal{H}$, the *Hamiltonian*. Since the choice of adjoint variables is arbitrary, we pick them such that $\delta J_a = 0$. Writing this out in full (as in [11]) and then equating the coefficients of $\delta \boldsymbol{x}$, $\delta \boldsymbol{p}$ and $\delta \boldsymbol{u}$ respectively to zero, yields

$$\frac{d\boldsymbol{x}}{dt} = \frac{\partial \mathcal{H}}{\partial \boldsymbol{p}} \tag{4.6}$$

$$\frac{d\boldsymbol{p}}{dt} = -\frac{\partial \mathcal{H}}{\partial \boldsymbol{x}} \tag{4.7}$$

$$0 = \frac{\partial \mathcal{H}}{\partial \boldsymbol{u}} \tag{4.8}$$

Together, these form the necessary conditions for the optimal control u^*, where (6) represents the system constraint, subject to the (7), the adjoint equation, and (8) is an algebraic relation used to rewrite (6) and (7) as a system of ODEs that can then be solved for the adjoint variables.

The boundary conditions for our dual system are chosen such that $\delta\boldsymbol{x}$, $\delta\boldsymbol{p}$ and $\delta\boldsymbol{u}$ are set to zero, and so minimise our cost function. The conditions used to derive them depend on whether we have a final time that is fixed or free.

Using initial data and/or other given information, we can then solve for the control u^* that minimises our cost function and hence provides a best fit between our model and the observation data.

4.2.1 Optimal Control Example[11]

Consider the system

$$\frac{dx_1}{dt} = x_2(t), \tag{4.9}$$

$$\frac{dx_2}{dt} = -x_2(t) + u(t), \tag{4.10}$$

subject to the initial condition

$$\boldsymbol{x}(0) = \mathbf{0}, and a final time that is not fixed.$$

We wish to conserve its control effort, such that the performance measure (energy)

$$J(u) = \int_{t_0}^{t_T} \frac{1}{2} u^2 dt,$$

is minimised. We begin by forming the Hamiltonian

$$\mathcal{H} = \frac{1}{2}u^2(t) + p_1(t)x_2(t) - p_2(t)x_2(t) + p_2(t)u(t), \tag{4.11}$$

and so our adjoint equations are

$$\frac{dp_1}{dt} = -\frac{\partial \mathcal{H}}{\partial x_1} = 0, \tag{4.12}$$

$$\frac{dp_2}{dt} = -\frac{\partial \mathcal{H}}{\partial x_2} = -p_1(t) + p_2(t), \tag{4.13}$$

and our algebraic relation relating u and p is

$$0 = \frac{\partial \mathcal{H}}{\partial u} = u(t) + p_2(t). \tag{4.14}$$

From the latter we can see that $u(t) = -p_2(t)$ and substituting this back into our state equations gives

$$\frac{dx_1}{dt} = x_2(t) \tag{4.15}$$
$$\frac{dx_2}{dt} = -x_2(t) - p_2(t). \tag{4.16}$$

(12),(13),(15) and (16) are a set of $2n$ linear first-order homogeneous ODEs which can be solved, giving us four constants of integration respectively;

$$\frac{dx_1}{dt} = c_1 + c_2(1 - e^{-t}) + c_3(-t - \frac{1}{2}e^{-t} + \frac{1}{2}e^{t}) + c_4(1 - \frac{1}{2}e^{-t} - \frac{1}{2}e^{t}) \tag{4.17}$$
$$\frac{dx_2}{dt} = c_2e^{t} + c_3(-1 + \frac{1}{2}e^{-t} + \frac{1}{2}e^{t}) + c_4(\frac{1}{2}e^{-t} - \frac{1}{2}e^{t}) \tag{4.18}$$
$$\frac{dp_1}{dt} = c_3 \tag{4.19}$$
$$\frac{dp_2}{dt} = c_3(1 - e^{t}) + c_4e^{t}. \tag{4.20}$$

And now we can evaluate our constants using our boundary conditions. $\boldsymbol{x}(0) = \mathbf{0}$ gives us $c_1 = c_2 = 0$. Using the conditions for the adjoint system for a free final time outlined in table 5.2 in [11], the latter two constants can be found by substituting these into (19) and (20). This yields $c_3 = -7.289$ and $c_4 = -6.103$, and so the optimal trajectory that minimises our cost function is

$$\dot{x_1}(t) = 7.289t - 6.103 + 6.696e^{-t} - 0.593e^{t} \tag{4.21}$$
$$\dot{x_2}(t) = 7.289 - 6.696e^{-t} - 0.593e^{t}. \tag{4.22}$$

4.3 Data Assimilation

As previously stated, the core purpose of data assimilation is the integration of observed data into a mathematical model. Perhaps the most apt description is that it is the 'science of compromises'. Natural dynamical systems can be modelled by a system of PDEs. Therefore if we have enough observational data, we can solve this system and accurately forecast the state at a given time. However, realistically the information available is often orders of magnitude smaller than the necessary boundary or initial data required to fully solve the system. Natural systems like ocean currents, waves or climate can have numerous parameters and variables that cannot always be quantified at every point necessary, they are often nonlinear, and the processing power needed to solve a system of such a high dimension is often not available. The problem can be summarised thus: what is the optimal way to integrate the information we *do* have?

Data Assimilation is used in geophysical applications like meteorology and oceanography etc., where the objective can be either forecasting the system evolution in future time, or solving the inverse problem; estimating initial and boundary conditions of the model based on its evolution. There are different mathematical approaches to how this is done. Essentially, the purpose is to generate an output using a mathematical model, and then 'correct' it using actual observed data at that point in space and/or time. A probabilistic approach may involve weighted correction based on probability of accuracy of the measurement. Another approach, introduced in the previous section is the variational approach, where the difference between the observation and the system output (also known as the residual) is expressed as a cost function, and then minimised. A simple example would be the method of least squares. The variational scheme used in Data Assimilation in three space variables and time is known as *4D-VAR.*[2]

All forecast models contain unknown parameters that when adjusted can affect the outcome of the model. The purpose of variational data assimilation is to estimate this set of parameters by minimising the residual; the difference between the model solution and observed data it is supposed to predict. The optimisation problem is then to adjust these parameters until we get this best fit. In this study, the parameter we aim to predict is the initial condition of a travelling wave (we also derive the mathematical algorithm for bathymetry as the control variable), based on observed data measured at a later point in time. So essentially, we are solving the inverse problem.

This presents us with a good question: if the observations we use have already happened (clearly since we have measured them), then what is the purpose of estimating initial conditions for a model that has already evolved? So for example, estimating the initial conditions for a weather front which has already occurred. This is answered by the consideration that our objective is to *optimise* a model using past information before we use it to forecast in future time. This way we can optimise a model many times using as many observations as feasible before we apply it to a domain where the aim is to predict an outcome accurately. Indeed, using paleo climate data assimilation can be significant towards predicting our future climate. One such simulation is called an analysis cycle, and often, the results of one cycle are used as inputs for the next.

Other important considerations within data assimilation include making sure our estimates are consistent with the laws of physics, and realistic in terms of our model evolution. For example, can an optimal estimate be found within a realistic amount of time? Does our model account for uncertainties and errors, whether they be in measurements or our numerical model?

In addition to these, a primary requirement for the variational algorithm is that our system can be solved in forward time from some initial time t_0 till a final time t_T, given an initial condition 'guess' from the set $\mathcal{S}$ of admissible initial conditions (meeting the afore-mentioned physical requirements). This forward model should then yield system outputs at some point in our spatial and time domain, that can be compared with our observations.

Secondly, we need a cost function J that is a function of our initial condition, model prediction and observations. Current variational algorithms in Numerical Weather Prediction (NWP) also sometimes include background state information found from previous forecast models, as well as error-covariance matrices. However, in this study we consider a simple case omitting these considerations, with the intent that they can be accounted for at a later stage once we have a successful algorithm for the simple case. For our cost function, we utilise the method of least squares and express it as a quadratic, as then it is guaranteed to have a minimum.

Once we have solved a forward model and formulated our cost function, the next step is to minimise it. The process (as outlined in the previous section) involves finding the initial conditions that minimise the cost function J. We start with an initial guess and then proceed to find each subsequent approximation using a descent algorithm. Within this algorithm, we do not necessarily need to find the value of J itself, rather just where it is at its minimum. Hence we consider where it is stationary, and thus where the gradient of J with respect to our inition condition is zero .We formulate an expression for $\nabla J(\eta_0)$ where $\eta_0(x,t)$ is our initial condition, in terms of its residual $m(x,t)-\eta(x0,t;\eta_0)$ where $m(x,t)$ is our observational data at some measurement point x_0, and $\eta(x0,t;\eta_0)$ is the prediction of our state generated by the forward solver at x_0 and time t, given our

current guess η_0). There are multiple ways to evaluate this, and here we use the adjoint technique.

The adjoint technique is an efficient way of calculating $\nabla J(\eta_0)$, or rather the necessary conditions for $\nabla J(\eta_0) = 0$. Rather than calculate an expression for $\nabla J(\eta_0)$ at every time step which can be costly and inefficient, using the the adjoint method we first solve the model in forward time from t_0 to t_T, and store the residual at every t. The second step is to solve it backwards starting from t_T, using an alternative set of "adjoint variables" in terms of the time step, $\tau = T - t$. Solving this adjoint system in backward time allows us to combine the residual obtained from our forward solver at time t with the adjoint information at time τ. Both of these are required in the mathematical expression we derive for $\nabla J(\eta_0)$, and it is much more efficient to do this using the adjoint system. Since we want a guess for our state at time t_0, we use these values to derive an expression for $\nabla J(\eta_0) = 0$ at t_0, and hence the optimal estimate for $\eta_0(x, t)$ to be used as the initial guess in the next iteration of our steepest descent algorithm.

This presents a general overview of the variational data assimilation algorithm, assuming error-free measurements. The steepest descent algorithm and issues like smoothing are discussed in chapter 5.

Chapter 5

Data Assimilation for the Shallow Water Equations

5.1 Initial Condition as Control Variable

5.1.1 Linear Case

The goal is to optimise estimate of Initial Condition from measurement of wave height $\eta(x_0, t)$. Our linearised system is

$$\begin{aligned} \frac{\partial \eta}{\partial t} + \frac{\partial u}{\partial x} &= 0, \\ \frac{\partial u}{\partial t} + \frac{\partial \eta}{\partial x} &= 0, \\ \eta(x, 0) &= \eta_0(x), \\ u(x, 0) &= 0, \end{aligned} \tag{5.1}$$

where we implement periodic boundary conditions in space, on a domain $x \in [-L, L]$, where $L >> 1$. The cost function representing the difference between our observation at some point x_0 at time t, and the output of our system at position x_0 at time t given the initial condition $\eta_0(x)$ is

$$J(\eta_0) = \frac{1}{2}\int_0^T \left(m(x_0,t) - \eta(x_0,t;\eta_0)\right)^2 \, dt.$$

We want to minimise $J(\eta_0)$, for $\eta_0 \in X$, and we do this by solving $\nabla J(\eta_0) = 0$. As discussed in chapter four, we look to a dual formulation for our system with a new set of suitably chosen adjoint variables that allow us to compute this gradient efficiently. to begin with, let $\eta'(x)$ be the wave height for the linearised shallow water equation with some perturbed initial condition $\eta_0'(x)$ .As we begin with the linear system in this case, this is equivalent to (5.1). We express the gradient of the cost function in terms of the Gateaux derivative, with a perturbation of scale ε in the direction of the linearised height η_0',

$$J'(\eta_0; {\eta'}_0) = \frac{J(\eta_0 + \varepsilon {\eta'}_0) - J(\eta_0)}{\varepsilon}$$

And use the Taylor expansion of the first term to order $\mathcal{O}(\varepsilon^2)$ to rewrite it as

$$J'(\eta_0; {\eta'}_0) = \int_0^T \left(m(t) - \eta(x_0,t;\eta_0)\right)\eta'(x) \, dt.$$

We want to find the dual (adjoint) variables $\eta*$, $u*$, and so we formulate the following duality relation, using the constraints of our linearised system,

$$\int_0^T \int_{-L}^{L} \eta^* \Big(\frac{\partial \eta'}{\partial t} + \frac{\partial u'}{\partial x} \Big) + u^* \Big(\frac{\partial u'}{\partial t} + \frac{\partial \eta'}{\partial x} \Big) \; dx \; dt = 0 \; .$$

Additionally, if we use the Riesz representation theorem, we have an equivalent form for $\nabla_x J$:

$$J'(\eta_0; \eta'_0) \; = \; \langle \nabla J, \eta'_0 \rangle_{L_2} \; = \; \int_{-L}^{L} \nabla^{L_2} J \; \eta'_0 \; dx,$$

We integrate the duality relation by parts in both space and time to derive an expression of the form $\int_0^L \nabla J \; \eta_0{}' \; dx$, by appropriately choosing constraints for the adjoint variables that allow us to do so, and subsequently find a formulation for for $\nabla_x J$. Whatever variable will be multipled by η' in the former integral will be our gradient.

Integrating by parts, we get

$$- \int_0^T \int_{-L}^{L} \eta' \Big(\frac{\partial \eta^*}{\partial t} + \frac{\partial u^*}{\partial x} \Big) + u' \Big(\frac{\partial u^*}{\partial t} + \frac{\partial \eta^*}{\partial x} \Big) \; dx \; dt + \int_{-L}^{L} \Big(\eta' \eta_0^* \mid_T - \eta' \eta^* \mid_0 + u' u* \mid_T - u' u^* \mid_0 \Big) \; dx = 0.$$

Using the fact that $\eta' \mid_{t=0} = \eta_0(x)$, $u' \mid_{t=0} = 0$, and assuming periodic boundary conditions in x, and we pick our adjoint variables such that

$$\begin{aligned} \frac{\partial \eta^*}{\partial t} + \frac{\partial u^*}{\partial x} &= (m(t) - \eta(x,t;\eta_0)) \delta(x - x_0) \; , \\ \frac{\partial u^*}{\partial t} + \frac{\partial \eta^*}{\partial x} &= 0 \; , \\ \eta^*(x,T) = \; u^*(x,T) &= 0 \; , \end{aligned} \tag{5.2}$$

where we have used the fact that we are free to set the initial conditions for the adjoint variable. And so our duality relation is reduced to

$$J'(\eta_0;\eta_0') \;=\; \int_0^T \big(m(t)-\eta(x_0,t;\eta_0)\big)\eta'(x)\;dt \;=\; -\int_{-L}^{L}\eta^*(x,0)\eta_0'(x)\;dx,$$

And thus we have $\nabla^{L^2}J = -\eta^*(x,0)$.

5.1.2 Non-linear Case

We now consider the case where we solve the non-linear shallow water equations in forward time. Our equations are

$$\begin{aligned}\frac{\partial h}{\partial t}+\frac{\partial}{\partial x}\Big(hu\Big)&=0,\\ \frac{\partial u}{\partial t}+\frac{\partial}{\partial x}\Big(\frac{1}{2}u^2+\eta\Big)&=0,\end{aligned}\tag{5.3}$$

where we impose $g=1$. When we formulated our Gateaux derivative in the linear case, we linearised about $\eta=0$, $u=0$ to get a perturbation in the η' and u' directions, where our linearisation was $u\to u'$ and $\eta\to\eta'$. In this case, we will linearise about the previous solution η and u, and so we have $\eta\to\eta+\eta'$ and $u\to u+u'$ where $\eta', u' << 1$. Replacing η and u in the nonlinear system with these, and discarding all non-linear terms in η' and u', we get

$$\begin{aligned}\frac{\partial h}{\partial t}+\frac{\partial}{\partial x}\Big(\eta' u+(1+\eta)u'\Big)&=0,\\ \frac{\partial u}{\partial t}+\frac{\partial}{\partial x}\Big(\eta'+uu'\Big)&=0.\end{aligned}\tag{5.4}$$

Using this, our duality relation becomes

$$\int_0^T \int_{-L}^{L} \eta^* \Big(\frac{\partial \eta'}{\partial t} + \frac{\partial \eta' u + (1+\eta)u'}{\partial x} \Big) + u^* \Big(\frac{\partial u'}{\partial t} + \frac{\partial \eta' + uu'}{\partial x} \Big) \, dx \, dt = 0.$$

We integrate by parts,

$$- \int_0^T \int_{-L}^{L} \eta' \frac{\partial \eta^*}{\partial t} + (\eta' u + (1+\eta)u') \frac{\partial \eta^*}{\partial x} + u' \frac{\partial u^*}{\partial t} + (\eta' + uu') \frac{\partial u^*}{\partial x} \Big) \, dx \, dt$$
$$+ \int_{-L}^{L} \Big(\eta' \eta_0^* \mid_T - \eta' \eta^* \mid_0 + u' u* \mid_T - u' u^* \mid_0 \Big) \, dx = 0,$$

and imposing $\eta^*(x,T) = \; u^*(x,T) = 0$ as before, we have

$$- \int_0^T \int_{-L}^{L} \eta' \Big(\frac{\partial \eta^*}{\partial t} + u \frac{\partial \eta^*}{\partial x} + \frac{\partial u^*}{\partial x} \Big) + u' \Big(\frac{\partial u^*}{\partial t} + (1+\eta) \frac{\partial \eta^*}{\partial x} + u \frac{\partial u^*}{\partial x} \Big) \, dx \, dt \; = \; - \int_{-L}^{L} \eta^*(x,0) \eta_0'(x) \, dx.$$

Choosing our adjoint system such that

$$\begin{aligned} &\frac{\partial \eta^*}{\partial t} + u \frac{\partial \eta^*}{\partial x} + \frac{\partial u^*}{\partial x} = \; (m(t) - \eta(x,t;\eta_0)) \delta(x - x_0) \; , \\ &\frac{\partial u^*}{\partial t} + (1+\eta) \frac{\partial \eta^*}{\partial x} + u \frac{\partial u^*}{\partial x} = \; 0 \; , \\ &\eta^*(x,T) = \; u^*(x,T) = \; 0 \; , \end{aligned} \tag{5.5}$$

once again we have

$$J'(\eta_0; \eta_0') \; = \int_0^T (m(t) - \eta(x_0, t; \eta_0)) \eta'(x) \; dt \; = \; - \int_{-L}^{L} \eta^*(x,0) \eta_0'(x) \; dx,$$

and subsequently $\nabla^{L^2} J = -\eta^*(x, 0)$. The main difference from our linear case is the form of the forward linearised system and the backward adjoint system to be solved, however the general process for our data assimilation algorithm remains the same.

5.1.3 Algorithm Summary

We solve these backwards in time to find $-\eta^*(x, 0) = \nabla^{L_2} J$. If we use a descent algorithm to nudge our initial condition such that $\eta^*(x, 0) = \nabla J = 0$, then the estimate that gives us this will be our optimal estimate. And so our objective is to iterate though different initial conditions until we can minimize $\nabla J = \eta^*(x, 0)$ to a sufficient tolerance level. And so we derive the following algorithm:

- Pick an initial estimate for $\eta_0^{(0)}(x)$.
- Solve the forward Initial Value Problem for (u, η) from 0 to T.
- Solve Adjoint Problem for (u^*, η^*) from T to 0 to find $\eta^*(x, 0)$.
- Define $\nabla^{L^2} J = -\eta^*(x, 0)$.
- Smooth $\nabla^{L^2} J \to \nabla^{H_p} J$ (see section 5.1.2)
- Compute the optimal time step τ_n through a line minimisation algorithm.
- Use a steepest descent algorithmn to compute $\eta_0^{(n+1)}(x) = \eta_0^{(n)}(x) - \tau_n \nabla^{H_p} J(\eta_0^{(n)}(x))$.
- Repeat until $\| \nabla^{H_p} J \| < \varepsilon$ for some small ε ($\| \eta^*(x, 0) \| \approx 0$).

5.1.4 Line Minimisation

Instead of arbitrarily picking some step size τ for our descent algorithm, we can use a lin minimisation algorithm to find the optimal step size that will allow us to find the best posible estimate for the next guess of $\eta_0(x)$. The optimal step size τ_n is one that will give us the maximum reduction in the cost function J. We want to choose it such that

$$\tau_n = \underset{\tau>0}{argmin}\ J\Big(\eta_0^{(n)} - \tau\nabla^{H_p} J(\eta_0^{(n)})\Big).$$

Our objective is to derive an expression for the above and solve it to find root τ.
Since we have defined $\nabla^{H_p} J(\eta_0^{(n)}(x)) = -\eta_0^*(x)$, the above expression simplifies to

$$\tau_n = \underset{\tau}{argmin}\ J(\eta_0^{(n)} + \tau\eta_0^*).$$

And so applying this at x_0 (where we have our observations) we have

$$G(\tau) = J(\eta_0^{(n)} + \tau\eta_0^*),$$

and the root of this function is our optimal stepsize τ. We find this root using the Matlab function *fminunc*, allowing us to find the local minimum for our function given a starting point τ_0, which we take as 2 for the first iteration, and then the optimal τ found at each nth step in the $(n+1)$th iteration. Applying this at every iteration in our data assimilation algorithm ensures we find the most efficient way to reach our optimised initial condition.

5.1.5 Smoothing the Cost Function Gradient

Another factor we take into account is to improve our formulation of the gradient by 'smoothing' it, transferring it to a Hilbert space H_p from the L_2 space we derived it in,

$$J'(\eta, \eta') = \langle \nabla^{L_2} J, \eta' \rangle_{L_2} = \langle \nabla^{H} J, \eta' \rangle_{H}.$$

To smooth the gradient, we take our gradient vector into Fourier Space, and use the following formula to dampen the frequencies k associated with our gradient for J above a designated value $\frac{1}{L}$,

$$\widehat{(\nabla^{H} J)}_k = \frac{1}{(1 + L^2 k^2)} \widehat{(\nabla^{L_2} J)}_k,$$

and then using the *ifft* Matlab function we take the inverse Fourier transform and bring our gradient back into real space. With the smoothed gradient, we should have a better approximation for the gradient in our steepest descent algorithm, and so get a better estimation for our initial condition.

Chapter 6

Kappa Test for Gradient used in Data Assimilation Variational Scheme

6.1 Kappa Test

The kappa test for the gradient of the cost function, is a test that ensures that our calculations for the gradient $\nabla J(\eta_0)$ are correct. This is obviously a necessary condition for the algorithm to work successfully. In our algorithm, we used two forms to calculate the gradient . The first is the Gateaux derivative,

$$J'(\eta_0, \eta_0') = \lim_{\varepsilon \to 0} \frac{J(\eta_0 + \varepsilon \eta_0{}') - J(\eta_0)}{\varepsilon} \tag{6.1}$$

where $\eta_0{}'$ is some perturbation direction vector and ε is a scalar representing the magnitude of the perturbation. The second representation we numerically calculated was the

Sobolev gradient form, which is equivalent to (1) by the Riesz representation theorem, where

$$J'(\eta_0; \eta'_0) \;=\; \langle \nabla_x J, \eta'_0 \rangle \;=\; \int_0^L \nabla_x J \; \eta'_0 \; dx. \tag{6.2}$$

And so, we define the kappa parameter as

$$\kappa(\varepsilon) = \varepsilon^{-1} \frac{J(\eta_0 + \varepsilon \eta_0') - J(\eta_0)}{\langle \nabla_x J, \eta'_0 \rangle}, \tag{6.3}$$

which is the ratio of the two equivalent representations of the derivative of the cost function. We calculate the value of $\kappa(\varepsilon)$ as $\varepsilon \to 0$. If our calculations for $J'(\eta_0)$ are correct, then as ε converges to zero, we expect $\kappa(\varepsilon) \approx 1$ for an intermediate range of ε, accounting for approximation and round-off errors.

We consider the case where η'_0 is the Gaussian initial condition $\eta_0(x)$ itself, and we conduct the test for both the linear and non-linear systems.

6.2 Numerical Results

The following four graphs show the results of the kappa test for a Gaussian η'_0, for $t \in [0, 2]$, $x \in [-8, 8]$, $N = 2000$, and our observations and model predictions are calculated at $x_0 = 1.5$. Our 'actual' initial condition is $e^{-\left(\frac{x}{0.2}\right)^2}$ for the linear system, and $0.1e^{-\left(\frac{x}{0.2}\right)^2}$ for the non-linear system. For our distorted initial condition we simply re-scaled the amplitude by 0.9.

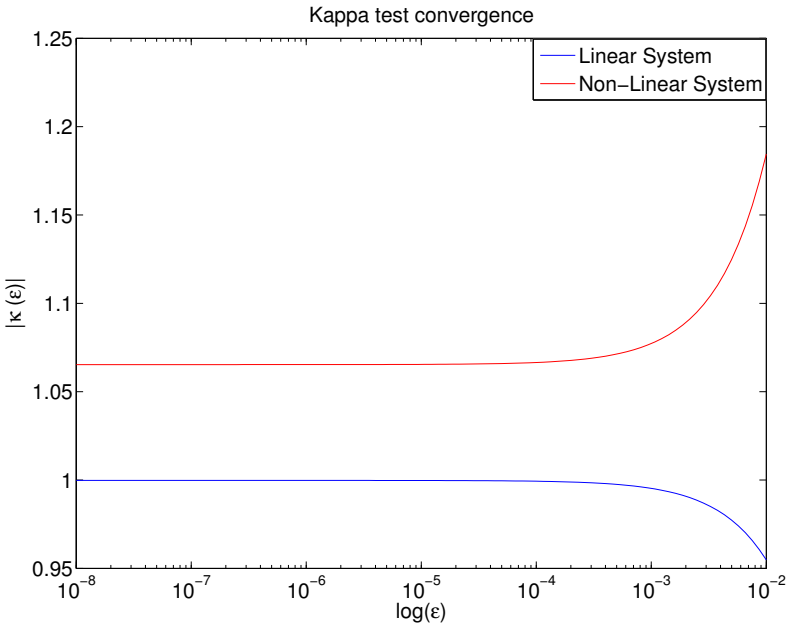

FIGURE 6.1: $\log \kappa(\varepsilon)$ vs $\log \varepsilon$. As we can see, as ε goes to zero, κ converges to approximately 1, indicating the different gradient formulations are equivalent to a high degree, in both the linear and non-linear systems

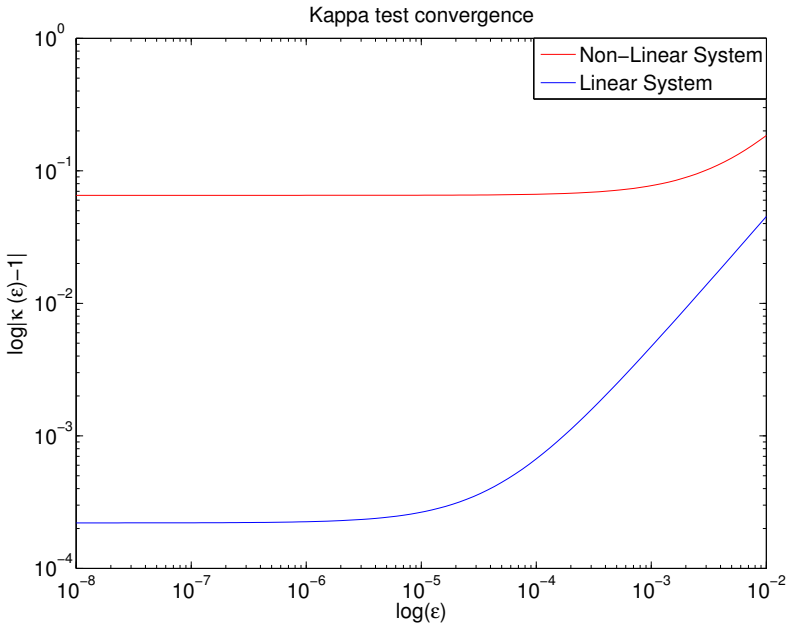

FIGURE 6.2: $|\kappa(\varepsilon) - 1|$ vs $\log \varepsilon$. This allows us to see how closely κ converges to 1: to $\approx 10^{-3}$ for the linear case, and to $\approx 10^{-1}$ for the non-linear case.

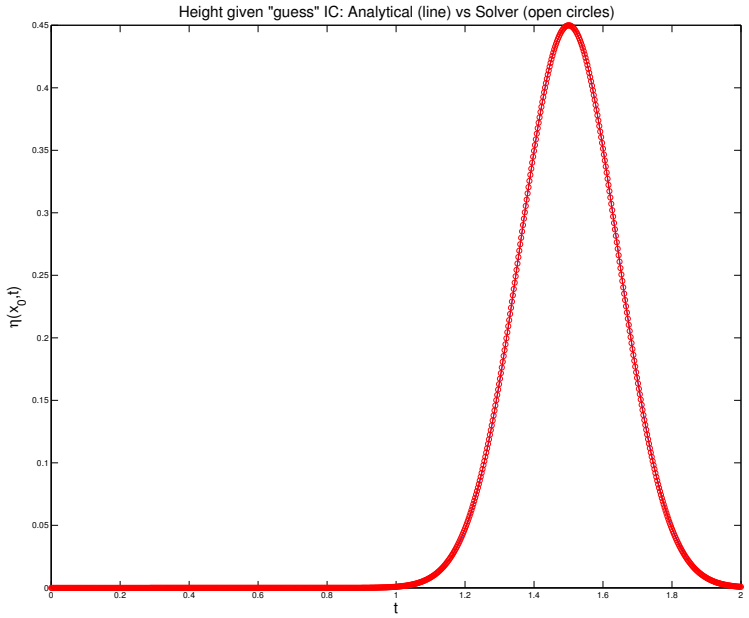

FIGURE 6.3: Linear Case: Analytical vs numerical height at x_0 given distorted IC. Results for analytically derived height and the output of the numerical scheme match almost exactly.

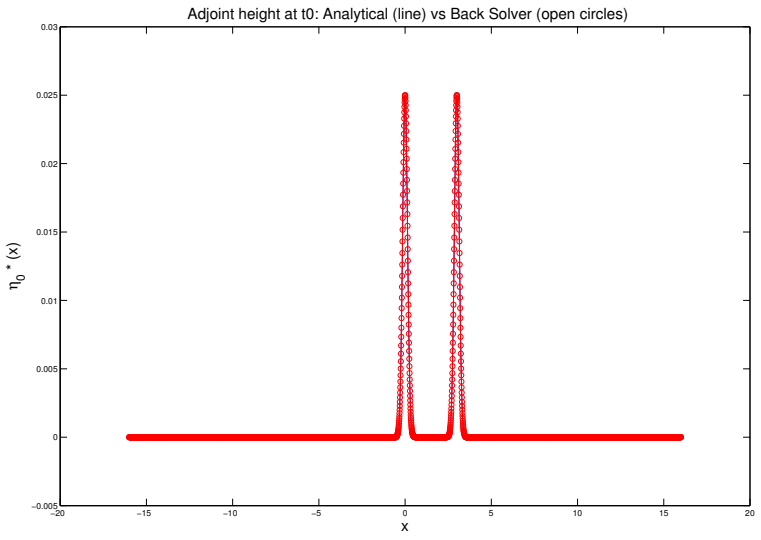

FIGURE 6.4: Linear Case: Analytical vs numerical adjoint height at t_0. Indicates our backward solver coincides very closely with the analytical solution for η^* over time.

Figures 6.1 and 6.2 shows the convergence of $|\kappa(\varepsilon)|$ and $|\kappa(\varepsilon) - 1|$ with respect to $\log(\varepsilon)$. As we can see, we see a strong convergence to ≈ 1 in figure 1, and ≈ 0 in figure 2. The errors in both cases can be attributed to numerical approximation, and it was found that by increasing our spatial grid points N, better convergence was achieved.

Figure 6.3 is the analytical vs numerical solution for the linear case, for height over time at x_0 given the distorted initial condition, $0.9 * \eta_0$. The numerical calculation is found using our finite difference Runge Kutta scheme from chapter three, and the analytical solution is derived using the formula

$$\eta(x_0, t) = \frac{1}{2}\eta_0(x_0 + t) + \frac{1}{2}\eta_0(x_0 - t). \tag{6.4}$$

As we can see, the plots for the numerical and analytical trends align almost perfectly. And so we can say that the forward numerical solver components passed our test.

Figure 6.4 compares the numerical and analytical solutions for our Adjoint system, and depicts the adjoint height at $t = 0$ starting at $t = 2$ with a zero initial condition, and going backwards in time. The analytical solution for the Adjoint system was derived using Duhamel's Principle [8]. If we have

$$\begin{aligned}
\frac{\partial \eta^*}{\partial \tau} + \frac{\partial u^*}{\partial x} &= f(x, \tau) \ , \\
\frac{\partial u^*}{\partial \tau} + \frac{\partial \eta^*}{\partial x} &= 0 \ , \\
\eta^*(x, 0) &= 0 \\
u^*(x, 0) &= 0 \ ,
\end{aligned}$$

we solve

$$\frac{\partial \eta^*}{\partial t} + \frac{\partial u^*}{\partial x} = 0 \ ,$$
$$\frac{\partial u^*}{\partial t} + \frac{\partial \eta^*}{\partial x} = 0 \ ,$$

with

$$\eta^*(x, s; s) = f(x, s)$$
$$u^*(x, s; s) = 0 \ .$$

And the general solution is then

$$\eta^*(x, \tau) = -F_1(x + \tau - s) + F_2(x - (\tau - s))$$
$$u^*(x, \tau) = F_1(x + \tau - s) + F_2(x - (\tau - s)) \ .$$

Using the initial condition,

$$-F_1(x) + F_2(x)) = f(x, s)$$
$$F_1(x) + F_2(x) \ = 0,$$

and so the solution is

$$\eta^*(x, \tau) = \frac{1}{2} \int_0^\tau f(x + \tau - s) + f(x - (\tau - s)) \ ds$$
$$u^*(x, \tau) = \frac{1}{2} \int_0^\tau -f(x + \tau - s) + f(x - (\tau - s)) \ ds.$$

When

$$f(x, \tau) = \big(m(T - \tau) - \eta(x, T - \tau; \eta_0)\big)\delta(x - x_0),$$

The analytical solution for the adjoint height at time zero is given by the formula

$$\eta^*(x,0) = \frac{1}{4}\Big((\eta_{ex}(x) - \eta_0(x)) + (\eta_{ex}(2x_0 - x) - \eta_0(2x_0 - x)) \Big), \tag{6.5}$$

where $\eta_{ex}(x)$ is the undistorted initial condition. And so we have

$$\eta^*(x,0) = \frac{1}{4}\Big((\eta_0(x) - 0.9\eta_0(x)) + (\eta_0(2x_0 - x) - 0.9\eta_0(2x_0 - x)) \Big). \tag{6.6}$$

Once again, we can see that the two solutions match almost exactly for the adjoint system as well.

To further illustrate how clearly the analytical and numerical solutions correspond, the following figures show the absolute difference between the two in both the distorted initial condition case and the adjoint height case:

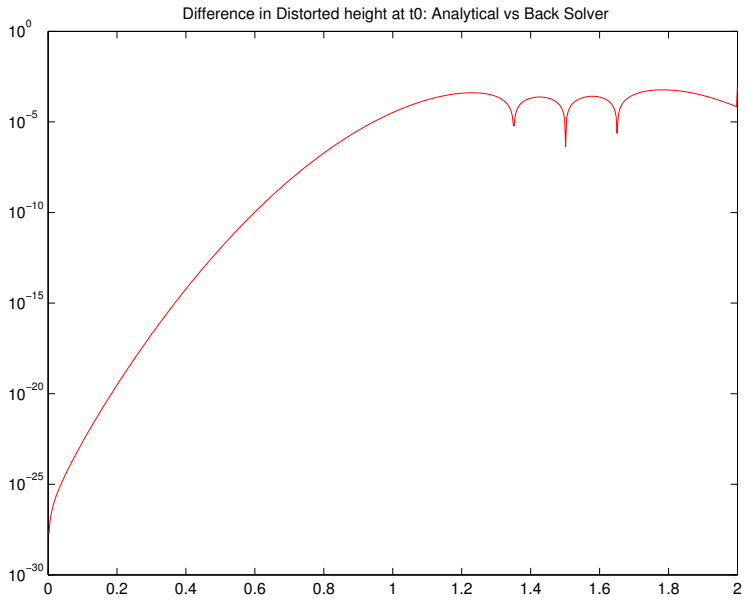

FIGURE 6.5: Difference in analytical vs numerical height at x_0 given distorted IC

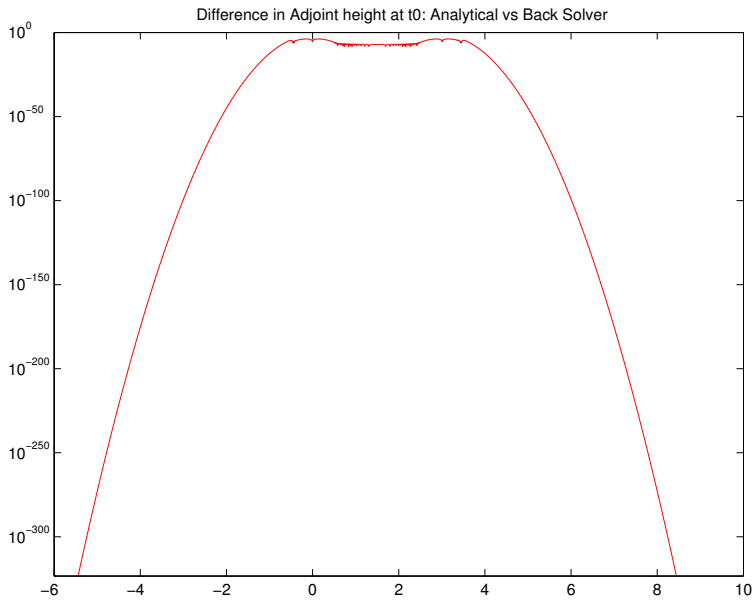

FIGURE 6.6: Difference in analytical vs numerical adjoint height at t_0

And so we conclude that our implementation of the gradient calculation in the kappa test is correct, as the components of each gradient match up almost exactly to their analytical solutions. With this in mind, we can finally proceed to the implementation of our data assimilation algorithm in chapter seven.

Chapter 7

Results

7.1 Data Assimilation for Linear Case

We began our analysis of the data assimilation scheme on the linear shallow water equations for different and multiple assimilation points. We chose a total time period from 0 to 2, and on a spatial domain of $[-16, 16]$, and our 'exact' initial condition is a Gaussian profile of the form ${\eta_0}^{ex}(x) = e^{-(\frac{x}{0.2})^2}$. We considered two cases of incomplete or distorted data, with a noise distortion and an amplitude distortion. The noise distortion was achieved using the Matlab function *agwn*, which adds Gaussian white noise to our signal, using a signal to noise ratio of 20. To smooth this slightly, we filtered the highest frequencies, using a filtering ratio of 0.4.This was achieved by taking the function into Fourier space, and then setting the highest 40% of frequencies to zero. The amplitude distortion is found by simply rescaling the exact initial condition by a ratio of 0.9.

We consider a total of eight different cases, with multiple x_0 assimilation points for both the noise and amplitude distortions. To illustrate the results we plot the trends in the cost function, and then take the best case result for both distortion types, and compare our optimised initial condition with our 'exact' undistorted initial condition.

In the noise distortion case, the cost function is proportional to the number of observations, and increases as more are added, as seen in figure 7.2, which shows the normalised cost function (error divided by the number of observation points) . The minimum cost function is indeed when we have a single observation point. This is an expected result; without normalisation, the total error should increase proportionally to the number of points, but as we can seem the increase in cost by adding more points is less, as more are added. So while the cost does increase, even when normalised by the number of points, it increases by a lower rate the more points we add, at least in the linear case.

There is a similar trend for the amplitude distortion; the cost increases as more than one point is added, but very marginally so thereafter . This can be seen in figures 7.1.

We consider the hypothesis that the number of assimilation points is only significant when the choice of measurement points is based on observation of the full wave; An analysis of the wave propagation in both cases showed that only points in the range $[0.5, 1.5]$ saw full information in the total time $T = 2$. And so the observation points in that range will provide more information than outside, which is why the cases where we take points outside this range do not give as good a result as when all x_0 points are within this range. And so we can stipulate that the more points we have within this range, the better our result. To check this we consider the x_0 assimilation points illustrated in figures 7.3 and 7.4, for both cases ,with five observation points each, but with one case considering the end points of our range and the latter within it. However, in the linear case we did not notice any significant evidence that having more points in this range decreases the overall error in the cost function.

We have analysed how well the cost function was reduced, as was the primary objective

of this algorithm. Now we consider how sufficiently this has allowed us to optimise our distorted initial conditions. We consider the case where we pick five assimilation points in the range $[0.5, 1.5]$.

As we can see in figure 7.5, the optimised initial condition has moved closer to the exact initial condition. While it could be expected to conform more closely to the exact Gaussian profile, it should be noted that a smaller wave has been created in the optimised profile, to the right of the initial exact wave. This additional wave is an 'alternative' form created by the optimisation process; the optimised height does not necessarily confirm to the exact shape of the undistorted condition, and instead converges to a profile that most minimises the error at the assimilation points. In this case, adding a smaller wave form serves this purpose as the cost function is sufficiently reduced. For the noise distortion, we see a similar result in figure 7.6, except instead of an additional wave form we have some additional noise added to our optimised initial height, to the right of the Gaussian.

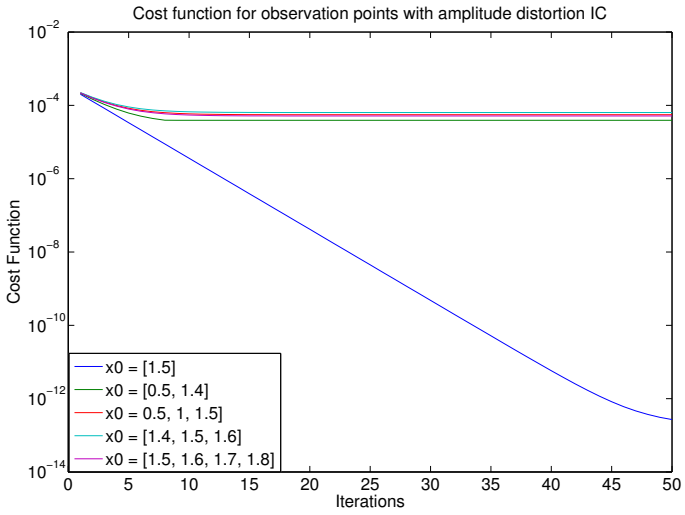

FIGURE 7.1: Linear Case: Cost Function normalised by number of observation points with amplitude distortion. Increasing the number of observation points (where points are chosen at any point in our domain) beyond a single point increases the error, but marginally with increasing points after that.

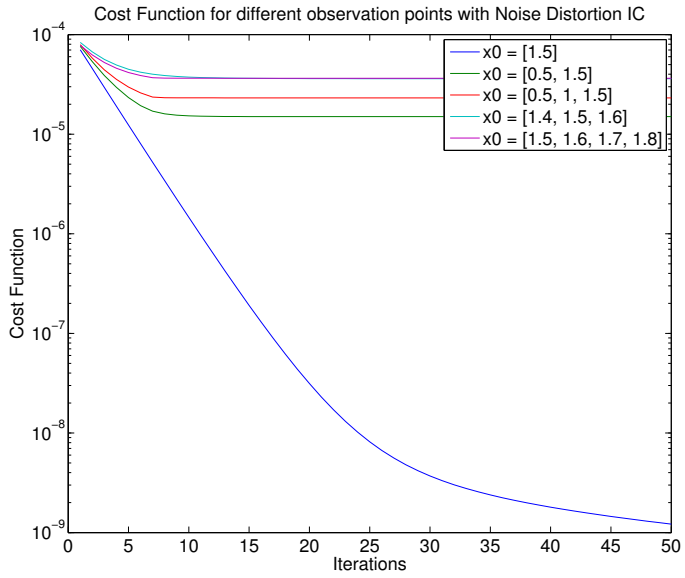

FIGURE 7.2: Linear Case:Cost Function normalised by number of observation points with noise distortion. Increasing the number of observation points (where points are chosen at any point in our domain) beyond a single point increases the error, but marginally with increasing points after that.

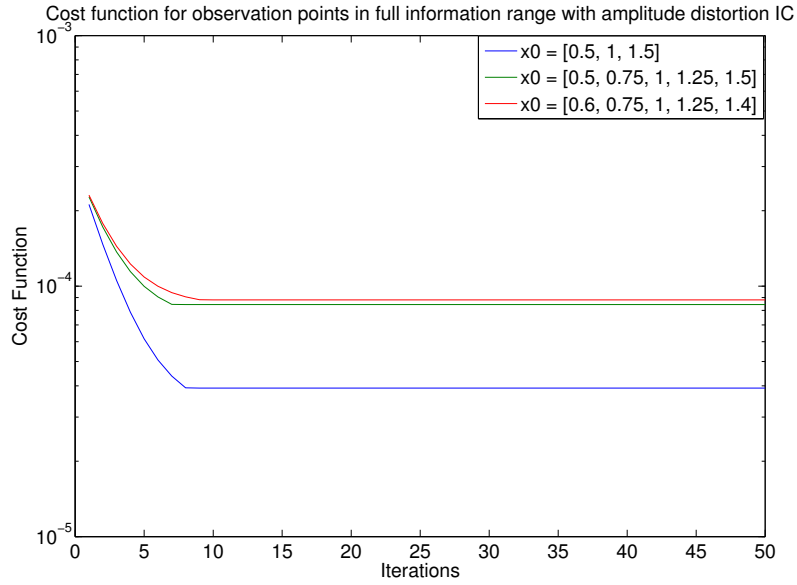

FIGURE 7.3: Linear Case: Cost Function normalised by number of observation points with Amplitude Distortion, for points in full observation range.

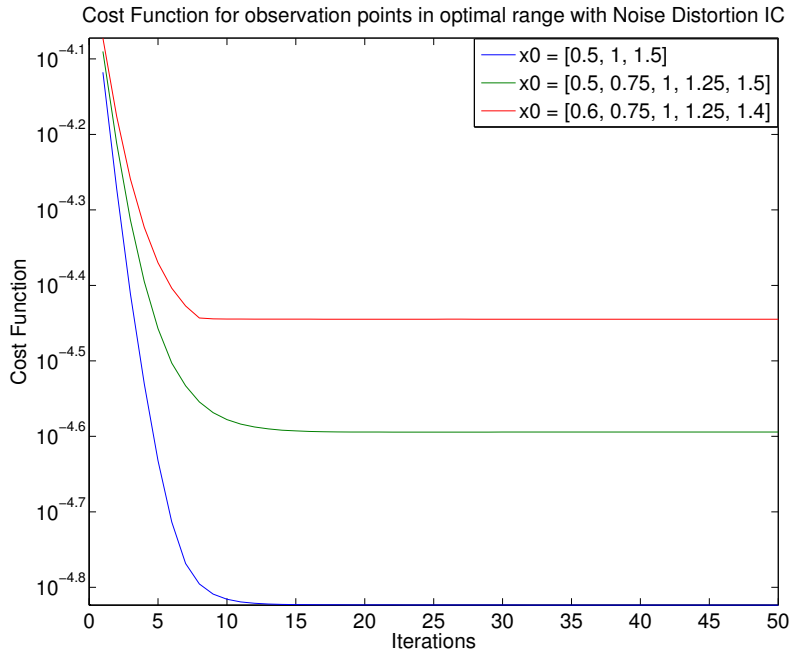

FIGURE 7.4: Linear Case: Cost Function normalised by number of observation points with Noise Distortion, for points in full observation range.

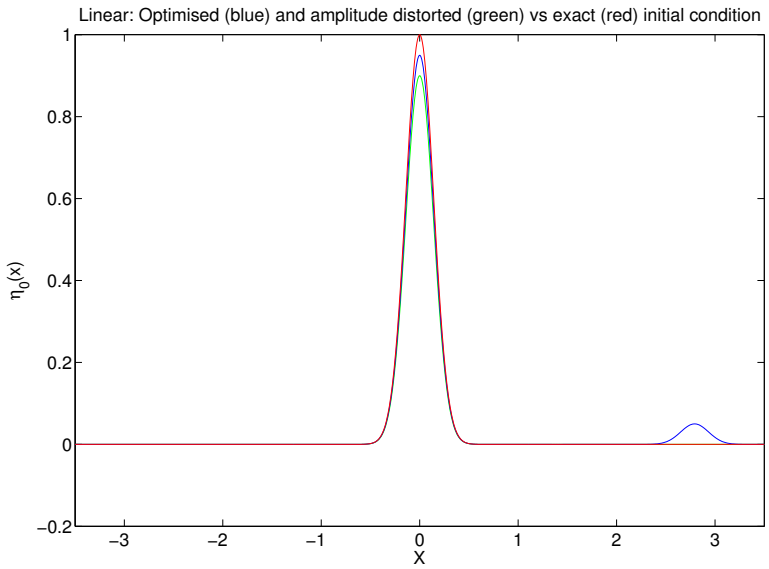

FIGURE 7.5: The optimised initial condition with amplitude distortion has an additional smaller wave to the right, in order to minimise the error at the assimilation points when it propagates.

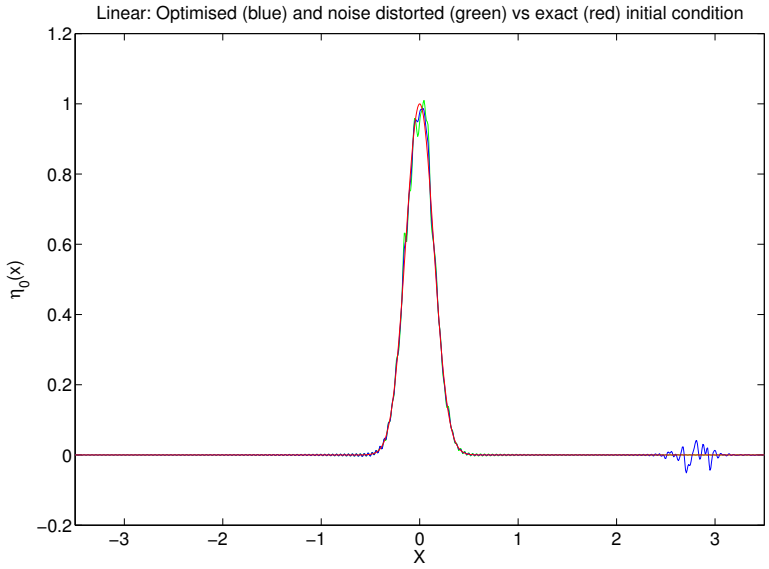

FIGURE 7.6: The optimised initial condition with noise has some additional noise to the right of the wave, in order to minimise the error at the assimilation points when it propagates.

7.2 Data Assimilation - Non-Linear Case

To further our analysis, we consider the results of our data assimilation algorithm applied to the fully non-linear shallow water equations. Our initial parameters remain the same, however the only significant change we make is to scale our initial condition height by a further factor of 0.1. This is because in the non-linear case, the balance of momentum equation is essentially Burger's equation, and due to the additional $u\frac{\partial u}{\partial x}$ term, as our wave propagates over time, without a viscosity term we see that the velocity u at the crest of the wave starts to exceed that of points close to its base, and we essentially see our wave 'breaking', if we do not re-scale. Since an amplitude of 0.1 is still large enough to produce some wave steepening we can no longer directly compare the linear and non-linear solution, however we can still make a qualitative assessment of the results Additionally, we checked that the non-linear assimilation results converged to the linear results in the limit of small initial wave amplitudes.

There is significant change in the normalised cost function trend as we increase the number of iterations, than in the linear case. In the previous case, we saw that while the cost function was sufficiently minimised, the normalised cost increased with the number of observation points, though by decreasing amounts as more points were added, and considering points in the full observation range did not make any difference. Now, as we see in figures 7.7 and 7.8, as the number of iterations increases, having less points does not necessarily mean the cost function is lower. In the amplitude distortion case in figure 7.7, the highest cost is when we have two points. This is also true for figure 7.8 where a single assimilation point has the highest cost function error. When we consider the cost function for observations in the full information range of points in figures 7.9 and 7.10, we immediately see that having more points gives a significantly lower cost, in both distortion cases. This potentially suggests that the error between observation and

system output for wave height does not stay constant, and may not be independent of the choice of our observations. This is likely due to the fact that in the linear case the solution is stationary in the frame of reference of the moving wave, while in the non-linear case the wave profile changes over time. Thus, unlike the linear case, in the nonlinear case adding more observation points provides more information about the waveform and hence the dynamics of the wave propagation. This is illustrated in figure 7.13, where we see a comparison between wave peaks at the final time T between the linear and non linear wave heights. As we can see, the non-linear wave profile are steeper and more skewed than the linear case.

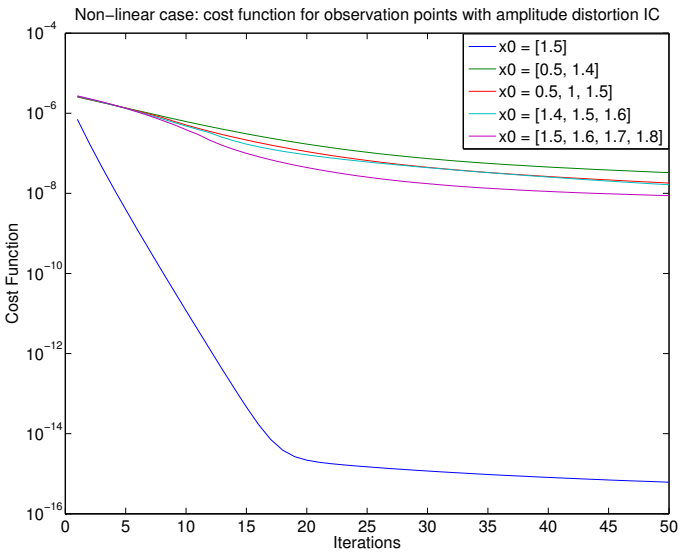

FIGURE 7.7: Non-linear case: Cost function with amplitude distortion. Aside from a single assimilation point case, adding more points seems to decrease the error (where points are chosen at any point in our domain).

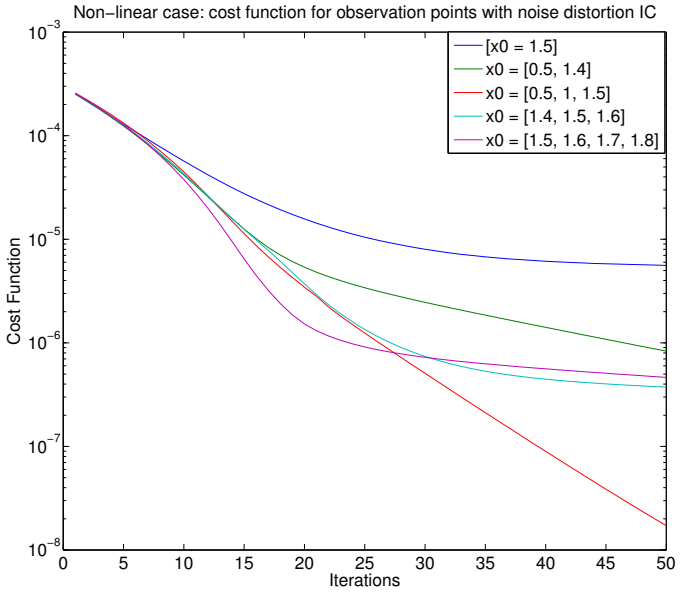

FIGURE 7.8: Non-linear case: Cost function with noise distortion. Adding more points decreases the error in the cost function (where points are chosen at any point in our domain).

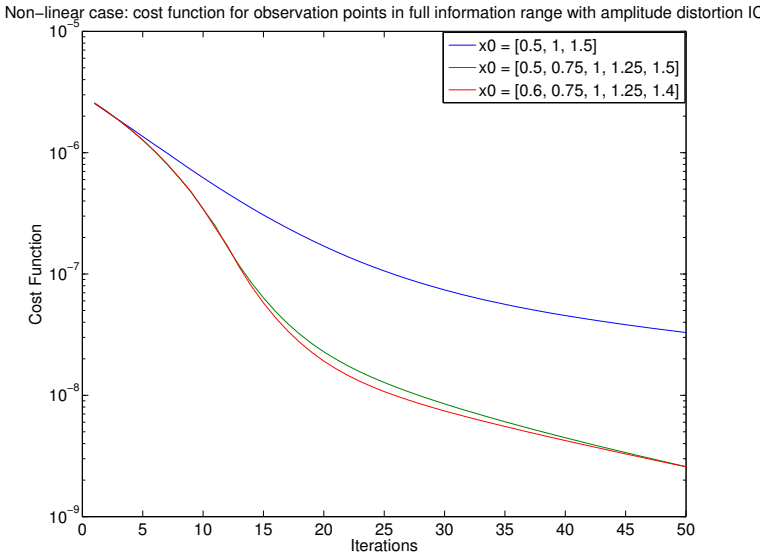

FIGURE 7.9: Non-linear case: Cost function with amplitude distortion. Increasing the number of observation points in full observation range reduces the cost function as more points are added.

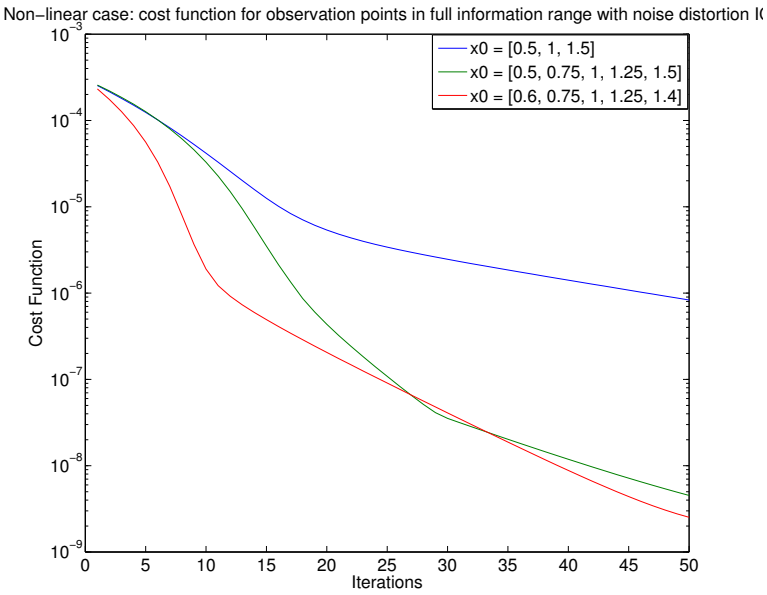

FIGURE 7.10: Non-linear case: Cost function with noise distortion. Increasing the number of observation points in full observation range reduces the cost function as more points are added.

On comparing our optimised inital condition after 50 iterations, and the exact initial conditions for the non-linear case, we see that the algorithm has enabled the distorted initial conditions to successfully converge closely to the exact initial height. In figure 7.11 we see for the amplitude distortion case that the two match almost exactly, and in figure 7.12 most of the noise distortion has been smoothed out. In the non-linear case, we do not see the waves and distortions created to the right of the initial curve in the optimised initial condition, like we did in the linear case.

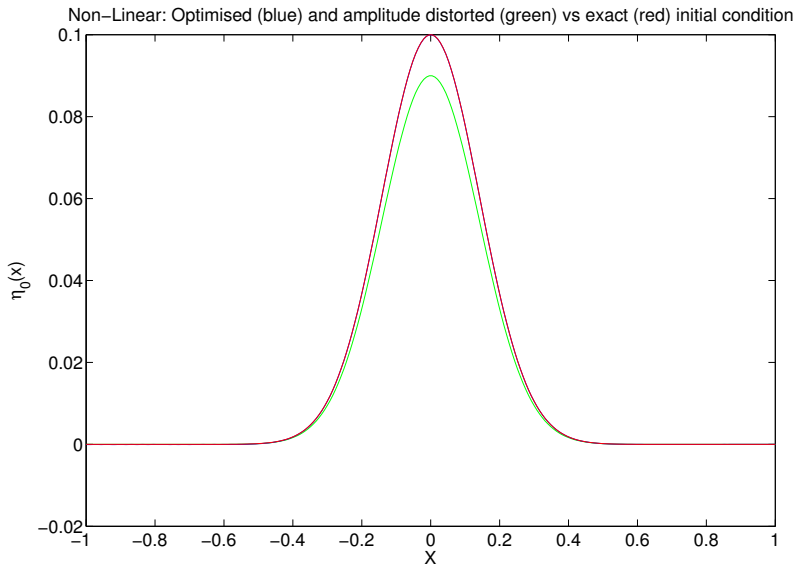

FIGURE 7.11: The optimised and exact initial conditions match almost exactly.

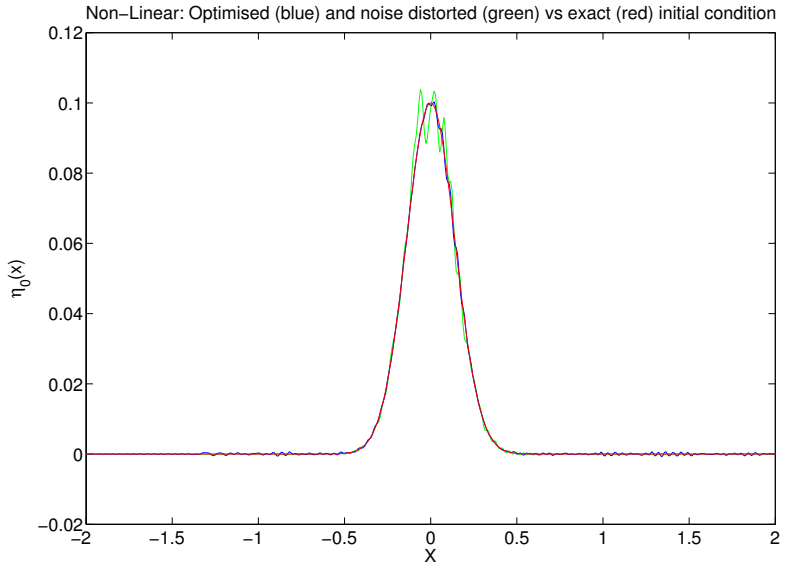

FIGURE 7.12: The noise distortion has been smoothed out in the optimised initial condition to a high degree.

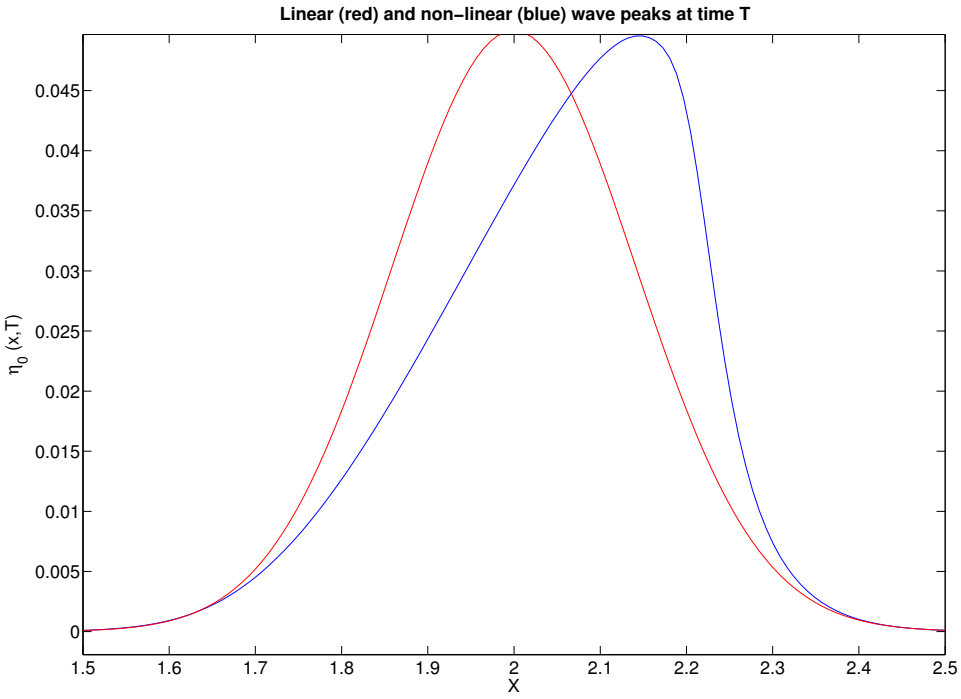

FIGURE 7.13: Comparison of wave profiles in linear and non-linear case. As can be seen, the non-linear wave is more steep, and does not translate a fixed form over time as in the linear case, hence adding more points decreases our overall error in the cost function in the non-linear case, whereas it does not prove to be significant in the linear case.

Chapter 8

Conclusions and Future Considerations

8.1 Conclusions

In conclusion, we can evaluate the efficacy of our data assimilation scheme through how closely our optimal estimates converge to the exact inital height, and the successful minimisation of the cost function. As seen in the results, the cost function is effectively minimised, across both both across linear and non-linear cases for our shallow water equations, and across different distortions (amplitude and noisy).

However, trends in cost function vary significantly across linear and non linear case. In linear case, we see the predictable result that after normalisation, adding more points increases error but less so as an increasing number of points is added; more points do not necessarily provide more information. This is not the case in the non-linear scheme. For the linear case, the fixed wave form simply translates. However for the nonlinear case the waveform changes shape. This means that adding more observation points in the full observation range definitely provides more information, unlike the linear case. This is consistent across both types of distortions in the initial condition.

This additional information added by the wave form in the non-linear case may also contribute to the better convergence of the optimised initial condition, than in the linear. While the cost function is sufficiently minimised in the latter, the algorithm does so by creating additional distortions in the optimised guess for the initial condition; the algorithm's primary objective is to create an initial condition profile that will generate the lowest error at our assimilation points, and so does not necessarily account for how closely this guess converges to the exact initial height profile. Setting constraints on where the optimised height can create this profile may allow for closer convergence to the exact height in future schemes.

This result suggests a relationship between the choice of observation points and the error between the actual and mathematically computed wave heights, where the qualitative information (dependent on the wave form) provided at each point impacts the additional increase in total cost. In theory, multiple observation points should not create an over-determination issue for the least squares fit we are trying to find in our model, however the results our analysis in the non-linear case perhaps provide cause for consideration into the choice of our observation points, and whether the error at one is independent of of the error measured at another. The information from multiple observation points is clearly redundant in the linear equation case.

Possible further considerations, such as additional parameters (like viscosity, Coriolis effect etc.), as well as different control variables may alter the efficacy of our data assimilation scheme.

8.2 Future Considerations

To conclude this study, we provide a brief overview of considerations for future research into Data Assimilation for tsunami modelling, that were not accounted for in this relatively simpler model.

8.2.1 Model Parameters & Grid Refinement

The logical next step is to introduce additional degrees of complexity to our model, such as including the Coriolis and viscosity parameters, and see how variation in these parameters affect our overall model. For example, [10] says that the Coriolis effect increases with the source width, whereas the importance of frequency dispersion decreases as source width increases. Accounting for such relations can help refine an existing model and subsequently decrease the probability of error in approximation, as can refining our grid to a more realistic global model that is dynamically adaptive.

8.2.2 Inverse Problem with Bathymetry Optimisation

Additionally, we consider different variations within our data assimilation method instead of initial conditions, we optimise our estimates of ocean bathymetry and consider how this could be an improvement over forecasting models with existing bathymetry data that is incomplete.

We now assume that we no longer have full information on the bathymetry, and so we wish to derive an optimal estimate of the bottom topography in our domain. If the height of our water column satisfies $h(x,t) = H + \eta(x,t) - \beta(x)$ where $\beta(x)$ is the bottom

topography, and we normalise by the average height H such that $c = \sqrt{gH} = 1$, then our linearised shallow water equations are

$$\begin{aligned}
\frac{\partial \eta}{\partial t} + \frac{\partial u}{\partial x} &= 0, \\
\frac{\partial u}{\partial t} + \frac{\partial \eta}{\partial x} &= \frac{\partial \beta}{\partial x}, \\
\eta(x,0) &= \eta_0(x), \ -1 \leqslant x \leqslant 1 \\
u(x,0) &= 0 \ .
\end{aligned}$$

As before, we minimise the following cost function

$$J(\beta) = \frac{1}{2}\int_0^T \big(m(t) - \eta(x_0, t; \beta)\big)^2 \ dt.$$

We formulate an expression for the gradient in terms of the Gateaux derivative, and take the Taylor series of the first term in t, to order $\mathcal{O}(\varepsilon^2)$, where β' and η' represent the linearised forms of the height and bathymetry functions;

$$\begin{aligned}
&= \frac{\int_0^T \frac{1}{2}\big(m(t) - \big(\eta(\beta) + \varepsilon\eta'(\beta + \eta\beta')\big)\big)^2 - \frac{1}{2}\big(m(t) - \eta(\beta)\big)^2 \ dt}{\varepsilon} \\
&= \frac{\int_0^T \frac{1}{2}\big(m(t) - \eta(\beta)\big)^2 - \big(m(t) - \eta(\beta)\big)\varepsilon\eta'(\beta) - \frac{1}{2}\big(m(t) - \eta(\beta)\big)^2 \ dt}{\varepsilon} \\
&= -\int_0^T \big(m(t) - \eta(\beta)\big)\eta'(\beta) \ dt.
\end{aligned}$$

it should be noted that this assumes a perturbation in bathymetry $\beta \to \beta + \varepsilon\beta'$ leads to a perturbation in height $\eta \to \eta + \varepsilon\eta'$. This is true for the linearised eqution satisfied by

(u', η'). Our goal is to find a Riesz representation for the gradient $J'(\beta, \beta')$, allowing us to rewrite it as $\int_0^L \nabla_x J\beta' \, dx$. Once again we use the duality relation to stipulate

$$\int_0^T \int_0^L u^* \Big(\frac{\partial u'}{\partial t} + \frac{\partial \eta'}{\partial x} - \frac{\partial \beta'}{\partial x} \Big) + \eta^* \Big(\frac{\partial \eta'}{\partial t} + \frac{\partial u'}{\partial x} \Big) \, dxdt = 0 \ .$$

We integrate by parts, and use the fact that the boundary terms in space go to 0 to derive

$$\begin{aligned}
&\int_0^T \int_0^L u^* \Big(\frac{\partial u'}{\partial t} + \frac{\partial \eta'}{\partial x} - \frac{\partial \beta'}{\partial x} \Big) + \eta^* \Big(\frac{\partial \eta'}{\partial t} + \frac{\partial u'}{\partial x} \Big) \, dxdt \\
&= - \int_0^T \int_0^L u' \Big(\frac{\partial u^*}{\partial t} + \frac{\partial \eta^*}{\partial x} \Big) + \eta' \Big(\frac{\partial \eta^*}{\partial t} + \frac{\partial u^*}{\partial x} \Big) \, dxdt + \int_0^L u' u^* \mid_0^T \ dx + \eta' \eta^* \mid_0^T + \int_0^T \int_0^L \beta' \frac{\partial u^*}{\partial x} \, dxdt \\
&= - \int_0^T \int_0^L \eta' (m(t) - \eta(x_0, t; \beta)) \, dxdt + \int_0^T \int_0^L \beta' \frac{\partial u^*}{\partial x} \, dxdt \\
&= 0 \ .
\end{aligned}$$

Our adjoint system is

$$\begin{aligned}
\frac{\partial \eta^*}{\partial t} + \frac{\partial u^*}{\partial x} &= (m(t) - \eta(x, t; \beta)) \delta(x - x_0) \ , \\
\frac{\partial u^*}{\partial t} + \frac{\partial \eta^*}{\partial x} &= 0 \ , \\
\eta^*(x, T) = \ u^*(x, T) &= 0 \ .
\end{aligned}$$

And so we have

$$\int_0^T \int_0^L \eta'(m(t) - \eta(x_0, t; \beta))\ dxdt = \int_0^T \int_0^L \beta' \frac{\partial u^*}{\partial x}\ dxdt$$
$$\Rightarrow J'(\beta, \beta') = \int_0^T \int_0^L \beta' \frac{\partial u^*}{\partial x}\ dxdt = \int_0^L \left(\int_0^T \frac{\partial u^*}{\partial x}\ dt \right) \beta'(x)\ dx = \int_0^L (\nabla_x J)\beta'(x)\ dx$$
$$\Rightarrow \nabla_x J = \int_0^T \frac{\partial u^*}{\partial x}\ dt.$$

With this expression for $\nabla_x J$, we proceed with the following algorithm:

- Pick an initial estimate for $\beta^{(0)}(x)$.
- Solve the forward Initial Value Problem for (u, η) from 0 to T.
- Solve Adjoint Problem for (u^*, η^*) from T to 0 to find $\eta^*(x, 0)$.
- Approximate $\int_0^T \frac{\partial u^*}{\partial x}\ dt$ at every point in our spatial domain.
- Define $\nabla^{L^2} J = \int_0^T \frac{\partial u^*}{\partial x}\ dt$.
- Smooth $\nabla^{L^2} J \to \nabla^{H_p} J$.
- Compute the optimal time step τ_n through a line minimisation algorithm.
- Use a steepest descent algorithmn to compute $\beta^{(n+1)}(x) = \beta^{(n)}(x) - \tau_n \nabla^{H_p} J(\beta^{(n)}(x))$.
- Repeat until $\| \nabla^{H_p} J \| < \varepsilon$ for some small ε ($\| \int_0^T \frac{\partial u^*}{\partial x}\ dt \| \approx 0$).

As we can see, there are parallels in the numerical computation of the data assimilation algorithm with bathymetry as the control variable, to the one outlined in chapter five for the initial condition. To extend this scheme to the bathymetry case would be the logical next step and was only excluded due to time limitations.

Bibliography

[1] D.J. Acheson. *Elementary Fluid Dynamics*. Clarendon Press, Oxford, 1990.

[2] R. Banner. Elementary 4d-var darc technical report. *Data Assimilation Research Centre*, (2):101–109, 2007.

[3] A.J. Chorin and J. E. Marsden. *A Mathematical Introduction to Fluid Mechanics*. Springer Verlag, 1993.

[4] C. Dawson and C. Mirabato. *Shallow Water Equations*. Computational Engineering and Sciences, University of Texas at Austin., 2008.

[5] L. Dongfang et al. Comparison between boussinesq and shallow-water models in predicting solitary wave runup on plane beaches. *Coastal Engineering Journal*.

[6] L. C. Evans. *An Introduction to Mathematical Optimal Control Theory Version 0.2*. University of Maryland, 1983.

[7] A. K. Griffith and N. K. Nichols. Data assimilation using optimal control theory. *Numerical Analysis Report*, 10(2):101–109, 1994.

[8] N. Kevlahan. *Private Communication*. University of McMaster, 2016.

[9] J.T. Kirby. *Boussinesq models and applications to nearshore wave propagation, surf zone processes and wave-induced currents*, volume 67. Elsevier Oceanography Series, 2003.

[10] J.T. Kirby et al. *Dispersive tsunami waves in the ocean: Model equations and sensitivity to dispersion and Coriolis effects*, volume 62. 2013.

[11] D. E Kirk. *Optimal Control Theory, An Introduction.* Dover Publications Inc., 2004.

[12] P. Kundu. *Fluid Mechanics.* Elsevier Academic Press, 2004.

[13] A Majda. *Introduction to PDEs and Waves for the Atmosphere and Ocean.* American Mathematical Society, 2003.

[14] K. Nakamura et al. Sequential data assimilation: Information fusion of a numerical simulation and large scale observation data. *Journal of Universal Computing*, 12(6):608–626, 2006.

[15] I. Simpson. *The Shallow Water Model.* 2010. PHY2504S - Advanced Atmospheric Dynamics.

[16] R. Spiteri and Ruuth S. A new class of optimal high-order strong-stability-preserving time discretization methods. *SIAM Journal on Numerical Analysis*, 40(2):469–491, 2006.

[17] L. Thompson. *Ocean Circulation and Climate: Kalvin Waves.* University of Washington., 2004.

[18] G. K. Vallis. *Atmospheric and Oceanic Fluid Dynamics.* Cambridge University Press, 2006.

[19] A Von der Heydt. *Physics of the Climate System: Rossby and Kalvin Waves.* Institute for Marine and Atmospheric research, 2014.

[20] M. Walkley. *A Numerical Method for Extended Boussinesq Shallow-Water Wave Equations*. PhD thesis, 1999.

[21] G. Wei and J. Kirby. Time-dependent numerical code for extended boussinesq equations. *Journal of Waterway, Port, Coastal, and Ocean Engineering*, 121(5):251–261, 1995.