The Numerical Calculation of Three-Dimensional Water Waves using a Boundary Integral Method

Thesis by
David J. Haroldsen

In Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy

California Institute of Technology
Pasadena, California

1997

(Defended May 29, 1997)

Acknowledgements

I could not complete this work without pausing to express appreciation to a number of individuals and institutions who have been a great support to me professionally and personally.

First, to my advisor, Professor Dan Meiron, I owe a great debt for his patience, encouragement, suggestions, advice, criticism, questions, and recommendations. I thank him for helping me to learn from my mistakes and for patiently repeating numerous explanations until I understood. I am also appreciative that he was willing to make time available to discuss my progress and questions. I am most fortunate to have associated with an individual of his caliber. I also express appreciation to Professor Thomas Hou for valuable discussions and suggestions concerning the stability of the numerical scheme. I thank the many other professors and instructors at Caltech with whom I have associated for the things which I learned from them.

I also express thanks to the Electrical Engineering Department at Duke University for making available the PMTA source code, which saved considerable time and spared me the additional stress of writing a fast multipole code.

This work would not have been possible without the financial of several groups. From 1992-1995 I was supported by a National Science Foundation Fellowship. Since 1995, I have been supported by NSF Grant DMS-94-7030. I thank especially the Achievement Rewards for College Scientists (ARCS) Foundation for two years of generous financial support (and two very nice lunches at the Beverly Hills Hotel!).

Finally, I extend thanks to a number of individuals on a personal level. I thank my parents for teaching me the value and importance of education. To the many friends which Jae and I have made in Pasadena, I am grateful for the friendship and the kind words of support. I take from Pasadena not only a degree but also fond memories of the rich association with so many wonderful people. I give thanks to my children, who have been my most inspirational instructors in the field of experimental studies of fluid motion and dynamics in (and from) small containers and in the study of wave motion in shallow, finite bodies of water. They have made the last 18 months more hectic and also more fun. They also motivated me to finish

in a timely manner.

Finally, I express my thanks to my wife, Jae, who has taught me more about myself than any one else. I am most appreciative of her devotion, her selflessness, her long suffering, her patience, her encouragement, and her love. To her I give my thanks and my love. Without her support, my work here would have been much more difficult, much less meaningful, and certainly much less enjoyable.

The Numerical Calculation of Three-Dimensional Water Waves using a Boundary Integral Method

by

David J. Haroldsen

In Partial Fulfillment of the
Requirements for the Degree of
Doctor of Philosophy

Abstract

In this work, we consider the numerical calculation of water waves in three dimensions. One well accepted method for studying surface waves is the boundary integral method, which defines the fluid velocities at the interface in terms of integrals over the boundary of the domain in which the problem is posed. There exists a considerable body of work on the numerical study of surface waves in two dimensions. However, until recently the numerical study of surface waves was considered intractable because of the high computational cost of approximating the defining integrals.

We discuss the boundary integral formulation for the three-dimensional water wave problem and present the point vortex approximation to the singular integrals which define the particle velocities. We consider three aspects of the point vortex approximation: accuracy of the approximation, efficient means of computing solutions, and numerical stability of the scheme.

Concerning the accuracy of the point vortex method, we analyze the error associated with the approximation and show that it can be expressed as a series in odd powers of the discretization parameter h. We present quadrature rules which are highly accurate.

The efficient computation of the point vortex approximation is achieved through the use of the fast multipole algorithm, which combines long distance particle inter-

actions into multipole expansions which can be efficiently evaluated. The underlying periodicity of the problem is reduced to a lattice sum which can be rapidly evaluated. We discuss the implementation of the numerical schemes in both serial and parallel computing environments.

The point vortex method is shown to be highly unstable for straightforward discretizations of the surface. We analyze the stability of the method about equilibrium and discuss methods for stabilizing the numerical schemes for both the linear and nonlinear regimes. We present numerical results which show that the method can be effectively stabilized.

In the final chapter, we present numerical results from several calculations of three-dimensional waves using the methods developed in the previous chapters.

Contents

List of Figures

List of Tables

Chapter 1

Introduction and Background

Many different physical problems involve the time evolution of propagating interfaces. In many instances, the motion of the interface can be defined in terms of surface integrals on the boundary of the domain in which the problem is posed. This type of formulation is advantageous for computing solutions numerically because the effective dimension of the problem is reduced. For example, surface waves in three dimensions can be computed in terms of two-dimensional integrals. Thus, boundary integral methods have the potential of being significantly less costly to compute than methods which define the fluid motion in terms of the entire volume of fluid. Additionally, free surface problems often involve quantities which are not continuous at the interface. Boundary integral methods are advantageous in that they avoid the difficulty of computing derivatives of variables which are not continuous across the interface.

One class of propagating interface problems to which boundary integral methods have previously been applied is the study of deep water waves in two dimensions. In one of the earliest of these studies, Longuet-Higgins and Cokelet [17] used a boundary integral approach to study plunging breakers on deep water in two dimensions. Baker, Meiron, and Orszag [1] applied iterative boundary integral methods to the study of two-dimensional water waves in order to improve both the computational storage requirement and the arithmetic operation count. Others have applied boundary integral methods to the study of two-dimensional water waves with sur-

face tension and with finite depth. These include Vinje and Brevig [29], Roberts [23], and New, McIver, and Peregrine [21]. Boundary integral methods have also been used to study multi-fluid interfaces and vortex sheet roll-up. See for example [28] and [27].

The primary difficulty in numerically calculating surfaces waves using boundary integral methods is the calculation of the velocity and velocity potential at the interface. These quantities are defined in terms of singular integrals. In the case of three-dimensional potential flows, the integral which defines the velocity is a singular Biot-Savart integral of the following form for a vortex sheet:

$$\mathbf{u}(\mathbf{x}) = \frac{1}{4\pi} \int_{S'} \boldsymbol{\kappa}(\mathbf{x}') \times \nabla_{\mathbf{x}} G(\mathbf{x}, \mathbf{x}') dS' \tag{1.1}$$

where $\boldsymbol{\kappa} = \hat{\mathbf{n}} \times [\mathbf{u}_1 - \mathbf{u}_2]$ is the vortex sheet strength, $\hat{\mathbf{n}}$ is the unit normal to the interface, $\mathbf{u}_1 - \mathbf{u}_2$ is the jump in interfacial velocity at the vortex sheet ($\mathbf{u}_1$ is the velocity above the interface), and

$$G(\mathbf{x}, \mathbf{x}') = -\frac{1}{|\mathbf{x} - \mathbf{x}'|}. \tag{1.2}$$

One approach to numerically evaluating this singular integral is the point vortex method. The point vortex method as we and others have applied it is the trapezoidal rule for integration, excluding the singular contribution. Goodman, Hou, and Lowengrub [7] showed that for the Euler equations in two dimensions, the point vortex approximation to the Biot-Savart integral converges and the error is a series in powers of h^2 where h is the local mesh size. Lowengrub, Shelley, and Merriman [18] explicitly computed the coefficients of the error series for the point vortex approximation to the two-dimensional Biot-Savart integral for a rectangular grid. In their study, the authors applied the Poisson summation formula generalized to singular functions to determine the coefficients of the series. Hou and Lowengrub [11] also showed convergence of the point vortex method for the Biot-Savart law in the case of the three-dimensional Euler equations. In the case of a vortex sheet in two dimensions, Shelley [25] showed that removing the first-order error term from

the point vortex method gives a spectrally accurate method. Hou, Lowengrub, and Krasny [12] proved convergence of the point vortex method using the desingularization technique that Shelley presented.

In their early work on overturning water waves in two dimensions, Longuet-Higgins and Cokelet [17] observed numerical instabilities. Roberts [23] discussed the existence of sawtooth instabilities in surface wave calculations. More recently, studies of the point vortex method applied to free surface problems in two dimensions have shown that straightforward spatial discretizations and quadrature rules can lead to extremely unstable schemes. By studying the leading order discrete quadrature and derivative operators, Beale, Hou, and Lowengrub [3] showed that some Fourier filtering of surface variables is necessary for the scheme to be stable.

The extension of the two-dimensional water wave problem to three dimensions introduces a number of challenges. In two dimensions, spectrally accurate quadrature rules can be defined to approximate the velocity and velocity potential. However, these quadrature rules cannot be extended to three dimensions. In addition, the approximation of the integrals using the point vortex method requires $O(N^4)$ calculations where N is the number of particles used to discretize the surface in each dimension. Even for relatively small values of N, the number of calculations required makes a straightforward application of the point vortex method impractical. Finally, as with the two-dimensional problem, the three-dimensional water wave problem is susceptible to severe numerical instabilities.

In the present work we consider the numerical calculation of water waves in three dimensions using a boundary integral formulation with the point vortex method used to approximate the singular integrals. In Chapter 2 we present the boundary integral formulation for water waves. In Chapter 3, we present and analyze the point vortex approximation to the singular integrals. We develop high-order quadrature rules for computing the singular Biot-Savart integrals using Richardson extrapolation. In particular, we show that the error in numerically approximating the integrals is a series in odd powers of h where h is the local mesh size. In Chapter 4 we present methods for efficiently computing the singular integrals and provide details on the

numerical implementation of the method. We approximate the point vortex method using a fast multipole algorithm which reduces the operation count for computing the integrals to $O(N^2)$. In Chapter 5 we analyze the stability of the point vortex method near equilibrium and present several methods for stabilizing the scheme. We apply these stabilization methods to a nonlinear problem and compare the results. In Chapter 6 we present results from several numerical experiments of nonlinear three-dimensional water waves. In Chapter 7 we briefly summarize our work.

Chapter 2

The Boundary Integral Formulation

2.0.1 General Formulation

We consider an interface separating two incompressible, inviscid, irrotational fluids as in Figure 2.1. We assume the interface to be free from surface tension. We parameterize the interface using surface parameters r and s so that $\mathbf{x}(r,s,t) = (x(r,s,t), y(r,s,t), z(r,s,t))$ where t is time. We label the region above the interface as region 1 and that below as region 2. The normalized tangent vectors of the surface $\hat{\mathbf{t}}_1$ and $\hat{\mathbf{t}}_2$ are defined by

$$\hat{\mathbf{t}}_1 = \frac{\partial \mathbf{x}}{\partial r} \Big/ \left|\frac{\partial \mathbf{x}}{\partial r}\right| \quad \text{and} \quad \hat{\mathbf{t}}_2 = \frac{\partial \mathbf{x}}{\partial s} \Big/ \left|\frac{\partial \mathbf{x}}{\partial s}\right| \tag{2.1}$$

and the normal to the surface $\hat{\mathbf{n}}$ is defined by

$$\hat{\mathbf{n}} = \frac{\dfrac{\partial \mathbf{x}}{\partial r} \times \dfrac{\partial \mathbf{x}}{\partial s}}{\left|\dfrac{\partial \mathbf{x}}{\partial r} \times \dfrac{\partial \mathbf{x}}{\partial s}\right|}. \tag{2.2}$$

The velocity field $\mathbf{u}_1$ ($\mathbf{u}_2$) is the velocity above (below) the interface. We define $\mathbf{u}_+$ to be the limit of $\mathbf{u}_1$ approaching the interface from Region 1 and $\mathbf{u}_-$ to be the limit of $\mathbf{u}_2$ approaching the interface from Region 2.

The flow in each region is irrotational, so we can introduce the velocity potentials ϕ_1 and ϕ_2 given by $\mathbf{u}_1 = \nabla\phi_1$ and $\mathbf{u}_2 = \nabla\phi_2$. Since the flows are incompressible,

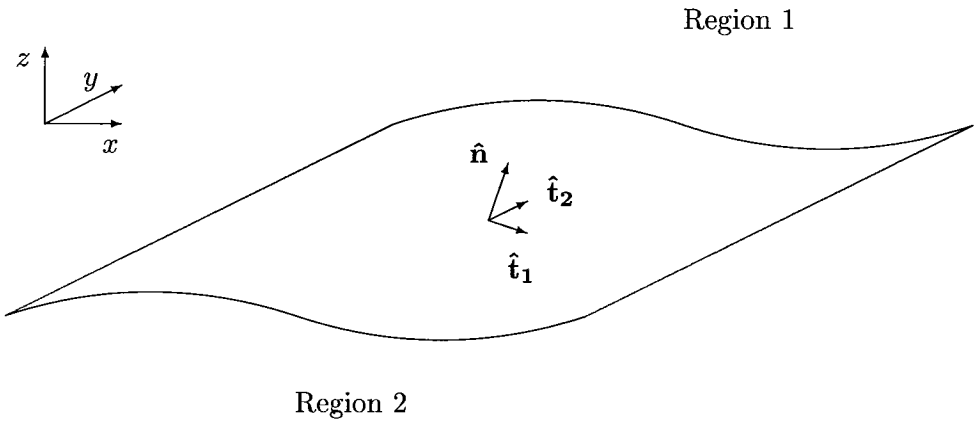

Figure 2.1: Regions 1 and 2 with the normal vector $\hat{\mathbf{n}}$ pointed into region 1 and tangent vectors $\hat{\mathbf{t}}_1$ and $\hat{\mathbf{t}}_2$.

the velocity potentials satisfy Laplace's equation:

$$\nabla^2\phi_2 = \nabla^2\phi_1 = 0. \tag{2.3}$$

The motion of the interface is defined by

$$\frac{\partial \mathbf{x}}{\partial t} = \tilde{\mathbf{u}} \tag{2.4}$$

where $\tilde{\mathbf{u}} = (\tilde{u}, \tilde{v}, \tilde{w})$ is the velocity of particles on the interface. The kinematic condition that the interface moves with the fluid requires that the normal component of velocity be continuous at the interface. However, the tangential velocity at the interface is arbitrary. We therefore have some freedom in choosing $\tilde{\mathbf{u}}$.

The continuity of normal stresses gives the Bernoulli equation which expresses the time evolution of ϕ_-. At the interface, we have

$$\frac{D\phi_-}{Dt} - \tilde{u}u_- - \tilde{v}v_- - \tilde{w}w_- + \frac{1}{2}(u_-^2 + v_-^2 + w_-^2) + gz = 0. \tag{2.5}$$

In order to use these evolution equations, we need a method for computing the velocities. The method we develop here relies on singular Biot-Savart integrals.

The interface can be considered as a distribution of dipoles of strength density $\hat{\mathbf{n}}(\phi_- - \phi_+)/4\pi$ (see, for example, Saffman [24]). The potential at a point off the interface is defined as

$$\phi(x,y,z) = -\frac{1}{4\pi}\int_{S'} \mu(\mathbf{x}')\hat{\mathbf{n}}(\mathbf{x}')\cdot\nabla_{\mathbf{x}}G(\mathbf{x},\mathbf{x}')dS', \tag{2.6}$$

where $\hat{\mathbf{n}}$ is the normal to the surface, $\mu = \phi_- - \phi_+$, and

$$G(\mathbf{x},\mathbf{x}') = -\frac{1}{|\mathbf{x}-\mathbf{x}'|}. \tag{2.7}$$

Depending on the direction from which we approach the interface, the potential assumes the value ϕ_- or ϕ_+. On the interface, the integral is defined by a principal value integral, which we denote by ϕ. The value ϕ is the mean of the two limits:

$$\phi(r,s) = \frac{\phi_+ + \phi_-}{2}. \tag{2.8}$$

Using this relation and the definition of μ, we obtain a Fredholm integral equation of the second kind:

$$\mu = 2\phi_- - 2\phi = 2\phi_- + \frac{1}{2\pi}\int_{S'} \mu(\mathbf{x}')\hat{\mathbf{n}}\cdot\nabla' G(\mathbf{x},\mathbf{x}')dS', \tag{2.9}$$

where the integral is a principal value integral. This equation can be solved by iteration on μ.

The velocity jump at the interface can be defined in terms of derivatives of the dipole sheet strength:

$$\mathbf{u}_- - \mathbf{u}_+ = \frac{\left(\frac{\partial\mathbf{x}}{\partial s}\times\hat{\mathbf{n}}\right)\frac{\partial\mu}{\partial r} - \left(\frac{\partial\mathbf{x}}{\partial r}\times\hat{\mathbf{n}}\right)\frac{\partial\mu}{\partial s}}{\left|\frac{\partial\mathbf{x}}{\partial r}\times\frac{\partial\mathbf{x}}{\partial s}\right|}. \tag{2.10}$$

This gives a formula for the vortex sheet strength $\boldsymbol{\kappa} = \hat{\mathbf{n}}\times(\mathbf{u}_+ - \mathbf{u}_-)$ in terms of the

derivatives of μ:

$$\boldsymbol{\kappa} = \frac{-\frac{\partial \mathbf{x}}{\partial s}\frac{\partial \mu}{\partial r} + \frac{\partial \mathbf{x}}{\partial r}\frac{\partial \mu}{\partial s}}{\left|\frac{\partial \mathbf{x}}{\partial r} \times \frac{\partial \mathbf{x}}{\partial s}\right|}. \tag{2.11}$$

The fluid velocity off the interface can also be defined using a boundary integral (see Saffman [24]):

$$\mathbf{u}(x,y,z) = \frac{1}{4\pi}\int_{S'} \boldsymbol{\kappa}(\mathbf{x}') \times \nabla_{\mathbf{x}} G(\mathbf{x},\mathbf{x}') dS'. \tag{2.12}$$

The value of the Biot-Savart integral on the interface $\mathbf{u}(r,s)$ is defined by a principal value integral whose value is given by

$$\mathbf{u}(r,s) = \frac{\mathbf{u}_+ + \mathbf{u}_-}{2}. \tag{2.13}$$

Since we can explicitly compute the tangential derivatives of ϕ, we need only compute one component of $\mathbf{u}$ using the Biot-Savart law. The other components are then determined by the following relations:

$$\frac{\partial \phi}{\partial r} = \frac{\partial \mathbf{x}}{\partial r} \cdot \mathbf{u} \tag{2.14}$$

$$\frac{\partial \phi}{\partial s} = \frac{\partial \mathbf{x}}{\partial s} \cdot \mathbf{u}. \tag{2.15}$$

We now define $\mathbf{u}_-$. Noting that $\mathbf{u}_- - \mathbf{u}_+ = -\boldsymbol{\kappa} \times \hat{\mathbf{n}}$, we have

$$\mathbf{u}_- = \mathbf{u} - \frac{1}{2}\boldsymbol{\kappa} \times \hat{\mathbf{n}}. \tag{2.16}$$

The particle velocity $\tilde{\mathbf{u}}$ often takes the form

$$\frac{D\mathbf{x}}{Dt} = \tilde{\mathbf{u}} = \mathbf{u} + \alpha(\mathbf{u}_- - \mathbf{u}_+). \tag{2.17}$$

Table 2.1 shows representations of $\tilde{\mathbf{u}}$ for various values of α. The value of α is usually taken to be 1.

α	$\tilde{\mathbf{u}}$
-1	$\mathbf{u}_+$
0	$\mathbf{u}$
1	$\mathbf{u}_-$

Table 2.1: Values of $\tilde{\mathbf{u}}$ for various α.

We now have a method for calculating the time evolution of water waves given an initial surface $\mathbf{x}$ and interfacial potential ϕ_-:

1. Compute μ from the Fredholm integral equation.
2. Compute κ from μ.
3. Compute $\mathbf{u}$ using the Biot-Savart law.
4. Compute $\mathbf{u}_-$ and $\tilde{\mathbf{u}}$.
5. Step $\mathbf{x}$ and ϕ_- forward in time.

2.0.2 Fixed Grid Formulation

We have implemented a simpler formulation that constrains the particles to move vertically. That is, we take

$$\frac{dx}{dt} = \tilde{u} = 0 \tag{2.18}$$

$$\frac{dy}{dt} = \tilde{v} = 0. \tag{2.19}$$

We define $r = x$, $s = y$, and $z = \eta(x, y)$ where $\eta(x, y)$ is a single valued function of x and y. The surface is then defined by $\mathbf{x} = (x, y, \eta(x, y))$. We are allowed to impose the vertical velocity constraint because the tangential velocities are arbitrary. The condition that the normal velocity be continuous at the interface then specifies the value of the vertical velocity $\tilde{w}$:

$$\tilde{\mathbf{u}} \cdot \hat{\mathbf{n}} = \mathbf{u}_- \cdot \hat{\mathbf{n}}. \tag{2.20}$$

Using this we write

$$\frac{\partial \eta}{\partial t} = \tilde{w} = w_- - u_- \frac{\partial \eta}{\partial x} - v_- \frac{\partial \eta}{\partial y}. \tag{2.21}$$

Using this formulation gives a simplified formula for the vortex sheet strength and the velocity $\mathbf{u}_-$. If we define

$$\gamma_x(x,y) = \frac{\partial}{\partial x}\mu(x,y) \text{ and } \gamma_y(x,y) = \frac{\partial}{\partial y}\mu(x,y), \tag{2.22}$$

then we have the following expression for $\boldsymbol{\kappa}$ in terms of μ:

$$\boldsymbol{\kappa}(x,y) = \hat{\mathbf{n}} \times [\mathbf{u}_+ - \mathbf{u}_-] = \frac{\gamma_y \hat{\mathbf{i}} - \gamma_x \hat{\mathbf{j}} + \left(\frac{\partial \eta}{\partial x}\gamma_y - \frac{\partial \eta}{\partial y}\gamma_x\right)\hat{\mathbf{k}}}{\sqrt{1 + \left(\frac{\partial \eta}{\partial x}\right)^2 + \left(\frac{\partial \eta}{\partial y}\right)^2}}. \tag{2.23}$$

After computing $\mathbf{u} = (u, v, w)$ from the Biot-Savart integral, we can compute $\mathbf{u}_-$ as follows:

$$\begin{aligned} w_- &= \frac{w - \frac{\partial \eta}{\partial x}\left(u - \frac{\partial \phi_-}{\partial x}\right) - \frac{\partial \eta}{\partial y}\left(v - \frac{\partial \phi_-}{\partial y}\right)}{1 + \left(\frac{\partial \eta}{\partial x}\right)^2 + \left(\frac{\partial \eta}{\partial y}\right)^2}, \\ u_- &= \frac{\partial \phi_-}{\partial x} - w_- \frac{\partial \eta}{\partial x}, \\ v_- &= \frac{\partial \phi_-}{\partial y} - w_- \frac{\partial \eta}{\partial y}. \end{aligned} \tag{2.24}$$

The algorithm for computing the velocities and stepping η and ϕ_- forward in time are essentially the same as in the general case, with the time evolution equations for η and ϕ given by

$$\frac{D\eta}{Dt} = \tilde{w} \tag{2.25}$$

$$\frac{D\phi_-}{Dt} = \tilde{w} w_- - \frac{1}{2}(u_-^2 + v_-^2 + w_-^2) - g\eta \tag{2.26}$$

$$\tilde{w} = w_- - u_- \frac{\partial \eta}{\partial x} - v_- \frac{\partial \eta}{\partial y}. \tag{2.27}$$

In the next chapter we present and analyze a method for approximating the

singular integrals which define the velocity and velocity potential of the surface wave.

Chapter 3

Numerical Analysis

3.1 The Point Vortex Method

We are interested in the computation of the singular Biot-Savart integrals which define the velocity and the velocity potential. We limit ourselves to the former and note that the analysis of the integral defining the potential is similar. We consider the case of the fixed grid formulation. The extension to the more general case is not difficult. We assume that η, γ_x, and γ_y are periodic in x and y with period 2π. We can express the velocity integral as follows:

$$\begin{aligned}\mathbf{u}(x,y) &= \frac{1}{4\pi}\int_{S'}\boldsymbol{\kappa}(\mathbf{x}')\times\nabla_{\mathbf{x}}G(\mathbf{x},\mathbf{x}')dS' \\ &= \frac{1}{4\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\frac{\left[\alpha\hat{\mathbf{i}}+\beta\hat{\mathbf{j}}+\delta\hat{\mathbf{k}}\right]dx'dy'}{\left[(x-x')^2+(y-y')^2+(\eta(x,y)-\eta(x',y'))^2\right]^{3/2}} \qquad (3.1)\end{aligned}$$

where

$$\begin{aligned}\alpha &= \left[\frac{\partial\eta}{\partial y}(x',y')\gamma_x(x',y')-\frac{\partial\eta}{\partial x}(x',y')\gamma_y(x',y')\right](y-y') \\ &\quad -\gamma_x(x',y')\left[\eta(x,y)-\eta(x',y')\right], \\ \beta &= \left[\frac{\partial\eta}{\partial x}(x',y')\gamma_y(x',y')-\frac{\partial\eta}{\partial y}(x',y')\gamma_x(x',y')\right](x-x') \\ &\quad -\gamma_y(x',y')\left[\eta(x,y)-\eta(x',y')\right], \\ \delta &= \gamma_x(x',y')(x-x')+\gamma_y(x',y')(y-y').\end{aligned}$$

In order to simplify the notation, we consider the evaluation of the velocity at the origin. That is, we set $(x, y) = (0, 0)$ and drop the primes. Our conclusions are valid for general (x, y). We also limit discussion to the evaluation of the z-component of $\mathbf{u}(x, y) = (u, v, w)$. The cases involving the other components are completely analogous. Applying the simplifications mentioned above gives

$$w(0,0) = -\frac{1}{4\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\frac{x\gamma_x(x,y) + y\gamma_y(x,y)}{[x^2 + y^2 + (\eta(0,0) - \eta(x,y))^2]^{3/2}}dxdy. \tag{3.2}$$

The point vortex method approximation to $w(0, 0)$ is an application of the two-dimensional trapezoidal rule, excluding the singular contribution. If we define the integrand to be $f(x, y)$, and denote the approximation of $w(0, 0)$ by $w_A(0, 0)$, then the method is defined as follows:

$$w_A(0,0) = -\frac{1}{4\pi}h^2 \sum_{\mathbf{j}\neq\mathbf{0}} f(j_1 h, j_2 h) \tag{3.3}$$

where $h = 2\pi/n$, $(x, y) = (j_1 h, j_2 h)$, and $\mathbf{j} = (j_1, j_2)$.

The computation of the sum in this form would be impractical because of the infinite range of summation and the slow decay of the Green's function kernel. In Chapter 4 we detail several methods for approximating this sum efficiently, including the Ewald summation technique and fast multipole algorithms. In the remainder of this chapter, we study the error of the point vortex approximation and develop high-order quadrature rules.

3.2 Richardson Extrapolation

In order to develop quadrature methods which are highly accurate, we examine the error

$$E = w_A(0,0) - w(0,0). \tag{3.4}$$

Based on the studies mentioned in the introduction, we suspect that E should be a series in powers of h. If this is the case, then we can use Richardson extrapolation

to obtain a higher-order method for computing w. In practice, we apply Richardson extrapolation by noting that it is equivalent to assigning weights to each grid point. This gives a modified definition of w_A:

$$w_A(0,0) = -\frac{1}{4\pi}h^2 \sum_{i=-\frac{n}{2}+1}^{\frac{n}{2}} \sum_{\substack{j=-\frac{n}{2}+1 \\ (i,j)\neq 0}}^{\frac{n}{2}} w_{ij} f(ih, jh). \tag{3.5}$$

Here w_{ij} is the weight associated with Richardson extrapolation. As an example, let $I_A[h]$ be the numerical approximation to I. If we know that

$$I_A[h] = I + C_1 h + C_2 h^2 + \cdots, \tag{3.6}$$

then we can use Richardson extrapolation by noting that

$$I_A[2h] = I + 2C_1 h + 4C_2 h^2 + \cdots. \tag{3.7}$$

Subtracting $I_A[2h]$ from $2I_A[h]$ gives

$$2I_A[h] - I_A[2h] = I - 2C_2 h^2 + \cdots. \tag{3.8}$$

We see that the first-order term has been removed, leaving a second-order approximation for I. To determine the weights w_{ij}, we define the function $g(i,j)$ to be:

$$g(i,j) = \begin{cases} 1 & \text{if } i \text{ and } j \text{ are both even} \\ 0 & \text{otherwise} \end{cases}. \tag{3.9}$$

Using this definition, we write

$$\begin{aligned} 2I_A[h] - I_A[2h] \;=\; & -\frac{2}{4\pi}h^2 \sum_{i=-\frac{n}{2}+1}^{\frac{n}{2}} \sum_{\substack{j=-\frac{n}{2}+1 \\ (i,j)\neq 0}}^{\frac{n}{2}} f(ih, jh) \\ & + \frac{1}{4\pi} 4h^2 \sum_{i=-\frac{n}{4}+1}^{\frac{n}{4}} \sum_{\substack{j=-\frac{n}{4}+1 \\ (i,j)\neq 0}}^{\frac{n}{4}} f(2ih, 2jh) \end{aligned} \tag{3.10}$$

$$= -\frac{1}{4\pi}h^2 \sum_{i=-\frac{n}{2}+1}^{\frac{n}{2}} \sum_{\substack{j=-\frac{n}{2}+1 \\ (i,j)\neq 0}}^{\frac{n}{2}} [2 - 4g(i,j)]\, f(ih, jh).$$

From this we see that the definition of w_{ij} for a second-order method would be

$$w_{ij} = 2 - 4g(i,j). \tag{3.11}$$

The condition that $g(i,j) = 1$ occurs when i and j are both even and this occurs when the lattice point (i,j) contributes to both $I_A[h]$ and $I_A[2h]$. Thus, lattice points contributing only to $I_A[h]$ have $w_{ij} = 2$ while those contributing to both sums have $w_{ij} = -2$.

In our numerical experiments with (3.5), the application of Richardson extrapolation led us to believe that E can be expressed as a series in odd powers of h. In Figure 3.1, we show a calculation of the error associated with the point-vortex method and the first two applications of Richardson extrapolation. The first application of Richardson extrapolation yields a third-order method. The second application of Richardson extrapolation is applied so as to explicitly remove the third-order term. This yields a fifth-order method. If we assume a series in integer powers of h and apply Richardson extrapolation, the first and second applications of the method both yield a third-order method because there is no second-order term. In Figure 3.2 we show the slopes (in absolute value) of the respective error curves in Figure 3.1. In this example, we have taken

$$\begin{aligned} \eta &= \sin(x+y) + \cos^2(x+y), \\ \gamma_x &= 0, \\ \gamma_y &= -2\left[\sin(y) + \cos(y)\right]. \end{aligned} \tag{3.12}$$

The figures show (as expected) that the point vortex method is a first-order method. The successive application of Richardson extrapolation yields third- and fifth-order methods, respectively. This suggests that the error is a series in odd powers of h.

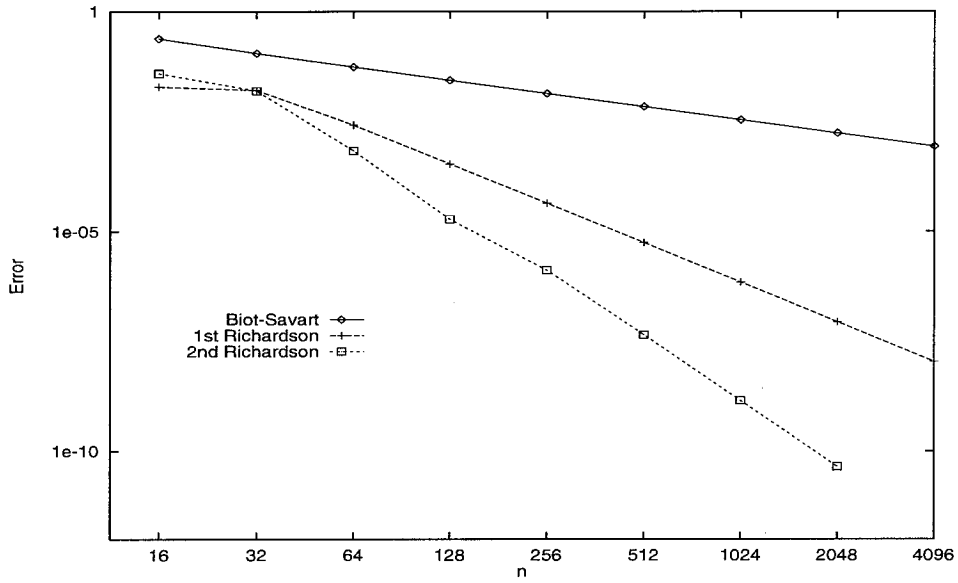

Figure 3.1: The line labeled *Biot-Savart* gives the error from the point vortex method. The other lines give the error for the first two applications of Richardson extrapolation.

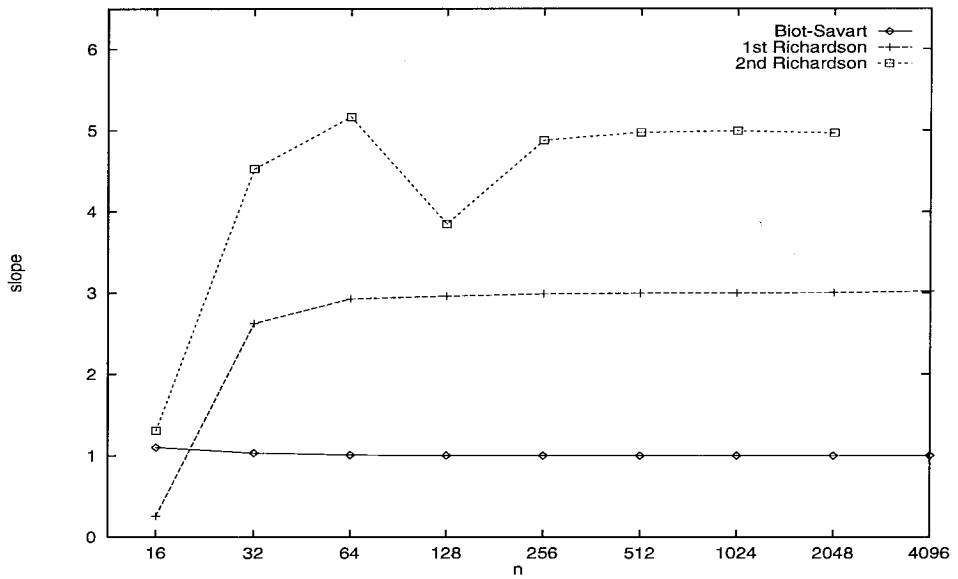

Figure 3.2: The order of the 3 different methods demonstrated in Figure 3.1.

The analysis of the method which follows confirms this claim.

3.3 Analysis of the Point Vortex Method

In order to justify the general use of Richardson extrapolation, we must demonstrate that the error associated with the point vortex method has the appropriate form. Our analysis of the method relies on two different approaches. The first approach, which demonstrates that the error is a series in odd powers of h, is based on work by Goodman, Hou, and Lowengrub [7]. The second method is an application of the two-dimensional Euler-Maclaurin summation formula which demonstrates how to construct the coefficients in this series. Because of the comparative simplicity of the method of Goodman *et al.*, we discuss their ideas first.

3.3.1 The Form of the Error

We consider the point vortex approximation to the integral

$$I = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} K(x,y)p(x,y)dxdy, \tag{3.13}$$

where

$$K(x,y) = \frac{1}{[x^2 + y^2 + (\eta(0,0) - \eta(x,y))^2]^{3/2}}, \tag{3.14}$$

and

$$p(x,y) = -\frac{1}{4\pi}\left[x\gamma_x(x,y) + y\gamma_y(x,y)\right]. \tag{3.15}$$

We assume that γ_x and γ_y are smooth, periodic, and bounded in x and y. The error E associated with this approximation is defined by

$$E = h^2 \sum_{i=-\infty}^{\infty} \sum_{\substack{j=-\infty \\ (i,j)\neq(0,0)}}^{\infty} K(x_i,y_j)p(x_i,y_j) - \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} K(x,y)p(x,y)dxdy \tag{3.16}$$

where we use the notation $(x_i, y_j) = (ih, jh)$.

Following the treatment of Goodman *et al.*, we split E into a near field part, σ, and a far field part, τ. To do this, we define a cutoff function $f_1(s)$. We take f_1 to

be a function which is smooth for all s and which satisfies the following: $f_1(s) \equiv 1$ if $|s| \leq 1$, $f_1(s) \equiv 0$ if $|s| \geq 2$, and $f_1(s)$ varies smoothly from 0 to 1 for $1 \leq |s| \leq 2$. We also define $f_2(s) = 1 - f_1(s)$. We take $0 < q < 1$ and define

$$\begin{aligned} \sigma &= h^2 \sum_{i=-\infty}^{\infty} \sum_{\substack{j=-\infty \\ (i,j)\neq 0}}^{\infty} K(x_i, y_j) p(x_i, y_j) f_1(h^{-q}|\mathbf{x_k}|) \\ &\quad - \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} K(x,y) p(x,y) f_1(h^{-q}|\mathbf{x}|) dx dy, \end{aligned} \tag{3.17}$$

and

$$\begin{aligned} \tau &= h^2 \sum_{i=-\infty}^{\infty} \sum_{\substack{j=-\infty \\ (i,j)\neq 0}}^{\infty} K(x_i, y_j) p(x_i, y_j) f_2(h^{-q}|\mathbf{x_k}|) dx dy \\ &\quad - \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} K(x,y) p(x,y) f_2(h^{-q}|\mathbf{x}|) dx dy, \end{aligned} \tag{3.18}$$

where $\mathbf{x} = (x, y)$ and $\mathbf{x_k} = (x_i, y_j)$. To analyze σ, we express $K(x,y)p(x,y)$ in a series expansion in x and y. Since $p(x,y)$ has no singularities, we have the following expansion for p:

$$p(x,y) = p_x^0 x + p_y^0 y + \cdots + O(|\mathbf{x}|^N) \tag{3.19}$$

for any N, where we use the notation

$$p_x^0 = \frac{\partial}{\partial x} p \Big|_{(x,y)=(0,0)}. \tag{3.20}$$

We expand the denominator of $K(x,y)$ as follows:

$$\begin{aligned} &\left[x^2 + y^2 + (\eta(0,0) - \eta(x,y))^2\right]^{3/2} = \\ &\qquad x^2 + y^2 + (\eta_x^0 x + \eta_y^0 y)^2 + \left(\eta_x^0 x + \eta_y^0 y\right)\left(\eta_{xx}^0 x^2 + \eta_{yy}^0 + 2\eta_{xy}^0 xy\right) \\ &\qquad + \cdots + O(|\mathbf{x}|^N) \\ &= \left(ax^2 + by^2 + dxy\right)^{3/2} \left(1 + \frac{(\eta_x^0 x + \eta_y^0 y)(\eta_{xx}^0 x^2 + \eta_{yy}^0 + 2\eta_{xy}^0 xy)}{(ax^2 + by^2 + dxy)^{3/2}} + \cdots\right)^{3/2} \end{aligned} \tag{3.21}$$

for any N. In this equation, we have

$$a = 1 + \eta_x^2(0,0), \tag{3.22}$$

$$b = 1 + \eta_y^2(0,0), \tag{3.23}$$

$$d = 2\eta_x(0,0)\eta_y(0,0). \tag{3.24}$$

From the above results we have that

$$K(x,y)p(x,y) = C_0(x,y) + C_1(x,y) + C_2(x,y) + \cdots + O(|\mathbf{x}|^N), \tag{3.25}$$

where $C_n(x,y)$ is homogeneous of degree $n-2$ in $\mathbf{x}$. In addition, C_n is even in $\mathbf{x}$ for odd n and odd in $\mathbf{x}$ for even n. Since $f_1(|\mathbf{x}|)$ is even in $\mathbf{x}$, we have

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} C_n(x,y) f_1(h^{-q}|\mathbf{x}|)dxdy = 0 \tag{3.26}$$

for n even. The substitution $(x,y) = h^q(\eta_1,\eta_2)$ gives, for n odd, that

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} C_n(x,y) f_1(h^{-q}|\mathbf{x}|)dxdy = h^{qn}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} C_n(\eta_1,\eta_2) f(|\eta|)d\eta_1 d\eta_2, \tag{3.27}$$

where $\eta = (\eta_1,\eta_2)$. We define a_n as follows:

$$a_n = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} C_n(\eta_1,\eta_2) f(|\eta|)d\eta_1 d\eta_2. \tag{3.28}$$

We now analyze the discrete sum in σ (3.17). Again using the fact that $f_1(\mathbf{x})$ is even in $\mathbf{x}$ and $C_n(x,y)$ is odd in $\mathbf{x}$ for odd n, we have

$$h^2 \sum_{i=-\infty}^{\infty} \sum_{\substack{j=-\infty \\ (i,j)\neq(0,0)}}^{\infty} C_n(hi,hj) f_1(h^{1-q}|k|) = 0 \tag{3.29}$$

for n even, where we define $k = (i,j)$. For n odd, we make use of the homogeneity

of C_n to rewrite the sum as follows:

$$h^2 \sum_{i=-\infty}^{\infty} \sum_{\substack{j=-\infty \\ (i,j)\neq(0,0)}}^{\infty} C_n(hi, hj) f_1(h^{1-q}|k|) = h^n S_n(h), \tag{3.30}$$

where

$$S_n(h) = \sum_{i=-\infty}^{\infty} \sum_{\substack{j=-\infty \\ (i,j)\neq(0,0)}}^{\infty} C_n(i, j) f_1(h^{1-q}|k|). \tag{3.31}$$

Goodman *et al.* [7] show that $S_n(h)$ can be expressed in the following form:

$$S_n(h) = D_n + b_n h^{-n(1-q)} + O(h^N) \text{ for any } N, \tag{3.32}$$

where D_n is constant and b_n depends on the cutoff function f_1. Combining this result with the result for the integral terms (3.28) gives

$$\sigma = D_1 h + D_3 h^3 + \cdots + d_1 h^q + d_3 h^{3q} + \cdots + O(h^N) \text{ for any } N \tag{3.33}$$

where we take $d_i = b_i - a_i$.

The theory of numerical quadrature tells us that $\tau = o(h^N)$ for any N since the integrand has no singularities and decays at infinity. This gives

$$E = \sigma + \tau = D_1 h + D_3 h^3 + \cdots + d_1 h^q + d_3 h^{3q} + \cdots + O(h^N). \tag{3.34}$$

Since E must be independent of q, we conclude that $d_i = 0$ for any i. This leaves

$$E = \sigma + \tau = D_1 h + D_3 h^3 + \cdots + O(h^N). \tag{3.35}$$

The fact that we have a series in odd powers of h rather than in even powers of h (as in Goodman *et al.*) is due to the form of K.

3.3.2 Computation of the Coefficients: Overview

The computation of the coefficients is based on studies of the Euler-Maclaurin summation formula. Our approach follows closely that of Sidi and Israeli [26] and Navot [20] who studied 1-dimensional singular integrals. We first sketch the approach and then cite some examples.

Recall that we are studying integrals of the form

$$I = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} K(x,y)p(x,y)dxdy \tag{3.36}$$

where $K(x,y)$ is defined by (3.14) and

$$\begin{aligned} p(x,y) &= -\frac{x\gamma_x(x,y) + y\gamma_y(x,y)}{4\pi} \\ &= xg(x,y) + yq(x,y). \end{aligned} \tag{3.37}$$

As before, we assume that $\gamma_x(x,y)$ and $\gamma_y(x,y)$ are smooth and periodic in x and y, with period 2π in each variable. We take I_A to be the point-vortex approximation to I:

$$I_A = h^2 \sum_{i=-\infty}^{\infty} \sum_{\substack{j=-\infty \\ (i,j)\neq 0}}^{\infty} f(x_i, y_j), \tag{3.38}$$

where we have defined

$$f(x,y) = K(x,y)p(x,y). \tag{3.39}$$

The error E is defined by $E = I_A - I$.

The derivation of the coefficients follows the method outlined by Navot [20] which extended the (1-dimensional) Euler-Maclaurin summation formula to include functions with branch singularities. Navot's basic approach is to apply the Euler-Maclaurin formula to a region where the integrand is non-singular, and treat the singular region separately. We here outline the approach and leave the details to the next section. We begin by defining the following:

$$T = \mathbb{R}^2 / \Big\{ [-h,h] \times [-h,h] \Big\}, \tag{3.40}$$

$$I_T = \int\int_T f(x,y)dxdy \tag{3.41}$$

$$I_S = \int_{-h}^{h}\int_{-h}^{h} f(x,y)dxdy \tag{3.42}$$

$$\begin{aligned} I_{A_S} &= \frac{f(h,h)+f(h,-h)+f(-h,h)+f(-h,-h)}{4} \\ &+ \frac{f(0,h)+f(h,0)+f(0,-h)+f(-h,0)}{2}. \end{aligned} \tag{3.43}$$

We rewrite I as follows:

$$\begin{aligned} I &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)dxdy \\ &= I_T + I_S. \end{aligned} \tag{3.44}$$

We define I_{A_T} as follows:

$$I_{A_T} = I_A - I_{A_S}. \tag{3.45}$$

We have defined I_{A_S} in such a way that I_{A_T} is the trapezoidal rule approximation to I_T. We then use the Euler-Maclaurin formula to analyze E_T:

$$E_T = I_{A_T} - I_T. \tag{3.46}$$

We analyze $E_S = I_{A_S} - I_S$ by expanding $p(x,y)$ in a series about $(0,0)$. Combining the results for E_T and E_S yields a series in odd powers of h. We remark that we have subdivided the domain of integration in such a way that the Euler-Maclaurin formula is applied only on a domain where the integrand and its derivatives are non-singular.

The Euler-Maclaurin formula is usually applied to integrals where the domain of integration is independent of h. Our application of the Euler-Maclaurin formula is different in that we apply the formula to an integral where the domain of integration depends on h. We note that while the domain upon which the formula is applied is arbitrarily close to the singular point, the use of the formula is valid since the integrand and its derivatives are nonsingular on the domain. We also note that

this approach can be used to study the error in approximating each of the singular integrals that we compute using the point-vortex method.

3.3.3 Examples

In order to simplify the notation, we make the following definitions:

$$D_x^j f \equiv \frac{\partial^j f}{\partial x^j}, \qquad \hat{D}_x^j f \equiv P_j(x)\frac{\partial^j f}{\partial x^j}. \tag{3.47}$$

The function $P_m(x)$ is given by

$$P_m(x) = \frac{\overline{B}_m(x)}{m!} \tag{3.48}$$

where $\overline{B}_m(x)$ is the Bernoulli polynomial of order m on $[0,1]$ extended periodically. B_i is the i'th Bernoulli number.

Because the notation and formulas are lengthy and somewhat cumbersome, we present the complete formula only for the simplest case involving a flat interface. We assume the following:

$$\eta(x,y) \equiv 0 \tag{3.49}$$

$$\gamma_y(x,y) \equiv 0 \tag{3.50}$$

$$\gamma_x(x,y) \equiv \gamma_x(x). \tag{3.51}$$

Given these assumptions, the formula for the error expansion is:

$$E = \sum_{k=1}^{m-1} D_{2k-1} h^{2k-1} + O(h^{2m-1}), \tag{3.52}$$

where

$$D_{2k-1} = -4 \sum_{\gamma=1}^{m} \frac{B_{2\gamma}}{(2\gamma)!} \left[\int_0^1 D_x^{2\gamma-1} C_{2k-1}(1,y)dy + \int_0^1 D_y^{2\gamma-1} C_{2k-1}(x,1)dx \right.$$
$$\left. + \sum_{\mu=1}^{m} \frac{B_{2\mu}}{(2\mu!)} D_x^{2\gamma-1} D_y^{2\mu-1} C_{2k-1}(1,1) \right.$$

$$+\int_0^1 \hat{D}_x^{2m+1} D_y^{2\gamma-1} C_{2k-1}(x,1)dx + \int_0^1 \hat{D}_x^{2\gamma-1} D_y^{2m+1} C_{2k-1}(1,y)dy\Bigg] \quad (3.53)$$
$$+4\int_1^\infty \int_0^\infty \hat{D}_x^{2m+1} C_{2k-1}(x,y)dydx + 4\int_0^1 \int_1^\infty \hat{D}_x^{2m+1} C_{2k-1}(x,y)dydx$$
$$+4\int_1^\infty \int_0^\infty \hat{D}_y^{2m+1} C_{2k-1}(x,y)dxdy + 4\int_0^1 \int_1^\infty \hat{D}_y^{2m+1} C_{2k-1}(x,y)dxdy$$
$$+4\int_1^\infty \int_0^\infty \hat{D}_x^{2m+1}\hat{D}_y^{2m+1} C_{2k-1}(x,y)dydx$$
$$+4\int_0^1 \int_1^\infty \hat{D}_x^{2m+1}\hat{D}_y^{2m+1} C_{2k-1}(x,y)dydx$$
$$-4\int_0^1 \int_0^1 C_{2k-1}(x,y)dxdy + [C_{2k-1}(1,0) + C_{2k-1}(1,1)]\,,$$

and

$$C_k(x,y) = \frac{\gamma_x^k(0)x^{k+1}}{k!(x^2+y^2)^{3/2}}. \quad (3.54)$$

We note here that given any m, we can calculate D_{2k-1} for all k up to and including $k = m-1$. For example, to calculate D_1, we need $m \geq 2$. We also observe that (3.53) suggests that there are multiple representations for the coefficients D_{2k-1}. The different representations depend on the value of m chosen. Because m is an arbitrary parameter, the value of the coefficients should be independent of m. We have done some numerical experiments which verify this fact.

We demonstrate the method by calculating D_1 and D_3 for a simple problem. We take $\gamma_x(x,y) = -2\sin(x)$, $\gamma_y \equiv 0$, and $\eta \equiv 0$. Recalling (3.15), we have

$$K(x,y)p(x,y) = \frac{x\sin(x)}{2\pi(x^2+y^2)^{3/2}}. \quad (3.55)$$

Expanding $\sin(x)$, we calculate C_1 and C_3 as follows:

$$\begin{aligned} C_1 &= \frac{x^2}{2\pi(x^2+y^2)^{3/2}}, \\ C_3 &= -\frac{x^4}{12\pi(x^2+y^2)^{3/2}}. \end{aligned} \quad (3.56)$$

Taking $m = 3$, we have expressions for D_1 and D_3 from (3.53):

$$
\begin{aligned}
D_1 = \Bigg\{ & -4 \sum_{\gamma=1}^{3} \frac{B_{2\gamma}}{(2\gamma)!} \Bigg[\int_0^1 D_x^{2\gamma-1} C_1(1,y)dy + \int_0^1 D_y^{2\gamma-1} C_1(x,1)dx \\
& + \sum_{\mu=1}^{3} \frac{B_{2\mu}}{(2\mu)!} D_x^{2\gamma-1} D_y^{2\mu-1} C_1(1,1) + \int_0^1 \hat{D}_x^7 D_y^{2\gamma-1} C_1(x,1)dx \\
& + \int_0^1 \hat{D}_y^7 D_x^{2\gamma-1} C_1(1,y)dy \Bigg] + 4 \int_1^\infty \int_0^\infty \hat{D}_x^7 C_1(x,y)dydx \\
& + 4 \int_0^1 \int_1^\infty \hat{D}_x^7 C_1(x,y)dydx + 4 \int_1^\infty \int_0^\infty \hat{D}_y^7 C_1(x,y)dxdy \\
& + 4 \int_0^1 \int_1^\infty \hat{D}_y^7 C_1(x,y)dxdy + 4 \int_1^\infty \int_0^\infty \hat{D}_x^7 \hat{D}_y^7 C_1(x,y)dydx \\
& + 4 \int_0^1 \int_1^\infty \hat{D}_x^7 \hat{D}_y^7 C_1(x,y)dydx + C_1(1,0) + C_1(1,1) - \int_{-1}^{1} \int_{-1}^{1} C_1(x,y)dxdy \Bigg\},
\end{aligned}
\tag{3.57}
$$

$$
\begin{aligned}
D_3 = \Bigg\{ & -4 \sum_{\gamma=1}^{3} \frac{B_{2\gamma}}{(2\gamma)!} \Bigg[\int_0^1 D_x^{2\gamma-1} C_3(1,y)dy + \int_0^1 D_y^{2\gamma-1} C_3(x,1)dx \\
& + \sum_{\mu=1}^{3} \frac{B_{2\mu}}{(2\mu)!} D_x^{2\gamma-1} D_y^{2\mu-1} C_3(1,1) + \int_0^1 \hat{D}_x^7 D_y^{2\gamma-1} C_3(x,1)dx \\
& + \int_0^1 \hat{D}_y^7 D_x^{2\gamma-1} C_3(1,y)dy \Bigg] + 4 \int_1^\infty \int_0^\infty \hat{D}_x^7 C_3(x,y)dydx \\
& + 4 \int_0^1 \int_1^\infty \hat{D}_x^7 C_3(x,y)dydx + 4 \int_1^\infty \int_0^\infty \hat{D}_y^7 C_3(x,y)dxdy \\
& + 4 \int_0^1 \int_1^\infty \hat{D}_y^7 C_3(x,y)dxdy + 4 \int_1^\infty \int_0^\infty \hat{D}_x^7 \hat{D}_y^7 C_3(x,y)dydx \\
& + 4 \int_0^1 \int_1^\infty \hat{D}_x^7 \hat{D}_y^7 C_3(x,y)dydx + C_3(1,0) + C_3(1,1) - \int_{-1}^{1} \int_{-1}^{1} C_3(x,y)dxdy \Bigg\}.
\end{aligned}
$$

We have computed D_1 and D_3 to be approximately $-.3103732$ and -1.33299×10^{-3}, respectively. These agree very well with the coefficients of the h and h^3 terms computed from the point vortex calculation. Figure 3.3 demonstrates the comparison between the numerically computed value and the analytical value for D_1.

In the case of an interface that is not flat, the formula is similar, but considerably more lengthy; we therefore omit the formula, but sketch the approach. We can

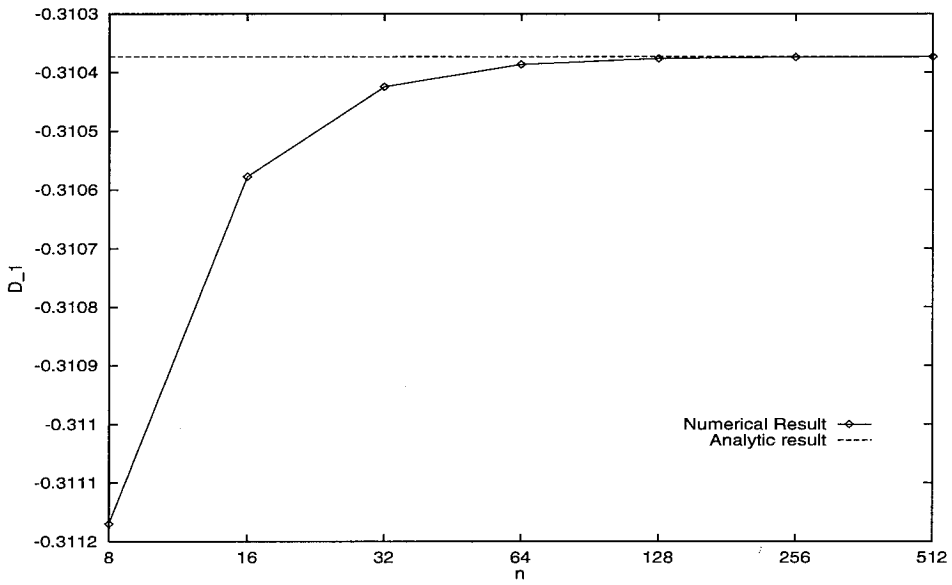

Figure 3.3: The solid line represents the value of D_1 using (3.57). The dashed line represents the value of D_1 computed using the point vortex approximation for various values of n.

limit consideration to the case where $p(x,y) = xg(x,y)$ since the more general case is analogous. We expand $K(x,y)p(x,y)$ as outlined in Section 3.3.1. Our analysis shows that the first term that contributes to the error expansion is C_1, which contains all of the terms which contribute to the first-order coefficient. We have

$$C_1 = \frac{x\left[g_x(0,0)x + g_y(0,0)y\right]}{(Ax^2+By^2+Cxy)^{3/2}} \qquad (3.58)$$
$$-\frac{3}{2}xg(0,0)\frac{\left[\eta_x(0,0)x+\eta_y(0,0)y\right]\left[\eta_{xx}(0,0)x^2+\eta_{yy}(0,0)y^2+2\eta_{xy}(0,0)xy\right]}{(Ax^2+By^2+Cxy)^{5/2}},$$

where

$$\begin{aligned} A &= 1+\eta_x^2(0,0), \\ B &= 1+\eta_y^2(0,0), \qquad (3.59) \\ C &= 2\eta_x(0,0)\eta_y(0,0). \end{aligned}$$

For a specific example, we take

$$\begin{aligned} \gamma_x &= -2(\sin(x)+\cos(x)), \\ \eta(x,y) &= \sin(x+y)+\cos(x+y), \\ \gamma_y &= 0. \end{aligned} \tag{3.60}$$

From C_1, we compute D_1 to be approximately $-.21733$. This gives good agreement with the value we computed using the point vortex method.

3.4 Computation of the Coefficients: Detail

3.4.1 Two dimensional Euler-Maclaurin Summation Formula

Before deriving the formula, we make the following definitions:

$$\begin{aligned} w_i &= \begin{cases} \frac{1}{2} & \text{if } i=0 \text{ or } i=n \\ 1 & \text{otherwise} \end{cases}, \\ D_x^j &\equiv \frac{\partial^j}{\partial x^j}, \qquad \tilde{D}_x^j \equiv P_{2m+1}(x/h)D_x^j. \end{aligned} \tag{3.61}$$

The function $P_m(x)$ is given by

$$P_m(x) = \frac{\overline{B}_m(x)}{m!} \tag{3.62}$$

where $\overline{B}_m(x)$ is the Bernoulli polynomial of order m on $[0,1]$ extended periodically. B_i is the i'th Bernoulli number. The two-dimensional version of the the Euler-Maclaurin summation formula is derived as below: First apply the 1-d formula in the y direction:

$$\begin{aligned} \int_a^b \int_c^d f(x,y)dxdy = \int_a^b \Big[h\sum_{j=0}^{n} w_j f(x,y_j) - \sum_{\mu=1}^{m} \frac{h^{2\mu}B_{2\mu}}{(2\mu)!}\left[D_y^{2\mu-1}f(x,d)\right. \\ \left. -D_y^{2\mu-1}f(x,c)\right] - h^{2m+1}\int_c^d \tilde{D}_y^{2m+1}f(x,y)dy\Big]dx. \end{aligned} \tag{3.63}$$

Next, we apply the trapezoidal rule in the x direction for the first sum term, and explicitly integrate the other terms:

$$\begin{aligned}
&\int_a^b \int_c^d f(x,y)dxdy = h^2 \sum_{i=0}^{n}\sum_{j=0}^{n} w_i w_j f(x_i,y_j) \\
&-\sum_{j=0}^{n}\sum_{\gamma=1}^{m} \frac{h^{2\gamma+1}B_{2\gamma}}{(2\gamma)!}\left[D_x^{2\gamma-1}w_j f(b,y_j) - D_x^{2\gamma-1}w_j f(a,y_j)\right] \\
&-h\sum_{j=0}^{n} h^{2m+1}\int_a^b \tilde{D}_x^{2m+1} w_j f(x,y_j)dx \\
&-\sum_{\mu=1}^{m}\frac{h^{2\mu}B_{2\mu}}{(2\mu)!}\left[\int_a^b \left[D_y^{2\mu-1}f(x,d) - D_y^{2\mu-1}f(x,c)\right]dx\right] \\
&-h^{2m+1}\int_c^d\int_a^b \tilde{D}_y^{2m+1} f(x,y)dxdy. \qquad (3.64)
\end{aligned}$$

In the second and third terms of this equation, the sum over j represents a trapezoidal rule integration, so we can apply the 1-d formula in each of these terms to give:

$$\begin{aligned}
&\int_a^b \int_c^d f(x,y)dxdy = h^2 \sum_{i=0}^{n}\sum_{j=0}^{n} w_i w_j f(x_i,y_j) \\
&-\sum_{\gamma=1}^{m}\frac{h^{2\gamma}B_{2\gamma}}{(2\gamma)!}\left[\int_c^d \left[D_x^{2\gamma-1}f(b,y) - D_x^{2\gamma-1}f(a,y)\right]dy\right. \\
&+\sum_{\mu=1}^{m}\frac{h^{2\mu}B_{2\mu}}{(2\mu)!}\left[D_x^{2\gamma-1}D_y^{2\mu-1}f(b,d) - D_x^{2\gamma-1}D_y^{2\mu-1}f(b,c)\right. \\
&\left.-D_x^{2\gamma-1}D_y^{2\mu-1}f(a,d) + D_x^{2\gamma-1}D_y^{2\mu-1}f(a,c)\right] \\
&\left.+h^{2m+1}\int_c^d \tilde{D}_y^{2m+1}\left[D_x^{2\gamma-1}f(b,y) - D_x^{2\gamma-1}f(a,y)\right]dy\right] \\
&-h^{2m+1}\int_a^b\int_c^d \tilde{D}_x^{2m+1} f(x,y)dydx \\
&-h^{2m+1}\int_a^b \sum_{\gamma=1}^{m}\frac{h^{2\gamma}B_{2\gamma}}{(2\gamma)!}\tilde{D}_x^{2m+1}\left[D_y^{2\gamma-1}f(x,d) - D_y^{2\gamma-1}f(x,c)\right]dx \\
&-h^{4m+2}\int_a^b\int_c^d \tilde{D}_x^{2m+1}\tilde{D}_y^{2m+1} f(x,y)dxdy \\
&-\sum_{\mu=1}^{m}\frac{h^{2\mu}B_{2\mu}}{(2\mu)!}\int_a^b \left[D_y^{2\mu-1}f(x,d) - D_y^{2\mu-1}f(x,c)\right]dx
\end{aligned}$$

$$-h^{2m+1}\int_c^d\int_a^b \tilde{D}_y^{2m+1} f(x,y)dxdy. \tag{3.65}$$

Regrouping the terms gives

$$\begin{aligned}
&h^2\sum_{i=0}^{n}\sum_{j=0}^{n} w_i w_j f(x_i,y_j) - \int_a^b\int_c^d f(x,y)dxdy = \\
&\sum_{\gamma=1}^{m}\frac{h^{2\gamma}B_{2\gamma}}{(2\gamma)!}\Bigg[\int_c^d\left[D_x^{2\gamma-1}f(b,y) - D_x^{2\gamma-1}f(a,y)\right]dy + \int_a^b\left[D_y^{2\gamma-1}f(x,d)\right. \\
&\left.-D_y^{2\gamma-1}f(x,c)\right]dx + \sum_{\mu=1}^{m}\frac{h^{2\mu}B_{2\mu}}{(2\mu)!}\left[D_x^{2\gamma-1}D_y^{2\mu-1}f(b,d)\right. \\
&\left.-D_x^{2\gamma-1}D_y^{2\mu-1}f(b,c) - D_x^{2\gamma-1}D_y^{2\mu-1}f(a,d) + D_x^{2\gamma-1}D_y^{2\mu-1}f(a,c)\right] \\
&+h^{2m+1}\int_a^b \tilde{D}_x^{2m-1}\left[D_y^{2\gamma+1}f(x,d) - D_y^{2\gamma-1}f(x,c)\right]dx \\
&+h^{2m+1}\int_c^d \tilde{D}_y^{2m+1}\left[D_x^{2\gamma-1}f(b,y) - D_x^{2\gamma-1}f(a,y)\right]dy\Bigg] \\
&+h^{2m+1}\int_a^b\int_c^d \tilde{D}_x^{2m+1}f(x,y)dydx + h^{2m+1}\int_c^d\int_a^b \tilde{D}_y^{2m+1}f(x,y)dxdy \\
&+h^{4m+2}\int_a^b\int_c^d \tilde{D}_x^{2m+1}\tilde{D}_y^{2m+1}f(x,y)dydx.
\end{aligned} \tag{3.66}$$

3.4.2 Calculation of the Error

In order to apply (3.66), we rewrite I as a sum of singular and non-singular integrals, $I = I_s + I_r$. We can then also write the error as $E = E_s + E_r$, where E_s is the error associated with the approximation to I_s and E_r is the error associated with I_r. We can write I as:

$$I = I_s + I_r = I_s + \sum_{i=1}^{4} I_{r_i}, \tag{3.67}$$

where

$$\begin{aligned}
I_s &= \int_{-\pi}^{\pi}\int_{-\pi}^{\pi} f(x,y)dxdy, \\
I_{r_1} &= \int_{-\infty}^{\pi}\int_{-\pi}^{\pi} f(x,y)dxdy, \\
I_{r_2} &= \int_{\pi}^{\infty}\int_{-\pi}^{\pi} f(x,y)dxdy,
\end{aligned} \tag{3.68}$$

$$\begin{aligned} I_{r_3} &= \int_{-\infty}^{\infty}\int_{-\infty}^{-\pi} f(x,y)dxdy, \\ I_{r_4} &= \int_{-\infty}^{\infty}\int_{\pi}^{\infty} f(x,y)dxdy. \end{aligned} \tag{3.69}$$

If we define Ω by

$$\Omega = I\!R^2 / \Big\{ [-\pi,\pi] \times [-\pi,\pi] \Big\}, \tag{3.70}$$

then we have

$$I_r = \int\int_{\Omega} f(x,y)dxdy. \tag{3.71}$$

We denote the trapezoidal rule approximation to I_{r_i} by I_{A_i} and the trapezoidal rule approximation to I_s by I_{A_s}. We define the respective errors as $E_{r_i} = I_{A_i} - I_{r_i}$ and $E_s = I_{A_s} - I_s$. Finally, we define $E_r = \sum_{i=1}^{4} E_{r_i}$.

3.4.3 Calculation of E_r

We can use the 2-d Euler Maclaurin formula to derive the following result for E_{r_1}.

$$\begin{aligned} E_{r_1} &= I_{A_1} - I_{r_1} = \\ &+ \sum_{\gamma=1}^{m} \frac{h^{2\gamma} B_{2\gamma}}{(2\gamma)!} \Bigg[\int_{-\pi}^{\pi} D_x^{2\gamma-1} f(-\pi,y)dy \\ &+ \int_{-\infty}^{-\pi} \left[D_y^{2\gamma-1} f(x,\pi) - D_y^{2\gamma-1} f(x,-\pi) \right] dx \\ &+ \sum_{\mu=1}^{m} \frac{h^{2\mu} B_{2\mu}}{(2\mu)!} [D_x^{2\gamma-1} D_y^{2\mu-1} f(-\pi,\pi) - D_x^{2\gamma-1} D_y^{2\mu-1} f(-\pi,-\pi)] \\ &+ h^{2m+1} \int_{-\infty}^{-\pi} \tilde{D}_x^{2m+1} \left[D_y^{2\gamma-1} f(x,\pi) - D_y^{2\gamma-1} f(x,-\pi) \right] dx \\ &+ h^{2m+1} \int_{-\pi}^{\pi} \tilde{D}_y^{2m+1} D_x^{2\gamma-1} f(-\pi,y)dy \Bigg] \\ &+ h^{2m+1} \int_{-\infty}^{-\pi}\int_{-\pi}^{\pi} \tilde{D}_x^{2m+1} f(x,y)dydx + h^{2m+1} \int_{-\pi}^{\pi}\int_{-\infty}^{-\pi} \tilde{D}_y^{2m+1} f(x,y)dxdy \\ &+ h^{4m+2} \int_{-\infty}^{-\pi}\int_{-\pi}^{\pi} \tilde{D}_x^{2m+1} \tilde{D}_y^{2m+1} f_{xy}^{2m+1,2m+1}(x,y)dydx. \end{aligned} \tag{3.72}$$

We can derive a similar result for each of the other three regular integrals. We then add all of the terms together to get:

$$
\begin{aligned}
E_r = \sum_{i=1}^{4}(I_{A_i} - I_{r_i}) = \sum_{\gamma=1}^{m} \frac{h^{2\gamma}B_{2\gamma}}{(2\gamma)!}\Bigg[\int_{-\pi}^{\pi}\left[D_x^{2\gamma-1}f(-\pi,y) - D_x^{2\gamma-1}f(\pi,y)\right]dy \\
+\int_{-\pi}^{\pi}\left[D_y^{2\gamma-1}f(x,-\pi) - D_y^{2\gamma-1}f(x,\pi)\right]dx + \sum_{\mu=1}^{m}\frac{h^{2\mu}B_{2\mu}}{(2\mu)!}[D_x^{2\gamma-1}D_y^{2\mu-1}f(-\pi,\pi) \\
-D_x^{2\gamma-1}D_y^{2\mu-1}f(-\pi,-\pi) - D_x^{2\gamma-1}D_y^{2\mu-1}f(\pi,\pi) + D_x^{2\gamma-1}D_y^{2\mu-1}f(\pi,-\pi)] \\
+h^{2m+1}\int_{-\pi}^{\pi}\tilde{D}_x^{2m+1}\left[D_y^{2\gamma-1}f(x,-\pi) - D_y^{2\gamma-1}f(x,\pi)\right]dx \qquad (3.73) \\
+h^{2m+1}\int_{-\pi}^{\pi}\tilde{D}_y^{2m+1}\left[D_x^{2\gamma-1}f(-\pi,y) - D_x^{2\gamma-1}f(\pi,y)\right]dy\Bigg] \\
+h^{2m+1}\int\int_{\Omega}\tilde{D}_x^{2m+1}f(x,y)dydx \\
+h^{2m+1}\int\int_{\Omega}\tilde{D}_y^{2m+1}f(x,y)dydx \\
+h^{4m+2}\int\int_{\Omega}\tilde{D}_x^{2m+1}\tilde{D}_y^{2m+1}f(x,y)dydx.
\end{aligned}
$$

We define $\partial\Omega$ to be the boundary of Ω as shown in Figure 3.4. The last three terms in the previous expression are integrals over Ω. We denote these terms by E_Ω. The rest of the terms in the expression are evaluated in terms of $\partial\Omega$. We denote these terms by $-E_{\partial\Omega}$ (we use the negative sign for convenience).

Thus we have

$$E_r = -E_{\partial\Omega} + E_\Omega, \qquad (3.74)$$

where

$$
\begin{aligned}
E_{\partial\Omega} = -\sum_{\gamma=1}^{m}\frac{h^{2\gamma}B_{2\gamma}}{(2\gamma)!}\Bigg[\int_{-\pi}^{\pi}\left[D_x^{2\gamma-1}f(-\pi,y) - D_x^{2\gamma-1}f(\pi,y)\right]dy \\
+\int_{-\pi}^{\pi}\left[D_y^{2\gamma-1}f(x,-\pi) - D_y^{2\gamma-1}f(x,\pi)\right]dx + \sum_{\mu=1}^{m}\frac{h^{2\mu}B_{2\mu}}{(2\mu)!}[D_x^{2\gamma-1}D_y^{2\mu-1}f(-\pi,\pi) \\
-D_x^{2\gamma-1}D_y^{2\mu-1}f(-\pi,-\pi) - D_x^{2\gamma-1}D_y^{2\mu-1}f(\pi,\pi) + D_x^{2\gamma-1}D_y^{2\mu-1}f(\pi,-\pi)] \\
+h^{2m+1}\int_{-\pi}^{\pi}\tilde{D}_x^{2m+1}\left[D_y^{2\gamma-1}f(x,-\pi) - D_y^{2\gamma-1}f(x,\pi)\right]dx \qquad (3.75) \\
+h^{2m+1}\int_{-\pi}^{\pi}\tilde{D}_y^{2m+1}\left[D_x^{2\gamma-1}f(-\pi,y) - D_x^{2\gamma-1}f(\pi,y)\right]dy\Bigg],
\end{aligned}
$$

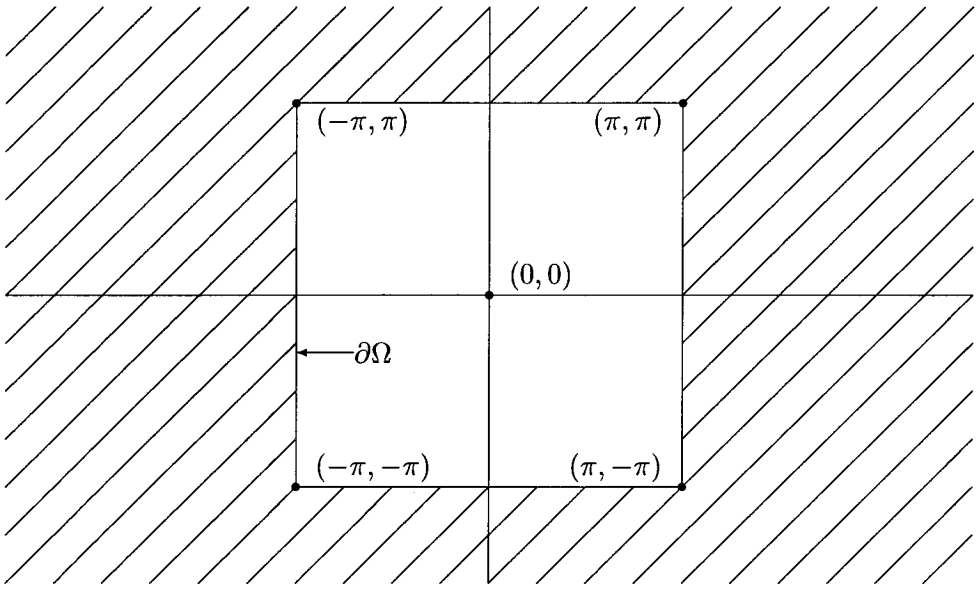

Figure 3.4: The shaded region is Ω and extends infinitely in both dimensions. The boundary is $\partial\Omega$.

and

$$\begin{aligned} E_\Omega &= h^{2m+1} \int\int_\Omega \tilde{D}_x^{2m+1} f(x,y) dy dx \\ &\quad + h^{2m+1} \int\int_\Omega \tilde{D}_y^{2m+1} f(x,y) dy dx \\ &\quad + h^{4m+2} \int\int_\Omega \tilde{D}_x^{2m+1} \tilde{D}_y^{2m+1} f(x,y) dy dx. \end{aligned} \tag{3.76}$$

We note that $E_{\partial\Omega}$ is a finite series in even powers of h. We comment later on E_Ω.

3.4.4 Calculation of E_s

We now consider the error that comes from applying the trapezoidal rule to the singular integral I_s. Because of the singularity, we must use the point vortex method. We can write

$$E_s = I_{A_s} - I_s = \sum_{i=-\frac{n}{2}}^{\frac{n}{2}} \sum_{\substack{j=-\frac{n}{2} \\ (i,j)\neq 0}}^{\frac{n}{2}} w_i w_j f(ih, jh) - \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} f(x,y) dx dy. \tag{3.77}$$

We want to express E_s in a form which allows the direct use of the Euler-Maclaurin formula. To do this, we break I_s and I_{A_s} into several integrals and sums and arrange them as follows:

$$E_s = E_1 + E_2 + E_3 + E_4 + E_5 \tag{3.78}$$

where

$$E_1 = h^2 \sum_{i=-\frac{n}{2}}^{-1} \sum_{j=-\frac{n}{2}}^{\frac{n}{2}} \tilde{w}_i \tilde{w}_j f(ih, jh) - \int_{-\pi}^{-h} \int_{-\pi}^{\pi} f(x,y)dydx, \tag{3.79}$$

$$E_2 = h^2 \sum_{i=1}^{\frac{n}{2}} \sum_{j=-\frac{n}{2}}^{\frac{n}{2}} \tilde{w}_i \tilde{w}_j f(ih, jh) - \int_{h}^{\pi} \int_{-\pi}^{\pi} f(x,y)dydx, \tag{3.80}$$

$$E_3 = h^2 \sum_{i=-1}^{1} \sum_{j=1}^{\frac{n}{2}} \tilde{w}_i \tilde{w}_j f(ih, jh) - \int_{-h}^{h} \int_{h}^{\pi} f(x,y)dydx, \tag{3.81}$$

$$E_4 = h^2 \sum_{i=-1}^{1} \sum_{j=-\frac{n}{2}}^{-1} \tilde{w}_i \tilde{w}_j f(ih, jh) - \int_{-h}^{h} \int_{-\pi}^{-h} f(x,y)dydx, \tag{3.82}$$

$$E_5 = \sum_{i=-1}^{1} \sum_{\substack{j=-1 \\ (i,j)\neq(0,0)}}^{1} \tilde{w}_i \tilde{w}_j f(ih, jh) - \int_{-h}^{h} \int_{-h}^{h} f(x,y)dydx. \tag{3.83}$$

The factor $\tilde{w}_i$ is defined by

$$\tilde{w}_i = \begin{cases} \frac{1}{2} & \text{if } i = i_{min} \text{ or } i = i_{max} \\ 1 & \text{otherwise} \end{cases} \tag{3.84}$$

where i_{max} and i_{min} are the respective upper and lower limits of the summation index i. For example, $i_{max} = -1$ and $i_{min} = -n/2$ in the equation defining E_1. Using the two dimensional Euler-Maclaurin formula, we can evaluate E_1 through E_4. After some manipulation, we arrive at the following result:

$$\begin{aligned} &E_1 + E_2 + E_3 + E_4 = \\ &-\sum_{\gamma=1}^{m} \frac{h^{2\gamma} B_{2\gamma}}{(2\gamma)!} \left[\int_{-\pi}^{\pi} \left[D_x^{2\gamma-1} f(-\pi, y) - D_x^{2\gamma-1} f(\pi, y) \right] dy \right. \\ &\left. + \int_{-\pi}^{\pi} \left[D_y^{2\gamma-1} f(x, -\pi) - D_y^{2\gamma-1} f(x, \pi) \right] dx + \sum_{\mu=1}^{m} \frac{h^{2\mu} B_{2\mu}}{(2\mu)!} [D_x^{2\gamma-1} D_y^{2\mu-1} f(-\pi, \pi) \right. \end{aligned}$$

$$
\begin{aligned}
&-D_x^{2\gamma-1}D_y^{2\mu-1}f(-\pi,-\pi)-D_x^{2\gamma-1}D_y^{2\mu-1}f(\pi,\pi)+D_x^{2\gamma-1}D_y^{2\mu-1}f(\pi,-\pi)]\\
&+h^{2m+1}\int_{-\pi}^{\pi}\tilde{D}_x^{2m+1}(x/h)\left[D_y^{2\gamma-1}f(x,-\pi)-D_y^{2\gamma-1}f(x,\pi)\right]dx \qquad (3.85)\\
&+h^{2m+1}\int_{-\pi}^{\pi}\tilde{D}_y^{2m+1}\left[D_x^{2\gamma-1}f(-\pi,y)-D_x^{2\gamma-1}f(\pi,y)\right]dy\Bigg]\\
&-\sum_{\gamma=1}^{m}\frac{h^{2\gamma}B_{2\gamma}}{(2\gamma)!}\Bigg[\int_{-h}^{h}\left[D_x^{2\gamma-1}f(h,y)-D_x^{2\gamma-1}f(-h,y)\right]dy\\
&+\int_{-h}^{h}\left[D_y^{2\gamma-1}f(x,h)-D_y^{2\gamma-1}f(x,-h)\right]dx+\sum_{\mu=1}^{m}\frac{h^{2\mu}B_{2\mu}}{(2\mu!)}\Big[D_x^{2\gamma-1}D_y^{2\mu-1}f(h,h)\\
&-D_x^{2\gamma-1}D_y^{2\mu-1}f(-h,h)-D_x^{2\gamma-1}D_y^{2\mu-1}f(h,-h)+D_x^{2\gamma-1}D_y^{2\mu-1}f(-h,-h)\Big]\\
&+h^{2m+1}\int_{-h}^{h}\tilde{D}_x^{2m+1}\Big[D_y^{2\gamma-1}f(x,h)-D_y^{2\gamma-1}f(x,-h)\Big]dx\\
&+h^{2m+1}\int_{-h}^{h}\tilde{D}_y^{2m+1}\Big[D_x^{2\gamma-1}f(h,y)-D_x^{2\gamma-1}f(-h,y)\Big]dy\Bigg]\\
&+h^{2m+1}\int\int_{\Lambda}\tilde{D}_x^{2m+1}f(x,y)dxdy+h^{2m+1}\int\int_{\Lambda}\tilde{D}_y^{2m+1}f(x,y)dxdy\\
&+h^{4m+2}\int\int_{\Lambda}\tilde{D}_x^{2m+1}\tilde{D}_y^{2m+1}f(x,y)dxdy,
\end{aligned}
$$

where

$$\Lambda=\Big\{[-\pi,\pi]\times[-\pi,\pi]\Big\}/\Big\{[-h,h]\times[-h,h]\Big\}. \qquad (3.86)$$

We define the inner boundary of Λ to be $\partial\Lambda$ and note that the outer boundary of Λ is $\partial\Omega$, the inner boundary of Ω, as shown in Figure 3.5. We define $E_{\partial\Lambda}$ and E_{Λ} by

$$
\begin{aligned}
E_{\partial\Lambda}=&\sum_{\gamma=1}^{m}\frac{h^{2\gamma}B_{2\gamma}}{(2\gamma)!}\Bigg[\int_{-h}^{h}\left[D_x^{2\gamma-1}f(h,y)-D_x^{2\gamma-1}f(-h,y)\right]dy\\
&+\int_{-h}^{h}\left[D_y^{2\gamma-1}f(x,h)-D_y^{2\gamma-1}f(x,-h)\right]dx+\sum_{\mu=1}^{m}\frac{h^{2\mu}B_{2\mu}}{(2\mu!)}\Big[D_x^{2\gamma-1}D_y^{2\mu-1}f(h,h)\\
&-D_x^{2\gamma-1}D_y^{2\mu-1}f(-h,h)-D_x^{2\gamma-1}D_y^{2\mu-1}f(h,-h)+D_x^{2\gamma-1}D_y^{2\mu-1}f(-h,-h)\Big]\\
&+h^{2m+1}\int_{-h}^{h}\tilde{D}_x^{2m+1}\Big[D_y^{2\gamma-1}f(x,h)-D_y^{2\gamma-1}f(x,-h)\Big]dx \qquad (3.87)\\
&+h^{2m+1}\int_{-h}^{h}\tilde{D}_y^{2m+1}\Big[D_x^{2\gamma-1}f(h,y)-D_y^{2\gamma-1}f(-h,y)\Big]dy\Bigg]
\end{aligned}
$$

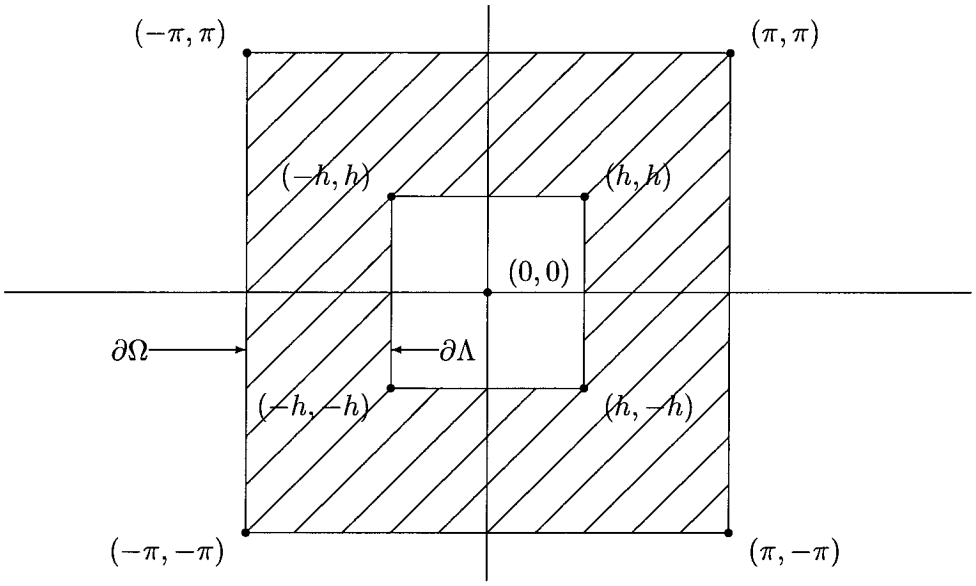

Figure 3.5: The shaded region is Λ. The inner boundary is $\partial\Lambda$. The outer boundary is $\partial\Omega$.

and

$$\begin{aligned} E_\Lambda &= h^{2m+1} \int\int_\Lambda \tilde{D}_x^{2m+1} f(x,y) dxdy + h^{2m+1} \int\int_\Lambda \tilde{D}_y^{2m+1} f(x,y) dxdy \\ &\quad + h^{4m+2} \int\int_\Lambda \tilde{D}_x^{2m+1} \tilde{D}_y^{2m+1} f(x,y) dxdy. \end{aligned}$$

Using the definitions of $E_{\partial\Lambda}$, E_Λ, and $E_{\partial\Omega}$ (recall (3.76)), we see that we can write

$$E_1 + E_2 + E_3 + E_4 = E_{\partial\Omega} - E_{\partial\Lambda} + E_\Lambda. \tag{3.88}$$

We can now write a simpler expression for E_s:

$$\begin{aligned} E_s &= E_1 + E_2 + E_3 + E_4 + E_5 \\ &= E_{\partial\Omega} - E_{\partial\Lambda} + E_\Lambda + E_5. \end{aligned} \tag{3.89}$$

3.4.5 Formula for E

Putting together our definitions of E_r and E_s, we can now write an expression for E:

$$\begin{aligned} E &= E_s + E_r \\ &= -E_{\partial\Omega} + E_\Omega + E_{\partial\Omega} + E_\Lambda - E_{\partial\Lambda} + E_5 \\ &= E_\Omega + E_\Lambda - E_{\partial\Lambda} + E_5 \\ &= E_\Omega + E_L, \end{aligned} \tag{3.90}$$

where

$$E_L = E_\Lambda - E_{\partial\Lambda} + E_5. \tag{3.91}$$

We show that E_L is a series in odd powers of h. It is significant that $E_{\partial\Omega}$ does not appear in E, as this means that there are no even powers of h in E. We show later that

$$E_\Omega = o(h^{2m}). \tag{3.92}$$

To simplify the computation of E_L, we use the notation $E_L(f)$ to mean that we plug f into the expression for E_L and we observe that E_L is linear. That is, if a and b are constants and f and g are functions of x and y, then $E_L(af + bg) = aE_L(f) + bE_L(g)$. Using this notation, the definition of E_L (3.91), and the definition of Λ (3.86), we have

$$\begin{aligned} E_L(f) &= h^{2m+1}\int\int_\Lambda \tilde{D}_x^{2m+1} f(x,y)dxdy + h^{2m+1}\int\int_\Lambda D_y^{2m+1} f(x,y)dxdy \\ &+h^{4m+2}\int\int_\Lambda \tilde{D}_x^{2m+1}\tilde{D}_y^{2m+1} f(x,y)dxdy \\ &-\sum_{\gamma=1}^{m}\frac{h^{2\gamma}B_{2\gamma}}{(2\gamma)!}\left[\int_{-h}^{h}\left[D_x^{2\gamma-1}f(h,y) - D_x^{2\gamma-1}f(-h,y)\right]dy\right. \\ &+\int_{-h}^{h}\left[D_y^{2\gamma-1}f(x,h) - D_y^{2\gamma-1}f(x,-h)\right]dx + \sum_{\mu=1}^{m}\frac{h^{2\mu}B_{2\mu}}{(2\mu!)}\left[D_x^{2\gamma-1}D_y^{2\mu-1}f(h,h)\right. \\ &\left.\left.-D_x^{2\gamma-1}D_y^{2\mu-1}f(-h,h) - D_x^{2\gamma-1}D_y^{2\mu-1}f(h,-h) + D_x^{2\gamma-1}D_y^{2\mu-1}f(-h,-h)\right]\right] \end{aligned} \tag{3.93}$$

$$-h^{2m+1}\int_{-h}^{h}\tilde{D}_x^{2m+1}\Big[D_y^{2\gamma-1}f(x,h)-D_y^{2\gamma-1}f(x,-h)\Big]dx$$
$$-h^{2m+1}\int_{-h}^{h}\tilde{D}_y^{2m+1}\Big[D_x^{2\gamma-1}f(h,y)-D_y^{2\gamma-1}f(-h,y)\Big]dy\Big]$$
$$+\sum_{i=-1}^{1}\sum_{\substack{j=-1\\(i,j)\neq(0,0)}}^{1}\tilde{w}_i\tilde{w}_j f(ih,jh)-\int_{-h}^{h}\int_{-h}^{h}f(x,y)dydx.$$

Recall that the integrand of I is defined by

$$f(x,y)=K(x,y)p(x,y). \tag{3.94}$$

To demonstrate our reasoning, we assume that $p(x,y)=p(x)=xg(x)$ and that $\eta(x,y)=0$. We comment later on the general case. Using these simplifications, we have

$$f(x,y)=\frac{xg(x)}{(x^2+y^2)^{3/2}}. \tag{3.95}$$

Using Taylor's theorem, we can write

$$f(x,y)=\frac{x}{(x^2+y^2)^{3/2}}\left[\sum_{k=0}^{2m}\frac{x^k}{k!}g^{(k)}(0)+\int_0^x\frac{(x-\tau)^{2m}}{(2m)!}g^{(2m+1)}(\tau)d\tau\right]. \tag{3.96}$$

If we define

$$C_k(x,y)=\frac{g^{(k)}(0)x^{k+1}}{k!(x^2+y^2)^{3/2}} \tag{3.97}$$

and

$$R(x,y)=\int_0^x\frac{x(x-\tau)^{(2m)}}{(2m)!(x^2+y^2)^{3/2}}g^{2m+1)}(\tau)d\tau, \tag{3.98}$$

then

$$f(x,y)=\sum_{k=0}^{2m}C_k(x,y)+R(x,y). \tag{3.99}$$

Since C_k is odd when k is even, we plug C_k into the expression for E_L and get

$$E_L(C_k)=0 \text{ for } k \text{ even}. \tag{3.100}$$

Using this, and the fact that E_L is linear, we can write

$$\begin{aligned} E_L(f(x,y)) &= E_L\left(\sum_{k=0}^{2m} C_k(x,y)\right) + E_L\left(R(x,y)\right) \\ &= \sum_{k=1}^{m} E_L\left(C_{2k-1}(x,y)\right) + E_L\left(R(x,y)\right) \qquad (3.101) \\ &= E_{LC} + E_{LR}. \end{aligned}$$

Using the definition of E_L we have

$$\begin{aligned} E_{LC} = 4\sum_{k=1}^{m}\Bigg\{ &-\sum_{\gamma=1}^{m}\frac{h^{2\gamma}B_{2\gamma}}{(2\gamma)!}\Bigg[\int_0^h D_x^{2\gamma-1}C_{2k-1}(h,y)dy \\ &+\int_0^h D_y^{2\gamma-1}C_{2k-1}(x,h)dx + \sum_{\mu=1}^{m}\frac{h^{2\mu}B_{2\mu}}{(2\mu!)}D_x^{2\gamma-1}D_y^{2\mu-1}C_{2k-1}(h,h) \\ &+h^{2m+1}\int_0^h \tilde{D}_x^{2m+1}D_y^{2\gamma-1}C_{2k-1}(x,h)dx \\ &+h^{2m+1}\int_0^h D_x^{2\gamma-1}\tilde{D}_y^{2m+1}C_{2k-1}(h,y)dy\Bigg] \\ &+h^{2m+1}\int_h^\pi\int_0^\pi \tilde{D}_x^{2m+1}C_{2k-1}(x,y)dydx + h^{2m+1}\int_0^h\int_h^\pi \tilde{D}_x^{2m+1}C_{2k-1}(x,y)dydx \\ &+h^{2m+1}\int_h^\pi\int_0^\pi \tilde{D}_y^{2m+1}C_{2k-1}(x,y)dxdy + h^{2m+1}\int_0^h\int_h^\pi \tilde{D}_y^{2m+1}C_{2k-1}(x,y)dxdy \\ &+h^{4m+2}\int_h^\pi\int_0^\pi \tilde{D}_x^{2m+1}\tilde{D}_y^{2m+1}C_{2k-1}(x,y)dydx \\ &+h^{4m+2}\int_0^h\int_h^\pi \tilde{D}_x^{2m+1}\tilde{D}_y^{2m+1}C_{2k-1}(x,y)dydx \\ &-\int_0^h\int_0^h C_{2k-1}(x,y) + \frac{h^2}{4}\left[C_{2k-1}(h,0) + C_{2k-1}(h,h)\right]\Bigg\}. \end{aligned}$$

Here we have taken advantage of the symmetry of $C_{2k-1}(x,y)$ with respect to x and y to reduce the domain of integration to the first quadrant. Now, let $x = hx'$ and $y = hy'$, substitute, and then drop the primes. The result is:

$$\begin{aligned} E_{LC} = \sum_{k=1}^{m} 4h^{2k-1}\Bigg\{ &-\sum_{\gamma=1}^{m}\frac{B_{2\gamma}}{(2\gamma)!}\Bigg[\int_0^1 D_x^{2\gamma-1}C_{2k-1}(1,y)dy \\ &+\int_0^1 D_x^{2\gamma-1}C_{2k-1}(x,1)dx + \sum_{\mu=1}^{m}\frac{B_{2\mu}}{(2\mu!)}D_x^{2\gamma-1}D_y^{2\mu-1}C_{2k-1}(1,1) \end{aligned}$$

$$
\begin{aligned}
&+\int_0^1 \hat{D}_x^{2m+1} D_y^{2\gamma-1} C_{2k-1}(x,1)dx + \int_0^1 D_x^{2\gamma-1} \hat{D}_y^{2m+1} C_{2k-1}(1,y)dy\Bigg] \\
&+\int_1^{n/2}\int_0^{n/2} \hat{D}_x^{2m+1} C_{2k-1}(x,y)dydx + \int_0^1\int_1^{n/2} \hat{D}_x^{2m+1} C_{2k-1}(x,y)dydx \\
&+\int_1^{n/2}\int_0^{n/2} \hat{D}_y^{2m+1} C_{2k-1}(x,y)dxdy + \int_0^1\int_1^{n/2} \hat{D}_y^{2m+1} C_{2k-1}(x,y)dxdy \\
&+\int_1^{n/2}\int_0^{n/2} \hat{D}_x^{2m+1}\hat{D}_y^{2m+1} C_{2k-1}(x,y)dydx \\
&+\int_0^1\int_1^{n/2} \hat{D}_x^{2m+1}\hat{D}_y^{2m+1} C_{2k-1}(x,y)dydx \qquad (3.102) \\
&-\int_0^1\int_0^1 C_{2k-1}(x,y)dxdy + \frac{1}{4}\left[C_{2k-1}(1,0)+C_{2k-1}(1,1)\right]\Bigg\},
\end{aligned}
$$

where $\hat{D}$ is defined by

$$
\hat{D}_x^j \equiv P_j(x)\frac{\partial^j}{\partial x^j}. \qquad (3.103)
$$

We have written this as a series in odd powers of h. However the coefficients are not independent of h because some of the double integrals have $n/2(=\pi/h)$ in the limits of integration. We rewrite E_{LC} as follows:

$$
\begin{aligned}
E_{LC} &= \sum_{k=1}^{m} 4h^{2k-1}\Bigg\{ \\
&-\sum_{\gamma=1}^{m}\frac{B_{2\gamma}}{(2\gamma)!}\Bigg[\int_0^1 D_x^{2\gamma-1}C_{2k-1}(1,y)dy + \int_0^1 D_y^{2\gamma-1}C_{2k-1}(x,1)dx \\
&+\sum_{\mu=1}^{m}\frac{B_{2\mu}}{(2\mu!)}D_x^{2\gamma-1}D_y^{2\mu-1}C_{2k-1}(1,1) \\
&+\int_0^1 \hat{D}_x^{2m+1}D_y^{2\gamma-1}C_{2k-1}(x,1)dx + \int_0^1 D_x^{2\gamma-1}\hat{D}_y^{2m+1}C_{2k-1}(1,y)dy\Bigg] \qquad (3.104) \\
&+\int_1^{\infty}\int_0^{\infty} \hat{D}_x^{2m+1} C_{2k-1}(x,y)dydx + \int_0^1\int_1^{\infty} \hat{D}_x^{2m+1} C_{2k-1}(x,y)dydx \\
&+\int_1^{\infty}\int_0^{\infty} \hat{D}_y^{2m+1} C_{2k-1}(x,y)dxdy + \int_0^1\int_1^{\infty} \hat{D}_y^{2m+1} C_{2k-1}(x,y)dxdy \\
&+\int_1^{\infty}\int_0^{\infty} \hat{D}_x^{2m+1}\hat{D}_y^{2m+1} C_{2k-1}(x,y)dydx \\
&+\int_0^1\int_1^{\infty} \hat{D}_x^{2m+1}\hat{D}_y^{2m+1} C_{2k-1}(x,y)dydx \\
&-\int_0^1\int_0^1 C_{2k-1}(x,y)dxdy + \frac{1}{4}\left[C_{2k-1}(1,0)+C_{2k-1}(1,1)\right]\Bigg\} + E_{LE}
\end{aligned}
$$

where E_{LE} is defined by:

$$E_{LE} = -\sum_{k=1}^{m} 4h^{2k-1}\Bigg\{ \int_{n/2}^{\infty}\int_{0}^{\infty} \hat{D}_x^{2m+1} C_{2k-1}(x,y)dydx + 4\int_{0}^{n/2}\int_{n/2}^{\infty} \hat{D}_x^{2m+1} C_{2k-1}(x,y)dydx + \int_{n/2}^{\infty}\int_{0}^{\infty} \hat{D}_y^{2m+1} C_{2k-1}(x,y)dxdy + 4\int_{0}^{n/2}\int_{n/2}^{\infty} \hat{D}_y^{2m+1} C_{2k-1}(x,y)dxdy + \int_{n/2}^{\infty}\int_{0}^{\infty} \hat{D}_x^{2m+1}\hat{D}_y^{2m+1} C_{2k-1}(x,y)dydx + \int_{0}^{n/2}\int_{n/2}^{\infty} \hat{D}_x^{2m+1} D_y^{2m+1} C_{2k-1}(x,y)dydx \Bigg\}.$$

Because the range of integration of the double integrals is infinite, we must discuss the issue of convergence. We examine

$$\frac{\partial^L}{\partial x^Q y^{L-Q}} C_{2k-1}(x,y) \tag{3.105}$$

where $L \geq 2m+1$ and $Q \geq 0$. This is equivalent to examining

$$\frac{\partial^L}{\partial x^Q y^{L-Q}} \frac{x^P}{(x^2+y^2)^{3/2}} \tag{3.106}$$

for values of $0 < P \leq 2m$. If we perform the differentiation and convert to polar coordinates, all of the terms involve r^{P-L-3}. Because of the restrictions on P and L, $P - L - 3 < -3$. Because the radial factor has an r dependence which goes as r^{-4}, all of the infinite integrals converge.

If we change back to the original variables (that is let $x = x'/h$, $y = y'/h$, substitute, then drop the primes), then we have

$$E_{LE} = -\sum_{k=1}^{m} 4h^{2m+1}\Bigg\{ \int_{\pi}^{\infty}\int_{0}^{\infty} \tilde{D}_x^{2m+1} C_{2k-1}(x,y)dydx + \int_{0}^{\pi}\int_{\pi}^{\infty} \tilde{D}_x^{2m+1} C_{2k-1}(x,y)dydx + \int_{\pi}^{\infty}\int_{0}^{\infty} \tilde{D}_y^{2m+1} C_{2k-1}(x,y)dxdy + \int_{0}^{\pi}\int_{\pi}^{\infty} \tilde{D}_y^{2m+1} C_{2k-1}(x,y)dxdy + h^{2m+1}\int_{\pi}^{\infty}\int_{0}^{\infty} \tilde{D}_x^{2m+1}\tilde{D}_y^{2m+1} C_{2k-1}(x,y)dydx$$

$$+h^{2m+1}\int_0^{\pi}\int_{\pi}^{\infty}\tilde{D}_x^{2m+1}\tilde{D}_y^{2m+1}C_{2k-1}(x,y)dydx\Bigg\}. \tag{3.107}$$

We know that the periodic Bernoulli functions are bounded. We define A as follows: $|P_m(x/h)| \leq A$ for all x. We now have a bound on E_{LE}:

$$\begin{aligned}
E_{LE} \leq 4Ah^{2m+1}\sum_{k=1}^{m}\Bigg\{ & \\
&\int_{\pi}^{\infty}\int_{0}^{\infty}\left|D_x^{2m+1}C_{2k-1}(x,y)\right|dydx+\int_{0}^{\pi}\int_{\pi}^{\infty}\left|D_x^{2m+1}C_{2k-1}(x,y)\right|dydx \\
&+\int_{\pi}^{\infty}\int_{0}^{\infty}\left|D_y^{2m+1}C_{2k-1}(x,y)\right|dxdy+\int_{0}^{\pi}\int_{\pi}^{\infty}\left|D_y^{2m+1}C_{2k-1}(x,y)\right|dxdy \\
&+Ah^{2m+1}\int_{\pi}^{\infty}\int_{0}^{\infty}\left|D_x^{2m+1}D_y^{2m+1}C_{2k-1}(x,y)\right|dydx \\
&+Ah^{2m+1}\int_{0}^{\pi}\int_{\pi}^{\infty}\left|D_x^{2m+1}D_y^{2m+1}C_{2k-1}(x,y)\right|dydx\Bigg\}.
\end{aligned} \tag{3.108}$$

We have already shown that these integrals converge. Therefore, the coefficient of h^{2m+1} in E_{LE} is bounded as h approaches 0. We conclude from this that $E_{LE} = o(h^{2m})$. We now express E_{LC} in the general form

$$\begin{aligned}
E_{LC} &= \sum_{k=1}^{m} D_{2k-1}h^{2k-1}+E_{LE} \\
&= \sum_{k=1}^{m} D_{2k-1}h^{2k-1}+o(h^{2m}),
\end{aligned} \tag{3.109}$$

where D_{2k-1} is defined by (3.53). The analysis of E_{LR} is similar, and is left until later in this section. We show there that

$$E_{LR} = O(h^{2m-1}). \tag{3.110}$$

Using (3.92), (3.109), and (3.110), we can express E as follows:

$$E = E_{\Omega}+E_L = E_{\Omega}+E_{LC}+E_{LR} = \sum_{k=1}^{m-1} D_{2k-1}h^{2k-1}+O(h^{2m-1}). \tag{3.111}$$

3.4.6 The General Case

We now briefly discuss the more general case of (3.94) where $\eta(x,y) \neq 0$ and p is a function of both x and y. In this case we have

$$f(x,y) = \frac{p(x,y)}{(x^2+y^2+(\eta(0,0)-\eta(x,y))^2)^{3/2}}. \tag{3.112}$$

We want to calculate and analyze E as in the previous section. First, we note that the analysis of E_Ω does not change in this case. To analyze E_L we rewrite $f(x,y)$ as in Section 4.1. That is, we write

$$f(x,y) = C_0 + C_1 + C_2 + C_3 + \cdots + R(x,y), \tag{3.113}$$

where C_n is homogeneous of order $n-2$ in (x,y). We are interested in calculating

$$E_L(f) = E_L(C_0) + E_L(C_1) + \cdots + E_L(R). \tag{3.114}$$

For n even, C_n is odd in (x,y) so we have

$$E_L(C_n) = 0 \text{ for } n \text{ even}. \tag{3.115}$$

The analysis of the nonzero terms in (3.114) is similar to the analysis of the previous section in terms of the basic method, but the details are significantly more complicated. We omit the full proof, and state the result:

$$E_L(f) = D_1 h + D_3 h^3 + \cdots + O(h^N), \tag{3.116}$$

where we compute D_k using a formula similar to (3.57). The primary difference is that the symmetry of the problem leading to (3.57) allows a simpler formula than in this more general case.

3.4.7 Analysis of E_Ω

We have

$$\begin{aligned} E_\Omega &= h^{2m+1}\int\int_\Omega P_{2m+1}(x/h)D_x^{2m+1}f(x,y)dydx \\ &+ h^{2m+1}\int\int_\Omega P_{2m+1}(y/h)D_y^{2m+1}f(x,y)dydx \\ &+ h^{4m+2}\int\int_\Omega P_{2m+1}(x/h)P_{2m+1}(y/h)D_x^{2m+1}D_y^{2m+1}f(x,y)dydx. \end{aligned} \tag{3.117}$$

We consider the first term of E_Ω:

$$h^{2m+1}\int\int_\Omega P_{2m+1}(x/h)D_x^{2m+1}f(x,y)dxdy. \tag{3.118}$$

We first define a new region Ω':

$$\Omega' = \left\{(x,y)|\sqrt{x^2+y^2} \geq 2\pi\right\}. \tag{3.119}$$

The region Ω' is the portion of Ω which lies outside a circle of radius 2π. (This will lead to a more pleasant geometry.) Because the region $\Omega - \Omega'$ is finite, we have

$$h^{2m+1}\int\int_{\Omega-\Omega'} P_{2m+1}(x/h)D_x^{2m+1}f(x,y)dxdy = O(h^{2m+1}). \tag{3.120}$$

This follows because the integrand has no singularities on the domain of integration, and because $P_{2m+1}(x/h)$ is bounded.

We now have to consider the integral over Ω'. We look first at a special case:

$$f(x,y) = \frac{xg(x)}{(x^2+y^2)^{3/2}} \tag{3.121}$$

where we have taken g to be a function of x alone, and $\eta = 0$. We have the following:

$$D_x^{2m+1}f(x,y) = \frac{xg^{2m+1}(x)}{(x^2+y^2)^{3/2}} + R(x,y) \tag{3.122}$$

where it can be seen by examination that $R(x,y)$ consists of a collection of terms

that vary as r^{-p} where $p \geq 3$ and $r^2 = x^2 + y^2$. We know that $|P_{2m+1}(x/h)| \leq C$ where C is a constant, so

$$h^{2m+1} \left| \int \int_{\Omega'} P_{2m+1}(x/h) R(x,y) dxdy \right| \leq h^{2m+1} C \int \int_{\Omega'} |R(x,y)| \, dxdy. \quad (3.123)$$

The integral can be seen to converge, given the way the integrand depends on the radial factor r. Thus, the coefficient of h^{2m+1} is bounded as h approaches 0. However, it is possible that the coefficient vanishes as h approaches 0. Specifically, the coefficient may be $o(1)$ or smaller in magnitude. Thus, we conclude that this term is $o(h^{2m})$. We note that this analysis serves only to place a bound on the magnitude of the term rather than to explicitly state the magnitude. (This does not affect the form of the expansion since m is arbitrary. To explicitly determine the magnitude of the next term in the expansion, we must choose a larger value of m in the Euler-Maclaurin formula.) We next define $q(x) = g^{2m+1}(x)$. We write $q(x)$ as the sum of its odd and even constituents: $q(x) = q_e(x) + q_o(x)$ where q_e is even in x and q_o is odd in x. Since $P_{2m+1}(x/h)$ is odd in x, we can write

$$\begin{aligned} h^{2m+1} \int \int_{\Omega'} P_{2m+1}(x/h) \frac{xq(x,y)}{(x^2+y^2)^{3/2}} dxdy = \quad (3.124) \\ h^{2m+1} \int \int_{\Omega'} P_{2m+1}(x/h) \frac{x(q_e(x) + q_o(x))}{(x^2+y^2)^{3/2}} \\ = \quad h^{2m+1} \int \int_{\Omega'} P_{2m+1}(x/h) \frac{xq_e(x)}{(x^2+y^2)^{3/2}} \end{aligned}$$

since

$$h^{2m+1} \int \int_{\Omega'} P_{2m+1}(x/h) \frac{xq_o(x)}{(x^2+y^2)^{3/2}} = 0, \quad (3.125)$$

(the integrand is odd in x).

For $m > 1$, the function $P_{2m+1}(x/h)q_e(x)$ is continuous for all x, odd in x, and satisfies the condition $P_{2m+1}(0) = P_{2m+1}(1) = 0$. Thus, we express $P_{2m+1}(x/h)q_e(x)$ as a Fourier (sine) series which converges absolutely and uniformly:

$$P_{2m+1}(x/h)q_e(x) = \sum_{i=1}^{\infty} A_i \sin(ix). \quad (3.126)$$

Because the series converges uniformly, we may integrate each term separately:

$$\begin{aligned}\int\int_{\Omega'} P_{2m+1}(x/h)\frac{xq_e(x)}{(x^2+y^2)^{3/2}}dxdy &= \int\int_{\Omega'} \frac{x}{(x^2+y^2)}\sum_{i=1}^{\infty} A_i \sin(ix) \\ &= \sum_{i=1}^{\infty}\int\int_{\Omega'} \frac{A_i x\sin(ix)}{(x^2+y^2)^{3/2}}dxdy. \quad (3.127)\end{aligned}$$

Next, we change to polar coordinates. If we define

$$\mathrm{Si}(x) = \int_0^x \frac{\sin(x)}{x}dx, \quad (3.128)$$

then we have

$$h^{2m+1}\sum_{i=1}^{\infty}\int\int_{\Omega'} \frac{A_i x\sin(ix)}{(x^2+y^2)^{3/2}}dxdy \quad (3.129)$$

$$= h^{2m+1}\sum_{i=1}^{\infty}\int_0^{2\pi}\int_{2\pi}^{\infty} \frac{A_i\cos(\theta)\sin(ir\cos(\theta))}{r}drd\theta \quad (3.130)$$

$$= h^{2m+1}\sum_{i=1}^{\infty} A_i\int_0^{2\pi}\left[\frac{\pi}{2}|\cos(\theta)| - \mathrm{Si}(2i\pi\cos(\theta))\cos(\theta)\right]d\theta. \quad (3.131)$$

It can be shown that the integrand of the latter integral is bounded by a constant which is determined by the maximum value of $\mathrm{Si}(x)$ (which occurs at $x = \pi$):

$$\left|\frac{\pi}{2}|\cos(\theta)| - \mathrm{Si}(2n\pi\cos(\theta))\cos(\theta)\right| \leq \frac{\pi}{2} + \mathrm{Si}(\pi). \quad (3.132)$$

If we define

$$C_{max} = 2\pi\left[\frac{\pi}{2} + \mathrm{Si}(\pi)\right] \quad (3.133)$$

then we can write

$$\left|\int\int_{\Omega'} \frac{A_i x\sin(ix)}{(x^2+y^2)^{3/2}}dxdy\right| \leq C_{max}\,|A_i| \quad (3.134)$$

for all i. We know from the theory of Fourier series that $\sum_{i=1}^{\infty}|A_i|$ is a convergent series. Hence, the series $C_{max}\sum_{i=1}^{\infty}|A_i|$ also converges. This in turn implies that

$$\sum_{i=1}^{\infty}\left|\int\int_{\Omega'} \frac{A_i x\sin(ix)}{(x^2+y^2)^{3/2}}dxdy\right| \quad (3.135)$$

converges. From this we conclude that

$$h^{2m+1}\int\int_{\Omega'} P_{2m+1}(x/h)\frac{xq_e(x)}{(x^2+y^2)^{3/2}}dxdy \tag{3.136}$$

$$= h^{2m+1}\sum_{i=1}^{\infty}\int\int_{\Omega'}\frac{A_i x\sin(ix)}{(x^2+y^2)^{3/2}}dxdy = O(h^{2m+1}). \tag{3.137}$$

Now we treat the more general problem. That is we want to show that

$$h^{2m+1}\int\int_{\Omega'} P_{2m+1}(x/h)D_x^{2m+1}f(x,y)dxdy = o(h^{2m+1}) \tag{3.138}$$

where

$$f(x,y) = \frac{xg(x,y)}{(x^2+y^2+(\eta(0,0)-\eta(x,y))^2)^{3/2}}. \tag{3.139}$$

We rewrite $f(x,y)$ in the following way:

$$f(x,y) = f_1(x,y) + f_2(x,y) \tag{3.140}$$

where

$$f_1(x,y) = \frac{xg(x,y)}{(x^2+y^2)^{3/2}} \tag{3.141}$$

$$f_2(x,y) = \frac{xg(x,y)}{(x^2+y^2+(\eta(0,0)-\eta(x,y))^2)^{3/2}} - \frac{xg(x,y)}{(x^2+y^2)^{3/2}}. \tag{3.142}$$

We can rewrite $f_2(x,y)$ in the following form:

$$f_2(x,y) = \frac{xg(x,y)\left[A^{3/2}-(A+B)^{3/2}\right]}{A^{3/2}(A+B)^{3/2}} \tag{3.143}$$

where

$$A = x^2+y^2 \tag{3.144}$$

$$B = (\eta(0,0)-\eta(x,y))^2. \tag{3.145}$$

If we rationalize the numerator, we have:

$$f_2(x,y) = \frac{xg(x,y)\left[A^3 - (A+B)^3\right]}{A^{3/2}(A+B)^{3/2}\left[A^{3/2} + (A+B)^{3/2}\right]}. \tag{3.146}$$

Expanding the numerator gives

$$f_2(x,y) = \frac{xg(x,y)\left[-3A^2B - 3AB^2 - B^3\right]}{A^{3/2}(A+B)^{3/2}\left[A^{3/2} + (A+B)^{3/2}\right]}. \tag{3.147}$$

We now observe that in terms of the radial variable r, A varies as r^2 and B is bounded. Thus, in polar coordinates, f_2 varies as r^{-4}. Using this, we can write

$$D_x^{2m+1} f(x,y) = \frac{xD_x^{2m+1} g(x,y)}{(x^2+y^2)^{3/2}} + R(x,y) \tag{3.148}$$

where, as in the simpler case, $R(x,y)$ consists of a collection of terms varying as r^{-p}, $p \geq 3$ in the radial factor r. ($R(x,y)$ includes all of the terms in $D_x^{2m+1} f_2(x,y)$.)

The analysis proceeds essentially as before from here. The integral involving $R(x,y)$ is $o(h^{2m})$. We define $q(x,y) = D_x^{2m+1} g(x,y)$. We can express $q(x,y)$ as a sum of terms

$$q(x,y) = q_{ee}(x,y) + q_{eo}(x,y) + q_{oe}(x,y) + q_{oo}(x,y) \tag{3.149}$$

where, for example q_{eo} is even in x and odd in y. We can then use symmetry to write

$$\int\int_{\Omega'} P_{2m+1}(x/h)\frac{xq(x,y)}{(x^2+y^2)^{3/2}}dxdy = \int\int_{\Omega'} P_{2m+1}(x/h)\frac{xq_{ee}(x,y)}{(x^2+y^2)^{3/2}}dxdy. \tag{3.150}$$

Next, we express $P_{2m+1}(x/h)q_{ee}(x,y)$ as a double Fourier series:

$$P_{2m+1}(x/h)q_{ee}(x,y) = \sum_{i=1}^{\infty}\sum_{j=1}^{\infty} A_{ij}\sin(ix)\cos(jy) \tag{3.151}$$

$$= \frac{1}{2}\sum_{i=1}^{\infty}\sum_{j=1}^{\infty} A_{ij}\left[\sin(ix+jy) + \sin(ix-jy)\right]. \tag{3.152}$$

From here, we use the uniform convergence of the series to enable us to interchange summation and integration. We use the fact that $\mathrm{Si}(x)$ is bounded to obtain a convergent series representation for

$$\int\int_{\Omega'} P_{2m+1}(x/h)\frac{xq(x,y)}{(x^2+y^2)^{3/2}}dxdy. \tag{3.153}$$

As in the first case, we again conclude that

$$h^{2m+1}\int\int_{\Omega'} P_{2m+1}(x/h)\frac{xq(x,y)}{(x^2+y^2)^{3/2}}dxdy = O(h^{2m+1}) \tag{3.154}$$

and hence,

$$h^{2m+1}\int\int_{\Omega} P_{2m+1}(x/h)\frac{xq(x,y)}{(x^2+y^2)^{3/2}}dxdy = O(h^{2m+1}). \tag{3.155}$$

The condition that the Fourier series representations used are uniformly convergent relies on the premise that g and the derivatives of g are smooth.

We have considered one of the terms in E_Ω. The analysis of the other terms is analogous. We conclude that

$$E_\Omega = o(h^{2m}). \tag{3.156}$$

3.4.8 Calculation of E_{LR}

We now consider E_{LR}. In order to analyze E_{LR}, we need the following results, which are derived from Leibniz's rule for differentiation of an integral:

$$\frac{\partial^k}{\partial x^k}\int_0^x \frac{x(x-\tau)^m}{m!(x^2+y^2)^{3/2}}f^{(m+1)}(\tau)d\tau = \int_0^x \frac{\partial^k}{\partial x^k}\left[\frac{x(x-\tau)^m}{m!(x^2+y^2)^{3/2}}\right]f^{(m+1)}(\tau)d\tau \tag{3.157}$$

for $k < m+1$,

$$\frac{\partial^k}{\partial x^k}\int_0^x \frac{x(x-\tau)^m}{m!(x^2+y^2)^{3/2}}f^{(m+1)}(\tau)d\tau =$$

$$\int_0^x \frac{\partial^{m+1}}{\partial x^{m+1}}\left[\frac{x(x-\tau)^m}{m!(x^2+y^2)^{3/2}}\right] f^{(m+1)}(\tau)d\tau + \frac{xf^{(m+1)}(x)}{(x^2+y^2)^{3/2}} \tag{3.158}$$

for $k = m+1$.

We can express E_{LR} as follows:

$$\begin{aligned}
E_{LR} = &-4\sum_{\gamma=1}^{m}\frac{h^{2\gamma}B_{2\gamma}}{(2\gamma)!}\left[\int_0^h D_x^{2\gamma-1}R(h,y)dy + \int_0^h D_y^{2\gamma-1}R(x,h)dx\right.\\
&+\sum_{\mu=1}^{m}\frac{h^{2\mu}B_{2\mu}}{(2\mu)!}D_x^{2\gamma-1}D_y^{2\mu-1}R(h,h) + h^{2m+1}\int_0^h \tilde{D}_x^{2m+1}D_y^{2\gamma-1}R(x,h)dx\\
&\left.+h^{2m+1}\int_0^h D_y^{2m+1}D_x^{2\gamma-1}R(h,y)dy\right]\\
&+4h^{2m+1}\int_h^\pi\int_0^\pi \tilde{D}_x^{2m+1}R(x,y)dydx + 4h^{2m+1}\int_0^h\int_h^\pi \tilde{D}_x^{2m+1}R(x,y)dydx\\
&+4h^{2m+1}\int_h^\pi\int_0^\pi \tilde{D}_y^{2m+1}R(x,y)dxdy + 4h^{2m+1}\int_0^h\int_h^\pi \tilde{D}_y^{2m+1}R(x,y)dxdy\\
&+4h^{4m+2}\int_h^\pi\int_0^\pi \tilde{D}_x^{2m+1}\tilde{D}_y^{2m+1}R(x,y)dydx\\
&+4h^{4m+2}\int_0^h\int_h^\pi \tilde{D}_x^{2m+1}\tilde{D}_y^{2m+1}R(x,y)dydx\\
&-\int_{-h}^h\int_{-h}^h R(x,y)dydx + R(h,0) + R(h,h)
\end{aligned}$$

where

$$R(x,y) = \int_0^x \frac{x(x-\tau)^{(2m)}}{(2m)!(x^2+y^2)^{3/2}}g^{2m+1)}(\tau)d\tau. \tag{3.159}$$

We start with the last terms in E_{LR}.

$$\begin{aligned}
R(h,0) &= \int_0^h \frac{h(h-\tau)^{2m}}{(2m)!h^3}g^{(2m+1)}(\tau)d\tau\\
&= g^{2m+1}(\xi)\int_0^h \frac{h(h-\tau)^{2m}}{(2m)!h^3}d\tau\\
&= \frac{g^{2m+1}(\xi)}{(2m+1)!}h^{2m-1}\\
&= O(h^{2m-1})
\end{aligned} \tag{3.160}$$

by the mean value theorem for integrals. This assumes that the derivatives of $g(x)$ are bounded. In an analogous manner, it can be shown that $R(h,h) = O(h^{2m-1})$.

Next we consider the integral

$$\begin{aligned}\int_{-h}^{h}\int_{-h}^{h} R(x,y)dxdy &= h^{2m+1}\int_{-1}^{1}\int_{-1}^{1} dxdy \int_0^x \frac{x(x-\tau)^{2m}}{(2m)!(x^2+y^2)^{3/2}} g^{(2m+1)}(h\tau)d\tau \\ &= h^{2m+1}\int_{-1}^{1}\int_{-1}^{1} \frac{x^{2m+2}g^{(2m+1)}(\xi)}{(2m+1)!(x^2+y^2)^{3/2}}dxdy \qquad (3.161) \\ &= O(h^{2m+1}),\end{aligned}$$

where, again, we have applied the mean value theorem of integrals. Next, we want to consider those terms in E_{LR} which involve a single integral. For example we can consider the first term in E_{LR}

$$h^{2\gamma}\int_0^h D_x^{2\gamma-1}R(h,y)dy.$$

If we make the change of variables $x = hx'$, $y = hy'$ then (dropping the primes) we get:

$$\begin{aligned}&h^{2\gamma}\int_0^h D_x^{2\gamma-1}R(h,y)dy = \\ &\quad h^{2m+1}\int_0^1 \frac{\partial^{2\gamma-1}}{\partial x^{2\gamma-1}}\left[\int_0^x \frac{x(x-\tau)^{2m}}{(2m)!(x^2+y^2)^{3/2}} g^{(2m+1)}(h\tau)d\tau\right]_{x=1} dy. \qquad (3.162)\end{aligned}$$

Since $\gamma \leq m$ we can take the differential operator inside the integral (using (3.157), perform the differentiation, and then apply the mean value theorem to verify that this integral is $O(h^{2m+1})$. Again, we have relied on the fact that $g(x,y)$ is bounded in all its derivatives. It is completely analogous to show that the terms

$$h^{2\gamma}h^{2\mu}D_x^{2\gamma-1}D_y^{2\mu-1}R(h,h)$$

are $O(h^{2m+1})$.

We have only to consider the terms involving the double integrals. To show the general approach, we consider the following term from E_{LR}:

$$h^{2m+1}\int_h^\pi \int_0^\pi \tilde{D}_x^{2m+1}R(x,y)dydx.$$

We know that

$$D_x^{2m+1}R(x,y) = \int_0^x \frac{\partial^{2m+1}}{\partial x^{2m+1}} \frac{x(x-\tau)^{2m}}{(2m)!(x^2+y^2)^{3/2}} g^{(2m+1)}(\tau)d\tau + \frac{xg^{2m+1}(x)}{(x^2+y^2)^{3/2}}. \tag{3.163}$$

Applying the differential operator and using the mean value theorem gives a series of terms, which, when converted to polar coordinates is of order r^{-2}. Thus, we can bound the integral as follows:

$$h^{2m+1}\left|\int_h^\pi \int_0^\pi \tilde{D}_x^{2m+1}R(x,y)dydx\right| \leq Ch^{2m+1}\int_{q_1}^{q_2}\int_{hs(\theta)}^{\pi s(\theta)} r^{-1}drd\theta = O(h^{2m}) \tag{3.164}$$

where C is some constant determined by the derivatives of g and the periodic Bernoulli functions.

We have now considered all of the terms of E_{LR} and conclude that

$$E_{LR} = O(h^{2m-1}). \tag{3.165}$$

Chapter 4

Implementation

In this chapter we present the numerical implementation of the point vortex approximation. We discuss efficient schemes for computing the boundary integrals using a direct approach and using two different fast multipole methods.

4.1 Direct Method - Ewald Summation

The primary computational difficulty in computing the velocity fields and the velocity potential is the efficient calculation of the point vortex approximation. This difficulty arises because the domain of integration is infinite and the Green's function is very slowly decaying. To demonstrate an efficient direct approach, we return to the formula for computing the third component of velocity w at the origin:

$$w(0,0) = -\frac{1}{4\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\frac{x\gamma_x(x,y)+y\gamma_y(x,y)}{[x^2+y^2+(\eta(0,0)-\eta(x,y))^2]^{3/2}}dxdy, \tag{4.1}$$

where γ_x, γ_y, and η are periodic with period 2π in x and y. We take advantage of the periodicity in order to redefine $w(0,0)$ in such a way that the domain of integration is finite. We then have

$$\begin{aligned} w(0,0) = &-\frac{1}{4\pi}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}dxdy \times \\ &\sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty}\frac{\gamma_x(x,y)(x-2n\pi)+\gamma_y(x,y)(y-2m\pi)}{[(x-2n\pi)^2+(y-2m\pi)^2+(\eta(0,0)-\eta(x,y))^2]^{3/2}}. \end{aligned} \tag{4.2}$$

In order to evaluate this lattice sum, we have applied the Ewald summation technique as outlined by Baker, Meiron, and Orszag [2] which converts the derivatives of the periodic Green's function into rapidly converging sums of error functions. Because this approach directly computes all particle interactions, the operation count is $O(N^4)$ per time step where N is the number of mesh points in each dimension.

4.2 Fast Multipole Methods

Because of the asymptotic operation count, direct methods are impractical for large scale problems. In order to make large scale computations feasible, it is necessary to use an algorithm with a lower asymptotic operation count. One general approach to reducing the operation count is the fast multipole method. The reduction in operation count is achieved by computing local particle interactions directly and combining distant particle interactions into multipole expansions. The calculation of the velocity and velocity potential is analogous to the calculation of the electrostatic forces due to a collection of charged particles. The fluid particles on the surface of the wave are the "charged particles" and the "charge distribution" is related to the shape of the interface and the dipole and vortex sheet strength. Because the calculations are analogous but simpler, we initially describe the multipole methods in terms of electrostatics, bearing in mind the analogy to the water wave problem.

While a detailed description of multipole methods is beyond the scope of this work, we give a sketch of the approach in order to describe our implementation. We consider a cube of side 2π centered at the origin. We refer to this cube as the center cell. The charged particles are distributed throughout the center cell. The center cell is subdivided into subcells, and multipole expansions for each of the subcells are computed about the subcell origin. The electrostatic force at any particular particle is a combination of direct interactions with particles in nearby cells and multipole interactions with cells further away. The error introduced by the multipole expansions is controlled by setting restrictions on how far apart subcells must be before permitting multipole expansions to describe the interactions between par-

ticles in the subcells. Various algorithms combine multipole expansions in different ways to achieve improved efficiency. The way in which multipole calculations are combined to compute the distant interactions determines the computational cost of the method. We have implemented algorithms with computational requirements which are $O(N^3)$ and $O(N^2)$. We describe the algorithms later.

We define a number of the terms generally used in describing the multipole expansions. We follow the definitions given by Greengard in [8] and [4]. We begin by describing the multipole expansion for an electrostatic field and field potential. The free space Green's function can be defined in terms of a multipole expansion by

$$\frac{1}{|\mathbf{x}-\mathbf{x}'|} = \sum_{n=0}^{\infty} \sum_{n=-m}^{m} \frac{\rho^n}{r^{n+1}} Y_N^{-m}(\alpha,\beta) Y_n^m(\theta,\phi) \tag{4.3}$$

where we assume that $r > \rho$ and where the the spherical coordinates of $\mathbf{x}$ and $\mathbf{x}'$ are taken to be (r,θ,ϕ) and (ρ,α,β), respectively. Using this definition, the electrostatic far field contribution of a collection of source terms contained in a box centered at the origin is given by

$$\Phi(\mathbf{x}) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} M_n^m \frac{Y_n^m(\theta,\phi)}{r^{n+1}} \tag{4.4}$$

where (r,θ,ϕ) are the spherical coordinates of $\mathbf{x}$ and $Y_n^m(\theta,\phi)$ are spherical harmonics defined by

$$Y_n^m(\theta,\phi) = \sqrt{\frac{(n-|m|)!}{(n+|m|)!}} P_n^{|m|}(\cos(\theta)) e^{im\phi}. \tag{4.5}$$

Here P_n^m denotes the associated Legendre function of degree n order m. Note that this definition varies by a constant factor from the usual definition of spherical harmonics. The multipole coefficients M_n^m are defined by

$$M_n^m = \sum_{k=1}^{K} q_k \rho_k^n Y_n^{-m}(\alpha_k,\beta_k) \tag{4.6}$$

where K is the number of point sources, $(\rho_k,\alpha_k,\beta_k)$ is the location of the kth source and q_k is its strength. The electrostatic field is defined to be $-\nabla\Phi$. The fact that the Green's function and its derivatives can be expressed in terms of a multipole

expansion is the basis for our use of multipole methods.

4.2.1 Periodicity

Before we describe the different multipole algorithms which we have applied, we discuss the problem of periodicity. Our problem can be viewed as a cube of side 2π centered at the origin which is extended periodically in two dimensions. We refer to those cells adjacent to the center cell as its neighbors. Because of the periodicity of the problem, the multipole coefficients of the center cell's neighbors are identical to those of the center cell. Hence, we only need to compute multipole expansions for the center cell. The effect of all cells beyond the neighbors of the center cell can be expressed as a local expansion about the origin in terms of lattice sums and the multipole moments of the center cell. We are not able to include the neighbors of the center cell in the lattice sum because we must compute some direct interactions between the center cell and its neighbors.

The lattice sums of interest are defined by

$$L_n^m = \sum_{p \in \Lambda_2} \frac{Y_n^m(\pi/2, \phi_p)}{r_p^{n+1}}, \qquad n \geq 1, m = 0, \ldots, n \tag{4.7}$$

where Λ_2 is the set of integer lattice points $\{(k_1, k_2, 0), |k_i| \geq 1, \quad i = 1, 2\}$. The efficient calculation of these sums is due to a method by Berman and Greengard [4]. Our application is slightly different from their approach in that we are summing spherical harmonics over a two-dimensional lattice rather than a three-dimensional lattice.

The terms L_1^m, for $m = 0, 1$ are conditionally convergent, and an appropriate value must be chosen. For the case $m = 0$ the Associated Legendre polynomial $P_1^0(0)$ is identically zero. For $m = 1$, we arrange the lattice sum using the symmetry of the problem in such a way that the sum is zero. This is analogous to the general case of m odd.

We define the multipole expansion of a collection of charges in the center cell to

be

$$\Phi = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} M_n^m \frac{Y_n^m(\theta, \phi)}{r^{n+1}}. \tag{4.8}$$

For a point $\mathbf{x}$ in the center cell with spherical coordinates (r, θ, ϕ), the far field contribution due to points in image cubes indexed by elements of Λ_2 can be expressed as a local expansion about the origin

$$\Psi(\mathbf{x}) = \sum_{j=0}^{\infty} \sum_{k=-j}^{j} N_j^k Y_j^k(\theta, \phi) r^j \tag{4.9}$$

where

$$N_j^k = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \frac{M_n^m \cdot (-1)^n \cdot J_k^m \cdot A_n^m \cdot A_j^k}{A_{j+n}^{m-k}} \cdot \sum_{p \in \Lambda_2} L_{j+n}^{m-k}. \tag{4.10}$$

The values of A_n^m and J_n^m are defined by

$$A_n^m = \frac{(-1)^n}{\sqrt{(n-m)!(n+m)!}} \tag{4.11}$$

and

$$J_n^m = \begin{cases} (-1)^{min(|n|,|m|)}, & \text{if} \quad nm > 0; \\ 1 & \text{otherwise.} \end{cases} \tag{4.12}$$

We now define the following sets of integer lattice points:

$$M = \{(k_1, k_2, 0) | |k_i| \leq 1, i = 1, 2\} \tag{4.13}$$

$$N = \{(k_1, k_2, 0) | |k_i| \leq 2, i = 1, 2\} \tag{4.14}$$

$$\Gamma = \{(k_1, k_2, 0) | |k_i| \leq 7, i = 1, 2\} - N \tag{4.15}$$

$$\tilde{\Lambda}_2 = \Lambda_2 - N. \tag{4.16}$$

We define the spherical coordinates of the lattice points to be $(\rho_p, \pi/2, \beta_p)$. If we define $\tilde{L}_n^m$ by

$$\tilde{L}_n^m = \sum_{p \in \tilde{\Lambda}_2} \frac{Y_n^m(\pi/2, \beta_p)}{\rho^{p+1}}, \tag{4.17}$$

then $\tilde{L}_n^m$ is computed by the recursion relation given by Berman and Greengard:

$$\tilde{L}_n^m - \sum_{j=n}^{\infty} \sum_{\substack{k= \\ m-(j-n)}}^{m-(j+n)} (-1)^{j+n} \frac{\tilde{L}_j^k}{3^{j+1}} \frac{J_m^{m-k} \cdot A_n^m \cdot A_{j-n}^{k-m}}{A_j^k} Q(j-n, m-k) = R(n,m) \quad (4.18)$$

where

$$Q(n,m) = \sum_{p \in M} Y_n^m(\pi/2, \beta_p) \rho_p^n \quad (4.19)$$

$$R(n,m) = \sum_{p \in \Gamma} \frac{Y_n^m(\pi/2, \beta_p)}{\rho_p^{n+1}}. \quad (4.20)$$

To compute $\tilde{L}_n^m$, we truncate the infinite sum in Equation 4.18. Berman and Greengard show that 100 terms is sufficient to guarantee fifteen digits of accuracy. To compute L_n^m we simply add

$$L_n^m = \tilde{L}_n^m + \sum_{p \in N-M} \frac{Y_n^m(\pi/2, \beta_p)}{\rho_p^{n+1}}. \quad (4.21)$$

In Table 4.1 we show the (nonzero) values of $\tilde{L}_n^m$ and L_n^m for $n \leq 16$. Because these lattice sums are independent of the particles and the particle locations, we compute the values before the calculation and then read in the coefficients for the calculation. Using the lattice sums, we can compute the contribution due to particles outside the center cell's neighbors using a simple local expansion. In practice we have found that the computational cost of adding the far field contributions represented by L_n^m is negligible compared to the computation of the multipole moments. The calculation of the lattice sums required approximately two minutes on a workstation using double precision arithmetic. For our calculations, we assumed that the center cell was of side length 2π. To convert L_n^m to the appropriate value for a cube of side length l requires multiplying L_n^m by a factor of $(2\pi/l)^{n+1}$.

n	m	$\tilde{L}_n^m$	L_n^m
2	0	-0.448919120391818E-02	-0.729571660209102E-02
4	0	0.355928447585322E-05	0.146724420257081E-04
4	4	0.968645018532362E-06	0.432381519979532E-05
6	0	-0.560223368175452E-08	-0.562312798832289E-07
6	4	-0.188769737998386E-08	-0.204646553329348E-07
8	0	0.109209474749826E-10	0.259631641534164E-09
8	4	0.449767384406115E-11	0.116255477660096E-09
8	8	-0.713854109869944E-13	0.966463765496442E-10
10	0	-0.236138345943999E-13	-0.130656679309689E-11
10	4	-0.112686043598392E-13	-0.686176344963998E-12
10	8	-0.889010941655920E-15	-0.521932356090937E-12
12	0	0.541982661085200E-16	0.690752418161287E-14
12	4	0.288676103361732E-16	0.410848727175016E-14
12	8	0.442299001094658E-17	0.325285579757940E-14
12	12	0.153504678753166E-16	0.110723533036711E-13
14	0	-0.129213929754837E-18	-0.377242571638190E-16
14	4	-0.748862385345919E-19	-0.247758239150935E-16
14	8	-0.165089305497221E-19	-0.205554746071491E-16
14	12	-0.333223327477159E-19	-0.462301034285150E-16
16	0	0.316172750372947E-21	0.210934440477164E-18
16	4	0.195889593444058E-21	0.150066186979569E-18
16	8	0.553877721975190E-22	0.129613718377271E-18
16	12	0.882519798892363E-22	0.238414258321495E-18
16	16	0.394283992843946E-21	0.348193153763391E-18

Table 4.1: The nonzero lattice sums $\tilde{L}_n^m$ and L_n^m for $n \leq 16$.

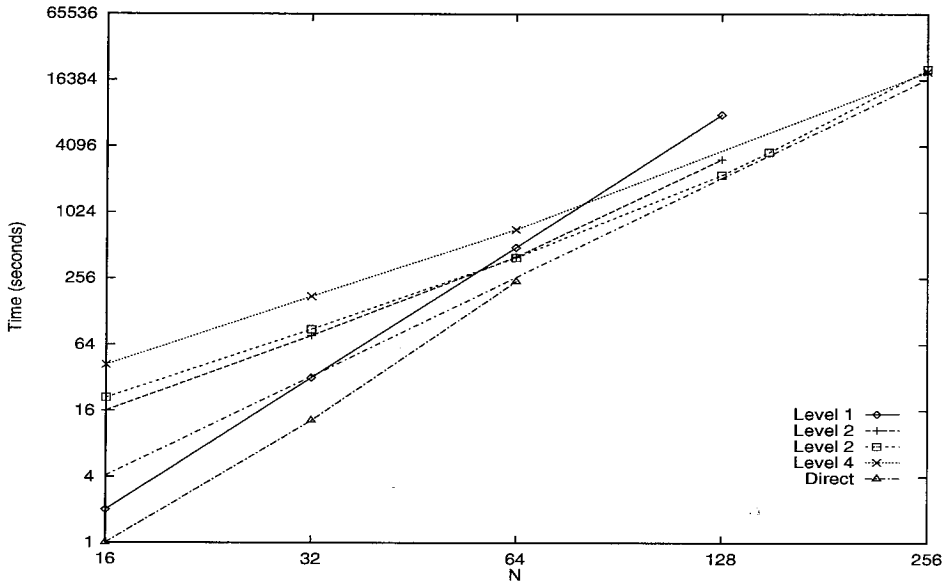

Figure 4.1: A comparison of the direct algorithm using Ewald summation and the Buttke algorithm for various levels of subdivision. For large N the Buttke algorithm is approximately $O(N^3)$. The curve without points is proportional to N^3, showing that for $N > 128$, the Buttke algorithm is $O(N^3)$.

4.2.2 Buttke Algorithm

One fast multipole method is described by Buttke [6]. In this algorithm, the center cell is subdivided and multipole coefficients are computed for each subcell. Interactions between cells are computed directly for neighboring cells. Interactions for separated cells are computed using the multipole expansion. This formulation has the advantage of not requiring a tree structure and the storage of numerous multipole expansions. The computational cost per time step is $O(N^3)$. We show in Figure 4.1 some sample timings for computing the velocity potential for the Ewald summation method and for the Buttke algorithm. For larger values of N, the asymptotic computational cost begins to be apparent. We did not develop a complete water wave method based on the Buttke algorithm because we determined, based on these timings, that the Fast Multipole algorithm would yield better results even for moderately small values of N. In addition, our original intention was to do large scale problems, for which the Fast Multipole algorithm would most likely be better suited.

4.2.3 Fast Multipole Algorithm

The Fast Multipole algorithm uses a tree structure of multipole coefficients to maximize efficiency. The center cell is divided into subcells, then each subcell is subdivided. This process continues to some level of subdivision. Multipole coefficients are computed at the lowest level on the tree structure (meaning the last level of subdivision) and translated to multipole coefficients of the parent cells. Beginning at the first level of subdivision, the multipole coefficients in a subcell are translated to local expansion coefficients about the center of each well-separated cell on the same level. When this process is complete, the local expansion coefficients of each cell are translated to the centers of its subcells, and the process repeats at the next level. After this process has reached the last level of subdivision, all of the distant interactions for a given cell are contained in the local expansion coefficients for that cell. The direct interactions are then computed. Thus, for each cell in the tree structure, it is necessary to store multipole coefficients and local expansion coefficients. While the implementation is quite complicated, the asymptotic computational requirement is $O(n^2)$.

PMTA

Because of the complexity of developing a full multipole algorithm, we used a source code developed at the Electrical Engineering Department at Duke University. The code is the Parallel Multipole Treecode Algorithm (PMTA) version 4.1. The code is described in a users manual [9] which accompanies the code and in several technical reports [22], [15], [16]. This code was originally developed to compute electrostatic potential and force calculations. Although the code was readily adapted to the water wave problem, we made a number of significant modifications to the code to maximize efficiency for our particular application. In this section we detail the most significant additions and modifications which we have made to the original source code.

Force Calculations

Given an initial charge distribution, the PMTA program computes the electrostatic potential and field using

$$\Phi(\mathbf{x_i}) = \sum_j q_j G(\mathbf{x_i} - \mathbf{x_j}) \tag{4.22}$$

$$\mathbf{E}(\mathbf{x_i}) = -\sum_j q_j \nabla_{\mathbf{x}} G(\mathbf{x_i} - \mathbf{x_j}) \tag{4.23}$$

where q_i is the charge on the ith particle and $\mathbf{x_i}$ is the position of the ith particle. For our purposes, the computation of Φ is irrelevant. The computation of the velocity and velocity potential take the form

$$A(\mathbf{x_i}) = \sum_j B_j G_x(\mathbf{x_i} - \mathbf{x_j}) + C_j G_y(\mathbf{x_i} - \mathbf{x_j}) + D_j G_z(\mathbf{x_i} - \mathbf{x_j}) \tag{4.24}$$

where (G_x, G_y, G_z) are the components of $\nabla_{\mathbf{x}} G$. A represents either the velocity potential or one component of velocity and B, C, and D are the appropriate coefficients as determined by the governing equations. For example, to compute the velocity potential we use the following definitions of B, C, and D with the surface normal defined by $\hat{\mathbf{n}} = (n_1, n_2, n_3)$:

$$B(\mathbf{x}) = -\frac{\mu(\mathbf{x}) n_1(\mathbf{x})}{4\pi} h^2 \tag{4.25}$$

$$C(\mathbf{x}) = -\frac{\mu(\mathbf{x}) n_2(\mathbf{x})}{4\pi} h^2 \tag{4.26}$$

$$D(\mathbf{x}) = -\frac{\mu(\mathbf{x}) n_3(\mathbf{x})}{4\pi} h^2. \tag{4.27}$$

Hence, the calculations of velocity and velocity potential are analogous to the electrostatic field calculations with the charge distribution being defined in terms of the normal to the surface, the vortex sheet strength, and the dipole sheet strength. For a single computation of Equation 4.24, there are three separate sets of multipole coefficients required because we have three different "charge density" functions. We rewrote the PMTA routines to simultaneously compute the necessary multipole co-

efficients. While this increases the memory requirement, only one subroutine call is necessary to compute a sum like Equation 4.24.

Adaptivity

The original PMTA source code was based on the assumption that the initial particle distribution would be roughly uniform in the center cell. Because a surface wave problem has a very nonuniform particle distribution, PMTA is quite inefficient for these types of problems. This is true both in terms of computations and memory storage. The original code allocated significant amounts of memory that were never accessed, and computed multipole and local expansion coefficients in boxes where there were no particles. The elimination of superfluous coefficient calculations was relatively simple and straightforward, involving only the addition of conditional tests for particles before performing calculations for a particular subcell.

The modifications involving memory use were considerably more difficult. The memory management was made completely dynamic. At the conclusion of each time step, our code checks each subcell for particles. When particles enter a subcell which previously had no particles, memory is allocated for the multipole coefficients. When all particles have left a subcell, the memory for storing coefficients is freed. In our implementation, the dynamic allocation of memory was not costly in terms of computational time required, and it significantly reduced the memory requirement. This is particularly important for large scale problems at high resolution, since the memory requirement is proportional to the square of the number of multipole terms multiplied by the number of particles on the interface.

Periodicity and Parallelism

The PMTA source code was originally adapted for use either on a workstation or a shared memory architecture, specifically the KSR architecture. Because the KSR implementation of shared memory loops is somewhat different from that used for SGI and CRAY machines, we were required to add some additional loops to the overall structure of the program. The addition of the appropriate compiler

directives to complete the parallelism was straightforward and simply mirrored the KSR directives in the code (with one exception, which we discuss forthwith).

The original PMTA source code allowed the computation of periodic particle arrays. However, we found their periodic computation to be very inefficient computationally. We have detailed our approach above. The lattice sums we used do not account for the contribution due to neighbors of the center cell. This is because some direct interactions must be computed between the center cell and its neighbors. The PMTA implementation computes the interactions due to neighbors of the center cells efficiently, but does not implement the algorithm in parallel. This is a significant consideration, since there are 8 neighbors of the center cell for which interactions must be computed. We modified the original code to first tabulate the necessary interactions which should occur between subcells of the neighbors and subcells of the center cell before making any computations. The computations were then ordered in a loop which could be efficiently parallelized.

Our experience with smaller runs (e.g. 64×64 mesh points) indicates that the parallelism is extremely efficient. Our experiments showed nearly ideal speed up for small numbers of processors. We have developed parallel codes which can be run on both CRAY and SGI shared memory machines. An important consideration for the CRAY implementation is that this code does not vectorize efficiently.

Robust Calculations

We also made several minor changes to the code to make the calculations more robust. The calculation of forces relies on computing spatial derivatives of the local expansions. For a local expansion of the form

$$\Phi(\mathbf{x}) = \sum_{n=0}^{p} \sum_{m=-n}^{n} N_n^m Y_n^m(\theta, \phi) r^n \tag{4.28}$$

where N_n^m are the local expansion coefficients, we need the derivatives of the terms $Y_n^m(\theta, \phi) r^n$.

The PMTA implementation uses the following form for computing the x deriva-

tives for $n > 0$ and m positive:

$$\frac{\partial}{\partial x}\left(Y_n^m(\theta,\phi)r^n\right) = \sqrt{\frac{(n-m)!}{(n+m)!}}e^{im\phi}r^{n-1} \times \Bigg[\tag{4.29}$$
$$\left(\frac{\cos(\theta)}{(1-\cos^2(\theta))}P_n^m(cos(\theta)) + \frac{1}{\sqrt{1-\cos^2(\theta)}}P_n^{m+1}(\cos(\theta))\right)\cos(\theta)\sin(\phi)\sin(\theta)$$
$$-P_n^m(cos(\theta))\frac{im\sin(\phi)}{\sin(\theta)} - n\cos(\phi)\sin(\theta)\Bigg].$$

The y and z derivatives are computed similarly. The difficulty arises when $\theta = \pm\pi$. While it can be shown that these derivatives are well defined for $\theta = \pm\pi$, the PMTA implementation does not robustly compute the derivatives for these values of θ.

We have implemented a different method for computing the spatial derivatives of the spherical harmonics which avoids division by zero. This approach is based on the definition of spherical harmonics given by Greengard [8].

Using the definitions given in Greengard, we have for $m = 0$,

$$\frac{Y_n^0(\theta,\phi)}{r^{n+1}} = A_n^0 \cdot \frac{\partial^n}{\partial z^n}\left(\frac{1}{r}\right). \tag{4.30}$$

For $m > 0$ we have:

$$\frac{Y_n^m(\theta,\phi)}{r^{n+1}} = A_n^m \cdot \left(\frac{\partial}{\partial x} + i\frac{\partial}{\partial y}\right)^m\left(\frac{\partial}{\partial z}\right)^{n-m}\left(\frac{1}{r}\right) \tag{4.31}$$

and

$$\frac{Y_n^{-m}(\theta,\phi)}{r^{n+1}} = A_n^{-m} \cdot \left(\frac{\partial}{\partial x} - i\frac{\partial}{\partial y}\right)^m\left(\frac{\partial}{\partial z}\right)^{n-m}\left(\frac{1}{r}\right), \tag{4.32}$$

where

$$A_n^m = \frac{(-1)^n}{\sqrt{(n-m)!\cdot(n+m)!}}. \tag{4.33}$$

The operators ∂_+, ∂_-, and ∂_z are defined by

$$\partial_\pm = \frac{\partial}{\partial x} \pm i \cdot \frac{\partial}{\partial y} \tag{4.34}$$

$$\partial z = \frac{\partial}{\partial z}. \tag{4.35}$$

If ϕ is harmonic, then

$$\partial_+\partial_-(\phi) = -\partial_z^2(\phi). \tag{4.36}$$

Therefore,

$$\begin{aligned}
\partial_+\Big(\frac{Y_n^0(\theta,\phi)}{r^{n+1}}\Big) &= A_n^0 \cdot \partial_+\Big(\frac{\partial}{\partial z}\Big)^n\Big(\frac{1}{r}\Big) && (4.37)\\
&= A_n^0 \cdot \partial_+\Big(\frac{\partial}{\partial z}\Big)^{(n+1)-1}\Big(\frac{1}{r}\Big) && (4.38)\\
&= \frac{A_n^0}{A_{n+1}^1} \cdot \frac{Y_{n+1}^1(\theta,\phi)}{r^{n+2}} && (4.39)
\end{aligned}$$

and

$$\begin{aligned}
\partial_-\Big(\frac{Y_n^0(\theta,\phi)}{r^{n+1}}\Big) &= A_n^0 \cdot \partial_-\frac{\partial}{\partial z}^n\Big(\frac{1}{r}\Big) && (4.40)\\
&= A_n^0 \cdot \partial_-\frac{\partial}{\partial z}^{(n+1)-1}\Big(\frac{1}{r}\Big) && (4.41)\\
&= \frac{A_n^0}{A_{n+1}^1} \cdot \frac{Y_{n+1}^{-1}(\theta,\phi)}{r^{n+2}}. && (4.42)
\end{aligned}$$

For $m > 0$ we have

$$\begin{aligned}
\partial_+\Big(\frac{Y_n^m(\theta,\phi)}{r^{n+1}}\Big) &= A_n^m \cdot \partial_+^{m+1}\Big(\frac{\partial}{\partial z}\Big)^{n-m}\Big(\frac{1}{r}\Big) && (4.43)\\
&= A_n^m \cdot \partial_+^{m+1}\Big(\frac{\partial}{\partial z}\Big)^{(n+1)-(m+1)}\Big(\frac{1}{r}\Big) && (4.44)\\
&= \frac{A_n^m}{A_{n+1}^{m+1}} \cdot \frac{Y_{n+1}^{m+1}(\theta,\phi)}{r^{n+2}} && (4.45)
\end{aligned}$$

and

$$\begin{aligned}
\partial_-\Big(\frac{Y_n^m(\theta,\phi)}{r^{n+1}}\Big) &= A_n^m \cdot \partial_-\partial_+^m\Big(\frac{\partial}{\partial z}\Big)^{n-m}\Big(\frac{1}{r}\Big) && (4.46)\\
&= -A_n^m \cdot \partial_+^{m-1}\Big(\frac{\partial}{\partial z}\Big)^{n-m+2}\Big(\frac{1}{r}\Big) && (4.47)\\
&= -A_n^m \cdot \partial_+^{m-1}\Big(\frac{\partial}{\partial z}\Big)^{(n+1)-(m-1)}\Big(\frac{1}{r}\Big) && (4.48)\\
&= -\frac{A_n^m}{A_{n+1}^{m-1}} \cdot \frac{Y_{n+1}^{m-1}(\theta,\phi)}{r^{n+2}}. && (4.49)
\end{aligned}$$

Similarly, we have

$$\partial_-\Big(\frac{Y_n^{-m}(\theta,\phi)}{r^{n+1}}\Big) = A_n^m\cdot\partial_-^{m+1}\Big(\frac{\partial}{\partial z}\Big)^{n-m}\Big(\frac{1}{r}\Big) \tag{4.50}$$

$$= A_n^m\cdot\partial_-^{m+1}\Big(\frac{\partial}{\partial z}\Big)^{(n+1)-(m+1)}\Big(\frac{1}{r}\Big) \tag{4.51}$$

$$= \frac{A_n^m}{A_{n+1}^{m+1}}\cdot\frac{Y_{n+1}^{-(m+1)}(\theta,\phi)}{r^{n+2}} \tag{4.52}$$

and

$$\partial_+\Big(\frac{Y_n^{-m}(\theta,\phi)}{r^{n+1}}\Big) = A_n^m\cdot\partial_+\partial_-^m\Big(\frac{\partial}{\partial z}\Big)^{n-m}\Big(\frac{1}{r}\Big) \tag{4.53}$$

$$= -A_n^m\cdot\partial_-^{m-1}\Big(\frac{\partial}{\partial z}\Big)^{n-m+2}\Big(\frac{1}{r}\Big) \tag{4.54}$$

$$= -A_n^m\cdot\partial_-^{m-1}\Big(\frac{\partial}{\partial z}\Big)^{(n+1)-(m-1)}\Big(\frac{1}{r}\Big) \tag{4.55}$$

$$= -\frac{A_n^m}{A_{n+1}^{m-1}}\cdot\frac{Y_{n+1}^{-(m-1)}(\theta,\phi)}{r^{n+2}}. \tag{4.56}$$

We also have for $m\geq 0$

$$\partial_z\Big(\frac{Y_n^m(\theta,\phi)}{r^{n+1}}\Big) = A_n^m\cdot\partial_+^m\Big(\frac{\partial}{\partial z}\Big)^{n+1-m}\Big(\frac{1}{r}\Big) \tag{4.57}$$

$$= \frac{A_n^m}{A_{n+1}^m}\cdot\frac{Y_{n+1}^m(\theta,\phi)}{r^{n+2}} \tag{4.58}$$

and

$$\partial_z\Big(\frac{Y_n^{-m}(\theta,\phi)}{r^{n+1}}\Big) = A_n^m\cdot\partial_-^m\Big(\frac{\partial}{\partial z}\Big)^{n+1-m}\Big(\frac{1}{r}\Big) \tag{4.59}$$

$$= \frac{A_n^m}{A_{n+1}^m}\cdot\frac{Y_{n+1}^{-m}(\theta,\phi)}{r^{n+2}}. \tag{4.60}$$

Therefore, for $m=0$ we have

$$\frac{\partial}{\partial x}\Big(\frac{Y_n^0(\theta,\phi)}{r^{n+1}}\Big) = \frac{A_n^0}{2r^{n+2}A_{n+1}^1}\Big(Y_{n+1}^1+Y_{n+1}^{-1}\Big) \tag{4.61}$$

$$\frac{\partial}{\partial y}\Big(\frac{Y_n^0(\theta,\phi)}{r^{n+1}}\Big) = \frac{A_n^0}{2ir^{n+2}A_{n+1}^1}\Big(Y_{n+1}^1-Y_{n+1}^{-1}\Big) \tag{4.62}$$

and for $m > 0$ we have

$$\frac{\partial}{\partial x}\Big(\frac{Y_n^m(\theta,\phi)}{r^{n+1}}\Big) = \frac{A_n^m}{2r^{n+2}}\Big(\frac{Y_{n+1}^{m+1}}{A_{n+1}^{m+1}} - \frac{Y_{n+1}^{m-1}}{A_{n+1}^{m-1}}\Big) \tag{4.63}$$

$$\frac{\partial}{\partial x}\Big(\frac{Y_n^{-m}(\theta,\phi)}{r^{n+1}}\Big) = \frac{A_n^m}{2r^{n+2}}\Big(\frac{Y_{n+1}^{-(m+1)}}{A_{n+1}^{m+1}} - \frac{Y_{n+1}^{-(m-1)}}{A_{n+1}^{m-1}}\Big) \tag{4.64}$$

$$\frac{\partial}{\partial y}\Big(\frac{Y_n^m(\theta,\phi)}{r^{n+1}}\Big) = \frac{A_n^m}{2ir^{n+2}}\Big(\frac{Y_{n+1}^{m+1}}{A_{n+1}^{m+1}} + \frac{Y_{n+1}^{m-1}}{A_{n+1}^{m-1}}\Big) \tag{4.65}$$

$$\frac{\partial}{\partial y}\Big(\frac{Y_n^{-m}(\theta,\phi)}{r^{n+1}}\Big) = -\frac{A_n^m}{2ir^{n+2}}\Big(\frac{Y_{n+1}^{-(m+1)}}{A_{n+1}^{m+1}} + \frac{Y_{n+1}^{-(m-1)}}{A_{n+1}^{m-1}}\Big). \tag{4.66}$$

We can use all of these relations to calculate the derivative of the local expansion. We begin with the x derivative of the terms of the local expansion:

$$\begin{aligned}\frac{\partial}{\partial x}\left(Y_n^m(\theta,\phi)r^n\right) &= \frac{\partial}{\partial x}\left(\frac{Y_n^m(\theta,\phi)}{r^{n+1}}r^{2n+1}\right) \\ &= \left(\frac{Y_n^m(\theta,\phi)}{r^{n+1}}\frac{\partial r^{2n+1}}{\partial x} + r^{2n+1}\frac{\partial}{\partial x}\left(\frac{Y_n^m(\theta,\phi)}{r^{n+1}}\right)\right) \\ &= \left((2n+1)Y_n^m(\theta,\phi)xr^{n-2} + r^{2n+1}\frac{\partial}{\partial x}\left(\frac{Y_n^m(\theta,\phi)}{r^{n+1}}\right)\right).\end{aligned} \tag{4.67}$$

In a similar manner, we express the derivatives of y and z as

$$\begin{aligned}\frac{\partial}{\partial y}\left(Y_n^m(\theta,\phi)r^n\right) &= \Bigg((2n+1)Y_n^m(\theta,\phi)yr^{n-2} + \\ &\qquad r^{2n+1}\frac{\partial}{\partial y}\left(\frac{Y_n^m(\theta,\phi)}{r^{n+1}}\right)\Bigg)\end{aligned} \tag{4.68}$$

$$\begin{aligned}\frac{\partial}{\partial z}\left(Y_n^m(\theta,\phi)r^n\right) &= \Bigg((2n+1)Y_n^m(\theta,\phi)zr^{n-2} + \\ &\qquad r^{2n+1}\frac{\partial}{\partial z}\left(\frac{Y_n^m(\theta,\phi)}{r^{n+1}}\right)\Bigg).\end{aligned} \tag{4.69}$$

We now express these derivatives in terms of the formulae just derived. For $m = 0$ we have

$$\frac{\partial}{\partial x}\left(Y_n^0(\theta,\phi)r^n\right) = r^{n-1}\Bigg((2n+1)Y_n^0(\theta,\phi)\sin(\theta)\cos(\phi) \tag{4.70}$$

$$+\frac{A_n^0}{2A_{n+1}^1}\left(Y_{n+1}^1+Y_{n+1}^{-1}\right)\Bigg)$$

$$\frac{\partial}{\partial y}\left(Y_n^0(\theta,\phi)r^n\right) = r^{n-1}\Bigg((2n+1)Y_n^0(\theta,\phi)\sin(\theta)\sin(\phi) \tag{4.71}$$

$$+\frac{A_n^0}{2iA_{n+1}^1}\left(Y_{n+1}^1-Y_{n+1}^{-1}\right)\Bigg)$$

$$\frac{\partial}{\partial z}\left(Y_n^0(\theta,\phi)r^n\right) = r^{n-1}\left((2n+1)Y_n^0(\theta,\phi)\cos(\theta)+\frac{A_n^0}{A_{n+1}^0}Y_{n+1}^0\right) \tag{4.72}$$

and for $m>0$ we have

$$\frac{\partial}{\partial x}\left(Y_n^m(\theta,\phi)r^n\right) = r^{n-1}\Bigg((2n+1)Y_n^m(\theta,\phi)\sin(\theta)\cos(\phi) \tag{4.73}$$

$$+\frac{A_n^m}{2}\left(\frac{Y_{n+1}^{m+1}}{A_{n+1}^{m+1}}-\frac{Y_{n+1}^{m-1}}{A_{n+1}^{m-1}}\right)\Bigg)$$

$$\frac{\partial}{\partial x}\left(Y_n^{-m}(\theta,\phi)r^n\right) = r^{n-1}\Bigg((2n+1)Y_n^{-m}(\theta,\phi)\sin(\theta)\cos(\phi) \tag{4.74}$$

$$+\frac{A_n^m}{2}\left(\frac{Y_{n+1}^{-(m+1)}}{A_{n+1}^{m+1}}-\frac{Y_{n+1}^{-(m-1)}}{A_{n+1}^{m-1}}\right)\Bigg)$$

$$\frac{\partial}{\partial y}\left(Y_n^m(\theta,\phi)r^n\right) = r^{n-1}\Bigg((2n+1)Y_n^m(\theta,\phi)\sin(\theta)\sin(\phi) \tag{4.75}$$

$$+\frac{A_n^m}{2i}\left(\frac{Y_{n+1}^{m+1}}{A_{n+1}^{m+1}}+\frac{Y_{n+1}^{m-1}}{A_{n+1}^{m-1}}\right)\Bigg)$$

$$\frac{\partial}{\partial y}\left(Y_n^{-m}(\theta,\phi)r^n\right) = r^{n-1}\Bigg((2n+1)Y_n^{-m}(\theta,\phi)\sin(\theta)\sin(\phi) \tag{4.76}$$

$$+\frac{A_n^m}{2i}\left(\frac{Y_{n+1}^{-(m+1)}}{A_{n+1}^{m+1}}+\frac{Y_{n+1}^{-(m-1)}}{A_{n+1}^{m-1}}\right)\Bigg)$$

$$\frac{\partial}{\partial z}\left(Y_n^m(\theta,\phi)r^n\right) = r^{n-1}\left((2n+1)Y_n^m(\theta,\phi)\cos(\theta)+\frac{A_n^m}{A_{n+1}^m}Y_{n+1}^m\right) \tag{4.77}$$

$$\frac{\partial}{\partial z}\left(Y_n^{-m}(\theta,\phi)r^n\right) = r^{n-1}\left((2n+1)Y_n^{-m}(\theta,\phi)\cos(\theta)+\frac{A_n^m}{A_{n+1}^m}Y_{n+1}^{-m}\right). \tag{4.78}$$

We use these formulas for computing the derivatives of the terms in the local expansions.

The PMTA implementation also does not always calculate ratios of the form $z/\sqrt{x^2+y^2+z^2}$ to be less than or equal to 1 when x and y are both zero or nearly

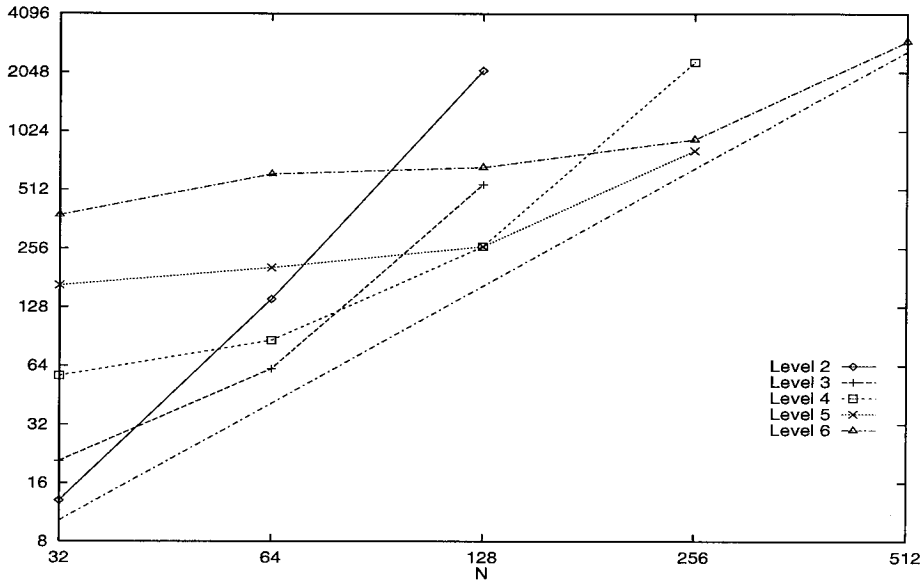

Figure 4.2: Timings for various values of N at various levels of subdivision with precision $p = 20$. The line without any points is proportional to N^2, showing that the optimal time is proportional to N^2.

zero. This is significant because this ratio must be less than or equal to one in order to properly compute the azimuthal angle. Some rescaling of the terms was sufficient to eliminate the problem.

4.2.4 Practical Considerations and Timings

The time required to perform a time step is dependent on a number of factors. The level of precision required determines the number of terms in the multipole expansion. For a given precision ϵ, the number of terms p required for the multipole expansion is given by $p = -\log_2(\epsilon)$. For a given choice of p, the number of levels of subdivision of the center cell is chosen to minimize the computational time. Figure 4.2 shows the timings for various values of n at various levels of subdivision, for a fixed precision $p = 20$. Hence, for $p = 20$, and $N = 64$, we use 3 levels of subdivision.

We are also interested in the computational cost associated with increasing the precision of the calculations. According to the theory, the computational time required is $O(p^4)$. This, however, assumes a fixed level of subdivision. Since the

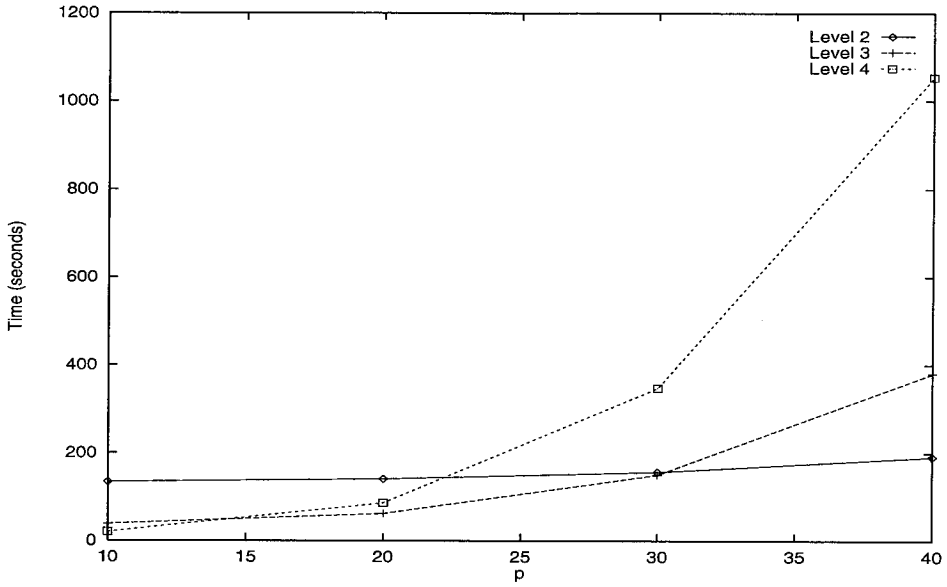

Figure 4.3: Timings for $N = 64$ for various values of p at various levels of subdivision. The increase in optimum time versus p is approximately linear in p.

optimal timing occurs at different levels for different values of p, the computational time required is much less. In Figure 4.3 we show timings for $N = 64$ as p increases. The increase in the optimal time appears to be only linear in p. We note that as p increases, the fastest timings occur at a lower level of subdivision. This suggests that for fixed N, the required memory can *decrease* as p increases. This is because the memory required is related to the number of levels of subdivision.

An additional cost factor is involved in computing high-order quadrature rules. Using a direct method, Richardson extrapolation can be accomplished by applying a weight function to the summand. The cost of this is very small. However, the weight function consists of both source and field terms which cannot be factored. Hence, in order to compute a third-order quadrature rule using Richardson extrapolation, we must compute an integral at the grid spacing h and 4 integrals at the grid spacing $2h$. This effectively doubles the computational cost when compared to computing one integral with grid spacing h. We can further reduce the computational cost by reducing the number of iterations needed to determine μ. To do this, we implemented an extrapolation scheme to approximate the value of μ at the next

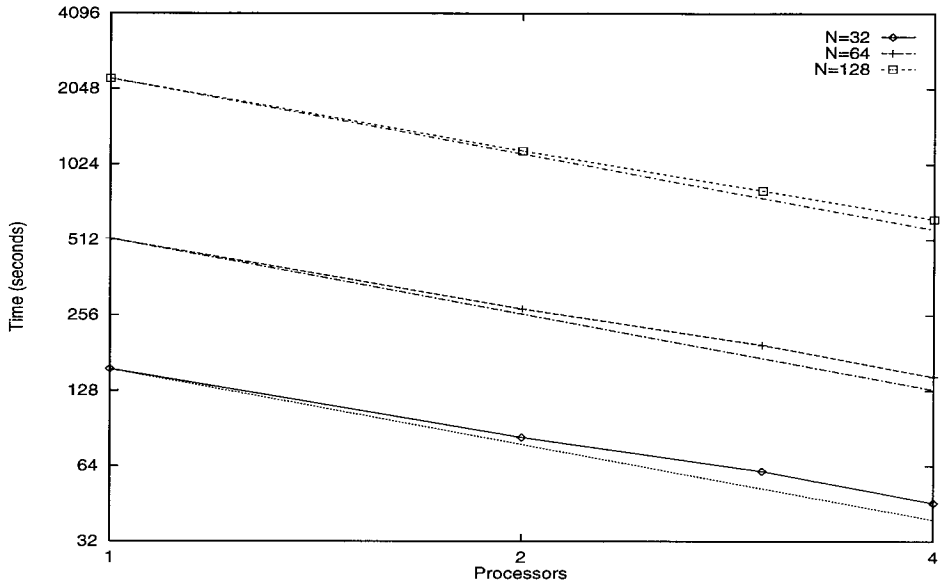

Figure 4.4: Timings per time step with $N = 32, 64$ and 128 for $1 - 4$ processors on the SGI Power Onyx. The lines without points show the ideal speed up.

time step.

In Figures 4.4 and 4.5 we show the timings as the number of processors is increased for various values of N on the SGI Power Onyx and on the Cray J90. The timings were done for one time step with the Stokes wave initial condition, using the desingularization stabilization method and the third-order scheme. These plots show characteristic timings for calculating each time step using the third-order quadrature rule.

For the majority of our numerical experiments, we took $N = 64$, $p = 20$ (single precision accuracy), 3 levels of subdivision, using 2 processors on the SGI Onyx. A typical experiment required approximately 8 MB of memory and required 2 iterations to compute μ. With these parameters, the typical time required per time step is about 3 minutes. A test problem with $N = 512$ at 6 levels of subdivision required approximately 220 MB of memory. For a problem of this size, we expect to need about 3 hours per time step.

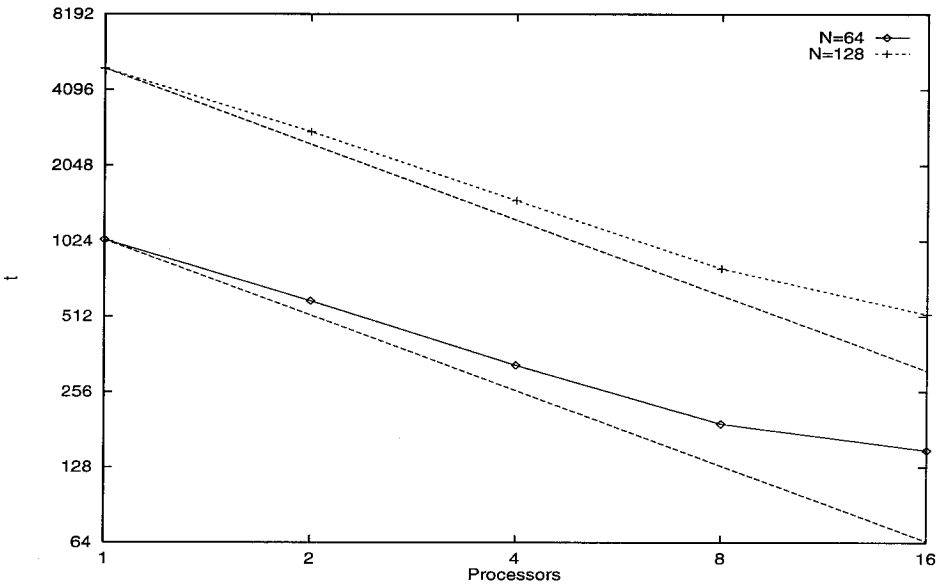

Figure 4.5: Timings per time step with $N = 64$ and $N = 128$ for $1 - 16$ processors on the Cray J90. The lines without points show the ideal speed up.

Chapter 5

Stability

The numerical stability of the boundary integral method for two-dimensional water waves is a problem which dates to the work by Longuet-Higgins and Cokelet. More recently, the stability issue has been examined in studies by Beale, Hou, and Lowengrub [3] and Hou [10]. In their work, Beale *et al.* showed that in order for the boundary integral method to be numerically stable, a compatibility condition between the discrete derivative operator and the quadrature rule must be satisfied. They demonstrate a Fourier smoothing method which can be used to impose the compatibility condition. A similar compatibility condition is required in the three-dimensional case. In the three-dimensional case, however, the coordinate system used to discretize the system must also satisfy certain additional conditions.

In this chapter we analyze the numerical stability of the boundary integral formulation near equilibrium. This analysis is based on a study of the continuous formulation for three-dimensional water waves by Hou, Teng, and Zhang [13] and on a study of the stability of the boundary integral method for three-dimensional water waves by Hou and Zhang [14]. We propose several methods for stabilizing the numerical scheme. We apply these methods to a non-linear problem and discuss the advantages and disadvantages of each method.

5.0.5 Preliminary Definitions

The discrete version of the defining equations is given by

$$\mu_{\mathbf{i}} = 2\phi_{\mathbf{i}} + \frac{1}{2\pi}\sum_{\mathbf{i}\neq\mathbf{j}} \mu_{\mathbf{j}}\hat{\mathbf{n}}_{\mathbf{j}} \cdot \nabla_{\mathbf{x}} G(\mathbf{x_i} - \mathbf{x_j})h^2 \tag{5.1}$$

$$\boldsymbol{\kappa}_{\mathbf{i}} = -D_{x_1 h}\mu_{\mathbf{i}} D_{x_2 h}\mathbf{x_i} + D_{x_2 h}\mu_{\mathbf{i}} D_{x_1 h}\mathbf{x_i} \tag{5.2}$$

$$\tilde{\mathbf{u}}_{\mathbf{i}} = \frac{1}{4\pi}\sum_{\mathbf{i}\neq\mathbf{j}} \boldsymbol{\kappa}_{\mathbf{j}} \times \nabla_{\mathbf{x}} G(\mathbf{x_i} - \mathbf{x_j})h^2 \tag{5.3}$$

$$\mathbf{u}_{\mathbf{i}} = \tilde{\mathbf{u}}_{\mathbf{i}} - \frac{1}{2}\frac{\boldsymbol{\kappa}_{\mathbf{i}} \times \hat{\mathbf{n}}_{\mathbf{i}}}{D_{x_1 h}\mathbf{x_i} \times D_{x_2 h}\mathbf{x_i}} \tag{5.4}$$

$$\frac{d\mathbf{x_i}}{dt} = \tilde{\mathbf{u}}_{\mathbf{i}} \tag{5.5}$$

$$\frac{d\phi_i}{dt} = \tilde{\mathbf{u}}_{\mathbf{i}} \cdot \mathbf{u_i} - \frac{1}{2}\mathbf{u_i} \cdot \mathbf{u_i} - \mathbf{g} \cdot \mathbf{x_i} \tag{5.6}$$

where $D_{x_l h}$ is the discrete derivative operator.

We begin the discussion of the numerical stability of the method by defining a number of discrete operators. We recall the definition of the discrete Fourier transform from [5]:

$$\hat{u}_k = \sum_{(j_1,j_2)=(-\frac{N}{2}+1,-\frac{N}{2}+1)}^{(\frac{N}{2},\frac{N}{2})} u(\mathbf{x}_j)e^{-i\mathbf{k}\cdot\mathbf{x_j}}. \tag{5.7}$$

The inversion formula is

$$u_j = \frac{1}{N^2}\sum_{(k_1,k_2)=(-\frac{N}{2}+1,-\frac{N}{2}+1)}^{(\frac{N}{2},\frac{N}{2})} \hat{u}_k e^{-i\mathbf{k}\cdot\mathbf{x_j}} \tag{5.8}$$

where $\mathbf{x_j} = (x_1 h, x_2 h)$ and $h = 2\pi/N$ The Fourier transform of a discrete derivative operator is expressed as follows:

$$\widehat{(D_{x_j h} f)}_k = ik_j \rho_j(h\mathbf{k})\hat{f}_k. \tag{5.9}$$

If, for example, the discrete derivative operator $D_{x_j h}$ is a spectral derivative applied directly in the Fourier transform, then $\rho(h\mathbf{k}) = 1$.

We also make use of the symbols H_{1h}, H_{2h} and Λ_h, defined by

$$H_{lh}(f_{\mathbf{i}}) = \frac{1}{2\pi} \sum_{\mathbf{i}\neq\mathbf{j}} \frac{(x_{l\mathbf{i}} - x_{l\mathbf{j}})f_i}{((x_{1\mathbf{i}} - x_{1\mathbf{j}})^2 + (x_{2\mathbf{i}} - x_{2\mathbf{j}})^2)^{3/2}} h^2 \qquad l = 1, 2 \tag{5.10}$$

and

$$\Lambda_h(f_{\mathbf{i}}) = \frac{1}{2\pi} \sum_{\mathbf{i}\neq\mathbf{j}} \frac{f_{\mathbf{i}} - f_{\mathbf{j}}}{((x_{1\mathbf{i}} - x_{1\mathbf{j}})^2 + (x_{2\mathbf{i}} - x_{2\mathbf{j}})^2)^{3/2}} h^2. \tag{5.11}$$

These operators arise from the quadrature rule used to define the dipole sheet strength when we linearize the equations about equilibrium.

The stability of the continuous water wave formulation depends on the relation

$$\Lambda(f) = (H_1 D_{x_1} + H_2 D_{x_2})(f) \tag{5.12}$$

where Λ, H_l, and D_{x_l} are the continuous analogs of Λ_h, H_{lh}, and $D_{x_l h}$. The discrete analog of this relation is not necessarily true. In general we have

$$\Lambda_h(f) \neq (H_{1h} D_{x_1 h} + H_{2h} D_{x_2 h})(f). \tag{5.13}$$

We show in the linear stability analysis that this condition gives rise to numerical instabilities.

We are interested in the Fourier transform of $H_{lh}f$ and $\Lambda_h f$. Because these are convolution operators, we need the definitions of the discrete convolution and its Fourier transform:

$$f * g(\mathbf{x}_{\mathbf{i}}) = \sum_{\mathbf{j}} f(\mathbf{x}_{\mathbf{i}} - \mathbf{x}_{\mathbf{j}}) g(\mathbf{x}_{\mathbf{j}}) \tag{5.14}$$

and

$$\widehat{f * g}_k = \hat{f}_k \hat{g}_k. \tag{5.15}$$

Taking $\mathbf{x}_{\mathbf{j}} = (j_1 h, j_2 h)$ and $h = 2\pi/N$, we can now compute the following:

$$(\widehat{H_{lh} f})_k = -i\rho^l(h\mathbf{k})\hat{f}_k \tag{5.16}$$

where

$$\rho^l(h\mathbf{k}) = \frac{i}{2\pi} \sum_{(j_1,j_2)} \frac{j_l e^{-ih(j_1k_1+j_2k_2)}}{(j_1^2+j_2^2)^{3/2}}. \tag{5.17}$$

Making use of the symmetry of the summand, we have the following definitions for $\rho^1(h\mathbf{k})$ and $\rho^2(h\mathbf{k})$:

$$\rho^1(h\mathbf{k}) = \frac{1}{2\pi} \sum_{(j_1,j_2)} \frac{j_1 \sin(j_1k_1h)\cos(j_2k_2h)}{(j_1^2+j_2^2)^{3/2}} \tag{5.18}$$

$$\rho^2(h\mathbf{k}) = \frac{1}{2\pi} \sum_{(j_1,j_2)} \frac{j_2 \sin(j_2k_2h)\cos(j_1k_1h)}{(j_1^2+j_2^2)^{3/2}}. \tag{5.19}$$

We express the Fourier transform of $\Lambda_h(f)$ using the following form

$$\widehat{(\Lambda_h f)}_k = |\mathbf{k}|\bar{\rho}(h\mathbf{k})\hat{f}_k. \tag{5.20}$$

To do this, we rewrite $\Lambda_h(f_i)$ as

$$\Lambda_h(f_\mathbf{i}) = \frac{f_i}{2\pi} \sum_{\mathbf{i}\neq\mathbf{j}} \frac{h^2}{((x_{1\mathbf{i}}-x_{1\mathbf{j}})^2+(x_{2\mathbf{i}}-x_{2\mathbf{j}})^2)^{3/2}} - \frac{1}{2\pi} \sum_{\mathbf{i}\neq\mathbf{j}} \frac{h^2 f_\mathbf{j}}{((x_{1\mathbf{i}}-x_{1\mathbf{j}})^2+(x_{2\mathbf{i}}-x_{2\mathbf{j}})^2)^{3/2}}. \tag{5.21}$$

Making use of the convolution theorem on the second sum in (5.22) we have

$$\widehat{(\Lambda_h f)}_k = \left[\frac{1}{2\pi h} \sum_{(j_1,j_2)} \frac{1}{(j_1^2+j_2^2)^{3/2}} - \frac{1}{2\pi h} \sum_{(j_1,j_2)} \frac{e^{-ih(j_1k_1+j_2k_2)}}{(j_1^2+j_2^2)^{3/2}} \right] \hat{f}_k. \tag{5.22}$$

Comparing the right-hand sides of Equations 5.20 and 5.22, and again using the symmetry of the summand gives the definition of $\bar{\rho}(h\mathbf{k})$:

$$\bar{\rho}(h\mathbf{k}) = \frac{1}{2\pi h|\mathbf{k}|} \sum_{(j_1,j_2)} \frac{1-\cos(k_1j_1h)\cos(k_2j_2h)}{(j_1^2+j_2^2)^{3/2}}. \tag{5.23}$$

Finally, we are interested in the Fourier transform of $R_h f$ where

$$R_h = H_{1h}D_{x_1h} + H_{2h}D_{x_2h}. \tag{5.24}$$

We express $(\widehat{R_h f})_k$ in the following form:

$$(\widehat{R_h f})_k = |\mathbf{k}|\tilde{\rho}(h\mathbf{k})\hat{f}_k. \tag{5.25}$$

From the definitions of H_{lh} and $D_{x_l h}$ we have

$$\tilde{\rho} = \frac{k_1\rho^1\rho_1 + k_2\rho^2\rho_2}{|\mathbf{k}|}. \tag{5.26}$$

5.1 Linear Stability

5.1.1 Analysis

We now examine the linear stability about equilibrium. Let

$$\mathbf{x} = (x_1 + x_1', x_2 + x_2', z'(x_1, x_2)) \qquad \phi = \frac{C}{2} + \phi' \tag{5.27}$$

where the primed quantities are assumed to be small and C is an order 1 constant. Substituting these into the equation defining μ gives to leading order

$$\mu_\mathbf{i} = C + 2\phi_\mathbf{i}' + \frac{1}{2\pi}\sum_{\mathbf{i}\neq\mathbf{j}} \mu_\mathbf{j} h^2 \times \left[\frac{-D_{x_1 h} z_\mathbf{j}'(x_{1\mathbf{i}} - x_{1\mathbf{j}})}{((x_{1\mathbf{i}} - x_{1\mathbf{j}})^2 + (x_{2\mathbf{i}} - x_{2\mathbf{j}})^2)^{3/2}}\right. \tag{5.28}$$

$$\left. + \frac{-D_{x_2 h} z_\mathbf{j}'(x_{2\mathbf{i}} - x_{2\mathbf{j}})}{((x_{1\mathbf{i}} - x_{1\mathbf{j}})^2 + (x_{2\mathbf{i}} - x_{2\mathbf{j}})^2)^{3/2}} + \frac{(z_\mathbf{i}' - z_\mathbf{j}')}{((x_{1\mathbf{i}} - x_{1\mathbf{j}})^2 + (x_{2\mathbf{i}} - x_{2\mathbf{j}})^2)^{3/2}}\right]. \tag{5.29}$$

We can write $\mu_\mathbf{i}$ as $\mu_\mathbf{i} = C + \mu_\mathbf{i}'$ where μ' is small. Substituting this into the right-hand side gives to leading order

$$\mu_\mathbf{i} = C + 2\phi_\mathbf{i}' + \frac{C}{2\pi}\sum_{\mathbf{i}\neq\mathbf{j}} h^2 \times \left[\frac{-D_{x_1 h} z_\mathbf{j}'(x_{1\mathbf{i}} - x_{1\mathbf{j}})}{((x_{1\mathbf{i}} - x_{1\mathbf{j}})^2 + (x_{2\mathbf{i}} - x_{2\mathbf{j}})^2)^{3/2}}\right. \tag{5.30}$$

$$\left. + \frac{-D_{x_2 h} z_\mathbf{j}'(x_{2\mathbf{i}} - x_{2\mathbf{j}})}{((x_{1\mathbf{i}} - x_{1\mathbf{j}})^2 + (x_{2\mathbf{i}} - x_{2\mathbf{j}})^2)^{3/2}} + \frac{(z_\mathbf{i}' - z_\mathbf{j}')}{((x_{1\mathbf{i}} - x_{1\mathbf{j}})^2 + (x_{2\mathbf{i}} - x_{2\mathbf{j}})^2)^{3/2}}\right]. \tag{5.31}$$

We simplify this expression by recalling the definitions of Λ_h and R_h. The expression becomes

$$\mu_\mathbf{i} = 2\phi_\mathbf{i}' + C + C(\Lambda_h - R_h)(z_\mathbf{i}'). \tag{5.32}$$

The vortex sheet strength, to leading order is given by

$$\kappa_{\mathbf{i}} = (D_{x_2 h}(2\phi_{\mathbf{i}}' + C(\Lambda_h - R_h)(z_{\mathbf{i}}')), -D_{x_1 h}(2\phi_{\mathbf{i}}' + C(\Lambda_h - R_h)(z_{\mathbf{i}}')), 0). \tag{5.33}$$

Substituting this equation into the equation defining the velocity gives

$$\tilde{\mathbf{u}}_{\mathbf{i}} = (0, 0, \frac{1}{2} R_h(2\phi_{\mathbf{i}}' + C(\Lambda_h - R_h)(z_{\mathbf{i}}'))). \tag{5.34}$$

The equations defining the time evolution become

$$\frac{d\mathbf{x_i}'}{dt} = (0, 0, R_h(\phi_{\mathbf{i}}') + \frac{C}{2} R_h(\Lambda_h - R_h)(z_{\mathbf{i}}'))) \tag{5.35}$$

$$\frac{d\phi_i'}{dt} = -g z_{\mathbf{i}}'. \tag{5.36}$$

We write this as a linear system:

$$\frac{d}{dt}\begin{bmatrix} x_1' \\ x_2' \\ z' \\ \phi' \end{bmatrix}_{\mathbf{i}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{C}{2} R_h(\Lambda_h - R_h) & R_h \\ 0 & 0 & -g & 0 \end{bmatrix} \begin{bmatrix} x_1' \\ x_2' \\ z' \\ \phi' \end{bmatrix}_{\mathbf{i}}. \tag{5.37}$$

We now transform the system of equations, which yields

$$\frac{d}{dt}\begin{bmatrix} \hat{x}_1' \\ \hat{x}_2' \\ \hat{z}' \\ \hat{\phi}' \end{bmatrix}_{\mathbf{k}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{C}{2}|\mathbf{k}|^2 \tilde{\rho}(\bar{\rho} - \tilde{\rho}) & |\mathbf{k}|\tilde{\rho} \\ 0 & 0 & -g & 0 \end{bmatrix} \begin{bmatrix} \hat{x}_1' \\ \hat{x}_2' \\ \hat{z}' \\ \hat{\phi}' \end{bmatrix}_{\mathbf{k}}. \tag{5.38}$$

The eigenvalues of this system are

$$\lambda_1, \lambda_2 = 0, \quad \lambda_3, \lambda_4 = \frac{C|\mathbf{k}|^2}{4} \tilde{\rho}(\bar{\rho} - \tilde{\rho}) \pm \frac{1}{4}\sqrt{C^2|\mathbf{k}|^4 \tilde{\rho}^2(\bar{\rho} - \tilde{\rho})^2 - 16g|k|\tilde{\rho}}. \tag{5.39}$$

If $C \neq 0$, $\tilde{\rho} \neq 0$, and $\bar{\rho} \neq \tilde{\rho}$, then the high mode instability is $O(|\mathbf{k}|^2)$. The preceding analysis suggests that if we are able to choose discrete derivative and quadrature

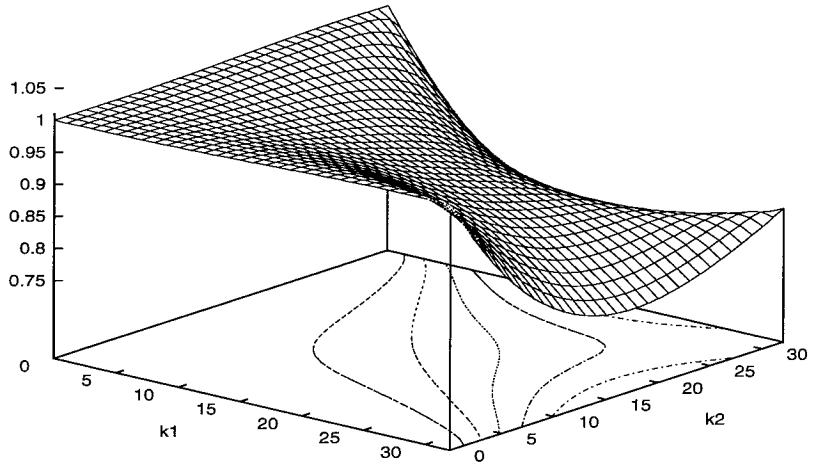

Figure 5.1: The computed value of $\bar{\rho}$ for $h = 2\pi/N$,$N = 64$.

rules such that

$$\Lambda_h(f) = (H_{1h}D_{x_1h} + H_{2h}D_{x_2h})(f) \tag{5.40}$$

then the method will be stable. In the two-dimensional case, this condition can be satisfied using operators with spectral accuracy. However, in the three-dimensional case, it is not possible to choose operators which satisfy the compatibility condition. In order to obtain a stable scheme, we use Fourier smoothing as explained in the next section.

5.1.2 Numerical Experiments

We have performed several numerical experiments to verify the predictions of the linear stability analysis and to demonstrate the high mode instability. We use the fast multipole implementation to approximate the Green's function kernel to single precision accuracy. We use the fixed grid formulation with third-order quadrature rules. We use spectral discrete derivative operators, so $\rho_1 = \rho_2 = 1$. We take $g = 1$, $N = 64$ and $h = 2\pi/N$. We show the computed values of $\bar{\rho}$ and $\tilde{\rho}$ in Figures 5.1 and 5.2, respectively.

The difference between $\bar{\rho}$ and $\tilde{\rho}$ gives rise to the high mode instability. We

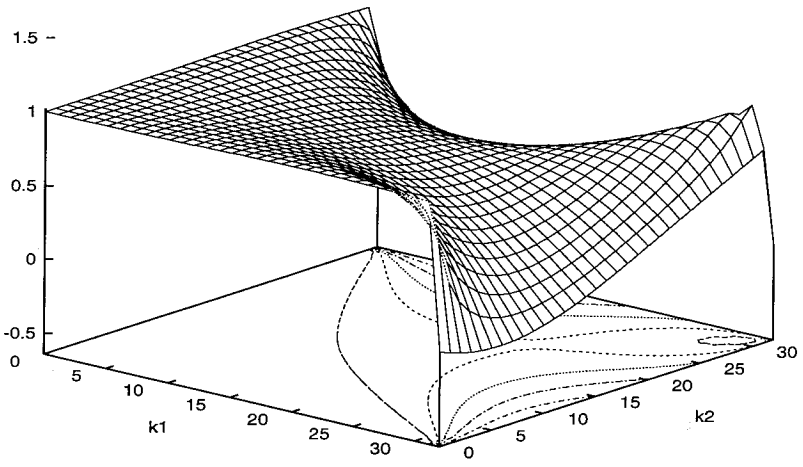

Figure 5.2: The computed value of $\tilde{\rho}$ for $h = 2\pi/N$,$N = 64$.

observe that the greatest deviation between $\bar{\rho}$ and $\tilde{\rho}$ occurs for larger values of $|\mathbf{k}|$

For our first experiment we take $C = 0$. We use for initial conditions solutions of the linear wave equation for various values of $\mathbf{k}$. (See, for example, Whitham [30]) . The initial condition has the form

$$\eta(x_1, x_2, 0) = \epsilon \cos(\mathbf{k} \cdot \mathbf{x}) \tag{5.41}$$

$$\phi(x_1, x_2, 0) = \frac{\epsilon}{\omega} \sin(\mathbf{k} \cdot \mathbf{x}) \exp(|\mathbf{k}| \eta(x_1, x_2)) \tag{5.42}$$

where ω is given by the dispersion relation $\omega = \sqrt{|\mathbf{k}|}$. We refer to this as the perturbed equilibrium initial condition.

For the case $C = 0$, the linear theory predicts that the method is stable, with a numerical dispersion relation ω_A given by

$$\omega_A = \sqrt{\tilde{\rho}|\mathbf{k}|}. \tag{5.43}$$

In Figure 5.3 we show the predicted numerical dispersion relation for $C = 0$. In Figure 5.4 we show the theoretical dispersion relation for linear water waves. In Figure 5.5 we compare ω and the predicted and computed values for ω_A for various

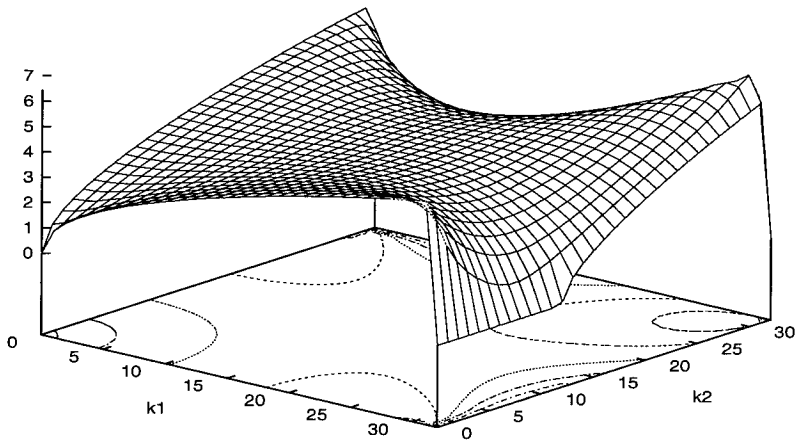

Figure 5.3: The predicted numerical dispersion relation for $h = 2\pi/N$,$N = 64$, and $C = 0$.

values of $\mathbf{k}$ along the line $k_1 = k_2$. We note that the numerical experiments give very good agreement with the predicted values for ω_A.

To demonstrate the instability, we set $C = 1$ and use the initial conditions

$$\eta(x_1, x_2, t) = 0 \qquad \phi(x_1, x_2, t) = \frac{1}{2}. \tag{5.44}$$

The initial condition is also the solution for all time. We refer to this as the equilibrium initial condition. For this case, the eigenvalues of Equation 5.38 generally have a real part. In Figure 5.6 we show the real part of λ_3 from Equation 5.39. Because the magnitudes of the eigenvalues differ greatly for various $\mathbf{k}$, we display the same information in Figure 5.7 on a logarithmic scale. (For a few values of $\mathbf{k}$ the eigenvalue is smaller than 10^{-5}. We have displayed these values as 10^{-5}.) Figure 5.7 shows that for nearly all values of $\mathbf{k}$, there is an eigenvalue with positive real part. In Figure 5.8 we show the complex part of the non-zero eigenvalues. We note that most of the high frequency modes have no oscillatory component.

After a single time step, the computed value of η is non-zero because the fast multipole method only approximates the Green's function kernel. In effect, the mul-

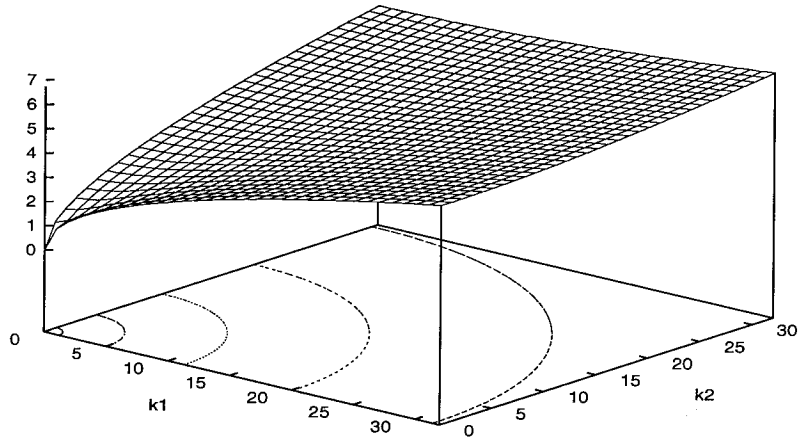

Figure 5.4: The theoretical dispersion relation for linear water waves.

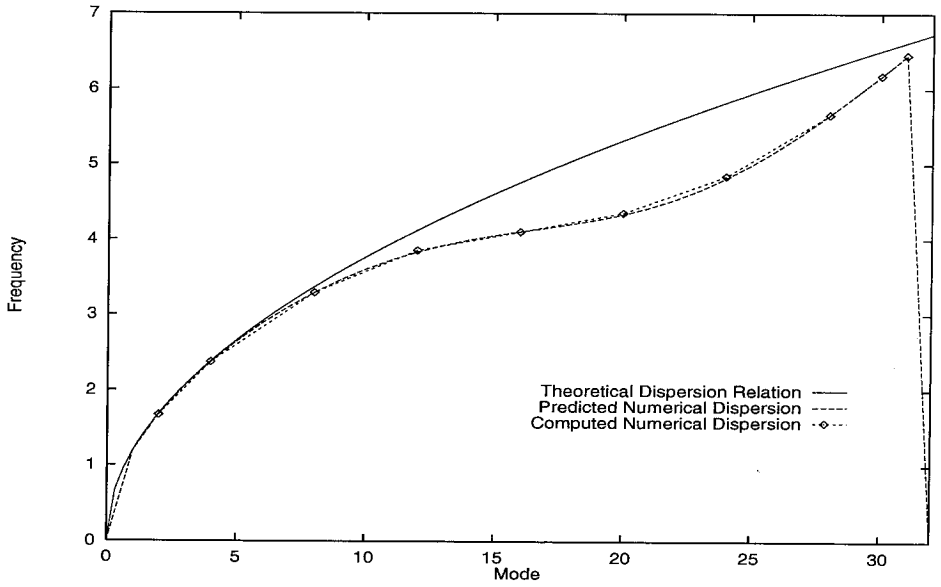

Figure 5.5: A comparison of the linear water wave dispersion relation, the predicted numerical dispersion relation and the computed numerical dispersion relation for various values of $\mathbf{k}$. For these computations we used $h = 2\pi/N$, $N = 64$, and $C = 0$.

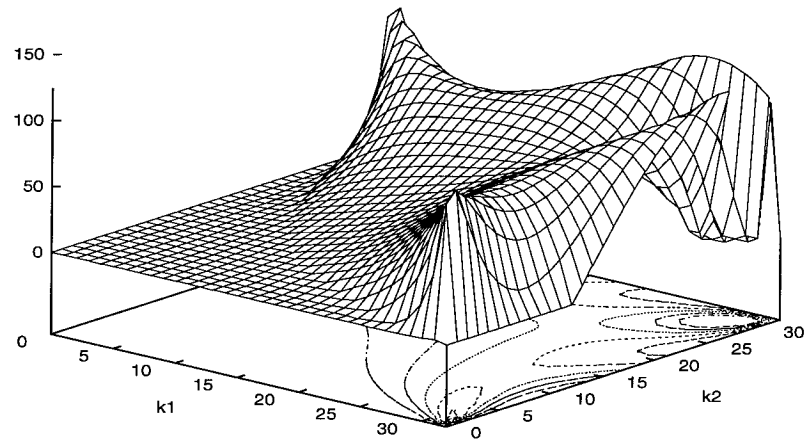

Figure 5.6: The real part of λ_3 from Equation 5.39 for $h = 2\pi/N$,$N = 64$,and $C = 1$.

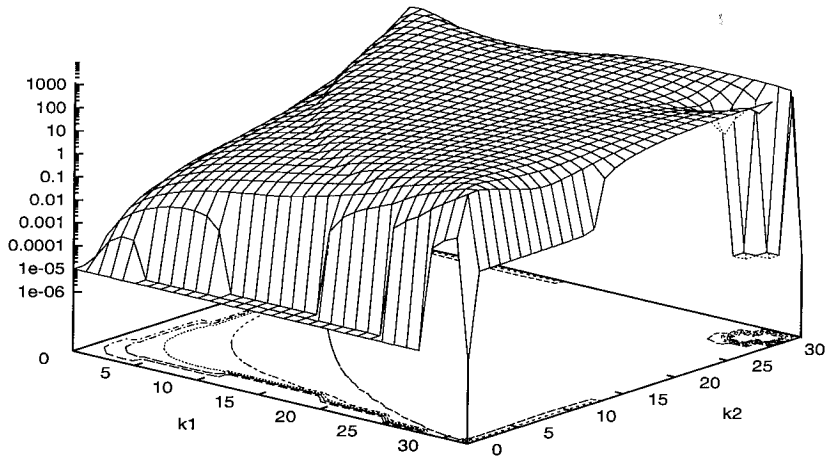

Figure 5.7: The real part of λ_3 for $h = 2\pi/N$,$N = 64$, and $C = 1$ on a logarithmic scale.

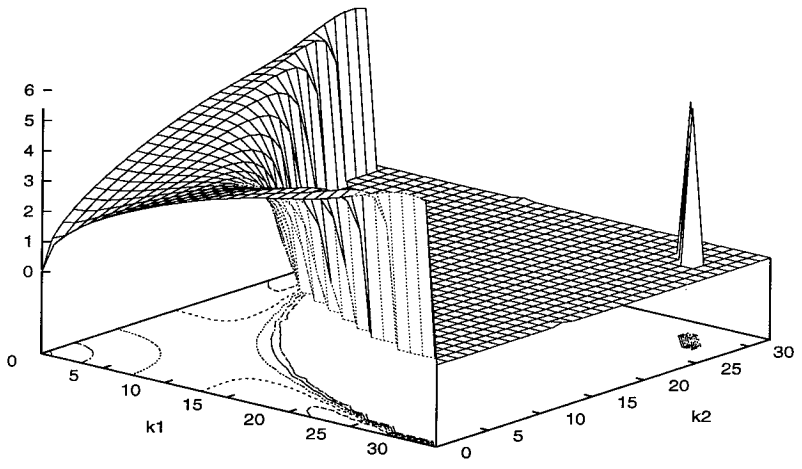

Figure 5.8: The complex part of λ_3 for $h = 2\pi/N$,$N = 64$, and $C = 1$.

tipole approximation uniformly perturbs the initial condition. We show in Figure 5.9 the spectral decomposition of η at time $t = \Delta t \simeq .001$ where $\Delta t = \pi/3000$. In Figure 5.10 we show the spectral analysis of η for time $t = 100\Delta t \simeq .105$ The growth of the higher modes qualitatively matches the prediction of the linear stability analysis. We especially note the striking similarity between Figures 5.6 and 5.10.

We are able to compare the observed growth rates with the growth rates predicted by the analysis. In order to examine the rate of growth for some of the modes with slower rate of growth, we use the perturbed equilibrium initial condition:

$$\eta(x_1, x_2, 0) = \epsilon \cos(\mathbf{k} \cdot \mathbf{x}) \tag{5.45}$$

$$\phi(x_1, x_2, 0) = \frac{1}{2} + \frac{\epsilon}{\omega} \sin(\mathbf{k} \cdot \mathbf{x}) \exp(|\mathbf{k}|\eta(x_1, x_2)) \tag{5.46}$$

where ω is given by the dispersion relation $\omega = \sqrt{|\mathbf{k}|}$. It is, however, difficult to accurately compute the growth rate in the lower modes because of the very rapid growth in the high modes.

In Table 5.1 we compare the predicted and computed growth rates for a number of modes. We denote the predicted and computed growth rates by λ_A and λ_n,

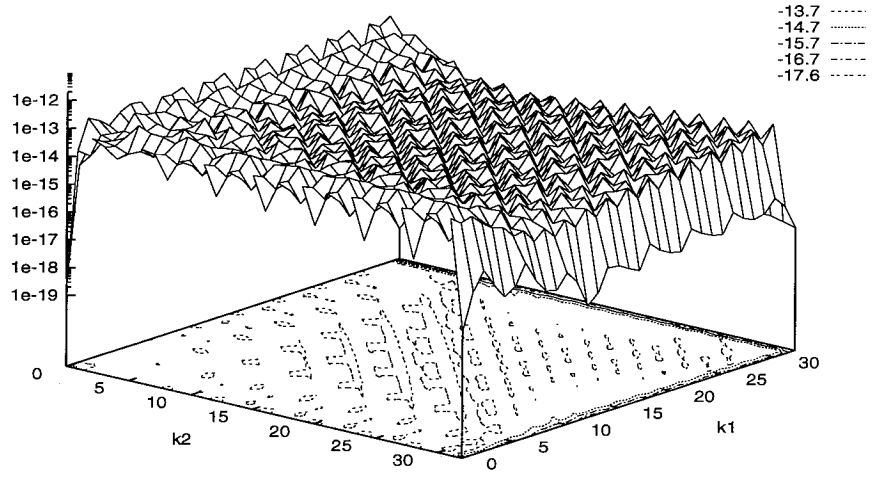

Figure 5.9: The spectrum of η after one time step using $\Delta t = \pi/3000$, $h = 2\pi/N$, $N = 64$, and $C = 1$. This corresponds to $t \simeq .001$.

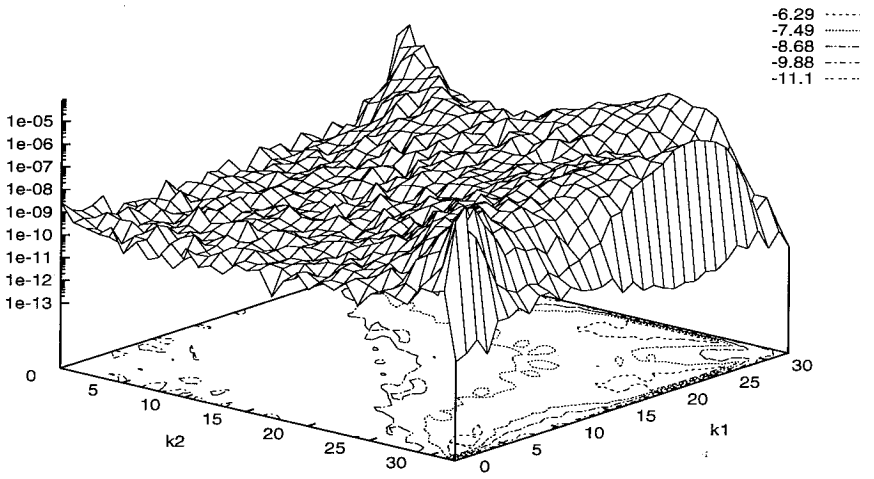

Figure 5.10: The spectrum of η after 100 time steps using $\Delta t = \pi/3000$,$h = 2\pi/N$, $N = 64$, and $C = 1$. This corresponds to $t \simeq .105$.

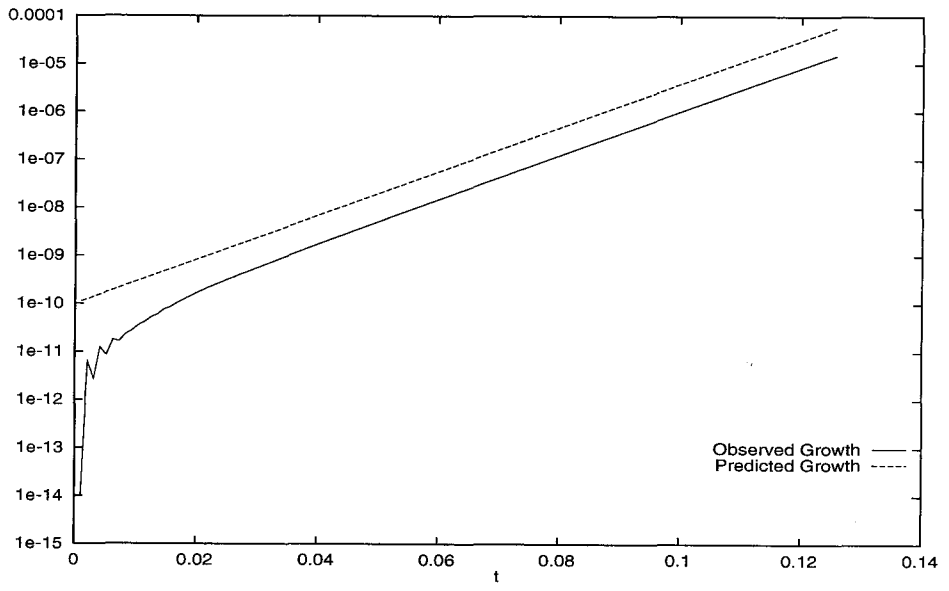

Figure 5.11: A comparison of the predicted and computed growth for the mode with wave number $(k_1, k_2) = (30, 3)$.

respectively. In Figure 5.11 we compare the growth of the mode $(k_1, k_2) = (30, 3)$ with the predicted growth rate.

5.1.3 Stabilization

We have implemented several approaches to stabilizing the numerical method, including Fourier smoothing, numerical viscosity, and a desingularization method. We discuss results for the linear theory in this section. In the next section we discuss the results for the nonlinear problem.

Desingularization

We make use of an alternate definition of the velocity potential at the interface

$$\phi = \frac{1}{4\pi} \int_{S'} (\mu(x) - \mu(x')) \hat{\mathbf{n}}' \cdot \nabla_{\mathbf{x}} G(\mathbf{x} - \mathbf{x}') \partial S' \tag{5.47}$$

since

$$\int_{S'} \hat{\mathbf{n}}' \cdot \nabla G(\mathbf{x} - \mathbf{x}') \partial S' = 0. \tag{5.48}$$

k_1	k_2	λ_A	λ_n
15	15	27.43167185	27.2
20	17	50.89035207	51.1
20	20	56.71974673	57.1
23	8	48.33265643	48.6
23	11	58.33481304	58.3
23	18	66.23485348	66.3
23	23	64.13379931	64.1
25	12	73.65631713	73.8
25	25	57.33661827	57.4
28	4	80.83032077	80.9
28	9	80.73287223	80.7
28	15	84.31467372	84.3
28	20	93.98547957	94.0
28	25	60.59166682	60.5
30	3	105.63923539	105.6
30	6	72.62142323	73.6
30	16	78.68377450	78.7
30	23	103.49593196	103.5
31	23	112.11161013	112.1
31	28	65.39791676	65.5

Table 5.1: Predicted and computed growth rates for various modes using the equilibrium initial condition. The predicted growth rate is denoted by λ_A and the computed growth rate is denoted by λ_n.

In two dimensions, this definition is used to define a spectrally accurate quadrature rule. The nonzero eigenvalues for this formulation are given by

$$\lambda_3, \lambda_4 = \sqrt{-g|k|\tilde{\rho}}. \tag{5.49}$$

This approach has a computational drawback. The integrand of the integral in Equation 5.48 cannot be factored into a product of source and field terms. We must break up the integral into two factorable integrals to perform the calculation using the multipole method. Hence, this approach doubles the amount of computational time needed to compute μ. For the equilibrium initial condition, this formulation gives the exact solution to 15 significant digits.

Numerical Viscosity

An alternative approach for stabilizing the method is the introduction of numerical viscosity. We modify the governing equations by adding a viscous damping term or terms in such way that the real parts of all eigenvalues are less than zero. In this case, we modify the equation which defines the time rate of change of z. The modified equation is

$$\frac{\partial z}{\partial t} = \tilde{w} + \nu \nabla^2 z. \tag{5.50}$$

The non zero eigenvalues are then given by

$$\lambda_3, \lambda_4 = \frac{|\mathbf{k}|^2}{4}\left(C\tilde{\rho}(\bar{\rho} - \tilde{\rho}) - 2\nu\right) \pm \frac{1}{4}\sqrt{|\mathbf{k}|^4\left(C\tilde{\rho}^2(\bar{\rho} - \tilde{\rho}) - 2\nu\right)^2 - 16g|k|\tilde{\rho}}. \tag{5.51}$$

A straightforward computation shows that for $C = 1$, $N = 64$, and $\nu > .1164$, all of the eigenvalues have negative real part. We have used the value $\nu = .12$ and have computed solutions for the equilibrium initial conditions using an implicit time stepping algorithm. While the method is stable, this approach has several side effects which must be considered. Because the eigenvalues have negative real part, we expect this method to be somewhat dissipative. In addition, our numerical experiments show that the addition of the viscous term necessitates a much smaller

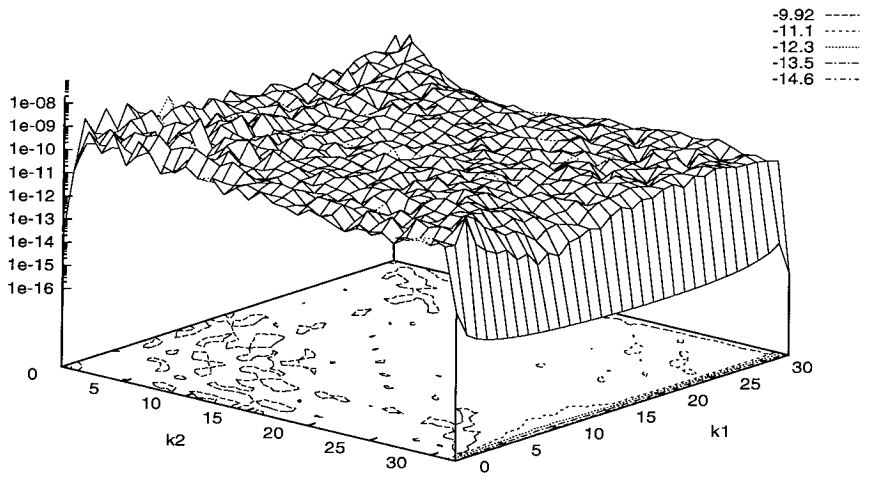

Figure 5.12: Spectrum of the equilibrium solution at time $t = .45$ using numerical viscosity.

time step for the implicit iteration to converge. We found that Δt must be on the order of 10^{-3} for the iteration to converge. A final consideration is that as N increases, the minimum value of ν which makes the scheme stable does not change very much. Thus, we don't expect the numerical stiffness to decrease as N increases. In Figure 5.12 we show the spectrum of the equilibrium solution computed at time $t = 429\Delta t = .45$, corresponding to a time step of $\Delta t = \pi/3000$. In Figure 5.13 we show the amplitude of the mode $(30, 0)$ as a function of time.

Fourier Smoothing

In the two-dimensional case, Fourier smoothing can be used to satisfy the compatibility condition given by Equation 5.40 in order to stabilize the method. This approach is outlined in [10] and [3]. For the linear case, this is easily generalized to three dimensions. The smoothing factor is determined by $\tilde{\rho}$ and $\bar{\rho}$. Specifically, we define the smoothed interface variable $\mathbf{x}^p$ so that

$$\mathbf{x}^p = (x_1 + (x_1')^p, x_2 + (x_2')^p, (z')^p) \tag{5.52}$$

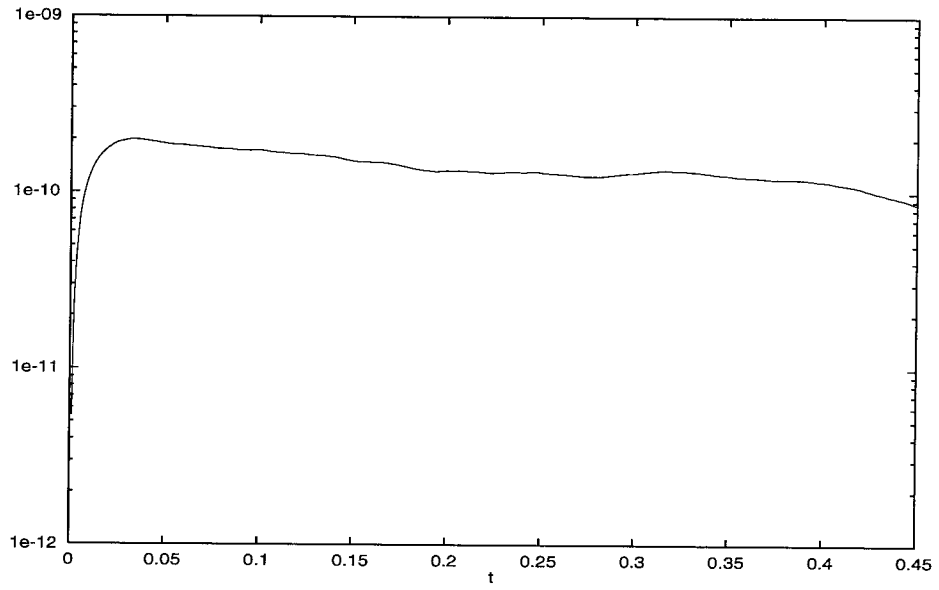

Figure 5.13: Amplitude of the mode $(30, 0)$ as a function of time with numerical viscosity.

where

$$(\hat{x}_1')_k^p = \frac{\tilde{\rho}}{\bar{\rho}}(\hat{x}_1')_k \quad (\hat{x}_2')_k^p = \frac{\tilde{\rho}}{\bar{\rho}}(\hat{x}_2')_k \quad (\hat{z}')_k^p = \frac{\tilde{\rho}}{\bar{\rho}}(\hat{z}')_k. \tag{5.53}$$

Note that we only apply the smoothing to the periodic part of the interfacial variable. Thus, to apply the smoothing to z' we transform z' to Fourier space, multiply by $\tilde{\rho}/\bar{\rho}$, then apply the inverse transformation. In Figure 5.14 we show the smoothing factor $\tilde{\rho}/\bar{\rho}$. As we expect, the smoothing factor deviates most from 1 for large values of $|\mathbf{k}|$. Hence, modes with low wave numbers are not modified substantially while many of the high wave number modes are modified significantly. To make the scheme stable, we modify the governing equations using the definition of $\mathbf{x}^p$:

$$\mu_\mathbf{i} = 2\phi_\mathbf{i} + \frac{1}{2\pi}\sum_{\mathbf{i}\neq\mathbf{j}} \mu_\mathbf{j}\hat{\mathbf{n}}_\mathbf{j} \cdot \nabla_\mathbf{x} G(\mathbf{x_i}^p - \mathbf{x_j}^p)h^2 \tag{5.54}$$

$$\boldsymbol{\kappa}_\mathbf{i} = -D_{x_1h}\mu_\mathbf{i} D_{x_2h}\mathbf{x_i} + D_{x_2h}\mu_\mathbf{i} D_{x_1h}\mathbf{x_i} \tag{5.55}$$

$$\tilde{\mathbf{u}}_\mathbf{i} = \frac{1}{4\pi}\sum_{\mathbf{i}\neq\mathbf{j}} \boldsymbol{\kappa}_\mathbf{j} \times \nabla_\mathbf{x} G(\mathbf{x_i}^p - \mathbf{x_j}^p)h^2 \tag{5.56}$$

$$\mathbf{u}_\mathbf{i} = \tilde{\mathbf{u}}_\mathbf{i} - \frac{1}{2}\frac{\boldsymbol{\kappa}_\mathbf{i} \times \hat{\mathbf{n}}_\mathbf{i}}{D_{x_1h}\mathbf{x_i} \times D_{x_2h}\mathbf{x_i}} \tag{5.57}$$

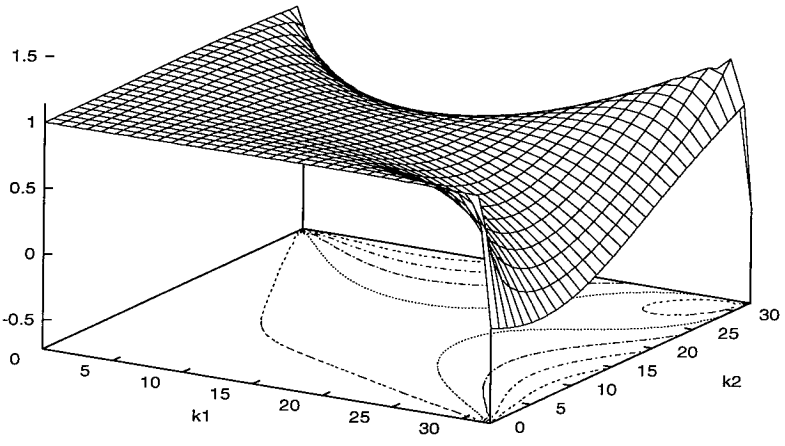

Figure 5.14: The Fourier smoothing factor $\tilde{\rho}/\bar{\rho}$.

$$\frac{d\mathbf{x_i}}{dt} = \tilde{\mathbf{u}}_\mathbf{i} \tag{5.58}$$

$$\frac{d\phi_i}{dt} = \tilde{\mathbf{u}}_\mathbf{i} \cdot \mathbf{u_i} - \frac{1}{2}\mathbf{u_i} \cdot \mathbf{u_i} - \mathbf{g} \cdot \mathbf{x_i}. \tag{5.59}$$

When we linearize this system, we find that all terms are the same except the term involving Λ_h. The linearized equations to leading order are now defined by

$$\frac{d\mathbf{x_i}'}{dt} = (0, 0, R_h(\phi_\mathbf{i}') + \frac{C}{2}R_h\Lambda_h(z_\mathbf{i}')^p - \frac{C}{2}R_hR_h(z_\mathbf{i}')) \tag{5.60}$$

$$\frac{d\phi_\mathbf{i}'}{dt} = -gz_\mathbf{i}'. \tag{5.61}$$

We compute the Fourier transform of $\Lambda_h(z')^p$ as follows:

$$(\widehat{\Lambda_h(z')^p})_k = |k|\bar{\rho}(\hat{z}')_k^p = |k|\bar{\rho}\frac{\tilde{\rho}}{\bar{\rho}}\hat{z}'_k = |k|\tilde{\rho}\hat{z}'_k. \tag{5.62}$$

The eigenvalues of the modified linear system are given by

$$\lambda_1, \lambda_2 = 0 \quad \text{and} \quad \lambda_3, \lambda_4 = \sqrt{-\tilde{\rho}g|k|}. \tag{5.63}$$

This analysis suggests that the modified method is stable. We have applied

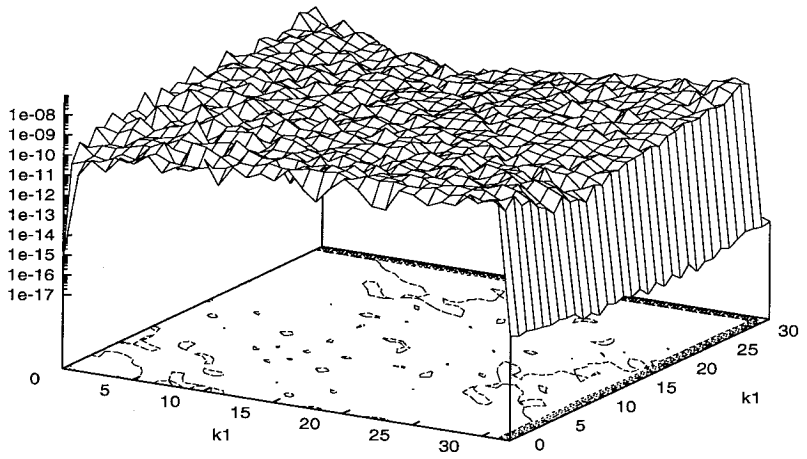

Figure 5.15: Spectrum of the solution for the equilibrium initial condition at time $t = 10\Delta t$ for $C = 1$ using Fourier smoothing.

the modified method using the equilibrium initial conditions (Equation 5.44) with $\Delta t = \pi/300$. We have computed the solution for a few hundred time steps, and observe that the method is now stable. In Figure 5.15 we show the spectrum of the solution after 10 time steps. In Figure 5.16 we show the spectrum after 500 time steps. This corresponds to $t = 5\pi/3 \simeq 5.23$. We show in Figure 5.17 the amplitude of the mode $(30, 3)$ with both regular and modified methods.

5.2 Non-Linear Stability

In this section we apply the methods of the previous section to a nonlinear problem. As an example of a nonlinear problem, we have computed a two-dimensional Stokes wave using a three-dimensional vortex sheet. The wave has amplitude .23 which is about 53% of the maximum amplitude for Stokes waves. We choose this type of wave because it has a steady profile, and because we know (theoretically) the solution for all time. We have found some numerical difficulties associated with wave numbers $\mathbf{k} = (k_1, k_2)$ where k_1 of k_2 equals $N/2$ or $N/2 - 1$. For the $N/2$ mode numbers, the difficulty probably arises as a result of aliasing due to the discrete Fourier transform.

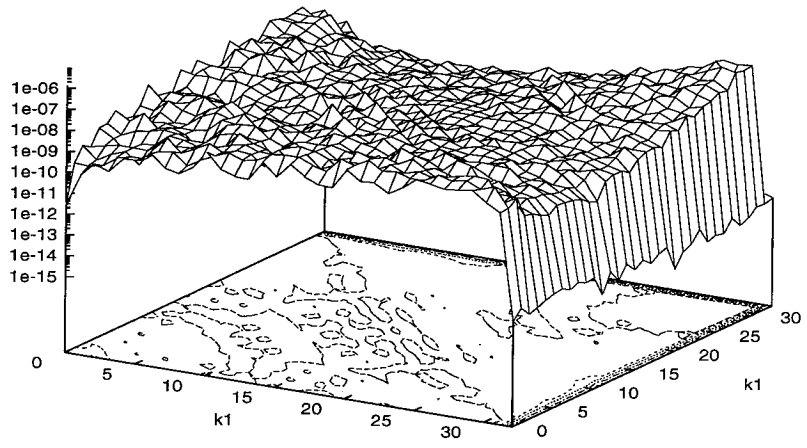

Figure 5.16: The spectrum of solution for the equilibrium initial condition at $t = 500\Delta t$ for $C = 1$ using Fourier smoothing.

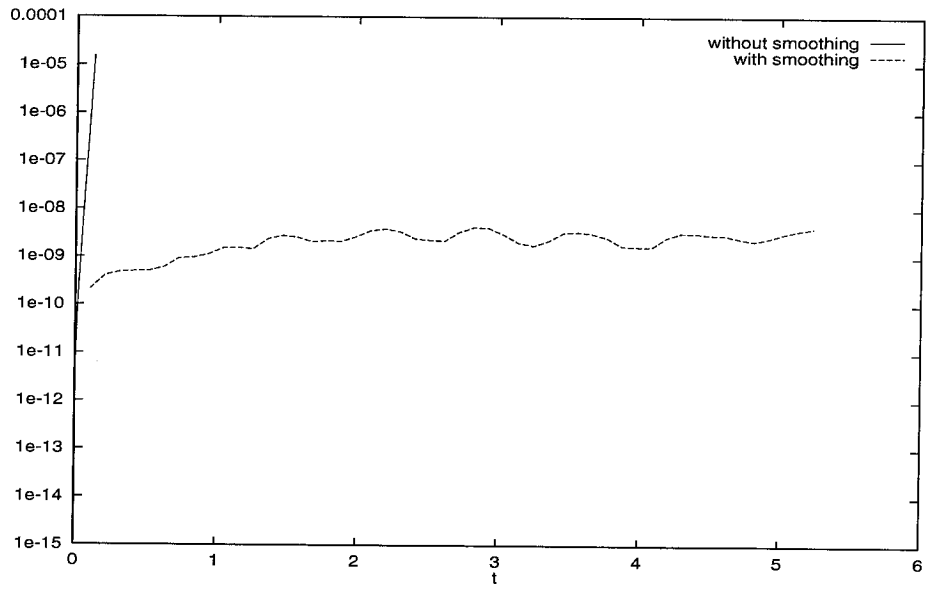

Figure 5.17: The amplitude of mode $(30, 3)$ with Fourier smoothing and without Fourier smoothing.

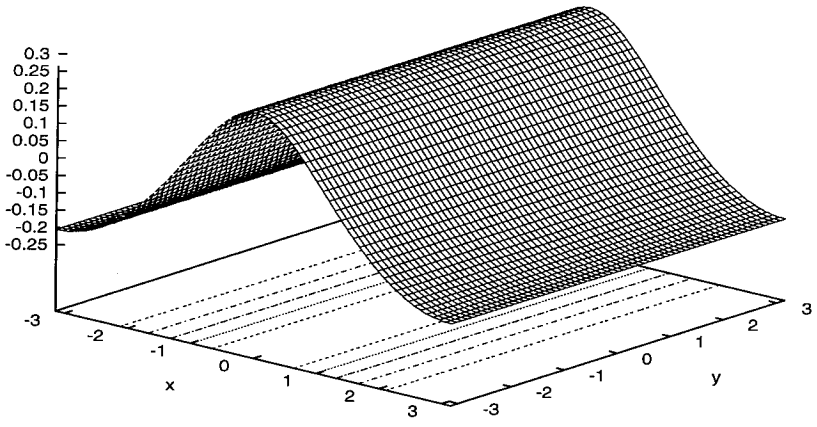

Figure 5.18: The initial wave profile for the Stokes waves.

For the $N/2-1$ mode numbers, the cause is less certain. It may also be due to aliasing errors. For all of our experiments in this section, we have explicitly removed these modes by Fourier filtering.

The Stokes wave solution is given by

$$\eta(x_1, x_2, t) = \sum_{j=0}^{n} a_j \cos(jx - ct) \tag{5.64}$$

$$\phi(x_1, x_2, t) = \sum_{j=1}^{n} b_j \sin(jx - ct) \exp(j\eta(x_1, x_2, t)). \tag{5.65}$$

For this particular amplitude, the wave speed c is approximately 1.02815. In Table 5.2 we give the first few values of a_j and b_j.

To demonstrate the instability, we use the fixed grid formulation with $N = 64$. We show the wave and wave spectrum at time $t = 0$ in Figures 5.18 and 5.19, respectively. In Figure 5.20 we show the spectrum of η at time $t = 35\Delta t \simeq .37$. In Figure 5.21 we show the wave profile at $t = 47\Delta t \simeq .49$. In Figure 5.22 we show the wave spectrum at $t = 47\Delta t$. As in the linear case, the high modes grow most rapidly. As the high frequency modes grow, more iterations are required to compute μ to within the required tolerance.

j	a_j	b_j
1	0.142933126659572E-21	0.000000000000000E-00
2	2.297779080059771E-01	2.278158324123190E-01
3	0.287217280543541E-01	0.156618016002308E-02
4	0.549751997819685E-02	0.826183380501244E-04
5	0.125813502941109E-02	0.508807495561693E-05
6	0.317766353303841E-03	0.338651044021795E-06
7	0.854325834662375E-04	0.239847505940169E-07
8	0.239808534629156E-04	0.178452706194413E-08
9	0.694707126702719E-05	0.137837218524794E-09
10	0.206152843897577E-05	0.109542630340699E-10
11	0.623464762469231E-06	0.890174655716639E-12
12	0.191465395356647E-06	0.736495331036325E-13
13	0.595469270779001E-07	0.618192187054263E-14
14	0.187169483334039E-07	0.527809033082833E-15
15	0.593650374261307E-08	0.405337834594774E-16
16	0.189759103973911E-08	0.105257671618492E-16
17	0.610678441909494E-09	-0.479315894205654E-17
18	0.197698146299351E-09	0.556527183698467E-18
19	0.643392907937666E-10	0.349248504423153E-17
20	0.210370808159800E-10	-0.505691133547518E-17

Table 5.2: Coefficients defining a Stokes wave.

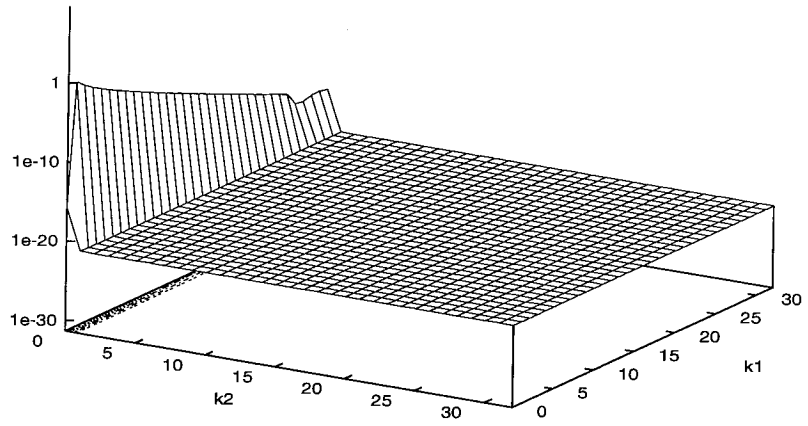

Figure 5.19: The initial wave spectrum for the Stokes waves.

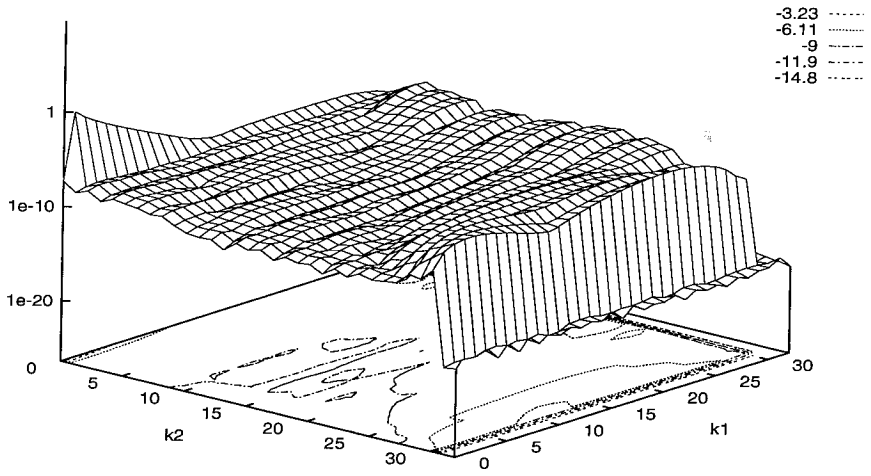

Figure 5.20: The Stokes wave spectrum for $t = 35\Delta t \simeq .37$.

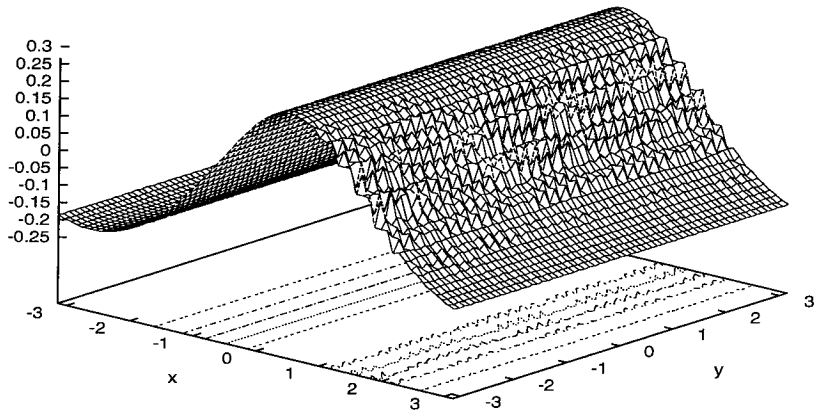

Figure 5.21: The Stokes wave profile for $t = 47\Delta t \simeq .49$.

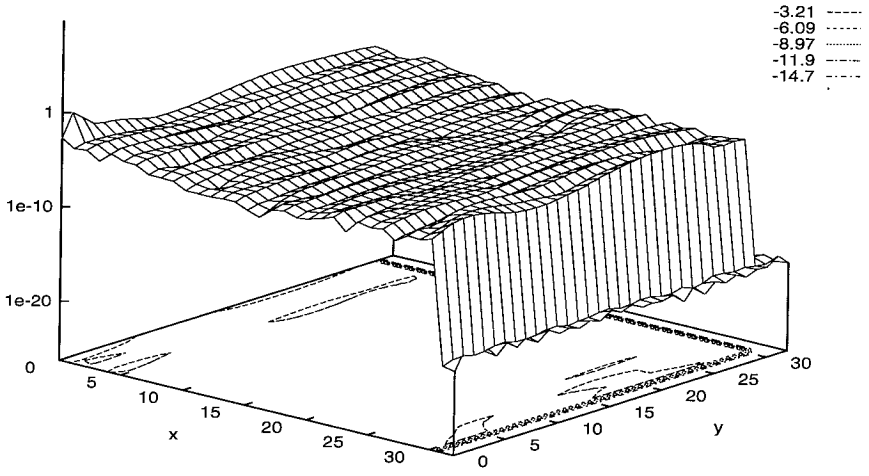

Figure 5.22: The Stokes wave spectrum for $t = 47\Delta t$.

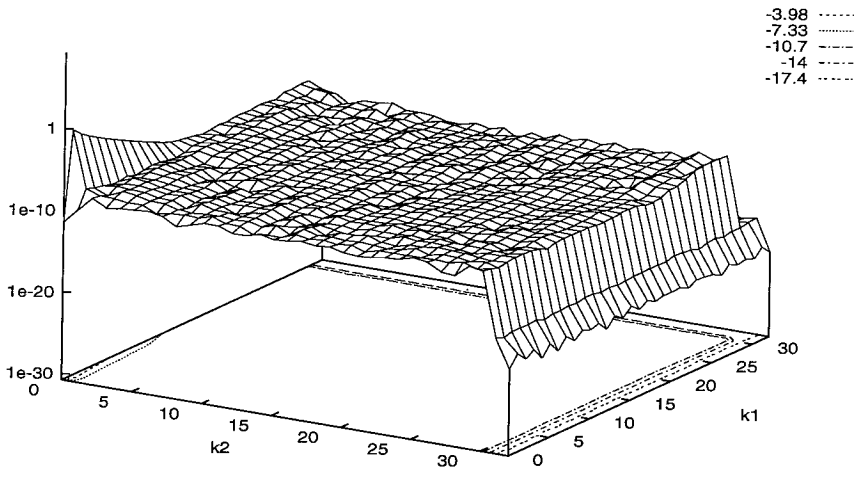

Figure 5.23: The Stokes wave spectrum for $t = 9.42$ using the desingularization method.

In the next sections, we discuss the application of the stabilization techniques previously discussed to the case of the Stokes wave.

5.2.1 Desingularization

The desingularization technique significantly stabilizes the numerical scheme. We show in Figure 5.23 the spectrum of the wave after approximately 1.5 periods. This corresponds to $t = 9.42$. In Figure 5.24 we show the amplitude of mode $(30, 30)$ as a function of time. There are also other modes which grow in time, particularly on the line $k_2 = 30$. While the results for this particular problem are encouraging, it is not clear how effective this approach will be for general three-dimensional problems. We note that these results are very similar to the results we report for the Fourier smoothing approach using the equilibrium smoothing factor, suggesting that in this case, the instability is similar to that of the equilibrium case.

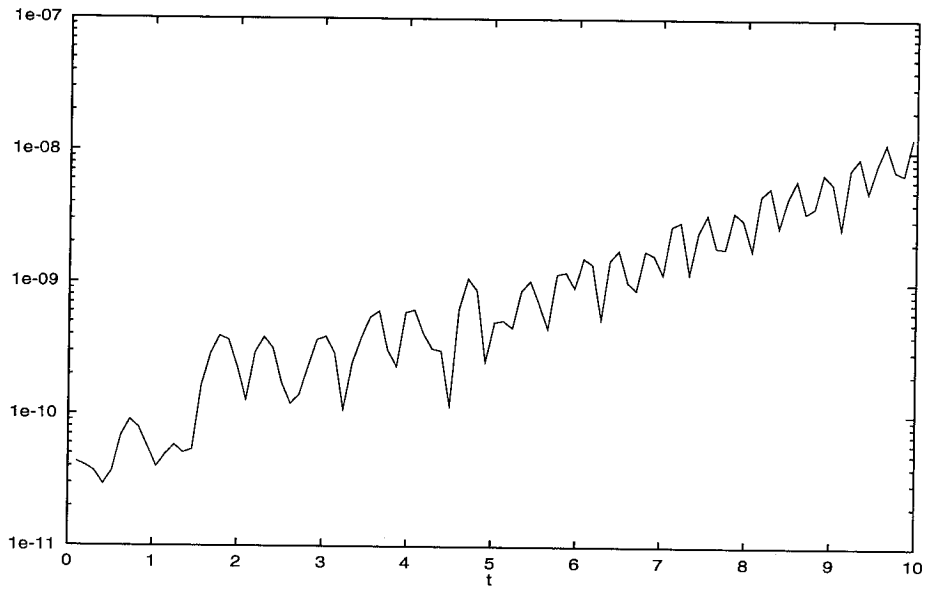

Figure 5.24: The amplitude of the mode $(30, 30)$ for the Stokes wave as a function of time, using the desingularization method.

5.2.2 Numerical Viscosity

We have investigated the effect of the viscous damping using the Stokes wave initial condition. Because of the stiffness of the numerical scheme, we did not propagate the wave for a long time. However, even in the short time for which we propagated the wave, we see the stabilizing effect of the viscous term . We used time step $\Delta t = \pi/2000$ and $\nu = .12$. In this particular case, we did not explicitly filter any modes. In Figure 5.25 we show the computed solution after 172 time steps, $t \simeq .27$. By comparison, the original Stokes wave calculation showed significant growth by this time in the calculations. As an example, we plot the amplitude of the mode $(30, 3)$ as a function of time. The growth appears to be at worst linear in time. While it is difficult to generalize for such a short time period, the method appears to be stable. We do not detect any modes growing exponentially. The dissipative effect of the scheme can be seen by observing the mean wave height as a function of time, as plotted in Figure 5.27. For the time period under consideration, the mean wave height has decreased approximately 6%.

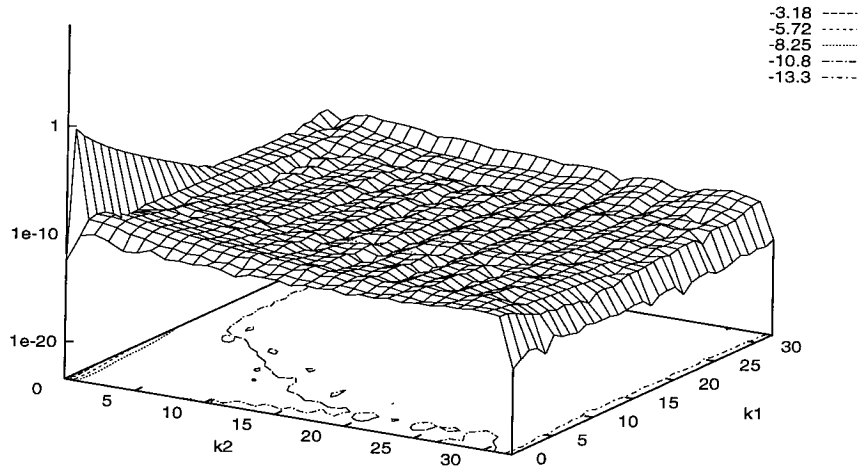

Figure 5.25: The Stokes wave spectrum at $t = .27$ using numerical viscosity.

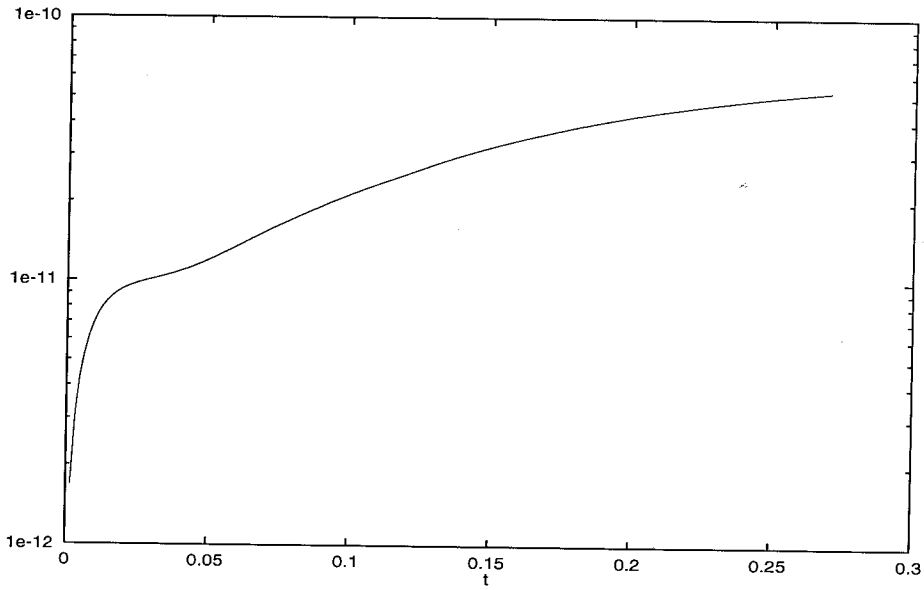

Figure 5.26: The amplitude of the mode $(30, 3)$ for the Stokes wave using numerical viscosity.

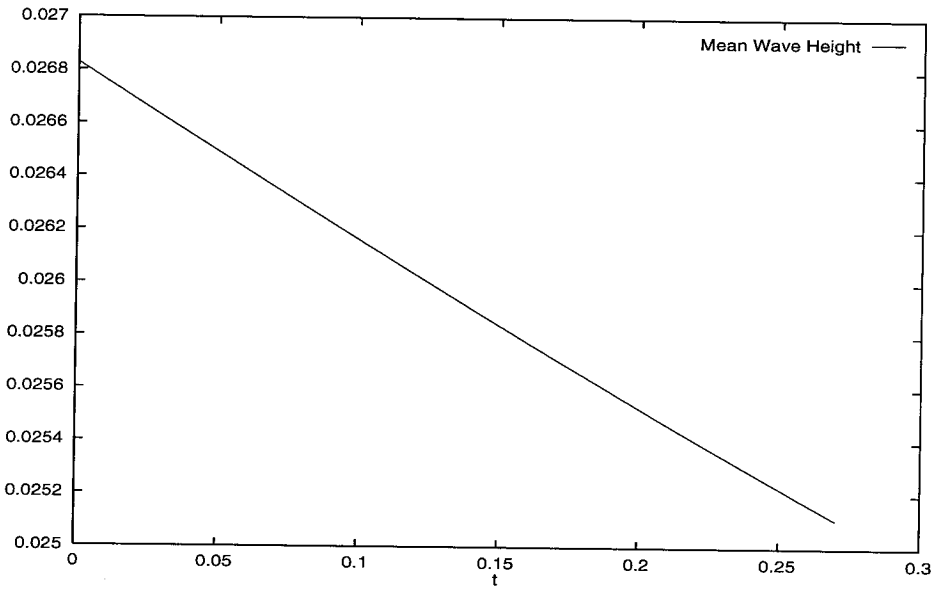

Figure 5.27: The mean wave height for the Stokes wave using numerical viscosity.

5.2.3 Fourier Smoothing

The Fourier smoothing approach for finite amplitude waves in three dimensions is considerably more complicated than for the two-dimensional case. The stability of the continuous boundary integral formulation for three-dimensional water waves far from equilibrium has been examined by Hou, Teng, and Zhang in [13]. That analysis suggests that the leading order instability in the discrete case is defined in terms of operators H_{lh} and Λ_h where H_{lh} and Λ_h are redefined as follows:

$$H_{lh}(f_{\mathbf{i}}) = \frac{h^2}{2\pi} \sum_{\mathbf{i}\neq\mathbf{j}} \frac{(x_{li} - x_{lj}) f_{\mathbf{i}}}{\left| |D_{x_{1h}} \mathbf{x_i}| \, (x_{1\mathbf{i}} - x_{1\mathbf{j}}) + |D_{x_{2h}} \mathbf{x_i}| \, (x_{2\mathbf{i}} - x_{2\mathbf{j}}) \right|^3} \tag{5.66}$$

$$\Lambda_h(f_{\mathbf{i}}) = \frac{h^2}{2\pi} \sum_{\mathbf{i}\neq\mathbf{j}} \frac{f_{\mathbf{i}} - f_{\mathbf{j}}}{\left| |D_{x_{1h}} \mathbf{x_i}| \, (x_{1\mathbf{i}} - x_{1\mathbf{j}}) + |D_{x_{2h}} \mathbf{x_i}| \, (x_{2\mathbf{i}} - x_{2\mathbf{j}}) \right|^3}. \tag{5.67}$$

This form suggests that the appropriate Fourier smoothing factor needs to be computed for each point on the grid and possibly at each time step. Since this would not be practical, we seek to define a surface parameterization $\beta = (\beta_1, \beta_2)$ such that

$$\frac{\partial \mathbf{x}}{\partial \beta_1} \cdot \frac{\partial \mathbf{x}}{\partial \beta_2} = 0 \tag{5.68}$$

$$\left|\frac{\partial \mathbf{x}}{\partial \beta_1}\right|^2 = \lambda_\beta \left|\frac{\partial \mathbf{x}}{\partial \beta_2}\right|^2 \tag{5.69}$$

where λ_β is independent of β. The operators H_{lh} and Λ_h take the form

$$H_{lh}(f_\mathbf{i}) = \frac{1}{2\pi} \sum_{\mathbf{i}\neq\mathbf{j}} \frac{(\beta_{l\mathbf{i}} - \beta_{l\mathbf{j}}) f_\mathbf{i}}{(\lambda_\beta(\beta_{1\mathbf{i}} - \beta_{1\mathbf{j}})^2 + (\beta_{2\mathbf{i}} - \beta_{2\mathbf{j}})^2)^{3/2}} h^2 \tag{5.70}$$

$$\Lambda_h(f_\mathbf{i}) = \frac{1}{2\pi} \sum_{\mathbf{i}\neq\mathbf{j}} \frac{f_\mathbf{i} - f_\mathbf{j}}{(\lambda_\beta(\beta_{1\mathbf{i}} - \beta_{1\mathbf{j}})^2 + (\beta_{2\mathbf{i}} - \beta_{2\mathbf{j}})^2)^{3/2}} h^2. \tag{5.71}$$

We then compute the smoothing factor as in the linear case, where we modify the definitions of ρ^1, ρ^2, and $\bar{\rho}$ as follows:

$$\rho^1(\mathbf{k}h, \lambda_\beta) = \frac{1}{2\pi} \sum_{(j_1,j_2)\neq(0,0)} \frac{j_1 \sin(j_1 k_1 h)\cos(j_2 k_2 h)}{(\lambda_\beta j_1^2 + j_2^2)^{3/2}} \tag{5.72}$$

$$\rho^2(\mathbf{k}h, \lambda_\beta) = \frac{1}{2\pi} \sum_{(j_1,j_2)\neq(0,0)} \frac{j_2 \sin(j_2 k_2 h)\cos(j_1 k_1 h)}{(\lambda_\beta j_1^2 + j_2^2)^{3/2}} \tag{5.73}$$

$$\bar{\rho}(\mathbf{k}h, \lambda_\beta) = \frac{1}{2\pi h|\mathbf{k}|} \sum_{(j_1,j_2)\neq(0,0)} \frac{1 - \cos(j_1 k_1 h)\cos(j_2 k_2 h)}{(\lambda_\beta j_1^2 + j_2^2)^{3/2}}. \tag{5.74}$$

We refer to a coordinate system satisfying Equations 5.68 and 5.69 as an equal-orthogonal coordinate system. A coordinate system which satisfies these conditions would eliminate the necessity of computing the smoothing factor at every time step. While the condition of orthogonality is readily imposed, the condition given by Equation 5.69 is more difficult to impose. To see this, we begin with the surface parameterization $\mathbf{x} = (x(x_1, x_2), y(x_1, x_2), \eta(x_1, x_2))$. We next consider x_1 and x_2 to be functions of β. Substituting into the equations given by 5.68 and 5.69 and applying the chain rule for differentiation gives

$$\begin{aligned} &\left|\frac{\partial \mathbf{x}}{\partial x_1}\right|^2 \frac{\partial x_1}{\partial \beta_1}\frac{\partial x_1}{\partial \beta_2} + \left|\frac{\partial \mathbf{x}}{\partial x_2}\right|^2 \frac{\partial x_2}{\partial \beta_1}\frac{\partial x_2}{\partial \beta_2} \\ &\quad + \frac{\partial \mathbf{x}}{\partial x_1}\cdot\frac{\partial \mathbf{x}}{\partial x_2}\left(\frac{\partial x_1}{\partial \beta_1}\frac{\partial x_2}{\partial \beta_2} + \frac{\partial x_1}{\partial \beta_2}\frac{\partial x_2}{\partial \beta_1}\right) = 0 \end{aligned} \tag{5.75}$$

$$\begin{aligned} &\left|\frac{\partial \mathbf{x}}{\partial x_1}\right|^2 \left(\left(\frac{\partial x_1}{\partial \beta_1}\right)^2 - \lambda_\beta\left(\frac{\partial x_1}{\partial \beta_2}\right)^2\right) + \left|\frac{\partial \mathbf{x}}{\partial x_2}\right|^2 \left(\left(\frac{\partial x_2}{\partial \beta_1}\right)^2 - \lambda_\beta\left(\frac{\partial x_2}{\partial \beta_2}\right)^2\right) \\ &\quad + 2\left(\frac{\partial \mathbf{x}}{\partial x_1}\cdot\frac{\partial \mathbf{x}}{\partial x_2}\right)\left(\frac{\partial x_1}{\partial \beta_1}\frac{\partial x_2}{\partial \beta_1} - \lambda_\beta\frac{\partial x_1}{\partial \beta_2}\frac{\partial x_2}{\partial \beta_2}\right) = 0 \end{aligned} \tag{5.76}$$

with periodic boundary conditions in β. The solution set of this system is not trivial.

An additional complication is that λ_β may be a function of time. To see this, we consider Equation 5.69 in the case where the surface does not vary in x_2. In this case, the equation simplifies to

$$\left|\frac{\partial \mathbf{x}}{\partial \beta_1}\right|^2 = \lambda_\beta. \tag{5.77}$$

If we now integrate over the surface, we conclude that $\lambda_\beta = L$ where L is the arc length of any curve on the surface such that x_2 is held constant. Hence, for C to be constant in time, the arc length must be constant in time. Allowing λ_β to be a function of time permits stretching of the vortex sheet. This permits, for example, overturning waves. It may be necessary, therefore, to compute the smoothing factor more than once throughout the computation.

It is unclear whether an equal-orthogonal coordinate system can always be constructed. Fortunately, we are able to construct such a coordinate system for the Stokes wave. Later in this chapter we discuss the details of computing the coordinate system and the associated smoothing factor.

5.2.4 Numerical Experiments

For our first experiment, we use the fixed grid formulation and apply the same smoothing factor computed for the linear problems. That is, we take $\lambda_\beta = 1$ in Equations 5.66 and 5.67. This formulation is considerably more stable than the method without smoothing. We have been able to propagate this wave for 4 periods, noting the onset of instability in a number of the high frequency modes. Since the fixed grid formulation gives an incorrect smoothing factor, we expect instabilities to occur. We show in Figures 5.28-5.31 the spectrum of the solution computed after 1,2,3, and 4 periods, respectively. The smoothing factor used eliminates most of the growing modes. We still detect growth (though at a slower rate) in several modes, particularly a number of modes where k_1 or k_2 has the value $N/2 - 2 = 30$. In Figure 5.32 we show the amplitudes of modes $(k_1, 30)$ where k_1 varies from 20 to

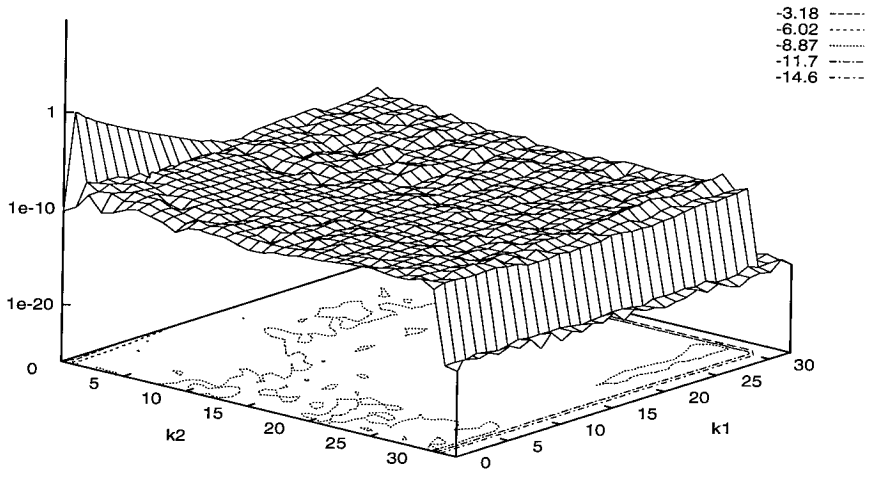

Figure 5.28: The spectrum of the Stokes wave after approximately one period using the smoothing factor from linear theory.

30.

Our next experiment uses the equal-orthogonal coordinate system and the associated smoothing factor to compute the same wave. The coordinate system is given by

$$x_1 = \beta_1 + S_1(\beta_1) \tag{5.78}$$

$$x_2 = \beta_2 \tag{5.79}$$

where S_1 is shown in Figure 5.33. In Figure 5.34, we compare the coordinate systems by projecting them onto the x-y plane. In Figure 5.35 we compare the ratio λ_x and λ_β for the Stokes wave initial condition where λ_x is defined by

$$\lambda_x(x_1, x_2) = \left|\frac{\partial \mathbf{x}}{\partial x_1}\right|^2 / \left|\frac{\partial \mathbf{x}}{\partial x_2}\right|^2. \tag{5.80}$$

For this wave, both coordinate systems are orthogonal. For this particular wave, we compute the value of λ_β to be approximately 1.0281. While using this coordinate system does stabilize most of the modes, we still see growth in modes on the line $k_1 =$

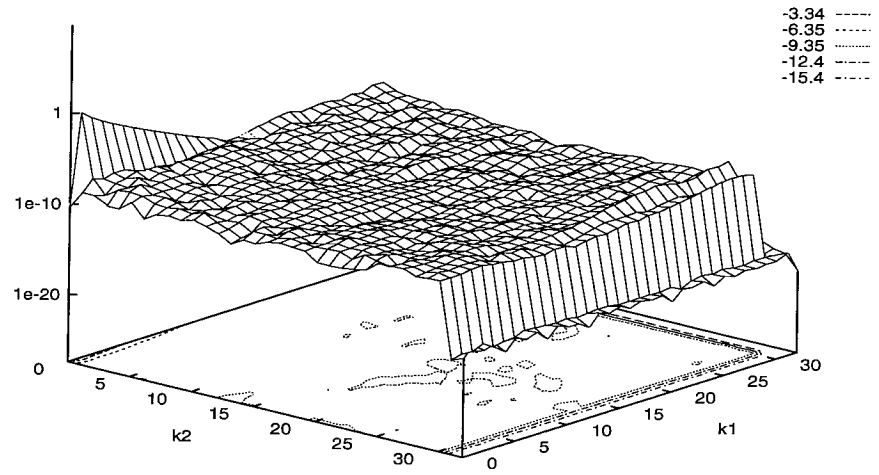

Figure 5.29: The spectrum of the Stokes wave after approximately two periods using the smoothing factor from linear theory.

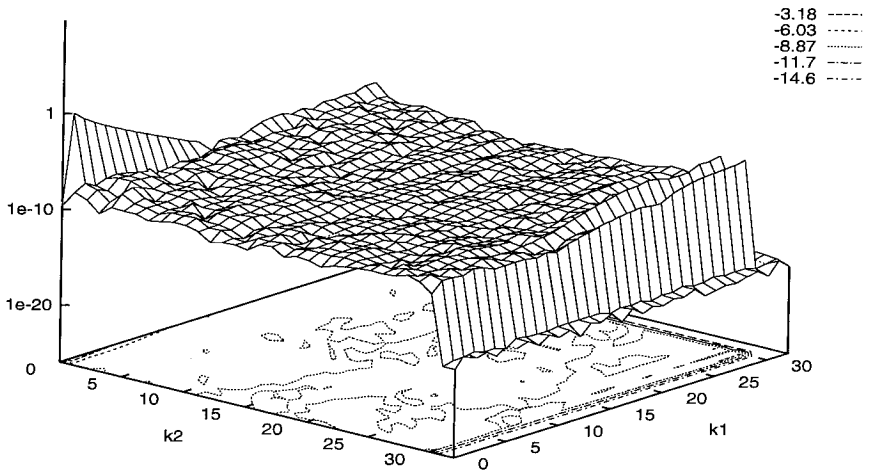

Figure 5.30: The spectrum of the Stokes wave after approximately three periods using the smoothing factor from linear theory.

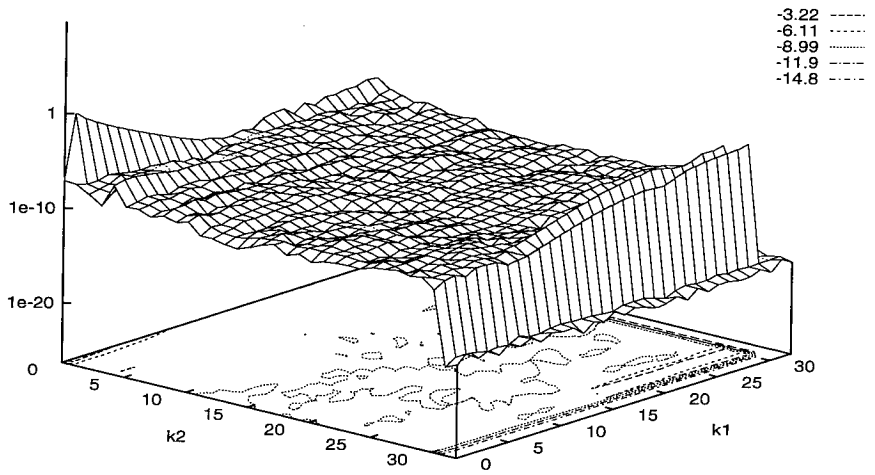

Figure 5.31: The spectrum of the Stokes wave after approximately four periods using the smoothing factor from linear theory.

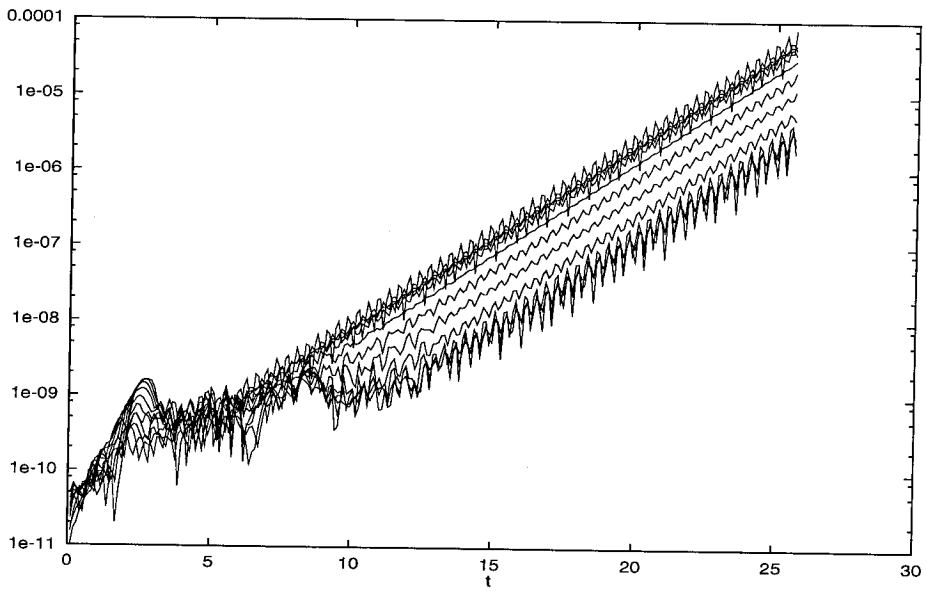

Figure 5.32: The amplitudes of modes $(k_1, 30)$ for k_1 from 20 to 30 plotted as function of time using the fixed coordinate system and the smoothing factor from linear theory.

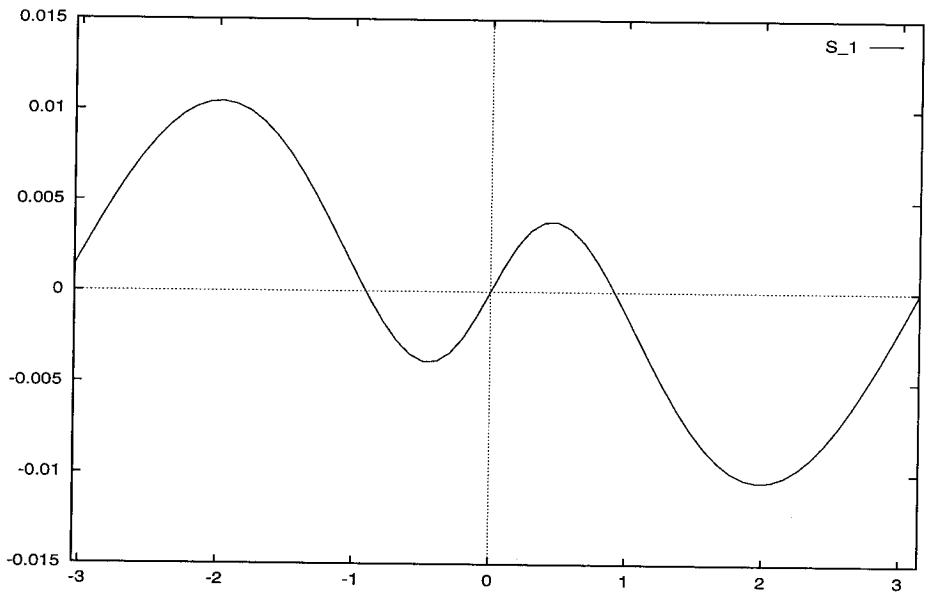

Figure 5.33: The function $S_1(\beta_1)$.

28. The cause of this instability is uncertain. The nonlinear theory suggests that even after the primary instability is removed, there could still be weaker instabilities. One possible method for controlling weaker instabilities is using Krasny filtering. This type of filter removes all modes with amplitude lower than some tolerance.

While we have not completely removed all of the instabilities, this example is significant in that it shows the effect that a slight change in the coordinate system and the smoothing factor can have on the calculations. One important consideration is that the accuracy of the multipole expansions for these calculations was roughly single precision. Higher precision (taking more terms in the multipole expansion) could have some effect on the growth of instabilities. This is because the stability theory assumes an exact Green's function kernel, while our method uses an approximate kernel. In Figures 5.36 - 5.39 we show the spectrum of the wave solution after 1, 2, 3, and 4 time steps, respectively. In Figure 5.40 we show the amplitudes of the mode $(k_1, 30)$ where k_1 varies from 20 to 30, as in the previous example. Comparing this figure with Figure 5.32 shows that these modes have been stabilized.

It is also useful to verify to what degree the equal-orthogonal condition is satis-

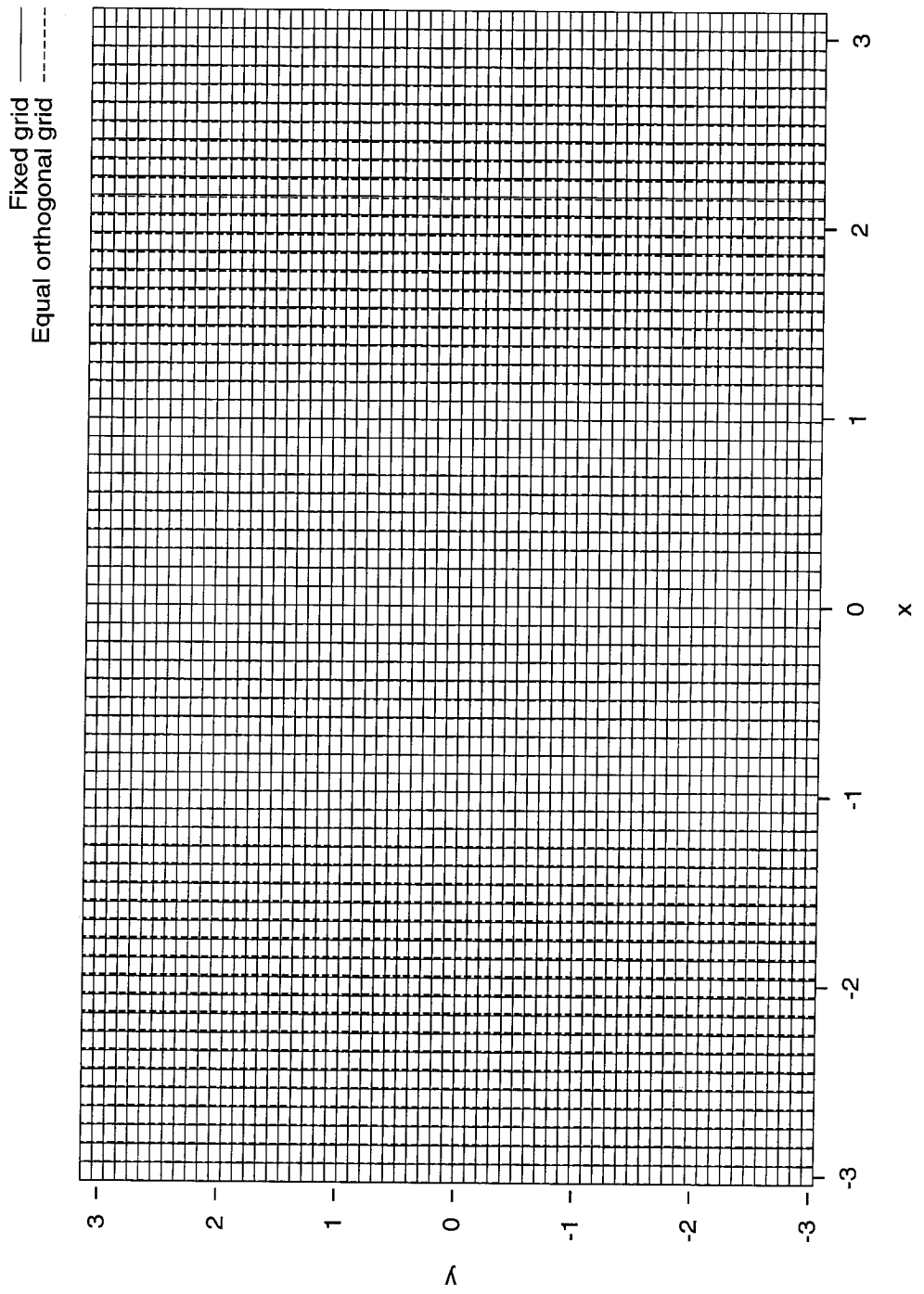

Figure 5.34: A comparison of the coordinate systems for the fixed grid formulation and the equal-orthogonal formulation as projected on the x-y plane. The fixed grid is represented by the solid line.

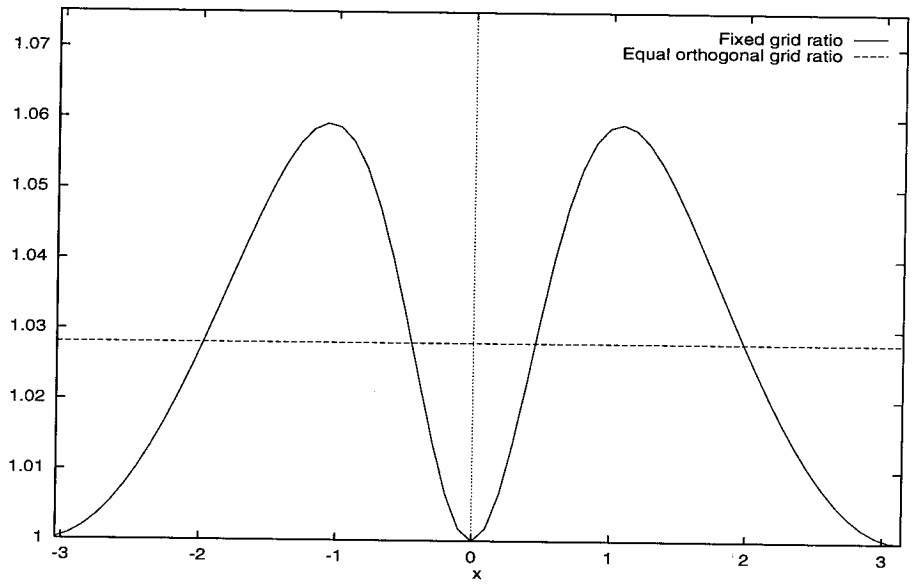

Figure 5.35: A comparison of the ratios λ_x and λ_β.

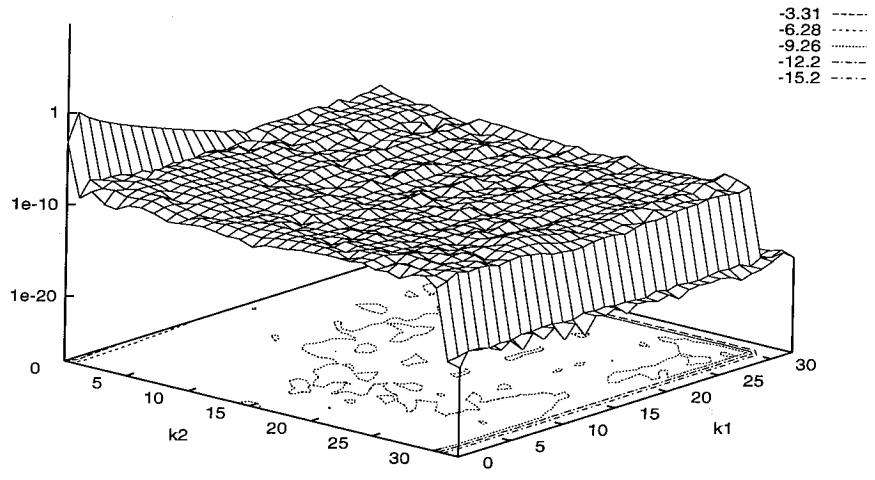

Figure 5.36: The spectrum of the Stokes wave after approximately one period using the equal-orthogonal coordinate system and smoothing factor.

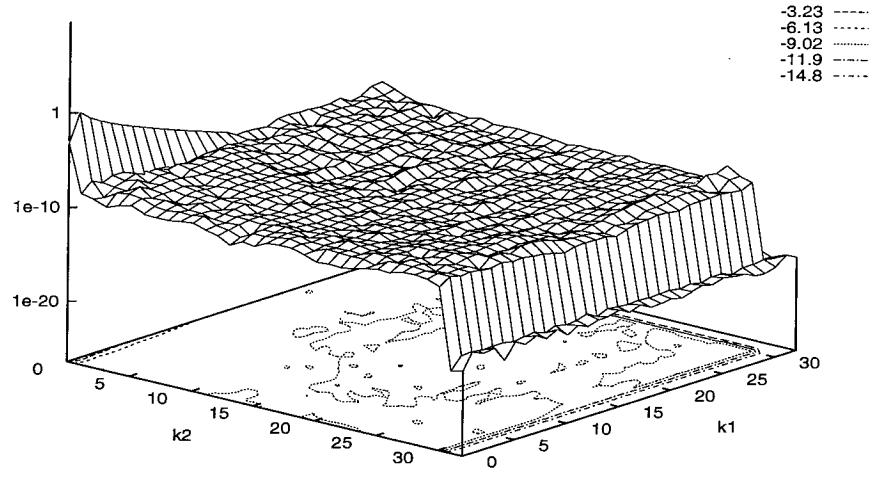

Figure 5.37: The spectrum of the Stokes wave after approximately two periods using equal-orthogonal coordinate system and smoothing factor.

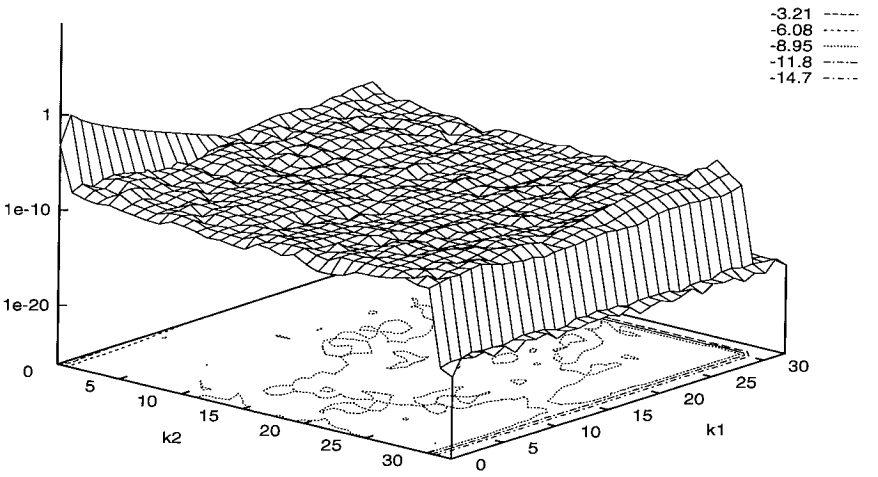

Figure 5.38: The spectrum of the Stokes wave after approximately three periods using the equal-orthogonal coordinate system and smoothing factor.

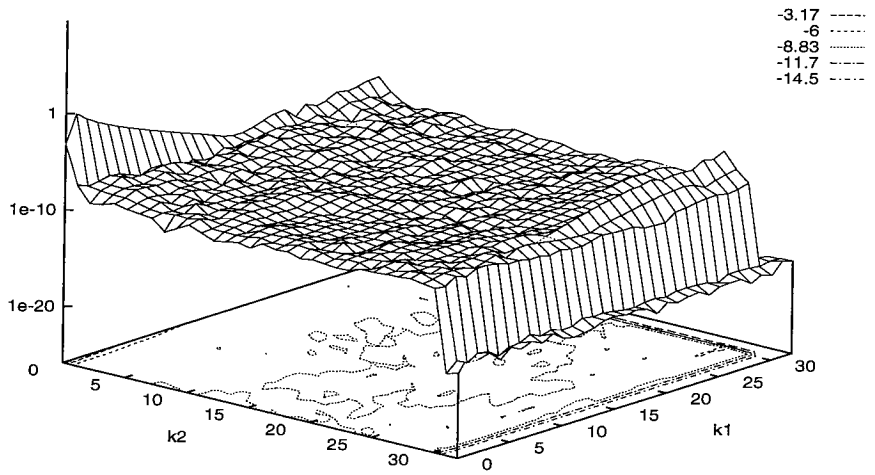

Figure 5.39: The spectrum of the Stokes wave after approximately four periods using the equal-orthogonal coordinate system and smoothing factor.

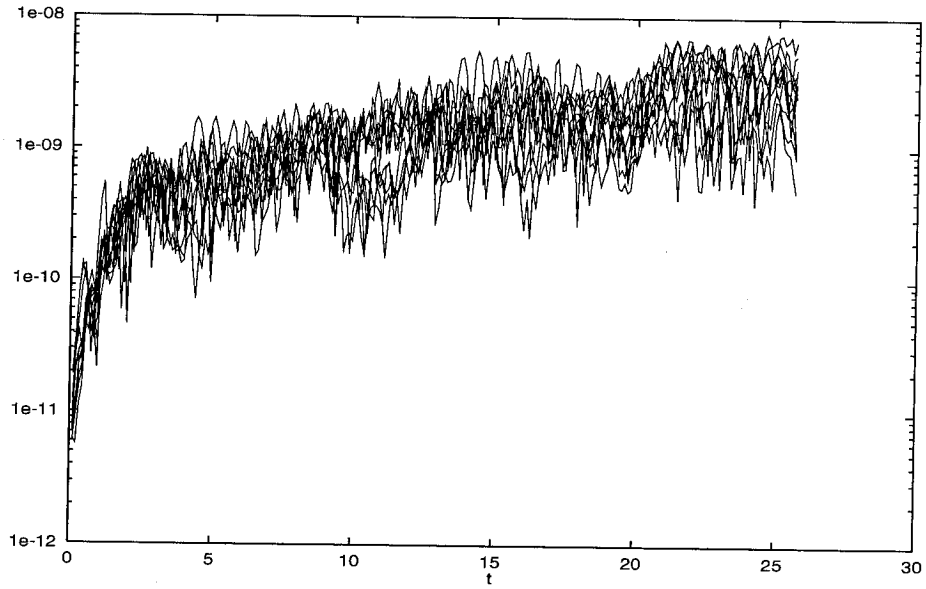

Figure 5.40: The amplitudes of modes $(k_1, 30)$ for k_1 from 20 to 30 plotted as function of time using the equal-orthogonal coordinate system and smoothing factor. Compare with Figure 5.32.

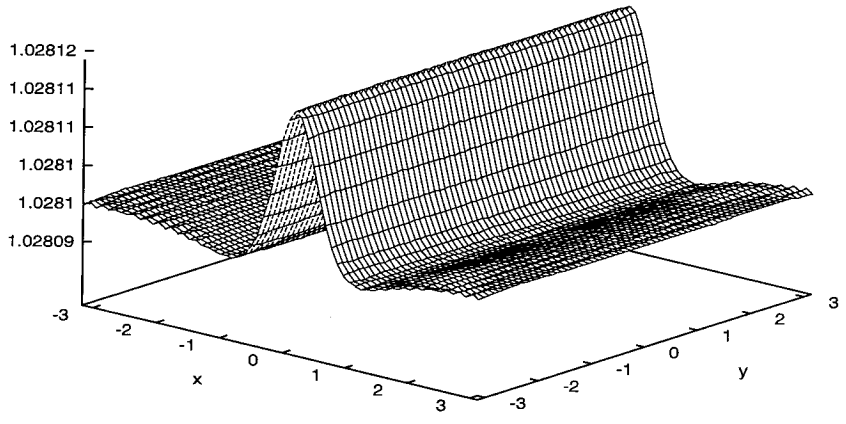

Figure 5.41: The computed value of λ_β after 1 period.

fied. We define γ_β to be

$$\gamma_\beta = \frac{\partial \mathbf{x}}{\partial \beta 1} \cdot \frac{\partial \mathbf{x}}{\partial \beta_2}. \tag{5.81}$$

We show the computed values of λ_β and γ_β after 1 and 4 periods in Figures 5.41 - 5.44.

In Figure 5.45 we show the wave profile after approximately 4 periods ($t = 26$). Because the wave solution is known for all time, we can compare the theoretical solution with the exact solution. We show the difference of exact and computed solutions after 1 period and after 4 periods in Figures 5.46 and 5.47.

5.3 Remarks

We have shown that for both the near-equilibrium and the non-linear regimes, the boundary integral scheme can be stabilized significantly. The numerical viscosity approach has the disadvantage of being particularly stiff. This approach, therefore, has a distinct computational disadvantage. We consider this method to be unsuitable for stabilizing the numerical scheme.

The desingularization approach works relatively well for the test problem we

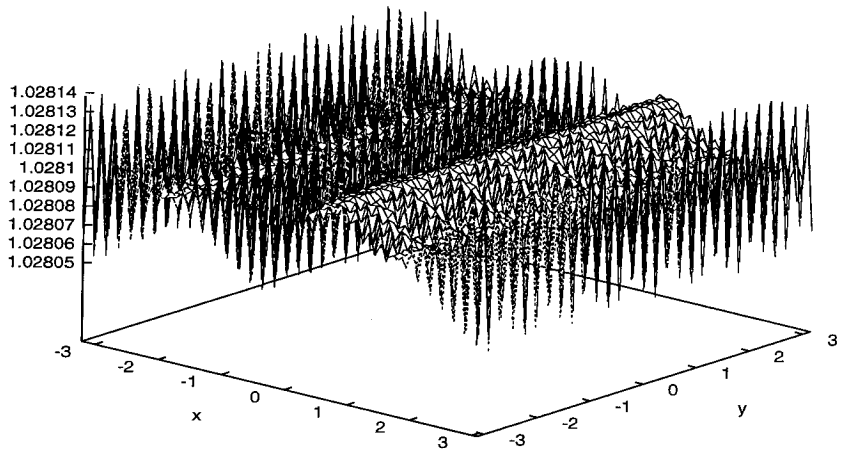

Figure 5.42: The computed value of λ_β after 4 periods.

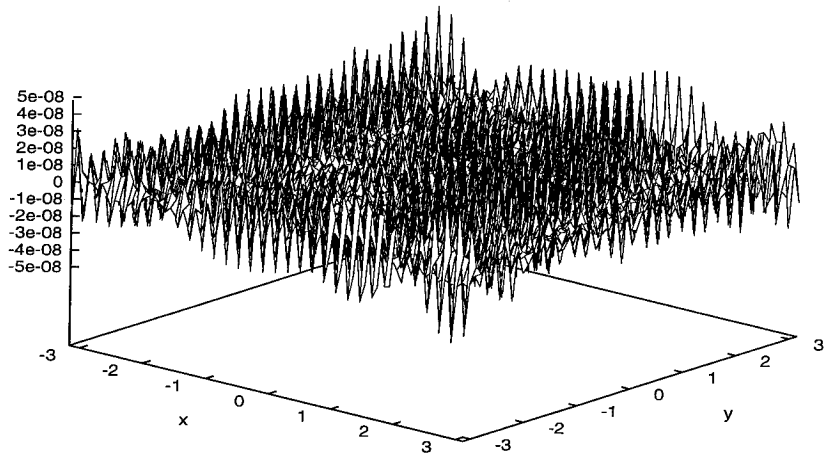

Figure 5.43: The computed value of γ_β after 1 period.

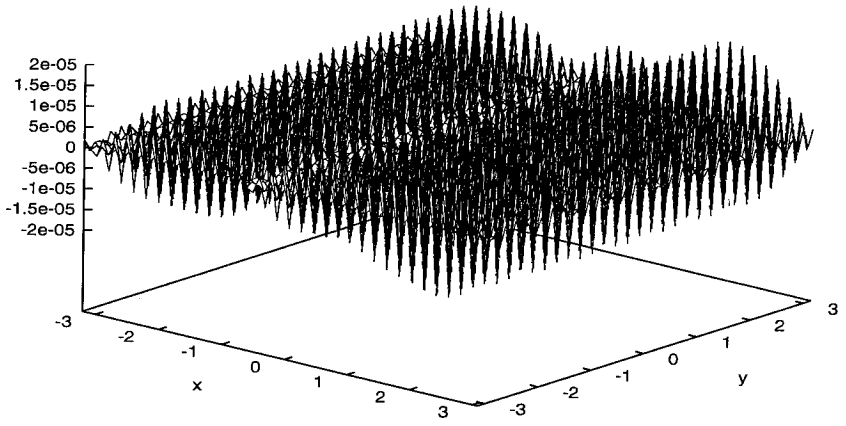

Figure 5.44: The computed value of γ_β after 4 periods.

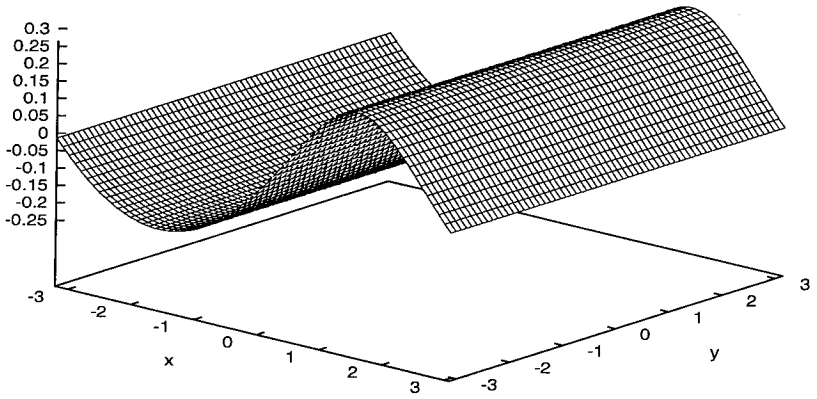

Figure 5.45: The Stokes wave profile after approximately 4 periods ($t = 26$).

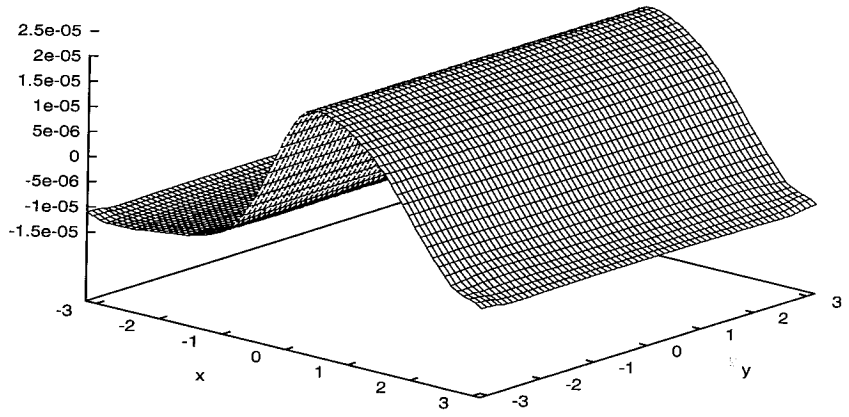

Figure 5.46: The Stokes wave error after 1 period.

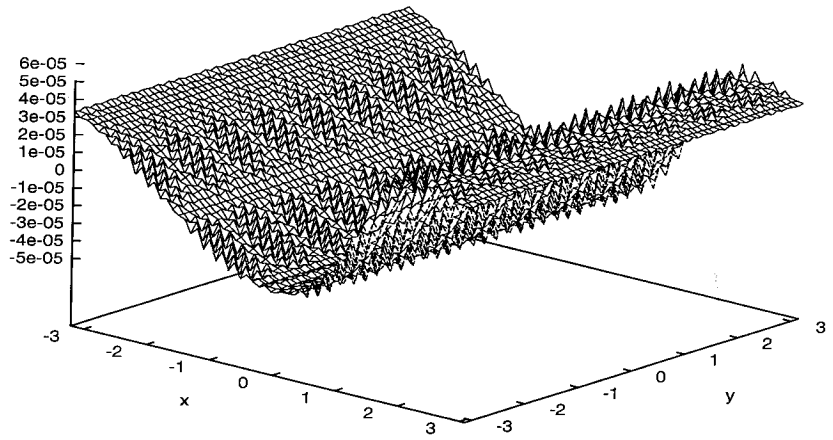

Figure 5.47: The Stokes wave error after 4 periods.

used. However, for this problem, the ratio λ_β varies by only a few percent over the surface. It is not certain how well this approach would work for a problem where λ_β varied more significantly over the surface such as for breaking waves. We also cannot tell how effective this approach will be for very large values of N. It is possible that this method moderates the $O(|\mathbf{k}|^2)$ instability but does not completely remove it. Hence, for large N, we could still see rapidly growing instabilities. In addition, this approach is more costly from a computational standpoint. The desingularization approach, however, is much simpler to implement.

The Fourier smoothing approach seems to have the best overall effect on the stability of the scheme when used with the appropriate coordinate system and smoothing factor. However, as we noted earlier, there could exist weaker instabilities that even this method does not remove. Moreover, it is not clear whether an equal-orthogonal coordinate system can be constructed for all possible initial conditions. A comparison of the results using Fourier smoothing both with and without the equal-orthogonal coordinate system suggests that for many problems, it may not be necessary to construct the equal-orthogonal coordinate system. However, for large N or for very complex surfaces (such as breaking waves), we expect this method to remove the high order instability.

5.4 Equal-Orthogonal Coordinate System

Initially we want to specify a parameterization of the surface $\mathbf{x}(\beta) = (x_1(\beta), x_2(\beta), z(\beta))$ where $\beta = (\beta_1, \beta_2)$ such that

$$\frac{\partial \mathbf{x}}{\partial \beta_1} \cdot \frac{\partial \mathbf{x}}{\partial \beta_2} = 0 \tag{5.82}$$

$$\frac{\partial \mathbf{x}}{\partial \beta_1} \cdot \frac{\partial \mathbf{x}}{\partial \beta_1} = C \frac{\partial \mathbf{x}}{\partial \beta_2} \cdot \frac{\partial \mathbf{x}}{\partial \beta_2}. \tag{5.83}$$

We need this condition to hold for all time, where the constant C is allowed to vary in time. We consider separately the calculation of the initial coordinate system and the time evolution of the coordinate system.

5.4.1 Initial Equal-Orthogonal System

If we consider the perturbation of a plane, we can derive an iterative method to compute solutions. We parameterize the surface using $\mathbf{x} = (x_1, x_2, \sqrt{\epsilon}\eta(x_1, x_2))$ and let

$$x_1 = \beta_1 + \epsilon S_1(\beta_1, \beta_2) \tag{5.84}$$

$$x_2 = \beta_2 + \epsilon S_2(\beta_1, \beta_2) \tag{5.85}$$

$$C = 1 + \epsilon C_1 \tag{5.86}$$

where S_1 and S_2 are periodic in β.

Substituting this into the system of equations and keeping the first two orders in ϵ gives

$$\frac{\partial S_1}{\partial \beta_2} + \frac{\partial S_2}{\partial \beta_1} = -\frac{\partial \eta}{\partial x_1}\frac{\partial \eta}{\partial x_2} + \epsilon F_1 \tag{5.87}$$

$$\frac{\partial S_1}{\partial \beta_1} - \frac{\partial S_2}{\partial \beta_2} = \frac{1}{2}\left(\left(\frac{\partial \eta}{\partial x_2}\right)^2 - \left(\frac{\partial \eta}{\partial x_1}\right)^2 + \epsilon F_2 + C_1\right) \tag{5.88}$$

where

$$\begin{aligned} F_1 = & -\frac{\partial S_1}{\partial \beta_1}\frac{\partial S_1}{\partial \beta_2} - \frac{\partial S_2}{\partial \beta_1}\frac{\partial S_2}{\partial \beta_2} \\ & -\frac{\partial S_1}{\partial \beta_2}\left(\frac{\partial \eta}{\partial x_1}\right)^2 - \frac{\partial S_2}{\partial \beta_1}\left(\frac{\partial \eta}{\partial x_2}\right)^2 - \frac{\partial \eta}{\partial x_2}\frac{\partial \eta}{\partial x_1}\left(\frac{\partial S_1}{\partial \beta_1} + \frac{\partial S_2}{\partial \beta_2}\right) \end{aligned} \tag{5.89}$$

$$\begin{aligned} F_2 = & -\left(\frac{\partial S_1}{\partial \beta_1}\right)^2 + \left(\frac{\partial S_2}{\partial \beta_2}\right)^2 - \left(\frac{\partial S_2}{\partial \beta_1}\right)^2 + \left(\frac{\partial S_1}{\partial \beta_2}\right)^2 - 2\left(\frac{\partial \eta}{\partial x_1}\right)^2 \frac{\partial S_1}{\partial \beta_1} \\ & + \left(\frac{\partial \eta}{\partial x_2}\right)^2 \frac{\partial S_2}{\partial \beta_2} + 2\frac{\partial \eta}{\partial x_1}\frac{\partial \eta}{\partial x_2}\left(\frac{\partial S_1}{\partial \beta_2} - \frac{\partial S_2}{\partial \beta_1}\right) \\ & + C_1\left(2\frac{\partial S_2}{\partial \beta_2} + \left(\frac{\partial \eta}{\partial x_2}\right)^2\right). \end{aligned} \tag{5.90}$$

We can express the system given by Equations 5.87 and 5.88 as a pair of coupled Poisson equations defined by

$$\triangle S_1 = \epsilon\left(\frac{\partial F_1}{\partial \beta_2} + \frac{\partial F_2}{\partial \beta_1}\right) - \frac{\partial}{\partial \beta_2}\left(\frac{\partial \eta}{\partial x_1}\frac{\partial \eta}{\partial x_2}\right) \tag{5.91}$$

$$
\begin{aligned}
& + \frac{\partial}{\partial \beta_1}\left(\left(\frac{\partial \eta}{\partial x_2}\right)^2 - \left(\frac{\partial \eta}{\partial x_1}\right)^2\right) \\
\triangle S_2 \;=\; & \epsilon\left(\frac{\partial F_1}{\partial \beta_1} - \frac{\partial F_2}{\partial \beta_2}\right) - \frac{\partial}{\partial \beta_1}\left(\frac{\partial \eta}{\partial x_1}\frac{\partial \eta}{\partial x_2}\right) \\
& - \frac{\partial}{\partial \beta_2}\left(\left(\frac{\partial \eta}{\partial x_2}\right)^2 - \left(\frac{\partial \eta}{\partial x_1}\right)^2\right).
\end{aligned} \tag{5.92}
$$

We have implemented an iterative method to solve this system. We begin with a fast Poisson solver and iterate on S_1 and S_2. The value of C_1 is determined by substitution of S_1 and S_2 into Equation 5.88 and integrating over β. The value of C_1 is thus chosen to satisfy a solvability condition between the right-hand side and the left-hand side of Equation 5.88. Since the left-hand side is periodic and a derivative, the right-hand side must have mean value 0. C_1 is chosen to satisfy this condition. We iterate on C_1 to determine the final value.

5.4.2 Tangential Velocities

In order to satisfy Equations 5.82 and 5.83 for all time, we make use of the freedom in choosing tangential velocities. That is, we want to define the particle velocities by

$$\mathbf{x}_t = \tilde{\mathbf{u}} + T^1\hat{\mathbf{t}}_1 + T^2\hat{\mathbf{t}}_2 \tag{5.93}$$

where T^1 and T^2 are chosen such that

$$(\mathbf{x}_t)_{\beta_1}\cdot\mathbf{x}_{\beta_2} + \mathbf{x}_{\beta_1}\cdot(\mathbf{x}_t)_{\beta_2} = 0 \tag{5.94}$$

$$(\mathbf{x}_t)_{\beta_1}\cdot\mathbf{x}_{\beta_1} = C(\mathbf{x}_t)_{\beta_2}\cdot\mathbf{x}_{\beta_2} + C_t\frac{|\mathbf{x}_{\beta_2}|^2}{2}. \tag{5.95}$$

Recalling that

$$\mathbf{x}_{\beta_1} = |\mathbf{x}_{\beta_1}|\,\hat{\mathbf{t}}_1 \tag{5.96}$$

$$\mathbf{x}_{\beta_2} = |\mathbf{x}_{\beta_2}|\,\hat{\mathbf{t}}_2 \tag{5.97}$$

we substitute Equation 5.93 into Equations 5.94 and 5.95 which gives

$$\begin{aligned} |\mathbf{x}_{\beta_1}| T^1_{\beta_2} + |\mathbf{x}_{\beta_2}| T^2_{\beta_1} &= -|\mathbf{x}_{\beta_2}| (\hat{\mathbf{t}}_{1\beta_1} \cdot \hat{\mathbf{t}}_2) T^1 - |\mathbf{x}_{\beta_1}| (\hat{\mathbf{t}}_{2\beta_2} \cdot \hat{\mathbf{t}}_1) T^2 \\ &\quad - |\mathbf{x}_{\beta_2}| \tilde{\mathbf{u}}_{\beta_1} \cdot \hat{\mathbf{t}}_2 - |\mathbf{x}_{\beta_1}| \tilde{\mathbf{u}}_{\beta_2} \cdot \hat{\mathbf{t}}_1 \qquad (5.98) \\ |\mathbf{x}_{\beta_1}| T^1_{\beta_1} - |\mathbf{x}_{\beta_2}| C T^2_{\beta_2} &= |\mathbf{x}_{\beta_2}| (\hat{\mathbf{t}}_{1\beta_2} \cdot \hat{\mathbf{t}}_2) C T^1 - |\mathbf{x}_{\beta_1}| (\hat{\mathbf{t}}_{2\beta_1} \cdot \hat{\mathbf{t}}_1) T^2 + \\ &\quad C |\mathbf{x}_{\beta_2}| \tilde{\mathbf{u}}_{\beta_2} \cdot \hat{\mathbf{t}}_2 - |\mathbf{x}_{\beta_1}| \tilde{\mathbf{u}}_{\beta_1} \cdot \hat{\mathbf{t}}_1 + C_t \frac{|\mathbf{x}_{\beta_2}|^2}{2}. \qquad (5.99) \end{aligned}$$

If we divide the first equation by $|\mathbf{x}_{\beta_2}|$ and the second equation by $|\mathbf{x}_{\beta_1}|$ we can write the equations as follows:

$$\begin{aligned} \sqrt{C} T^1_{\beta_2} + \quad T^2_{\beta_1} &= G_1 \qquad (5.100) \\ T^1_{\beta_1} - \sqrt{C} T^2_{\beta_2} &= G_2 + C_t \frac{|\mathbf{x}_{\beta_2}|}{2} \qquad (5.101) \end{aligned}$$

where

$$\begin{aligned} G_1 &= -(\hat{\mathbf{t}}_{1\beta_1} \cdot \hat{\mathbf{t}}_2) T^1 - \sqrt{C} (\hat{\mathbf{t}}_{2\beta_2} \cdot \hat{\mathbf{t}}_1) T^2 - \tilde{\mathbf{u}}_{\beta_1} \cdot \hat{\mathbf{t}}_2 - \sqrt{C} \tilde{\mathbf{u}}_{\beta_2} \cdot \hat{\mathbf{t}}_1 \qquad (5.102) \\ G_2 &= \sqrt{C} (\hat{\mathbf{t}}_{1\beta_2} \cdot \hat{\mathbf{t}}_2) T^1 - (\hat{\mathbf{t}}_{2\beta_1} \cdot \hat{\mathbf{t}}_1) T^2 + \sqrt{C} \tilde{\mathbf{u}}_{\beta_2} \cdot \hat{\mathbf{t}}_2 - \tilde{\mathbf{u}}_{\beta_1} \cdot \hat{\mathbf{t}}_1. \qquad (5.103) \end{aligned}$$

As with the system for initially determining the coordinate system, the right-hand sides of these equations must have mean 0 over the surface. For the first equation, it is not certain whether we can always satisfy this condition. For the second equation, we choose C_t to satisfy the mean zero constraint.

In order to solve the system numerically, we rewrite it to resemble Equations 5.91 and 5.93. We make the following substitutions:

$$\begin{aligned} C_1 &= \sqrt{C} - 1 \qquad (5.104) \\ \alpha &= 1 - \frac{|\mathbf{x}_{\beta_2}|}{2}. \qquad (5.105) \end{aligned}$$

We then rewrite the system as

$$T^1_{\beta_2} + T^2_{\beta_1} = R_1 \tag{5.106}$$

$$T^1_{\beta_1} - T^2_{\beta_2} = R_2 + C_t \tag{5.107}$$

where

$$R_1 = G_1 - C_1 T^1_{\beta_2} \tag{5.108}$$

$$R_2 = G_2 + C_1 T^2_{\beta_2} - C_t \alpha. \tag{5.109}$$

By differentiating both equations, we can rewrite the system as a pair of Poisson equations:

$$\triangle T^1 = \frac{\partial R_1}{\partial \beta_2} + \frac{\partial R_2}{\partial \beta_1} \tag{5.110}$$

$$\triangle T^2 = \frac{\partial R_1}{\partial \beta_1} - \frac{\partial R_2}{\partial \beta 2}. \tag{5.111}$$

We now proceed as with the initial coordinate system. We solve the Poisson equations, then integrate Equation 5.107 to update the value of C_t. We repeat this process iteratively to determine the final value of C_t. We can then update $\mathbf{x}$ and C.

We have tested this algorithm for the case of a two-dimensional steady wave. In this case, we anticipate that C_t should be zero for all time. Our experiments show that in this case, the algorithm maintains the orthogonality of the coordinate system and the value of C remains roughly constant.

5.5 Calculation of the Fourier Smoothing Factor

One difficulty in using the Fourier smoothing as defined in Chapter 5 is that the smoothing factors are defined in terms of very slowly converging sums. In order to compute the factor as time evolves, we define ρ^1, ρ^2, and $\bar{\rho}$ to depend on a parameter

α:

$$\rho^1(\mathbf{k}h,\alpha) = \frac{1}{2\pi}\sum_{(j_1,j_2)\neq(0,0)}\frac{j_1\sin(j_1k_1h)\cos(j_2k_2h)}{(j_1^2+\alpha j_2^2)^{3/2}} \tag{5.112}$$

$$\rho^2(\mathbf{k}h,\alpha) = \frac{1}{2\pi}\sum_{(j_1,j_2)\neq(0,0)}\frac{j_2\sin(j_2k_2h)\cos(j_1k_1h)}{(j_1^2+\alpha j_2^2)^{3/2}} \tag{5.113}$$

$$\bar{\rho}(\mathbf{k}h,\alpha) = \frac{1}{2\pi h|\mathbf{k}|}\sum_{(j_1,j_2)\neq(0,0)}\frac{1-\cos(j_1k_1h)\cos(j_2k_2h)}{(j_1^2+\alpha j_2^2)^{3/2}}. \tag{5.114}$$

We can view these sums as the point vortex approximation to the integrals I^1, I^2, and $\bar{I}$, respectively, defined by

$$I^1(k_1,k_2,\alpha) = \frac{1}{2\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\frac{x\sin(k_1x)\cos(k_2y)}{(x^2+\alpha y^2)^{3/2}} \tag{5.115}$$

$$I^2(k_1,k_2,\alpha) = \frac{1}{2\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\frac{y\sin(k_2y)\cos(k_1x)}{(x^2+\alpha y^2)^{3/2}} \tag{5.116}$$

$$\bar{I}(k_1,k_2,\alpha) = \frac{1}{2\pi|\mathbf{k}|}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\frac{1-\cos(k_1x)\cos(k_2y)}{(x^2+\alpha y^2)^{3/2}}. \tag{5.117}$$

The integrals I^1,I^2, and $\bar{I}$ can be computed explicitly (see [13]):

$$\frac{1}{2\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\frac{x\sin(k_1x)\cos(k_2y)}{(x^2+\alpha y^2)^{3/2}} = \frac{k_1}{\sqrt{\alpha k_1^2+k_2^2}} \tag{5.118}$$

$$\frac{1}{2\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\frac{y\sin(k_2y)\cos(k_1x)}{(x^2+\alpha y^2)^{3/2}} = \frac{k_2}{\alpha\sqrt{\alpha k_1^2+k_2^2}} \tag{5.119}$$

$$\frac{1}{2\pi|\mathbf{k}|}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\frac{1-\cos(k_1x)\cos(k_2y)}{(x^2+\alpha y^2)^{3/2}} = \frac{1}{|\mathbf{k}|}\frac{\sqrt{\alpha k_1^2+k_2^2}}{\alpha}. \tag{5.120}$$

We can make use of the analysis of 3 to compute the sums defined by ρ^1, ρ^2, and $\bar{\rho}$. For example, to compute ρ, we have that

$$\rho^1(k_1h,k_2h) = \frac{k_1}{\sqrt{k_1^2+k_2^2}} + \sum_{l=1}^{m} D_{2l-1}h^{2l-1} + O(h^{2m-3}), \tag{5.121}$$

where D_{2l-1} is defined by Equation 3.53. In practice, the formulas describing the coefficients of the error series are unwieldy and inefficient to compute. By using a different approach, we need only compute the coefficient D_1 using the definition

3.53.

The remaining coefficients are computed using a formulation suggested by Lowengrub, Shelley and Merriman [18]. Their work derives the Poisson summation formula for singular functions and shows how to compute the coefficients of the series describing the error of the point vortex approximation. We encountered some difficulty in computing the first term of the error series using their approach. However, we were able to compute all other terms of the series using their approach. Thus, we use the analysis of Chapter 3 to compute the first term in the series, and the analysis of Lowengrub, Shelley and Merriman to compute all other terms in the series. We now give the formulas for computing the coefficients. In the next section, we outline the Poisson summation approach.

Using the notation of Chapter 3, we define $c_{1,0}$ as

$$\begin{aligned}
c_{1,0} = -2B_2 \Bigg[& \int_0^1 D_x^1 C_1(1,y)dy + \int_0^1 D_y^1 C_1(x,1)dx \\
& + \frac{B_2}{2} D_x^1 D_y^1 C_1(1,1) \\
& + \int_0^1 \hat{D}_x^3 D_y^1 C_1(x,1)dx + \int_0^1 \hat{D}_x^1 D_y^3 C_1(1,y)dy \Bigg] \\
& + 4\int_1^\infty \int_0^\infty \hat{D}_x^3 C_1(x,y)dydx + 4\int_0^1 \int_1^\infty \hat{D}_x^3 C_1(x,y)dydx \\
& + 4\int_1^\infty \int_0^\infty \hat{D}_y^3 C_1(x,y)dxdy + 4\int_0^1 \int_1^\infty \hat{D}_y^3 C_1(x,y)dxdy \\
& + 4\int_1^\infty \int_0^\infty \hat{D}_x^3 \hat{D}_y^3 C_1(x,y)dydx \\
& + 4\int_0^1 \int_1^\infty \hat{D}_x^3 \hat{D}_y^3 C_1(x,y)dydx \\
& - 4\int_0^1 \int_0^1 C_1(x,y)dxdy + [C_1(1,0) + C_1(1,1)],
\end{aligned} \tag{5.122}$$

where

$$C_1(x,y) = \frac{x^2}{2\pi(x^2+\alpha y^2)^{3/2}}. \tag{5.123}$$

For $i > 1$ we have

$$c_{i,j}(\alpha) = \frac{1}{(2i-1)!(2\pi)^{2i-1}} \binom{2i-1}{2j} \sum_{(l_1,l_2)\neq(0,0)} \frac{\partial^{2i-1}}{\partial l_1^{2i-1-2j} \partial l_2^{2j}} \left(\frac{l_1}{\sqrt{\alpha l_1^2 + l_2^2}} \right) \quad (5.124)$$

where j ranges from 0 to $i-1$.

The single integrals in $c_{1,0}$ can be computed in closed form. We compute the double integrals numerically. The sums in $c_{i,j}$ for $i > 1$ can be computed using a Maple program. In order to compute the factor in real time, we used Maple to generate the source code needed to compute the sums numerically. For large values of i, only one or two terms of each series is significant. For smaller values of i, the acceleration techniques described by Lowengrub, Shelley, and Merriman can be used so that only a few terms are necessary to compute the series. Similar calculations are used to compute the values of ρ^2 and $\bar{\rho}$.

Having defined $c_{i,j}$, we now give formulas for computing the smoothing factors. The formula for computing ρ^1 is given by

$$\rho^1 = \frac{k_1}{\sqrt{\alpha k_1^2 + k_2^2}} + \sum_{i=1}^{n} h^{2i-1} \sum_{j=0}^{i-1} c_{i,j}(\alpha) k_1^{2i-1-2j} k_2^{2j}. \quad (5.125)$$

We can use the definition of ρ^1 to compute ρ^2:

$$\begin{aligned} \rho^2(k_1 h, k_2 h, \alpha) &= \sum_{(j_1,j_2)\neq(0,0)} \frac{j_2 \sin(j_2 k_2 h)\cos(j_1 k_1 h)}{(j_1^2 + \alpha j_2^2)^{3/2}} && (5.126) \\ &= \frac{1}{\alpha^{3/2}} \sum_{(j_1,j_2)\neq(0,0)} \frac{j_2 \sin(j_2 k_2 h)\cos(j_1 k_1 h)}{\left(\frac{j_1^2}{\alpha} + j_2^2\right)^{3/2}} && (5.127) \\ &= \frac{\rho^1(k_2 h, k_1 h)(1/\alpha)}{\alpha^{3/2}}. && (5.128) \end{aligned}$$

Finally, we consider the calculation of $\bar{\rho}$. We have

$$|\mathbf{k}|\bar{\rho}(k_1, k_2, \alpha) = \frac{1}{\alpha}\sqrt{\alpha k_1^2 + k_2^2} + \sum_{i=1}^{n} h^{2i-1} \sum_{j=0}^{i} c'_{i,j+1}(\alpha) k_1^{2i-2j} k_2^{2j} \quad (5.129)$$

where the coefficients $c'_{i,j}(\alpha)$ are defined by:

$$c'_{i,j}(\alpha) = \frac{1}{(2i)!(2\pi)^{2i-1}} \binom{2i}{2j} \frac{\partial^{2i}}{\partial l_1^{2i-2j} \partial l_2^{2j}} \left(\frac{\sqrt{\alpha l_1^2 + l_2^2}}{\alpha} \right). \tag{5.130}$$

We can relate the coefficients $c'_{i,j}$ to $c_{i,j}$ by

$$c'_{i,j} = \begin{cases} \frac{1}{(2i-2j)} c_{i,j}(\alpha) & j < i \\ \frac{1}{2i\alpha^{3/2}} c_{i,0}(1/\alpha) & i = j \end{cases}. \tag{5.131}$$

For our calculations, we computed the values of $c_{i,j}$ up to $i = 30$. This gives single precision accuracy. The source code for generating $c_{i,j}$ for $i > 1$ takes up more than 2 megabytes of space. However, the code takes less than a second to execute. The reason is that only one or two terms in the series defining $c_{i,j}$ are necessary. The calculation of ρ^1,ρ^2, and $\bar{\rho}$ takes less than 15 seconds on a workstation. While this is not negligible, it is a small percentage of the computational cost at each time step.

5.5.1 The Poisson Summation Method

The derivation of the formula for calculating the coefficients comes from the Poisson summation formula for singular function, defined by Lowengrub *et al.* as follows:

$$\sum_{\mathbf{j}\neq\mathbf{0}} f(h\mathbf{j})h^2 - \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\mathbf{x})d\mathbf{x} = \lim_{\epsilon\to 0}\left[\sum_{\mathbf{l}\neq\mathbf{0}} \hat{f}_\epsilon\left(\frac{2\pi}{h}\mathbf{l}\right) - f_\epsilon(\mathbf{0})h^2\right] \tag{5.132}$$

where f is singular at the origin, δ_ϵ is some smoothing function and $f_\epsilon = f * \delta_\epsilon$. If we define $f = K\omega$ where

$$K(\mathbf{x}) = \frac{x_1}{(x_1^2 + x_2^2)^{3/2}} \tag{5.133}$$

and ω is periodic in $\mathbf{x}$ and odd in x_1. For example, to compute ρ^1, we take $\omega = \sin(k_1 x_1)\cos(k_2 x_2)$. Using the Poisson formula for singular functions, we have

$$\sum_{\mathbf{j}\neq\mathbf{0}} (K\omega)(\mathbf{x_j})h^2 - \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(K\omega)(\mathbf{x})d\mathbf{x} = \lim_{\epsilon\to 0}\left[\sum_{\mathbf{l}\neq\mathbf{0}} \hat{\delta}_\epsilon \widehat{K\omega}\left(\frac{2\pi}{h}\mathbf{l}\right) - (K\omega)_\epsilon(\mathbf{0})h^2\right]. \tag{5.134}$$

We here take $\alpha = 1$, and note that the more general case is similar. We are interested in expanding the series on the right-hand side in powers of h. We first define $F(\mathbf{l}, \mathbf{m}, \tau)$ to be

$$F(\mathbf{l}, \mathbf{m}, \tau) = \frac{l_1 - \tau m_1}{\sqrt{(l_1 - \tau m_1)^2 + (l_2 - \tau m_2)^2}}, \tag{5.135}$$

where we $\tau = 2\pi/h$. We compute the Fourier transform of K to be $\hat{K}(\mathbf{l}) = -iF(\mathbf{l})$. From the convolution theorem, we have that

$$\widehat{K\omega}(\mathbf{l}) = \frac{1}{(2\pi)^2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\hat{K}(\mathbf{l}-\mathbf{m})\hat{\omega}(\mathbf{m})d\mathbf{m}. \tag{5.136}$$

Using this result, we compute the following:

$$\widehat{K\omega}\left(\frac{1}{\tau}\mathbf{l}\right) = \frac{-i}{(2\pi)^2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}F(\mathbf{l}, \mathbf{m}, \tau)\hat{\omega}(\mathbf{m})d\mathbf{m}. \tag{5.137}$$

Using this result, we have

$$\sum_{\mathbf{l}\neq\mathbf{0}} \widehat{(K\omega)}_\epsilon\left(\frac{2\pi}{h}\mathbf{l}\right) = -\sum_{\mathbf{l}\neq\mathbf{0}}\hat{\delta}_\epsilon\left(\frac{1}{\tau}\right)\frac{i}{(2\pi)^2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}F(\mathbf{l}, \mathbf{m}, \tau)\hat{\omega}(\mathbf{m})d\mathbf{m}. \tag{5.138}$$

Since τ is small, we can Taylor expand F about $\tau = 0$ away from the singular point.

$$F(\mathbf{l}, \tau\mathbf{m}, \tau) = \sum_{n=0}^{\infty}\partial_\tau^n F(\mathbf{l}, \mathbf{m}, 0)\frac{\tau^n}{n!}. \tag{5.139}$$

We notice that the terms in this series can be separated as follows:

$$\partial_\tau^n F(\mathbf{l}, \mathbf{m}, 0) = \sum_{j=0}^{n} c^{n-j,j} m_1^{n-j} m_2^j \tag{5.140}$$

with

$$c^{n-j,j} = (-1)^n \frac{\partial^n}{\partial l_1^{n-j} \partial l_2^j} \frac{l_1}{\sqrt{l_1^2 + l_2^2}}. \tag{5.141}$$

We further relate the moments of $\hat{\omega}$ with derivatives of ω:

$$\frac{1}{(2\pi)^2} \int m_1^r m_2^q \hat{\omega}(\mathbf{m}) d\mathbf{m} = i^{-r-q} \left[\partial_{x_1}^r \partial_{x_2}^q \omega\right](\mathbf{0}). \tag{5.142}$$

This result together with the Taylor expansion of F suggest the following form for the series:

$$\sum_{\mathbf{l} \neq \mathbf{0}} \widehat{(K\omega)}_\epsilon \left(\frac{2\pi}{h}\mathbf{l}\right) = -\sum_{n=0}^{N} \frac{\tau^n}{n!} i^{1-n} \sum_{\mathbf{l} \neq \mathbf{0}} \hat{\delta}_\epsilon \sum_{j=0}^{n} c^{n-j,j} \left[\partial_{x_1}^{n-j} \partial_{x_2}^j \omega\right](\mathbf{0}) + O(\tau^N) \tag{5.143}$$

for N sufficiently large. The even terms in n vanish because the summand is odd, leaving a series in odd powers of h, since $\tau = h/2\pi$. Also for j odd, $c^{i,j}$ is odd, so these terms also vanish. Using this and the relation

$$c^{i,j} = (-1)^{i+j} \binom{i}{j} \frac{\partial^i}{\partial l_1^{i-j} \partial l_2^j} \left(\frac{l_1}{\sqrt{l_1^2 + l_2^2}}\right) \tag{5.144}$$

suggests the following series:

$$\sum_{\mathbf{l} \neq \mathbf{0}} \widehat{(K\omega)}_\epsilon \left(\frac{2\pi}{h}\mathbf{l}\right) = \tag{5.145}$$

$$-\sum_{n=0}^{N} h^{2n-1} (-1)^{1-n} \sum_{\mathbf{l} \neq \mathbf{0}} \hat{\delta}_\epsilon \sum_{j=0}^{n} c^{2n-1-2j,2j} \left[\partial_{x_1}^{2n-1-2j} \partial^2 j_{x_2} \omega\right](\mathbf{0}) \tag{5.146}$$

$$+O(\tau^N) \tag{5.147}$$

$$= \sum_{i=0}^{N} h^{2i-1} (-1)^{i-1} \sum_{j=0}^{i-1} c_{i,j} \partial_{x_1}^{2i-1-2j} \left[\partial_{x_2}^{2j}\right] \omega(\mathbf{0}) + O(\tau^N) \tag{5.148}$$

for any N sufficiently large and where $c_{i,j}$ are defined by

$$c_{i,j} = \frac{1}{(2i-1)!(2\pi)^{2i-1}} \binom{2i-1}{2j} \sum_{(l_1,l_2) \neq (0,0)} \frac{\partial^{2i-1}}{\partial l_1^{2i-1-2j} \partial l_2^{2j}} \left(\frac{l_1}{\sqrt{l_1^2 + l_2^2}}\right). \tag{5.149}$$

From this we obtain the formula for computing the smoothing factors, by noting that the sums $c_{i,j}$ are absolutely convergent for $n > 2$. Hence, the choice of smoothing function does not matter for these sums, and we can compute these sums independent of the limit in ϵ.

We remark here that we have not rigorously shown that this approach is valid, but have (somewhat naively) applied the approach to our particular problem. In particular, Lowengrub assumed that ω was a decaying function, whereas in our case, it is periodic. Hence the rigor of all of the steps which lead to the final formula is not clear. Surprisingly, we have compared these results with our approach using the Euler-Maclaurin formula and find that for $n > 1$, the coefficients agree. The derivation of the series for general α is similar.

5.5.2 The Calculation of $c_{1,0}$

The most computationally intensive step in the calculation of the smoothing factors is the computation of $c_{1,0}$ using Equation 5.123. We present an alternative approach to computing this coefficient which is less rigorous, but gives the correct answer and is more efficient computationally.

For $n = 1$, the Poisson summation approach suggests that for general α,

$$c_{1,0}(\alpha) = \frac{1}{(2\pi)} \sum_{(l_1,l_2)\neq(0,0)} \hat{\delta}_\epsilon \frac{\partial}{\partial l_1} \left(\frac{l_1}{\sqrt{\alpha l_1^2 + l_2^2}} \right) \tag{5.150}$$

$$= \frac{1}{2\pi} \sum_{(l_1,l_2)\neq(0,0)} \hat{\delta}_\epsilon \frac{l_2^2}{(\alpha l_1^2 + l_2^2)^{3/2}}. \tag{5.151}$$

Because the underlying sum is divergent, no choice of smoothing factor can make this sum converge in the limit as ϵ approaches zero. This suggests that this term cannot be evaluated separately from the self induction term in the limit as ϵ approaches zero. Because the coefficient should be finite, we expect that the singularity in the sum should be balanced by a singularity in the self induction term. We must

evaluate $c_{1,0}$ using

$$c_{1,0} = \lim_{\epsilon \to 0} \left(\sum_{\mathbf{j} \neq \mathbf{0}} \hat{\delta}_\epsilon \frac{l_2^2}{(\alpha l_1^2 + l_2^2)^{3/2}} - (K\omega)_\epsilon (\mathbf{0})\, h \right) \tag{5.152}$$

where the limit cannot be applied to each term separately. This differs from the case which Lowengrub studied, in that the sums are conditionally convergent and a careful choice of smoothing function allows this limit to be separated.

However, since we know that the term is finite, we can sum the series minus the singularity. We then must determine the nonsingular contribution of the self induction term. To sum the series, we choose the smoothing function which gives a sharp square cut-off. We let $\epsilon = \frac{1}{M}$ and

$$\hat{\delta}_{1/M} = \begin{cases} 1 & \text{if } |l_1| \leq M \text{ and } |l_2| \leq M \\ 0 & \text{otherwise} \end{cases} . \tag{5.153}$$

Using this definition of $\hat{\delta}_\epsilon$ gives the following lattice sum (which we call $S(\alpha)$):

$$S(\alpha) = \lim_{M \to \infty} \sum_{(l_1,l_2) \neq (0,0)} \frac{l_2^2}{(\alpha l_1^2 + l_2^2)^{3/2}}. \tag{5.154}$$

We can rewrite this sum using the symmetry of the summand and by reordering the terms in the sum:

$$\begin{aligned} S(\alpha) &= \lim_{M \to \infty} 4 \sum_{m=1}^{M} {\sum_{k=0}^{m}}' \left[\frac{m^2}{(\alpha k^2 + m^2)^{3/2}} + \frac{k^2}{(\alpha m^2 + k^2)^{3/2}} \right] && (5.155) \\ &= \lim_{M \to \infty} 4 \sum_{m=1}^{M} \frac{1}{m} {\sum_{k=0}^{m}}' \left[\frac{1}{(\alpha (k/m)^2 + 1)^{3/2}} + \frac{(k/m)^2}{(\alpha + (k/m)^2)^{3/2}} \right] && (5.156) \\ &= \lim_{M \to \infty} 4 \sum_{m=1}^{M} g(m;\alpha), && (5.157) \end{aligned}$$

where

$$g(m;\alpha) = \frac{1}{m} {\sum_{k=0}^{m}}' \left[\frac{1}{(\alpha (k/m)^2 + 1)^{3/2}} + \frac{(k/m)^2}{(\alpha + (k/m)^2)^{3/2}} \right] \tag{5.158}$$

and $\sum'$ denotes that terms with $k = 0$ or $k = m$ are weighted by $1/2$. The function $g(m;\alpha)$ is the trapezoidal rule approximation of the integral

$$\int_0^1 \left[\frac{1}{(\alpha r^2 + 1)^{3/2}} + \frac{r^2}{(\alpha + r^2)^{3/2}} \right] dr = \operatorname{arcsinh}(1/\sqrt{\alpha}). \tag{5.159}$$

Using the Euler-Maclaurin summation formula, we know that

$$g(m;\alpha) - \operatorname{arcsinh}(1/\sqrt{\alpha}) = O(1/m^2) \tag{5.160}$$

for large m. Hence, we can define a modified sum S_1 which converges:

$$S_1(\alpha) = \sum_{m=1}^{\infty} \left[g(m;\alpha) - \operatorname{arcsinh}(1/\sqrt{\alpha}) \right]. \tag{5.161}$$

Since the singularity in S must balance with the singularity in $f_\epsilon(\mathbf{0})h$, we need only determine the nonsingular contribution of the latter term. The correct term was determined by an educated guess. The formula for computing the first-order coefficient is

$$c_{1,0}(\alpha) = S_1(\alpha) - 2\frac{\operatorname{arcsinh}(1/\sqrt{\alpha})}{\pi}. \tag{5.162}$$

We have compared values of $c_{1,0}(\alpha)$ computed using Equation 5.162 with values computed using Equation 5.123 from the Euler-Maclaurin approach and find that the results give good agreement. Using this alternative definition of $c_{1,0}$, the calculation of the smoothing factors requires only 2 or 3 seconds on a workstation.

Chapter 6

Numerical Experiments

In this chapter we report the results of several numerical experiments performed using the methods developed in the previous chapters.

6.0.3 Gaussian Initial Condition

We consider the time evolution of a wave with initial condition given by

$$\eta(x_1, x_2) = \frac{\exp(-10(x_1^2 + x_2^2))}{20} \tag{6.1}$$

$$\phi(x_1, x_2) = 0. \tag{6.2}$$

We encountered some difficulty in setting up the equal-orthogonal coordinate system for this initial condition. For this initial condition, we used the Fourier smoothing with the equilibrium smoothing factor in the fixed grid formulation. This proved to be sufficient to suppress instabilities for the length of time which we considered. We propagated the wave until approximately $t = 10$.

In Figure 6.1 we show the initial wave profile. In Figures 6.2, 6.3, and 6.4 we show the wave profile at times $t = .95$, 5.23, and 9.5 respectively.

For our next experiment, we double the amplitude of the initial condition and propagate the wave using the Lagrangian formulation with the desingularization method for stability. The initial condition is

$$\eta(x_1, x_2) = \frac{\exp(-10(x_1^2 + x_2^2))}{10} \tag{6.3}$$

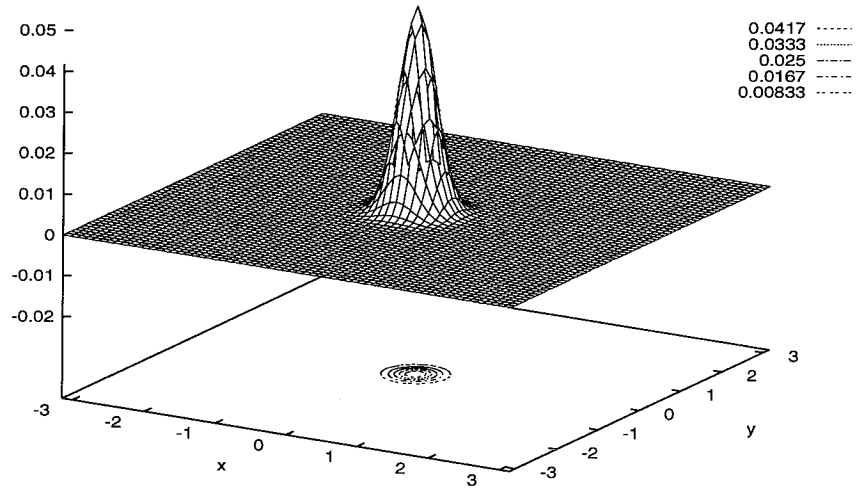

Figure 6.1: Gaussian wave initial condition.

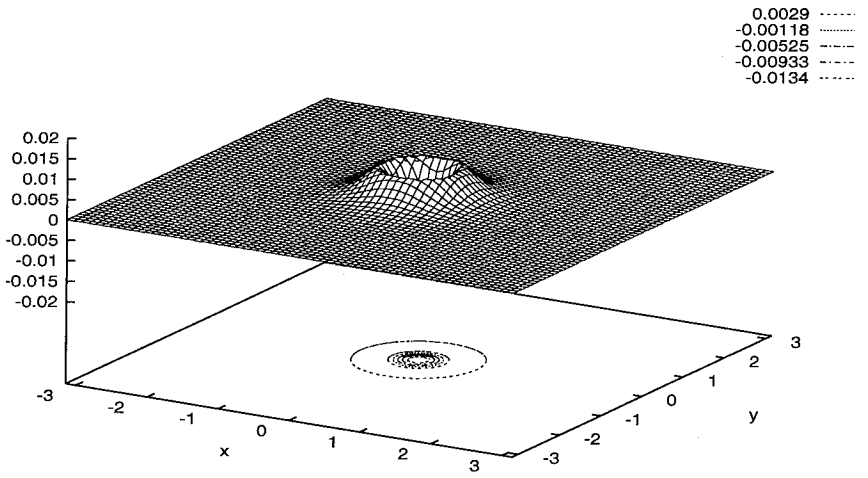

Figure 6.2: Gaussian wave at $t = .95$.

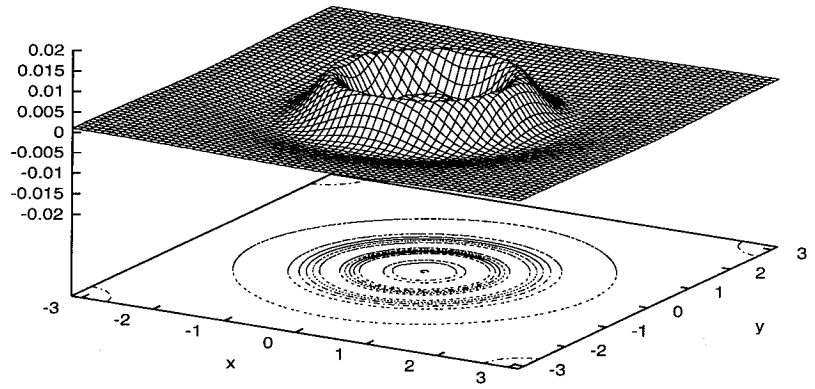

Figure 6.3: Gaussian wave at $t = 5.23$.

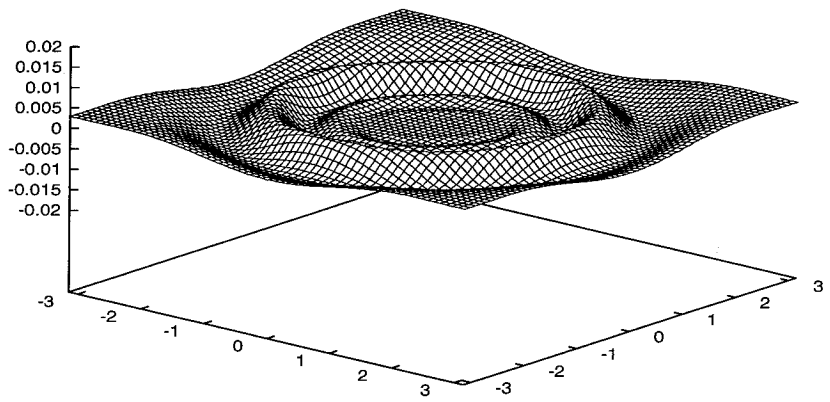

Figure 6.4: Gaussian wave at $t = 9.5$.

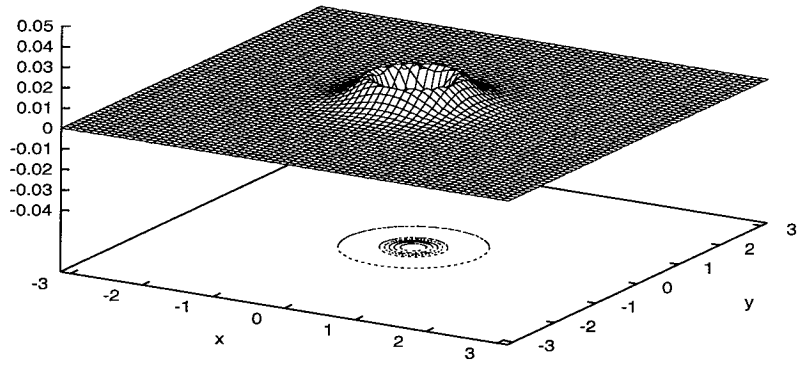

Figure 6.5: Gaussian wave profile at $t = .105$.

$$\phi(x_1, x_2) = 0. \tag{6.4}$$

In Figures 6.5-6.8 we show the wave profile at times $t = .105$, 4.71, 6.81, and 9.42.

6.0.4 Three-Dimensional Steady Wave

Meiron, Saffman, and Yuen [19] showed that three-dimensional waves of permanent form can be obtained as bifurcations of two-dimensional Stokes waves. In this section we present results of the time evolution of a three-dimensional wave of permanent form.

The initial condition is given by

$$z(x,y) = \sum_{m=0}^{M}\sum_{n=0}^{N} a_{m,n}\cos(\frac{1}{2}mx)\cos(\frac{3}{2}ny) \tag{6.5}$$

$$\phi(x,y) = \sum_{m=1}^{M-1}\sum_{n=0}^{N} b_{m,n}\sin(\frac{1}{2}mx)\cos(\frac{3}{2}ny)\exp(\sqrt{\frac{1}{4}m^2+\frac{9}{4}n^2}z(x,y)). \tag{6.6}$$

This wave is 4π periodic in x and y. The wave we computed propagates in the positive x direction with wave speed 1.0222. In Table 6.1 we give the first few

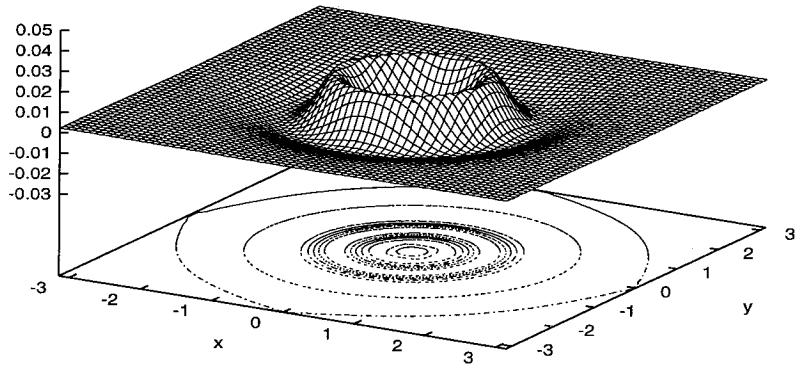

Figure 6.6: Gaussian wave at $t = 4.71$.

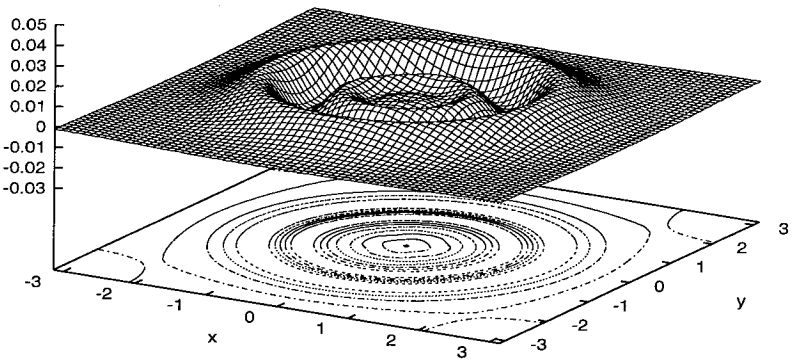

Figure 6.7: Gaussian wave at $t = 6.81$.

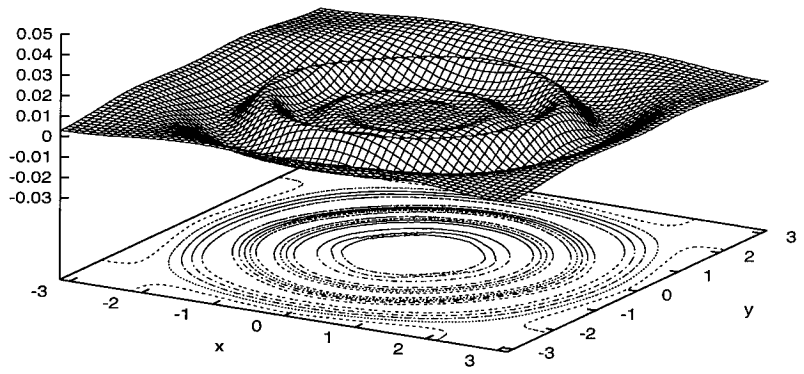

Figure 6.8: Gaussian wave profile at $t = 9.42$.

nonzero coefficients. The initial wave profile is shown in Figure 6.9. The coordinate system is given by

$$x_1 = \beta_1 + S_1(\beta_1, \beta_2) \tag{6.7}$$

$$x_2 = \beta_2 + S_2(\beta_1, \beta_2) \tag{6.8}$$

where S_1 and S_2 are shown in Figures 6.10 and 6.11. The value of the constant C is approximately 1.02173.

We have propagated the wave for approximately one half of a period. The wave profile at time $t = 7.5$ is shown in Figure 6.12. The difference between the computed value and the theoretical value at $t = 7.5$ is shown in Figure 6.13. The error appears to grow linearly as a function of time. The reason for this is currently under investigation.

n	m	$a_{m,n}$	$b_{m,n}$
0	2	0.157732286726428E-03	0.000000000000E+00
0	4	0.479964514039474E-05	0.000000000000E+00
0	6	0.550038364206870E-07	0.000000000000E+00
0	8	0.140409205611713E-09	0.000000000000E+00
1	1	-0.898422335587618E-03	-0.503284290781E-02
1	3	0.597223725336634E-04	0.113022690894E-04
1	5	0.105367887717518E-05	-0.177810574629E-06
1	7	0.108145424302587E-07	-0.539736250622E-08
2	0	0.203623047421752E+00	0.202290992971E+00
2	2	-0.262869354349005E-03	-0.807610958614E-03
2	4	0.820223586888213E-05	0.309415997044E-05
2	6	0.118348697558678E-06	-0.694402295192E-07
2	8	-0.468303283517227E-09	0.125805700996E-10
3	1	0.259423437502074E-01	0.181733360205E-01
3	3	-0.461481062245146E-04	-0.968188684960E-04
3	5	0.133061116113115E-05	0.793014193891E-06
3	7	0.283833638881459E-08	0.123460513610E-07
3	9	0.539817727098814E-09	-0.249037209185E-09
4	0	0.222972428278761E-01	0.112828359184E-02
4	2	0.296740463245126E-02	0.173857579653E-02
4	4	-0.702897007969855E-05	-0.127935406013E-04
4	6	0.223848033943067E-06	0.245483233623E-06
4	8	0.469881950477252E-08	0.973029021872E-09

Table 6.1: A few of the nonzero coefficients for a three-dimensional wave of permanent form.

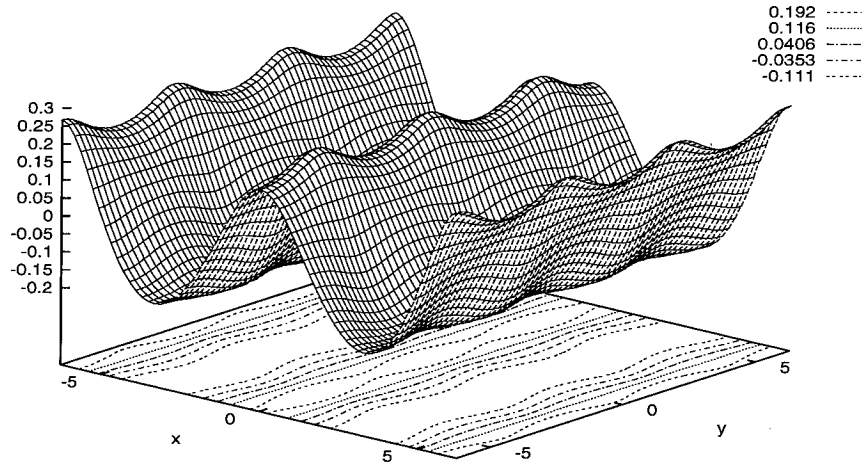

Figure 6.9: Three-dimensional steady wave initial profile.

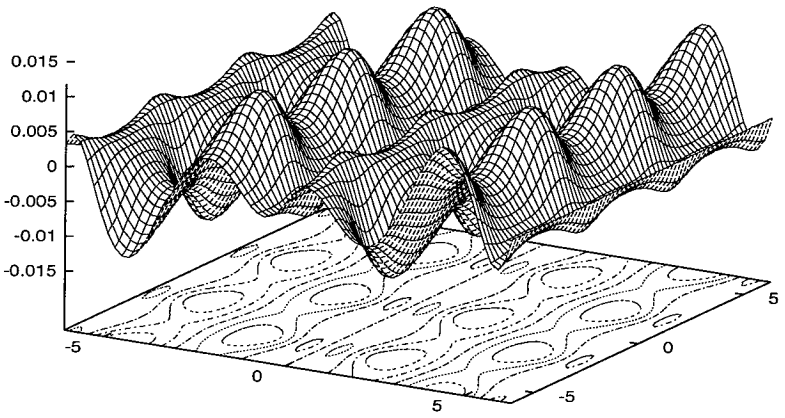

Figure 6.10: The function $S_1(\beta_1, \beta_2)$.

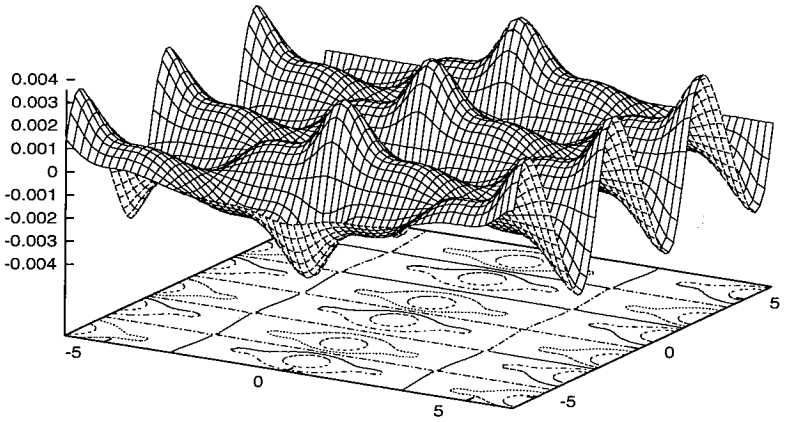

Figure 6.11: The function $S_2(\beta_1, \beta_2)$.

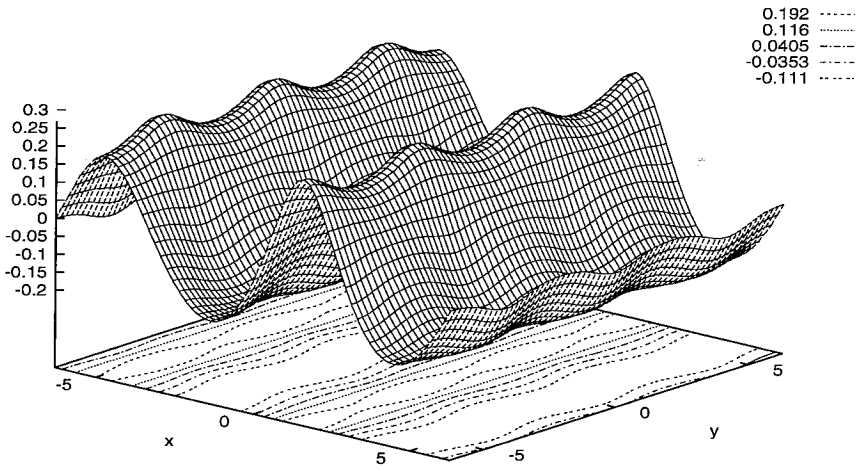

Figure 6.12: The three-dimensional steady wave at $t = 7.5$.

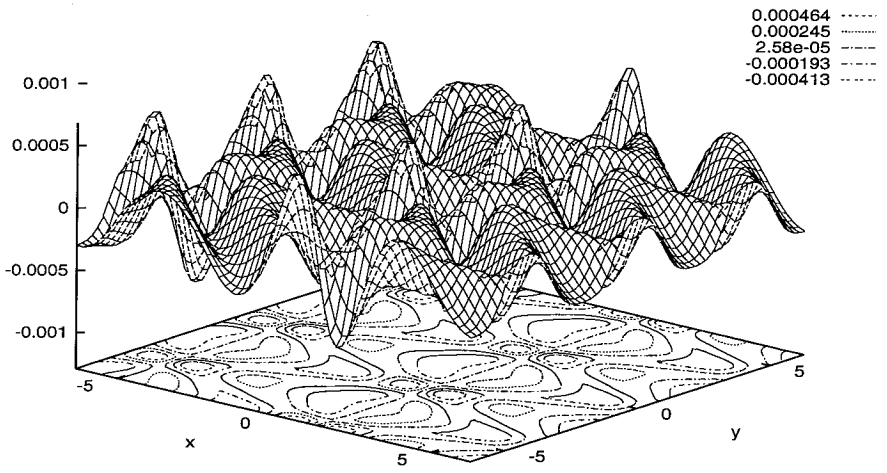

Figure 6.13: The error in the three-dimensional steady wave at $t = 7.5$.

Chapter 7

Summary and Conclusions

In this work, we have presented some investigations of the numerical calculation of water waves in three dimensions. We have developed efficient, accurate and stable schemes for describing the evolution of surface waves. While the original formulation of the boundary integral is unstable, we find that both the Fourier smoothing and the desingularization methods seem to remove the most severe numerical instabilities. Although the Fast Multipole algorithm is not as fast as we had hoped, the method still has promise because the asymptotic operation count is $O(N^2)$ and because the algorithm is readily adapted to parallel architectures. As with any scientific work, however, there remain open questions.

From a practical standpoint, the computational time required to calculate waves using our methods is daunting, even for problems with relatively coarse spatial discretizations. We have not considered how readily these schemes can be adapted to distributed parallel architectures. Because of the high computational cost, we have not tested the stability of the schemes at high resolution. From a theoretical standpoint, we do not know how well the stabilization methods work when the surface becomes highly distorted, as with breaking and plunging waves. The theoretical work for the Fourier smoothing method suggests that it should yield robust results for large N and for distorted interfaces. It is not certain how well the desingularization method will work for large N or for distorted interfaces. Moreover, it is not certain whether an equal-orthogonal coordinate system can be efficiently computed

for a wide range of problems.

We have shown that the boundary integral approach, when properly stabilized, works well for the problems which we have investigated. While no single method is well-suited to all problems, the boundary integral approach is a promising method for numerically studying water waves and general interfacial fluid flow in three dimensions.

Bibliography

[1] G. R. Baker, D. I. Meiron, and S. A. Orszag. Generalized vortex methods for free-surface flow problems. *J. Fluid Mech.*, 123:477–501, 1982.

[2] G. R. Baker, D. I. Meiron, and S. A. Orszag. Boundary integral methods for axisymmetric and three-dimensional Rayleigh-Taylor instability problems. *Physica 12D*, pages 19–31, 1984.

[3] J. T. Beale, T. Y. Hou, and J. S. Lowengrub. Convergence of a boundary integral method for water waves. *SIAM J. Num. Anal.*, 33:1797–1843, 1996.

[4] C. L. Berman and L. Greengard. A renormalization method for the evaluation of lattice sums. *Journal of Mathematical Physics*, 35:6036–6048, November 1994.

[5] E. O. Brigham. *The Fast Fourier Transform and its Applications.* Prentice-Hall,Inc, Englewood Cliffs, New Jersey, 1988.

[6] T. F. Buttke. Fast vortex methods in three dimensions. *Lectures in Applied Mathematics: Vortex Dynamics and Vortex Methods*, 28:51–66, 1991.

[7] J. Goodman, T. Y. Hou, and J. Lowengrub. Convergence of the point vortex method for the 2-d Euler equations. *Comm. on Pure and Applied Mathematics*, 43:415–430, 1990.

[8] L. F. Greengard. *The Rapid Evaluation of Potential Fields in Particle Systems.* The MIT Press, Cambridge, Mass., 1988.

[9] Z.S. Hakura et al. *Parallel Multipole Treecode Algorithm PMTA 4.1 Manual.* Dept. of Electrical Engineering, Duke University, 1995. version 4.1.

[10] T. Y. Hou. Numerical solutions to free boundary problems. *Acta Numerica*, pages 335–415, 1995.

[11] T. Y. Hou and J. Lowengrub. Convergence of the point vortex method for the 3-d Euler equations. *Comm. on Pure and Applied Mathematics*, 43:965–981, 1990.

[12] T. Y. Hou, J. Lowengrub, and R. Krasny. Convergence of a point vortex method for vortex sheets. *SIAM J. Num. Analysis*, 28:308–320, 1991.

[13] T.Y. Hou, Z.H. Teng, and P. W. Zhang. Well-posedness of linearized motion for 3-d water-waves far from equilibrium. *Communications in Partial Differential Equations*, 21:1551–1585, 1996.

[14] T.Y. Hou and P. W. Zhang. Stability of boundary integral method for 3-d water waves. unpublished manuscript and private communications, 1996.

[15] C.G. Lambert. Multipole-based algorithms for efficient calculation of forces and potentials in macroscopic periodic assemblies of particles. Technical Report 94-004, Dept. of Electrical Engineering, Duke University, 1994.

[16] J.F. Leathrum, Jr and J.A. Board, Jr. The parallel fast multipole algorithm in three dimensions. Technical report, Dept. of Electrical Engineering, Duke University, 1992.

[17] M.S. Longuet-Higgins and E. D. Cokelet. The deformation of steep surface waves on water, I. a numerical method of computation. *Proc. Roy. Soc. London A*, 350:1–26, 1976.

[18] J. S. Lowengrub, M. J. Shelley, and B. Merriman. High order and efficient methods for the vorticity formulation of the Euler equations. *SIAM Journal on Scientific Computing*, 14:1107–1142, 1993.

[19] D. I Meiron, P. G. Saffman, and H. C. Yuen. Calculation of steady three-dimensional deep-water waves. *J. Fluid Mech.*, 124:109–121, 1982.

[20] I. Navot. An extension of the Euler-Maclaurin summation formula to functions with a branch singularity. *J. Math. and Phys.*, 40:271–276, 1961.

[21] A. L. New, P. McIver, and D. H. Peregrine. Computations of overturning waves. *J. Fluid Mech.*, 150:233–251, 1985.

[22] W.T. Rankin and J. A. Board, Jr. A portable distributed implementation of the parallel multipole tree algorithm. Technical Report 95-002, Dept. of Electrical Engineering, Duke University, 1995.

[23] A. J. Roberts. A stable and accurate numerical method to calculate the motion of a sharp interface between fluids. *IMA J. Appl. Math.*, 31:13–35, 1983.

[24] P. G. Saffman. *Vortex Dynamics.* Cambridge University Press, New York, NY, 1992.

[25] M. J. Shelley. A study of singularity formation in vortex-sheet motion by a spectrally accurate vortex method. *J. Fluid Mech.*, 244:493–526, 1992.

[26] A. Sidi and M. Israeli. Quadrature methods for periodic singular and weakly singular fredholm integral equations. *Journal of Scientific Computing*, 3:201–231, 1988.

[27] G. Tryggvason. Simulations of vortex sheet roll-up by vortex methods. *J. Comp. Phys.*, 80:1–16, 1989.

[28] A. van de Vooren. A numerical investigation of the rolling-up of vortex sheets. *Proc. Roy. Soc. London Ser. A*, 373:67–91, 1980.

[29] T. Vinje and P. Brevig. Numerical simulation of breaking waves. *Adv. Water Resources*, 4:77–82, 1981.

[30] G. B. Whitham. *Linear and Nonlinear Waves.* Wiley and Sons, New York, 1973.