Internal Tides Near Steep Topographies

by

Sydney Glass Sroka

Submitted to the Department of Mechanical Engineering
in partial fulfillment of the requirements for the degree of

Master of Science in Mechanical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2016

Signature redacted

Author .....
Department of Mechanical Engineering
August 12, 2016

Signature redacted

Certified by .....
Pierre Lermusiaux
Associate Professor of Mechanical Engineering
Massachusetts Institute of Technology
Thesis Supervisor

Signature redacted

Accepted by .....
Rohan Abeyaratne
Chairman, Department Committee on Graduate Theses
Professor of Mechanical Engineering
Massachusetts Institute of Technology

Internal Tides Near Steep Topographies

by

Sydney Glass Sroka

Submitted to the Department of Mechanical Engineering
on August 12, 2016, in partial fulfillment of the
requirements for the degree of
Master of Science in Mechanical Engineering

Abstract

The primary contributions of this thesis include the first stages of development of a 2D, finite-volume, non-hydrostatic, σ-coordinate code and beginning to apply the Dynamically Orthogonal field equations to study the sensitivity of internal tides to perturbations in the density field. First, we ensure that the 2D Finite Volume (2DFV) code that we use can accurately capture non-hydrostatic internal tides since these dynamics have not yet been carefully evaluated for accuracy in this framework. We find that, for low-aspect ratio topographies, the z-coordinate mesh in the 2DFV code produces numerical artifacts near the bathymetry. To ameliorate these staircasing effects, and to develop the framework towards a moving mesh with free-surface dynamics, we have begun to implement a non-hydrostatic σ-coordinate framework which significantly improves the representation of the internal tides for low-aspect ratio topographies. Finally we investigate the applicability of stochastic density perturbations in an internal tide field. We utilize the Dynamically Orthogonal field equations for this investigation because they achieve substantial model order reduction over ensemble Monte-Carlo methods.

Thesis Supervisor: Pierre Lermusiaux
Title: Associate Professor of Mechanical Engineering
Massachusetts Institute of Technology

Acknowledgments

I would like to thank my adviser, Pierre Lermusiaux, for the helpful guidance, incredible creativity, and generous advice from many, many meetings- and even more emails -over the course of this work. I would like to thank Pat Haley for help with MATLAB, LaTeX, Linux, and general numerical queries - you are truly amazing. I would like to thank the entire MSEAS crew - especially Deepak for late-night advice and conversations, Jing for inspired answers to numerics questions and thoughtful input on too many items to name, Tapovan for being so generous with his time, incredibly helpful with my research questions, and wonderfully welcoming when I first joined the group, John for brilliant conversations and many entertaining moments, and Jen for all the lunchtime conversations and advice for the future. I would also like to thank Sonya Legg for generously providing simulation results that were invaluable for accomplishing this work. I would like to thank the National Science Foundation for supporting me with through the GRFP, and also I would also like to thank the National Science Foundation for partial support under the grant OCE-1061160 (ShelfIT) and the Office of Naval Research for partial support under grant N00014-11-1-0701 (MURI-IODA), each to MIT and WHOI. I also thank the ONR for partial support under grant N00014-15-1-2626 (DRI-FLEAT). Finally, to all of my friends and family who have provided unconditional love and support, I am deeply grateful.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The profound influence of internal waves on biological systems and human activities has prompted the study of internal wave dynamics with observations and empirical models. Two functions in particular motivate interest and enthusiasm for a thorough dynamical characterization; first, internal waves are primary drivers of diapycnal mixing (Munk and Wunsch, 1998) and, second, internal waves are responsible for nutrient transport in coastal regions (Shea and Broenkow, 1982). While any internal wave activity can encourage diapycnal mixing, since the osciallations are perpendicular to resting isopycnals, the extremely energetic mixing events caused by breaking internal waves (which we now know can exceed 200m in height (Alford et al., 2015)) may prove to be a critical componant of global ocean circulation (Nikurashin and Ferrari, 2013). Internal waves are also important mechanisms for nutrient transport. Some coastal systems depend on the regular cycling of water between deep, high-nutrient regions and shallow, euphotic regions brought about by internal wave activity (Wang et al., 2007). Many studies show that internal waves are influenced by the strength of the barotropic tide, the steepness of bathymetry, and the stratification profile. Here we begin to develop a numerical model that will characterize the sensitivity of internal waves to these and other governing parameters.

Previous work has demonstrated a relationship between the strength of the barotropic tide ans the mixing potential of internal tides (internal waves at tidal frequency) (Legg and Huijts, 2006). This thesis describes simulations that aim to understand the influence of density perturbations on the mixing potential internal tides. We conduct numerical simulations with a 2D Finite Volume Framework (2DFV), that has been developed by the MSEAS group over the last four years (Ueck-

ermann et al., 2013; Ueckermann and Lermusiaux, 2012). While many other numerical models are available, we elect to conduct all of our simulations with the 2DFV code to take advantage of the existing architecture, specifically the novel Dynamically Orthogonal (DO) field equations. The novel DO field equations (Sapsis and Lermusiaux, 2009, 2012; Ueckermann et al., 2013) have been used to efficiently solve the stochastic incompressible, Boussinesq, Navier-Stokes equations (Ueckermann et al., 2013).

1.0.1 Verification of Internal Tide Dynamics

In chapter 2 we conduct a verification study to demonstrate the 2DFV code accurately captures internal wave dynamics. This verification study uses the internal wave dispersion relation, linear theory from Khatiwala (2003), and simulation results generously provided by Sonya Legg for comparison (Legg and Huijts, 2006). We find the 2DFV code can accurately capture the dynamics of internal tides generated at steep topographies.

1.0.2 A σ-coordinate Mesh

In chapter 3 we describe a coordinate transformation which ameliorated numerical artifacts due to a staircased representation of low-aspect ratio topographies. Refining the z-coordinate grid, instead of changing the coordinate system, to smooth the representation of bathymetry may be a sufficient solution for a deterministic problem, because the computational cost will only scale with the grid resolution. However, when a more expensive algorithm is required, like the DO field equations where the cost scales as the square of the number of modes, an efficient mesh can be advantageous, if not necessary.

There are two common ways to modify a z-coordinate representation of bathymetry: re-define the vertical coordinate, as in a σ-coordinate mesh, or modify the geometry of the boundary cells, as in the cut-cell method. While a σ-coordinate mesh has been shown to be an effective solution for stair-casing effects (Gallus Jr and Klemp, 2000), the cut-cell method, which includes the shaved cell formulation in the MITgcm (Adcroft et al., 1997) and the immersed boundary method (Tseng and Ferziger, 2003), have also been shown to be effective solutions. A comparison of cut-cell and σ-coordinate grids conducted by Shaw and Weller (Shaw and Weller, 2016) acknowledged the merits

and drawbacks of both techniques and did not find either method to be superior. An advantage of implementing the σ-coordinate mesh, versus a cut-cell formulation, in the 2DFV code is that it is a helpful intermediate step towards implementing a moving mesh with free-surface dynamics.

There are many well-known σ-coordinate models including the Regional Ocean Modeling System (ROMS; Mellor, 1998), the Penn State/NCAR Mesoscale Model (MM5; Grell et al., 1995), the Finite Volume Community Ocean Model (FVCOM; Chen et al., 2003a), the Non-Hydrostatic WAVE model (NHWAVE; Ma et al., 2012), and the Princeton Ocean Model (POM; Blumberg and Mellor, 1987). These models vary in terms of the dynamics they represent (hydrostatic versus non-hydrostatic) and numerical methodology (finite volume versus finite difference), and therefore all manage challenges imposed by a σ-coordinate mesh in slightly different ways. In chapter 3 we describe solutions to challenges which arose in our σ-coordinate model, including how the algorithm maintained both de-coupled horizontal and vertical momentum equations and fully implicit diffusion scheme from the z-coordinate 2DFV code. Additionally, we explain the way the issue of hydrostatic inconsistency was ameliorated.

1.0.3 The DO Field Equations

For the first time we utilize the DO field equations to study the sensitivity of internal tides to perturbations in the stratification. In these preliminary simulations, we initialize variability in the density field near the peak of a steep sea mount, representing sub-grid scale eddies that stir the fluid in this region, and find that initial perturbations to this region result in greater mixing potential after tidal forcing compared to the unperturbed deterministic run in chapter 2. While the stochastic simulations will require more rigorous verification than is presented in this thesis to ensure the fidelity of the simulations, the first few verification study presented in chapter 4 is encouraging.

Chapter 2

Verification with Linear Theory and MITgcm

Internal gravity waves are generated when a stably stratified fluid is perturbed such that a heavier parcel of fluid moves over a parcel of lighter fluid. Gravity works to restore the stably stratified configuration, and the resultant oscillations are internal gravity waves. A primary generation mechanism of internal waves in both the ocean and atmosphere, and the mechanism which we consider in this thesis, is stratified fluid flow over rough topography resulting in Lee waves downstream.

In this chapter we demonstrate that the 2D Finite Volume framework (2DFV) accurately captures internal wave dynamics using three resources: the internal wave dispersion relation, linear theory developed by Khatiwala (2003), and a simulation performed by Legg and Huijts (2006) on the MITgcm .

In the first section we describe the setup of a test-case from Legg and Huijts (2006) in the 2DFV framework. In the second section, we briefly review the derivation of the internal wave dispersion relation and verify the angle of propagation and wavelength of the internal tides in the numerical simulations is correct. Next we confirm linear theory is an appropriate approximation with the power spectra of the wavefield. Finally, we compare the particle velocity fields between the 2DFV and MITgcm simulations.

2.1 Problem Setup

We reproduce the "Tall-Narrow" test-case from (Legg and Huijts, 2006), where the barotropic, M2 tide (period of 12.4 hours) oscillates over an idealized Gaussian topography of height h_0 $\exp\left(\frac{-(x)^2}{2L^2}\right)$. The dynamical regime of this test-case is characterized by two dimensionless numbers, the Froude number (F_r) and the tidal excursion parameter (R_L), which are defined as

$$F_r = \frac{U_0}{\sqrt{gh_0}} \qquad R_L = \frac{U_0}{\omega L}$$

where U_0 is the magnitude of the barotropic tide, ω is the tidal frequency, h_0 is the height of the topography, g is gravitational acceleration, and L is approximately the e-folding length of the Gaussian topography. The Froude number is a measure of mixing potential; narrower topographies have more mixing potential than wider topographies. The tidal excursion parameter is a measure of topographic impedance to the flow; tall topographies impede flow more than shorter topographies. The Tall-Narrow (TN) topography has $h_0 = 2350$m and $L = 1215$m. With a 2 cm/s barotropic tide, we have a Froude number of 0.011 and a tidal excursion parameter of 0.117.

The fluid is initially at rest with a linear background stratification corresponding to a constant Brunt-Väisälä frequency (N)

$$\frac{\partial}{\partial y}\rho'(x, y, t = 0) = -\frac{\rho_0 N^2}{g} \tag{2.1}$$

where the total density ρ is the sum of ρ_0, the constant background density, and ρ', the density anomaly. The initial pressure field is hydrostatic such that

$$\frac{\partial}{\partial y}p(x, y, t = 0) = -\rho g. \tag{2.2}$$

We add the tidal forcing term from Legg and Huijts (2006) and Khatiwala (2003) to the horizontal momentum equation such that $\boldsymbol{F}(\boldsymbol{x}, t)$ in Equation 2.5 is

$$\begin{bmatrix} F_u \\ F_v \end{bmatrix} = \begin{bmatrix} U_0\omega\cos(\omega t) \\ 0 \end{bmatrix}. \tag{2.3}$$

The tidal velocity can be found by differentiating Equation 2.3, $u_{barotropic} = U_0 \sin(\omega t)$.

The 2DFV Code

The 2DFV code solves the incompressible Navier-Stokes equations with the eddy viscosity and Boussinesq approximations. For our simulation we have

$$\nabla \cdot \boldsymbol{u} = 0, \tag{2.4}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} = -\nabla p' + \nu \nabla^2 \boldsymbol{u} + \frac{\rho'}{\rho_0} \boldsymbol{g} + \boldsymbol{F}(\boldsymbol{x}, t), \tag{2.5}$$

$$\frac{\partial \rho'}{\partial t} + \boldsymbol{u} \cdot \nabla \rho' = 0. \tag{2.6}$$

where density (ρ) is defined $\rho_{\text{total}} = \rho_0 + \rho'(x, y, t)$ and $\rho = \frac{\rho'}{\rho_0} g$ such that $\frac{|\rho'|}{|\rho_0|} \ll 1$, and pressure is defined $p = p_0 + p' \rho_0$ such that $\nabla p_0 = -\rho_0 \boldsymbol{g}$ and $\frac{|p'|}{|p_0/\rho_0|} \ll 1$. The boundary conditions on velocity, shown in Figure 2-1, are zero-Neumann conditions on the inlet and outlet ($\partial u/\partial x = \partial v/\partial x = 0$), free-slip, no-penetration conditions on the top and bottom of the domain ($\partial u/\partial y = v = 0$), and no-slip conditions on the sea mount ($u = v = 0$). The boundary conditions on the pressure-correction term (q) are imposed by the velocity boundary conditions according to the pressure-correction formulation described in Appendix A. The pressure-correction term has zero-Neumann conditions on the top of the domain, bottom of the domain, and sea mount, and open ($\frac{\partial^2 q}{\partial x^2} = 0$) conditions on the left and right domain boundaries. The boundary conditions on density are zero-Neumann on all boundaries. Simulation results for the velocity and density fields after 13.9 hours of tidal forcing are shown in Figure 2-2. We notice the zonal velocity field is antisymmetric about the horizontal centerline (at y = -2350 m depth) and symmetric about the vertical centerline (at x = 0 km). The azimuthal symmetric about the horizontal centerline velocity is antisymmetric about the vertical centerline. For these relatively weak internal tides, the undulations in the density field are very mild.

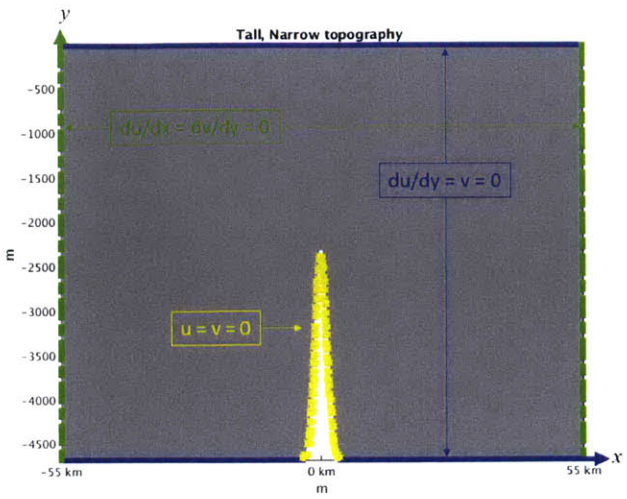


Figure 2-1: A schematic illustrating the boundary conditions on velocity in the 2DFV simulation.

2.2 A Cut-Cell Boundary Condition for a Structured Finite Volume Mesh

The simulations performed by Legg and Huijts (2006) impose free-slip velocity boundary conditions on the sea mount, as opposed to the no-slip conditions described in the previous section. If we wish to impose free-slip conditions on the topography, one option is to use a cut-cell method. Instead of requiring interpolations like the immersed ghost cell boundary condition (Tseng and Ferziger, 2003), we can solve for the values of boundary cells already present in the structured finite volume mesh such that the stress-free condition is observed. Figure 2-3 shows a few cells on either side of a masked region. The cut-cell algorithm we propose solves implicitly for the boundary values (v_2, v_4, v_6, u_3, u_5 in Figure 2-3) according to the stress-free, no-penetration conditions

$$\vec{u} \cdot \hat{n} = 0 \tag{2.7}$$

$$\frac{\partial}{\partial n}(\vec{u} \cdot \hat{t}) = 0 \tag{2.8}$$

where $\hat{n}$ and $\hat{t}$ are a unit vectors in the normal and tangential directions, respectively.

2.2.1 Review of the Analytical Solution

The test-case with which we choose to study this new cut-cell boundary condition is Couette flow between two concentric cylinders. The analytical solution for incompressible flow between

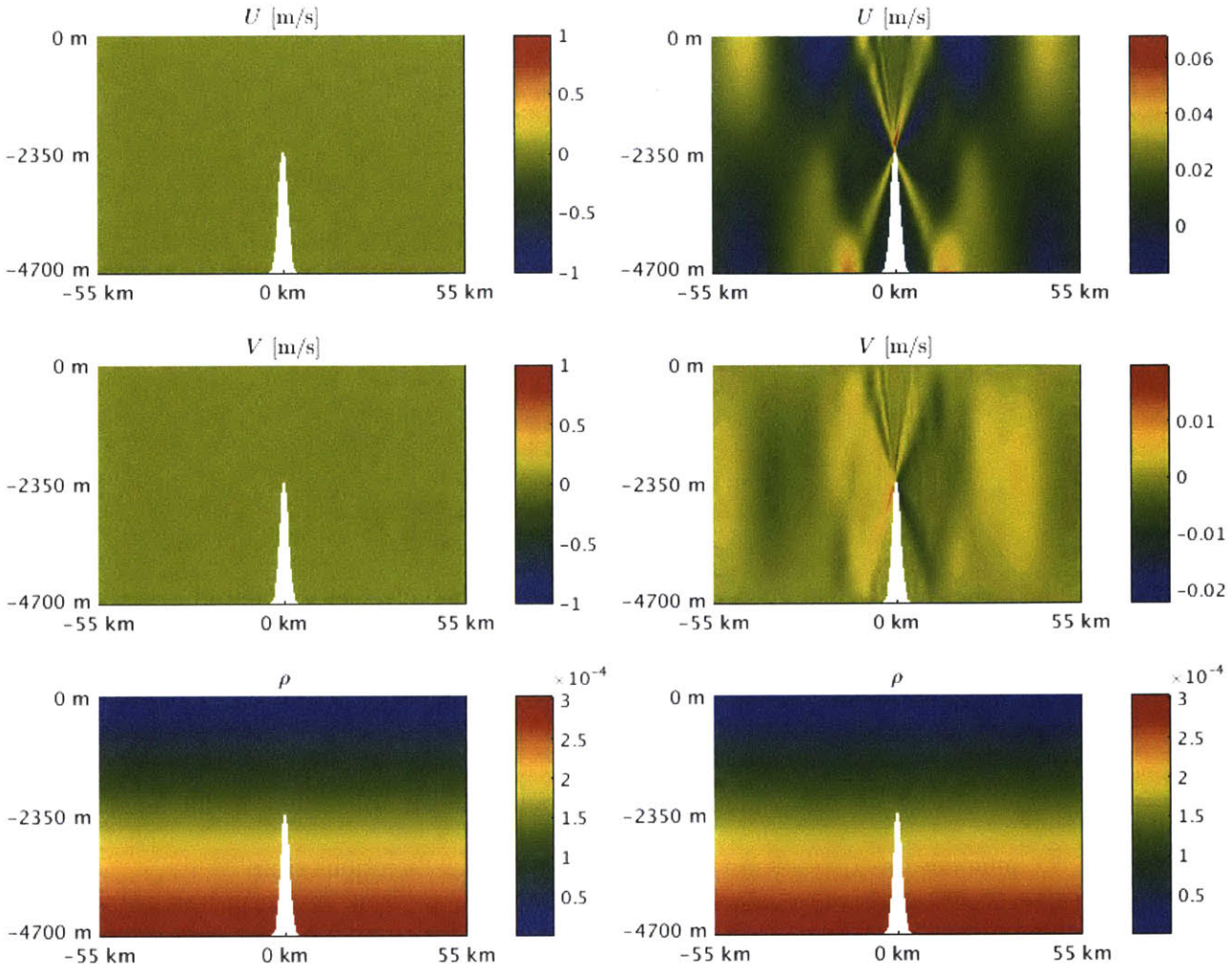


Figure 2-2: The zonal velocity (top), azimuthal velocity (middle), and density (bottom) are shown fields initially (left) and after 13.9 hours of tidal forcing (right).

concentric cylinders in polar coordinates is outlined below.

The inner cylinder has no-slip boundary conditions and rotates at a fixed angular velocity (Ω); the outer cylinder has free-slip, no-penetration boundary conditions as shown in Figure 2-4a. Since this test-case is radially symmetric and has a steady-state solution, we expect derivatives in the θ and time to vanish. Thus the continuity equation simplifies to

$$\frac{1}{r}\frac{\partial}{\partial r}(\rho r u_r) = 0 \tag{2.9}$$

where u_r is the velocity in the radial direction and ρ is the fluid density. Equation 2.9 simplifies to $ru_r = C_1$, where C_1 is a constant. Since we enforce no-penetration at both boundaries ($u_r(r =$

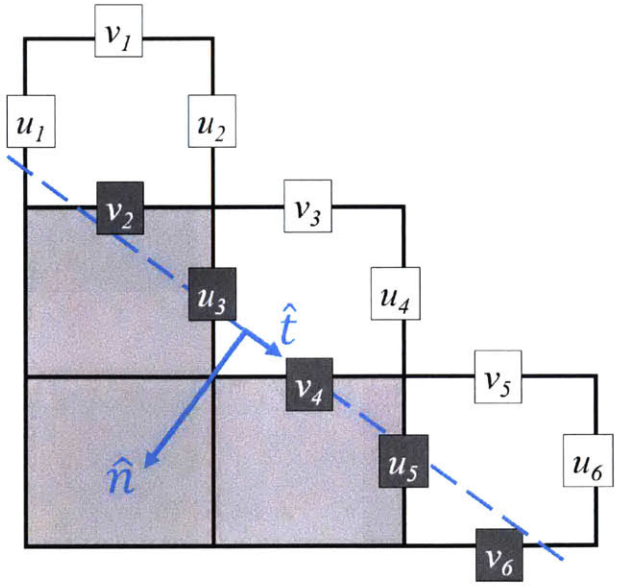


Figure 2-3: The structured C-grid mesh considers the center of the velocity cells to be at the center of the scalar grid cell faces. Boundary cells on the scalar grid and on the velocity grids are shown in grey, and the analytical boundary of the masked region is represented by the blue dashed line. The local normal and tangential directions are identified with blue arrows.

$R_1) = u_r(r = R_2) = 0)$ we see that the radial velocity must vanish everywhere in the flow.

Next, the momentum equations in the radial and angular directions simplify to

$$-\rho\frac{u_\theta^2}{r} = -\frac{\partial p}{\partial r} \tag{2.10}$$

$$0 = \mu\left\{\frac{\partial}{\partial r}\left[\frac{1}{r}\frac{\partial}{\partial r}(ru_\theta)\right]\right\} \tag{2.11}$$

where u_θ is the angular velocity, μ is the dynamic viscosity, and p is the pressure. Equation 2.11 can be easily solved by integrating twice such that the solution to the annular Couette flow test-case is

$$u_r = 0 \tag{2.12}$$

$$u_\theta = rC_2 + \frac{C_3}{r}. \tag{2.13}$$

The boundary conditions for this flow are

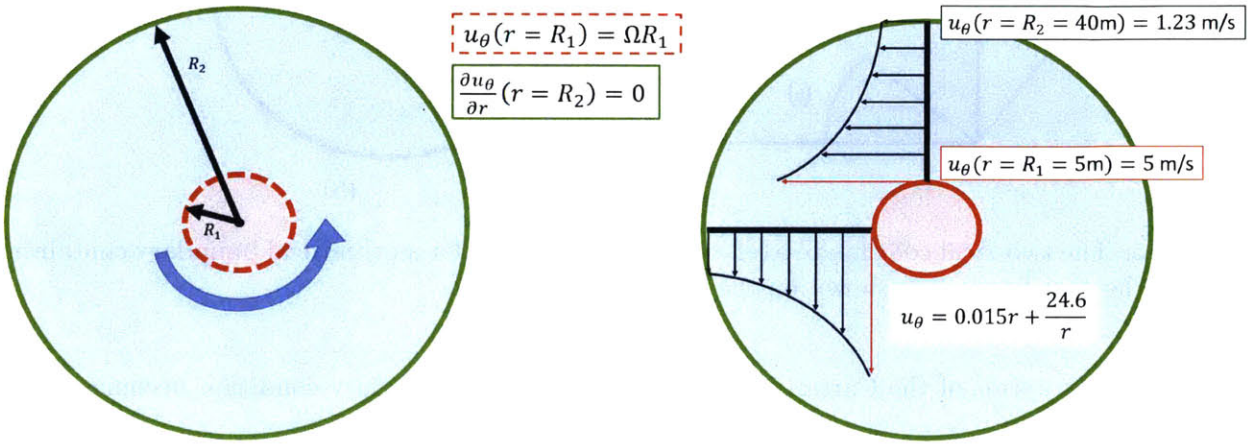
$$u_\theta(r = R_1) = \Omega R_1 \tag{2.14}$$

$$\frac{\partial u_\theta(r = R_2)}{\partial r} = 0 \tag{2.15}$$

so the constants C_2 and C_3 are

$$C_2 = \frac{R_1^2 \Omega}{R_1^2 + R_2^2} \tag{2.16}$$

$$C_3 = \frac{R_1^2 R_2^2 \Omega}{R_1^2 + R_2^2}. \tag{2.17}$$



(a) (b)

Figure 2-4: Two schematics of the setup of the annular Couette flow (a) and a sketch of the solution for $R_1 = 5$ m, $R_2 = 40$ m, and $\Omega = 1$ rad/s. The solution for velocity everywhere in the flow is in the third box of (b).

2.2.2 The Algorithm

If we know the analytical function for the boundary (e.g. a Gaussian, or a cylinder), then we can calculate exactly the angle θ between the (x, y) coordinate system and the (n, t) coordinate system as shown in Figure 2-5. This cut-cell algorithm effectively considers the analytical boundary to be composed of piecewise linear segments rather than discrete and even steps. In this particular test-case (with the origin at the center of cylinders) the radial and angular directions are exactly the normal and tangential directions. The rotation matrix which converts velocities between the Cartesian (u, v) and normal-tangential (u_n, u_t) coordinate systems is

$$\begin{bmatrix} u_n \\ u_t \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}. \tag{2.18}$$

We can use the rotation matrix in Equation 2.18 to write the boundary conditions in Equations

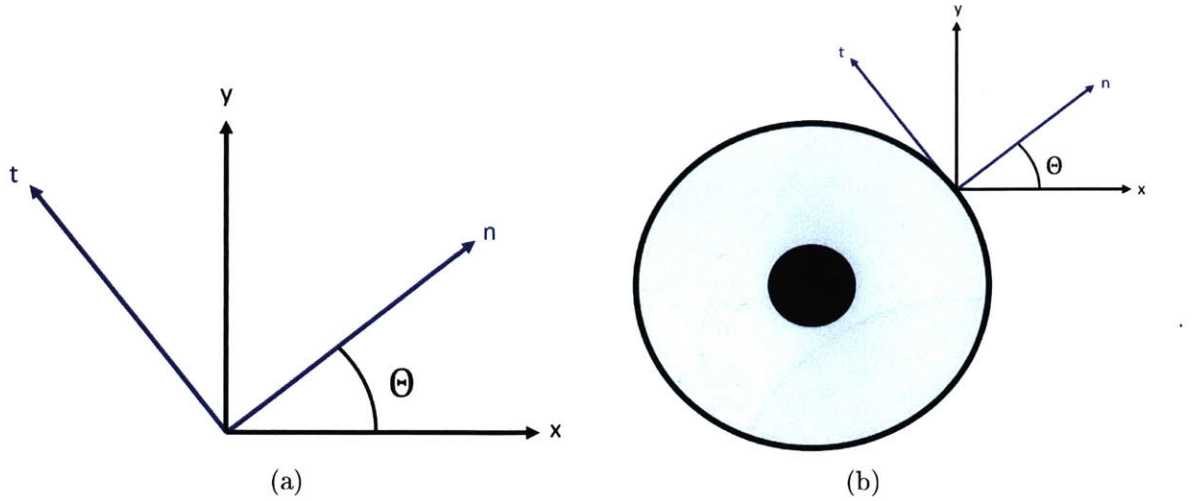


Figure 2-5: The two local coordinate axes are shown in (a) and for a cylindrical boundary containing fluid (b) the coordinates are shown on the boundary.

2.12 and 2.15 in terms of the Cartesion velocities. The normal boundary condition becomes

$$u_n = 0 \tag{2.19a}$$

$$u\cos(\theta) + v\sin(\theta) = 0, \tag{2.19b}$$

and the tangential boundary condition becomes

$$\frac{\partial}{\partial n}(u_t) = 0 \tag{2.20a}$$

$$\frac{\partial}{\partial n}\left(-u\sin(\theta) + v\cos(\theta)\right) = 0 \tag{2.20b}$$

$$-\frac{\partial u}{\partial n}\sin(\theta) + \frac{\partial v}{\partial n}\cos(\theta) = 0 \tag{2.20c}$$

$$-\left(\frac{\partial u}{\partial x}\frac{\partial x}{\partial n} + \frac{\partial u}{\partial y}\frac{\partial y}{\partial n}\right)\sin(\theta) + \left(\frac{\partial v}{\partial x}\frac{\partial x}{\partial n} + \frac{\partial v}{\partial y}\frac{\partial y}{\partial n}\right)\cos(\theta) = 0 \tag{2.20d}$$

$$-\sin(\theta)\left(\frac{\partial u}{\partial x}\cos(\theta) + \frac{\partial u}{\partial y}\sin(\theta)\right) + \left(\frac{\partial v}{\partial x}\cos(\theta) + \frac{\partial v}{\partial y}\sin(\theta)\right)\cos(\theta) = 0 \tag{2.20e}$$

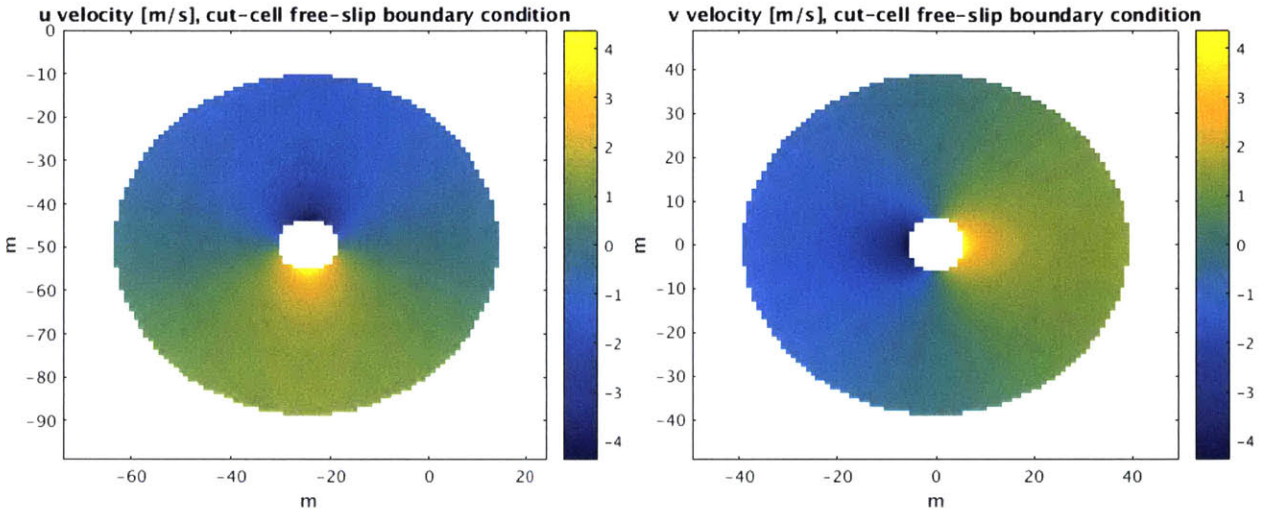
$$-\sin(\theta)\cos(\theta)\frac{\partial u}{\partial x} - \sin^2(\theta)\frac{\partial u}{\partial y} + \cos^2(\theta)\frac{\partial v}{\partial x} + \sin(\theta)\cos(\theta)\frac{\partial v}{\partial y} = 0. \tag{2.20f}$$

The two conditions (2.19b and 2.20f) couple the u and v velocities, and so the solution for the predictor velocity (see Appendix A) is nolonger two separate linear systems for u and v, but is solved as one coupled system. At each timestep we solve implicitly for both the interior predictor

velocity values, as well as for velocities in the boundary cells that form the perimeter of the outer cylinder. All of the derivatives in Equation 2.20f are computed with first-order accuracy.

2.2.3 Results

After the simulation has reached a steady state the numerical velocity, shown in Figure 2-6, is compared with the analytical solution. Samples of the velocity field along two radii of the setup



(a) (b)

Figure 2-6: The numerical u (a) and v (b) velocity fields for annular Couette flow with the free-slip boundary condition on the outer cylinder after the simulation has reached steady state agrees with the overall profile of the analytical solution, but the numerical solution is a little faster than the analytical solution.

are shown in Figure 2-7 are compared with the analytical solution. The velocity profile respects the boundary conditions (Equations 2.12, 2.14, 2.15), but the numerical velocity field is slightly faster than the analytical field. The maximum relative error, found in the cells adjacent to the outer cylinder, is approximately 13 %.

A more sophisticated algorithm and careful verification needs to be conducted to ensure accuracy of the cut-cell boundary condition. Potential explanations for the discrepancy between the numerical and analytical solutions include truncation error, which can be confirmed with a grid-refinement study and ameliorated by increasing the order of accuracy of the computed derivatives. We will proceed to apply with new boundary condition to the Low, Wide topography with a 2 cm/s tide, cognizant that the accuracy of these results is predicated on a thorough evaluation of

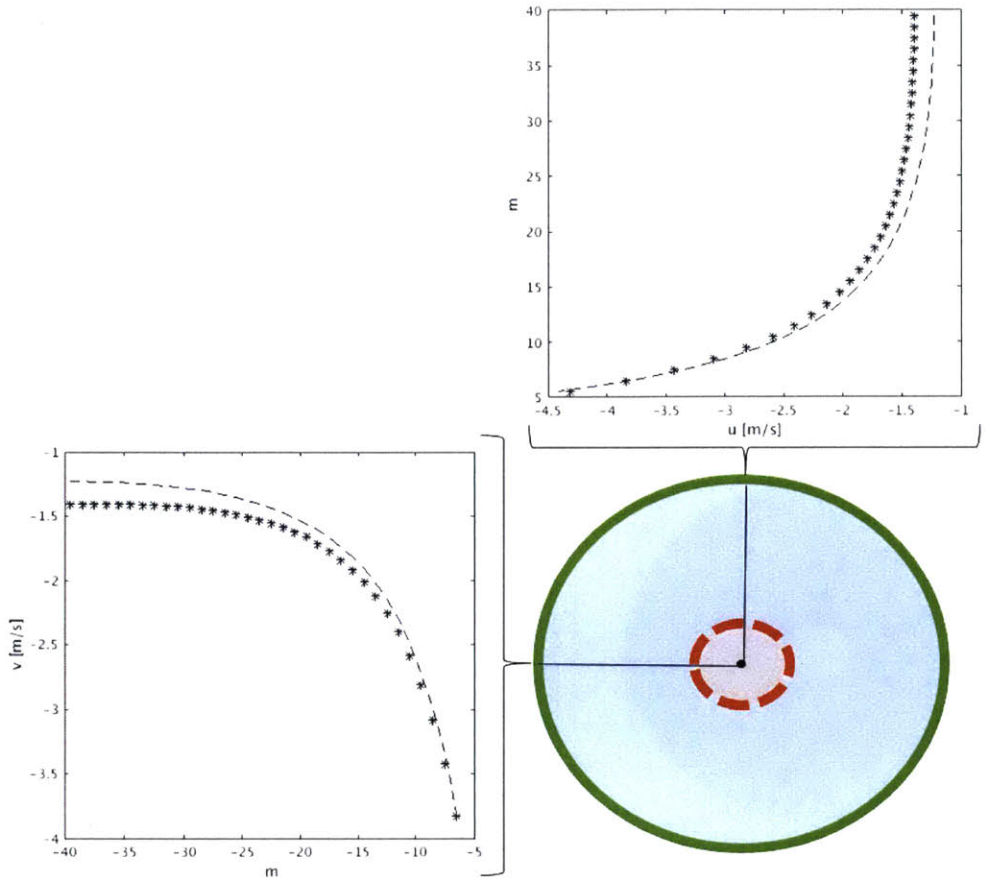
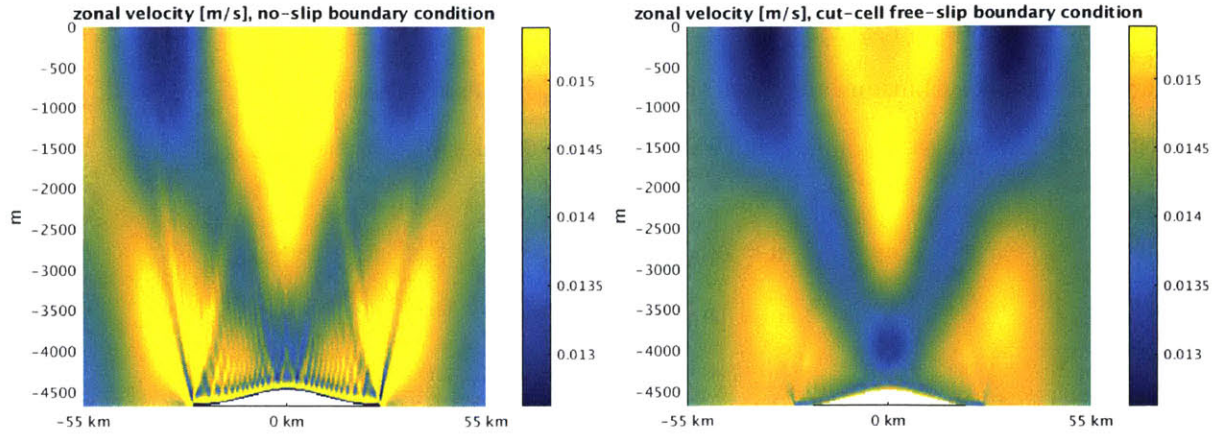


Figure 2-7: The numerical solution in the cells identified by the black lines in the diagram are shown above and to the left of the annular Couette flow sketch. The plots show that near the outer cylinder the boundary condition is observed, but there is approximately a 13 percent error in the velocity value.

the cut-cell boundary condition. Figure 2-8 shows the free-slip boundary condition significantly improves but does not eliminate the numerical artifacts emanating from the Low, Wide topography. Improvements to the accuracy of the algorithm, such as computing the derivatives with a higher order of accuracy, may help further ameliorate the observed numerical artifacts and improve the representation of internal waves in the 2DFV code.

2.3 Verification

We have identified four metrics to evaluate the ability of the 2DFV code to sufficiently represent the dynamics of internal tides. The first two, angle of propagation and horizontal wavelength, confirm the stratification is correct and that advective and diffusive transport are dominated by the



(a) (b)

Figure 2-8: The Low, Wide topography with a no-slip velocity boundary condition on the topography (a) shows numerical artifacts after 13.9 hours of tidal forcing. The cut-cell free-slip velocity boundary condition (b) improves the representation of the internal waves.

pressure gradient force. Next we study the power spectra of the wavefield to confirm that linear theory is a reasonable simplification. Finally, the particle velocity fields from the 2DFV code and the MITgcm are compared, and discrepancies are discussed.

2.3.1 Verification of Wavebeam angle with Linear Theory

First, we overview the derivation of the internal wave dispersion relation, which can be found in many geophysical fluid dynamics textbooks, including Cushman-Roisin and Beckers (2011), and use it to evaluate the angle of propagation of the internal tides. If we consider the regime where we can ignore ambient rotation (large Rossby number), dissipative effects (small Ekman number), non-linearity (the wave amplitude is much smaller than the wavelength), and make the Boussinesq approximation, we can simplify the primitive equations to the following system:

$$\frac{\partial u}{\partial t} = -\frac{1}{\rho_0}\frac{\partial p'}{\partial x} \tag{2.21}$$

$$\frac{\partial v}{\partial t} = -\frac{1}{\rho_0}\frac{\partial p'}{\partial y} \tag{2.22}$$

$$\frac{\partial w}{\partial t} = -\frac{1}{\rho_0}\frac{\partial p'}{\partial z} - \frac{1}{\rho_0}g\rho' \tag{2.23}$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \tag{2.24}$$

$$\frac{\partial \rho'}{\partial t} + w\frac{d\overline{\rho}}{dz} = 0. \tag{2.25}$$

where $\rho = \rho_0 + \overline{\rho}(z) + \rho'(x,y,z,t)$ and $p = p_0(z) + p'(x,y,z,t) = -\int \rho_0 g dz + p'(x,y,z,t)$. The magnitudes of the perturbations are much smaller than the mean values $\frac{|\rho'|}{|\rho_0|} \ll 1$, $\frac{|\rho'|}{|\rho(z)|} \ll 1$, $\frac{|p'|}{|p_0|} \ll 1$. This system of five constant coefficient PDE's above has solutions of the form

$$u = \mathcal{U}(y,z)e^{i(k_x x + k_y y + k_z z - \omega t)} \tag{2.26a}$$

$$v = \mathcal{V}(x,z)e^{i(k_x x + k_y y + k_z z - \omega t)} \tag{2.26b}$$

$$w = \mathcal{W}(x,y)e^{i(k_x x + k_y y + k_z z - \omega t)} \tag{2.26c}$$

$$p' = \mathcal{P}(t)e^{i(k_x x + k_y y + k_z z - \omega t)} \tag{2.26d}$$

$$\rho' = \mathcal{R}(x,y)e^{i(k_x x + k_y y + k_z z - \omega t)}. \tag{2.26e}$$

Writing the system 2.26 in matrix form we have

$$\begin{bmatrix} -i\omega & 0 & 0 & \frac{k_x}{\rho_0} & 0 \\ 0 & -i\omega & 0 & \frac{k_y}{\rho_0} & 0 \\ 0 & 0 & -i\omega & \frac{k_z}{\rho_0} & \frac{g}{\rho_0} \\ k_x & k_y & k_z & 0 & 0 \\ 0 & 0 & \frac{d\overline{\rho}}{dz} & 0 & -i\omega \end{bmatrix} \begin{bmatrix} \mathcal{U}(y,z) \\ \mathcal{V}(x,z) \\ \mathcal{W}(x,y) \\ \mathcal{P}(t) \\ \mathcal{R}(x,y) \end{bmatrix} = \vec{0}. \tag{2.27}$$

If we set the determinate of the matrix in 2.27 to zero, and solve for ω, we find the dispersion relation

$$\omega^2 = \left(\frac{g}{\rho_0}\frac{d\overline{\rho}}{dz}\right)\frac{k_x^2 + k_y^2}{k_x^2 + k_y^2 + k_z^2}. \tag{2.28}$$

If we consider a 2D case ($k_y = 0$) and take γ as the angle between the horizontal and the wavenumber vector (k_x, k_z), the dispersion relation can be simplified to

$$\omega = N\cos(\gamma), \tag{2.29}$$

where $N^2 = -\left(\frac{g}{\rho_0}\frac{d\overline{\rho}}{dz}\right)$. For a buoyancy frequency of $N = 8(10^{-4})$ rad/s, and the M@ tidal frequency of $\omega = 1.41(10^{-4})$ rad/s, we expect the phase velocity vector to propagate at an angle of 1.39 radians (79.84°) to the horizontal. Recall that the phase velocity propagates at 90° to the group velocity, so a line of constant phase will be at $90° - \gamma$. Let $\theta = 90° - \gamma$ so that $\omega = N\cos(90° - \gamma) = N\sin(\theta)$.

For the aformentioned values of N and ω, $\theta = 10.2°$. In Figure 2-10 we superpose dashed lines at an

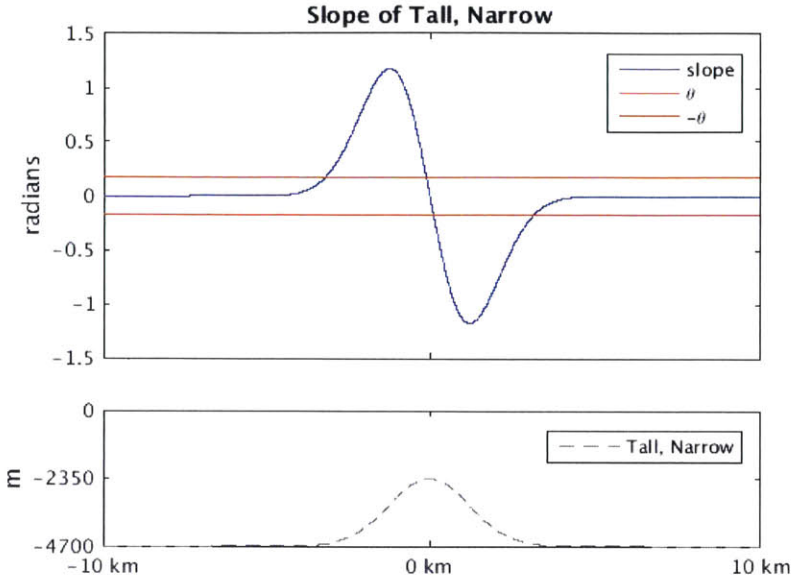


Figure 2-9: The slope of the TN topography is shown in blue alongside the theoretical angle of propagation, $\theta = 0.178$ radians. We see the topography is supercritical, or that the slope of the topography exceeds the slope of the internal tides, within approximately three kilometers of the apex.

angle θ on to a snapshot of the zonal velocity field from the 2DFV code. Note that the topography is supercritical within 3 kilometers of the apex, as seen in Figure 2-9. We observe excellent agreement

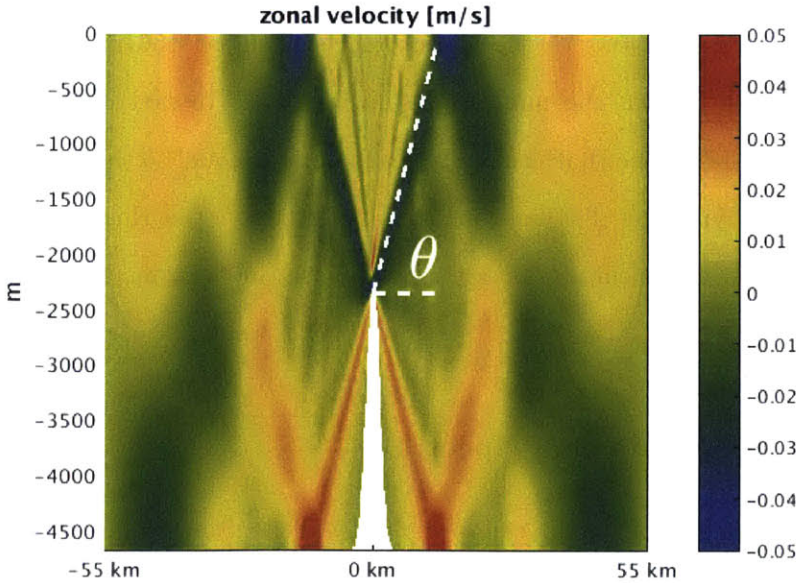


Figure 2-10: The dashed lines forming an angle, θ, are superposed onto a snapshot of the 2DFV zonal velocity field of the TN topography ($U_0 = 2$cm/s) simulation after 13 hours of tidal forcing. The lines of constant phase agree well with the superposed dashed line.

between the line of constant phase and the wave trajectory, and conclude that the waves produced in this simulation are well-predicted by the linear theory derived above.

2.3.2 Verification of Horizontal Wavelength with Linear Theory

We will briefly describe the derivation performed by Khatiwala (2003) for the internal tide wavenumber given an oscillating flow over an arbitrary topography. For a 2D domain, we can describe an arbitrary topography with a function $h(x)$ and a tide in the form of a time-dependent body force $U(t) = U_0\omega\cos(\omega t)$, oscillating the fluid above the topography. After a Galilean transformation to a reference frame that moves with the tide, Khatiwala takes a Fourier transform in both spatial dimensions of the momentum equations and solves for the horizontal wavenumber,

$$k_{jn} = \frac{j\pi}{H}\left[\frac{n^2\omega^2 - f^2}{N^2 - n^2\omega^2}\right]^{1/2},$$

where H is the maximum fluid depth, n is the vertical mode number, and j is the horizontal mode number. We assume no damping in this system. The analytical, horizontal wavelength for the first mode ($j = n = 1$), can be found from $k_x = 2\pi/\lambda_x$. The first mode will have the longest wavelength and so it will travel the fastest and be the first wave we will clearly observe at the edge of the wavefield. To find the numerical wavelength from the 2DFV simulation, we create a Hovmöller plot from horizontal cross-sections of the zonal velocity field at 1567m depth from the surface. The horizontal wavelength from the numerical solution is the distance between two nulls in the Hovmöller. In Figure 2-11 we see that the analytical solution (equal to the length of the superposed black line) agrees well with the distance between two nulls in the Hovmöller. We conclude the horizontal wavelength from the simulation agrees well with the horizontal wavelength prescribed by linear theory.

2.3.3 Verification of Frequency Spectra

If the frequency spectra of the waves is principally at the tidal frequency, and higher harmonics thereof, linear theory is an appropriate simplification. Energy between harmonics of the tidal frequency can be attributed to non-linearities associated with the internal waves that are spawned by internal tides (Helfrich and Grimshaw; 2008). Figure 2-12 shows power spectra of the horizontal

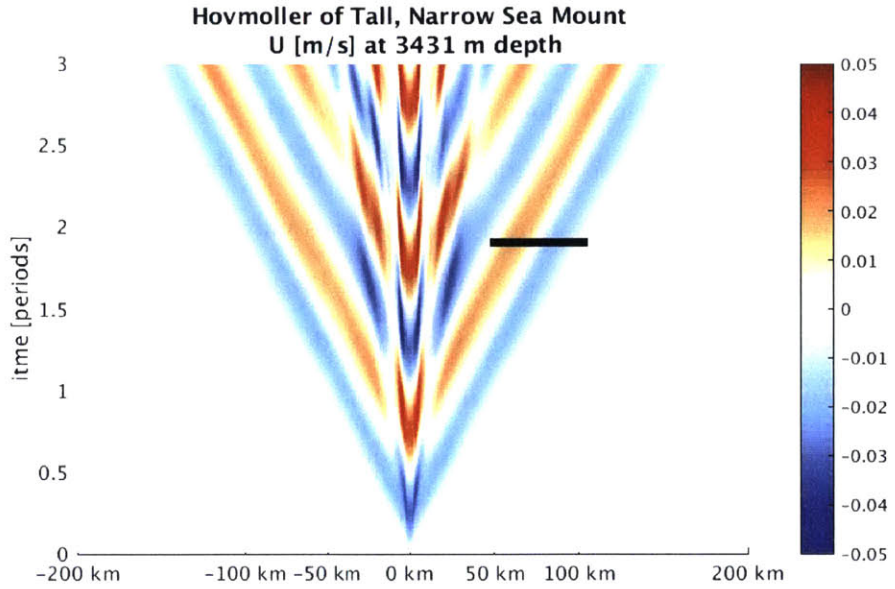


Figure 2-11: The analytical horizontal wavelength, equal to the length of the superposed black line, agrees well with the wavelength in the numerical simulation.

and vertical velocity fields have peaks at the tidal frequency and higher harmonics. There appears to be slightly more energy between peaks in the 2DFV simulation than in the MITgcm simulation (see Figure 9B in Legg and Huijts (2006)).

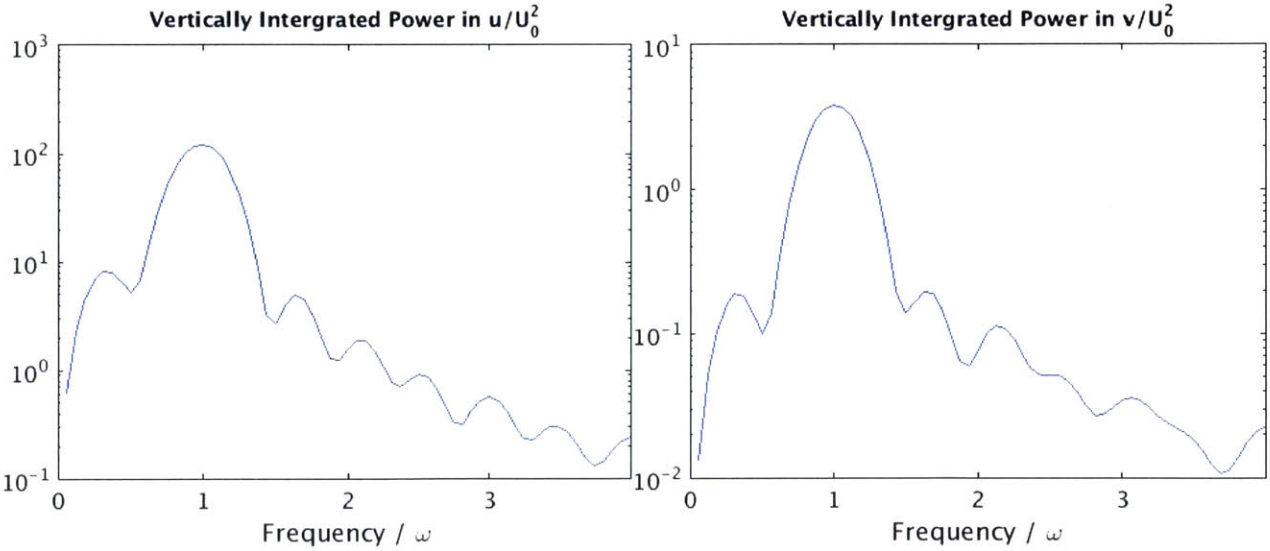


Figure 2-12: The vertically integrated power spectra of the horizontal (left) and vertical (right) velocity field, normalized by the barotropic velocity, at 11 km from the TN topography reveals a large peak at the forcing frequency (ω) and monotonically lower peaks at increasingly higher harmonics confirming linear theory is appropriate.

2.3.4 Verification of Particle Velocity Fields

Finally, we compare the particle velocity fields between the 2DFV and MITgcm simulations. Since this metric is most sensitive to the details of the model implementation, we discuss the properties of both frameworks and how model discrepancies may affect the result. A snapshot of the zonal total velocity field after 13.9 hours in Figure 2-13 shows very good agreement between the simulations, but the velocity fields from the 2DFV code are a little larger at the intersection between the wavebeams and the domain boundary (about 10 km to the left and right of the sea mount on both the top and bottom of the domain). Additionally, the center of the waves radiating away from the topography exhibit slightly larger velocities (about 50 km to the left and right of the sea mount on both the top and bottom of the domain).

The MITgcm is a 3D, non-hydrostatic, structured finite volume framework; the 2DFV code is a 2D, non-hydrostatic, structured finite volume framework. Descriptions of each model and the available numerical schemes can be found in Marshall et al. (1997) and Ueckermann and Lermusiaux (2012), respectively.

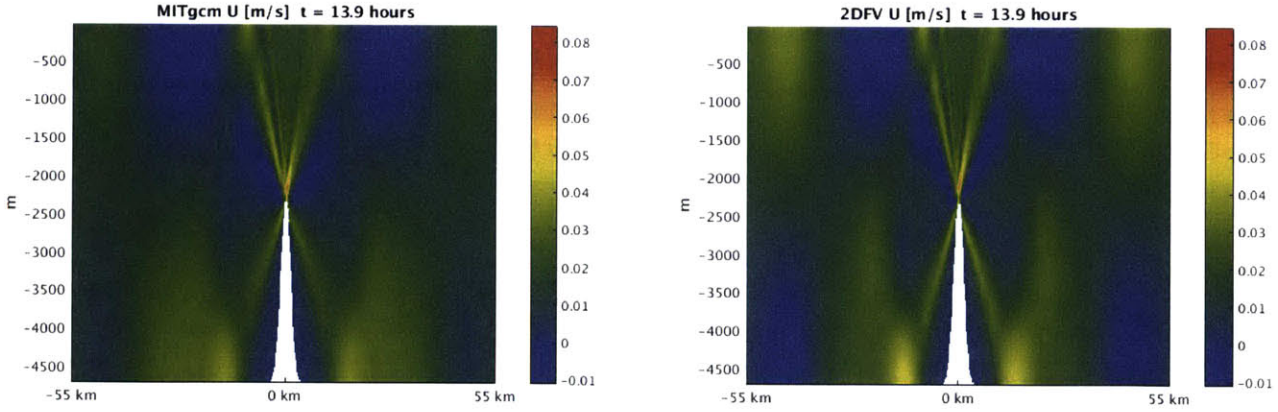


Figure 2-13: A comparison of the zonal velocity fields for the TN topography and a barotropic tide of 2cm/s between the MITgcm simulation (left) and the 2DFV simulation (right) after 13.9 hours of tidal forcing.

The simulations use the same grid size in the vertical (31.3 m) and approximately the same grid size in the horizontal (the resolution of the 2DFV code is 10% finer with 200 m cell widths). The 2DFV code differs in five respects from the MITgcm simulation which are enumerated in Table 2.1. We will address each of these discrepancies to evaluate their influence on the simulation results.

Table 2.1: Model Characteristics of the 2DFV code and the MITgcm

	2DFV	MITgcm
Coriolis Term f	0	$8(10^{-4})$
Surface Coundary Condition	rigid lid	free-surface
Inlet/Outlet Boundary Condition	sponge	radiative boudnary condition
Bathymetry Boundary Condition	no-slip	stress-free
Bathymetry Representation	z-coordinate (staircased)	shaved-cell

Coriolis terms in 2DFV and the MITgcm

The MITgcm simulation uses an f-plane formulation, where the Coriolis parameter, $f = 2\Omega \sin(\phi)$, is assumed to be constant. However, in a 2D flow with zonal and azimuthal directions, there is only one Coriolis term, and it appears in the zonal momentum equation. This term is multiplied by the meridional velocity, and thus, vanishes. While the reciprocal Coriolis term, $f_* = 2\Omega \cos(\phi)$, appears in both the zonal and azimuthal momentum equations, they are negligible compared to the Coriolis terms in most geophysical flows. A simple scaling argument demonstrates that a flow with a small height-to-width aspect ratio renders the reciprocal Coriolis terms, much smaller than the Coriolis terms. The aspect ratio of this simulation domain, 1:23, is very small. Since we can neglect the reciprocal Coriolis terms and the only Coriolis term vanishes, we can set $f = 0$ in the 2DFV code without influencing the results.

Top Boundary Condition

We use a scaling argument to demonstrate that the rigid lid boundary condition used in the 2DFV code is a good approximation for the TN topography with a 2cm/s barotropic tide. The MITgcm simulation solves for the free-surface parameter (η), and we will use the results from this simulation to demonstrate that free-surface effects on the flow are small compared to the other forcing terms. The evolution equation for the free surface height (η) is

$$\frac{\partial \eta}{\partial t} + \vec{u} \cdot \nabla \eta + \eta \nabla \cdot \vec{u} = 0 \tag{2.30}$$

for a velocity field $\vec{u}$.

If we non-dimensionalize equation 2.30 with characteristic scales:

$$t^* = \frac{t}{\tau} \qquad u^* = \frac{u}{U} \qquad w^* = \frac{w}{W}$$

$$x^* = \frac{x}{L} \qquad z^* = \frac{z}{H} \qquad \eta^* = \frac{\eta}{\Upsilon}$$

we have

$$\underbrace{\frac{\Upsilon}{\tau}\frac{\partial \eta^*}{\partial t^*}}_{①} + \underbrace{\frac{U\Upsilon}{L}u^*\frac{\partial \eta^*}{\partial x^*} + \frac{W\Upsilon}{H}w^*\frac{\partial \eta^*}{\partial y^*}}_{②} + \underbrace{\frac{U\Upsilon}{L}\eta^*\frac{\partial u^*}{\partial x^*} + \frac{W\Upsilon}{H}\eta^*\frac{\partial w^*}{\partial z^*}}_{③}, \tag{2.31}$$

and observe that

$$\frac{②}{①} = \frac{③}{①} = \frac{U\tau}{L} + \frac{W\tau}{H}. \tag{2.32}$$

We choose the characteristic values to be the barotropic tidal velocity (U = 0.02 m/s), the barotropic tidal period (τ = 12.4 hours), the internal tide wavelength (L = 52,800 m), and the domain depth (H = 4700 m). The magnitude of the vertical velocity is generally 10% of the horizontal velocity (recall the domain height-to-width aspect ratio of 1:23), so we choose W = (0.1) U = 0.002 m/s. Evaluating the terms in expression 2.32 with these characteristic values gives us two small numbers

$$\frac{U\tau}{L} \approx 0.016 \quad \text{and} \quad \frac{W\tau}{H} \approx 0.019.$$

Since $\frac{U\tau}{L}$ and $\frac{W\tau}{H}$ are much smaller than 1, it seems we can neglect terms ② and ③ with respect to ①, and so it seems we can neglect free surface effects. We also note that the slope of the free-surface is small; Figure 2-14 shows snapshots of the free surface elevation every 1.39 hours for the first 13.9 hours. The total variation in η over the 110 km domain is within 0.012 m of the resting surface height; a liberal estimate of the slope would be (0.012 − −0.012)m/110000m = $2.8(10^{-7})$. However, a free-surface would introduce a source of damping. Lack of a free-surface in the 2DFV code may partially account for the slightly higher velocities we see in the 2DFV simulation at $x = \pm 10km$.

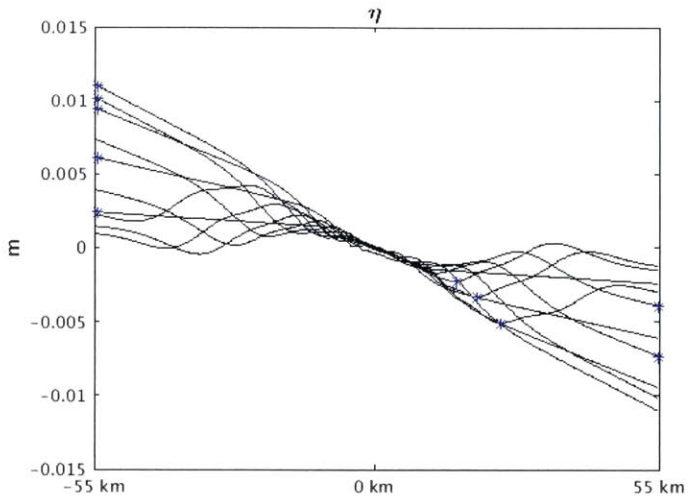


Figure 2-14: We plot snapshots of the free-surface displacement (left) at 10 evenly spaced intervals for the first 13.9 hours of the simulation. A $*$ identifies the maximum displacement of each free surface snapshot, and we observe the change in height to be much smaller than the domain width. Note that, since the tide starts flowing to the right, we expect the free-surface will first buldge up on the left of the topography.

Inlet and Outlet Boundary Conditions

The MITgcm simulation uses a radiative boundary condition at the inlet and outlet of the domain to prevent spurious reflections from the left and right boundaries. This boundary condition could damp the velocity field near the inlet and outlet of the domain, possibly accounting for the discrepancy between the 2DFV and MITgcm simulations at the $x = \pm 50km$. The 2DFV uses a sponge to similarly prevent waves from reflecting off the left and right boundaries and propagate back towards the generation site, but the 2DFV simulation is also initialized on a domain that is four times wider than the MITgcm (440 km). This is to ensure that internal tides generated at the topography have not had time to reach the left or right boundaries before the snapshots at 13.9 hours. The center 110 km is used for the comparisons. Additionally, the first mode internal tides that are generated with the aforementioned buoyancy frequency will have wavelengths of approximately 53 km. Therefore we extend the computational domain in order to observe a full wavelength on either side of the generation site that is not contaminated by the artificial boundary forcing. The difference in the inlet/outlet boundary conditions should not impact the accuracy of the internal tides, but may account for the discrepancy at the edge of the MITgcm simulation.

Table 2.2: Froude Number and Tidal Excursion Paramter for all four topographies tested by Legg and Huijts (2006), with a 2cm/s barotropic tide.

	F_r	R_L
Tall-Narrow	0.011	0.117
Tall-Wide	0.011	0.029
Low-Narrow	0.125	0.343
Low-Wide	0.125	0.014

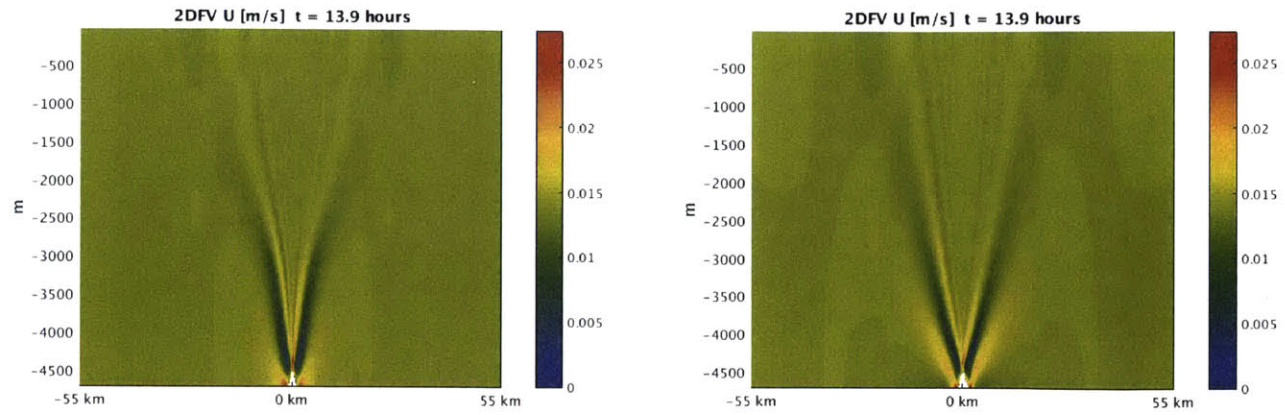
Bathymetry Representation and Boundary Condition

While the MITgcm simulation imposes a stress-free velocity boundary condition on a shaved-cell bathymetry (Alistar Adcroft and Marshall, 1997), the 2DFV code imposes a zero-Dirichlet velocity boundary condition on a staircased bathymetry.

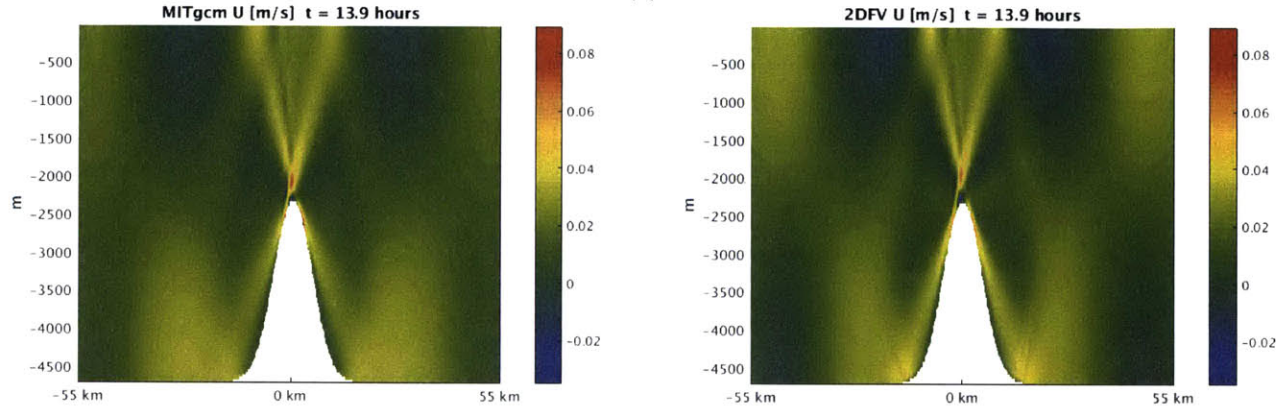
It is possible that the staircased representaton and no-slip boundary conditions results in numerical artifacts in the form of large vertical velocities near the topography contributing to the slightly larger velocities in the 2DFV code. This effect is difficult to observe with the TN test-case, so we examine all four topographies tested by Legg and Huijts (2006) with a barotropic tide of 2 cm/s; the corresponding F_r and R_L numbers are listed in Table 2.2. The MITgcm and 2DFV simulation results compare well for the Low-Narrow (LN), and Tall-Wide (TW) topographies, as shown in Figures 2-15a and 2-15b. However, we see numerical artifacts in the 2DFV simulation with the Low-Wide (LW) topography in Figure 2-15c due to the staircased representation of, and boundary condition on, the topography.

The influence of a staircased representation is going to be most noticeable in a low-aspect ratio topography because the streamlines are nearly horizontal and parallel, so the staircasing effects result in vertical velocities that significantly influence the flow. In the TN topography, the streamlines go up and over the topography, so staircasing effects are less noticeable. We can understand low-aspect topographies to be most different (and steep topographies most similar) between shaved-cell and staircased representations by considering the limiting case of a knife-edge topography. The representations of shaved-cell and z-coordinate topographies converge for a knife-edge topography, a unit-cell width boundary, and as the aspect ratio decreases, so does the z-coordinate approximation of the shaved-cell boundary.

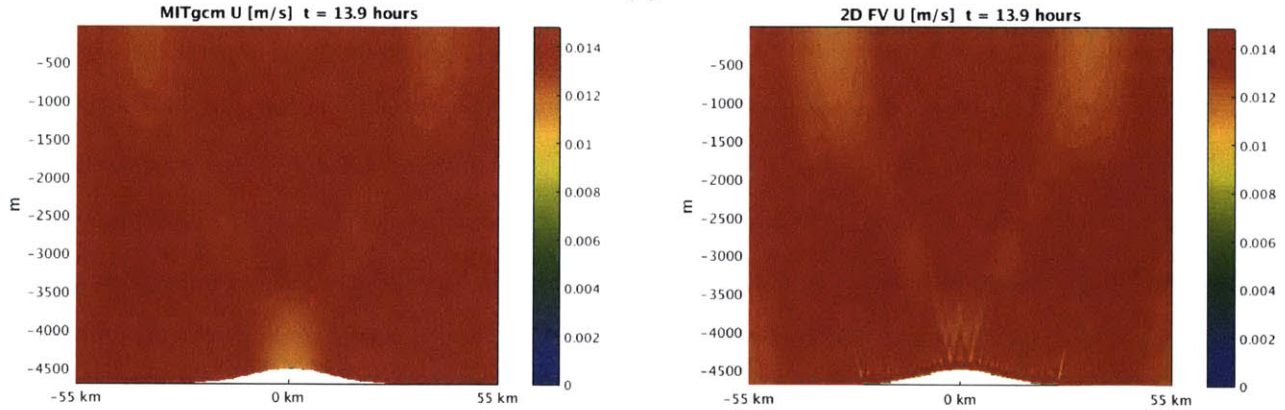
One solution to ameliorate staircasing effects is to refine the spatial grid; simulations with the LW topography and a reduced grid-cell size are shown in Figure 2-16. While grid refinement



(a)



(b)



(c)

Figure 2-15: A comparison of the zonal velocity fields between the MITgcm simulation (left) and the 2DFV simulation (right) for the LN (a), TW (b), and LW (c) topographies with a barotropic tide of 2cm/s after 13.9 hours of tidal forcing. Numerical artifacts near the LW bathymetry are visible.

does improve the results, it does not eliminate staircasing effects. We cannot afford to refine the grid by more than half (twice as many elements in each dimension) without shrinking the computational domain. Unfortunately shrinking the computational domain introduces boundary

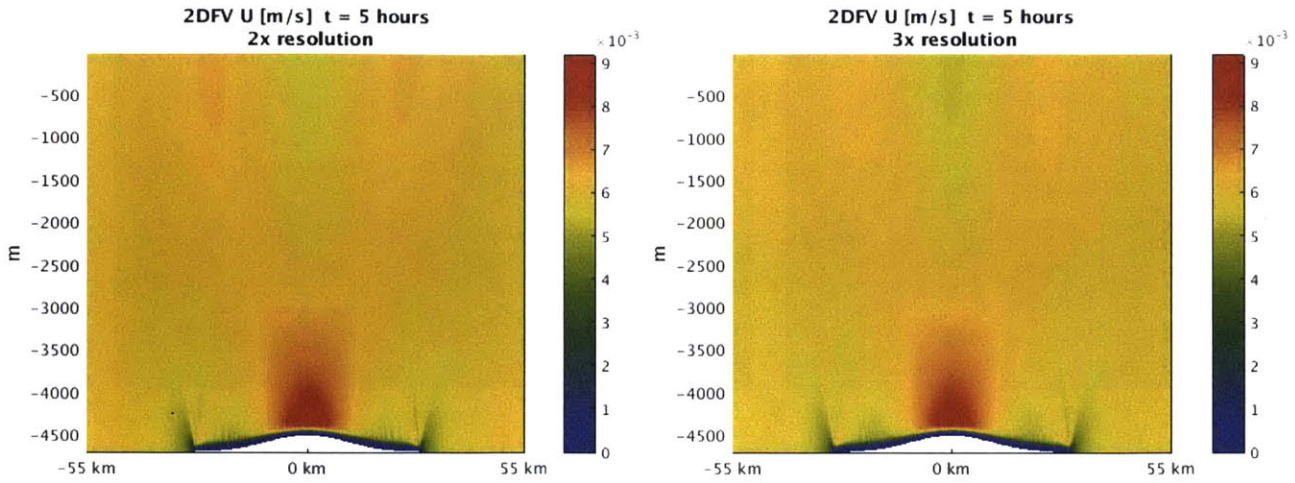


Figure 2-16: Both grid cell dimensions are reduced to half (left) and to one-third (right) of the original discretization. The no-slip condition and the stair-cased representation seem to affect the flow near the topography even for a substantially refined grid.

effects that contaminate the simulation much earlier, so the refined plots are snapshots after only five hours of tidal forcing.

2.4 Conclusion

We conclude that we can reproduce the correct internal tide dynamics for the TN topography with a 2cm/s barotropic tide based on our evaluation of the angle of propagation, horizontal wavelength, power spectra, and particle velocity field. The 2DFV code simulations exhibit slightly larger particle velocities, and one possible explanation is that the free-surface in the MITgcm simulation introduces a source of damping that is absent in the 2DFV simulation. The discrepancies are small enough to consider the 2DFV code capable of capturing internal tide dynamics with sufficient accuracy. Additionally, we found the staircased representation can lead to numerical artifacts that could noticeably disrupt the wavefield.

The particle velocity fields compare well for three of the four topographies studied by Legg and Huijts (2006). To improve the comparison for the LW topography we develop a σ-coordinate framework, described in the following chapter.

Chapter 3

A Non-hydrostatic, Finite Volume, σ-coordinate Framework

In this chapter we describe the σ-coordinate grid and algorithms for each term in the momentum and tracer-convection equations. Next we demonstrate the accuracy of the framework with two test-cases. Finally, we address the σ-coordinate pressure-gradient error and show the improvement that the σ-coordinate code has made from the z-coordinate framework.

3.1 The σ-coordinate Grid

Different definitions of the dimensionless coordinate σ are used to achieve specific mesh properties (Haley and Lermusiaux, 2010). The definition we use in this framework is

$$\tilde{x} = x \qquad \sigma = \frac{y}{H(x)}, \tag{3.1}$$

where $H(x)$ is the fluid depth, but the algorithm's architecture is generic such that re-definition of σ is simple to implement. Figure 3-1 shows a grid where ϕ is on the scalar grid of an Arakawa C-grid mesh (Arakawa and Lamb, 1977). Definition 3.1 creates the σ levels at evenly spaced fractions of the total fluid depth, where σ varies between 0 at the surface and -1 at the bathymetry. The horizontal discretization is uniform. The bathymetry is represented with piecewise-linear segments, in contrast to the staircased representation of a z-coordinate mesh, as seen in Figure 3-2.

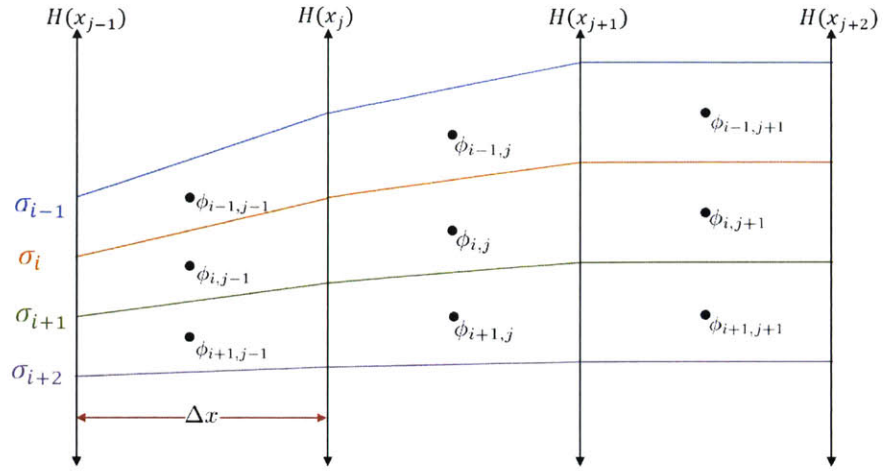
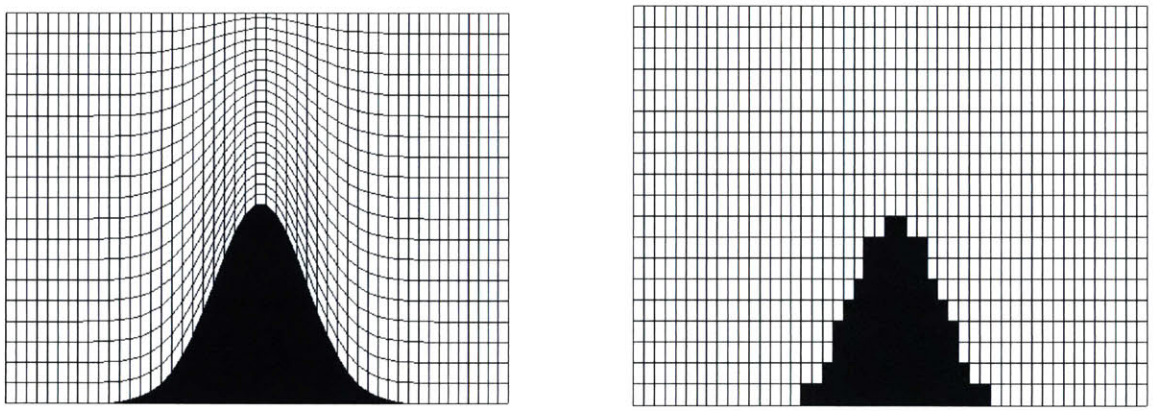


Figure 3-1: The spacing between vertical cell walls is uniformly Δx, and the length of the vertical cell walls is uniformly $H(x)\Delta\sigma$ in the vertical at any given horizontal coordinate.



(a) (b)

Figure 3-2: A σ-coordinate mesh (a) and z-coordinate (b) mesh with the same number of cells in the vertical and horizontal.

3.2 The σ-coordinate Scheme

A σ-coordinate scheme often removes complexity from the physical grid. For constant values of Δx and $\Delta\sigma$, where $\Delta\sigma = \sigma_i - \sigma_{i+1}$, the computational grid in $\tilde{x}, \sigma$ space is a structured, rectangular grid with uniform rectangular cells Δx wide and $\Delta\sigma$ high. Unfortunately, if we were to evaluate the terms in the momentum and tracer-convection equations using simple rectangular ($\Delta x \times H(x)\Delta\sigma$) cells, we would need to couple the horizontal and vertical momentum conservation equations. Instead, we use cells like those in Figure 3-2a, and keep the zonal and azimuthal momentum equations decoupled. This sacrifices some of the simplicity offered by σ-coordinates in

exchange for the computational efficiency of decoupled momentum equations. We first present a detailed discussion of the Laplacian and gradient operators evaluated in a cell bordered by other interior cells or Dirichlet boundaries, and demonstrate that the scheme in the limit of a Cartesian grid reduces to the central difference scheme used in the 2DFV z-coordinate code. Then we describe modifications to each algorithm for cells that impinge on Neumann or open boundaries. Finally, we describe the advection scheme. The time-marching scheme is a second-order backward difference scheme, which has been preserved from the 2DFV z-coordinate code.

3.2.1 The Laplacian Operator

The Laplacian operator is used in the diffusion terms of the horizontal momentum, vertical momentum, and tracer-convection equations. The operator is also used to calculate the Laplacian of the pressure-correction (q) as part of the pressure-correction projection method. We assume the horizontal eddy viscosity (ν_H), vertical eddy viscosity (ν_V), and eddy diffusivity (κ) coefficients are constants. The Laplacian operator maps values to and from the same grid in our C-grid mesh; we will see in the next section that this is not the case for gradient operations. We construct this operator and perform an LU factorization at the beginning of the simulation before time-marching, in preparation for the inversion that occurs at each timestep. If we wish to use the same numerical operator for the Laplacian of the velocity fields and the Laplacian of the pressure-correction field, we need to construct it such that the definitions imposed by the pressure-correction method are observed. See Appendix A for an explanation of the projection method and the way we have guaranteed discrete consistency.

The Importance of an Implicit Scheme

Diffusion is the stiff term in our equations of motion, and while many σ-coordinate and unstructured frameworks compute the diffusion term explicitly (Fringer et al., 2006; Marshall et al., 1997; Chen et al., 2003b), we maintain a fully implicit scheme to avoid a stringent Courant-Friedrichs-Lewy (CFL) condition (Courant et al., 1967). Von Neumann stability analysis can be used to show an explicit diffusion operator would require a CFL condition of

$$\Delta t \leq \frac{\Delta y^2}{2\nu_V},$$

(the limiting direction for this problem is the vertical). This restricts the timestep in the TN test-case to $\Delta t \leq \frac{31.3[\mathrm{m}^2]}{2(0.1[\mathrm{ms}^{-1}])^2} = 4.91(10^3)$s. The CFL condition imposed by advection is

$$\Delta t \leq \frac{\Delta y}{V_{max}}$$

where V_{max} is the largest velocity the simulation must capture. With a barotropic tide of 2 cm/s, the advection-imposed CFL condition is conservatively $\Delta t \leq \frac{31.3[\mathrm{m}]}{0.004[\mathrm{m}^2\mathrm{s}^{-1}]} = 7.83(10)^3$s, 60% larger than the condition imposed by diffusion. A fully implicit diffusion operator permits a larger timestep and substantially improves computational cost.

The Algorithm

The finite volume formulation, which comes from control volume theory, casts divergence and gradient operations as surface integrals around individual finite volumes by way of the divergence theorem. This design is a principle advantage of the finite volume method for computational fluid dynamics; by calculating surface fluxes between cells, local and global conservation of mass, momentum, and often energy, is guaranteed in a discrete sense. The construction of the method also means state-variables are considered in a cell-averaged sense.

We first write the Laplacian as the divergence of a gradient, then take averages over individual cell volumes and, finally, use the divergence theorem to arrive at a surface integral around each cell. Once we know the gradient of the state variable at the center of each cell face, as shown in Figure 3-8, we can take the divergence of those gradients to arrive at the Laplacian.

$$(\nabla \cdot \nabla\phi)_{\text{Volume Averaged}} = \frac{1}{V} \iiint_V \nabla \cdot \nabla\phi dV \tag{3.2}$$

$$(\nabla \cdot \nabla\phi)_{\text{Volume Averaged}} = \frac{1}{V} \oint_S \nabla\phi \cdot \boldsymbol{n}\ dS \tag{3.3}$$

$$(\nabla \cdot \nabla\phi)_{\text{Volume Averaged}} = \frac{1}{V} \oint_S \left\{ \frac{1}{V_{\text{satellite}}} \iiint_{V_{\text{satellite}}} \nabla\phi\ \ dV_{\text{satellite}} \right\} \cdot \boldsymbol{n}\ dS \tag{3.4}$$

$$(\nabla \cdot \nabla\phi)_{\text{Volume Averaged}} = \frac{1}{V} \oint_S \left\{ \frac{1}{V_{\text{satellite}}} \oint_{S_{\text{satellite}}} \phi\, \boldsymbol{n}_{\text{satellite}}\ \ dS_{\text{satellite}} \right\} \cdot \boldsymbol{n}\ dS \tag{3.5}$$

The first step is to calculate the gradients, the terms inside the braces of Equation 3.5.

We focus on the North gradient first. The gradient needs to be evaluated at the center of the top cell face $(x_j + \frac{\Delta x}{2}\ ,\ H(x_{i,j+\frac{1}{2}})\ \sigma_i)$. The closed region over which we will evaluate the gradient, identified by the dashed line in Figure 3-4, has the same width as the center cell, and extends from the midpoint of the cell above to the midpoint of the center cell. For clarity, we will enumerate cell faces as shown in Figure 3-3.

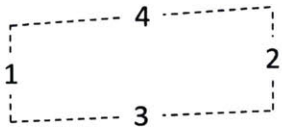


Figure 3-3: The left, right, bottom, and top faces of cells are the first, second, third, and fourth faces, respectively.

Notice we need to interpolate ϕ to faces 1 and 2 of the North satellite cell, and we will also need to interpolate ϕ for the other satellite cells to $\phi_{i+\frac{1}{2},j-\frac{1}{2}}$ and $\phi_{i+\frac{1}{2},j+\frac{1}{2}}$. We use a first-order, bi-directional interpolation.

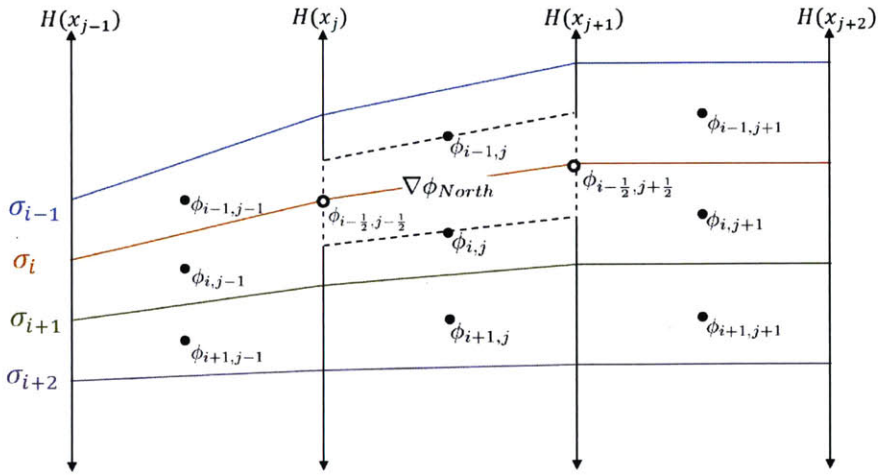


Figure 3-4: The North satellite cell.

$$\phi_{i-\frac{1}{2},j-\frac{1}{2}} = \frac{1}{4}\left(\phi_{i-1,j-1} + \phi_{i-1,j} + \phi_{i,j-1} + \phi_{i,j}\right) \quad \phi_{i-\frac{1}{2},j+\frac{1}{2}} = \frac{1}{4}\left(\phi_{i-1,j} + \phi_{i-1,j+1} + \phi_{i,j} + \phi_{i,j+1}\right)$$
$$\phi_{i+\frac{1}{2},j-\frac{1}{2}} = \frac{1}{4}\left(\phi_{i,j-1} + \phi_{i,j} + \phi_{i+1,j-1} + \phi_{i+1,j}\right) \quad \phi_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{1}{4}\left(\phi_{i,j} + \phi_{i,j+1} + \phi_{i+1,j} + \phi_{i+1,j+1}\right) \tag{3.6}$$

Similarly, we interpolate values of σ with a first-order scheme, $\sigma_{i+\frac{1}{2}} = \frac{\sigma_i + \sigma_{i+1}}{2}$. The normal vectors corresponding to the four faces of the North satellite cell can be easily calculated.

$$\boldsymbol{n_{N,1}} = [-1, 0] \tag{3.7}$$

$$\boldsymbol{n_{N,2}} = [1, 0] \tag{3.8}$$

$$\boldsymbol{n_{N,3}} = \left[\frac{H(x_{j+1})\sigma_{i-1/2} - H(x_j)\sigma_{i-1/2} \ , \ x_j - (x_j + \Delta x)}{\sqrt{(H(x_{j+1})\sigma_{i-1/2} - H(x_j)\sigma_{i-1/2})^2 + \Delta x^2}} \right] \tag{3.9}$$

$$\boldsymbol{n_{N,4}} = \left[\frac{H(x_j)\sigma_{i+1/2} - H(x_{j+1})\sigma_{i+1/2} \ , \ -(x_j - (x_j + \Delta x))}{\sqrt{(H(x_{j+1})\sigma_{i+1/2} - H(x_j)\sigma_{i+1/2})^2 + \Delta x^2}} \right] \tag{3.10}$$

The North gradient of ϕ is calculated from the surface integral around North cell as

$$\begin{aligned} \nabla\phi_{North} = \frac{1}{V_{North}} \Bigg(& \int_{(x_j, H(x_j)\sigma_{i-\frac{1}{2}})}^{(x_j, H(x_j)\sigma_{i+\frac{1}{2}})} \phi_{i-\frac{1}{2}, j-\frac{1}{2}} \boldsymbol{n_{N,1}} dS_1^{North} + \int_{(x_j, H(x_j)\sigma_{i+\frac{1}{2}})}^{(x_j+\Delta x, H(x_{j+1})\sigma_{i+\frac{1}{2}})} \phi_{i,j} \boldsymbol{n_{N,3}} dS_3^{North} + \ldots \\ & \int_{(x_j+\Delta x, H(x_{j+1})\sigma_{i+\frac{1}{2}})}^{(x_j+\Delta x, H(x_{j+1})\sigma_{i-\frac{1}{2}})} \phi_{i-\frac{1}{2}, j+\frac{1}{2}} \boldsymbol{n_{N,2}} dS_2^{North} + \int_{(x_j, H(x_j)\sigma_{i-\frac{1}{2}})}^{(x_j+\Delta x, H(x_{j+1})\sigma_{i-\frac{1}{2}})} \phi_{i-1,j} \boldsymbol{n_{N,4}} dS_4^{North} \Bigg) \end{aligned} \tag{3.11}$$

The right hand side of Equation 3.11, written in terms of coefficients of the six values of ϕ used to compute $\nabla\phi_{North}$, is

$$\begin{aligned} \nabla\phi_{North} = \frac{1}{V_{North}} \Bigg(& \phi_{i-\frac{1}{2}, j-\frac{1}{2}} [-1, 0] H(x_j)(\sigma_{i-\frac{1}{2}} - \sigma_{i+\frac{1}{2}}) + \phi_{i-\frac{1}{2}, j+\frac{1}{2}} [1, 0] H(x_{j+1})(\sigma_{i-\frac{1}{2}} - \sigma_{i+\frac{1}{2}}) + \ldots \\ & \phi_{i,j} \left[\frac{(H(x_{j+1}) - H(x_j))\sigma_{i+\frac{1}{2}}, \ -\Delta x}{S_3^{North}} \right] S_3^{North} + \phi_{i-1,j} \left[\frac{(H(x_j) - H(x_{j+1}))\sigma_{i-\frac{1}{2}}, \ \Delta x}{S_4^{North}} \right] S_4^{North} \Bigg) \end{aligned} \tag{3.12}$$

We can group the right hand side of 3.12 according to the ϕ value so that the gradient is the sum of six vectors, one for each ϕ in the gradient stencil.

$$\begin{aligned} \nabla\phi_{North} = \frac{1}{V_{North}} \Big(& \phi_{i,j}[a_{C_N}, b_{C_N}] + \phi_{i-1,j}[a_{N_N}, b_{N_N}] + \phi_{i-1,j-1}[a_{NW_N}, b_{NW_N}] + \\ & \phi_{i-1,j+1}[a_{NE_N}, b_{NE_N}] + \phi_{i,j-1}[a_{E_N}, b_{E_N}] + \phi_{i,j+1}[a_{W_N}, b_{W_N}] \Big) \end{aligned} \tag{3.13}$$

The coefficients in Equation 3.13, which are constant for the entire simulation since the grid does

not evolve in time, are:

$$
\begin{aligned}
a_{C_N} &= (H(x_{j+1}) - H(x_j))\sigma_{i+\frac{1}{2}} + \frac{1}{4}H(x_j)(\sigma_{i+\frac{1}{2}} - \sigma_{i-\frac{1}{2}}) + \frac{1}{4}H(x_{j+1})(\sigma_{i-\frac{1}{2}} - \sigma_{i+\frac{1}{2}}) \\
b_{C_N} &= -\Delta x \\
a_{N_N} &= (H(x_j) - H(x_{j+1}))\sigma_{i-\frac{1}{2}} + \frac{1}{4}H(x_j)(\sigma_{i+\frac{1}{2}} - \sigma_{i-\frac{1}{2}}) + \frac{1}{4}H(x_{j+1})(\sigma_{i-\frac{1}{2}} - \sigma_{i+\frac{1}{2}}) \\
b_{N_N} &= \Delta x \\
a_{NW_N} &= \frac{1}{4}H(x_j)(\sigma_{i+\frac{1}{2}} - \sigma_{i-\frac{1}{2}}) \\
b_{NW_N} &= 0 \\
a_{NE_N} &= \frac{1}{4}H(x_{j+1})(\sigma_{i-\frac{1}{2}} - \sigma_{i+\frac{1}{2}}) \\
b_{NE_N} &= 0 \\
a_{W_N} &= \frac{1}{4}H(x_j)(\sigma_{i+\frac{1}{2}} - \sigma_{i-\frac{1}{2}}) \\
b_{W_N} &= 0 \\
a_{E_N} &= \frac{1}{4}H(x_{j+1})(\sigma_{i-\frac{1}{2}} - \sigma_{i+\frac{1}{2}}) \\
b_{E_N} &= 0
\end{aligned}
$$

Now we repeat this procedure for the South (Figure 3-5), East (Figure 3-6), and West (Figure 3-7) satellite cells until we have the four gradients in Figure 3-8.

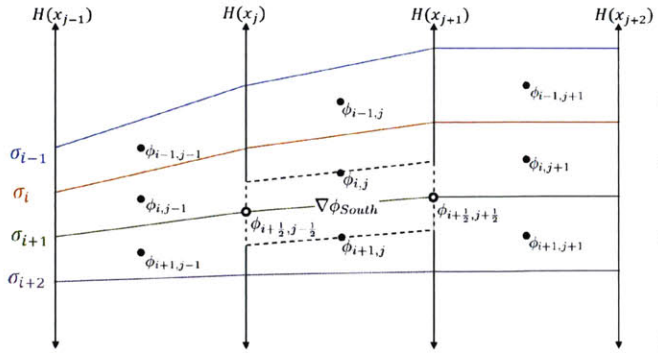


Figure 3-5: The South satellite cell.

$$\boldsymbol{n_{S,1}} = [-1, 0]$$

$$\boldsymbol{n_{S,2}} = [1, 0]$$

$$\boldsymbol{n_{S,3}} = \left[\frac{(H(x_{j+1}) - H(x_j))\sigma_{i+\frac{3}{2}}, x_j - (x_j + \Delta x)}{\sqrt{(H(x_{j+1})\sigma_{i+\frac{3}{2}} - H(x_j)\sigma_{i+\frac{3}{2}})^2 + \Delta x^2}}\right]$$

$$\boldsymbol{n_{S,4}} = \left[\frac{(H(x_j) - H(x_{j+1}))\sigma_{i+\frac{1}{2}}, -(x_j - (x_j + \Delta x))}{\sqrt{(H(x_{j+\frac{1}{2}})\sigma_{i+\frac{1}{2}} - H(x_j)\sigma_{i+\frac{1}{2}})^2 + \Delta x^2}}\right]$$

$$\begin{aligned}\nabla\phi_{South} =& \frac{1}{V_{South}}\left(\int_{(x_j, H(x_j)\sigma_{i+\frac{1}{2}})}^{(x_j, H(x_j)\sigma_{i+\frac{3}{2}})} \phi_{i+\frac{1}{2},j-\frac{1}{2}}\boldsymbol{n_{S,1}}dS_1^{South} + \int_{(x_j, H(x_j)\sigma_{i+\frac{3}{2}})}^{(x_j+\Delta x, H(x_{j+1})\sigma_{i+\frac{3}{2}})} \phi_{i+1,j}\boldsymbol{n_{S,3}}dS_3^{South} + \ldots \right. \\ & \left. \int_{(x_j+\Delta x, H(x_{j+1})\sigma_{i+\frac{3}{2}})}^{(x_j+\Delta x, H(x_{j+1})\sigma_{i+\frac{1}{2}})} \phi_{i+\frac{1}{2},j+\frac{1}{2}}\boldsymbol{n_{S,2}}dS_2^{South} + \int_{(x_j, H(x_j)\sigma_{i+\frac{1}{2}})}^{(x_j+\Delta x, H(x_{j+1})\sigma_{i+\frac{1}{2}})} \phi_{i,j}\boldsymbol{n_{S,4}}dS_4^{South}\right)\end{aligned} \tag{3.14}$$

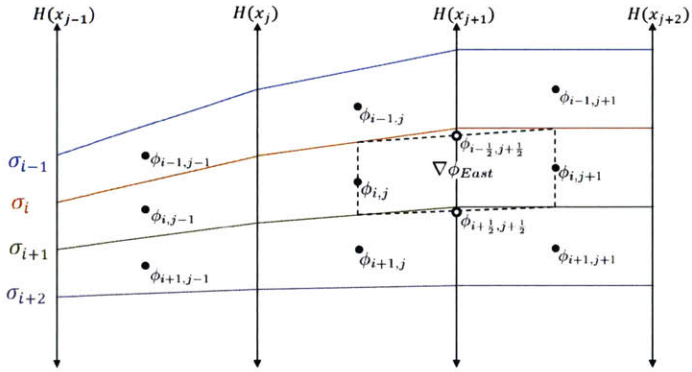


Figure 3-6: The East satellite cell.

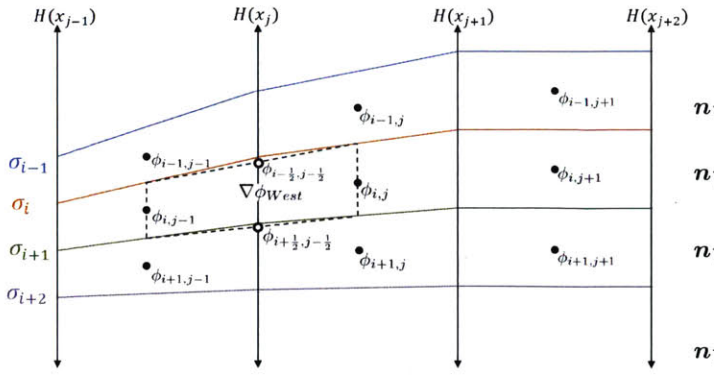
$$\boldsymbol{n_{E,1}} = [-1, 0]$$

$$\boldsymbol{n_{E,2}} = [1, 0]$$

$$\boldsymbol{n_{E,3}} = \left[\frac{(H(x_{j+\frac{3}{2}}) - H(x_{j+\frac{1}{2}}))\sigma_{i+1}, x_j - (x_j + \Delta x)}{\sqrt{(H(x_{j+\frac{1}{2}})\sigma_{i+1} - H(x_{j+\frac{3}{2}})\sigma_{i+1})^2 + \Delta x^2}}\right]$$

$$\boldsymbol{n_{E,4}} = \left[\frac{(H(x_{j+\frac{1}{2}}) - H(x_{j+\frac{3}{2}}))\sigma_i, -(x_j - (x_j + \Delta x))}{\sqrt{(H(x_{j+\frac{1}{2}})\sigma_i - H(x_{j+\frac{3}{2}})\sigma_i)^2 + \Delta x^2}}\right]$$

$$\begin{aligned}\nabla\phi_{East} =& \frac{1}{V_{East}}\left(\int_{(x_j+\frac{\Delta x}{2}, H(x_{j+\frac{1}{2}})\sigma_i)}^{(x_j+\frac{\Delta x}{2}, H(x_{j+\frac{1}{2}})\sigma_{i+1})} \phi_{i,j}\boldsymbol{n_{E,1}}dS_1^{East} + \int_{(x_j+\frac{\Delta x}{2}, H(x_{j+\frac{1}{2}})\sigma_{i+1})}^{(x_j+3\frac{\Delta x}{2}, H(x_{j+\frac{3}{2}})\sigma_{i+1})} \phi_{i+\frac{1}{2},j+\frac{1}{2}}\boldsymbol{n_{E,3}}dS_3^{East} + \ldots \right. \\ & \left. \int_{(x_j+3\frac{\Delta x}{2}, H(x_{j+\frac{3}{2}})\sigma_{i+1})}^{(x_j+3\frac{\Delta x}{2}, H(x_{j+\frac{3}{2}})\sigma_i)} \phi_{i,j+1}\boldsymbol{n_{E,2}}dS_2^{East} + \int_{(x_j+\frac{\Delta x}{2}, H(x_{j+\frac{1}{2}})\sigma_i)}^{(x_j+3\frac{\Delta x}{2}, H(x_{j+\frac{3}{2}})\sigma_i)} \phi_{i-\frac{1}{2},j+\frac{1}{2}}\boldsymbol{n_{E,4}}dS_4^{East}\right)\end{aligned} \tag{3.15}$$



$$\boldsymbol{n_{W,1}} = [-1, 0]$$

$$\boldsymbol{n_{W,2}} = [1, 0]$$

$$\boldsymbol{n_{W,3}} = \left[\frac{(H(x_{j+\frac{1}{2}}) - H(x_{j-\frac{1}{2}}))\sigma_{i+1}, x_j - (x_j + \Delta x)}{\sqrt{(H(x_{j+\frac{1}{2}})\sigma_{i+1} - H(x_{j-\frac{1}{2}})\sigma_{i+1})^2 + \Delta x^2}}\right]$$

$$\boldsymbol{n_{W,4}} = \left[\frac{(H(x_{j-\frac{1}{2}}) - H(x_{j+\frac{1}{2}}))\sigma_i, -(x_j - (x_j + \Delta x))}{\sqrt{(H(x_{j+\frac{1}{2}})\sigma_i - H(x_{j-\frac{1}{2}})\sigma_i)^2 + \Delta x^2}}\right]$$

Figure 3-7: The West satellite cell.

$$\nabla\phi_{West} = \frac{1}{V_{West}}\left(\int_{(x_j-\frac{\Delta x}{2},H(x_{j-\frac{1}{2}})\sigma_{i+1})}^{(x_j-\frac{\Delta x}{2},H(x_{j-\frac{1}{2}})\sigma_i)} \phi_{i,j-1}\boldsymbol{n_{W,1}}dS_1^{West} + \int_{(x_j-\frac{\Delta x}{2},H(x_{j-\frac{1}{2}})\sigma_{i+1})}^{(x_j+\frac{\Delta x}{2},H(x_{j+\frac{1}{2}})\sigma_{i+1})} \phi_{i-\frac{1}{2},j+\frac{1}{2}}\boldsymbol{n_{W,3}}dS_3^{West} + \ldots \right.$$
$$\left. \int_{(x_j+\frac{\Delta x}{2},H(x_{j+\frac{1}{2}})\sigma_{i+1})}^{(x_j+\frac{\Delta x}{2},H(x_{j+\frac{1}{2}})\sigma_i)} \phi_{i,j}\boldsymbol{n_{W,2}}dS_2^{West} + \int_{(x_j-\frac{\Delta x}{2},H(x_{j-\frac{1}{2}})\sigma_i)}^{(x_j+\frac{\Delta x}{2},H(x_{j+\frac{1}{2}})\sigma_i)} \phi_{i-\frac{1}{2},j-\frac{1}{2}}\boldsymbol{n_{W,4}}dS_4^{West}\right) \tag{3.16}$$

Since the East and West satellite cells do not precisely outline cells on the scalar grid (i.e. the East

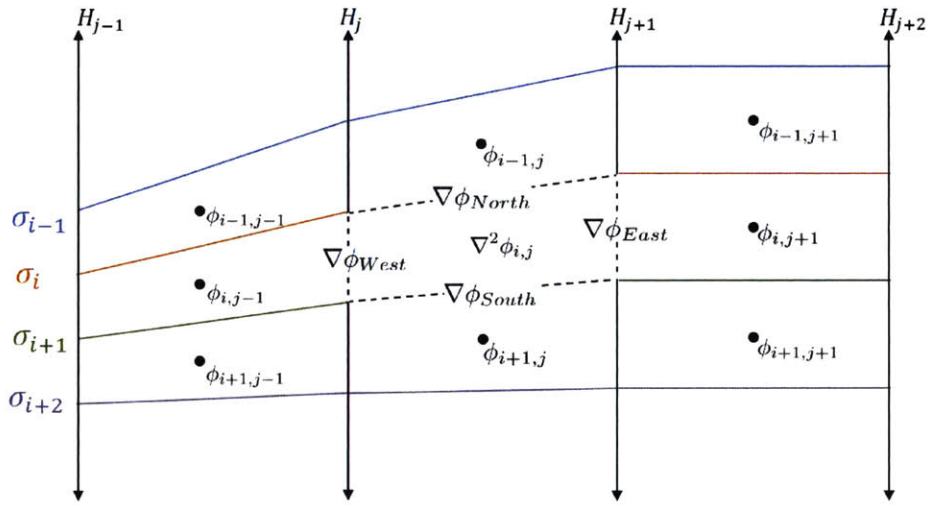


Figure 3-8: The Laplacian numerical stencil uses nine values of ϕ. First the four gradients at the center of each cell face are computed, then a divergence of these gradients gives us the Laplacian at the center.

and West satellite cells do not have a "kink" in the middle), we cannot guarantee conservation of mass and momentum as with a z-coordinate mesh. However, for small changes in slope we do not

expect a significant error (Ferziger and Peric, 2002). Next we calculate the Laplacian by taking the divergence of the four gradients

$$(\nabla \cdot \nabla\phi)_{\text{Volume Averaged}} = \int_{x_j, H_j\sigma_{i+1}}^{x_j, H_j\sigma_i} \nabla\phi_{West} \cdot \boldsymbol{n_1} dS_1 + \int_{x_j, H_j\sigma_{i+1}}^{x_j+\Delta x, H_{j+1}\sigma_{i+1}} \nabla\phi_{South} \cdot \boldsymbol{n_3} dS_3 + \ldots$$
$$\int_{x_j+\Delta x, H_{j+1}\sigma_{i+1}}^{x_j+\Delta x, H_{j+1}\sigma_i} \nabla\phi_{East} \cdot \boldsymbol{n_2} dS_2 + \int_{x_j+\Delta x, H_j\sigma_i}^{x_j, H_{j+1}\sigma_i} \nabla\phi_{North} \cdot \boldsymbol{n_4} dS_4. \tag{3.17}$$

The four normal vectors of the center cell are

$$\boldsymbol{n_1} = [-1, 0] \tag{3.18}$$

$$\boldsymbol{n_2} = [1, 0] \tag{3.19}$$

$$\boldsymbol{n_3} = \left[\frac{H_{j+1}\sigma_i - H_j\sigma_i \ , \ x_j - (x_j + \Delta x)}{\sqrt{(H_{j+1}\sigma_i - H_j\sigma_i)^2 + \Delta x^2}} \right] \tag{3.20}$$

$$\boldsymbol{n_4} = \left[\frac{H_j\sigma_{i+1} - H_{j+1}\sigma_{i+1} \ , \ -(x_j - (x_j + \Delta x))}{\sqrt{(H_{j+1}\sigma_{i+1} - H_j\sigma_{i+1})^2 + \Delta x^2}} \right]. \tag{3.21}$$

We rearrange Equation 3.17 into coefficients for each of the nine values of the state variable as in Equation 3.13.

$$\begin{aligned}
\nabla \cdot \nabla\phi = &\phi_{i,j} \left[\frac{a_{C_N}}{V_{North}} + \frac{a_{C_S}}{V_{South}} + \frac{a_{C_E}}{V_{East}} + \frac{a_{C_W}}{V_{West}}, \frac{b_{C_N}}{V_{North}} + \frac{b_{C_S}}{V_{South}} \right] \cdot \left[\frac{a_C}{V}, \frac{b_C}{V} \right] + \ldots \\
&\phi_{i-1,j} \left[\frac{a_{N_N}}{V_{North}} + \frac{a_{N_E}}{V_{East}} + \frac{a_{N_W}}{V_{West}}, \frac{b_{N_N}}{V_{North}} + \frac{b_{N_S}}{V_{South}} \right] \cdot \left[\frac{a_N}{V}, \frac{b_N}{V} \right] + \ldots \\
&\phi_{i+1,j} \left[\frac{a_{S_S}}{V_{South}} + \frac{a_{S_E}}{V_{East}} + \frac{a_{S_W}}{V_{West}}, \frac{b_{S_N}}{V_{North}} + \frac{b_{S_S}}{V_{South}} \right] \cdot \left[\frac{a_S}{V}, \frac{b_S}{V} \right] + \ldots \\
&\phi_{i,j-1} \left[\frac{a_{W_N}}{V_{North}} + \frac{a_{W_S}}{V_{South}} + \frac{a_{W_W}}{V_{West}}, \frac{b_{W_N}}{V_{North}} + \frac{b_{W_S}}{V_{South}} \right] \cdot \left[\frac{a_W}{V}, 0 \right] + \ldots \\
&\phi_{i,j+1} \left[\frac{a_{E_N}}{V_{North}} + \frac{a_{E_S}}{V_{South}} + \frac{a_{E_E}}{V_{East}}, \frac{b_{E_N}}{V_{North}} + \frac{b_{E_S}}{V_{South}} \right] \cdot \left[\frac{a_E}{V}, 0 \right] + \ldots \\
&\phi_{i-1,j-1} \left[\frac{a_{NW_N}}{V_{North}} + \frac{a_{NW_W}}{V_{West}}, \frac{b_{NW_N}}{V_{North}} \right] \cdot \left[\frac{a_{NW}}{V}, 0 \right] + \ldots \\
&\phi_{i+1,j-1} \left[\frac{a_{SW_S}}{V_{South}} + \frac{a_{SW_W}}{V_{West}}, \frac{b_{SW_S}}{V_{South}} \right] \cdot \left[\frac{a_{SW}}{V}, 0 \right] + \ldots \\
&\phi_{i+1,j+1} \left[\frac{a_{SE_S}}{V_{South}} + \frac{a_{SE_E}}{V_{East}}, \frac{b_{SE_S}}{V_{South}} \right] \cdot \left[\frac{a_{SE}}{V}, 0 \right] + \ldots \\
&\phi_{i-1,j+1} \left[\frac{a_{NE_N}}{V_{North}} + \frac{a_{NE_E}}{V_{East}}, \frac{b_{E_N}}{V_{North}} \right] \cdot \left[\frac{a_{NE}}{V}, 0 \right]
\end{aligned} \tag{3.22}$$

These values are indexed into the Laplacian operator, which is subsequently LU factored to expedite inversion.

The Algorithm on a z-coordinate Grid

We can evaluate the truncation error of the Laplacian operator by Taylor expanding each of the nine values in the stencil about the center value ($\phi_{i,j}$), and find that for arbitrary $H(x)$ the truncation error is first-order, and for constant domain height,$H(x) = H_0$ the truncation error is second-order. This operator will reduce to the second-order central difference scheme used in the 2DFV z-coordinate code on a z-coordinate mesh. On a z-coordinate mesh, the domain will have constant height, $H(x) = H_0$, and all of the normal vectors will reduce to either the standard basis vectors or the negative of the standard basis. This implies the coefficients corresponding to the four "corners" of the stencil ($\phi_{i-1,j-1}, \phi_{i+1,j-1}, \phi_{i-1,j+1}, \phi_{i+1,j+1}$) vanish. The four gradients for a z-coordinate grid are then

$$\nabla\phi_{North} = [0, \frac{\phi_{i-1,j} - \phi_{i,j}}{\Delta x \Delta y}], \quad \nabla\phi_{South} = [0, \frac{\phi_{i,j} - \phi_{i+1,j}}{\Delta x \Delta y}]$$
$$\nabla\phi_{East} = [\frac{\phi_{i,j+1} - \phi_{i,j}}{\Delta x \Delta y}, 0], \quad \nabla\phi_{West} = [\frac{\phi_{i,j} - \phi_{i,j-1}}{\Delta x \Delta y}, 0].$$

We can take the divergence of these gradients to arrive at the Laplacian.

$$\nabla \cdot \nabla\phi = \frac{0 - 0 + (\frac{\phi_{i-1,j} - \phi_{i,j}}{\Delta x \Delta y}) - (\frac{\phi_{i,j} - \phi_{i+1,j}}{\Delta x \Delta y})}{\Delta x \Delta y} + \frac{(\frac{\phi_{i,j+1} - \phi_{i,j}}{\Delta x \Delta y}) - (\frac{\phi_{i,j} - \phi_{i,j-1}}{\Delta x \Delta y}) + 0 - 0}{\Delta x \Delta y} \tag{3.23}$$
$$= \frac{\phi_{i-1,j} + \phi_{i+1,j} - 4\phi_{i,j} + \phi_{i,j+1} + \phi_{i,j-1}}{\Delta x^2 \Delta y^2} \tag{3.24}$$

and find the second-order central difference scheme is recovered exactly.

3.2.2 The Gradient Operator

Calculating gradients is required for several terms each timestep; the gradient of pressure appears in the momentum equations, and the gradient of velocity appears in the pressure-correction step. Unlike the Laplacian, these operators must map between grids in our C-grid mesh. The two pressure gradient operators $\left(\frac{\partial p}{\partial x}, \frac{\partial p}{\partial y}\right)$ map values from the scalar grid, to the u and v grids. The two velocity gradient operators $\left(\frac{\partial u}{\partial x}, \frac{\partial v}{\partial y}\right)$ map values from the u and v grids, to the scalar grid.

Figure 3-9 shows the locations of the u and v values on the σ-cooridnate grid. Another difference

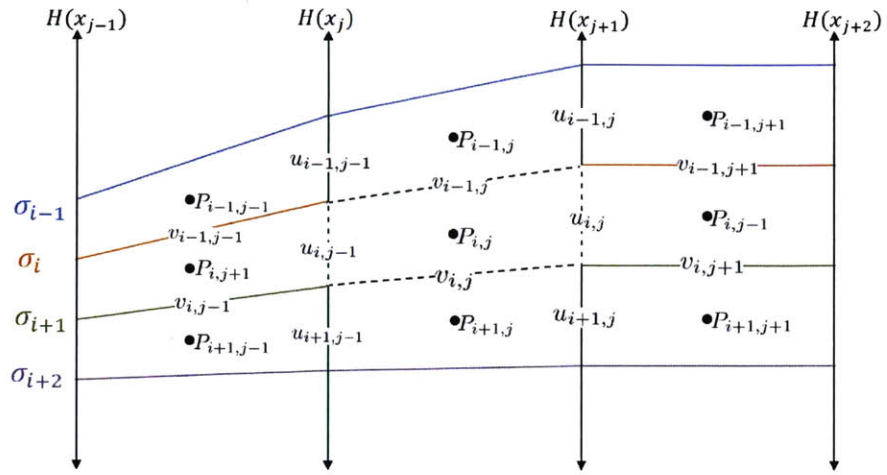


Figure 3-9: The σ-coordinate grid maintains a C-grid configuration for the locations of the u and v velocities.

between the Laplacian and the gradient operators is that the gradient is always part of an explicit calculation. We can construct the four gradient operators once at the beginning of the simulation, and evaluate gradients with matrix-vector multiplication at each timestep.

The Algorithm

Since the Laplacian operator required the computation of gradients, we will recycle part of that algorithm for the gradient operators. The interpolated values of u, v, and p, just as with the Laplacian in Equation 3.6, are computed as the average of the four neighboring values. We first consider the horizontal pressure gradient. The dashed line in Figure 3-10, denotes the closed surface over which we will evaluate the horizontal pressure gradient.

$$\nabla p_{i,j+\frac{1}{2}} = \frac{1}{V}\left(\int_{(x_j+\frac{\Delta x}{2},H(x_{j+\frac{1}{2}})\sigma_{i+1})}^{(x_j+\frac{\Delta x}{2},H(x_{j+\frac{1}{2}})\sigma_i)} p_{i,j}\ \boldsymbol{n_1}\ dS_1 + \int_{(x_j+\frac{\Delta x}{2},H(x_{j+\frac{1}{2}})\sigma_{i+1})}^{(x_j+3\frac{\Delta x}{2},H(x_{j+\frac{3}{2}})\sigma_{i+1})} p_{i+\frac{1}{2},j+\frac{1}{2}}\ \boldsymbol{n_3}\ dS_3 + \ldots \right.$$
$$\left. \int_{(x_j+3\frac{\Delta x}{2},H(x_{j+\frac{3}{2}})\sigma_{i+1})}^{(x_j+3\frac{\Delta x}{2},H(x_{j+\frac{3}{2}})\sigma_i)} p_{i,j+1}\ \boldsymbol{n_2}\ dS_2 + \int_{(x_j+\frac{\Delta x}{2},H(x_{j+\frac{1}{2}})\sigma_i)}^{(x_j+3\frac{\Delta x}{2},H(x_{j+\frac{3}{2}})\sigma_i)} p_{i-\frac{1}{2},j+\frac{1}{2}}\ \boldsymbol{n_4}\ dS_4 \right) \tag{3.25}$$

Notice Equation 3.25 is exactly the equation for $\nabla\phi_{East}$ in Equation 3.15. Similarly, the vertical pressure gradient on the v grid in Equation 3.26 is exactly $\nabla\phi_{North}$ from Equation 3.12. The

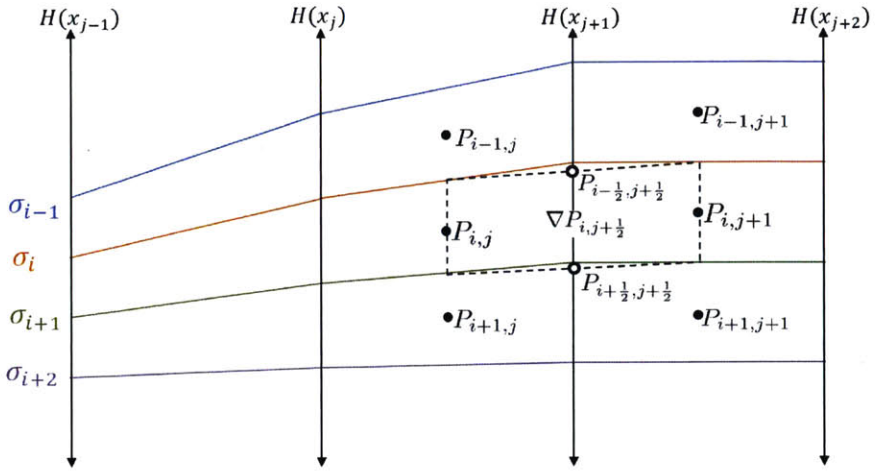


Figure 3-10: We integrate over the surface identified by the dashed line to find $\frac{\partial p}{\partial x}$. We can then extract the horizontal gradient $\frac{\partial p}{\partial x} = \nabla p \cdot [1, 0]$.

dashed line in Figure 3-11, denotes the closed surface over which we will evaluate the vertical pressure gradient.

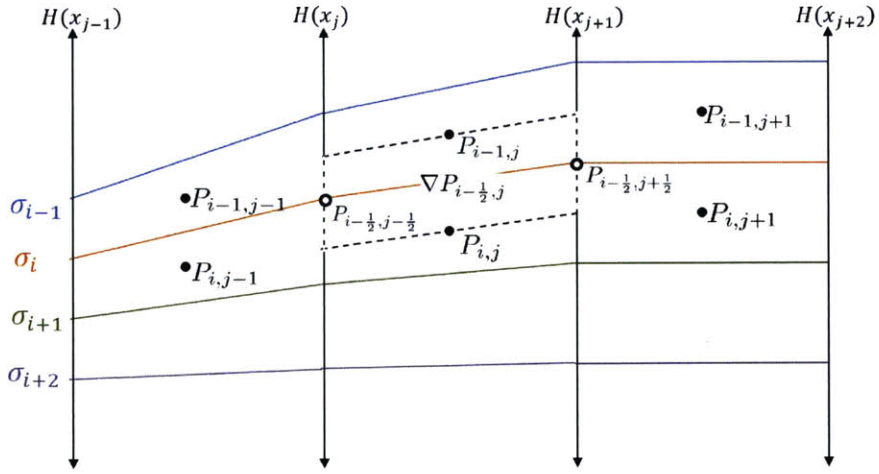


Figure 3-11: We integrate over the surface identified by the dashed line to find $\frac{\partial p}{\partial y}$. We can then extract the vertical gradient $\frac{\partial p}{\partial y} = \nabla p \cdot [0, 1]$.

$$\nabla p_{i-\frac{1}{2},j} = \frac{1}{V}\Bigg(p_{i-\frac{1}{2},j-\frac{1}{2}}[-1,0]H(x_j)(\sigma_{i-\frac{1}{2}} - \sigma_{i+\frac{1}{2}}) + p_{i-\frac{1}{2},j+\frac{1}{2}}[1,0]H(x_{j+1})(\sigma_{i-\frac{1}{2}} - \sigma_{i+\frac{1}{2}}) + \ldots$$
$$p_{i,j}\left[\frac{(H(x_{j+1}) - H(x_j))\sigma_{i+\frac{1}{2}},\ -\Delta x}{S_3}\right] S_3 + p_{i-1,j}\left[\frac{(H(x_j) - H(x_{j+1}))\sigma_{i-\frac{1}{2}},\ \Delta x}{S_4}\right] S_4\Bigg) \tag{3.26}$$

Finally, the velocity gradient operators are identical to the surface integral around the center cell in Equation 3.17, and we see why this is crucial in Appendix A. The dashed line in Figure 3-12a (Figure 3-12b), denotes the closed surface over which we will evaluate the horizontal (vertical) velocity gradient.

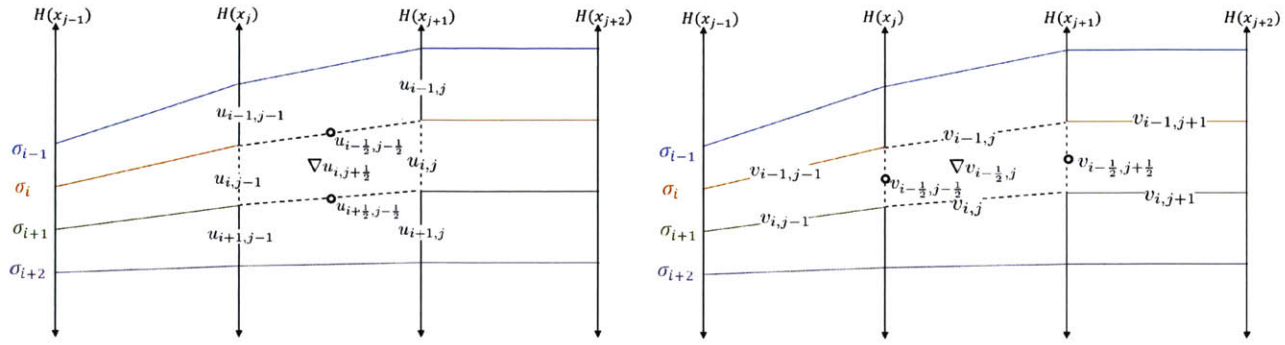


Figure 3-12: We integrate over the surface identified by the dashed line to find the gradients of velocity, and as with the pressure gradients, we extract the relevant components ($\frac{\partial u}{\partial x} = \nabla u \cdot [1,0]$ $\frac{\partial v}{\partial y} = \nabla v \cdot [0,1]$).

$$\nabla u_{i,j+\frac{1}{2}} = \int_{x_j,H_j\sigma_{i+1}}^{x_j,H_j\sigma_i} u_{i,j-1} \cdot \boldsymbol{n_1} dS_1 + \int_{x_j,H_j\sigma_{i+1}}^{x_j+\Delta x,H_{j+1}\sigma_{i+1}} u_{i+\frac{1}{2},j-\frac{1}{2}} \cdot \boldsymbol{n_3} dS_3 + ... \\ \int_{x_j+\Delta x,H_{j+1}\sigma_{i+1}}^{x_j+\Delta x,H_{j+1}\sigma_i} u_{i,j} \cdot \boldsymbol{n_2} dS_2 + \int_{x_j+\Delta x,H_j\sigma_i}^{x_j,H_{j+1}\sigma_i} u_{i-\frac{1}{2},j-\frac{1}{2}} \cdot \boldsymbol{n_4} dS_4 \tag{3.27}$$

$$\nabla v_{i-\frac{1}{2},j} = \int_{x_j,H_j\sigma_{i+1}}^{x_j,H_j\sigma_i} v_{i-\frac{1}{2},j-\frac{1}{2}} \cdot \boldsymbol{n_1} dS_1 + \int_{x_j,H_j\sigma_{i+1}}^{x_j+\Delta x,H_{j+1}\sigma_{i+1}} v_{i,j} \cdot \boldsymbol{n_3} dS_3 + ... \\ \int_{x_j+\Delta x,H_{j+1}\sigma_{i+1}}^{x_j+\Delta x,H_{j+1}\sigma_i} v_{i-\frac{1}{2},j+\frac{1}{2}} \cdot \boldsymbol{n_2} dS_2 + \int_{x_j+\Delta x,H_j\sigma_i}^{x_j,H_{j+1}\sigma_i} v_{i-1,j} \cdot \boldsymbol{n_4} dS_4. \tag{3.28}$$

The Algorithm on a z-coordinate Grid

In the limit of a z-coordinate grid, the gradient operators also reduce to the second-order central difference scheme implemented in the 2DFV code. For example, the horizontal gradient of zonal velocity ($\frac{\partial u}{\partial x}$), shown in Figure 3-12, would have no contribution from faces 3 and 4 to the horizontal flux. The contributions from faces 1 and 2 are

$$\nabla u_{i,j} = \left(\frac{u_{i,j}\Delta y[1,0] + u_{i,j-1}\Delta y[-1,0] + u_{i-\frac{1}{2},j-\frac{1}{2}}\Delta x[0,1] + u_{i+\frac{1}{2},j-\frac{1}{2}}\Delta x[0,-1]}{\Delta x \Delta y} \right) \tag{3.29}$$

$$\nabla u_{i,j} \cdot [1,0] = \frac{1}{\Delta x \Delta y}\left(u_{i,j}\Delta y - u_{i,j-1}\Delta y, u_{i-\frac{1}{2},j-\frac{1}{2}}\Delta x - u_{i+\frac{1}{2},j-\frac{1}{2}}\Delta x\right) \tag{3.30}$$

$$\frac{\partial u}{\partial x} = \frac{u_{i,j} - u_{i,j1-1}}{2(\frac{\Delta x}{2})}. \tag{3.31}$$

Again, we exactly recover the second-order central difference scheme used in the z-coordinate code. An advantage of a C-grid mesh illustrated here is the convenience with which second-order accurate gradients can be computed with few or no interpolated values.

3.2.3 The Advection Scheme

In the σ-coordinate framework we use an upwind advection scheme and compute the advected flux through six regions of each cell, shown in black, blue, and orange in Figure 3-13. The values of the horizontal and vertical velocity interpolated to the center of each cell face, shown in Figure 3-12, are used to compute the conservative form of the advection term in the horizontal (Equation 3.32) and vertical (Equation 3.33) momentum equations

$$\frac{\partial uu}{\partial x} + \frac{\partial vu}{\partial y} \tag{3.32}$$

$$\frac{\partial uv}{\partial x} + \frac{\partial vv}{\partial y}. \tag{3.33}$$

The integral for Equation 3.32 is

$$\begin{aligned}\nabla \cdot \boldsymbol{u}u = \frac{1}{V}\Bigg(& \int_{(x_j+\frac{\Delta x}{2},H(x_{j+\frac{1}{2}})\sigma_{i+1})}^{(x_j+\frac{\Delta x}{2},H(x_{j+\frac{1}{2}})\sigma_i)} u_{i,j-1}[u_{i,j-1}, v_{i-\frac{1}{2},j-\frac{1}{2}}] \cdot \boldsymbol{n_1} dS_1 + ... \\ & \int_{(x_j+\frac{\Delta x}{2},H(x_{j+\frac{1}{2}})\sigma_{i+1})}^{(x_j+3\frac{\Delta x}{2},H(x_{j+\frac{3}{2}})\sigma_{i+1})} u_{i,j}[u_{i,j}, v_{i-\frac{1}{2},j+\frac{1}{2}}] \ \boldsymbol{n_3} \ dS_3 + ... \\ & \int_{(x_j+3\frac{\Delta x}{2},H(x_{j+\frac{3}{2}})\sigma_{i+1})}^{(x_j+3\frac{\Delta x}{2},H(x_{j+\frac{3}{2}})\sigma_i)} u_{i-\frac{1}{2},j-\frac{1}{2}}[u_{i-\frac{1}{2},j-\frac{1}{2}}, v_{i-1,j}]\boldsymbol{n_2} \ dS_2 + ... \\ & \int_{(x_j+\frac{\Delta x}{2},H(x_{j+\frac{1}{2}})\sigma_i)}^{(x_j+3\frac{\Delta x}{2},H(x_{j+\frac{3}{2}})\sigma_i)} u_{i+\frac{1}{2},j-\frac{1}{2}}[u_{i+\frac{1}{2},j-\frac{1}{2}}, v_{i,j}] \ \boldsymbol{n_4} \ dS_4\Bigg).\end{aligned} \tag{3.34}$$

This term represents an important difference between the z-coordinate and σ-coordinate frameworks. The z-coordinate code uses a Total Variation Diminishing advection scheme while the σ-coordinate code uses a first-order upwind scheme. Although upwind schemes lead to more nu-

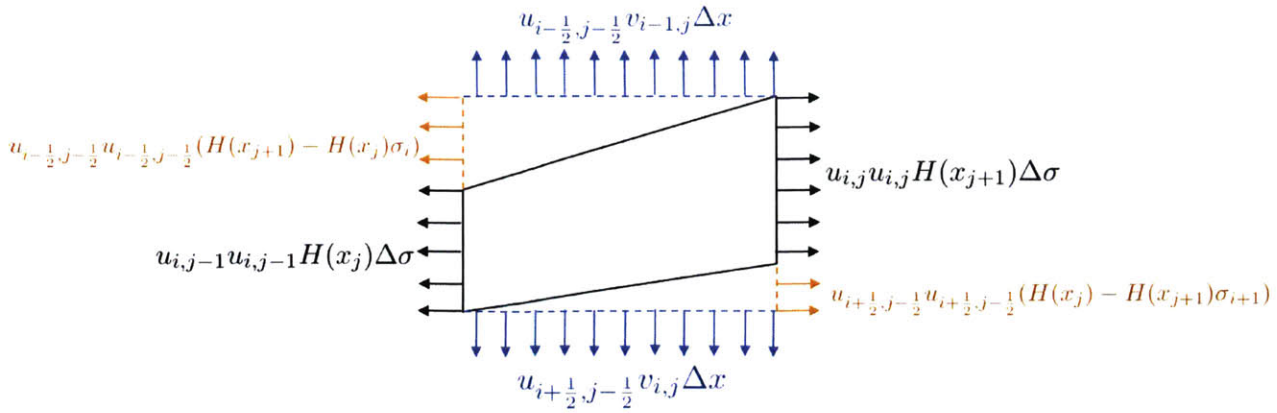


Figure 3-13: The horizontal flux terms are in black and orange and the vertical fluxes are in blue.

merical dissipation than other flux-limiting schemes, we did not observe any significant dissipation in the test-cases used to evaluated the σ-coordinate code.

3.2.4 Neumann and Open Boundary Conditions

The aforementioned schemes apply to all interior cells and cells that impinge upon a Dirichlet boundary ($\phi = \partial_D$). There are two other boundary conditions we wish to implement, a Neumann boundary condition ($\frac{\partial \phi}{\partial n} = \partial_N$), and an open boundary condition ($\frac{\partial^2 \phi}{\partial n^2} = 0$). For a Neumann boundary condition, we use a first-order approximation and set the value of ϕ in the boundary cell equal to the value of ϕ in the adjacent interior cell. For example, a zero-Neumann condition on the top domain boundary for the Laplacian operator would set the following equalities

$$\phi_{i-1,j-1} = \phi_{i,j-1}, \quad \phi_{i-1,j} = \phi_{i,j}, \quad \phi_{i-1,j+1} = \phi_{i,j+1}.$$

So we can replace $\phi_{i-1,j-1}, \phi_{i-1,j}$ and, $\phi_{i-1,j+1}$ in Equation 3.22, and find new coefficients for $\phi_{i,j-1}, \phi_{i,j}$ and, $\phi_{i,j+1}$. An issue with this first-order approximation is that, the steepness of the slope of the σ surface is proportional to the error computed in the Neumann boundary condition. For a σ-surface with a small slope, the trajectory between the center of the bounadry cell and the center of the interior cell will very well approximate the normal direction to the σ-surface. As the σ-surfaces steepen, the trajectory between the center of the boundary cell and the center of the interior cell will less-accurately approximate the normal direction. Specifically for our sea mount test-cases, this should not be an issue for either the LW (the primary motivation for this coordinate

transformation) or TW topographies. However, in order to use the σ-coordinate framework to study the TN topography, a more sophisticated Neumann boundary condition, like a ghost-cell immersed boundary method (Tseng and Ferziger, 2003), would be required.

For an open boundary condition, which we define as $\frac{\partial^2 \phi}{\partial n^2} = 0$, we ensure that the Laplacian in the normal direction to the domain boundary vanishes. We use the open condition on the pressure-correction variable for the inlet and outlet (green regions of Figure 2-1). For example, if the East domain boundary was open, the terms corresponding to $\frac{\partial^2 \phi_{i,j}}{\partial x^2}$ in Equation 3.22, need to sum to zero. Thus, for all nine coefficients, we set the part of the coefficient in Equation 3.22 that corresponds to $\frac{\partial^2 \phi_{i,j}}{\partial x^2}$ to zero.

3.3 Verification

We use two test-cases to verify the accuracy of the σ-coordinate framework. The first test-case is on a z-coordinate grid, so we can compare results from the z-coordinate code with the σ-coordinate code directly. We also use results from this test-case to verify the order of accuracy. The second test-case has an analytical solution.

3.3.1 Verification on z-coordinate Grid

We first consider the Lid-Driven Cavity test-case. The domain is a square with unit area; there are three static walls and one wall with unit velocity as pictured in Figure 3-14. Figure 3-15

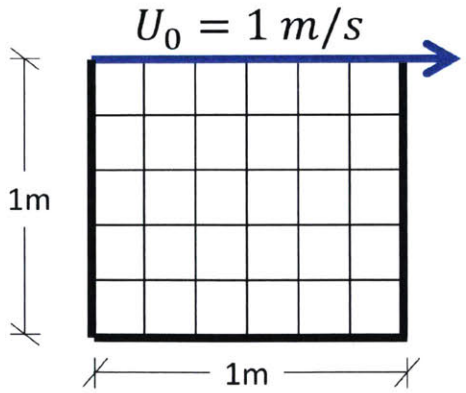


Figure 3-14: The Lid-Driven Cavity test-case has four Dirichlet boundaries and a square domain (where $\Delta x = \Delta y$).

shows the streamlines at the same instance in time from the z-coordinate and σ-coordinate codes,

and we see the results agree very well.

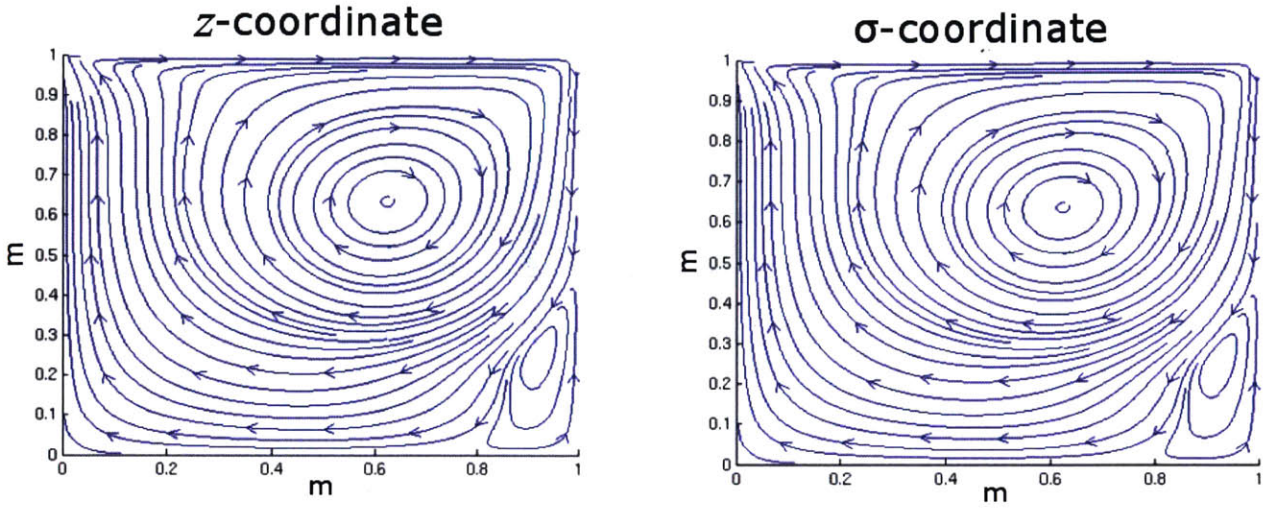


Figure 3-15: Streamlines from the z-coordinate and σ-coordinate codes compare well for a test-case on a z-coordinate grid.

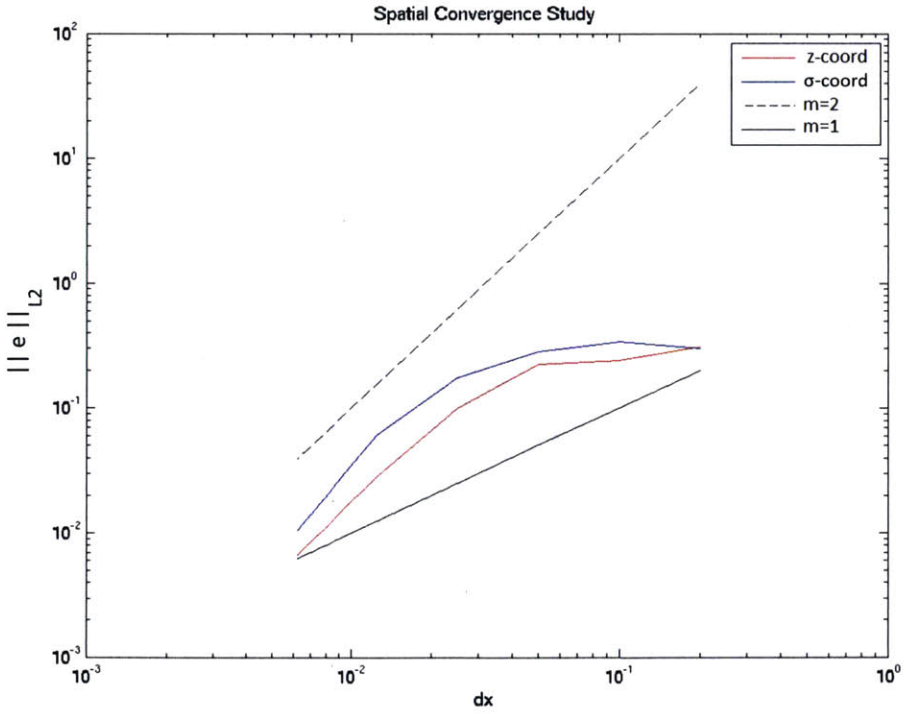


Figure 3-16: A grid refinement study, using the centerline velocity from the Lid-Driven Cavity test-case, demonstrates that the convergence rate for both codes is between first-order and second-order as expected.

A spatial convergence study, shown in Figure 3-16, using the center-line velocity values from

the Lid-Driven Cavity velocity field confirms a convergence rate between first and second-order, as expected. Recall that while the σ-coordinate Laplacian and gradient operators reduce exactly to the second-order, central difference scheme on a z-coordinate mesh, the solution results will not be identical because the advection schemes differ.

3.3.2 Verification with an Analytical Solution

Now we show that the results from the σ-coordinate code agree very well with an analytical solution. We study Couette flow between two parallel plates, but incline both plates at a 45° angle as shown in Figure 3-17. The inlet velocity profile is that of a fully developed Couette flow. The velocity boundary conditions at each plate are zero-Dirichlet conditions, and the right boundary has zero-Neumann conditions for velocity ($\frac{\partial u}{\partial x} = \frac{\partial v}{\partial x} = 0$). The right boundary has an open condition on the pressure-correction, corresponding to the definition implied by the projection method, and the boundary condition for the pressure-correction on the other three boundaries is a zero-Neumann boundary condition.

The mesh for this simulation is pictured in Figure 3-18. This is a very efficient mesh compared to what would be required in a z-coordinate setup to achieve the same resolution; many more cells would need to be generated and then masked in a z-coordinate code because the domain needs to be rectangular.

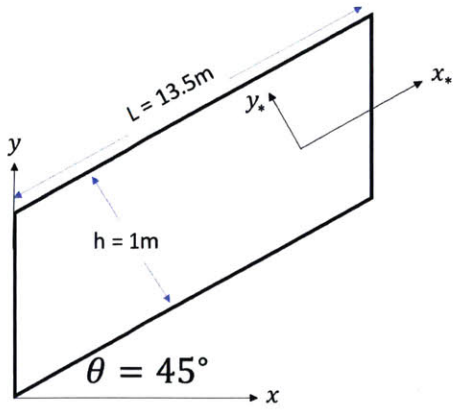


Figure 3-17: The two parallel plates are 1m apart and the domain has a small width to length aspect ratio such that velocity in the across-channel direction vanishes.

The solution of this test-case is the same for any angle θ since there is no gravitational body force, or any other influence that works to break the flow's symmetry about the centerline between

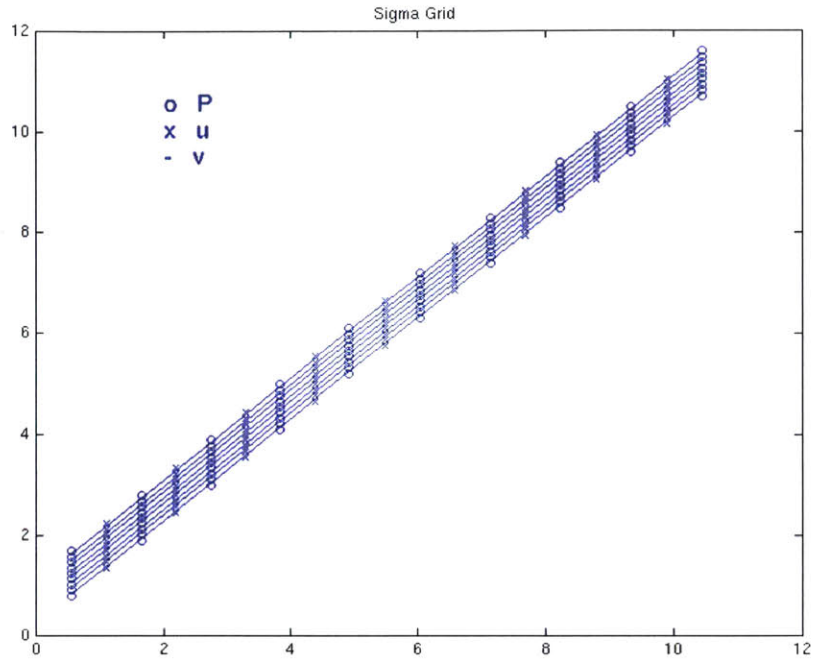


Figure 3-18: σ-coordinate mesh for planar Couette flow

the plates. The analytical velocity field is

$$u(y_*) = \frac{1}{2\nu}\frac{\partial p'}{\partial x}(y_*^2 - hy_*) \tag{3.35}$$

for distance along the planar flow L, and width between the parallel plates h. The x_* and y_* axes are oriented along, and transverse to, the boundaries, respectively. Since the simulation evolves horizontal and vertical velocities, a simple rotation matrix is used to convert the $[u, v]$ velocity field from the simulation to the rotated coordinate system (x_*, y_*).

A velocity profile is sampled midway along the length of the domain after the flow has reached a steady state. The u velocity (shown in Figure 3-19) and v velocity fields are linearly combined to rotate the solution into the (x_*, y_*) coordinate system. The velocity in x_* direction compares very well with the analytical solution, as shown in Figure 3-20. We see excellent agreement between the σ-coordinate results and the analytical solution. Since the σ-coordinate code performs well in both verification tests, we revisit tidal flow over the LW topography.

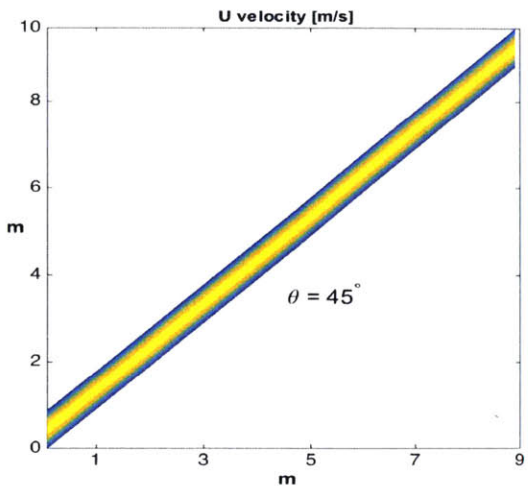


Figure 3-19: The zonal velocity ($\frac{\partial x}{\partial t}$) from the σ-coordinate mesh for planar Couette flow agrees very well with our intuition and with the analytical solution.

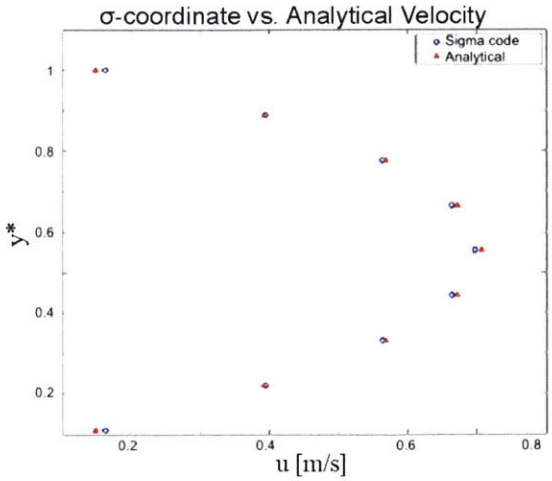


Figure 3-20: A simple rotation is used to write the analytical solution in terms of along-plate velocity and across-plate velocity. The velocity profile from the σ-coordinate code agrees very well with the analytical solution.

3.4 Mitigating Hydrostatic Inconsistency

The zonal velocity field, shown in Figure 3-21, reveals a convective instability above the topography very early in the simulation. We explore and ameliorate this behavior in this section.

3.4.1 The Pressure Gradient Error

A well-documented problem in σ-coordinate formulations is known as the pressure gradient error, or the issue of hydrostatic inconsistency (Lin, 1997; Kanarska and Maderich, 2003; Berntsen,

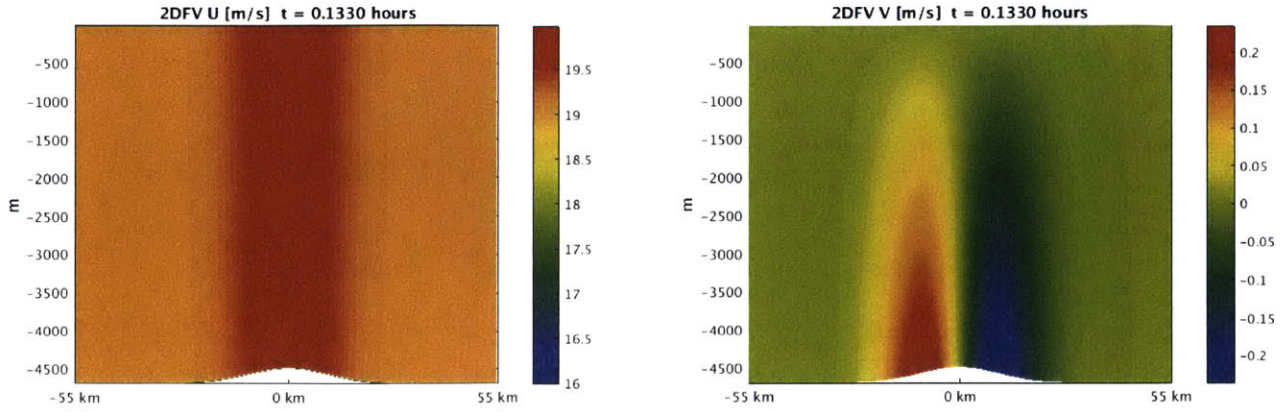


Figure 3-21: We observe convection after a very short time in the cells above the sea mount.

2002; Mesinger, 1982). We will illustrate this challenge with a simple finite difference formulation and then describe the relation to the finite volume formulation. If we convert the horizontal pressure gradient into σ coordinates we have

$$\frac{\partial P}{\partial x} = \frac{\partial \sigma}{\partial x}\frac{\partial P}{\partial \sigma} + \frac{\partial \tilde{x}}{\partial x}\frac{\partial P}{\partial \tilde{x}}. \tag{3.36}$$

In hydrostatic equilibrium $\frac{\partial P}{\partial x} = 0$, so we expect $\frac{\partial \sigma}{\partial x}\frac{\partial P}{\partial \sigma} + \frac{\partial \tilde{x}}{\partial x}\frac{\partial P}{\partial \tilde{x}} = 0$. These two terms are often large, compared to numerical epsilon, and will have opposite signs, but the truncation errors of each of these terms can differ significantly such that there is a remainder after summation. The result of this residual truncation error is spontaneous convection, which is strongest where the σ levels are the steepest. The analogy in the finite volume formulation is that the $\frac{\partial \tilde{x}}{\partial x}\frac{\partial P}{\partial \tilde{x}}$ is the horizontal flux through the two vertical faces (1 and 2) and the $\frac{\partial \sigma}{\partial x}\frac{\partial P}{\partial \sigma}$ term is the horizontal flux through the top and bottom faces (3 and 4).

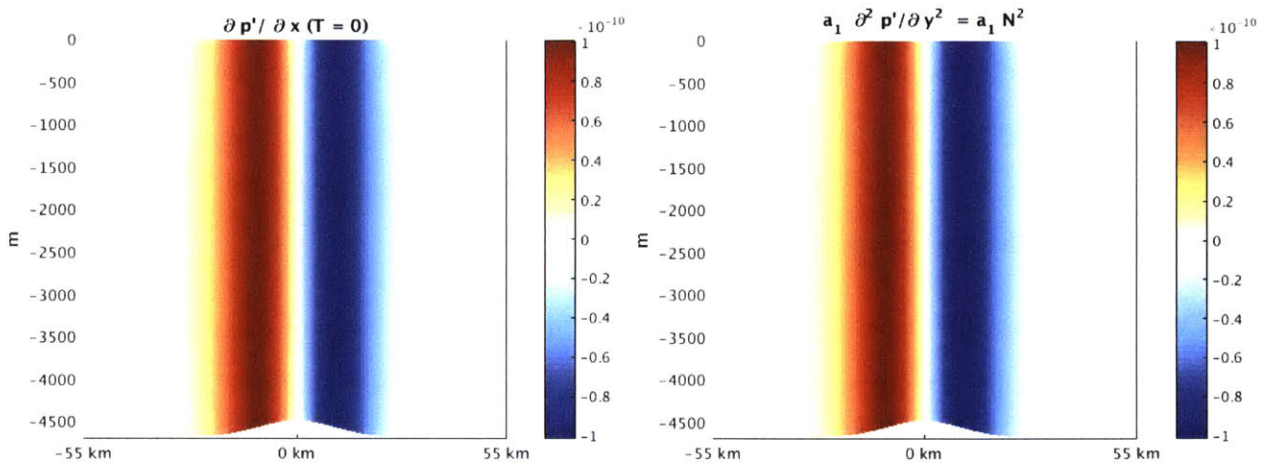
We can see this truncation error in our LW topography simulation. After initializing hydrostatic equilibrium, the pressure gradient is computed and plotted in Figure 3-22a. Next we compute the analytical truncation error by Taylor-expanding the six values of pressure that constitute the numerical stencil for the horizontal pressure gradient, given in Equation 3.37. For constant N, $p'(y) = N^2(H_0 - y)^2/2$, where H_0 is the maximum height of the domain. The analytical values of the second derivatives are $\frac{\partial^2 p'}{\partial x \partial y} = 0$ and $\frac{\partial^2 p'}{\partial y^2} = N^2$.

$$\frac{\partial p}{\partial x} = 1 \cdot \frac{\partial p}{\partial x} + a_1\frac{\partial^2 p}{\partial y^2} + a_2\frac{\partial^2 p}{\partial x \partial y} + a_3\frac{\partial^2 p}{\partial x^2} + \mathcal{O}(\Delta x^2, \Delta\sigma^2, \Delta x \Delta\sigma) \tag{3.37}$$

where

$$\begin{aligned}
a_1 =& \frac{3}{8}\frac{H(x_j) - H(x_{j+1})}{(H(x_{j+1}) + H(x_j))\Delta x}\Bigg((\Delta\sigma^2 - \frac{4}{3}\sigma_i\Delta\sigma + \frac{4}{3}\sigma_i^2)H(x_{j+1})^2 - \ldots \\
& \frac{2}{3}H(x_j)(\Delta\sigma - 2\sigma_i)^2 H(x_{j+1}) + H(x_j)^2(\Delta\sigma^2 - \frac{4}{3}\sigma_i\Delta\sigma + \frac{4}{3}\sigma_i^2)\Bigg) \\
a_2 =& \frac{(H(x_j) - H(x_{j+1}))^2(\Delta\sigma - 2\sigma_i)}{(4H(x_{j+1}) + 4H(x_j))} \\
a_3 =& 0.
\end{aligned}$$

The leading order term, plotted in Figure 3-22b, very well matches the truncation error we observe.



(a) (b)

Figure 3-22: The horizontal pressure gradient, computed upon initializing a hydrostatic pressure field (a), is non-zero in the cells that are skewed by the topography. The first term in the horizontal pressure gradient truncation error (b) agrees very well with the observed error.

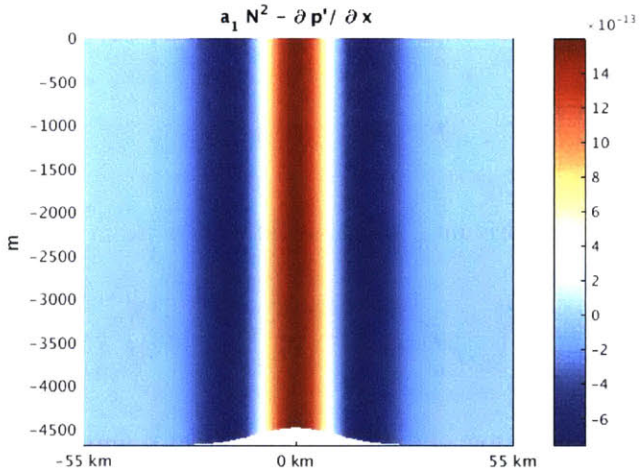


Figure 3-23: The difference between the pressure gradient error in Figure 3-22a and the analytical truncation error in Figure 3-22b can be attributed to higher order terms in the truncation error.

We attribute the discrepancy in magnitude between the observed error and the computed error, in Figure 3-23, to the higher-order terms.

Hydrostatic Model Solutions

One technique used to ameliorate this error in hydrostatic models is to remove the mean density profile (Gary, 1973), this technique is used by the MSEAS PE (Haley and Lermusiaux,

2010). Several formulations that carefully interpolate pressure (Lin, 1997) or density (Song, 1998) to more accurately calculate the pressure gradient force are referred to as pressure-Jacobian and density-Jacobian methods, respectively. Unfortunately, these techniques are not applicable to a non-hydrostatic framework.

Non-hydrostatic Model Solutions

While other models, like NHWAVE, are able to mitigate the effects of the pressure-gradient error with a spatial discretization scheme that inhibits spontaneous flow (NHWAVE uses a second-order Godunov-type finite volume method discretization unlike the 2DFV code), we ameliorate this issue by removing the hydrostatic pressure from the momentum equations and refining the grid spacing. Removing the hydrostatic component of the pressure is common practice in many numerical models since the hydrostatic pressure is generally much larger than the non-hydrostatic pressure, and its removal can free the simulation of errors that are proportional to this larger quantity. First the pressure is separated into two components, the hydrostatic and non-hydrostatic pressures

$$p = p_{\text{hyd}} + p_{\text{non-hyd}}. \tag{3.38}$$

Now we substitute this into our horizontal momentum equation.

$$\frac{\partial u}{\partial t} + \boldsymbol{u} \cdot \nabla u = -\frac{1}{\rho_0}\left(\frac{\partial}{\partial x} p_{\text{non-hyd}} + \frac{\partial}{\partial x}\int_y^0 \frac{\rho'}{\rho_0} g d\zeta\right) + \nu \nabla^2 u + F_u(\boldsymbol{x}, t). \tag{3.39}$$

Using the Leibniz Rule we interchange the order of the integral and the derivative

$$\frac{\partial u}{\partial t} + \boldsymbol{u} \cdot \nabla u = -\frac{1}{\rho_0}\left(\frac{\partial}{\partial x} p'_{\text{non-hyd}} - \int_y^0 \frac{\partial}{\partial x}\left(\frac{\rho'}{\rho_0} g\right) d\zeta\right) + \nu \nabla^2 u + F_u(\boldsymbol{x}, t). \tag{3.40}$$

In the vertical momentum equation we have

$$\frac{\partial v}{\partial t} + \boldsymbol{u} \cdot \nabla v = -\frac{1}{\rho_0}\frac{\partial}{\partial y}(p_{\text{hyd}} + p_{\text{non-hyd}}) + \nu \nabla^2 v + F_v(\boldsymbol{x}, t) - \frac{\rho'}{\rho_0} g \tag{3.41}$$

$$\frac{\partial v}{\partial t} + \boldsymbol{u} \cdot \nabla v = -\frac{1}{\rho_0}\frac{\partial}{\partial y}(p_{\text{non-hyd}}) - \frac{\partial}{\partial y} - \int_y^0 \frac{\rho'}{\rho_0} g d\zeta + \nu \nabla^2 v + F_v(\boldsymbol{x}, t) - \frac{\rho'}{\rho_0} g \tag{3.42}$$

$$\frac{\partial v}{\partial t} + \boldsymbol{u} \cdot \nabla v = -\frac{1}{\rho_0}\frac{\partial}{\partial y}(p_{\text{non-hyd}}) + \frac{\rho;}{\rho_0} g + \nu \nabla^2 v + F_v(\boldsymbol{x}, t) - \frac{\rho'}{\rho_0} g \tag{3.43}$$

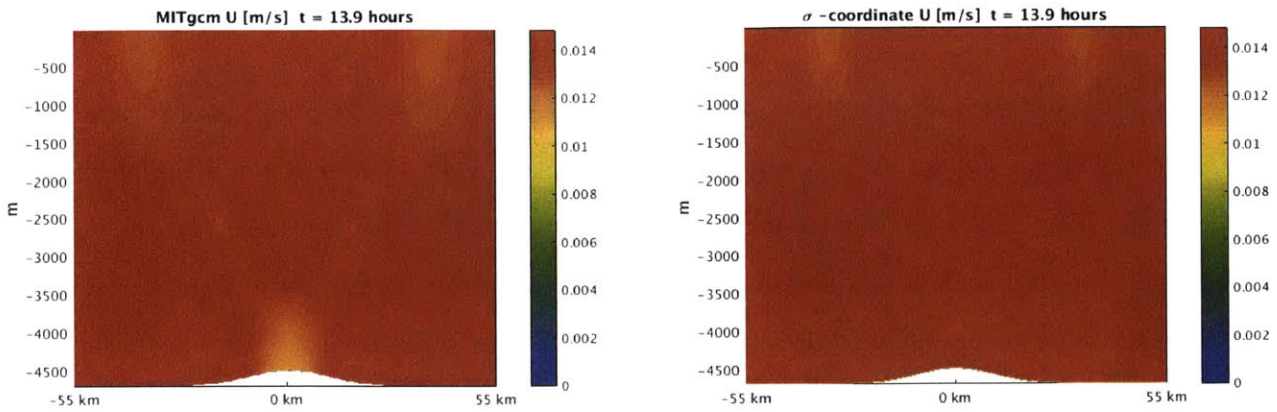


Figure 3-24: The σ-coordinate simulation of the LW topography with the hydrostatic pressure removed is initially free from spontaneous convection due to the pressure gradient error. However, a boundary layer develops near the topography that accelerates or decelerates the flow near the boundary.

$$\frac{\partial v}{\partial t} + \boldsymbol{u} \cdot \nabla v = -\frac{1}{\rho_0}\frac{\partial}{\partial y}(p_{\text{non-hyd}}) + \nu \nabla^2 v + F_v(\boldsymbol{x}, t). \tag{3.44}$$

Analytically, the two formulations (Equations 2.5 and Equations 3.40 and 3.44) have identical velocity field solutions. However, numerically we expect to see differences due to the relative scaling of the computed quantities and their associated round-off and truncation errors.

Second, while it is commonly understood that truncation error is proportional to the grid size, the pressure gradient error in particular has been shown to diminish with the square of the horizontal and vertical grid spacing (Mellor et al., 1994); instead of using the same resolution as the MITgcm simulation, we use a grid that is twice as fine in the horizontal (dx =100m instead of 200m) to mitigate the truncation error. The boundary conditions on velocity and the pressure-correction term are the same as the z-coordinate simulation.

After these modifications the results of the LW simulation, shown in Figure 3-24, significantly improve from the previous simulation, shown in Figure 3-21, although the zero-Dirichlet condition on the sea mount does cause a boundary layer with slightly larger velocities to develop near the topography. There is room improve upon this solution and bring the result of the LW simulation closer to that of the shaved cell simulation, but this procedure is an important first step towards that goal.

3.5 Conclusion

We have discussed in detail the algorithms in our non-hydrostatic σ-coordinate framework. Our goal was to ameliorate stair-casing effects and make progress towards a moving-mesh, free surface code. The σ-coordinate code better represents the bathymetry and ameliorates the stair-casing effect we saw with the z-coordinate code. The algorithm is first-order accurate and the fully implicit diffusion operator reduces the computational cost substantially. Future work includes improving the schemes to further reduce the prevalence of the hydrostatic inconsistency truncation error, which may require a more sophisticated boundary condition.

Chapter 4

Application of the Dynamically Orthogonal Field Equations

For the first time we utilize the Dynamically Orthogonal (DO) field equations to study the sensitivity of internal tide dynamics to perturbations in the linear stratification. These simulations are conducted with the 2DFV, z-coordinate code which we demonstrated in Chapter 2 can accurately capture the relevant dynamics of internal tides for the TN test case. The DO field equations achieve substantial model order reduction , and efficiently explore a solution space (Sapsis and Lermusiaux, 2009, 2012; Ueckermann et al., 2013; Subramani and Lermusiaux, 2016). The DO field equation have also been utilized for data assimilation (Sondergaard and Lermusiaux, 2013a,b; Lolla and Lermusiaux, 2016a,b).Conceptually, the advantage of the DO methodology over a Monte-Carlo approach can be characterized as projecting a high-dimensional problem onto a lower-dimensional subspace, and then evolving the subspace in time.

In the first section we outline the DO methodology. In the second section we discuss the initialization of the mean, modes, and stochastic coefficients. In the third and fourth sections we discuss the results and verification of the DO simulation. We acknowledge that the verification of the DO simulation, while encouraging, is not definitive, and that more work need to be done to ensure the fidelity of the simulation. In the fifth section, we proceed with a preliminary evaluation of the influence of initial perturbations to the density field on the resultant available potential energy. This analysis, though contingent on the accuracy of the DO simulation, does suggest a substantial sensitivity to the density anomalies introduced at the internal tide generation site.

4.1 DO Methodology

We briefly discuss the DO methodology and governing equations. Let the vector containing all the state variables be

$$\mathcal{S} = \begin{bmatrix} u \\ v \\ \rho \end{bmatrix}. \tag{4.1}$$

We can define a generic evolution equation for the state vector as

$$\frac{\partial \boldsymbol{\mathcal{S}}(\boldsymbol{x}, t; \Omega)}{\partial t} = \mathcal{L}(\boldsymbol{\mathcal{S}}(\boldsymbol{x}, t; \Omega), \boldsymbol{x}, t; \Omega) \tag{4.2}$$

where $\boldsymbol{\mathcal{S}}$ is a function of two spatial dimensions ($\boldsymbol{x} = [x, y]$), time (t), and random variable (Ω). The operator $\mathcal{L}$ in our simulation represents the Navier-Stokes equations, a DO derivation for which is found in Ueckermann et al. (2013). After a Karhunen-Loéve (KL) decomposition,

$$\mathcal{S}(\boldsymbol{x}, t; \Omega) = \bar{\mathcal{S}}(\boldsymbol{x}, t) + \sum_{i=1}^{s} \mathcal{S}_i(\boldsymbol{x}, t) Y_i(t; \Omega), \tag{4.3}$$

Sapsis and Lermusiaux (2009) apply an orthogonality condition to the stochastic subspace spanned by s DO modes.

$$\langle \frac{\partial \mathcal{S}_i}{\partial t}, \mathcal{S}_j \rangle = \delta \qquad \text{for i,j = 1,2,3...,s} \tag{4.4}$$

Finally, Sapsis and Lermusiaux (2009) use 4.3 and 4.4 to find evolution equations for the mean ($\bar{\mathcal{S}}$), modes ($\mathcal{S}_i$), and stochastic coefficients (Y_i).

$$\frac{\partial \bar{\boldsymbol{u}}(x, t)}{\partial t} = E^{\Omega}[\mathcal{L}[\boldsymbol{u}(x, t; \Omega), \boldsymbol{x}, t; \Omega]] \tag{4.5}$$

$$\frac{\partial \boldsymbol{u_i}(x, t)}{\partial t} = \Pi_{V_S^{\perp}}[E^{\Omega}[\mathcal{L}[\boldsymbol{u}(x, t; \Omega), \boldsymbol{x}, t; \Omega] Y_j(t; \Omega)]] C^{-1}_{Y_i Y_j} \tag{4.6}$$

$$\frac{dY_i(t; \Omega)}{dt} = \langle \mathcal{L}[\boldsymbol{u}(\boldsymbol{x}, t; \Omega), \boldsymbol{x}, t; \Omega] - E^{\Omega}[\mathcal{L}[\boldsymbol{u}(\boldsymbol{x}, t; \Omega), \boldsymbol{x}, t; \Omega]], \boldsymbol{u_i}(\boldsymbol{x}, t) \rangle \tag{4.7}$$

where $\Pi_{V_S^{\perp}}[F(x)] = F(x) - \Pi_{V_S}[F(x)] = F(x) - \langle F(\boldsymbol{x}), u_k(x, t) \rangle u_k(x, t)$ and $C_{Y_i Y_j} = E^{\Omega}[Y_i(t; \Omega) Y_j(t; \Omega)]$. The expectation of a quantity with respect to a random variable Ω is written $E^{\Omega}[\bullet]$, and a spatial inner product between two quantities is written $\langle \bullet_i, \bullet_j \rangle$. Thus, we find that the computational cost of evaluating the DO field equations is aproximately equal to $s^2 + 1 + 1$

deterministic simulations. The s^2 term comes from the modes equations, since any mode can be advected by any other mode. The other two equations are the mean and coefficient equations which have approximately the same cost as a deterministic simulation. For a complete derivation and explanation of the DO equations and their numerical implementation, see Sapsis and Lermusiaux (2009); Ueckermann et al. (2013), and Sapsis (2011).

4.2 Initialization

We need to initialize mean fields, mode fields, and stochastic coefficients for each component of $\mathcal{S}$. The mean fields for velocity and density are initially equal to the deterministic initial conditions where the fluid is at rest and stably stratified. Next we identify a region within the domain where we will initialize variability (i.e. where the stochastic term $\mathcal{S}_i Y_i$ will be non-zero). In our simulation we use a rectangular region surrounding the peak of the topography that is $2L_x = 9720$m wide and $2L_y = 1880$m high. We initialize the stochastic term, $\mathcal{S}_i Y_i$ from a singular value decomposition of a covariance kernel. To compute the covariance between any two points we multiply the correlation between the two points by the standard deviation at each point

$$\text{cov}_{1,2} = \kappa_{1,2}\sigma_1\sigma_2. \tag{4.8}$$

The correlation between any two points (x_1, y_1) and (x_2, y_2) in this variability region will be determined by the distance between them according to

$$\kappa_{1,2} = e^{\left(\frac{|x_1-x_2|}{L_x}\right)^2+\left(\frac{|y_1-y_2|}{L_y}\right)^2}, \tag{4.9}$$

and zero for all other points. This formulation works to prevent steep density gradients, since points which are closer together are more closely correlated. The standard deviation (σ) of density will be defined at each point in this region according to a Gaussian function $\sigma_0 e^{-(x/m_x)^2-(y/m_y)^2}$, as seen in Figure 4-1, where $\sigma_0 = 2$, $m_x = \frac{L_x}{2}$, and $m_y = \frac{L_y}{2}$. The s orthonormal modes,

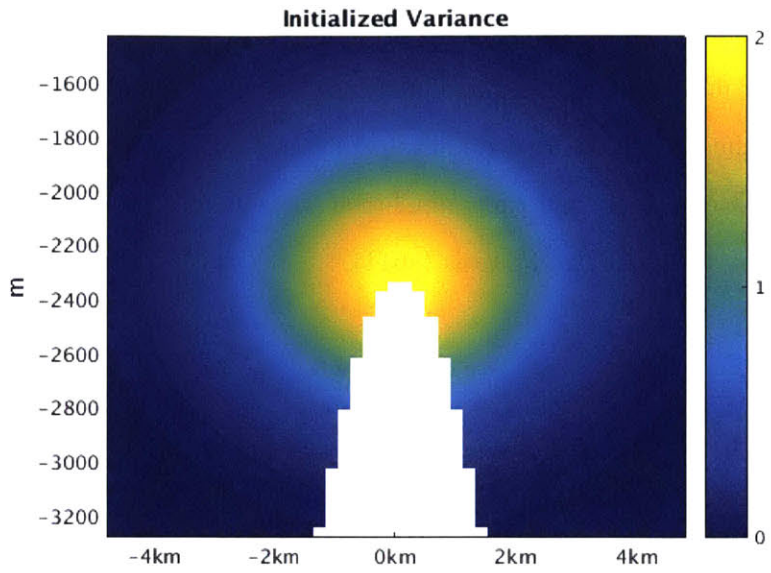


Figure 4-1: The standard deviation of ρ at each point in the variability region is $\sigma_0 = 2$ weighted by a Gaussian concentric with the variability region.

$$\mathcal{S}_i = \begin{bmatrix} \boldsymbol{u_i} \\ \boldsymbol{v_i} \\ \boldsymbol{\rho_i} \end{bmatrix}$$

have zero variability in the velocity fields ($\boldsymbol{u_i} = \boldsymbol{v_i} = \vec{0}$) and variability in the density field ($\boldsymbol{\rho_i}$) is proportional to the first s singular vectors of the covariance matrix. The first four modes are pictured in Figure 4-2. The stochastic coefficients are proportional to the square-root of the singular values and a standard normal distribution.

For the following simulations we use 12,000 realizations and 28 modes. If we carry too few modes or too few realizations, we will insufficiently characterize the total variability in the system. To ensure that we have enough modes, we check that the variance in the highest stochastic coefficients (corresponding to the highest modes) is small compared to the lowest stochastic coefficients (corresponding to the lowest modes), as in Figure 4-3. To ensure we have enough realizations, we validate the DO simulation with an ensemble Monte-Carlo simulation described in section 4.4.

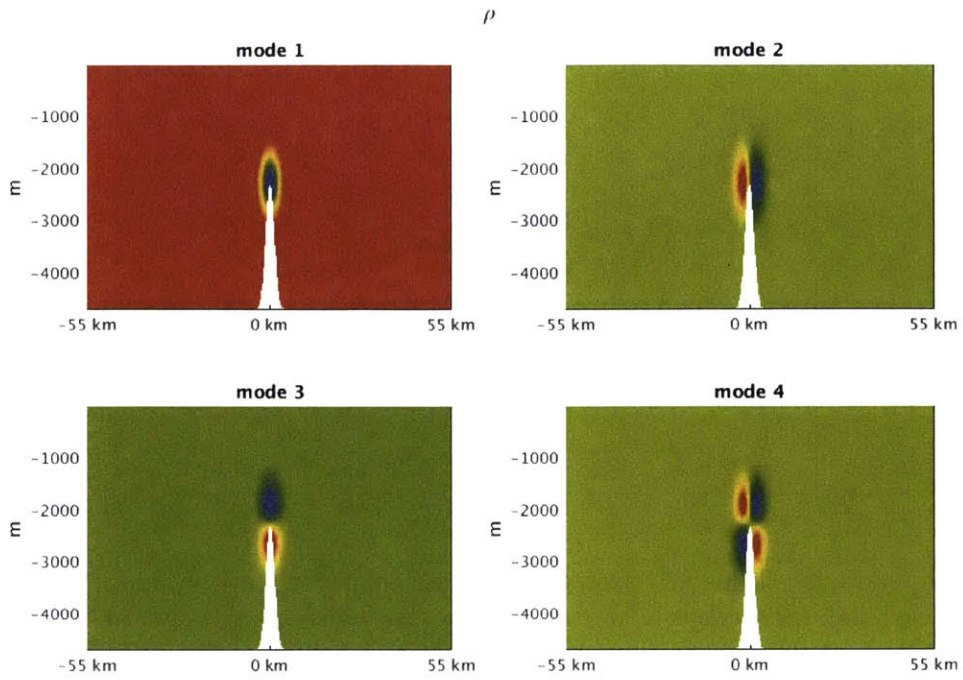


Figure 4-2: The first four of 28 orthonormal modes are proportional to the first four singular vectors of the covariance kernel.

4.3 Results

We describe the evolution of the mean, modes, and coefficients, as well as several realizations from the DO simulation.

4.3.1 Mean

The mean fields of velocity and density, shown in Figure 4-4, closely resemble the fields in the deterministic simulation shown in Figure 2-2. This is expected, since, random perturbations like the ones we apply, are not likely to exert a net force on the mean field.

4.3.2 Modes

The first four modes for u, v, and ρ after 13.9 hours of tidal forcing are pictured in Figure 4-5. We see internal tide-like structures in the modes. The modes of the zonal velocity are anti-symmetric about a vertical centerline, and the modes of the azimuthal velocity are symmetric about the horizontal centerline; the same symmetries and anti-symmetries exhibited by internal tides as seen in Figure 2-2.

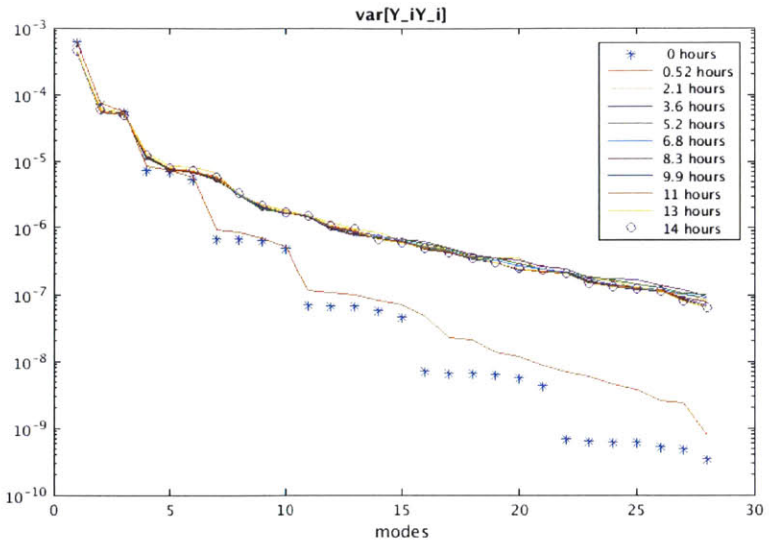


Figure 4-3: The variance drops off between the first and last modes by more than three orders of magnitude, and the distribution of variance stays very constant throughout 13.9 hours of tidal forcing.

4.3.3 Stochastic Coefficients

Figure 4-6a shows the stochastic coefficients are initially uncorrelated Gaussian distributions. As the simulations evolve, features in the marginals of the stochastic coefficients develop in response to coupling of the modes and nonlinearities in the perturbed internal tide simulation dynamics. Figure 4-6b shows the marginal pdf's of the stochastic coefficients after 13.9 hours of stochastic forcing.

4.3.4 Realizations

Finally, we show four realizations (the realization number is indicated by the reference number in the parenthesis) out of the 12,000 after 13.9 hours of tidal forcing in Figure 4-7. Recall that the DO methodology allowed us to evolve 12,000 realizations at the computational cost of $s^2+1+1 = 786$ deterministic simulations. The influence of the different perturbations in each realization can be clearly observed.

4.4 Verification with Monte-Carlo

To verify the DO simulation results, we perform an ensemble Monte-Carlo simulation where each initial condition in the ensemble corresponds to an initial realization of the DO simulation. The

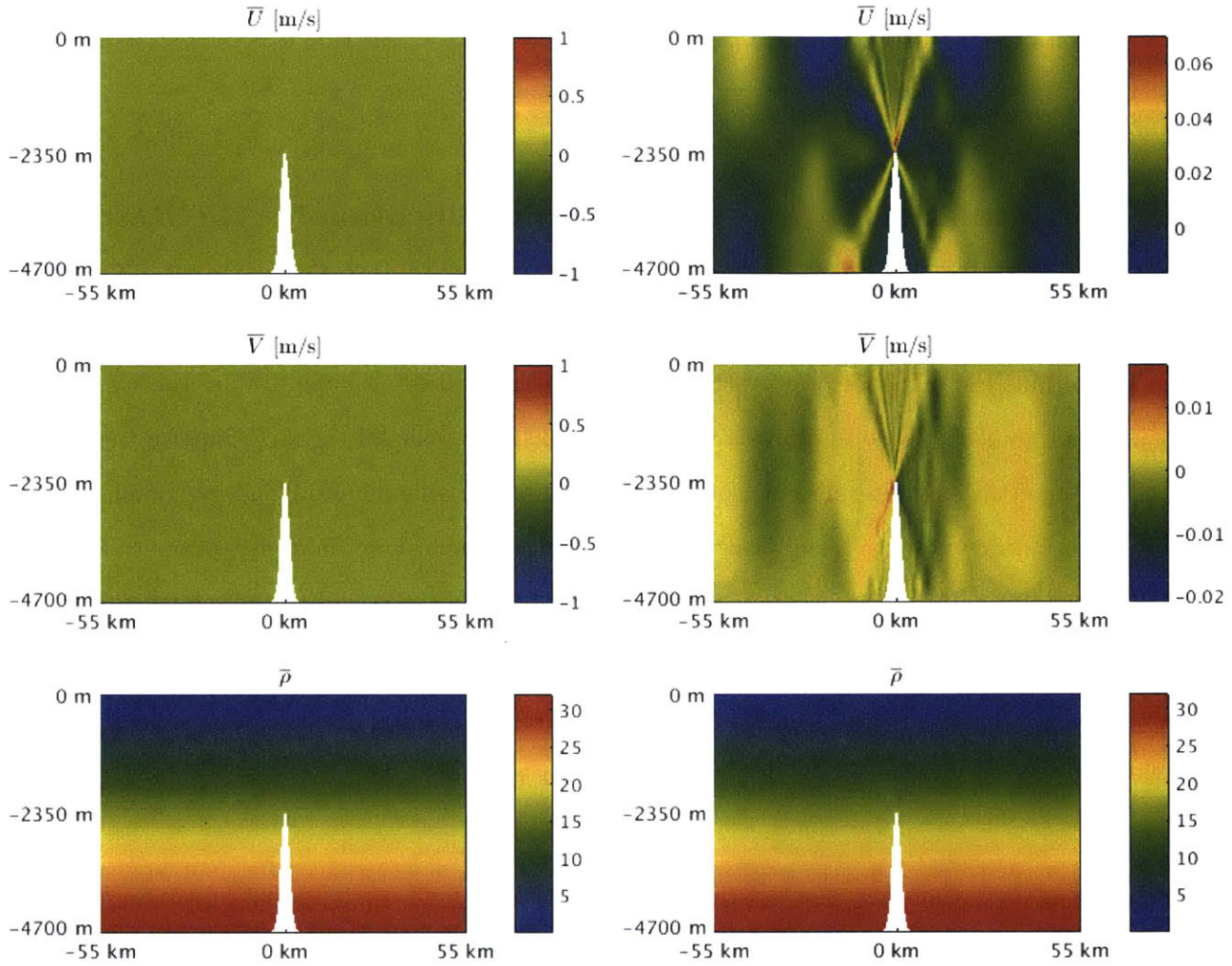


Figure 4-4: The mean fields initially match the deterministic initial conditions (left) and the mean fields after 13.9 hours (right) agree well with the deterministic simulation results after tidal forcing.

ensemble Monte-Carlo simulation agrees very well with the DO simulation; Figure 4-8 shows that after 13.9 hours of tidal forcing, the relative error between the DO simulation and the Monte-Carlo simulation is about 1%. This result is encouraging, but does not guarantee the accuracy of the DO simulation. To fully ensure the accuracy of the DO simulation, a careful investigation of the stochastic density and velocity fields would be required. This task is complicated by the osciallatory nature of the tides and the wave interactions between the tides and gravity waves generated by the density anomalies. A more thorough description of these challenges can be found at the end of this chapter.

We will proceed with the energetics analysis and draw some preliminary conclusions about the relative influence of variability in the density field on the mixing potential of the stratified fluid,

but understand that a more rigorous verification should be conducted to confirm the validity of the DO simulation.

4.5 The Implications of a Perturbed Stratification for the Resultant Available Potential Energy

To examine the influence of perturbations to the density field near the peak of the sea mount, we calculate the available potential energy for the 12,000 DO realizations and compare them with that of the unperturbed, deterministic simulation. We use available potential energy (Lorenz, 1955) as a proxy for mixing potential, and draw some preliminary conclusions about the amount of mixing potential the internal tides can accomplish with and without perturbations to the stratification. We partition the potential energy as in Winters et al. (1995), where we define the total potential energy as the sum of a background potential energy E_b and available potential energy E_a which can be written as the following volume integrals

$$E_b = g \int_V \frac{\rho'}{\rho_0} y_* dV \tag{4.10}$$

$$E_a = g \int_V \frac{\rho'}{\rho_0} \| y - y_* \| dV \tag{4.11}$$

where we set $y_* = H - \frac{\rho'}{\rho_0} \frac{g}{N^2}$, which is exactly Equation 2.1[1]. The notions of "background" and "available" come from the capacity of the potential energy to do work. It is easily observed from Equation 4.11 that the deterministic simulation is initialized with zero available potential energy, and can only acquire it when tidal forcing across the sea mount bends the isopycnals. By contrast, the DO simulations are initialized with some available potential energy since the fluid in the variability region is not necessarily stably stratified. Figure 4-9a shows the initial available potential energy of the DO simulation realizations is between 1% and 90% of the available potential energy in the deterministic simulations after 13.9 hours. The abscissa is the maximum deviation, within the variability region, from the linear profile given by our constant buoyancy frequency. We define the

[1]The definition we use here differs from that in (Winters et al., 1995). The background potential energy definition used by (Winters et al., 1995) is the minimum potential energy that can be achieved by adiabatically re-arranging the fluids parcels. Since the internal tide simulation does not introduce large variations in the relative amount of fluid parcels of a given density, our definition of y_* will closely approximate the definition in (Winters et al., 1995).

maximum perturbation as

$$\max\left(\frac{|\rho^r - \overline{\rho(y)}|}{\overline{\rho(y)}} \times 100\right) \tag{4.12}$$

where ρ^r is the initial density field for a given DO realization r and the linear profile is $\overline{\rho(y)}$. Almost all of the DO simulations maintain a larger available potential energy than the deterministic simulation, as seen in Figure 4-9b. We consider one realization to clarify the results in Figures 4-9a and 4-9b. This realization is initialized with a perturbation that has a maximum departure from the linear profile of 20% and is configured such that the initial available potential energy is 30% of the available potential energy in the deterministic simulation after 13.9 hours. After the same amount of tidal forcing, this perturbed simulation has double the available potential energy of the unperturbed deterministic simulation. The likely explanation for this is that the wavefield in the DO simulations is the interaction of internal tides and gravity waves spawned by the perturbation. Even without tidal forcing, the density anomalies introduced initially in the stochastic simulations would generate internal gravity waves since the fluid in the variability region is unlikely to be stably stratified. It is plausible that these internal gravity waves interfere with internal tides in such a way as to bolster the total available potential energy in the system.

4.5.1 Conclusion

We have applied the DO field equations to study internal tides. The marginals of the stochastic coefficients show the modes of the simulation couple and interact, and the verification of the DO simulation with the ensemble Monte-Carlo simulation is encouraging, but as we have mentioned, not definitive. If we explore the results of this DO simulation the amount of available potential energy created by our characterization of the sub-grid scale eddies is impressive, and the efficiency of the DO methodology allowed us to study a wide range of perturbations and better understand the relationship between the amount of initial available potential energy and the final available potential energy after tidal forcing. A complete dynamical characterization of internal tide-internal gravity wave interaction would be required to fully explain the energetics that we have begun to explore with these stochastic simulations. The DO field equations may not only prove to be a useful technique to efficiently study the sensitivity of internal tides to variability in the environment, but also for

efficiently studying wave-wave interactions such as internal tide-internal gravity wave interactions or internal tide-surface wave interactions.

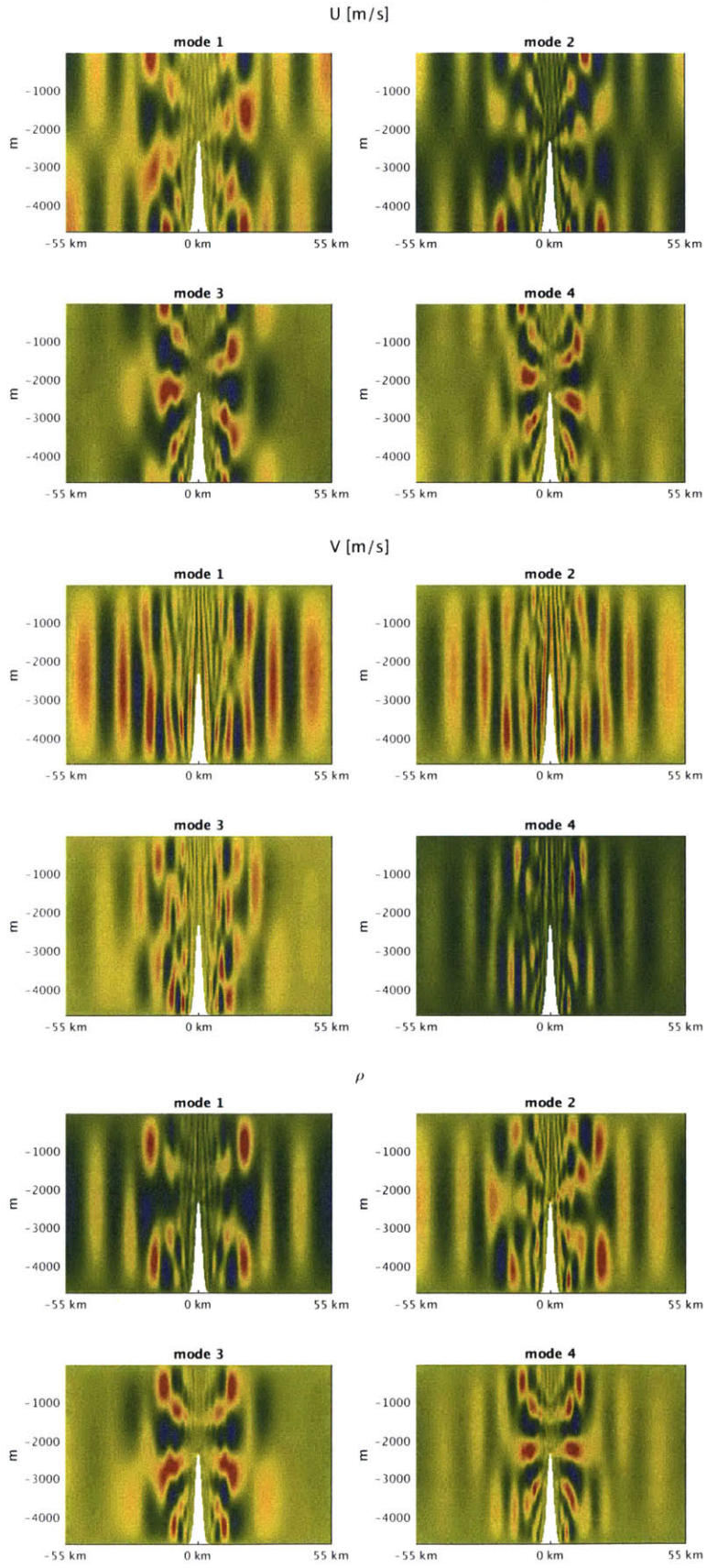
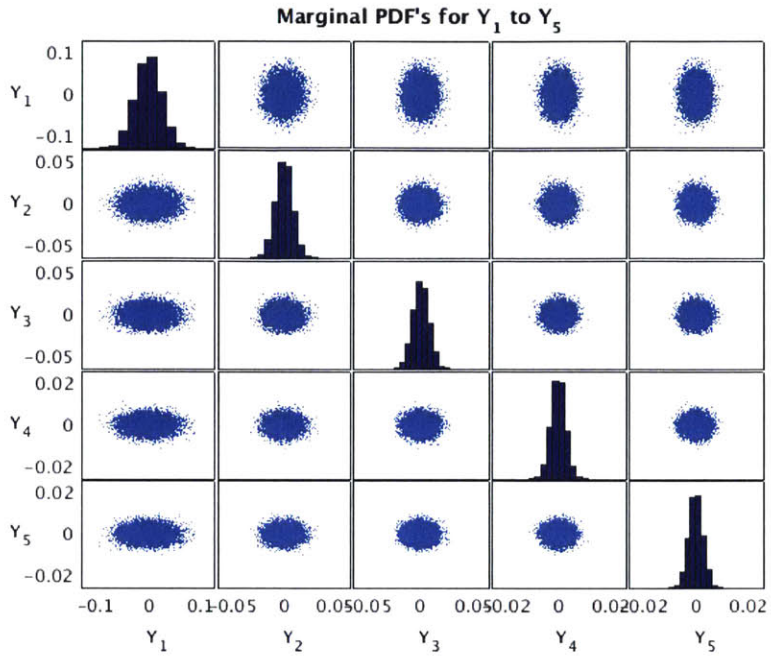
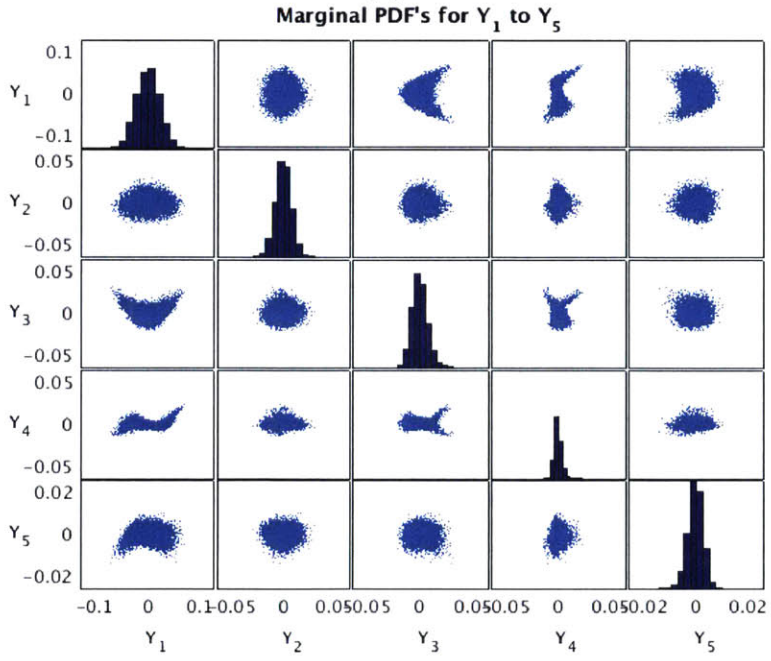


Figure 4-5: The first four modes of zonal velocity (top), azimuthal velocity (middle) and density (bottom) after 13.9 hours of tidal forcing.

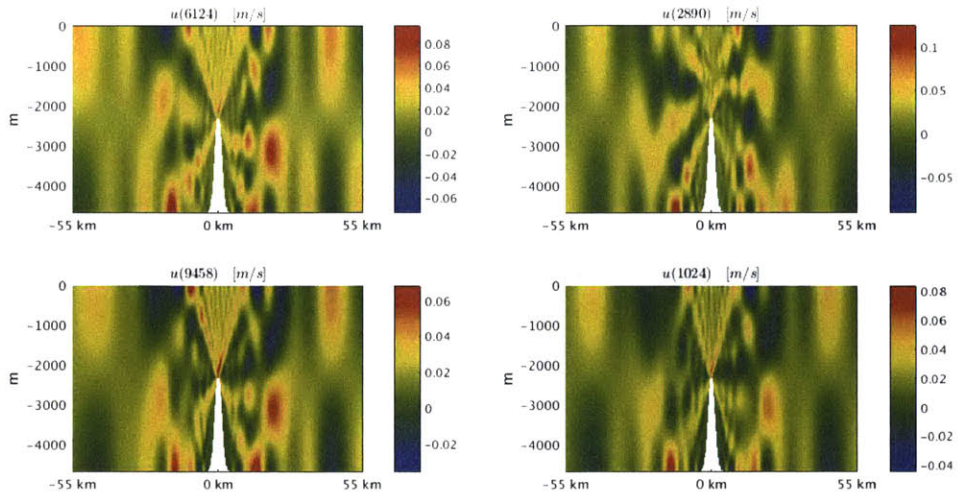


(a)

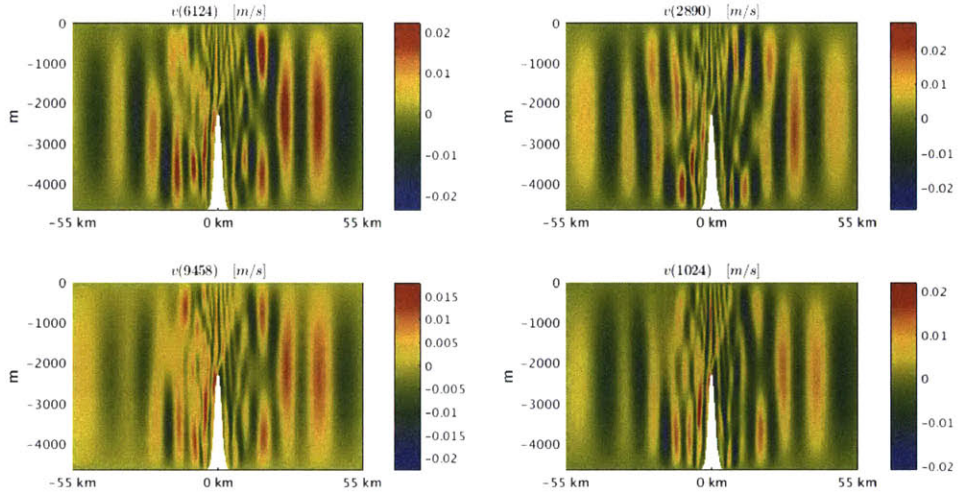


(b)

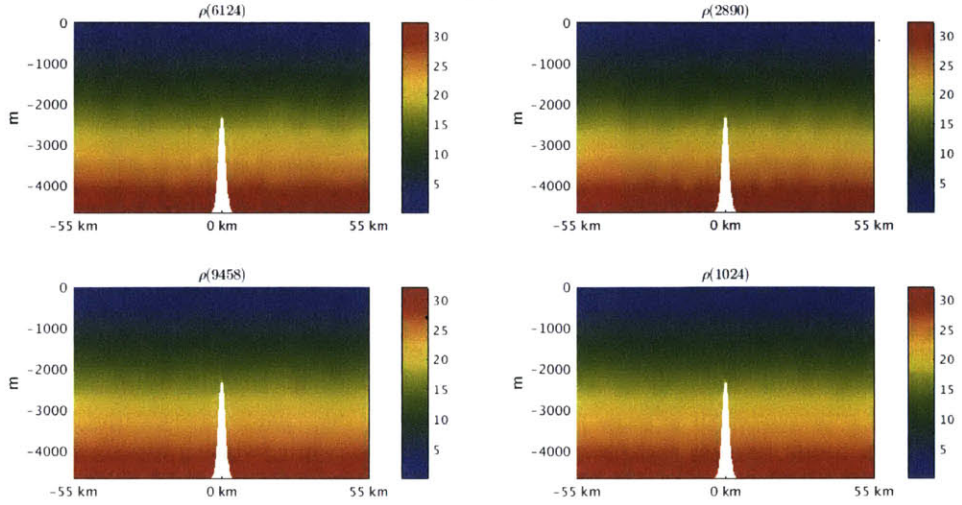
Figure 4-6: The stochastic coefficients are initially uncorrelated Gaussians, then as the simulation progresses, the probability density functions of the coefficients reflects the coupling of the modes and interaction of the gravity waves and internal tides.



(a)



(b)



(c)

Figure 4-7: Four of the 12,000 realizations (the realization numbers are indicated above each figure) of the zonal velocity (top), azimuthal velocity (middle), and density (bottom) have distinct wave patterns due to the different perturbations to the density field.

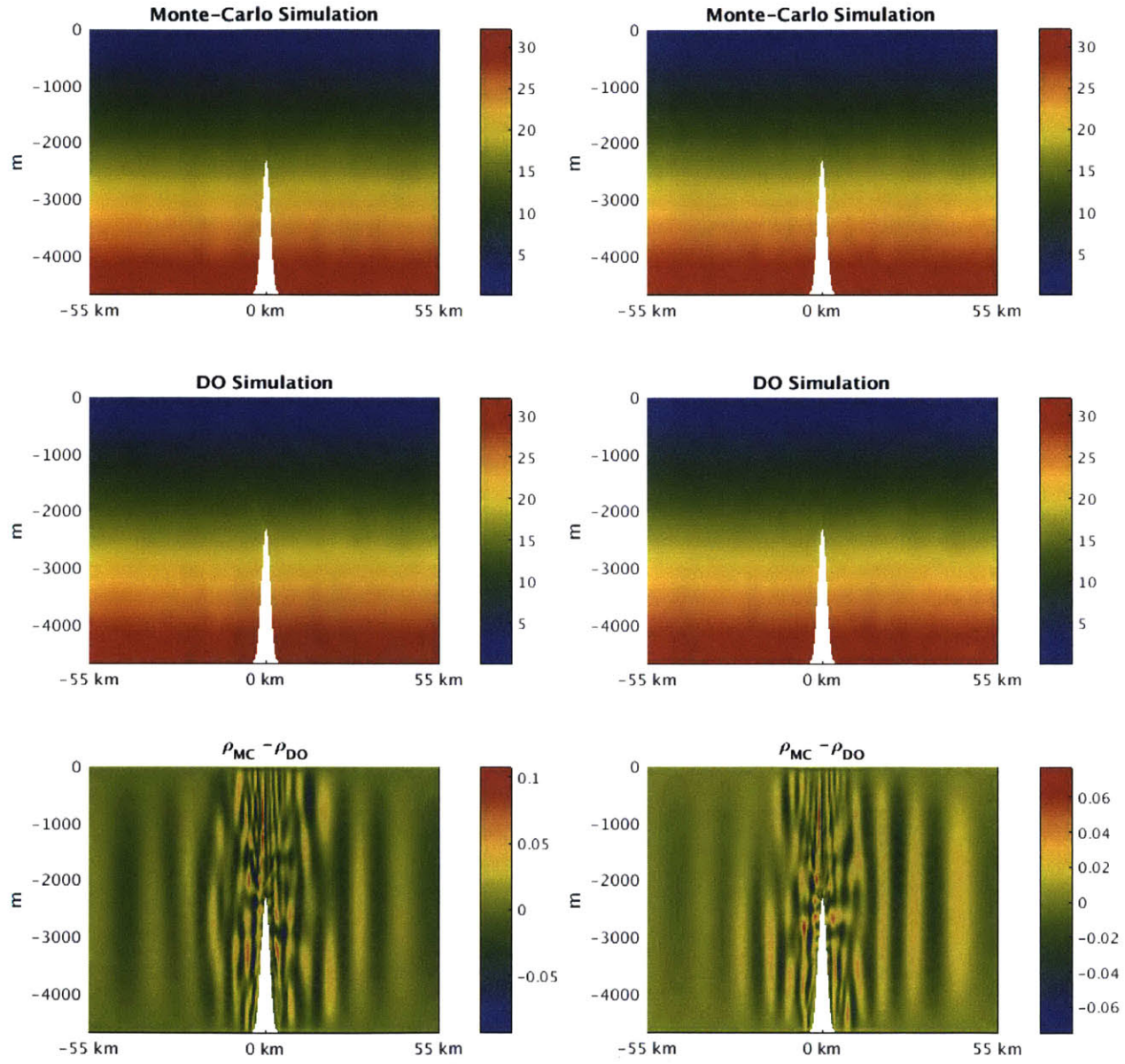
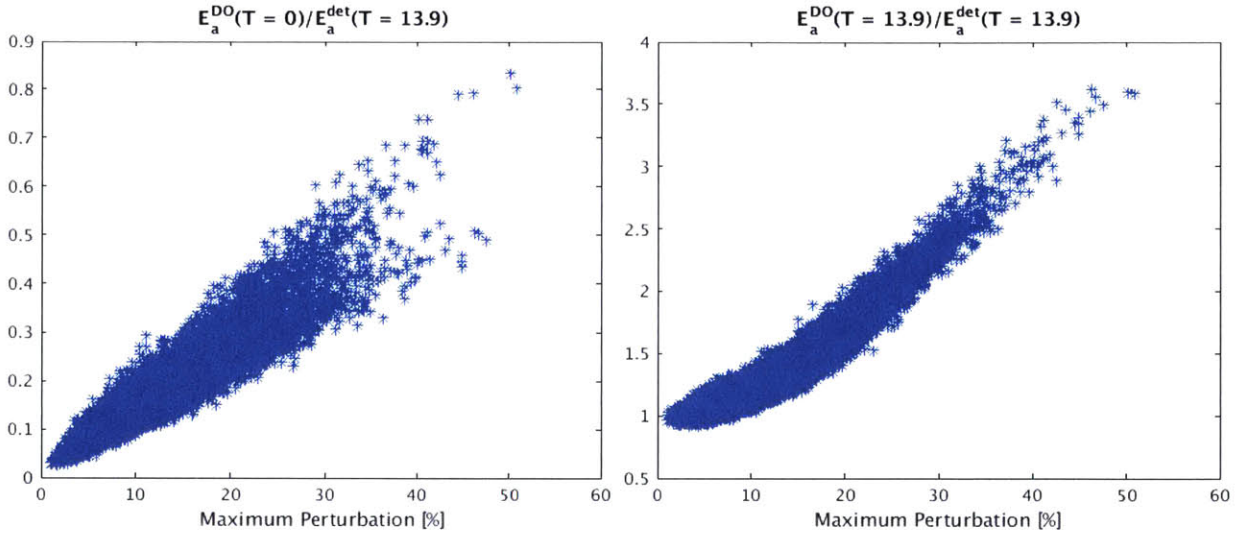


Figure 4-8: The Monte-Carlo simulations (top) initialized with the initial conditions from a DO realization agrees very well with the corresponding DO realizations (middle) after 13.9 hours of tidal forcing. The relative error $\frac{\Delta\rho}{\rho}$ is within 1 percent. A careful investigation into both the density and velocity fields is required for a complete verification of this simulation.



(a) (b)

Figure 4-9: The maximum perturbations away from the linear profile initially (a) and after 13.9 hours of tidal forcing (b) for all 12,000 realizations is pictured above.

Chapter 5

Conclusions and Future Work

5.1 Conclusion

We have shown internal tide dynamics can be sufficiently captured with the 2DFV code by demonstrating excellent agreement between the simulation results and those predicted by linear theory for both the angle of propagation and horizontal wavelength. The power spectra of the internal tides confirmed linear theory is an appropriate simplification and the 2DFV particle velocity field matched the MITgcm results well for the TN topography used in the DO simulations. For low-aspect ratio topographies, a coordinate transformation can significantly improve the particle velocity field between the MITgcm and 2DFV code. Our 2D, non-hydrostatic, σ-coordinate framework has a fully implicit diffusion operator which offers a substantial advantage in computational cost. The issue of hydrostatic inconsistancy was ameliorated by removing the hydrostaic pressure and refining the grid spacing. Finally, we have, for the first time, utilized the Dynamically Orthogonal field equations to study internal tides. Specifically we aimed to study the sensitivity of internal tides to sub-grid scale phenomena paramaterized by perturbations in the stratification near the apex of a steep topography.

5.2 Future Work

Some possible directions to move forward with the σ-coordinate framework and the stochastic simulations are described below. Broadly the goals would generally focus on improving the repre-

sentation of internal tides in the LW simulation with the σ-coordinate framework and conducting a careful verification of the stochastic simulation.

5.2.1 σ-coordinate Framework

Future work on the σ-coordinate framework includes reducing the truncation error to more effectively control the pressure-gradient error and improving the topographic boundary condition. A finite-volume scheme like a Godunov-type method has been shown to lend itself to substantially control the pressure-gradient error, and might be a viable option for the σ-coordinate framework. Since the Neumann boundary condition is only first-order, and the position of the velocity values are vertically aligned, the accuracy of the velocity near the topography is inversely proportional to the steepness of the topography. As mentioned earlier, this does not pose significant challenges for the shallow, low-aspect ration topographies, but in order for this code to be extended for steep topographies a more sophisticated Neumann boundary condition would be required. The immersed ghost cell method is one possible option.

The σ-coordinate code would ease the incorporation of a free-surface dynamics into the 2DFV code, which is important for shallow-water simulations. Future work on the 2DFV code would includes incorporating free-surface dynamics with a moving mesh as is in the MSEAS PE (Haley and Lermusiaux, 2010; Haley et al., 2015).

5.2.2 Application of the DO Field Equations

A careful deterministic investigation of the internal tide-internal gravity wave dynamics would be required to carefully verify the stochastic simulations. The work in this thesis represents the first stages of exploring these dynamics, and demonstrates a potential for the wave-wave interactions to influence the properties of the wavefield far away from the internal tide generation site.

Future work includes incorporating stochastic forcing to study the influence of a variable barotropic tide and surface waves on the internal tides. We hope to combine stochastic forcing and stochastic initialization to study the separate and combined influences of sub-grid scale eddies, external wind-driven surface waves, and tidal forcing. We have begun work towards this goal by implementing modifications for stochastic forcing to the DO equations for a variable barotropic tide

$F_{u_{tide}} = U_0(\Omega)\omega\cos(\omega t)$ in Equations 5.1c(a-c).

$$\frac{\partial \bar{\boldsymbol{u}}(\boldsymbol{x},t)}{\partial t} = E^{\Omega}[\mathcal{L}] + \begin{bmatrix} E^{\Omega}[U_0]\omega\cos(\omega t) \\ 0 \\ 0 \end{bmatrix} \tag{5.1a}$$

$$\begin{aligned} \frac{\partial \boldsymbol{u_i}(\boldsymbol{x},t)}{\partial t} = C^{-1}_{Y_iY_j}\left(E^{\Omega}[\mathcal{L}Y_j] + \begin{bmatrix} E^{\Omega}[U_0Y_j]\omega\cos(\omega t) \\ 0 \\ 0 \end{bmatrix} \right) - \ldots \\ C^{-1}_{Y_iY_j}\left\langle E^{\Omega}[\mathcal{L}Y_j] + \begin{bmatrix} E^{\Omega}[U_0Y_j]\omega\cos(\omega t) \\ 0 \\ 0 \end{bmatrix}, \boldsymbol{u_k} \right\rangle \boldsymbol{u_k} \end{aligned} \tag{5.1b}$$

$$\frac{dY_i(t;\Omega)}{dt} = \left\langle \mathcal{L} - E^{\Omega}[\mathcal{L}] + \begin{bmatrix} (U_0 - E^{\Omega}[U_0])\omega\cos(\omega t) \\ 0 \\ 0 \end{bmatrix}, \boldsymbol{u_i} \right\rangle \tag{5.1c}$$

The DO methodology can allow us to efficiently study the sensitivity of internal tides to variations in the stratification and the associated interactions between internal tides and internal gravity waves, variation in the tidal velocity, and variations in other external forcings such as surface waves.

Appendix A

The Projection Method

Here we discuss the projection method utilized in the z-coordinate and σ-coordinate code, and discuss the associated restriction on the σ-coordinate Laplacian operator. The method used is a Chorin-Temam projection method Chorin (1968). We start with the temporally discretized system of equations A.1 to A.3: continuity, conservation of momentum, and tracer-convection, respectively. We treat the nonlinear and source terms explicitly and the linear terms implicitly.

$$\nabla \cdot \vec{u}^k = \nabla \cdot \vec{u}^{k+1} = 0 \tag{A.1}$$

$$\frac{\vec{u}^{k+1} - \vec{u}^k}{\Delta t} + (\nabla \cdot (\vec{u}\vec{u}))^k = -\frac{1}{\rho}\nabla p^k + \nu \nabla \cdot (\nabla \vec{u}^{k+1}) + F^k \tag{A.2}$$

$$\frac{\rho^{k+1} - \rho^k}{\Delta t} + (\nabla \cdot (\vec{u}\rho))^k = \kappa \nabla \cdot (\nabla \rho^{k+1}) \tag{A.3}$$

In the first step of this fractional-step method, a predictor velocity ($\vec{u_*}^{k+1}$) is computed using the pressure field from the previous timestep.

$$\frac{\vec{u}_*^{k+1} - \vec{u}^k}{\Delta t} + (\nabla \cdot (\vec{u}\vec{u}))^k = -\frac{1}{\rho}\nabla p^k + \nu \nabla \cdot (\nabla \vec{u}_*^{k+1}) + F^k \tag{A.4}$$

$$\begin{aligned} \frac{\vec{u}^{k+1} - \vec{u}^k}{\Delta t} - \frac{\vec{u}_*^{k+1} - \vec{u}^k}{\Delta t} + (\nabla \cdot \vec{u}\vec{u})^k - (\nabla \cdot \vec{u}\vec{u})^k &= -\frac{1}{\rho}\nabla p^{k+1} - -\frac{1}{\rho}\nabla p^k + \ldots \\ \nu \nabla^2 \vec{u}^{k+1} - \nu \nabla^2 \vec{u}_*^{k+1} + F^k - F^k & \end{aligned} \tag{A.5}$$

$$\frac{\vec{u}^{k+1} - \vec{u}_*^{k+1}}{\Delta t} = -\frac{1}{\rho}\nabla p^{k+1} + \frac{1}{\rho}\nabla p^k + \nu(\nabla^2 \vec{u}^{k+1}) - \nu(\nabla^2 \vec{u}_*^{k+1}) \tag{A.6}$$

We recall the vector ID, $\nabla^2\vec{v} = \nabla(\nabla\cdot\vec{v}) - \nabla\times(\nabla\times\vec{v})$, and substitute for the diffusion term.

$$\frac{\vec{u}^{k+1}-\vec{u}_*^{k+1}}{\Delta t} = -\frac{1}{\rho}\nabla p^{k+1}+\frac{1}{\rho}\nabla p^k+\nu[\nabla(\nabla\cdot\vec{u}^{k+1})-\nabla\times(\nabla\times\vec{u}^{k+1})]-\nu[\nabla(\nabla\cdot\vec{u}_*^{k+1})-\nabla\times(\nabla\times\vec{u}_*^{k+1})] \tag{A.7}$$

We make the approximation $\nabla\times(\nabla\times\vec{u}^{k+1}) \approx \nabla\times(\nabla\times\vec{u}_*^{k+1})$.

$$\frac{\vec{u}^{k+1}-\vec{u}_*^{k+1}}{\Delta t} = -\frac{1}{\rho}\nabla p^{k+1}+\frac{1}{\rho}\nabla p^k+\nu\nabla(\nabla\cdot\vec{u}^{k+1})-\nu\nabla(\nabla\cdot\vec{u}_*^{k+1}) \tag{A.8}$$

We compute the pressure-correction ($q = \frac{1}{\rho}p^{k+1} - \frac{1}{\rho}p^k - \nu(\nabla\cdot\vec{u}^{k+1}) + \nu(\nabla\cdot\vec{u}_*^{k+1})$) by taking the divergence of A.8, and this is the step where we see the restriction imposed on the σ-coordinate Laplacian operator.

$$\nabla\cdot\left(\frac{\vec{u}^{k+1}-\vec{u}_*^{k+1}}{\Delta t}\right. = -\frac{1}{\rho}\nabla p^{k+1}+\frac{1}{\rho}\nabla p^k+\nu\nabla(\nabla\cdot\vec{u}^{k+1})-\nu\nabla(\nabla\cdot\vec{u}_*^{k+1})) \tag{A.9}$$

$$\frac{-\nabla\cdot\vec{u}_*^{k+1}}{\Delta t} = -\frac{1}{\rho}\nabla\cdot\nabla p^{k+1}+\frac{1}{\rho}\nabla\cdot\nabla p^k+\nabla\cdot\nabla\nu(\nabla\cdot\vec{u}^{k+1}-\nabla\cdot\nabla\nu(\nabla\cdot\vec{u}_*^{k+1}) \tag{A.10}$$

$$\frac{-\nabla\cdot\vec{u}_*^{k+1}}{\Delta t} = \nabla\cdot\nabla[-\frac{1}{\rho}p^{k+1}+\frac{1}{\rho}p^k+\nu(\nabla\cdot\vec{u}^{k+1})-\nu(\nabla\cdot\vec{u}_*^{k+1})] = \nabla\cdot\nabla(-q) \tag{A.11}$$

Mathematically, the divergence operator in red is equivalent to the divergence operations in black. However, since the same mathematical operators can be constructed with different numerical schemes, each red divergence operator needs to be numerically equivalent to guarantee conservation. This is why it is important that Equation 3.17 is numerically identical to Equations 3.27 and 3.28.
Next, we solve for (q) and compute the corrector velocity (u^{k+1}).

$$\nabla\cdot\nabla(-q) = \frac{-\nabla\cdot\vec{u}_*^{k+1}}{\Delta t} \tag{A.12}$$

$$\frac{\vec{u}^{k+1}-\vec{u}_*^{k+1}}{\Delta t} = -\frac{1}{\rho}\nabla p^{k+1}+\frac{1}{\rho}\nabla p^k+\nu\nabla(\nabla\cdot\vec{u}^{k+1})-\nu\nabla(\nabla\cdot\vec{u}_*^{k+1}) \tag{A.13}$$

$$\frac{\vec{u}^{k+1}-\vec{u}_*^{k+1}}{\Delta t} = \nabla q \tag{A.14}$$

$$\vec{u}^{k+1} = -\vec{u}_*^{k+1} - \nabla q\Delta t \tag{A.15}$$

Finally, we correct the pressure with the corrector velocity. Equation A.16 is an incremental, rotational pressure-correction scheme.

$$\frac{1}{\rho}p^{k+1} = q + \frac{1}{\rho}p^{k} + \nu\nabla(\nabla\cdot\vec{u}^{k+1}) - \nu(\nabla\cdot\vec{u}_{*}^{k+1}) \tag{A.16}$$

Bibliography

Adcroft, A., Hill, C., and Marshall, J. (1997). Representation of topography by shaved cells in a height coordinate ocean model. *Monthly Weather Review*, 125(9):2293–2315.

Alford, M. H., Peacock, T., MacKinnon, J. A., Nash, J. D., Buijsman, M. C., Centuroni, L. R., Chao, S.-Y., Chang, M.-H., Farmer, D. M., Fringer, O. B., et al. (2015). The formation and fate of internal waves in the south china sea. *Nature*, 521(7550):65–69.

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

Arakawa, A. and Lamb, V. R. (1977). Computational design of the basic dynamical processes of the ucla general circulation model. *Methods in computational physics*, 17:173–265.

Berntsen, J. (2002). Internal pressure errors in sigma-coordinate ocean models. *Journal of atmospheric and oceanic technology*, 19(9):1403–1414.

Blumberg, A. F. and Mellor, G. L. (1987). A description of a three-dimensional coastal ocean circulation model. *Three-dimensional coastal ocean models*, pages 1–16.

Chen, C., Liu, H., and Beardsley, R. C. (2003a). An unstructured grid, finite-volume, three-dimensional, primitive equations ocean model: application to coastal ocean and estuaries. *Journal of atmospheric and oceanic technology*, 20(1):159–186.

Chen, C., Liu, H., and Beardsley, R. C. (2003b). An unstructured grid, finite-volume, three-dimensional, primitive equations ocean model: Application to coastal ocean and estuaries. *American Meteorological Society*, 20:159–186.

Chorin, A. J. (1968). Numerical solution of the navier-stokes equations. *Mathematics of computation*, 22(104):745–762.

Courant, R., Friedrichs, K., and Lewy, H. (1967). On the partial difference equations of mathematical physics. *IBM J. Res. Dev.*, 11:215–234.

Cushman-Roisin, B. and Beckers, J.-M. (2011). *Introduction to Geophysical Fluid Dynamics*. Elsevier, Waltham,MA, 2 edition.

Ferziger, J. and Peric, M. (2002). *Computational Methods for Fluid Dynamics.* 3 edition.

Fringer, O. B., Gerritsen, M., and Street, R. L. (2006). An unstructured-grid, finite-volume, non-hydrostatic, parallel coastal ocean simulator. *Ocean Modeling*, 14:139/173.

Gallus Jr, W. A. and Klemp, J. B. (2000). Behavior of flow over step orography. *Monthly weather review*, 128(4):1153–1164.

Gary, J. M. (1973). Estimate of truncation error in transformed coordinate, primitive equation atmospheric models. *Journal of the Atmospheric Sciences*, 30(2):223–233.

Grell, G., Dudhia, J., and Stauffer, D. (1995). A description of hte fifth-generation penn state/ncar mesoscale model (mm5). *NCAR Technical Notes (User Guide).*

Haley, Jr., P. J., Agarwal, A., and Lermusiaux, P. F. J. (2015). Optimizing velocities and transports for complex coastal regions and archipelagos. *Ocean Modeling*, 89:1–28.

Haley, Jr., P. J. and Lermusiaux, P. F. J. (2010). Multiscale two-way embedding schemes for free-surface primitive equations in the "Multidisciplinary Simulation, Estimation and Assimilation System". *Ocean Dynamics*, 60(6):1497–1537.

Helfrich, K. and Grimshaw, R. (2008). Nonlinear disintegration of the internal tide. *Journal of Physical Oceanography*, 38:686–701.

Kanarska, Y. and Maderich, V. (2003). A non-hydrostatic numerical model for calculating free-surface stratified flows. *Ocean Dynamics*, 53(3):176–185.

Khatiwala, S. (2003). Generation of internal tides in an ocean of finite depth: analytical and numerical calculations. *Deep-Sea Research*, 1:3–21.

Legg, S. and Huijts, K. M. (2006). Preliminary simulations of internal waves and mixing generated by finite amplitude tidal flow over isolated topography. *Deep-Sea Research*, 2:140–156.

Lin, S.-J. (1997). A finite-volume integration method for computing pressure gradient force in general vertical coordinates. *Quarterly Journal of the Royal Meteorological Society*, 123(542):1749–1762.

Lolla, T. and Lermusiaux, P. F. J. (2016a). A Gaussian–Mixture Model Smoother for Continuous Nonlinear Stochastic Dynamical Systems. *Monthly Weather Review.* Submitted.

Lolla, T. and Lermusiaux, P. F. J. (2016b). A Gaussian–Mixture Model Smoother for Continuous Nonlinear Stochastic Dynamical Systems: Applications. *Monthly Weather Review.* Submitted.

Lorenz, E. N. (1955). Available potential energy and the maintenance of the general circulation. *Tellus*, 7(2):157–167.

Ma, G., Shi, F., and Kirby, J. T. (2012). Shock-capturing non-hydrostatic model for fully dispersive surface wave processes. *Ocean Modelling*, 43:22–35.

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

Mellor, G. L. (1998). *Users guide for a three dimensional, primitive equation, numerical ocean model.* Program in Atmospheric and Oceanic Sciences, Princeton University Princeton, NJ 08544-0710.

Mellor, G. L., Ezer, T., and Oey, L.-Y. (1994). The pressure gradient conundrum of sigma coordinate ocean models. *Journal of atmospheric and oceanic technology*, 11(4):1126–1134.

Mesinger, F. (1982). On the convergence and error problems of the calculation of the pressure gradient force in sigma coordinate models. *Geophysical & Astrophysical Fluid Dynamics*, 19(1-2):105–117.

Munk, W. and Wunsch, C. (1998). Abyssal recipes ii: Energetics of tidal and wind mixing. *Deep Sea Research Part I: Oceanographic Research Papers*, 45(12):1977–2010.

Nikurashin, M. and Ferrari, R. (2013). Overturning circulation driven by breaking internal waves in the deep ocean. *Geophysical Research Letters*, 40(12):3133–3137.

Sapsis, T. P. (2011). *Dynamically Orthogonal Field Equations for Stochastic Fluid Flows and Particle Dynamics.* PhD thesis, Massachusetts Institute of Technology, Department of Mechanical Engineering, Cambridge, MA.

Sapsis, T. P. and Lermusiaux, P. F. J. (2009). Dynamically orthogonal field equations for continuous stochastic dynamical systems. *Physica D: Nonlinear Phenomena*, 238(23–24):2347–2360. doi:10.1016/j.physd.2009.09.017.

Sapsis, T. P. and Lermusiaux, P. F. J. (2012). Dynamical criteria for the evolution of the stochastic dimensionality in flows with uncertainty. *Physica D: Nonlinear Phenomena*, 241(1):60–76.

Shaw, J. and Weller, H. (2016). Comparison of terrain following and cut cell grids using a non-hydrostatic model. *Monthly Weather Review*, (2016).

Shea, R. E. and Broenkow, W. W. (1982). The role of internal tides in the nutrient enrichment of monterey bay, california. *Estuarine, Coastal and Shelf Science*, 15(1):57–66.

Sondergaard, T. and Lermusiaux, P. F. J. (2013a). Data assimilation with Gaussian Mixture Models using the Dynamically Orthogonal field equations. Part I: Theory and scheme. *Monthly Weather Review*, 141(6):1737–1760.

Sondergaard, T. and Lermusiaux, P. F. J. (2013b). Data assimilation with Gaussian Mixture Models using the Dynamically Orthogonal field equations. Part II: Applications. *Monthly Weather Review*, 141(6):1761–1785.

Song, Y. T. (1998). A general pressure gradient formulation for ocean models. part i: Scheme design and diagnostic analysis. *Monthly Weather Review*, 126(12):3213–3230.

Subramani, D. and Lermusiaux, P. F. J. (2016). Energy-optimal path planning by stochastic dynamically orthogonal level-set optimization. *Ocean Modeling*.

Tseng, Y.-H. and Ferziger, J. H. (2003). A ghost-cell immersed boundary method for flow in complex geometry. *Journal of computational physics*, 192(2):593–623.

Ueckermann, M. P. and Lermusiaux, P. F. J. (2012). 2.29 Finite Volume MATLAB Framework Documentation. MSEAS Report 14, Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA.

Ueckermann, M. P., Lermusiaux, P. F. J., and Sapsis, T. P. (2013). Numerical schemes for dynamically orthogonal equations of stochastic fluid and ocean flows. *Journal of Computational Physics*, 233:272–294.

Wang, Y.-H., Dai, C.-F., and Chen, Y.-Y. (2007). Physical and ecological processes of internal waves on an isolated reef ecosystem in the south china sea. *Geophysical Research Letters*, 34(18).

Winters, K. B., Lombard, P. N., Riley, J. J., and D'Asaro, E. A. (1995). Available potential energy and mixing in density-stratified fluids. *Journal of Fluid Mechanics*, 289:115–128.