# Efficient Numerical Methods for Water Wave Propagation in Unbounded Domains

by

Geri Izbicki Jennings

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Applied and Interdisciplinary Mathematics)
in The University of Michigan
2012

Doctoral Committee:

Professor Smadar Karni, Co-Chair
Professor Robert F. Beck, Co-Chair
Professor Jeffrey B. Rauch
Professor Peter S. Smereka
Associate Research Scientist Okey Nwogu

## ACKNOWLEDGEMENTS

Completing this work was most definitely a personal journey, but there are a few people without whom it would not have been possible. To my advisor Smadar Karni, I do not know how to say thank you enough. You have been a mentor to me from the start, and it was a pleasure working with you. I have learned from you to be more patient and thorough with my work, and I hope to one day be able to communicate my ideas with the same careful clarity that you do. To Jeffrey Rauch, for teaching me from the beginning of my thesis work up until the very last minute, often in ways that were subtle enough that I didn't realize until much later what I had learned. To Robert Beck and Sijue Wu, without whom I would have never discovered this project. I also gratefully acknowledge that this work was supported in part by NSF under Award DMS #0609766, and that travel to conferences would not have been possible without generous funding from Rackham and SIAM.

To my parents, who drove me to Michigan (with some regret, I think) and have been rooting for me ever since. Thank you for your support and especially for all of YOUR hard work, helping to make sure everything else in my life goes smoothly. To my son Grant, who really hates it when I work on my computer but who puts up with it anyway (most of the time). And last, but far from least, to my husband Brian; your strength, calm support, and continuous encouragement helped me to find my way. "We are all travelers in the wilderness of this world, and the best we can find in our travels is an honest friend." – Robert Louis Stevenson

# TABLE OF CONTENTS

# LIST OF FIGURES

**Figure**

# LIST OF TABLES

**Table**

# CHAPTER I

# Introduction

In models of water waves, it is often sensible to seek solutions on an unbounded domain. For example, when modeling flow around a vessel on an open sea, effects from the distant ocean bottom or shorelines are negligible compared to the surface wave-body interactions. In order to generate computer simulations of such models, the finiteness of computer memory demands that the domain must first be truncated to finite size. This is done by introducing an artificial boundary at an arbitrary distance from the region of interest, ideally not too far away in order to minimize computational costs.

To complete the description of the computational problem, boundary conditions need to be prescribed on the artificial boundary. To effectively model the unbounded domain, the boundary conditions should make the artificial boundary invisible to outgoing waves, so that waves leaving the computational domain act as if the boundary did not exist at all. This type of boundary condition is often referred to in the literature as an "absorbing" or "non-reflecting" boundary condition, and they are non-trivial to formulate. Deriving and implementing such a boundary condition for the Water Wave Equation (WWE), which describes linearized two-dimensional incompressible, irrotational, inviscid free surface flow in deep water, is the focus of this

work.

## 1.1 Artificial Boundary Conditions

What follows is a brief review of commonly used artificial boundary conditions (ABCs) for free surface flow problems. These ABCs fall into one of several categories, and in numerical computations it is common to use combinations of techniques from several categories in order to combine the strengths of the individual techniques. The categories are the so-called absorbing beach, the Sommerfeld radiation condition, and approximations to exact absorbing boundary conditions.

### 1.1.1 Sommerfeld radiation condition

In the free surface flow literature, Sommerfeld's radiation condition [46] is implemented by requiring

$$\frac{\partial \phi}{\partial t} + C\frac{\partial \phi}{\partial x} = 0 \tag{1.1}$$

where $C$ is the wave velocity. A major problem with its use is that the phase velocity $C$ is typically not known a priori, and if it is set incorrectly noise may be generated near the boundary that will corrupt the interior solution [29]. Before the paper of Orlanski [38], $C$ was estimated by setting it to the (constant) value given by the maximum numerical phase velocity, $\frac{\Delta x}{\Delta t}$ [30], setting it to $\pm\sqrt{gh}$ for shallow water problems with depth $h$ [19], or estimated for the dominant wave using the linearized dispersion relation [42]. Orlanski improved these techniques by using a time-dependent approach, in which $C$ is estimated numerically near the boundary at each timestep and equation (1.1) is imposed using the estimated value. This technique is widely used for 2D free surface flow problems, but it assumes that the wave speed may be well approximated using only the local gradients.

### 1.1.2 Absorbing Beach

The absorbing beach is a layer near the boundary in which some mechanism is provided to damp the outgoing waves before they reach the limits of the computational domain, ideally with no reflection. There are several different techniques in use to damp outgoing waves, and we describe some of these strategies. One technique is to add terms directly to the PDE. For example, we may modify the one dimensional scalar wave equation (given in system form)

$$u_t = w_x + \mu(x) u_{xx} - \nu(x) u \tag{1.2}$$

$$w_t = u_x \tag{1.3}$$

by adding a "viscous" term with coefficient $\mu$ and a "friction" term with coefficient $\nu$ [28], either of which may be zero. The coefficients $\mu$ and $\nu$ are allowed to vary over space, so that the damping effects may be ramped up over the absorbing beach. Indeed, it is preferable to slowly turn on damping to minimize reflections - [12] discusses the relationship between effectiveness of the numerical beach and smoothness of the transition from undamped region to damped region. In [28], Israeli and Orszag also selectively apply the damping to waves traveling into the computational domain from the artificial boundary by solving

$$u_{tt} = u_{xx} - \nu(x) \left( \frac{\partial}{\partial t} + \frac{\partial}{\partial x} \right) u$$

with $\nu(x) > 0$. For free surface flow problems, the damping idea from [28] is often implemented by adding dissipative terms to the free surface kinematic or dynamic boundary conditions (or both) [3, 5, 12].

Another method for damping waves in free surface flow is to add a pressure disturbance at the surface, which creates a negative work against incident waves [22].

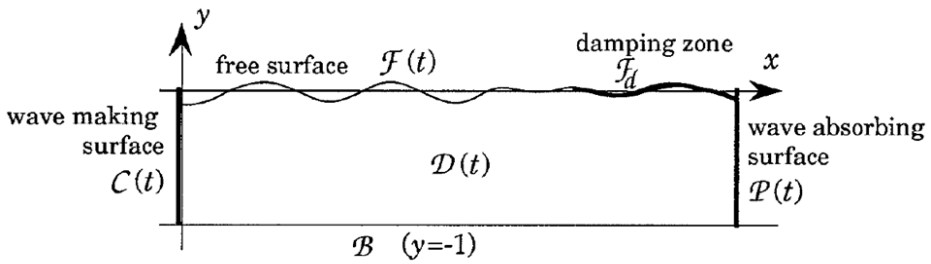



Figure 1.1: Clement's computational domain for piston-like absorbing boundaries [12]

Finally, one may implement what is called a "piston-like" absorbing boundary, which replaces the local Neumann derivative in the Sommerfield radiation condition (1.1) with [12]

$$\frac{\partial \phi}{\partial x} = \int_{\mathcal{P}(t)} \left( -\frac{\partial \phi}{\partial t} \right) dl \tag{1.4}$$

where the integral is over the line $\mathcal{P}$ that makes up the piston, as shown in figure 1.1. That is, the local Sommerfeld radiation condition is replaced by condition (1.4), which is non-local in the vertical spatial variable.

As a general rule, when using an absorbing beach, the beach length should be on the order of one wavelength [28], so that longer waves require a longer beach. Even in this case, reflections are expected; for the simple model in equations (1.2)-(1.3), setting $\nu = 0$ and $\mu =$ constant in a damping layer, the layer must be *15 wavelengths long* to get less than 1% reflection. Also, the piston-like absorbing boundary (1.4) is very effective for low frequencies, but as the frequency of outgoing waves increases, so does the magnitude of the reflection coefficient [12].

### 1.1.3 Exact Absorbing Boundary Conditions

For the wave equation, exact non-local boundary conditions were derived in the 1977 paper by Engquist and Majda [16]. The authors consider the two-dimensional

scalar wave equation

$$u_{tt} = u_{xx} + u_{yy}$$

in the half plane $x \leq 0$. This equation has solutions of the form $u = e^{i(\xi x+\eta y+\omega t)}$. Without loss of generality let $\omega > 0$ only. For solutions of this type, the dispersion relation is given by

$$\omega^2 = \xi^2 + \eta^2$$

The surfaces of constant phase are the lines $\omega t + \xi x + \eta y =$ constant, and therefore the direction of propagation of the wave is given by $(-\xi, -\eta)$. The goal is to find a boundary condition on $x = 0$ that eliminates incoming waves, i.e. waves with direction of propagation in the negative $x$ direction, or plane waves with $\xi > 0$. If we assume that $\omega^2 - \eta^2 > 0$, then we can derive the one-way dispersion relations:

$$\xi = \sqrt{\omega^2 - \eta^2} \tag{1.5}$$

which is satisfied by incoming waves, and

$$\xi = -\sqrt{\omega^2 - \eta^2} \tag{1.6}$$

which is satisfied by outgoing waves.

Taking the Fourier transform of the wave equation with respect to $y$ and $t$, we find

$$\frac{d^2\hat{u}}{dx^2} = (\eta^2 - \omega^2)\hat{u} = -\xi^2\hat{u}$$

Two linearly independent solutions of this ODE are $\hat{u}_1 = C_1(y,t)e^{-i|\xi|x}$ and $\hat{u}_2 = C_2(y,t)e^{i|\xi|x}$. Since, from equations (1.5)-(1.6), we know that $|\xi| = \sqrt{\omega^2 - \eta^2}$, once we take the inverse Fourier transforms we see that the $\hat{u}_2$ solution corresponds to incoming waves and is the solution we wish to annihilate at $x = 0$. In other words, we

would like for the vector $(\hat{u}(x), \frac{d\hat{u}}{dx})^T$ to be parallel at $x = 0$ to the vector $(\hat{u}_1(x), \frac{d\hat{u}_1}{dx})^T$, or

$$\left(\frac{d\hat{u}}{dx} + i\sqrt{\omega^2 - \eta^2}\hat{u}\right)|_{x=0} = 0 \tag{1.7}$$

This gives a local condition in the transformed space, but as we will see, a non-local condition in physical space. If we take the PDE

$$\frac{d\hat{u}}{dx} + i\sqrt{\omega^2 - \eta^2}\hat{u} = 0$$

and transform it back into physical space, we get a formula for a more general $u$ which is a superposition of left-going plane waves

$$u(x, y, t) = \int\int e^{i(\sqrt{\omega^2-\eta^2}x+\eta y+\omega t)}\rho(\eta, \omega)\hat{u}(0, \eta, \omega)d\eta d\omega. \tag{1.8}$$

Here $\hat{u}(0, \eta, \omega)$ is a special amplitude function with support for large $(\eta, \omega)$ in the cone $\omega^2 > \eta^2$ and $\rho(\eta, \omega)$ is a smooth function that is homogeneous of degree zero for $|\eta| + |\omega|$ large with support in $\omega^2 > \eta^2$ for $(\eta, \omega)$ large and identically equal to one in a neighborhood of the support of $\hat{u}(0, \eta, \omega)$. By superposition of the conditions (1.7), we derive the boundary condition which exactly annihilates wave packets of the form (1.8) at the boundary $x = 0$:

$$\left(\frac{du}{dx} + \int\int e^{i(\eta y+\omega t)}i\sqrt{\omega^2 - \eta^2}\rho\hat{u}(0, \eta, \omega)d\eta d\omega\right)|_{x=0} = 0 \tag{1.9}$$

This is an exact absorbing boundary condition, and it is nonlocal in space and time.

Writing this in the symbolic notation of pseudodifferential operators, we have

$$\left(\frac{d}{dx} + \rho\left(\frac{\partial}{\partial y}, \frac{\partial}{\partial t}\right)\sqrt{\frac{\partial^2}{\partial t^2} - \frac{\partial^2}{\partial y^2}}\right)u|_{x=0} = 0$$

For the wave equation, using the non-local operator $\sqrt{\frac{\partial^2}{\partial t^2} - \frac{\partial^2}{\partial y^2}}$ in physical space to advance the numerical solution one timestep at one point in the domain would

require information about all previous times over the entire domain. In order to reduce computational cost, local approximations to boundary condition (1.9) were derived. Engquist and Majda set $\rho \equiv 1$ and using Padé and Taylor approximations of $\sqrt{1-\eta^2/\omega^2}$ to produce a hierarchy of approximate (local) boundary conditions. This technique is exact for waves impinging on the boundary at normal incidence, but waves reaching the boundary away from normal incidence generate reflected waves with reflection coefficient magnitude given by $\left|\frac{\cos\theta-1}{\cos\theta+1}\right|$, where $\theta$ is the angle off normal incidence. The first two local conditions are:

$$\left(\frac{\partial}{\partial x} - \frac{\partial}{\partial t}\right) u = 0$$

$$\left(\frac{\partial^2}{\partial x \partial t} - \frac{\partial^2}{\partial t^2} + \frac{1}{2}\frac{\partial^2}{\partial y^2}\right) u = 0$$

Wagatha [48] proposed instead using a least-squares approximation to $\sqrt{1-\eta^2/\omega^2}$, and Halpern and Trefethen [24, 47] explored using several other methods, including Chebyshev, Chebyshev-Padé, Newman, $L^2$, and $L^\infty$ approximations. In the free surface flow literature, the boundary conditions derived by Engquist and Majda, Wagatha, and Higdon for hyperbolic flows are applied directly to the equations for free surface flow (see, for example, [44]). In addition, it is accepted (with the notable exception of the 1D wave equation case) that only approximate absorbing boundary conditions can be used in numerical computations.

#### 1.1.4 Other Techniques

Other techniques beyond those already mentioned include matching the interior (often, nonlinear) solution to a simpler (linear) solution in a boundary region (Romate [44] has a good review of this technique), mapping the infinite fluid domain to a finite domain [23], and the infinite element technique, in which beyond some

central region the grid elements are progressively stretched until they encompass the entire infinite domain (see, for example, [41] for an application and brief review).

## 1.2 Overview

The focus of this work is the derivation and numerical implementation of absorbing boundary conditions for the water wave equation, a one-dimensional nonlocal partial differential equation which describes linearized two-dimensional incompressible, irrotational, inviscid free surface flow in deep water. This thesis is organized as follows. In Chapter II, we give the details of the derivation of the WWE. We follow the paper of Wu [49], who derives a quasilinear system of equations which when linearized about the zero solution give the WWE. In Chapter III, we derive a one-way version of the water wave equation which supports the propagation of water waves essentially only in the outgoing direction. The one-way equation is a fractional partial differential equation involving a nonlocal operator corresponding to half a derivative. The fractional derivative is implemented as a derivative of a convolution with a singular kernel with locally integrable singularity. Properties of the OWWWE are given. Numerical methods for the OWWWE are discussed in Chapter IV, and a family of efficient numerical methods are derived.

We apply the OWWWE in a layer near the boundary, matching it to the full solution to the water wave equation (WWE) at the interface between the interior domain and the absorbing layer. Damping may or may not be added within the absorbing layer to further dissipate the waves before they reach the edge of the computational domain. In chapter V we describe our absorbing boundary condition in more detail and present numerical results.

# CHAPTER II

# Preliminaries

## 2.1 Notation

We begin with some useful notation. We write $H^s(\mathbb{R})$, $-\infty < s < \infty$, to represent the Sobolev space consisting of $u \in \mathfrak{S}'(\mathbb{R})$ such that $(1+|\xi|)^s \hat{u}(\xi) \in L^2(\mathbb{R})$, with the Sobolev norm given by

$$\|u\|_s = \left( \int (1+|\xi|)^{2s} |\hat{u}(\xi)|^2 d\xi \right)^{1/2}.$$

We use $[A, B] = AB - BA$ as an abbreviation for the commutator.

We let $H : L^2(\mathbb{R}) \to L^2(\mathbb{R})$ be the Hilbert transform

$$Hf(x) = \frac{1}{\pi} \text{P.V.} \int_{-\infty}^{\infty} \frac{f(y)}{x-y} dy$$

with the convention that $H1 = 0$. It will also be helpful to note that $H^2 = -I$ for functions in $H^s$. We use that the Fourier transform of the Hilbert transform is given by

$$\widehat{Hf}(\xi) = -i \operatorname{sgn} \xi \, \widehat{f}(\xi) \tag{2.1}$$

We occasionally regard the two dimensional space as the complex space $\mathbb{C}$ and write the pair $(x, y)$ as $x + iy$. We denote the real and imaginary parts of a complex-valued

function $f$ as $\Re f$ and $\Im f$, respectively. If we write $f \circ g$ we mean composition of $f$ and $g$ in terms of the spatial variables (for fixed time), so that $f \circ g(x,t) = f(g(x,t),t)$. According to this definition, given $f^{-1}$, the inverse of $f$, we have for any fixed time $t$ that $f \circ f^{-1}(x,t) = f(f^{-1}(x,t),t) = x$ for all $x$.

Finally, we use the following definition of the Fourier transform:

$$\widehat{f}(\xi) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i x \xi} dx, \qquad f(x) = \int_{-\infty}^{\infty} f(\xi) e^{2\pi i x \xi} d\xi$$

## 2.2 Derivation of the Water Wave Equation (WWE)

In what follows, we give the details of the derivation of the WWE. This derivation was introduced by Wu in [49], and is presented here with minor modifications for completeness. We follow the setup of [49]; we consider the motion of the interface separating an incompressible, irrotational, inviscid fluid in deep water under the influence of gravity from a region of zero density in two-dimensional space. We neglect surface tension and assume that the density of the fluid is one and the gravitational force is given by $(0,-1)$. At time $t \geq 0$ we denote the free interface by $\Gamma(t)$ and we let $\Omega(t)$ denote the region occupied by the fluid (below $\Gamma(t)$) (note that as this is a moving interface, our domain inherently depends on time). Thus the motion of the fluid is given by the incompressible, irrotational, inviscid Euler equations:

$$\vec{v}_t + (\vec{v} \cdot \nabla)\vec{v} = -(0,1) - \nabla p, \qquad x \in \Omega(t), t \geq 0 \tag{2.2}$$

$$\nabla \cdot \vec{v} = 0 \qquad x \in \Omega(t), t \geq 0 \tag{2.3}$$

$$\nabla \times \vec{v} = 0 \qquad x \in \Omega(t), t \geq 0 \tag{2.4}$$

where $\vec{v} = (v_1, v_2)$ is the fluid velocity and $p$ is the fluid pressure. Additionally, we

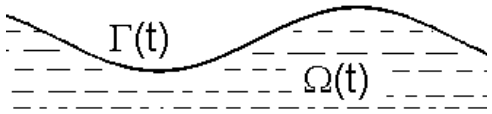



Figure 2.1: The domain of the fluid in equations (2.2) - (2.4)

have the kinematic and dynamic boundary conditions

$$p = \text{constant} \qquad x \in \Omega(t), t \geq 0 \tag{2.5}$$

$$(1, v_1, v_2) \text{ is tangent to the free surface } \Gamma(t) \qquad \forall\ t \geq 0 \tag{2.6}$$

The velocities in equation (2.6) are evaluated at $(\Gamma(t), t)$. Equation (2.5) follows because we have neglected surface tension, and equation (2.6) states that the free surface is a streamline. We are seeking a solution of equations (2.2) - (2.6) such that $\Gamma(t)$ approaches the horizontal axis at infinity for every $t \geq 0$ and $|\vec{v}(x, y, t)| \to 0$ as $|(x, y)| \to \infty$. In [49], Wu proves existence and uniqueness of solutions of this system of equations, locally in time, for any initial interface which is nonself-intersecting. In the process, Wu derives the aforementioned system of quasilinear equations that are equivalent to (2.2) - (2.4) and that we seek to use in our own work. In the next several sections we present the details of the derivation of the system.

### 2.2.1 Derivation of Quasilinear System

**Rewriting System In Terms of Lagrangian Variable $\alpha$**

First we rewrite equations (2.2) - (2.6) in terms of a Lagrangian variable $\alpha$, so that the interface $\Gamma(t)$ is described by a curve $z(\alpha, t) = x(\alpha, t) + iy(\alpha, t)$, $-\infty < \alpha < \infty$. Given this curve, we can write the velocity along the interface as

$$z_t(\alpha, t) = \vec{v}(z(\alpha, t), t) = v_1(z(\alpha, t), t) + iv_2(z(\alpha, t), t). \tag{2.7}$$

We are only interested in solving for the fluid velocity along the interface, since once we have solved for $z_t$, the fluid velocity in all of $\Omega(t)$ can be recovered by solving

Laplace's equation:

$$\Delta v = 0 \qquad \text{in } \Omega(t)$$

$$v = z_t \qquad \text{in } \Gamma(t)$$

which $v$ satisfies because the system is irrotational and incompressible. Taking another derivative with respect to $t$ of equation (2.7), we find that on the interface,

$$\vec{z}_{tt} = \vec{v}_t + (\vec{v} \cdot \nabla)\vec{v} = -(0,1) - \nabla p. \tag{2.8}$$

In writing the equation (2.8), we have implicitly used vector notation. At this point we find it more convenient to rewrite it using complex variables notation. First we need an important result that we use several times in our derivation, and the proof follows a result in [40].

**Lemma II.1.** *Let $f(z)$ be analytic in the region $\Im z = y \leq 0$ such that $f(x) \in L^2 \quad \forall x \in \mathbb{R}$. Then*

$$f(x) = -i(Hf)(x)$$

*that is,*

$$v(x) = (Hu)(x) \qquad a.e. \qquad x \in \mathbb{R}$$

$$u(x) = -(Hv)(x) \qquad a.e. \qquad x \in \mathbb{R}$$

*where $f(x) = u(x) + iv(x)$.*

*Proof.* We first prove the result for $f \in C^1$. Consider the Cauchy-type integral

$$\int_L \frac{f(t)}{t-z} dt \tag{2.9}$$

where $L$ is the real line $-\infty < t < \infty$. Fix $t_0 \in \mathbb{R}$. We would like to take the limit first as $z \to t_0$ in equation (2.9). To do so, we write

$$\int_L \frac{f(t)}{t-z} dt = \int_{L-L_\epsilon} \frac{f(t)}{t-z} dt + \int_{L_\epsilon} \frac{f(t) - f(z_L)}{t-z} dt + \int_{L_\epsilon} \frac{f(z_L)}{t-z} dt$$

where $z_L$ is the real part of $z$ (i.e. the point on the line $L$ that is closest to $z$) and $L_\epsilon = L\backslash(t_0 - \epsilon, t_0 + \epsilon)$. The first and third integrals converge as $z \to t_0$ and $\epsilon \to 0$ to

$$\text{P.V.} \int \frac{f(t)}{t - t_0} dt - \pi i f(t_0).$$

For the second integral, we have

$$\lim_{\epsilon \to 0} \lim_{z \to t_0} \int_{L_\epsilon} \frac{f(t) - f(z_L)}{t - z} dt = 0,$$

using differentiability of $f$ along the real axis. Thus we have

$$\lim_{z \to t_0} \int_L \frac{f(t)}{t - z} dt = \text{P.V.} \int \frac{f(t)}{t - t_0} dt - \pi i f(t_0)$$

Now, $f$ analytic in $P_- = \{z \in \mathbb{C} \mid \Im z \leq 0\}$ implies by the Cauchy integral formula that

$$-2\pi i f(z) = \int_L \frac{f(t)}{t - z} dt,$$

and this gives the desired result for functions $f \in C^1$. For general $f \in L^2$, the result follows from the density of $C_0^\infty$ functions in $L^2$ and the property of the Hilbert transform that $\|Hf\|_{L^2} \leq \|f\|_{L^2}$. □

We know a system is incompressible and irrotational if and only if $\bar{v}$ is an analytic function, and we have assumed $|v| \to 0$ as $|(x, y)| \to \infty$. By Lemma 1, we then have that $\bar{z}_t = -iH\bar{z}_t$. Also, by assumption, $p = 0$ on $\Gamma(t)$ (since we are neglecting surface tension), and so we expect that $\nabla p$ will be normal to the interface. Since $z_\alpha$ is tangent to the interface, $iz_\alpha$ will be normal to it and thus there exists some $a = a(\alpha, t)$ such that $-\nabla p = iaz_\alpha$. In vector notation we have written Euler's equation as $\vec{z}_{tt} = -(0, 1) - \nabla p$, and if we rewrite it using complex notation with our new representation of $p$ we find the original system (equations (2.2) - (2.6)) is

equivalent to

$$z_{tt} + i = iaz_\alpha \qquad \Omega(t) \tag{2.10}$$

$$\bar{z}_t = -iH\bar{z}_t \qquad \Gamma(t) \tag{2.11}$$

**Riemann Mapping to Lower Half Plane**

Next, we map the fluid domain into the lower half plane (which we write as $P_-$). For fixed time $t$, the interface is given by $\Gamma(t) = z(\alpha, t) = x(\alpha, t) + iy(\alpha, t)$, and $\Gamma(t)$ approaches the real axis as $\alpha \to \pm\infty$. If we assume that $\Gamma(t)$ is nonself-intersecting, then $\Gamma(t)$ divides the complex plane into two simply connected regions: an upper region of zero density and a lower region ($\Omega(t)$) in which the fluid lies. Let $P_-$ denote the lower half of the complex plane $\{x + iy \mid y \leq 0\}$. By the Riemann mapping theorem, there is a unique Riemann mapping $\Phi(\cdot, t) : \Omega(t) \to P_-$ that extends to an orientation preserving homeomorphism $\Phi : \Gamma(t) \to \{y = 0\}$, where $\lim_{z\to\infty}\Phi_z(z, t) = 1$ ($\Phi_z = \frac{d\Phi}{dz}$).

Let

$$h(\alpha, t) = \Phi(z(\alpha, t), t).$$

Then $h(\alpha, t) : \mathbb{R} \to \mathbb{R}$, $h(\alpha, t)$ is increasing in $\alpha$, and $h_\alpha > 0$ for all $\alpha$. Note that $h$ is real valued for all $\alpha$ and for each fixed $t$. We also know that $\Phi$ is analytic for each fixed $t$. We write $z(\alpha, t) = \Phi^{-1}(h(\alpha, t), t)$ and $\Phi^{-1}(\alpha, t) = z(h^{-1}(\alpha, t), t)$.

**Notation and Properties of Mapped Functions in $P_-$**

At this point it is useful to develop some more notation. We write $Z(\beta, t) = z \circ h^{-1}(\beta, t) = z(h^{-1}(\beta, t), t)$, $Z_t = z_t \circ h^{-1}$, etc.; $Z_{,\beta} = \partial_\beta Z(\beta, t)$, $Z_{t,\beta} = \partial_\beta Z_t(\beta, t)$, etc.; and we use similar notation for the real and imaginary parts of $z$, i.e. we let $X_t = x_t \circ h^{-1}$, $Y_t = y_t \circ h^{-1}$, etc. Notice that $Z(\beta, t) = \Phi^{-1}(\beta, t)$.

Recall that the complex conjugate of the velocity is an analytic function. Let us consider $\bar{v} \circ \Phi^{-1}(\alpha, t)$. Since $\Phi$ is a conformal mapping, $\Phi$ and $\Phi^{-1}$ are analytic functions and we know $\Phi^{-1} : P_- \to \Omega(t)$. Therefore, $\bar{v} \circ \Phi^{-1}(\alpha, t)$ must be analytic in $P_-$, and we have for all $\beta \in \mathbb{R}$,

$$\bar{v}(\Phi^{-1}(\beta, t), t) = \bar{v}(z(h^{-1}(\beta, t), t), t) = \bar{z}_t(h^{-1}(\beta, t), t) = \bar{Z}_t(\beta, t) \tag{2.12}$$

where equation (2.12) gives the value of the holomorphic function $\bar{v} \circ \Phi^{-1}$ along the boundary of $P_-$.

Based on lemma II.1, since $\bar{z}_t \circ h^{-1}$ is the boundary value of a holomorphic function we have that (writing $U_h f(x, t) = f(h(x, t), t)$),

$$\bar{z}_t \circ h^{-1} = -iHU_h^{-1}\bar{z}_t$$

$$\Rightarrow \bar{z}_t = -iU_h H U_h^{-1}\bar{z}_t$$

That is, we have that $(I + iU_h H U_h^{-1})\bar{z}_t = 0$. Also, since $\bar{Z}_t = \bar{z}_t \circ h^{-1}$, we have that

$$\bar{Z}_t = -iH\bar{Z}_t$$

$$\Rightarrow X_t - iY_t = -iHX_t - HY_t$$

so that

$$X_t = -HY_t \tag{2.13}$$

$$Y_t = HX_t. \tag{2.14}$$

From now on we write $K = U_h H U_h^{-1}$. By lemma II.1, we know that $K$ has the property that for any analytic function $f(z) = u(z) + iv(z)$ that is $o(1)$ as $|z| \to \infty$ in the lower half plane, $u - iKu$ is the value of the function $f(z)$ on the boundary. We also know that since $H^2 = -1$, we have $K^2 = -1$ also.

Since $\Phi^{-1}$ is an analytic function, $Z(\beta,t) = \Phi^{-1}(\beta,t)$ and $Z_{,\beta}(\beta,t) = \Phi_{\beta}^{-1}(\beta,t)$ are both boundary values of analytic functions. Since $Z(\beta,t) = z \circ h^{-1}$ and we assume that the interface approaches the horizontal axis as $\beta \to \pm\infty$, $Z(\beta,t) \to \beta$ as $|\beta|$ gets large. Also, since we assume that $\Phi_z \to 1$ as $z \to \infty$, and $(\Phi^{-1})_z = \frac{1}{\Phi_z \circ \Phi^{-1}}$, we must have that

$$\lim_{\beta \to \pm\infty} Z_{,\beta}(\beta,t) \to 1. \tag{2.15}$$

Based on equation (2.15), if we write $Z_{,\beta} = [(X_{,\beta} - 1) + iY_{,\beta}] + 1$ we expect the expression inside the brackets to go to 0 at infinity and to be holomorphic. Thus by lemma II.1 we can write

$$X_{,\beta} - 1 = HY_{,\beta}, \qquad Y_{,\beta} = -H(X_{,\beta} - 1) = -HX_{,\beta}.$$

From here on, we write when convenient $u = X_t$, $w = X_{tt}$, and $b = h_t \circ h^{-1}$. Note that due to the definition of $X_{tt}$ as $x_{tt} \circ h^{-1}$, we do not have that $w = u_t$. There is, however, a relationship between $u$ and $w$ that we explore later.

We have already seen in equation (2.14) that $Y_t = HX_t$, and we would like to show that $Y_{tt}$ can also be recovered from $X_t$ and $X_{tt}$ so that we can concern ourselves with deriving a PDE in $X$ only. Before we can show this, however, we need another lemma:

**Lemma II.2.** $[\partial_t, K]f(\alpha,t) = [h_t, K]U_h \partial_\alpha U_h^{-1} f(\alpha,t)$

*Proof.*

$$\begin{aligned}
[\partial_t, K]f(\alpha,t) &= \partial_t U_h H U_h^{-1} f - K f_t \\
&= h_t U_h \partial_\alpha H U_h^{-1} f + U_h \partial_t H U_h^{-1} f - K f_t \\
&= h_t U_h H \partial_\alpha U_h^{-1} f + U_h H((U_h^{-1}\partial_\alpha f)(\partial_t h^{-1}) + U_h^{-1}\partial_t f) - K f_t \\
&= h_t U_h H \partial_\alpha U_h^{-1} f + U_h H U_h^{-1}(\partial_\alpha f)\left(\frac{h_t}{h_\alpha}\right) \\
&= h_t U_h H \partial_\alpha U_h^{-1} f + U_h H U_h^{-1} h_t U_h \partial_\alpha U_h^{-1} f \\
&= [h_t, K] U_h \partial_\alpha U_h^{-1} f(\alpha,t)
\end{aligned}$$

□

Using lemma II.2, we can examine $Y_{tt}$:

$$\begin{aligned}
Y_{tt} = U_h^{-1} y_{tt} = U_h^{-1}\partial_t y_t &= U_h^{-1}\partial_t U_h H X_t \\
&= U_h^{-1}[\partial_t, K]x_t + U_h^{-1} K x_{tt} \\
&= U_h^{-1}[h_t, K] U_h \partial_\alpha X_t + H X_{tt} \\
&= [b, H]\partial_\alpha u + Hw,
\end{aligned}$$

and we write $u_1 = [b, H]\partial_\alpha u + Hw$. Given the relationship

$$Y_{tt} = [b, H]\partial_\alpha u + Hw, \tag{2.16}$$

together with (2.14), we see that we can restrict ourselves to finding a PDE in $X$, since $Y_t$ and $Y_{tt}$ can be recovered from $X_t$ and $X_{tt}$.

**Manipulating the PDE**

At this point, we return to equations (2.10)-(2.11) and derive the final quasilinear system. If we take another derivative with respect to $t$ of equation (2.10), we find

that

$$z_{ttt} - iaz_{t\alpha} = ia_t z_\alpha$$

If we compose this equation with $h^{-1}$, we have

$$\bar{Z}_{ttt} + i(a \circ h^{-1})\bar{Z}_{t,\alpha}(h_\alpha \circ h^{-1}) = (i\bar{a_t} z_\alpha \circ h^{-1}) \tag{2.17}$$

Taking the real part of $\bar{Z}_{ttt}$, we get $X_{ttt}$. We can simplify this and write it in terms of $u$:

$$X_{ttt} = U_h^{-1}\partial_t^2 U_h X_t \tag{2.18}$$

$$= U_h^{-1}\partial_t U_h U_h^{-1}\partial_t U_h X_t \tag{2.19}$$

$$= (\partial_t + b\partial_\alpha)^2 u \tag{2.20}$$

The progression in equations (2.18)-(2.20) follows from a straightforward application of derivative rules and is easy to check. We also note that

$$\Re(i\bar{Z}_{t,\alpha}) = Y_{t,\alpha} = \partial_\alpha H X_t = \partial_\alpha H u$$

The only remaining term on the left hand side of our PDE is $(ah_\alpha) \circ h^{-1}$. We know from equation (2.10) that $z_{tt} + i = iaz_\alpha$, and if we multiply this by $\frac{\bar{z}_\alpha}{h_\alpha}$ we find

$$\frac{\bar{z}_\alpha}{h_\alpha}(z_{tt} + i) = ia\frac{|z_\alpha|^2}{h_\alpha} = iA_1 \circ h \tag{2.21}$$

which defines a real-valued function $A_1$ since $a$ and $h$ are real valued. Composing equation (2.21) with $h^{-1}$, we find

$$iA_1 = \bar{Z}_{,\alpha}(Z_{tt} + i) \tag{2.22}$$

Returning to the expression $(ah_\alpha)\circ h^{-1}$, since $A_1 \circ h = \frac{a|z_\alpha|^2}{h_\alpha}$ and $iA_1 = \bar{Z}_{,\alpha}(Z_{tt}+i)$,

we can write

$$\begin{aligned}(ah_\alpha)\circ h^{-1} &= \left(\frac{A_1\circ h}{|\ \frac{z_\alpha}{h_\alpha}\ |^2}\right)\circ h^{-1}\\ &= \frac{A_1}{|\ Z_{,\alpha}\ |^2}\\ &= \frac{|\ Z_{tt}+i\ |^2}{A_1}\end{aligned}$$

Thus if we let $A = \frac{|Z_{tt}+i|^2}{A_1} = \frac{u^2+(u_1+1)^2}{A_1}$, when we take the real part of the PDE (2.17) we get

$$(\partial_t + b\partial_\alpha)^2 u + A|D|u = \Re(\overline{ia_t z_\alpha}\circ h^{-1}) \tag{2.23}$$

where

$$|D| = H\partial_\alpha = \partial_\alpha H. \tag{2.24}$$

At this point all that remains is to simplify the right hand side of the equation (2.23). We can write

$$\begin{aligned}\Re(\overline{ia_t z_\alpha}\circ h^{-1}) &= \Re((\frac{a_t}{a}\circ h^{-1})(\overline{iaz_\alpha}\circ h^{-1}))\\ &= \Re((\frac{a_t}{a}\circ h^{-1})\overline{Z_{tt}+i})\\ &= w(\frac{a_t}{a}\circ h^{-1})\end{aligned}$$

and thus we are left only to find a way to express $\frac{a_t}{a}\circ h^{-1}$ in terms of the variables we are considering. In doing so it is helpful to recall that since $a$ is real valued, $a_t$ will also be real valued. Recalling that $iaz_\alpha = z_{tt}+i$ and denoting

$$G = (I+iK)(\overline{ia_t z_\alpha}) = (I+iK)(\bar{z}_{ttt} + iaz_{t\alpha})$$

we can write

$$\begin{aligned}\frac{z_\alpha}{h_\alpha}(\bar{z}_{ttt}+ia\bar{z}_{t\alpha}) &= \overline{i\frac{a_t}{a}az_\alpha}\frac{z_\alpha}{h_\alpha}\\ &= \frac{a_t}{a}\overline{[(z_{tt}+i)\frac{\bar{z}_\alpha}{h_\alpha}]}\\ &= \frac{a_t}{a}\overline{[iA_1\circ h]}\\ &= -i\frac{a_t}{a}(A_1\circ h)\end{aligned}$$

Thus, since $(I+iK)(I-iK)f=0$ and $\frac{z_\alpha}{h_\alpha}$ is holomorphic,

$$\begin{aligned}(I+iK)(-i\frac{a_t}{a}(A_1\circ h)) =&\frac{1}{2}(I+iK)[\frac{z_\alpha}{h_\alpha}(I+iK)(\bar{z}_{ttt}+ia\bar{z}_{t\alpha})+\\ &\frac{z_\alpha}{h_\alpha}(I-iK)(\bar{z}_{ttt}+ia\bar{z}_{t\alpha})] \qquad (2.25)\\ =&\frac{1}{2}(I+iK)[\frac{z_\alpha}{h_\alpha}(I+iK)(\bar{z}_{ttt}+ia\bar{z}_{t\alpha})] \qquad (2.26)\end{aligned}$$

Taking the imaginary part of equation (2.26) and composing with $h^{-1}$, we find

$$-(\frac{a_t}{a}\circ h^{-1})A_1=\frac{1}{2}\Im[(I+iH)(Z_{,\alpha}U_h^{-1}G)].$$

Thus, we have succeeded in simplifying the PDE (2.17) to

$$(\partial_t+b\partial_\alpha)^2u+A|D|u=\frac{u}{A_1}(-\frac{1}{2}\Im[(I+iH)(Z_{,\alpha}U_h^{-1}G)])$$

We are nearly done. What remains is only to write $(-\frac{1}{2}\Im[(I+iH)(Z_{,\alpha}U_h^{-1}G)])$ as $f_1(u,w)$ for some function $f_1$. This would require much more tedious calculation, so we just post here the result derived in the paper [49]. That is,

$$\begin{aligned}f_1(u,w) =&4[w,H]u_\alpha+4[u,H]w_\alpha\\ &-4[u,H]\Bigg\{\partial_\alpha\left(\Re\frac{(u-iHu)(u_\alpha+iHu_\alpha)(w+i(u_1+1))}{iA_1}\right)\\ &+\Re\frac{(u_\alpha+iHu_\alpha)^2(w+i(u_1+1))}{iA_1}\Bigg\}\\ &+2[u^2-(Hu)^2,H]\partial_\alpha\left(\Re\frac{(u_\alpha+iHu_\alpha)(w+i(u_1+1))}{iA_1}\right)\end{aligned} \qquad (2.27)$$

In summary, the new (quasilinear) system of equations is

$$w_t + bw_\beta + A|D|u = \frac{w}{A_1} f_1(u, w) \tag{2.28}$$

$$u_t + bu_\beta - w = 0 \tag{2.29}$$

where

$$u_1 = Hw + [b, H]u_\beta \tag{2.30}$$

$$b = 2[u, H]\frac{w}{A_1} + 2u \tag{2.31}$$

$$A_1 = 2[u, H]u_\beta + 1 \tag{2.32}$$

$$A = \frac{w^2 + (u_1 + 1)^2}{A_1} \tag{2.33}$$

and $f_1(u, w)$ is as given in equation (2.27).

### 2.2.2 Initial Data

Without loss of generality we let $h(\alpha, 0) = \alpha$ for $\alpha \in \mathbb{R}$, which implies also that $x_\alpha(\alpha, 0) \to 1$ as $\alpha \to \pm\infty$ (see equation (2.15)). In solving the original problem in equations (2.2) - (2.6), we had initial data at the free surface

$$x_t(\alpha, 0) = u_0(\alpha)$$

Recalling that $u = X_t = x_t \circ h^{-1}$, we see that

$$u(\beta, 0) = x_t(h^{-1}(\beta, 0), 0) = x_t(\alpha, 0) = u_0(\alpha)$$

To compute the initial data $w(\beta, 0)$, we use the property of $x_\alpha$ at infinity to write

$$x_\alpha(\alpha, 0) = x^0(\alpha) + 1$$

Recalling $u = X_t$ and $w = X_{tt}$, we use equation (2.22) together with the relations $Y_{tt} = [b, H]\partial_\alpha w + Hu$ and $Y_{,\beta} = -HX_{,\beta}$ to find that $w(\beta, 0)$ satisfies

$$(x^0 + 1 + iHx^0)(w(\alpha, 0) + i([b, H]\partial_\alpha u_0 + Hw(\alpha, 0) + 1)) = i(2[u_0, H]\partial_\alpha u_0 + 1)$$

This is the equation to solve for $w(\beta, 0)$ in terms of the known functions $x^0(\alpha)$ and $u_0(\alpha)$, using the formula (2.31) for $b$ (in terms of $u$ and $w$).

### 2.2.3 Linearized Equation

In this work, we are primarily interested in small-amplitude solutions to the system (2.28)-(2.29). A small perturbation $u(x,t)$ to the zero solution of the system (2.28)-(2.29) satisfies the one-dimensional equation

$$u_{tt} + |D|u = f(u, x, t) \tag{2.34}$$

where $f$ is a function that is at least quadratic in $u$.

The linearized equation for $u$ is therefore given by

$$u_{tt} + |D|u = 0 \tag{2.35}$$

where $u = x_t \circ h^{-1}$ and $y_t \circ h^{-1}$ can be recovered using the formula $Y_t = HX_t$ (from equation (2.14)).

As in section 2.2.2, we can give linearized initial data for the linearized equation (2.35) which relates back to the original problem. We choose

$$\begin{aligned} u(\alpha, 0) &= u_0(\alpha) \\ u_t(\alpha, 0) &= \frac{(Hx^0)[2(1+x^0)+1]}{(x^0+1)^2 + (Hx^0)^2} \end{aligned}$$

Note that if $x^0 \ll 1$, $u_t(\alpha, 0) = 3Hx^0$.

**Proposition II.3.** (Energy Estimate for Linearized Equation) *A solution* $u(x,t)$ *to equation* (2.35) *with initial data*

$$u(x, 0) = u_0(x)$$

$$u_t(x, 0) = u_1(x)$$

*satisfies*

$$\|u_t(\cdot,t)\|_{H^s}^2 + \||D|^{1/2}u(\cdot,t)\|_{H^s}^2 = \|u_t(\cdot,0)\|_{H^s}^2 + \||D|^{1/2}u(\cdot,0)\|_{H^s}^2 \tag{2.36}$$

*Proof.* If we take the Fourier transform of equation (2.34) in $x$, since $\hat{H} = -i\text{sgn}\xi$ and $\hat{\partial}_x = i\xi$, the PDE becomes

$$\hat{u}_{tt} + |2\pi\xi|\hat{u} = 0.$$

Multiplying by $(1+\xi)^{2s}\hat{u}_t$ and integrating with respect to $\xi$, we find that

$$\begin{aligned} \frac{1}{2}\frac{d}{dt}\int (1+\xi)^{2s}|\hat{u}_t|^2 d\xi + \frac{1}{2}\frac{d}{dt}\int (1+\xi)^{2s}||2\pi\xi|^{1/2}\hat{u}|^2 d\xi = 0 \\ \Rightarrow \frac{1}{2}\frac{d}{dt}\|u_t(\cdot,t)\|_{H^s}^2 + \frac{1}{2}\frac{d}{dt}\||D|^{1/2}u(\cdot,t)\|_{H^s}^2 = 0 \end{aligned}$$

leading to the desired result.

□

## CHAPTER III

# One Way Water Wave Equation

We have seen that the horizontal velocity on the free surface, $u = x_t(h^{-1}(\beta, t), t)$, satisfies (for small $u$)

$$u_{tt} + |D|u = 0 \tag{3.1}$$

$$u(\beta, 0) = u_0(\beta) \tag{3.2}$$

$$u_t(\beta, 0) = u_1(\beta) \tag{3.3}$$

where $\beta$ originates from the Riemann mapping and represents some parametrization of the free surface. For convenience, we switch notation $\beta \to x$ and from now on we write $u = u(x, t)$, where $x$ is related but not identical to the $x$ coordinate.

The one dimensional scalar wave equation can be factorized

$$u_{tt} - u_{xx} = \left(\frac{\partial}{\partial t} - \frac{\partial}{\partial x}\right)\left(\frac{\partial}{\partial t} + \frac{\partial}{\partial x}\right) u = 0$$

into its left- $(-)$ and right- $(+)$ moving components. We derive such a factorization of the water wave equation (3.1),

$$u_{tt} + |D|u = \left(\frac{\partial}{\partial t} - ??\right)\left(\frac{\partial}{\partial t} + ??\right) = 0,$$

and we call the individual factors the One Way Water Wave Equations (OWWWEs). In this chapter, we derive the one way water wave equations, discuss some of their

properties, and show that the essential one-sidedness of these equations yields improved behavior near the artificial boundary of the truncated domain over the solution to the full WWE. In chapter IV, we also introduce damped versions of the one-way equations to further improve their absorption properties near the boundary.

Let us begin by getting a better understanding of the PDE (3.1). In Fourier space, the operator $|D|$ is given by

$$\widehat{|D|u}(\xi) = |2\pi\xi|\,\widehat{u}(\xi).$$

As we saw in equation (2.24), it can be related to the Hilbert transform as

$$|D| = H\partial_x = \partial_x H,$$

which can be readily seen if we recall from equation (2.1) that the Hilbert transform is represented in Fourier space as $\widehat{Hu}(\xi) = -i\,\mathrm{sgn}\,\xi\widehat{u}(\xi)$. Using the relationship between $|D|$ and the Hilbert transform and writing the Hilbert transform in integral form, we may rewrite the WWE (3.1) as

$$u_{tt}(x,t) + \frac{\partial}{\partial x}\frac{1}{\pi}\mathrm{P.V.}\int \frac{u(y,t)}{x-y}dy = 0 \tag{3.4}$$

The WWE is linear, and as one might expect from the underlying incompressibility, it is non-local in space. Its dispersion relation is given by

$$(2\pi\omega)^2 = |2\pi\xi|$$

with phase velocity

$$\left|\frac{\omega}{\xi}\right| = \frac{1}{\sqrt{|2\pi\xi|}} \tag{3.5}$$

and group velocity

$$\left|\frac{\partial\omega}{\partial\xi}\right| = \frac{1}{2\sqrt{|2\pi\xi|}}. \tag{3.6}$$

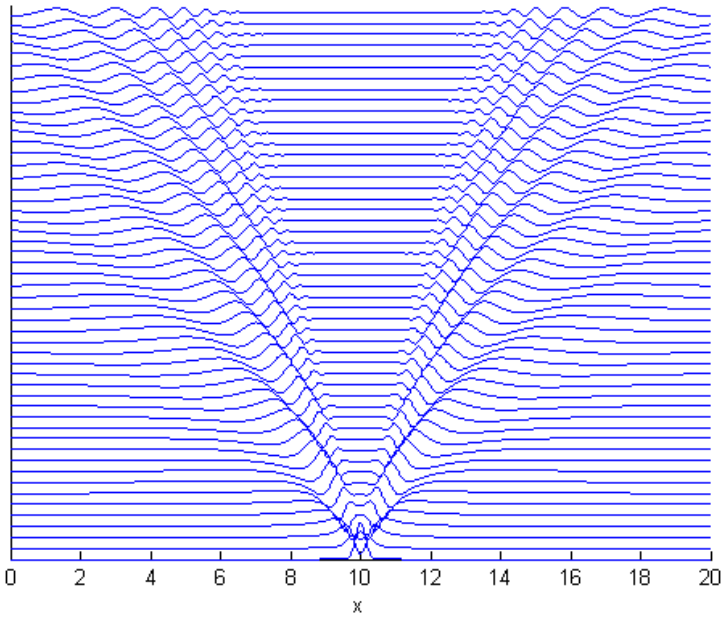



Figure 3.1: Solution of WWE with increasing time on the vertical axis, computed with initial data (3.7)-(3.8).

The phase and group velocities reflect the dispersive nature of the equation, namely that waves with different wave numbers propagate at different speeds. In particular, small wave numbers propagate infinitely fast.

Typical solutions for the WWE are shown in figure 3.1 for an initial Gaussian surface perturbation of flow at rest

$$u(x,0) = \frac{1}{2}\exp^{-\frac{x^2}{0.04}} \tag{3.7}$$

$$u_t(x,0) = 0 \tag{3.8}$$

The wave splits (symmetrically, in this case) into left and right moving waves, and disperses as it moves out. The equation is solved in physical space by approximating the Hilbert transform using the trapezoid rule and the spatial and temporal derivatives using centered differencing. The Hilbert transform is truncated by assuming that the solution $u \equiv 0$ outside of the computational domain.

We are interested in the behavior of the WWE as the wave reaches the artificial

boundary. Figure 3.2 shows the solution of the WWE with wave packet initial data

$$u(x,0) = \begin{cases} \cos^6\left(\frac{\pi}{6}x - \frac{5\pi}{3}\right)\sin\left(5\left(\frac{\pi}{6}x - \frac{5\pi}{3}\right)\right) & : 7 \le x \le 13 \\ 0 & : \text{else} \end{cases} \tag{3.9}$$

$$u_t(x,0) = 0 \tag{3.10}$$

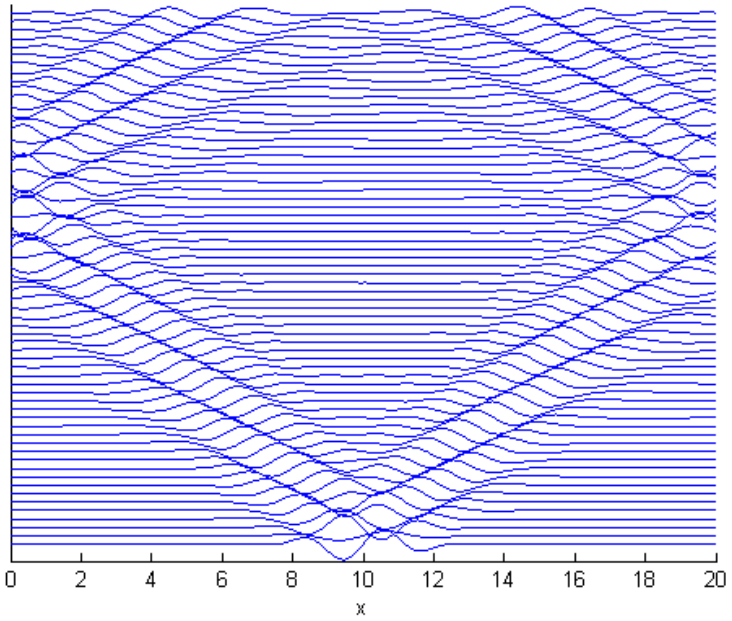



Figure 3.2: Solution of WWE with increasing time on the vertical axis, computed with initial data (3.9)-(3.10). Note the reflection of the wave as it reaches the boundary of the computational domain.

Notice that as the wave reaches the boundary, it is reflected back into the computational domain. Without the boundary, the waves would move outward, separating into components moving to the left and the right, leaving behind a quiet region which tends to zero. With the boundary present, we see that the solution of the WWE is far from zero at the center for large $t$. This is the behavior that we hope to mitigate with our boundary technique, by replacing the full WWE by the one-way equation near the computational boundary so that waves will only be allowed to propagate in

the outgoing direction.

## 3.1 Derivation of the One Way Water Wave Equation (OWWWE)

Recall the dispersion relation of the WWE (3.4)

$$(2\pi i\omega)^2 + |2\pi i\xi| = 0 \tag{3.11}$$

Expressing $\omega$ in terms of $\xi$ yields two branches

$$\omega = \pm\frac{1}{\sqrt{2\pi}}\sqrt{|\xi|} \tag{3.12}$$

The corresponding phase and group velocities are easily obtained

$$C_p = \frac{\omega}{\xi} = \pm\,\mathrm{sgn}(\xi)\frac{1}{\sqrt{|2\pi\xi|}}$$
$$C_g = \frac{\partial\omega}{\partial\xi} = \pm\frac{1}{2\sqrt{2\pi}}\,\mathrm{sgn}(\xi)|\xi|^{-1/2}$$

The branches of the dispersion relation (3.12) are illustrated in figure 3.3. Notice that neither of the branches describe propagation of waves in essentially one direction; rather, each branch contains both left and right moving waves. However, there is an

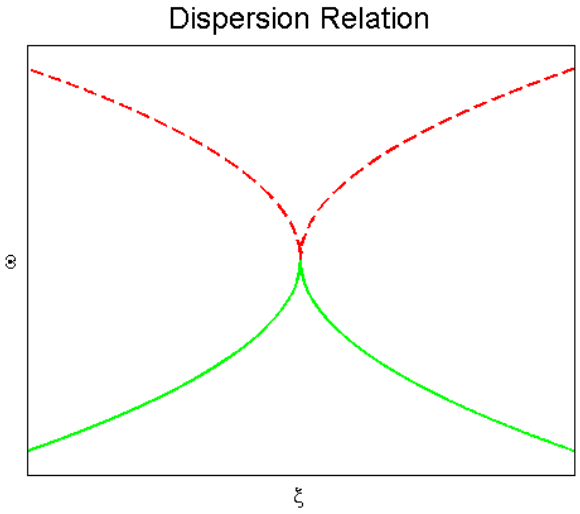



Figure 3.3: Square root of WWE dispersion relation using the standard square root

alternative expression for $\omega$ in terms of $\xi$, and we can re-write the dispersion relation (3.12) by taking a smart square root (SSR) so that the dispersion relation becomes

$$\omega = \pm \frac{1}{\sqrt{2\pi}} \operatorname{sgn}(\xi) \sqrt{|\xi|} \tag{3.13}$$

Dispersion relation (3.13) is illustrated in figure 3.4. The branches of the dispersion relation now have definite sign, and the PDE with dispersion relation (3.13) has phase/group velocities of definite sign

$$C_p = \pm \frac{1}{\sqrt{|2\pi\xi|}}$$
$$C_g = \frac{\partial \omega}{\partial \xi} = \pm \frac{1}{2\sqrt{2\pi}} |\xi|^{-1/2}$$

Thus, the PDE with dispersion relation (3.13) supports the propagation of waves

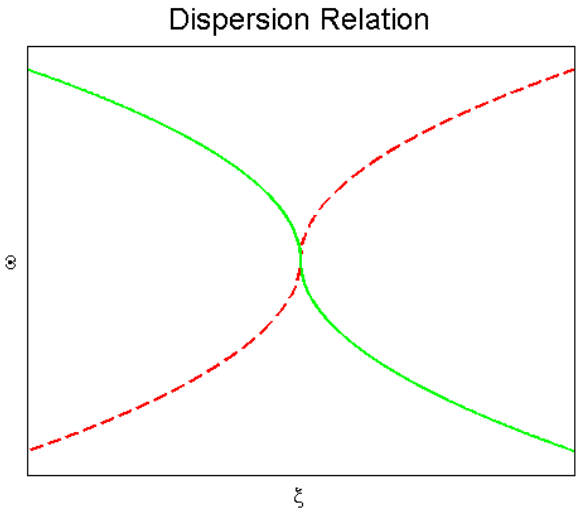



Figure 3.4: Square root of WWE dispersion relation using the "Smart Square Root"

in essentially one way. What remains is to find that PDE. To that end, we write equation (3.13) as

$$2\pi i\omega = \pm i \operatorname{sgn}(\xi) \sqrt{|2\pi\xi|} \tag{3.14}$$

We recall that the Hilbert transform is represented in Fourier space as $\widehat{Hf}(\xi) = -i \operatorname{sgn} \xi \widehat{f}(\xi)$, from which it follows that the PDE whose dispersion relation is (3.14)

is given by

$$u_t \pm H|D|^{1/2}u = 0 \tag{3.15}$$

The factorization of the WWE into one-way operators is therefore given by

$$\partial_{tt} + |D| = (\partial_t + H|D|^{1/2})(\partial_t - H|D|^{1/2}) = 0 \tag{3.16}$$

with the $(+)$ corresponds to left-moving waves and the $(-)$ corresponds to right-moving waves.

In the literature it is common to write the spatial pseudodifferential operator $|D|^{1/2}$ as the fractional Laplacian $-(-\Delta)^{1/4}$ (see for example [50], which gives details for more general fractional derivatives). The integral representation of $|D|^{1/2}$ is given by

$$|D|^{1/2}u(x) = \frac{\partial}{\partial x}\frac{1}{\sqrt{2\pi}}\int \frac{u(y)}{\sqrt{|x-y|}}\,\mathrm{sgn}(x-y)dy.$$

In practice, we would not like to implement the operator $H|D|^{1/2}$ because it involves computing two integral transforms and a derivative. It is less computationally costly to instead compute the equivalent operator

$$H|D|^{1/2} = -i\frac{D}{|D|^{1/2}}$$

where the operator $|D|^{-1/2}$ is represented by the singular convolution integral

$$f(u,x) = \frac{1}{|D|^{1/2}}u(x) = \frac{1}{\sqrt{2\pi}}\int \frac{u(y)}{\sqrt{|x-y|}}dy. \tag{3.17}$$

This reduces the problem to computation of a single integral transform and a derivative. Using this integral representation of $f$, we may write the respective OWWWEs as

$$\frac{\partial u}{\partial t} \pm \frac{1}{\sqrt{2\pi}}\frac{\partial}{\partial x}\int \frac{u(y)}{\sqrt{|x-y|}}dy = 0 \tag{3.18}$$

where in this case, the (+) corresponds to right-going waves and the (−) to left-going waves.

We discuss numerical techniques for approximating solutions to equation (3.18) in Chapter IV, but it is interesting to note here the improved behavior of the one-way equations near the boundary compared to the full WWE. In approximating the solutions to the OWWWE, the integral transform is also truncated by assuming the solution $u \equiv 0$ outside of the computational domain. Assuming $u \equiv 0$ outside of the computational domain is essentially the same as imposing a Dirichlet boundary condition, but since the one way equations allow motion in essentially only one direction, as the outgoing wave reaches the edge of the computational domain it does not reflect and pollute the interior solution in the same way that the solution to the full WWE does. This is illustrated in figure 3.5, where the plots of the solution to the full WWE and the OWWWE over time are shown together.

## 3.2 Properties of the OWWWEs

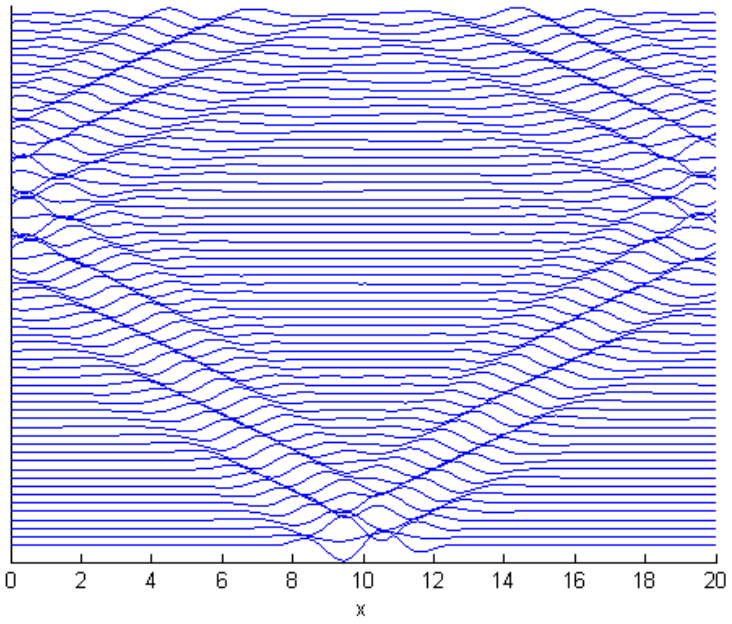
The OWWWE is a fractional partial differential equation (FPDE), involving an operator $(\partial_x |D|^{-1/2})$ which corresponds to half a derivative. In fact, the non-local operator $f$ may be recognized as

$$f(u,x) = \frac{1}{\sqrt{2}} \left( (D_{-\infty+}^{-1/2} u)(x) + (D_{\infty-}^{-1/2} f)(x) \right)$$

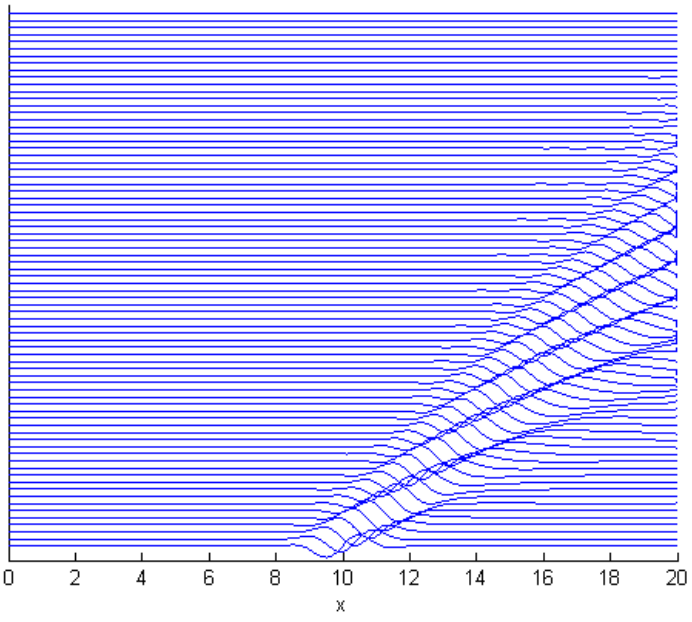
where the operators $D_{-\infty+}^{-1/2}$ and $D_{\infty-}^{-1/2}$ are the left-handed (+) and right-handed (−) Riemann-Liouville integrals of order $\frac{1}{2}$ [39] defined by

$$(D_{-\infty+}^{-1/2} u)(x) = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{x} \frac{u(y)}{\sqrt{x-y}} dy \tag{3.19}$$

$$(D_{\infty-}^{-1/2} u)(x) = \frac{1}{\sqrt{\pi}} \int_{x}^{\infty} \frac{u(y)}{\sqrt{y-x}} dy \tag{3.20}$$




(a)




(b)

Figure 3.5: Figure (a) shows the solution of the WWE computed with initial data (3.9)-(3.10), and figure (b) shows the solution of the right-going OWWWE computed with initial data (3.9), each with increasing time on the vertical axis. Note that in the solution to the WWE, the wave reflects as it reaches the edge of the computational domain, but the one-sidedness of the OWWWE prevents reflections as the wave reaches the edge of the computational domain.

It is only within the past few decades that researchers have begun using fractional order derivatives in models to better incorporate non-local effects. Today, space-fractional partial differential equations in particular arise as models in a variety of applications including sedimentation ([4], [8]), signal denoising ([2]), finance ([1], [6], [15], [34]), and hydrology ([36], [50]). The fractional derivatives are incorporated into the PDEs to represent the solution's dependence on its values upstream and/or downstream in the rest of the domain, whereas in corresponding PDEs with only integer-order derivatives, these effects are neglected. For a more thorough discussion of the theory of fractional calculus, see for example [26], [37], or [43].

The OWWWE may be viewed as a conservation law with a linear nonlocal singular flux with integrable singularity. A typical solution to the OWWWE is shown in figure 3.6 for the right-going equation with wave packet initial data (3.9).

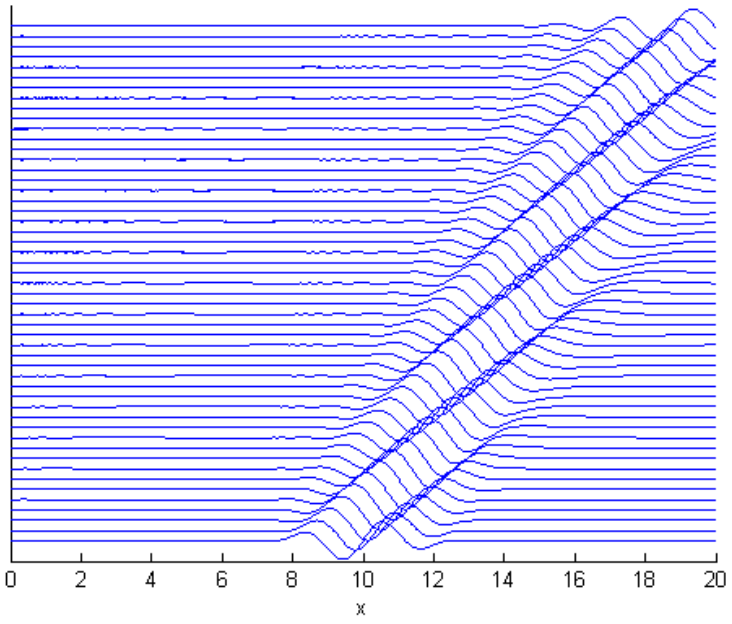



Figure 3.6: Solution of OWWWE with increasing time on the vertical axis, computed with initial data (3.9)

### 3.2.1 Relation to Full WWE

The solutions to the left- and right-going equations can be related back to the solution of the full WWE. The exact solution of the WWE can be written

$$u(x,t) = \int \left( \alpha_1(\xi) e^{-i \operatorname{sgn}(\xi)\sqrt{|2\pi\xi|}t} + \alpha_2(\xi) e^{i \operatorname{sgn}(\xi)\sqrt{|2\pi\xi|}t} \right) e^{2\pi i x \xi} d\xi$$

where the functions $\alpha_1$ and $\alpha_2$ are determined from the initial data

$$u(x,0) = u_0(x) \tag{3.21}$$

$$u_t(x,0) = u_1(x) \tag{3.22}$$

as

$$\alpha_1(\xi) = \frac{1}{2}\left( \frac{\hat{u}_0 i \operatorname{sgn}(\xi)\sqrt{|2\pi\xi|} - \hat{u}_1}{i \operatorname{sgn}(\xi)\sqrt{|2\pi\xi|}} \right)$$
$$\alpha_2(\xi) = \frac{1}{2}\left( \frac{\hat{u}_0 i \operatorname{sgn}(\xi)\sqrt{|2\pi\xi|} + \hat{u}_1}{i \operatorname{sgn}(\xi)\sqrt{|2\pi\xi|}} \right)$$

The data $\alpha_1$ and $\alpha_2$ may be recognized as

$$\alpha_1(\xi) = \frac{1}{2}\mathfrak{F}\left( u_0 - H|D|^{-1/2} u_1 \right)$$
$$\alpha_2(\xi) = \frac{1}{2}\mathfrak{F}\left( u_0 + H|D|^{-1/2} u_1 \right)$$

Similarly, the solution of the one way equation (3.15) can be given as

$$u(x,t) = \int \alpha(\xi) e^{\pm i \operatorname{sgn}(\xi)\sqrt{|2\pi\xi|}t} e^{2\pi i x \xi} d\xi$$

where $\alpha(\xi)$ is the Fourier transform of the initial data for the OWWWE $u(x,0)$, with $(-)$ corresponding to right-going waves and $(+)$ corresponding to left-going waves. From this we see that, for example, when $u_1 \equiv 0$, the solution to the full equation is the sum of the solutions to the left- and right-going one way equations with initial data $\frac{1}{2}u_0$. This can be verified numerically, and figure 3.7 shows the solution to the

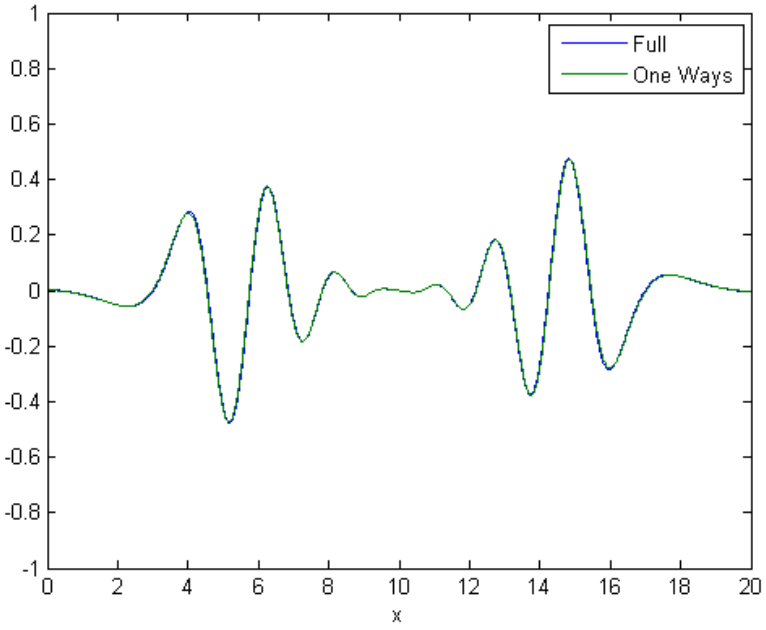



Figure 3.7: Solution of WWE with initial data (3.9)-(3.10) plotted with the sum of the one way solutions with initial data given by half of equation (3.9)

full equation with initial data (3.9)-(3.10) plotted together with the sum of the left- and right-going solutions computed with half the initial data. In general, the full equation with initial data $u_0$ and $u_1$ as in equations (3.21)-(3.22) is equal to the solution of the left-going equation with initial data

$$\frac{1}{2}\left(u_0(x) + H|D|^{-1/2}u_1(x)\right)$$

plus the solution to the right-going equation with initial data

$$\frac{1}{2}\left(u_0(x) - H|D|^{-1/2}u_1(x)\right).$$

### 3.2.2 Asymptotic Behavior at Infinity

As we saw in the previous section, the solution to the OWWWE

$$u_t \mp H|D|^{1/2}u = 0$$

$$u(x,0) = u_0(x)$$

can be written in terms of a Fourier transform as

$$u(x,t) = \int_{-\infty}^{\infty} \widehat{u}_0(\xi) e^{\mp i \operatorname{sgn}\xi \sqrt{|2\pi\xi|}t} e^{2\pi i x\xi} d\xi \tag{3.23}$$

We show, using Theorem 4 from [18], that $u(x,t)$ is $O(x^{-3/2})$ as $|x| \to \infty$ as long as $\widehat{u}_0$ is compactly supported and $\widehat{u}_0 \in C^3(\mathbb{R})$.

**Proposition III.1.** *Let $u(x,t)$ be the solution to the OWWWE (3.15) with initial data $u(x,0) = u_0(x)$ which satisfies*

1. *$\widehat{u}_0(\xi)$ is compactly supported*
2. *The support of $\widehat{u}_0(\xi)$ is contained within the interval $(-a,a)$ for some $a \in \mathbb{R}, \quad a > 0$*
3. *$\widehat{u}_0 \in C^3(\mathbb{R})$*

*Then $u(x,t) = O(x^{-3/2})$ as $|x| \to \infty$.*

*Proof.* Without loss of generality we consider only the right-going solutions; that is, the solutions to

$$u_t - H|D|^{1/2}u = 0$$

Let $t$ be fixed. First we split the integral (3.23) in two, and make the change of variables $\xi = y^2$ for positive $\xi$ and $\xi = -y^2$ for negative $\xi$. Then

$$u(x,t) = \int_0^{\infty} \widehat{u}_0(y^2) e^{\mp i\sqrt{2\pi}yt} e^{2\pi i x y^2} 2y dy + \int_0^{\infty} \widehat{u}_0(-y^2) e^{\pm i\sqrt{2\pi}yt} e^{-2\pi i x y^2} 2y dy$$

Let

$$I_1(x,t) = \int_0^{\sqrt{a}} \widehat{u}_0(y^2) e^{\mp i\sqrt{2\pi}yt} e^{2\pi i x y^2} 2y dy$$
$$I_2(x,t) = \int_0^{\sqrt{a}} \widehat{u}_0(-y^2) e^{\pm i\sqrt{2\pi}yt} e^{-2\pi i x y^2} 2y dy$$

We first derive an asymptotic expansion for each of the integrals $I_1$ and $I_2$, and then sum them to show that $u(x,t) = O(x^{-3/2})$ for large positive $x$. For $I_1$ we have (using the notation of [18])

$$g_1(y) = 2y\widehat{u}_0(y^2)e^{-i\sqrt{2\pi}yt}$$

$$h_1(y) = 2\pi y^2$$

$$\lambda_1 = \mu_1 = \sigma_1 = 1$$

$$\rho_1 = 2$$

with $\alpha = 0$ and $\beta = \sqrt{a}$. This leads to

$$k_1(y) = \frac{1}{\pi} y\widehat{u}_0\left(\frac{y^2}{2\pi}\right) e^{-iyt}$$
$$l_1(y) = -\frac{1}{2\pi}\widehat{u}_0\left(\frac{h_1(\beta) - y}{2\pi}\right) e^{-i\sqrt{h_1(\beta)-y}t}$$

We see that $k_1(0) = 0$. We may write

$$I_1(x,t) = \sum_{n=1}^{N-1} \frac{1}{2(n!)}\Gamma\left(\frac{n+1}{2}\right) k_1^{(n)}(0) e^{\frac{i\pi(n+1)}{4}} x^{-\frac{n+1}{2}} -$$
$$\sum_{n=0}^{N-1} \frac{1}{n!}\Gamma(n+1) l_1^{(n)}(0) e^{-i\pi(n+1)/2} x^{-(n+1)} e^{ixh_1(\sqrt{a})} + o(x^{-N/2})$$

where $g_1(y) \in C^N(\mathbb{R})$.

To consider $I_2$, we first need to make the change of variables $y \to -y$ so that the function $h_2(y)$ will be increasing over the range of integration. Thus

$$I_2(x,t) = -\int_{-\sqrt{a}}^{0} \widehat{u}_0(-y^2) e^{-i\sqrt{2\pi}yt} e^{-2\pi i x y^2} 2y dy$$

For now, we forget about the minus sign in front of the integral and write the

asymptotic expansion for the integral only. We have

$$g_2(y) = 2y\widehat{u}_0(-y^2)e^{-i\sqrt{2\pi}yt}$$

$$h_2(y) = -2\pi y^2$$

$$\lambda_2 = \mu_2 = \rho_2 = 1$$

$$\sigma_2 = 2$$

with $\alpha = -\sqrt{a}$ and $\beta = 0$. This leads us to

$$k_2(y) = -\frac{1}{2\pi}\widehat{u}_0\left(\frac{y + h_2(\alpha)}{2\pi}\right)e^{-i\sqrt{-y-h_2(\alpha)}t}$$
$$l_2(y) = \frac{1}{\pi}y\widehat{u}_0\left(-\frac{y^2}{2\pi}\right)e^{-iyt}$$

Similar to the $I_1$ case, we see that $l_2(0) = 0$. Thus (factoring in the minus sign again)

$$I_2(x,t) = \sum_{n=1}^{N-1}\frac{1}{2(n!)}\Gamma\left(\frac{n+1}{2}\right)l_2^{(n)}(0)e^{-\frac{i\pi(n+1)}{4}}x^{-\frac{n+1}{2}} - \sum_{n=0}^{N-1}\frac{1}{n!}\Gamma(n+1)k_2^{(n)}(0)e^{i\pi(n+1)/2}x^{-(n+1)}e^{ixh_2(-\sqrt{a})} + o(x^{-N/2})$$

where $g_2(y) \in C^N(\mathbb{R})$.

Adding $I_1$ and $I_2$ back together, we have

$$u(x,t) = \sum_{n=1}^{N-1}\frac{1}{2(n!)}\Gamma\left(\frac{n+1}{2}\right)\left(k_1^{(n)}(0)e^{\frac{i\pi(n+1)}{4}} + l_2^{(n)}(0)e^{-\frac{i\pi(n+1)}{4}}\right)x^{-\frac{n+1}{2}} - \sum_{n=0}^{N-1}\frac{1}{n!}\Gamma(n+1)\left(l_1^{(n)}(0)e^{-i\pi(n+1)/2}e^{ixh_1(\sqrt{a})} + k_2^{(n)}(0)e^{i\pi(n+1)/2}e^{ixh_2(-\sqrt{a})}\right)x^{-(n+1)} + o(x^{-N/2})$$

Without any further work, this gives $u(x,t) = O(x^{-1})$ for large $x > 0$ if $\widehat{u}_0 \in C^2(\mathbb{R})$, but if we plug in $k_1'(0)$ and $l_2'(0)$ we find

$$k_1^{(1)}(0)e^{\frac{i\pi(1+1)}{4}} + l_2^{(1)}(0)e^{-\frac{i\pi(1+1)}{4}} = \frac{1}{\pi}\widehat{u}_0(0)\left(e^{i\pi/2} + e^{-i\pi/2}\right)$$
$$= 0$$

and

$$\begin{aligned} l_1^{(0)}(0)e^{-i\pi/2}e^{ixh_1(\sqrt{a})} + k_2^{(0)}(0)e^{i\pi/2}e^{ixh_2(-\sqrt{a})} = & -\frac{1}{2\pi}e^{-i\sqrt{2\pi a}t}(\widehat{u}_0(a)e^{-i\pi/2}e^{ix2\pi a}+ \\ & \widehat{u}_0(-a)e^{i\pi/2}e^{-ix2\pi a}) \\ = & 0 \end{aligned}$$

because $\widehat{u}_0$ has support contained within $(-a, a)$, and finally

$$\begin{aligned} l_1^{(1)}(0)e^{-i\pi}e^{ixh_1(\sqrt{a})} + k_2^{(1)}(0)e^{i\pi}e^{ixh_2(-\sqrt{a})} = & \frac{1}{(2\pi)^2}e^{-i\sqrt{2\pi a}t}(\widehat{u}_0'(a)e^{-i\pi}e^{ix2\pi a} \\ & - \widehat{u}_0'(-a)e^{i\pi}e^{-ix2\pi a})+ \\ & \frac{it}{4\pi\sqrt{2\pi a}}e^{-i\sqrt{2\pi a}t}(\widehat{u}_0(-a)e^{i\pi}e^{-ix2\pi a} \\ & - \widehat{u}_0(a)e^{-i\pi}e^{ix2\pi a}) \\ = & 0 \end{aligned}$$

again because of the compact support of $\widehat{u}_0$. Thus for $\widehat{u}_0 \in C^3(\mathbb{R})$ (so that $g_1, g_2 \in C^3(\mathbb{R})$), the terms with powers of $x$ greater than $-3/2$ disappear, leading to our desired result.

For $x < 0$, we have

$$\begin{aligned} k_1(y) &= -\frac{1}{2\pi}\widehat{u}_0\left(\frac{2\pi a - y}{2\pi}\right)e^{i\sqrt{2\pi a - y}t} \\ l_1(y) &= \frac{1}{\pi}y\widehat{u}_0\left(\frac{y^2}{2\pi}\right)e^{iyt} \\ k_2(y) &= -\frac{1}{\pi}y\widehat{u}_0\left(-\frac{y^2}{2\pi}\right)e^{iyt} \\ l_2(y) &= -\frac{1}{2\pi}\widehat{u}_0\left(\frac{y - 2\pi a}{2\pi}\right)e^{i\sqrt{y - 2\pi a}t} \end{aligned}$$

so that

$$u(x,t) = \sum_{n=1}^{N-1} \frac{1}{2(n!)}\Gamma\left(\frac{n+1}{2}\right) x^{-(n+1)/2}\left(l_1^{(n)}(0)e^{-i\pi(n+1)/4} + k_2^{(n)}(0)e^{i\pi(n+1)/4}\right) - \sum_{n=0}^{N-1} \frac{1}{n!}\Gamma(n+1)x^{-(n+1)}\left(k_1^{(n)}(0)e^{i\pi(n+1)/2}e^{-2\pi ixa} + l_2^{(n)}(0)e^{-i\pi(n+1)/2}e^{2\pi ixa}\right) + o(x^{-N/2})$$

Again, we have

$$l_1^{(1)}(0)e^{-i\pi/2} + k_2^{(n)}(0)e^{i\pi/2} = 0$$

$$k_1^{(0)}(0)e^{i\pi/2}e^{-2\pi ixa} + l_2^{(0)}(0)e^{-i\pi/2}e^{2\pi ixa} = 0$$

$$k_1^{(1)}(0)e^{i\pi}e^{-2\pi ixa} + l_2^{(1)}(0)e^{-i\pi}e^{2\pi ixa} = 0$$

The proof for left-going waves proceeds similarly, with the only difference being the the sign of the power of the exponential term in the formulas for $g_1$ and $g_2$. This does not change the result. □

We can confirm the decay rate of the OWWWE at infinity numerically. Let the initial data $u_0(x)$ be such that its Fourier transform is given by

$$\widehat{u}_0(\xi) = (1+\cos(\xi))^p$$

for $\xi \in [-\pi, \pi]$, and zero otherwise. The Fourier transform of the initial data is therefore an even function of $\xi$, so that the solution $u(x,t)$ to the right-going OWWWE may be written

$$u(x,t) = 2\int_0^\pi (1+\cos(\xi))^p \cos(\sqrt{|2\pi\xi|}t - 2\pi x\xi)d\xi$$

Fix $t = 1$. We compute the integral numerically using, for example, Matlab's "quad" function for a sequence of $x$ values, and we expect $u(x,t) = O(x^{-3/2})$. The results from performing this calculation with $p = 2$ and $p = 4$ are shown in table 3.1, and the expected decay rates are achieved.

| $x$ | $\lvert u(x,1)\rvert, p=2$ | $\lvert u(x,1)\rvert, p=4$ |
|---|---|---|
| $2^2$ | 7.3022E-002 | 2.9316E-001 |
| $2^3$ | 2.3165E-002 | 9.2742E-002 |
| $2^4$ | 7.5750E-003 | 3.0306E-002 |
| $2^5$ | 2.5318E-003 | 1.0128E-002 |
| $2^6$ | 8.4881E-004 | 3.4372E-003 |
| | $\lvert u(x,t)\rvert = O(x^{-1.6047})$ | $\lvert u(x,t)\rvert = O(x^{-1.6023})$ |

Table 3.1: Decay rate at infinity computed with initial data $\widehat{u}_0(\xi) = (1+\cos(\xi))^p$

### 3.2.3 Long Time Decay

We show the long time behavior of solutions to the OWWWEs using van der Corput's lemma as stated in Stein [45].

**Proposition III.2.** *Let $u(x,t)$ be the solution to the OWWWE (3.15) with initial data $u(x,0) = u_0(x)$ which satisfies*

1. *$\widehat{u}_0(\xi)$ is compactly supported*
2. *The support of $\widehat{u}_0(\xi)$ is contained within the interval $(a,b) \subset \mathbb{R}\backslash\{0\}$*
3. *$\widehat{u}_0$ is absolutely continuous*

*Then $u(x,t)$ satisfies*

$$\max_x |u(x,t)| = O(t^{-1/2})$$

*as $t \to \infty$.*

*Proof.* Without loss of generality we consider the right-going OWWWE only. The results for the left-going equation can be obtained similarly. The solution to the right-going OWWWE may be written

$$u(x,t) = \int \widehat{u}_0(\xi) e^{-i \operatorname{sgn}\xi \sqrt{|2\pi\xi|}t} e^{2\pi i x \xi} d\xi$$

By assumption, $\widehat{u}_0$ is compactly supported, and we assume the support lies within an interval $(a, b)$. For each $t$, we have

$$\max_x |u(x,t)| = \max_c |u(ct,t)| \tag{3.24}$$

We proceed by deriving an estimate for $u(ct, t)$ instead of $u(x, t)$. According to the statement of van der Corput's lemma in [45], for phase function $\phi$ which is real valued and smooth in $(a, b)$ with $|\phi''(x)| \geq 1 \quad \forall x \in (a, b)$

$$\left| \int_a^b e^{i\lambda\phi(x)} \psi(x) dx \right| \leq 8\lambda^{-1/2} \left( |\psi(b)| + \int_a^b |\psi'(x)| dx \right)$$

In our case, let

$$\phi(\xi, c) = -\operatorname{sgn}\xi \sqrt{|2\pi\xi|} + 2\pi c\xi$$

Since $(a, b)$ does not include the point $\xi = 0$, $\phi$ is smooth on $(a, b)$. We have

$$\phi_{\xi\xi}(\xi, c) = \pi^2 |2\pi\xi|^{-3/2} \operatorname{sgn}\xi$$

so that

$$|\phi_{\xi\xi}(\xi, c)| \geq \pi^2 \alpha$$

where

$$\alpha = \min\left( \frac{1}{|2\pi a|^{3/2}}, \frac{1}{|2\pi b|^{3/2}} \right)$$

Let

$$\widetilde{\phi}(\xi, c) = (\pi^2\alpha)^{-1} \phi(\xi, c)$$

We know that $|\widetilde{\phi}_{\xi\xi}| \geq 1 \quad \forall (\xi, c) \in (a, b) \times \mathbb{R}$. We may write

$$u(ct, t) = \int_a^b \widehat{u}_0(\xi) e^{i\widetilde{\phi}(\xi,c)\pi^2\alpha t} d\xi$$

and the properties of $\widetilde{\phi}$ together with the assumptions on $\widehat{u}_0$ and van der Corput's lemma imply

$$\begin{aligned} |u(ct,t)| &\leq 8(\pi^2\alpha t)^{-1/2}\left(|\widehat{u}_0(b)| + \int_a^b |\widehat{u}_0'(\xi)|d\xi\right) \\ &= \frac{8\|\widehat{u}_0'\|_{L^1}}{\pi\sqrt{\alpha}}t^{-1/2} \end{aligned}$$

The right hand side is independent of $c$, so we may take the maximum over $c$ of both sides to arrive at the desired result. □

The long-time decay of the (one-dimensional) water wave equation is comparable to the long-time decay of the two-dimensional scalar wave equation. Since waves decay as they propagate, moving the artificial boundary out further makes the waves weaker, and thus easier to absorb. Our experience suggests that the long-time decay rate from Proposition III.2 can also be obtained for a broader class of initial data $u_0$, a point which we illustrate in Chapter IV.

#### 3.2.4 Spectrum of $H|D|^{1/2}$

**Proposition III.3.** *The spectrum of the operator $H|D|^{1/2}$ (with domain $H^{1/2}(\mathbb{R})$) is purely continuous, and given by the purely imaginary numbers.*

*Proof.* The operator $H|D|^{1/2}$, which acts on functions in $H^{1/2}(\mathbb{R})$, is unitarily equivalent (via Fourier transform) to the operator given by multiplication by $-i\,\mathrm{sgn}\,\xi|2\pi\xi|^{1/2}$, with domain $\mathrm{dom}A = \mathcal{F}H^{1/2}(\mathbb{R})$. Thus, these two operators have the same spectrum. Let us consider the spectrum of the latter operator, which we from here on denote as $A$.

For $f \in \mathrm{dom}A$,

$$(Af)(\xi) = (-i\,\mathrm{sgn}\,\xi|2\pi\xi|^{1/2})f(\xi)$$

Consider for $\lambda \in \mathbb{C}$ the operator $C_\lambda = A - \lambda I$. The operator $C_\lambda$ is invertible when $\operatorname{Re}\lambda \neq 0$; the inverse is the operator $B$ given by

$$(Bf)(\xi) = (-i \operatorname{sgn}\xi |2\pi\xi|^{1/2} - \lambda)^{-1} f(\xi)$$

When $\operatorname{Re}\lambda = 0$, however, the operator $C_\lambda$ is not invertible. In this case, we may write

$$\lambda = i\gamma, \qquad \gamma \in \mathbb{R} \tag{3.25}$$

For $\lambda = i\gamma$ as in equation (3.25), the operator $C_\lambda$ may be written

$$C_\lambda f = -i(\operatorname{sgn}\xi |2\pi\xi|^{1/2} + \gamma) f(\xi)$$

The function $\operatorname{sgn}\xi|2\pi\xi|^{1/2} + \gamma = 0$ has a zero at one point which we will call $\xi_\gamma$. For any function $f \in L^2(\mathbb{R})$, the function $f_n$ defined by

$$f_n(\xi) = \begin{cases} \frac{1}{-i(\operatorname{sgn}\xi|2\pi\xi|^{1/2}+\gamma)} f(\xi) & : \text{a.e.} \quad \xi \notin \left(\xi_\gamma - \frac{1}{n}, \xi_\gamma + \frac{1}{n}\right) \\ 0 & : \text{a.e.} \quad \xi \in \left[\xi_\gamma - \frac{1}{n}, \xi_\gamma + \frac{1}{n}\right] \end{cases}$$

is also in $L^2(\mathbb{R})$:

$$\|f_n\|_{L^2}^2 = \int_{\xi \notin \left(\xi_\gamma - \frac{1}{n}, \xi_\gamma + \frac{1}{n}\right)} \left| \frac{1}{\operatorname{sgn}\xi |2\pi\xi|^{1/2} + \gamma} f(\xi) \right|^2 d\xi < \infty$$

since $\xi_\gamma$ is the only zero of $\operatorname{sgn}\xi|2\pi\xi|^{1/2} + \gamma$, the range of integration is bounded away from $\xi_\gamma$, and $f \in L^2(\mathbb{R})$. Moreover, $f_n \in \operatorname{dom} A$:

$$\|Af_n\|_{L^2}^2 = \int_{\xi \notin \left(\xi_\gamma - \frac{1}{n}, \xi_\gamma + \frac{1}{n}\right)} \left| \frac{\operatorname{sgn}\xi |2\pi\xi|^{1/2}}{\operatorname{sgn}\xi |2\pi\xi|^{1/2} + \gamma} f(\xi) \right|^2 d\xi < \infty.$$

This sequence $\{f_n\}$ satisfies

$$\|C_\lambda f_n - f\|_{L^2}^2 = \int_{\xi_\gamma - \frac{1}{n}}^{\xi_\gamma + \frac{1}{n}} |f(\xi)|^2 d\xi \to 0 \quad \text{as} \quad n \to +\infty.$$

Therefore the range of $C_\lambda$ is dense in $L^2(\mathbb{R})$, and all $\lambda$ of the form in equation (3.25) are in the continuous spectrum of the operator $A$. Thus we see that the spectrum of the operator $A$ (and therefore, the operator $H|D|^{1/2}$ with domain $H^{1/2}(\mathbb{R})$) is purely continuous, and given by the purely imaginary numbers, $\{z \in \mathbb{C} \quad | \quad \operatorname{Re} z = 0\}$. $\square$

### 3.2.5 Conservation of Energy

**Proposition III.4.** (Energy Estimate for the OWWWE) *A solution* $u(x,t)$ *to the OWWWE* (3.15) *with initial data* $u(x,0) = u_0(x)$ *satisfies*

$$\| u(\cdot,t) \|_{L^2} = \| u(\cdot,0) \|_{L^2}$$

*Proof.* The solution to the OWWWE satisfies

$$\widehat{u}(\xi,t) = e^{\pm i \operatorname{sgn}\xi |2\pi\xi|^{1/2} t} \, \widehat{u}(\xi,0)$$

Therefore,

$$\| \widehat{u}(\cdot,t) \|_{L^2} = \| \widehat{u}(\cdot,0) \|_{L^2}$$

so that by Plancherel's theorem, the desired result is obtained. □

## CHAPTER IV

# Numerical Methods for the One Way Water Wave Equation

In the previous chapter we introduced the one way water wave equation (OWWWE)

$$\frac{\partial u(x,t)}{\partial t} \mp \frac{1}{\sqrt{2\pi}} \frac{\partial}{\partial x} \int \frac{u(y,t)}{\sqrt{|x-y|}} dy = 0 \tag{4.1}$$

It is a fractional partial differential equation (FPDE) involving an operator which corresponds to half a derivative. We view equation (4.1) as a conservation law, with a nonlocal linear flux involving convolution with a locally integrable singular kernel.

In this chapter, we are concerned with efficient numerical methods for solving (4.1). As remarked in Chapter III, the nonlocal flux $f$ given by

$$f(u,x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \frac{u(y)}{\sqrt{|x-y|}} dy \tag{4.2}$$

may be recognized as

$$f(u,x) = \frac{1}{\sqrt{2}} \left( (D_{-\infty+}^{-1/2} u)(x) + (D_{\infty-}^{-1/2} f)(x) \right)$$

where the operators $D_{-\infty+}^{-1/2}$ and $D_{\infty-}^{-1/2}$ are the left-handed $(+)$ and right-handed $(-)$ Riemann-Liouville integrals of order $\frac{1}{2}$ (see equations (3.19)-(3.20)).

Numerical approaches to approximate this type of integral transform include standard quadrature rules ([2], [7], [13], [15]), integration by parts to yield convolution involving $u_x$, which is then approximated by finite difference ([11], [33]), and the

shifted Grünwald formula presented in [35], which utilizes an alternative definition of the fractional derivative ([10], [14]). Most relevant to the numerical methods derived in this chapter are [31] and [21]. In [31], numerical methods for fractional integrals are studied using a finite difference technique. Li et al. present three algorithms for approximating the fractional integral

$$(D_{a+}^{\alpha}u)(x) = \frac{1}{\Gamma(\alpha)}\int_a^x (x-y)^{\alpha-1}u(y)dy, \qquad \alpha > 0,$$

all of which involve replacing the function $u$ with a polynomial and integrating the transform of the polynomial exactly. For convenience, we call this an Exact Polynomial Integration Computation (EPIC). In the second order method $u$ is replaced by a linear spline, and in the fourth order methods $u$ is replaced either by a piecewise cubic function or by a cubic spline. In the cubic cases, the polynomials are computed using first and possibly second order derivatives of $u$, which if unknown are approximated using finite difference techniques. The authors present a numerical method for solving a fractional ODE, but approximate the fractional derivative in the ODE using the product trapezoidal quadrature formula or compound Simpson formula. In [21], the author presents several techniques for computing the fractional derivative operator in a fractional diffusion equation

$$u_t - c|D|^{\alpha}u = 0$$

where $1 \leq \alpha \leq 2$, including approximating the function $u$ by a linear spline $g$ and integrating $|D|^{\alpha}g$ exactly. The author also argues the benefit of writing the fractional diffusion equation in flux form, in order to construct conservative computational algorithms.

In this chapter, we develop a hierarchy of efficient numerical methods to solve the fractional partial differential equation (3.15) using EPIC techniques for polynomials

of increasing accuracy. We adopt numerical approaches from hyperbolic conservation laws to compute the flux (4.2) by using conservative piecewise polynomial reconstruction of the solution [25]. Solution cell averages are used to reconstruct the solution in each computational cell, and the flux at cell interfaces is computed by evaluating *exactly* the integral transform associated with the singular convolution. No information about solution derivatives is required, and to achieve a higher order of spatial accuracy one need only increase the order of the interpolating polynomial. Time integration uses Runge-Kutta (RK) schemes of matching order. We analyze the stability of the resulting schemes, study the convergence of the numerical solution, and present numerical results.

## 4.1 Numerical Scheme

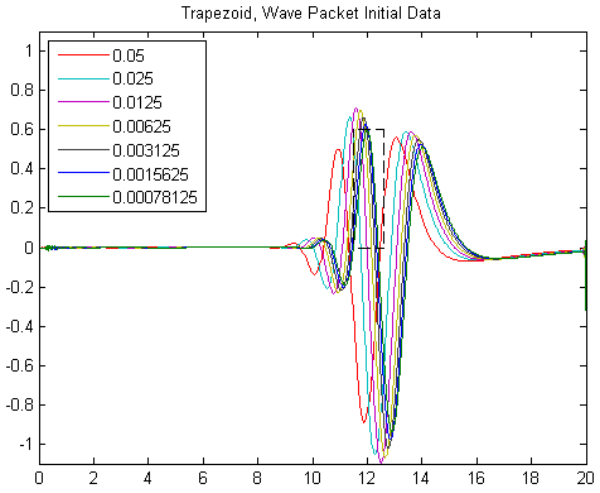
We specialize the discussion to right-going wave propagation only. Left-going wave propagation is a mirror image and is approximated in complete analogy.

For comparison, figures 4.1a - 4.1b shows the solution to the OWWWE at time $T$ with transform computed with the trapezoid rule and time derivative approximated with forward Euler on a sequence of grids. By the finest grid picture, the solution has yet to converge, and is already taking greater than 14 hours to compute. A GNU library routine for adaptive quadrature numerical integration of integrals with singularities failed to converge with the desired tolerance. In Section 4.3.2, we show similar figures for our higher-order schemes and compare to the results from the trapezoid rule.

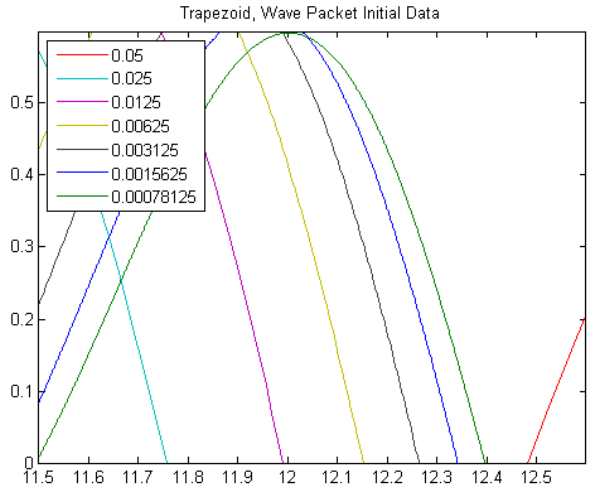
We write the one way water wave equation (OWWWE) (3.15) as

$$u_t + f(u, x)_x = 0 \tag{4.3}$$

where $f$ is as in (4.2).




(a)




(b)

Figure 4.1: Solution $u(x,T)$ to OWWWE with initial data (3.9) for a sequence of grids as computed using the trapezoid rule. (b) shows the zoomed in area indicated by the box in figure (a).

Equation (4.3) is a linear conservation law with a non-local flux $f$ involving a convolution with a singular integrable kernel. We begin by writing the equation in semi-discrete finite volume framework. We introduce a spatial grid with uniform space intervals $I_j = [x_{j-1/2}, x_{j+1/2}]$ of length $\Delta x$ and centers at $x_j = (j+1/2)\Delta x$. We denote by $\overline{u}_j(t)$ the cell-average of the solution $u(\cdot,t)$ over the corresponding

intervals $I_j$

$$\overline{u}_j(t) = \frac{1}{\Delta x}\int_{I_j} u(x,t)\mathrm{d}x \ .$$

Integrating (4.3) over $I_j$ and dividing by $\Delta x$ gives

$$\frac{\mathrm{d}}{\mathrm{d}t}\overline{u}_j(t) = \frac{1}{\Delta x}\left(f(u(\cdot,t),x_{j+1/2}) - f(u(\cdot,t),x_{j-1/2})\right) \ . \tag{4.4}$$

A semi-discrete finite volume scheme is defined by computing the approximate intercell flux $F(u(\cdot,t),x)$ at $x_{j\pm1/2}$, denoted here as $F_{j\pm1/2}(t)$.

### 4.1.1 Conservative Polynomial Reconstruction of $u$

We proceed by approximating $u(\cdot,t)$ by a conservative piecewise polynomial interpolant

$$u(x,t) \approx \mathcal{P}(x,t) = \sum_j \mathcal{P}_j(x,t)\chi_{I_j}(x) \tag{4.5}$$

where the piecewise polynomial $\mathcal{P}$ is reconstructed at each time step from the cell averages at time $t$, $\{\overline{u}_j(t)\}$. This well-known procedure is described below for completeness, since it is put to use in a new context. We omit the time dependence from all reconstructed quantities, but time dependence is implied. Let $v(x)$ be the primitive function of $u(x)$

$$v(x) = \int^x u(y)dy$$

with intercell values

$$v_{j+\frac{1}{2}} = v(x_{j+\frac{1}{2}}) = \int^{x+\frac{1}{2}} u(y)dy = \sum^j \int_{I_i} u(y)dy = \sum^j \Delta x\overline{u}_i$$

and consider the local polynomial of degree $r$ which interpolates $v(x)$ at $r+1$ consecutive cell interfaces straddling $I_j$

$$v(x) \approx p_j(x) = \sum_{l=0}^{r} c_l(x-x_j)^l \qquad x \in I_j \ .$$

We have written the polynomial coefficients here as $c_l$, but the coefficients really depend not just on the power $l$ but also on the choice of stencil, order of polynomial interpolant, and gridcell. For a polynomial interpolant of order $r$, there are $r$ possible stencils to choose from, and we enumerate them by the index $k = 0, ..., r-1$. Computing the coefficients of the interpolant requires knowing the values of $v$ at the endpoints of $r$ gridcells, and the index $k$ indicates which set of gridcells are being used. When $k = 0$, to compute the polynomial coefficients on $I_j$ requires data from the set of $r$ contiguous cells whose rightmost cell is $I_j$, and when $k = r-1$ computing the polynomial coefficients on $I_j$ requires data from the set of $r$ contiguous cells whose leftmost cell is $I_j$. Intermediate values of $k$ use intermediate stencils. Table 4.1 gives a visualization of the stencils used for various $(r, k)$ pairs.

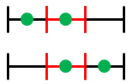
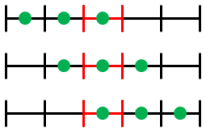
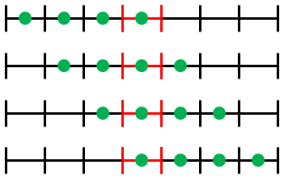
| Polynomial Order | Interpolation Order $r$ | Stencil $k$ | Stencil Schematic |
|---|---|---|---|
| Linear | 2 | 0 |  |
| | 2 | 1 | |
| Quadratic | 3 | 0 |  |
| | 3 | 1 | |
| | 3 | 2 | |
| Cubic | 4 | 0 |  |
| | 4 | 1 | |
| | 4 | 2 | |
| | 4 | 3 | |

Table 4.1: Visualization of stencils used for interpolation.

In each gridcell $I_j$, the coefficients $c_l$ for interpolant order / stencil pair $(r, k)$ are

determined by solving the system of equations

$$v_{j+k-r+1/2} = \sum_{l=0}^{r} c_l (x_{j+k-r+1/2} - x_j)^l$$

$$\cdots$$

$$v_{j+k+1/2} = \sum_{l=0}^{r} c_l (x_{j+k+1/2} - x_j)^l$$

The coefficients $c_l$ in terms of $j$ are given in table 4.2 for values of $r$ and $k$ which give stable numerical methods for the one way water wave equation (up to $r = 4$).

Once we have computed the coefficients $c_l$ of the polynomial $p_j$, the polynomial approximation of $u(x)$ on $I_j$ is recovered by differentiating

$$u(x) \approx p_j'(x) = \mathcal{P}_j(x) \ , \qquad x \in I_j \tag{4.6}$$

with obvious relation between the coefficients of $\mathcal{P}_j(x)$ and $p_j(x)$. On each interval $I_j$, the polynomial reconstruction satisfies

$$(i)\ p_j(x) = v_{j-\frac{1}{2}} + \int_{x_{j-\frac{1}{2}}}^{x} v(y)dy \ , \qquad (ii)\ \frac{1}{\Delta x}\int_{I_j} \mathcal{P}_j(y)dy = \overline{u}_j$$

### 4.1.2 Flux Approximation

To approximate the flux at cell interfaces, we replace $u(y)$ by its local polynomial approximation, $u(y) \approx \mathcal{P}_j(y)$ for $x \in I_j$,

$$\begin{aligned}
f_{j+\frac{1}{2}} &= \frac{1}{\sqrt{2\pi}} \int \frac{u(y)}{\sqrt{|x_{j+\frac{1}{2}} - y|}} dy \\
&= \frac{1}{\sqrt{2\pi}} \int^{x^-_{j+\frac{1}{2}}} \frac{u(y)}{\sqrt{x_{j+\frac{1}{2}} - y}} dy + \frac{1}{\sqrt{2\pi}} \int_{x^+_{j+\frac{1}{2}}} \frac{u(y)}{\sqrt{y - x_{j+\frac{1}{2}}}} dy \\
&= \frac{1}{\sqrt{2\pi}} \left( \sum_{i \le j} \int_{I_i} \frac{u(y)}{\sqrt{x_{j+1/2} - y}} dy + \sum_{i > j} \int_{I_i} \frac{u(y)}{\sqrt{y - x_{j+1/2}}} dy \right) \\
&\approx \frac{1}{\sqrt{2\pi}} \left( \sum_{i \le j} \int_{I_i} \frac{\mathcal{P}_i(y)}{\sqrt{x_{j+1/2} - y}} dy + \sum_{i > j} \int_{I_i} \frac{\mathcal{P}_i(y)}{\sqrt{y - x_{j+1/2}}} dy \right) \\
&= \frac{1}{\sqrt{2\pi}} \left( (I_+ u)(x_{j+1/2}) + (I_- u)(x_{j+1/2}) \right) = F_{j+1/2}
\end{aligned}$$

The integrals involve expressions of the form

$$\begin{aligned}
(I_+ u)(x_{j+1/2}) &= \sum_{i=0}^{j} \sum_{l=0}^{r-1} C_l \int_{I_i} \frac{(y - x_i)^l}{\sqrt{x_{j+1/2} - y}} dy \\
(I_- u)(x_{j+1/2}) &= \sum_{i=j+1}^{N-1} \sum_{l=0}^{r-1} C_l \int_{I_i} \frac{(y - x_i)^l}{\sqrt{y - x_{j+1/2}}} dy
\end{aligned}$$

and require the evaluation of integral transform of monomials

$$\begin{aligned}
I^{i,l}_+(x_{j+1/2}) &= \int_{I_i} \frac{(y - x_i)^l}{\sqrt{x_{j+1/2} - y}} dy \\
I^{i,l}_-(x_{j+1/2}) &= \int_{I_i} \frac{(y - x_i)^l}{\sqrt{y - x_{j+1/2}}} dy.
\end{aligned}$$

These integrals can be evaluated *exactly*. We also note that these integrals are independent of the solution. Once a grid is specified, they may be pre-computed *once* and stored.

Making the change of variables so that only $\sqrt{y}$ remains in the denominator of

each integral

$$I_+^{i,l}(x_{j+1/2}) = \int_{x_{j+1/2}-x_{i+1/2}}^{x_{j+1/2}-x_{i-1/2}} \frac{(x_{j+1/2}-x_i-y)^l}{\sqrt{y}}dy \quad = \int_{(j-i)h}^{(j-i+1)h} \frac{((j-i+1/2)h-y)^l}{\sqrt{y}}dy$$
$$I_-^{i,l}(x_{j+1/2}) = \int_{x_{i-1/2}-x_{j+1/2}}^{x_{i+1/2}-x_{j+1/2}} \frac{(y+x_{j+1/2}-x_i)^l}{\sqrt{y}}dy \quad = \int_{(i-j-1)h}^{(i-j)h} \frac{(y-(i-j-1/2)h)^l}{\sqrt{y}}dy$$

we see that the integrals $I_+$ and $I_-$ may be related to the single integral

$$w[l][i] = \int_{I_i} \frac{(y-x_i)^l}{\sqrt{y}}dy = \int_{ih}^{(i+1)h} \frac{(y-(i+1/2)h)^l}{\sqrt{y}}dy.$$

That is,

$$I_+^{i,l}(x_{j+1/2}) = (-1)^l w[l][j-i]$$
$$I_-^{i,l}(x_{j+1/2}) = w[l][i-j-1].$$

The elements of the array $w$ can be found in table 4.3. Given this representation of the integrals $I_+^{i,l}$ and $I_-^{i,l}$, we rewrite the approximation of the flux function $f$ as

$$f(u, x_{j+1/2}) \approx f(\mathcal{P}, x_{j+1/2}) = \frac{1}{\sqrt{2\pi}}\Bigg( \sum_{i=0}^{j}\sum_{l=0}^{r-1}(l+1)c_{l+1}(-1)^l w[l][j-i] + \sum_{i=j+1}^{N-1}\sum_{l=0}^{r-1}(l+1)c_{l+1}w[l][i-j-1]\Bigg)$$

so that to form the intercell flux, the coefficients of the polynomial reconstruction $\mathcal{P}_i(y)$ of $u(y)$ in cell $I_i$ are multiplied by the integrated monomials over the cell, and summed over all cells $I_j$.

*Remark* IV.1. In hyperbolic conservation laws, there is often interest in discontinuous solutions. Consequently, numerical schemes often include numerical mechanisms to suppress/control oscillations near solution discontinuities. For example, ENO reconstruction chooses the stencil adaptively, WENO adapts the weights. Left/right approximations of the intercell flux are combined to ensure the solution is TVD. In the present context, we are not primarily concerned with discontinuities, and have

used fixed stencils in the reconstruction process. For flows involving large gradients or discontinuities, stencil adaptivity may be easily incorporated.

*Remark* IV.2. We fix the choice of stencil in our computations, but the boundaries require special treatment. We only have data $\overline{U}_j$ in cells $\{I_0, ..., I_{N-1}\}$. The polynomial coefficients $c_l$ may depend on the values of $\{v_{j+k-r+1/2}, ..., v_{j+k+1/2}\}$, where we have defined $v_{j+1/2} = \sum^j \Delta x \overline{u}_i$. Due to limited data, this sum may only be computed with indices from $i = 0$ to $j$, so that when $j < 0$, this sum will be computed to be zero. Computing $v_{j+1/2}$ for $j > N-1$ also involves unknown values of $\overline{u_j}$. To avoid this issue and maintain higher order accuracy in boundary cells, we shift the stencils used for the polynomial interpolation as the cell approaches the boundary. Precisely, in cells $\{I_j\}$ on the left boundary for $j = 0, 1, ..., r-(k+1)$, we shift from stencil $k$ to stencil $r-1+j$, while in cells $\{I_j\}$ on the right boundary for $j = N-1, N-2, ..., N-(k+1)$ we shift from stencil $k$ to stencil $N-1-j$.

#### 4.1.3 Accuracy of Scheme

The spatial accuracy of the resulting scheme depends on the order of the piecewise polynomial reconstruction (4.6). On each interval $I_j$, for sufficiently smooth $u$ the polynomial approximant satisfies $\mathcal{P}_j(x) = u(x) + O(\Delta x^r)$, from which it follows that there exists positive constant $\gamma_j$ such that for sufficiently small $\Delta x$

$$|u(x) - \mathcal{P}_j(x)| \leq \gamma_j \Delta x^r \tag{4.7}$$

For a function $u$ with support in $[-L, L]$, the error in integrating $\mathcal{P}_j$ instead of $u$ is

$$\begin{aligned}\left| \int \frac{u(y)}{\sqrt{|x-y|}} dy - \sum_j \int_{I_j} \frac{\mathcal{P}_j(y)}{\sqrt{|x-y|}} dy \right| &= \left| \sum_j \int \frac{u(y) - \mathcal{P}_j(y)}{\sqrt{|x-y|}} dy \right| \\ &\leq \sum_j \gamma_j \Delta x^r \int \frac{1}{\sqrt{|x-y|}} dy \\ &= \sum_j \gamma_j \Delta x^r 2(\sqrt{L+x} + \sqrt{L-x}) \\ &\leq 4\sqrt{2L} \sum_j \gamma_j \Delta x^r \end{aligned} \tag{4.8}$$

where $|x|$ is assumed to be in $[0, L]$. Thus the flux approximation inherits the order of approximation of the polynomial reconstruction.

The resulting semi-discrete approximation of (3.15) forms a system of time dependent ODEs. To obtain a fully-discrete scheme, (4.4) needs to be integrated in time by a stable ODE solver. We have used Runge-Kutta schemes of matching order to the spatial discretization and determined the time step for stable integration as described below.

*Remark* IV.3. The nonlocal flux function $f(u, x)$ involves an integral on the unbounded domain, with a square root decay of far field contributions to the flux at $x$. In the numerical implementation of the scheme, solution values outside the computational domain are assumed to be zero. This inevitably results in boundary errors. The one-sided character of the OWWWE prevents those errors from propagating back into the domain and polluting the solution.

## 4.2 Numerical Stability

The intercell flux $f_{j+1/2}(t) = f(u(\cdot, t), x_{j+1/2})$ is a linear function of $u$, implying that the semi-discrete numerical approximation of equation (4.3) has a matrix

representation

$$\frac{d}{dt}\overline{U}(t) = -\frac{1}{\sqrt{\Delta x}}A\overline{U}(t) \tag{4.9}$$

where $\overline{U}(t) = (\bar{u}_1(t), \bar{u}_2(t), ..., \bar{u}_N(t))$ are the cell averages of the numerical solution, and $A$ is an $N \times N$ matrix, representing spatial derivatives of the integral transform $\frac{\partial}{\partial x}f(u,x)$. Due to the nonlocal nature of the transform $f$, $A$ is a dense matrix. The stability bounds are given in terms of the parameter

$$\nu = \frac{\Delta t}{\sqrt{\Delta x}} \tag{4.10}$$

as should be expected from a FPDE involving half a derivative. The eigenvalues of the matrix $A$, denoted by $\lambda_k,\ k = 1, \cdots, N$, cannot be obtained analytically, but can be computed numerically (for example using the 'eig' function of MatLab), and are then used to identify stability regions for the various numerical schemes.

The matrix $A$ is independent of the numerical solution $\bar{u}_j(t)$, and the stability region needs to be computed only once. Furthermore, our numerical studies show that as the grid is refined or as $L$ is increased, the eigenvalues of the corresponding matrix $A$ appear to converge to a well-defined curve. Further refinement seem to only fill in the corresponding eigenvalue curve, but not change its shape. We have used a reasonably converged eigenvalue curve to obtain accurate stability estimates.

We have used Runge-Kutta ODE solvers to integrate the semi-discrete solution of (4.4) in time. For a given RK integrator, we denote by $\Gamma = \{z : |g(z)| = 1\}$ the boundary of the stability region, and determine the allowable $\nu$ for stable integration by requiring that $-\nu\lambda_k$ is inside the stability region enclosed by $\Gamma$, for all eigenvalues $\lambda_k$. That is, we require

$$\max_k |g(-\nu\lambda_k)| \leq 1$$

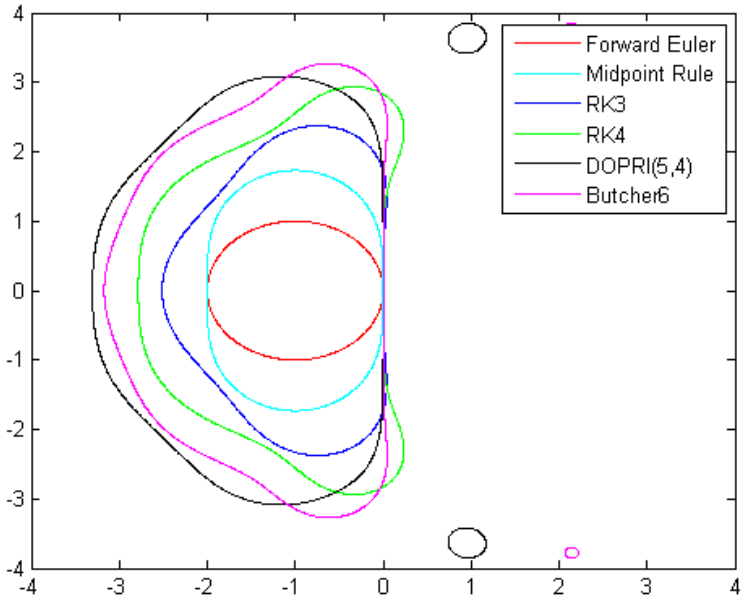


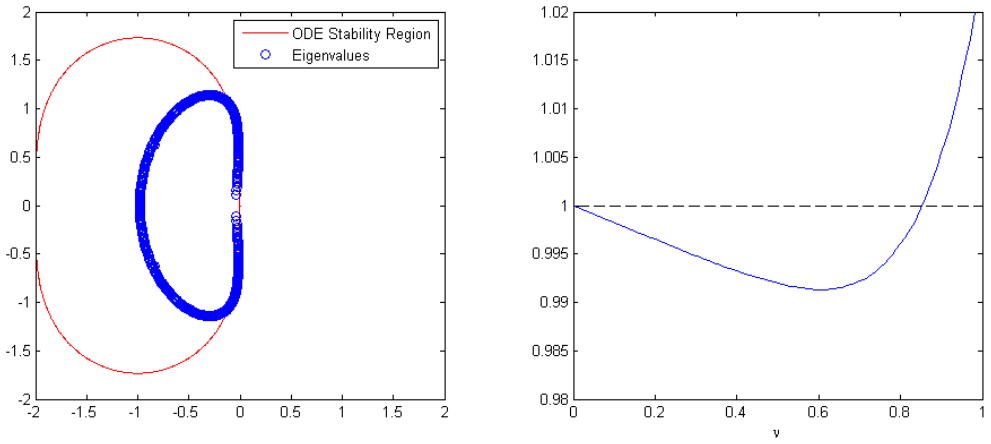

Figure 4.2: Stability regions of various ODE solvers

Figure 4.2 shows the stability regions for various ODE solvers.

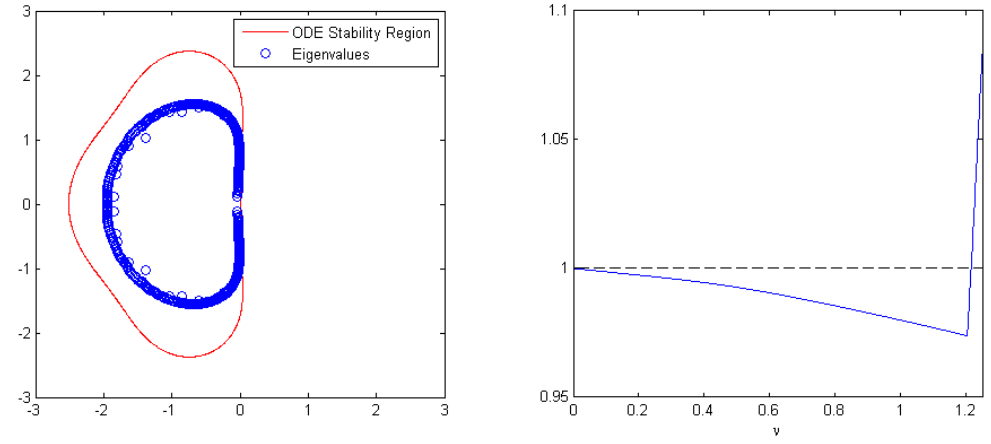
Figures 4.3a-4.3d show the eigenvalues $-\lambda_k$ of $-A$ for various order of spatial reconstruction/stencil pairs $(r, k)$. The eigenvalues are plotted together with the corresponding RK stability region $\Gamma$. Also plotted is $\max_k |g(-\nu\lambda_k)|$ versus $\nu$, from which the bound on $\nu$ is estimated. These three polynomial order/stencil pairs are stable for small $\nu$ for the right-going problem, but are unstable for the left-going problem. By contrast, figure 4.3e shows a polynomial/stencil pair which is only stable for the left-going problem; $\max_k |g(\nu\lambda_k)|$ is plotted since the left-going problem is represented by $u_t = \Delta x^{-1/2} Au$. Finally, figure 4.3f shows a polynomial/stencil pair which is not stable for either the right-going or left-going problem. Table 4.4 summarizes the stability constraints we found for each polynomial order and stencil. In particular, from this table we see that downwind biased stencils and heavily upwind biased stencils are not stable for any $\nu$.

Also of interest is the location of the eigenvalues. As shown in Proposition III.3,

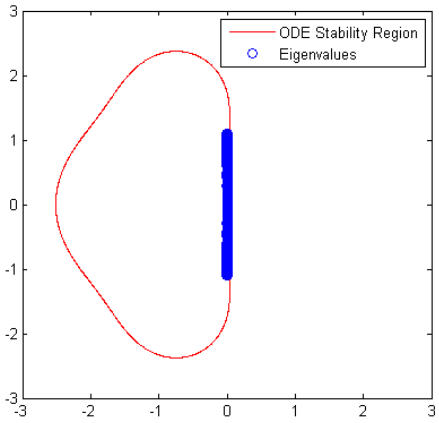
for the continuous problem the operator $\frac{\partial}{\partial x} f(u, x)$ has purely continuous spectrum given by the purely imaginary numbers. Figures 4.3a to 4.3f show that in general, our method introduces some dissipation due to the nonzero real part of the eigenvalues $\pm\lambda_k$. Symmetric stencils, however, as seen in figure 4.3c, are stable for both left- and right-going equations and have all strictly imaginary eigenvalues. For these stencils, other time integrators which include more of the imaginary axis may be better suited.

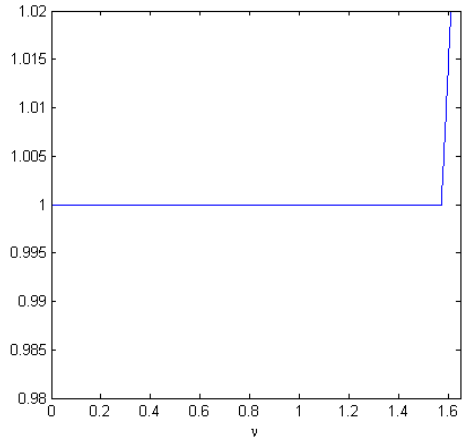



(a)

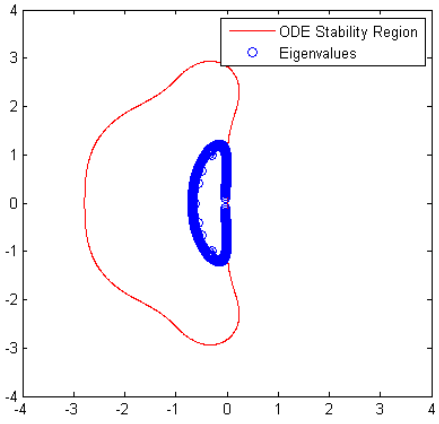


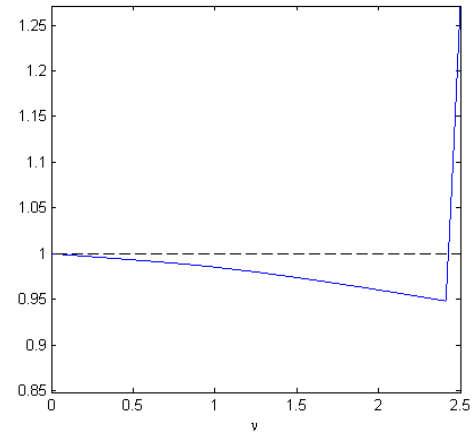

(b)


ODE Stability Region
Eigenvalues

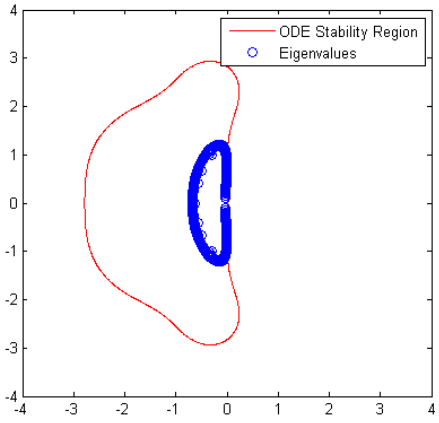

ν

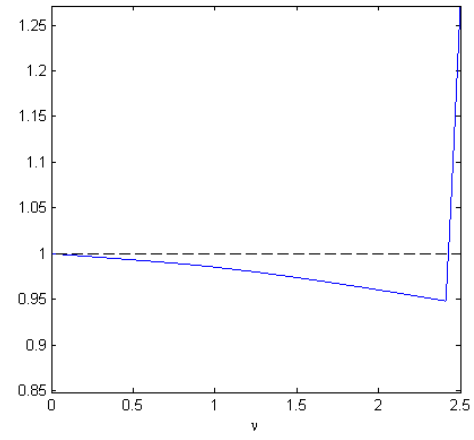
(c)


ODE Stability Region
Eigenvalues

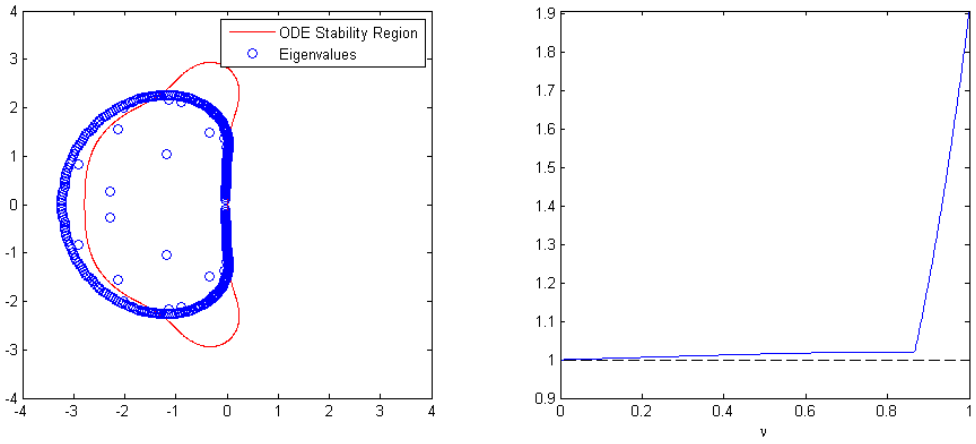

ν

(d)


ODE Stability Region
Eigenvalues


ν

(e)




(f)

Figure 4.3: Computed eigenvalues plotted with RK stability regions for various polynomial order / stencil pairs. Figures (a), (b), (c), (d), and (f) show eigenvalues $-\lambda_k$ plotted with RK stability region on the left and $\max_k |g(-\nu\lambda_k)|$ on the right; (e) shows the eigenvalues $\lambda_k$ plotted with RK stability region on the left and $\max_k |g(\nu\lambda_k)|$ on the right. (a): linear stencil 0, (b): quadratic stencil 0, (c): quadratic stencil 1, (d): cubic stencil 1, (e): cubic stencil 2, and (f): cubic stencil 0.

## 4.3 Numerical Results

### 4.3.1 Accuracy of Polynomial Interpolation and Integral Transform

In Section 4.1.3, we derived the expected order of accuracy for the polynomial interpolant (equation (4.7)) and for the flux (equation (4.8)) for polynomials of order $r-1$. In this section, to verify that the numerical schemes achieve the expected order of accuracy we present the following numerical examples. In each example, we compute the numerical error for a set of four polynomial and stencil pairs. First we estimate the error from the polynomial approximation by computing the polynomial coefficients and using them to compute values of the polynomial interpolant on 100 points in each gridcell. These values are compared to the known exact values of $u$ to generate the error

$$E_{poly} = \| \, u(x) - \mathcal{P}(x) \, \| \leq M_{poly} h_{poly}^{p}$$

where $M_{poly}$ is a constant for each $(r, k)$ pair.

Next we compute the integral transform for a sequence of $h$ values and estimate the error

$$E_{trans} = \| f(u,x) - f(\mathcal{P},x) \| \leq M_{trans} h_{trans}^{p}$$

by comparing $f(\mathcal{P})^h$ to $f(\mathcal{P})^{h/2}$, $f(\mathcal{P})^{h/2}$ to $f(\mathcal{P})^{h/4}$, etc. Again, $M_{trans}$ is a constant which depends on $r$ and $k$. $L^2$ errors as well as computed values of $M$ and $p$ are given in tables with each example.

**Example 1.** Consider the function

$$u(x) = \sin(\pi/Lx), \qquad x \in [0, L], \qquad L = 20$$

Table 4.5 shows the $L^2$ error in approximating $u$ by a polynomial for the various $(r,k)$ pairs, and table 4.6 shows the resultant values of $M_{poly}$ and $p_{poly}$.

Table 4.7 shows the approximate $L^2$ error in computing the integral transform for the various $(r,k)$ pairs, and table 4.8 shows the resultant values of $M_{trans}$ and $p_{trans}$.

**Example 2.** Consider the function

$$u(x) = \begin{cases} \cos^6\left(\frac{\pi}{6}x - \frac{5\pi}{3}\right) \sin\left(5\left(\frac{\pi}{6}x - \frac{5\pi}{3}\right)\right) & : 7 \leq x \leq 13 \\ 0 & : \text{else} \end{cases}$$

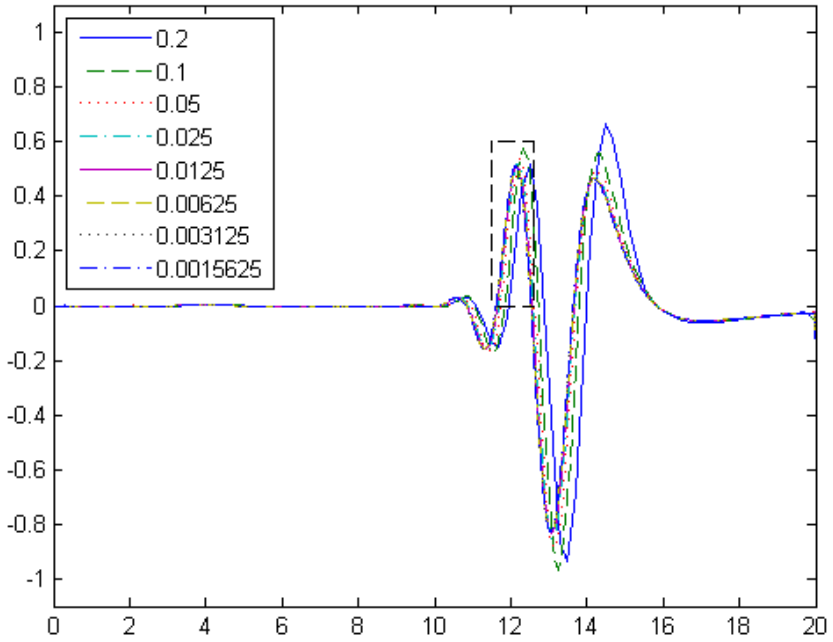
Table 4.9 shows the $L^2$ error in approximating $u$ by a polynomial for various values of $(r,k)$, and table 4.10 shows the resultant values of $M_{poly}$ and $p_{poly}$.

Table 4.11 shows the approximate $L^2$ error in computing the integral transform for various values of $(r,k)$, and table 4.12 shows the resultant values of $M_{poly}$ and $p_{poly}$.

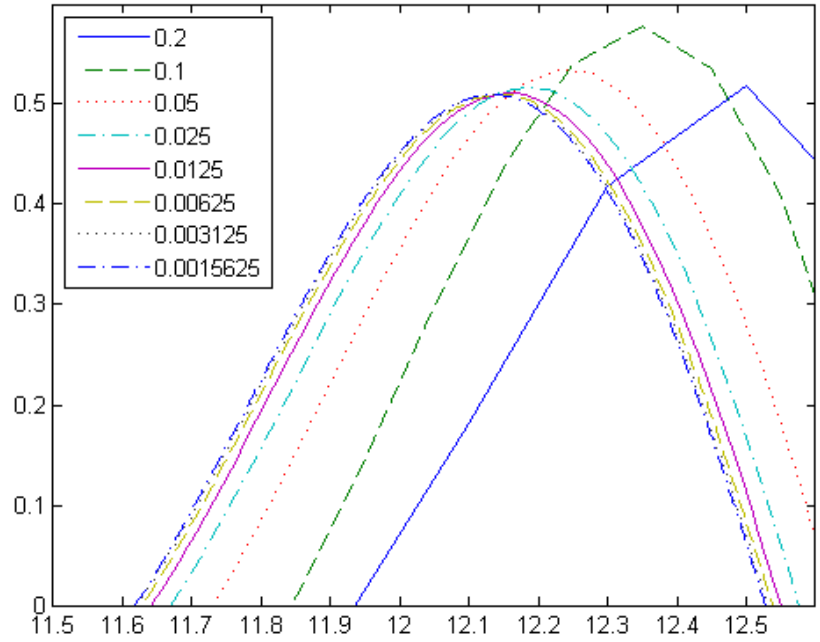
### 4.3.2 Accuracy of Solution to the OWWWE

The numerical method described in Section 4.1 was implemented to solve the time-dependent OWWWE. While the EPIC methods clearly give faster convergence

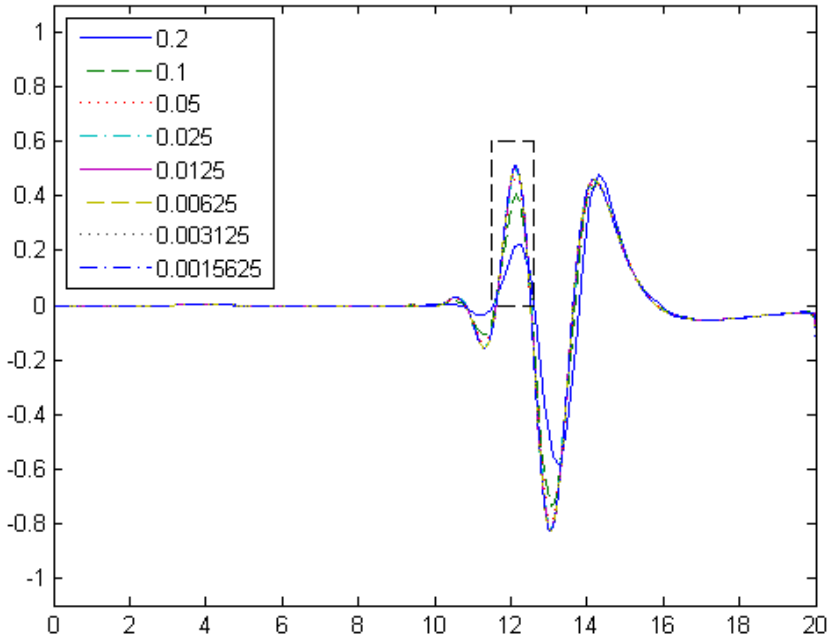
and better accuracy, we were unable to establish the expected higher rate of convergence. The polynomial interpolation algorithm achieves the expected accuracy (4.7) when $u$ is sufficiently smooth. It is easy to enforce the necessary smoothness with the initial data, but once solutions to the OWWWE are computed and waves approach the boundaries of the computational domain, a discontinuity appears at the boundaries (outside of which, the solution is assumed to be zero) which we suspect affects the numerical accuracy of solutions. In the following figures, the solution to the OWWWE is shown on a sequence of grids for a selection of interpolant order and stencil pairs. The figures illustrate the increasing order of accuracy being achieved by polynomials of increasing order. The solutions were computed on either a 2.53 GHz Intel Xeon E5540 processor with 48 GB RAM or a 2.67 GHz Intel Xeon X5650 processor with 48 GB RAM. On average, it takes five times as long to compute on a grid with stepsize $\Delta x/2$ as compared to a grid with stepsize $\Delta x$ for any given interpolant order. It takes on average somewhere between 1.5-2 times as long to compute with the same grid using the polynomial interpolant of next highest order. From the grid convergence plots in figure 4.4, one can see that there is a clear advantage to using a higher-order scheme rather than refining the grid; typically, one can get away with computing on the next coarsest grid with the next highest order polynomial interpolant and achieve a *savings* of at least sixty percent of computational time, with comparable results. In addition, compare for example the reasonably converged solution using cubic, stencil 1 and $\Delta x = 0.0125$ to the results for the solution computed on the finest grid (still not converged!) with the trapezoid rule; it takes over 550 times as long to produce the solution using the trapezoid rule, and the quality is much worse! Clearly there is a significant advantage from computing using the EPIC scheme.


0.2
0.1
0.05
0.025
0.0125
0.00625
0.003125
0.0015625
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
2
4
6
8
10
12
14
16
18
20

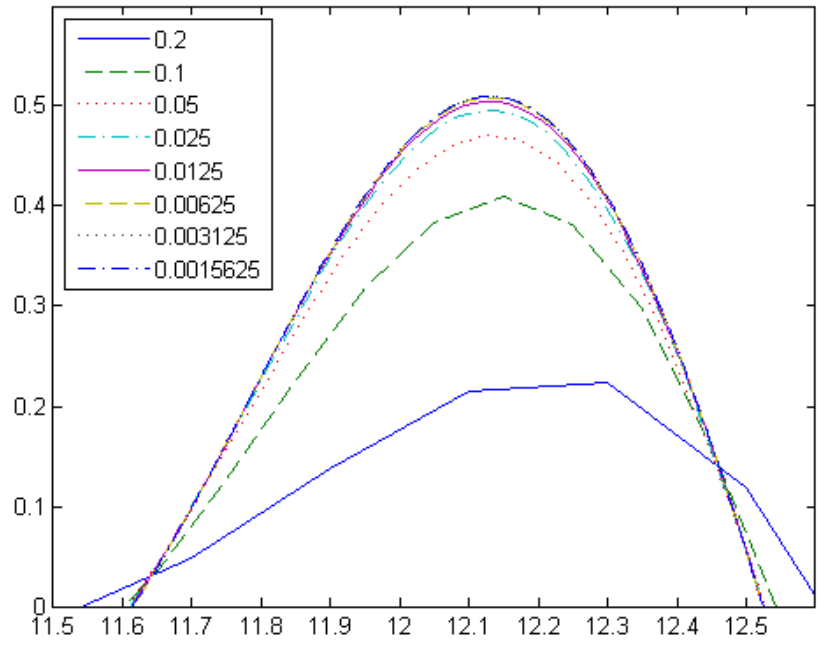
(a)


0.2
0.1
0.05
0.025
0.0125
0.00625
0.003125
0.0015625
0.5
0.4
0.3
0.2
0.1
0
11.5
11.6
11.7
11.8
11.9
12
12.1
12.2
12.3
12.4
12.5

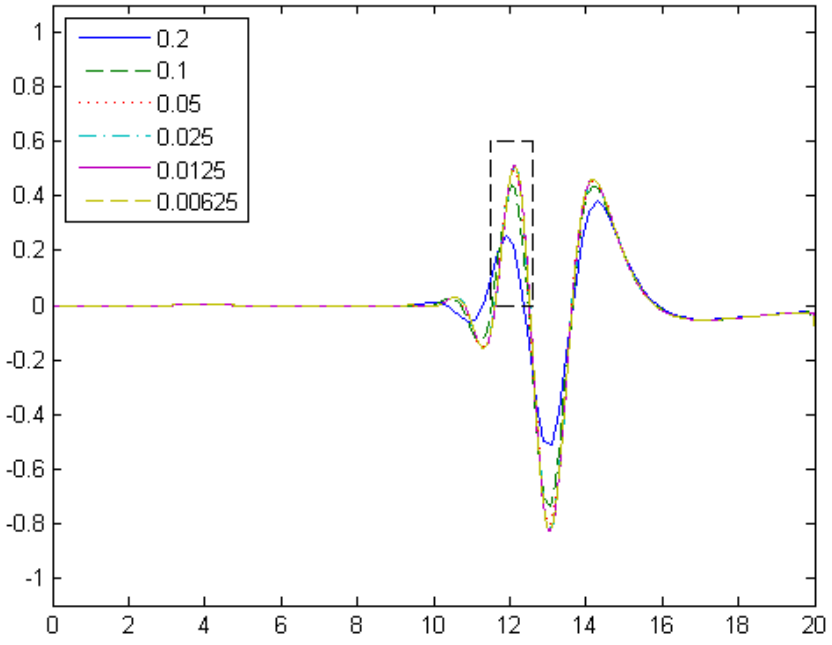
(b)


0.2
0.1
0.05
0.025
0.0125
0.00625
0.003125
0.0015625
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
2
4
6
8
10
12
14
16
18
20

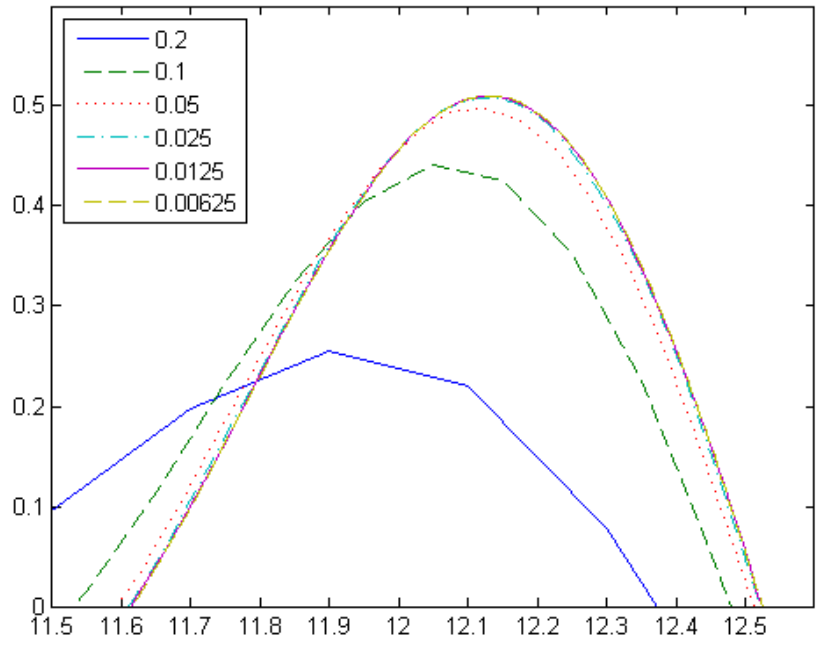
(c)


0.2
0.1
0.05
0.025
0.0125
0.00625
0.003125
0.0015625
0.5
0.4
0.3
0.2
0.1
0
11.5
11.6
11.7
11.8
11.9
12
12.1
12.2
12.3
12.4
12.5

(d)




(e)




(f)

Figure 4.4: Solution $u(x,T)$ to OWWWE with initial data (3.9) for a sequence of grids. (a) and (b) show results for linear stencil 0, (c) and (d) show results for quadratic stencil 0, and (e) and (f) show results for cubic stencil 1, with images on the right displaying the zoomed-in area indicated by the box in the images on the left.

### 4.3.3 Long Time Decay

In Chapter III, we remarked that in our experience, the $O(t^{-1/2})$ decay of solutions to the OWWWE is achievable for a broader class of initial data. In this section we use initial data (3.9) to compute solutions to the OWWWE. Initial data given by (3.9) does not have compactly supported Fourier transform; its Fourier transform is given by

$$\frac{6}{\pi}e^{20\pi i\xi}\frac{(61440i\xi^3(72\xi^2+235)+1947520i\xi)\cos(6\pi\xi)}{(16\xi^2-9)(16\xi^2-1)(144\xi^2-121)(144\xi^2-49)(144\xi^2-25)(144\xi^2-1)}$$

Hence this initial data does not satisfy the conditions of Proposition III.2. We compute the solution on the domain $[0, 3L]$ until time $T = 100$. For $n = 1$ to 100, we compute $\max_x |u(x, n)|$. Figure 4.5 shows $n$ plotted versus $\max_x |u(x, n)|$. Fitting a straight line through the data $\log(n)$ versus $\log(\max_x |u(x, n)|)$ gives slope $-0.4858$; in other words, $\max_x |u(x, t)| = O(t^{-0.4858})$.

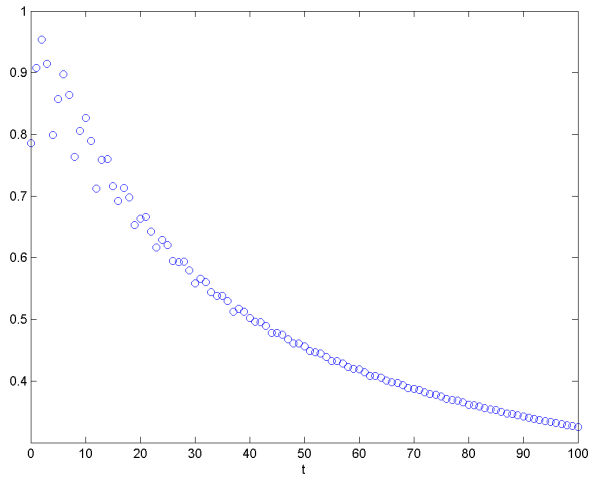



Figure 4.5: Plot of $t$ versus $\max_x |u(x, t)|$, where $u(x, t)$ is the solution to the OWWWE with initial data (3.9)

### 4.3.4 Conservation of Energy

In section 3.2.5 we showed that the solution $u(x,t)$ to the OWWWE satisfies

$$\| u(\cdot,t) \|_{L^2} = \| u(\cdot,0) \|_{L^2}$$

Figure 4.6 illustrates the conservation of energy achieved by our numerical scheme. We compute solutions to the right-going OWWWE using initial data (3.9). As expected, we see that the total energy (as computed using the $l^2$ norm) stays approximately constant until enough of the waves reach the edge of the computational domain, at which point the total energy in the system decreases.

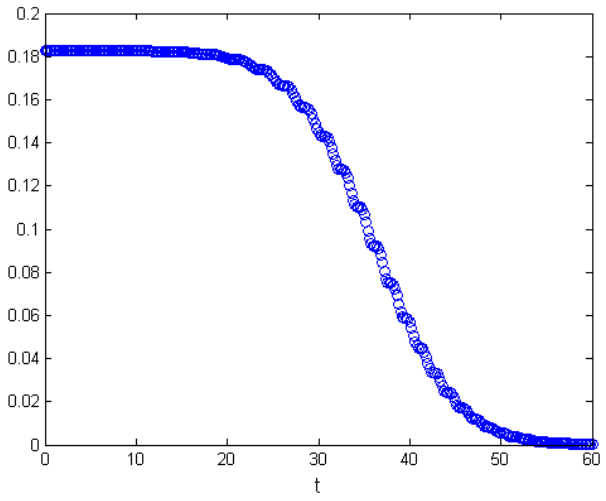



Figure 4.6: Plot of $\| u(\cdot,t) \|_{l^2}$ versus $t$ for the solution $u(x,t)$ to the right-going OWWWE with initial data (3.9)

| $r$ | $k$ | $l=1$ | $l=2$ | $l=3$ | $l=4$ | |
|---|---|---|---|---|---|---|
| 2 | 0 | $\frac{1}{\Delta x}(V_{j+1/2}-V_{j-1/2})$ | $\frac{1}{2\Delta x^2}(V_{j+1/2}-2V_{j-1/2}+V_{j-3/2})$ | | | |
| | 1 | $\frac{1}{\Delta x}(V_{j+1/2}-V_{j-1/2})$ | $\frac{1}{2\Delta x^2}(V_{j+3/2}-2V_{j+1/2}+V_{j-1/2})$ | | | |
| 3 | 0 | $\frac{1}{24\Delta x}(V_{j-5/2}-3V_{j-3/2}-21V_{j-1/2}+23V_{j+1/2})$ | $\frac{1}{4\Delta x^2}(-V_{j-5/2}+5V_{j-3/2}-7V_{j-1/2}+3V_{j+1/2})$ | $\frac{1}{6\Delta x^3}(-V_{j-5/2}+3V_{j-3/2}-3V_{j-1/2}+V_{j+1/2})$ | | |
| | 1 | $\frac{1}{24\Delta x}(V_{j-3/2}-27V_{j-1/2}+27V_{j+1/2}-V_{j+3/2})$ | $\frac{1}{4\Delta x^2}(V_{j-3/2}-V_{j-1/2}-V_{j+1/2}+V_{j+3/2})$ | $\frac{1}{6\Delta x^3}(-V_{j-3/2}+3V_{j-1/2}-3V_{j+1/2}+V_{j+3/2})$ | | |
| | 2 | $\frac{1}{24\Delta x}(-23V_{j-1/2}+21V_{j+1/2}+3V_{j+3/2}-V_{j+5/2})$ | $\frac{1}{4\Delta x^2}(3V_{j-1/2}-7V_{j+1/2}+5V_{j+3/2}-V_{j+5/2})$ | $\frac{1}{6\Delta x^3}(-V_{j-1/2}+3V_{j+1/2}-3V_{j+3/2}+V_{j+5/2})$ | | |
| 4 | 1 | $\frac{1}{24\Delta x}(V_{j-3/2}-27V_{j-1/2}+27V_{j+1/2}-V_{j+3/2})$ | $\frac{1}{48\Delta x^2}(-5V_{j-5/2}+32V_{j-3/2}-42V_{j-1/2}+8V_{j+1/2}+7V_{j+3/2})$ | $\frac{1}{6\Delta x^3}(-V_{j-3/2}+3V_{j-1/2}-3V_{j+1/2}+V_{j+3/2})$ | $\frac{1}{24\Delta x^4}(V_{j-5/2}-4V_{j-3/2}+6V_{j-1/2}-4V_{j+1/2}+V_{j+3/2})$ | |
| | 2 | $\frac{1}{24\Delta x}(V_{j-3/2}-27V_{j-1/2}+27V_{j+1/2}-V_{j+3/2})$ | $\frac{1}{48\Delta x^2}(7V_{j-3/2}+8V_{j-1/2}-42V_{j+1/2}+32V_{j+3/2}-5V_{j+5/2})$ | $\frac{1}{6\Delta x^3}(-V_{j-3/2}+3V_{j-1/2}-3V_{j+1/2}+V_{j+3/2})$ | $\frac{1}{24\Delta x^4}(V_{j-3/2}-4V_{j-1/2}+6V_{j+1/2}-4V_{j+3/2}+V_{j+5/2})$ | |

Table 4.2: Coefficients $c_l$ of the local polynomial interpolant $\mathcal{P}_j(x)$ for various orders $r$ and stencils $k$

| $l$ | $w[l][j]$ |
|---|---|
| 0 | $2\left(\sqrt{x_{j+1/2}} - \sqrt{x_{j-1/2}}\right)$ |
| 1 | $\sqrt{x_{j+1/2}}\left(-2x_j + \frac{2}{3}x_{j+1/2}\right) - \sqrt{x_{j-1/2}}\left(-2x_j + \frac{2}{3}x_{j-1/2}\right)$ |
| 2 | $\sqrt{x_{j+1/2}}\left(2x_j^2 - \frac{4}{3}x_j x_{j+1/2} + \frac{2}{5}x_{j+1/2}^2\right) - \sqrt{x_{j-1/2}}\left(2x_j^2 - \frac{4}{3}x_j x_{j-1/2} + \frac{2}{5}x_{j-1/2}^2\right)$ |
| 3 | $\sqrt{x_{j+1/2}}\left(-2x_j^3 + 2x_j^2 x_{j+1/2} - \frac{6}{5}x_j x_{j+1/2}^2 + \frac{2}{7}x_{j+1/2}^3\right) - \sqrt{x_{j-1/2}}\left(-2x_j^3 + 2x_j^2 x_{j-1/2} - \frac{6}{5}x_j x_{j-1/2}^2 + \frac{2}{7}x_{j-1/2}^3\right)$ |

Table 4.3: Table of integral values $w[l][j] = \int_{I_j} \frac{(y-x_j)^l}{\sqrt{y}} dy$ for fixed $j$

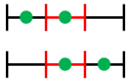
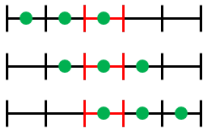
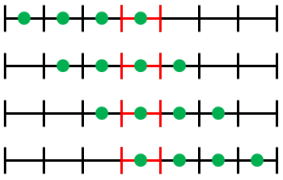
| Polynomial Order | Stencil | Stability Bound |
|---|---|---|
| Linear |  | $0 \leq \nu \leq 0.853$<br>$-0.853 \leq \nu \leq 0$ |
| Quadratic |  | $0 \leq \nu \leq 1.213$<br>$-1.573 \leq \nu \leq 1.573$<br>$-1.213 \leq \nu \leq 0$ |
| Cubic |  | Unstable<br>$0 \leq \nu \leq 2.426$<br>$-2.426 \leq \nu \leq 0$<br>Unstable |

Table 4.4: Stability bounds for $u_t = -\Delta x^{-1/2} A u$ for polynomial reconstruction of various orders/stencils, $L = 20$. Negative values of $\nu$ correspond to stable conditions for the left-going problem.

| $h$ | $(r,k) = (2,0)$ | $(r,k) = (3,0)$ | $(r,k) = (3,1)$ | $(r,k) = (4,1)$ |
|---|---|---|---|---|
| 0.2 | 1.0306311E-004 | 2.1207875E-006 | 1.2183324E-006 | 1.8382884E-008 |
| 0.1 | 2.5766663E-005 | 2.6598678E-007 | 1.4884834E-007 | 1.1485795E-009 |
| 0.05 | 6.4427359E-006 | 3.3302938E-008 | 1.8386116E-008 | 7.1784198E-011 |
| 0.025 | 1.6104354E-006 | 4.1662530E-009 | 2.2843759E-009 | 4.4870189E-012 |
| 0.0125 | 4.0260910E-007 | 5.2099415E-010 | 2.8467459E-010 | 2.9758835E-013 |

Table 4.5: Example 1: $L^2$ error in approximating $u$ by a polynomial for various $(r,k)$ pairs

| $r$ | $k$ | $M$ | $p$ |
|---|---|---|---|
| 2 | 0 | 2.5765486E-003 | 1.99998481 |
| 3 | 0 | 2.6447403E-004 | 2.99785508 |
| 3 | 1 | 1.5493713E-004 | 3.01525123 |
| 4 | 1 | 1.1044352E-005 | 3.98292489 |

Table 4.6: Example 1: Values of $M_{poly}$ and $p_{poly}$ for various $(r,k)$ pairs

| $h$ | $(r,k) = (2,0)$ | $(r,k) = (3,0)$ | $(r,k) = (3,1)$ | $(r,k) = (4,1)$ |
|---|---|---|---|---|
| 0.2 | 7.1845386E-006 | 2.8375790E-007 | 1.4729194E-007 | 9.2275084E-010 |
| 0.1 | 1.2494326E-006 | 2.3370712E-008 | 9.2602698E-009 | 3.8217108E-011 |
| 0.05 | 2.1919850E-007 | 1.9830778E-009 | 5.8095211E-010 | 1.6502970E-012 |
| 0.025 | 3.8610608E-008 | 1.7142037E-010 | 3.6413100E-011 | 6.0527523E-014 |
| 0.0125 | 6.8137882E-009 | 1.4981113E-011 | 2.2825213E-012 | 5.4548118E-014 |

Table 4.7: Example 1: $L^2$ error in computing SSR transform for various $(r,k)$ pairs

| $r$ | $k$ | $M$ | $p$ |
|---|---|---|---|
| 2 | 0 | 4.9554072E-004 | 2.51302264 |
| 3 | 0 | 9.4795811E-005 | 3.56376074 |
| 3 | 1 | 9.7363794E-005 | 3.23956022 |
| 4 | 1 | 1.6694854E-006 | 4.62216077 |

Table 4.8: Example 1: Values of $M_{trans}$ and $p_{trans}$ for various $(r, k)$ pairs

| $h$ | $(r,k) = (2,0)$ | $(r,k) = (3,0)$ | $(r,k) = (3,1)$ | $(r,k) = (4,1)$ |
|---|---|---|---|---|
| 0.2 | 9.7861759E-003 | 4.6486669E-003 | 2.5619507E-003 | 1.0072027E-003 |
| 0.1 | 2.5027162E-003 | 6.1024771E-004 | 3.3328856E-004 | 6.7314224E-005 |
| 0.05 | 6.2925615E-004 | 7.7225958E-005 | 4.2081885E-005 | 4.2788227E-006 |
| 0.025 | 1.5753866E-004 | 9.6830454E-006 | 5.2734915E-006 | 2.6856066E-007 |
| 0.0125 | 3.9398720E-005 | 1.2113141E-006 | 6.5960151E-007 | 1.6802820E-008 |

Table 4.9: Example 2: $L^2$ error in approximating $u$ by a polynomial for various $(r, k)$ pairs

| $r$ | $k$ | $M_{poly}$ | $p_{poly}$ |
|---|---|---|---|
| 2 | 0 | 0.24296019 | 1.99026222 |
| 3 | 0 | 0.57242920 | 2.97898471 |
| 3 | 1 | 0.31634598 | 2.98285903 |
| 4 | 1 | 0.61667707 | 3.97121020 |

Table 4.10: Example 2: Values of $M_{poly}$ and $p_{poly}$ for various $(r, k)$ pairs

| $h$ | $(r,k) = (2,0)$ | $(r,k) = (3,0)$ | $(r,k) = (3,1)$ | $(r,k) = (4,1)$ |
|---|---|---|---|---|
| 0.2 | 1.4987151E-003 | 8.8119181E-004 | 4.0768937E-004 | 1.7892586E-004 |
| 0.1 | 1.9832740E-004 | 6.4794713E-005 | 2.4393837E-005 | 5.7249801E-006 |
| 0.05 | 2.8165944E-005 | 5.0364005E-006 | 1.4266410E-006 | 1.8309317E-007 |
| 0.025 | 4.3502440E-006 | 4.1513944E-007 | 8.4103144E-008 | 6.2620009E-009 |
| 0.0125 | 7.1494763E-007 | 3.5459857E-008 | 5.0220311E-009 | 2.3283230E-010 |

Table 4.11: Example 2: $L^2$ error in computing SSR transform for various $(r, k)$ pairs

| $r$ | $k$ | $M_{trans}$ | $p_{trans}$ |
|---|---|---|---|
| 2 | 0 | 0.15495930 | 2.81011027 |
| 3 | 0 | 0.34909540 | 3.68403443 |
| 3 | 1 | 0.31086497 | 4.08248914 |
| 4 | 1 | 0.51569524 | 4.93737570 |

Table 4.12: Example 2: Values of $M_{trans}$ and $p_{trans}$ for various $(r, k)$ pairs

# CHAPTER V

## Absorbing Boundary Condition

In Chapter III we introduced the one-way water wave equations

$$u_t \pm H|D|^{1/2}u = 0 \tag{5.1}$$

$$u(x,0) = g(x) \tag{5.2}$$

We presented a hierarchy of high-order numerical schemes to approximate (5.1)-(5.2) in Chapter IV. These equations represent one-sided versions of the full water wave equation,

$$u_{tt} + |D|u = 0 \tag{5.3}$$

$$u(x,0) = u_0(x) \tag{5.4}$$

$$u_t(x,0) = u_1(x) \tag{5.5}$$

and we showed in Chapter III that the solution to (5.3)-(5.5) can be recovered by solving

$$v_t + H|D|^{1/2}v = 0$$

$$v(x,0) = \frac{1}{2}\left(u_0(x) + H|D|^{-1/2}u_1(x)\right)$$

and

$$w_t - H|D|^{1/2}w = 0$$
$$w(x,0) = \frac{1}{2}\left(u_0(x) - H|D|^{-1/2}u_1(x)\right)$$

and setting

$$u(x,t) = v(x,t) + w(x,t)$$

In this chapter, we describe the implementation of the one-way wave equations in layers near the boundary to propagate the one-way components of the full wave equation. At the differential equation level, the matching is perfect; outgoing waves are completely transmitted. Once the equations are discretized, reflections can arise from discrete imperfections in the matching, but we show that the matching is improved by using higher order techniques for the one-way wave equations. Damping may also lead to some non-physical reflections at the discrete level, but we have seen that damping is not necessary, since as the wave reaches the edge of the computational domain, the one-sidedness of the OWWWEs does not allow the wave to be reflected (despite the Dirichlet-like assumption that $u \equiv 0$ outside of the computational domain) and minimizes the effect of errors at the boundary on the interior solution.

We begin by describing the damping that is used, and then give the numerical setup for the absorbing boundary layers. Finally, numerical results are presented.

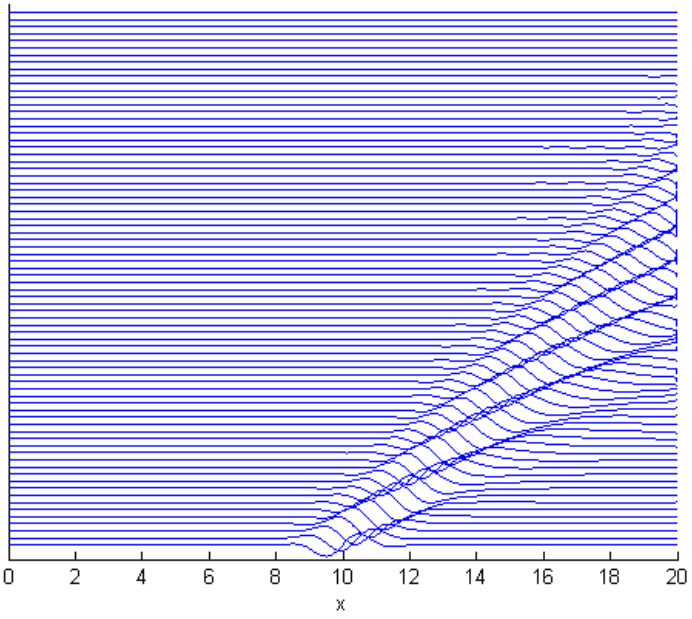
## 5.1 Damped One Way Water Wave Equations

The decay properties of the OWWWEs, as given in Proposition III.2 and Proposition III.1, indicate that moving the artificial boundary further away from the region of interest makes the waves weaker as they reach the boundary, and thus easier to

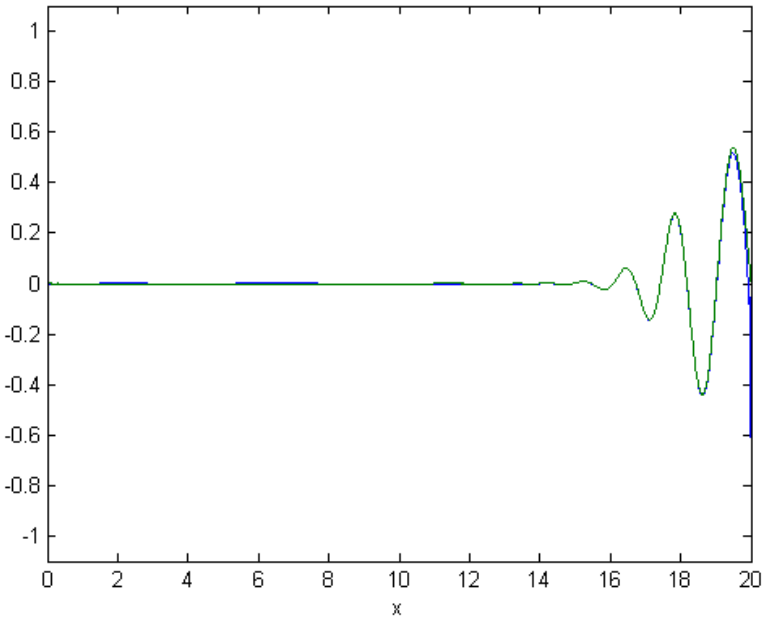
absorb. Of course, this is impractical as a numerical technique, because we would like to limit the size of our computational domain as much as possible.

The decay of solutions to the OWWWE is rather slow, so that the solution is still of significant amplitude when it reaches the boundary. Since the solution is taken to be zero outside the computational domain, boundary errors are inevitable. Our experience is that the one-way character of the equation prevents errors that may be generated at the artificial boundary from propagating back and polluting the solution at the interior of the domain, so that extending the computational domain to take advantage of the decay of solutions is not necessary. In computations, the one way solution on the truncated domain $[0, L]$ still does a good job of matching the true solution away from the boundary $x = L$. Figure 5.1 shows the time evolution of the one way solution solved on $[0, L]$ and a comparison to the one way solution computed on the larger domain $[0, 2L]$. The wave moves to the right, and as it reaches the boundary it does not reflect, because the one way equation does not provide a mechanism for waves to move to the left despite the presence of the nonlocal transform. The waves seem to "stick" a little bit to the boundary before ultimately passing through. Compare this to the results for the full equation near the boundary from figure 3.2, when the waves were fully reflected back into the computational domain. Because of this, damping may not be necessary in the present 1D context, but we continue to consider it as it may prove to be more important in 2D water wave problems.

It is interesting to compare damping strategies for the full WWE and for the OWWWE. For the WWE, following [9] we add a linear damping term $-\nu\vec{v}$ to the




(a)




(b)

Figure 5.1: Solution to OWWWE with initial data (3.9). (a) shows the solution with increasing time on the vertical axis; note that as the wave reaches the edge of the computational domain it does not reflect. (b) shows the solution at time $T$ computed on the domain $[0, L]$ plotted together with the solution at time $T$ as computed on the extended domain $[0, 2L]$; note how well the solutions match.

momentum equations

$$\vec{v}_t + (\vec{v} \cdot \nabla)\, \vec{v} = -(0,1) - \nabla p - \nu \vec{v}$$

so that equation (2.10) becomes

$$z_{tt} + i = iaz_\alpha - \nu z_t$$

Following the derivation from Chapter II of the WWE, we take another derivative with respect to $t$, compose the equation with $h^{-1}$, and take the real part to find that the additional term takes the form

$$-\nu X_{tt} = -\nu w = -\nu(u_t + bu_\alpha)$$

which reduces to the additional term $-\nu u_t$ when linearized about the zero solution. Thus, the damped WWE (DWWE) is given by

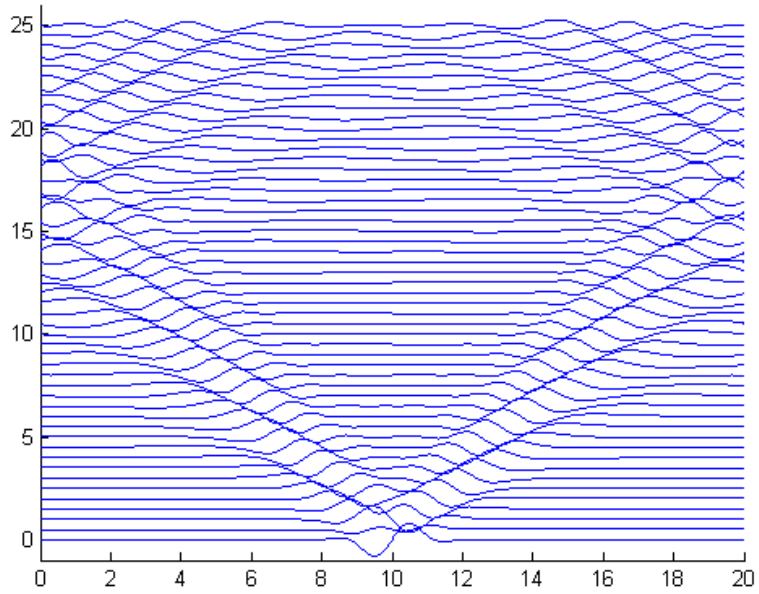
$$u_{tt} + |D|u = -\nu u_t \tag{5.6}$$

Figure 5.2a shows the solution to the DWWE from equation (5.6). In addition, we implemented exponential damping by reasoning that $v(x,t) = e^{-d(x-x_0)}u(x,t)$, where $u(x,t)$ satisfies the undamped WWE, satisfies

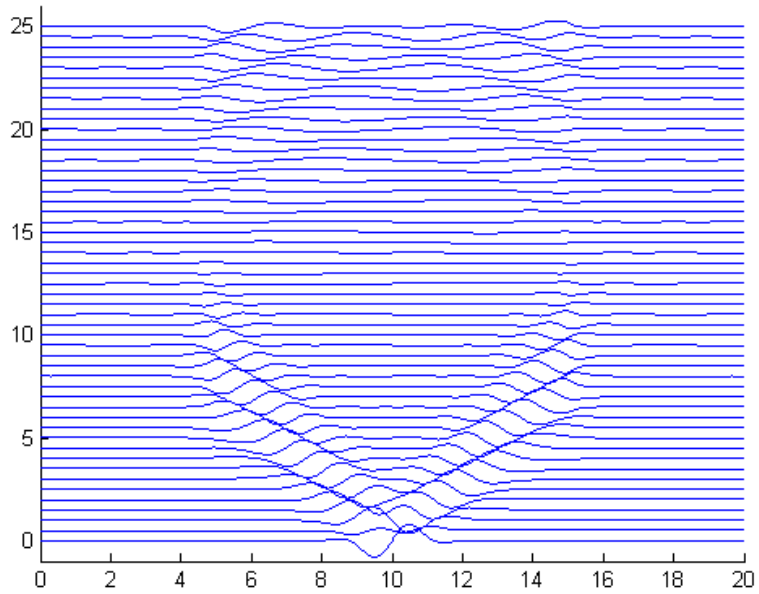
$$v_{tt} + e^{-d(x-x_0)}|D|\left(e^{d(x-x_0)}v\right) = 0 \tag{5.7}$$

Figure 5.2b shows the solution to the DWWE from equation (5.7).

The results from the DWWE are not impressive; ultimately, reflections are clearly visible not from the boundary itself but due to the introduction of the damping terms, making the solution far from nonzero as expected in the central region. This is further validation that our choice to impose the OWWWEs in layers near the boundary is beneficial, because as we have seen, the OWWWEs are effective at




(a)




(b)

Figure 5.2: Solution of DWWE with increasing time on the vertical axis, computed with initial data (3.9)-(3.10). (a) uses equation (5.6) (additive damping) and (b) uses equation (5.7) (exponential damping).

carrying waves away from the central domain without allowing waves to reflect at the computational boundary. It is still possible and not difficult to implement damping

for the OWWWEs as well, and to that end we consider two different techniques to introduce additional damping to the OWWWE (similarly to what was done for the WWE). The first is an additive damping,

$$u_t \pm \left(H|D|^{1/2} - r(x)\right) u = 0 \tag{5.8}$$

where the function $r(x)$ is a (possibly regularized) Heaviside function that controls the rate of damping. The second type is multiplicative damping, where again we would like to solve for the function $u(x,t) = e^{-d(x-x_0)}v(x,t)$ instead of solving for the usual (undamped) solution $v$ to the OWWWE. The function $u$ satisfies

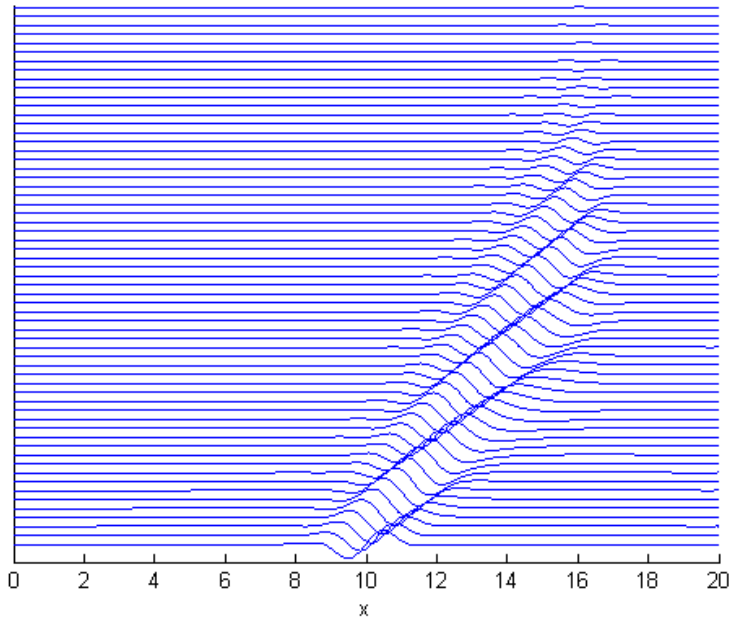
$$u_t \pm e^{-d(x-x_0)}(H|D|^{1/2}(e^{d(x-x_0)}u)) = 0 \tag{5.9}$$

where $d > 0$ for right-moving waves and $d < 0$ for left moving waves in the damping layer. Exponential damping analytically gives no reflections, but reflections may be generated at the discrete level on the order of the numerical error.

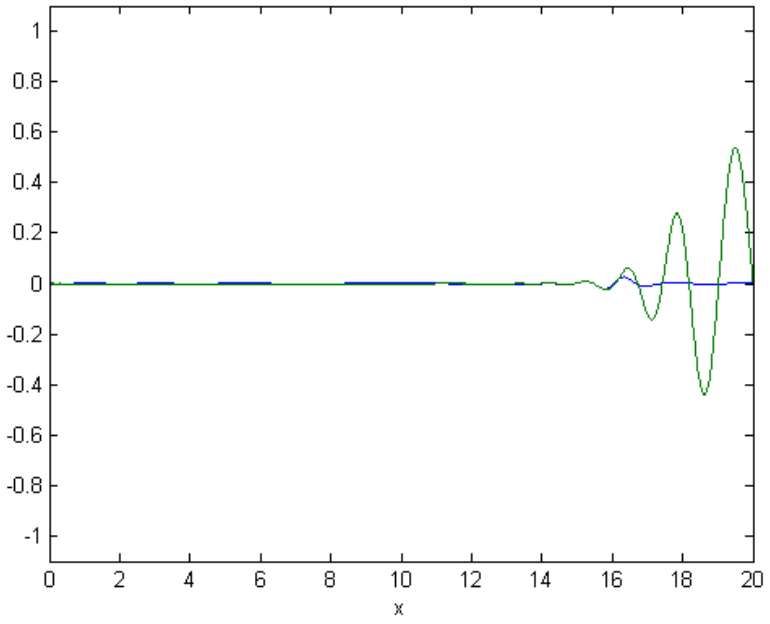
Figures 5.3a and 5.4a show the one way solution with damping in the region $x \geq 15$, and figures 5.3b and 5.4b show the damped solution plotted with the one way solution without damping. For the examples shown, the multiplicative damping is stronger as indicated by the reduced magnitude of the solution in the damping layer as compared to the additive damping, but the strength of the damping can be adjusted by changing $d$ or $\max_x |r(x)|$. Both do a good job of matching the undamped solution outside of the damping layer.

## 5.2 Absorbing Boundary Layer

In this section we describe an algorithm for solving the full water wave equation by imposing the one-way water wave equations in a layer near the boundary. Our goal is to allow outgoing waves to leave the domain without polluting the interior



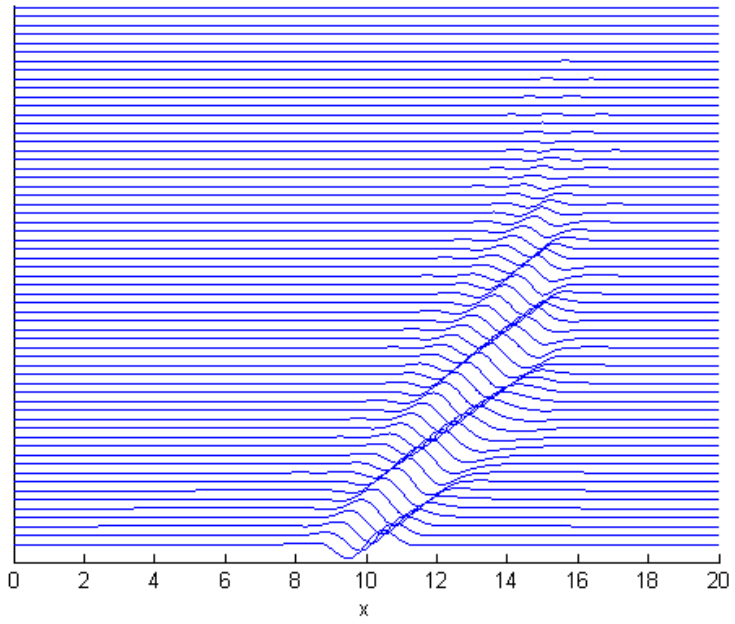

(a)



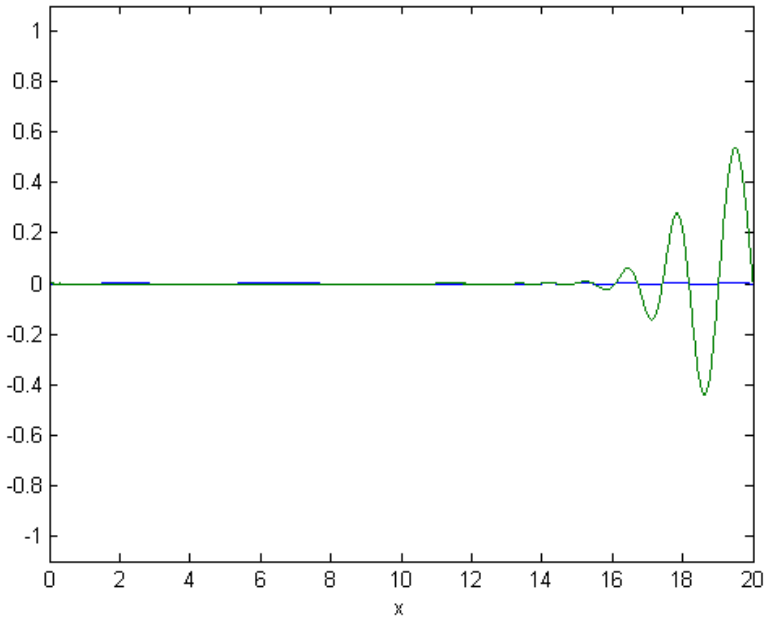

(b)

Figure 5.3: Solution to OWWWE with additive damping, with initial data (3.9). (a) shows the solution with increasing time on the vertical axis. (b) shows the damped solution at time $T$ plotted together with the undamped solution at time $T$; note how well the solutions match.

full solution. Let the matched solution be written as $\widetilde{u}$. Using the relationship between the solution to the full equation and the solutions to the one way water




(a)




(b)

Figure 5.4: Solution to OWWWE with exponential damping, with initial data (3.9). (a) shows the solution with increasing time on the vertical axis. (b) shows the damped solution at time $T$ plotted together with the undamped solution at time $T$; note how well the solutions match.

wave equations, we solve the full equation

$$u_{tt} + |D|u = 0$$

$$u(x,0) = u_0(x)$$

$$u_t(x,0) = u_1(x),$$

the right going equation

$$v_t - H|D|^{1/2}v = 0$$
$$v(x,0) = \frac{1}{2}\left(u_0(x) - H|D|^{-1/2}u_1(x)\right),$$

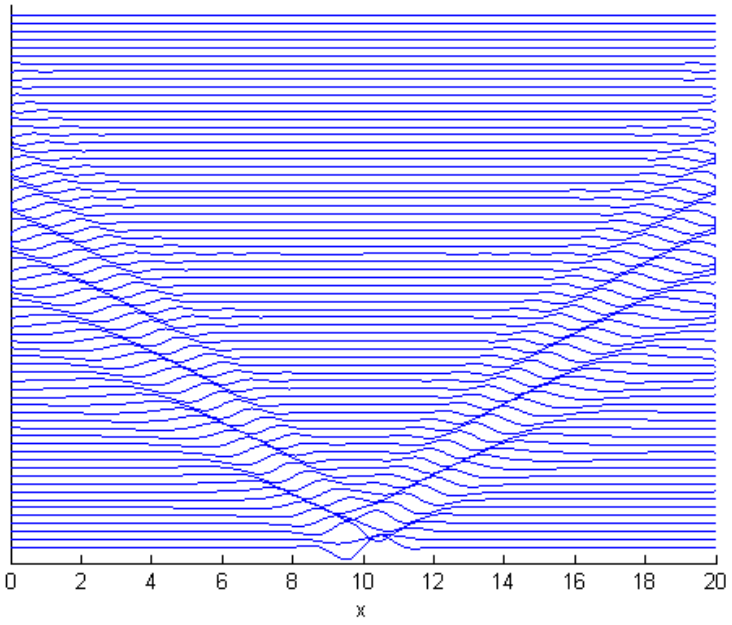
and the left going equation

$$w_t + H|D|^{1/2}w = 0$$
$$w(x,0) = \frac{1}{2}\left(u_0(x) + H|D|^{-1/2}u_1(x)\right)$$

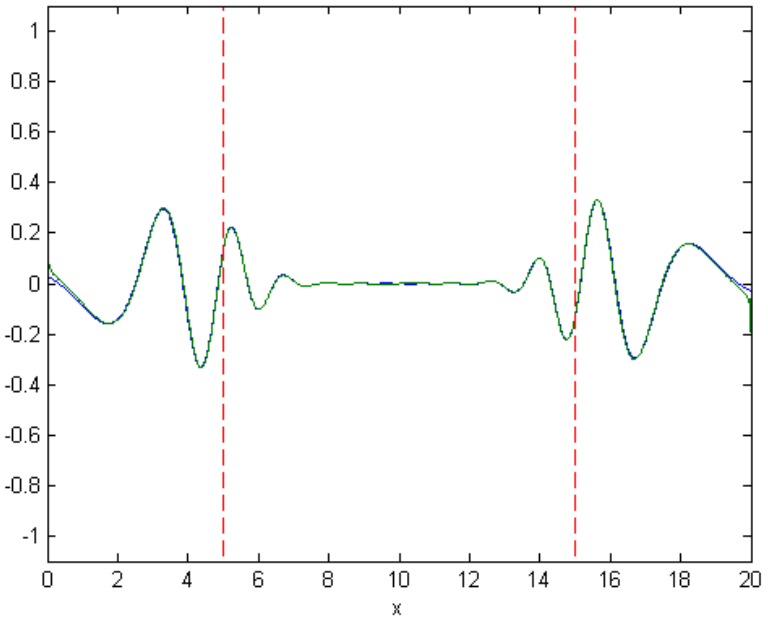
Then we set

$$\widetilde{u}(x,t) = \begin{cases} u(x,t) & : c_1 L \le x \le c_2 L \\ v(x,t) & : x > c_2 L \\ w(x,t) & : x < c_1 L \end{cases} \tag{5.10}$$

Figures 5.5a and 5.6a show the time evolution of the matched solution with and without damping with initial data (3.9)-(3.10). Figures 5.5b and 5.6b show the matched solutions plotted together with the full solutions solved on an extended domain, to show that the matched solutions do a good job of matching the full solution in the interior domain.

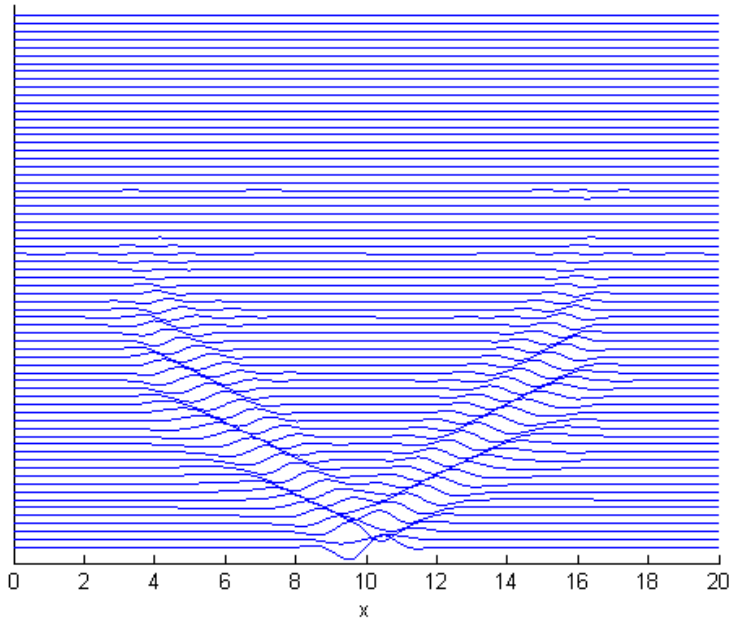



(a)

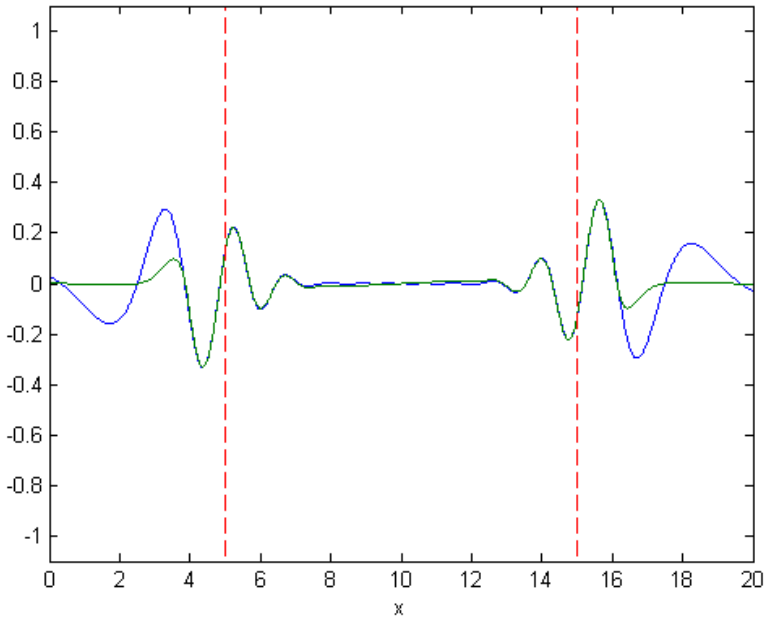



(b)

Figure 5.5: Solution of full WWE matched with one ways in layers near the boundary, with initial data (3.9)-(3.10). (a) shows the solution with increasing time on the vertical axis. (b) shows the solution at time $T$ plotted together with the solution to the WWE at time $T$; note how well the solutions match.




(a)




(b)

Figure 5.6: Solution of full WWE matched with damped one ways in layers near the boundary, with initial data (3.9)-(3.10). (a) shows the solution with increasing time on the vertical axis. (b) shows the solution at time $T$ plotted together with the solution to the WWE at time $T$; note how well the solutions match.

# CHAPTER VI

# Conclusions and Future Work

## 6.1 Conclusions

In this work we derived and implemented numerically a novel absorbing boundary for the Water Wave Equation (WWE), which describes linearized two-dimensional incompressible, irrotational, inviscid free surface flow in deep water. The equation is nonlocal and involves the Hilbert transform. In numerical simulations, the domain is truncated to a finite size and the solution is not available outside of the computational domain. Cutting off the integral transform at the edge of the computational domain (which is equivalent to assuming the solution vanishes outside the domain) causes outgoing waves to reflect back into the domain. Commonly used artificial boundary conditions include the so-called "absorbing beach" in which outgoing waves are damped. Damping the WWE directly is not very effective, as non-physical waves of nontrivial size return to the region of interest in finite time.

We derive a one-way version of the WWE (the OWWWE) which supports the propagation of water waves essentially only in one direction. The one-way equation is a fractional partial differential equation involving a nonlocal operator corresponding to half a derivative. The fractional derivative is implemented as a derivative of a convolution with a singular kernel with locally integrable singularity. We derive a

family of numerical methods using an Exact Polynomial Integration Computation (EPIC) which efficiently and accurately approximate solutions to the OWWWEs. The one way equation offers an advantage over the full water wave equation in that errors that are generated do not have a mechanism to propagate back and pollute the solution in the interior of the domain. The OWWWEs, as approximated using the EPIC schemes, very effectively model the behavior of the solution to the OWWWE on the unbounded domain.

We use the OWWWEs to absorb outgoing water waves in absorbing layers near the computational boundary. We solve the full WWE in the interior domain and replace it by the OWWWE in a narrow layer near the outer boundary. The matching of the two equations is reflection free. Damping may or may not be added within the absorbing layer to further dissipate the waves before they reach the edge of the computational domain. We show that this technique is very effective at allowing the waves to leave the domain.

## 6.2 Future Work

The work in this thesis may be generalized in several directions. The numerical algorithm to approximate solutions to the OWWWEs may be improved by adding the ability to use ENO-type adaptive stencil selection [25] or WENO techniques [32] to better handle propagation of waves with sharp fronts. A more robust implementation of the EPIC schemes for higher order polynomials may be derived, again taking inspiration from the vast literature on polynomial interpolating schemes from the hyperbolic conservation law literature; currently, the exact polynomial integration is very sensitive to roundoff errors and may lead to instability when high order methods are implemented on fine grids. The schemes may be generalized to non-uniform grids,

which will allow for the algorithms to be implemented on grids that are stretched near the boundary.

On the analytic side, a major generalization is to derive a OWWWE for three dimensional problems. This can be accomplished as a first attempt by using two-dimensional ideas in the direction normal to the boundary, and more generally by following ideas used by Engquist and Majda [16]. In both two- and three-dimensional problems, the usefulness of OWWWEs as boundary conditions rather than in an absorbing layer can be investigated.

# BIBLIOGRAPHY

# BIBLIOGRAPHY


[1] Almendral, A. and C.W. Oosterlee. *Accurate evaluation of European and American options under the CGMY process*, SIAM Journal on Scientific Computing (2007), pp. 93–117.

[2] Azerad, P., A. Bouharguane, and J.-F. Crouzet. *Simultaneous denoising and enhancement of signals by a fractal conservation law*, arXiv:1004.5193v1 [math.AP] 29 April 2010.

[3] Baker, G.R., D.I. Meiron, and S.A. Orszag. *Generalized vortex methods for free-surface flow problems*, The Journal of Fluid Mechanics **123** (1982), pp. 477–501.

[4] Betancourt, F., R. Burger, K.H. Karlsen, and E.M. Tory. *On nonlocal conservation laws modelling sedimentation*, Nonlinearity. **24** (2011), pp. 855–885.

[5] Betts, P.L. and T.T. Mohamad. in Proc., 4th Int. Symp. Finite Element Methods in Flow Problems, T. Kawai (ed.), Tokyo (1982), pp. 923.

[6] Briani, M., C. La Chioma, and R. Natalini. *Convergence of numerical schemes for viscosity solutions to integro-differential degenerate parabolic problems arising in financial theory*, Numerische Mathematik **98** (2004), pp. 607–646.

[7] Briani, M. and R. Natalini. *Asymptotic high-order schemes for integro-differential problems arising in markets with jumps*, Communications in Mathematical Sciences **4** (2006), pp. 81–96.

[8] Chae, D., A. Córdoba, D. Córdoba, and M.A. Fontelos. *Finite time singularities in a 1D model of the quasi-geostrophic equation*, Advances in Mathematics **194** (2005), pp. 203–223.

[9] Chan, R. K.-C. *Two-dimensional time-dependent calculations of large-amplitude surface gravity waves due to a surface disturbance* in Proceedings of the First International Conference on Numerical Ship Hydrodynamics, J.W. Schot et al (eds.), Bethesda, MD (1975), pp. 315–331.

[10] Choi, H.W., S.K. Chung, and Y.J. Lee. *Numerical solutions for space fractional dispersion equations with nonlinear source terms*, Bulletin of the Korean Mathematical Society **47** (2010), pp. 1225–1234.

[11] Ciesielski, M. and J. Leszczynski. *Numerical treatement of an initial-boundary value problem for fractional partial differential equations*, Signal Processing **86** (2006), pp. 2619–2631.

[12] Clement, A. *Coupling of two absorbing boundary conditions for 2D time-domain simulations of free surface gravity waves*, Journal of Computational Physics **126** (1996), pp.139–151.

[13] Cont, R. and E. Voltchkova. *A finite difference scheme for option pricing in jump diffusion and exponential Lévy models*, SIAM Journal on Numerical Analysis **4** (2005), pp. 1596–1626.

[14] Ding, Z., A. Xiao, and M. Li. *Weighted finite difference methods for a class of space fractional partial differential equations with variable coefficients*, Journal of Computational and Applied Mathematics **233** (2010), pp. 1905–1914.

[15] D'Halluin, Y., P.A. Forsyth, and K.R. Vetzal. *Robust Numerical Methods for contingent claims under jump diffusion processes*, IMA Journal of Numerical Analysis **25** (2005), pp. 87–112.

[16] Engquist, B. and A. Majda. *Absorbing Boundary Conditions for the Numerical Simulation of Waves*, Mathematics of Computation **31** (1977), pp. 629-651

[17] Engquist, B. and A. Majda. *Radiation Boundary Conditions for Acoustic and Elastic Wave Calculations*, Communications on Pure and Applied Mathematics **32** (1979), pp. 313–357.

[18] Erdélyi, A. *Asymptotic representations of Fourier integrals and the method of stationary phase.* Journal of the Society for Industrial and Applied Mathematics **3** (1955), pp. 17–27.

[19] Ertekin, R.C. *Soliton generation by moving disturbances in shallow waters: theory, computation, and experiment*, Ph.D. Dissertation, University of California at Berkeley, 1984.

[20] Givoli, D. Numerical Methods for Problems in Infinite Domains, Elsevier Science Publishers B.V., Amsterdam, 1992.

[21] Goloviznin, V.M. and I.A. Korotkin. *Numerical methods for some one-dimensional equations with fractional derivatives*, Differential Equations **42** (2006), pp. 967–973.

[22] Guerber, E., M. Benoit, S.T. Grilli, and C. Buvat. *Modeling of fully nonlinear wave interactions with moving submerged structures*, Proceedings of the Twentieth (2010) International Offshore and Polar Engineering Conference. Beijing, China (2010).

[23] Grosch, C.E. and S.A. Orszag. *Numerical Solution of Problems in Unbounded Regions: Coordinate Transforms*, Journal of Computational Physics **25** (1977), pp. 273–296.

[24] Halpern, L. and L.N. Trefethen. *Wide-angle One-way Wave Equations*, Journal of the Acoustical Society of America **84** (1988), pp. 1397–1404.

[25] Harten, A., B. Engquist, S. Osher, and S. Chakravarthy. *Uniformly High Order Accurate Essentially Non-oscillatory Schemes, III*, Journal of Computational Physics **71** (1987), pp. 231–303.

[26] Herrmann, R. Fractional Calculus: An Introduction for Physicists. World Scientific Publishing Co. Pte. Ltd.: Singapore, 2011.

[27] Higdon, R.L. *Absorbing Boundary Conditions for Difference Approximations to the Multi-Dimensional Wave Equation*, Mathematics of Computation **47** (1986), pp. 437–459.

[28] Israeli, M. and S.A. Orszag. *Approximation of radiation boundary conditions*, Journal of Computational Physics **41** (1981), pp. 115–135.

[29] Kantha, L.H. and C.A. Clayson. Numerical models of oceans and oceanic processes. Academic Press: San Diego, 2000.

[30] Kreiss, H.O., Proceedings of a Symposium at the University of Wisconsin, May 1966. D Greenspan (eds.). Wiley, New York (1966).

[31] Li, C., A. Chen, and J. Ye. *Numerical approaches to fractional calculus and fractional ordinary differential equation*, Journal of Computational Physics **230** (2011), pp. 3362–3368.

[32] Liu, X.-D., S. Osher, and T. Chan. *Weighted essentially non-oscillatory schemes*, Journal of Computational Physics **115** (1994), pp. 200–212.

[33] Lynch, V.E., B.A. Carreras, D. del-Castillo-Negrete, K.M. Ferreira-Mejias, and H.R. Hicks. *Numerical methods for the solution of partial differential equations of fractional order*, Journal of Computational Physics **192** (2003), pp. 406–421.

[34] Matache, A.-M., C. Schwab, and T.P. Wihler. *Fast numerical solution of parabolic integrodifferential equations with applications in finance*, SIAM Journal on Scientific Computing **27** (2005), pp. 369–393.

[35] Meerschaert, M.M., H.-P. Scheffler, and C. Tadjeran, *Finite difference methods for two-dimensional fractional dispersion equation*, Journal of Computational Physics **211** (2006), pp. 249–261.

[36] Momani, S. and Z. Odibat. *Numerical solutions of the space-time fractional advection-dispersion equation*, Numerical Methods for Partial Differential Equations **24** (2008), pp. 1416–1429.

[37] Oldham, K. and J. Spanier. The Fractional Calculus. Academic Press: New York, 1974.

[38] Orlanski, I. *A Simple Boundary Condition for Unbounded Hyperbolic Flows*, Journal of Computational Physics **21** (1976), pp. 251–269.

[39] Ortigueira, M.D. Fractional Calculus for Scientists and Engineers. Springer: Dordrecht, 2011.

[40] J.N. Pandey, The Hilbert Transform of Schwartz Distributions and Applications, John Wiley & Sons, Inc., New York, 1996.

[41] Park, W.-S., C.-B. Yun, and C.-K. Pyun. *Infinite elements for 3-dimensional wave-structure interaction problems*, Engineering Structures **14** (1992), pp. 335–346.

[42] Pearson, R.A. *Consistent boundary conditions for numerical models of systems that admit dispersive waves*, Journal of the Atmospheric Sciences **31** (1974), pp. 1481–1489.

[43] Podlubny, I. Fractional Differential Equations. Academic Press: San Diego, 1999.

[44] Romate, J.E. *Absorbing Boundary Conditions for Free Surface Waves*, Journal of Computational Physics **99** (1992), pp. 135-145.

[45] E.M. Stein and T.S. Murphy, *Harmonic Analysis: Real-Variable Methods, Orthogonality, and Oscillatory Integrals*, Princeton University Press, Princeton, N.J., 1993.

[46] Sommerfeld, A. *Partial Differential Equations in Physics*, Academic Press, New York, 1949.

[47] Trefethen, L.N. and L. Halpern. *Well-Posedness of One-Way Equations and Absorbing Boundary Conditions*, Mathematics of Computation **47** (1986), pp. 421–435.

[48] Wagatha, L. *Approximation of Pseudodifferential Operators in Absorbing Boundary Conditions for Hyperbolic Equations*, Numerische Mathematik **42** (1983), pp. 51–64.

[49] Wu, S. *Well-posedness in Sobolev spaces of the full water wave problem in 2-D*, Inventiones mathematicae **130** (1997), pp. 39–72.

[50] Yang, Q., F. Liu, and I. Turner. *Numerical methods for fractional partial differential equations with Riesz space fractional derivatives*, Applied Mathematical Modelling **34** (2010), pp. 200–218.