Topics in Numerical Ocean Simulation

by

B. Gazi YILDIRIM

M.Sc., Applied Mathematics, Brown University; Providence, RI, 2008
M.Sc., Aerospace Engineering, Mississippi State University; Starkville, MS, 2003
B.Sc., Aeronautical Engineering, Istanbul Technical University; Turkey, 2000

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in The Division of Engineering at Brown University

PROVIDENCE, RHODE ISLAND

May 2012
This dissertation by B. Gazi YILDIRIM is accepted in its present form by The Division of Engineering as satisfying the dissertation requirement for the degree of Doctor of Philosophy.

Date________________________

George Em Karniadakis, Ph.D., Advisor

Recommended to the Graduate Council

Date________________________

Martin Maxey, Ph.D., Reader

Date________________________

Kenneth Breuer, Ph.D., Reader

Approved by the Graduate Council

Date________________________

Peter M. Weber, Dean of the Graduate School
Acknowledgements

I would like to thank to my advisor Professor George Em Karniadakis for giving me opportunity to work with his group and for fixing many wrong decisions of my life during this study. I also thank to my Ph.D. dissertation readers Professor Martin Maxey and Kenny Breuer for useful feedbacks and discussions.

Professor Changsheng Chen provided their numerical simulation output and introduced action balance equation to me. I am grateful to him for his contribution to this dissertation work. Dr. Nico Booij helped me to understand numerical ocean wave better and correct my misunderstandings. I am indebted to him for his help. I also want to thank to Prof. Rainald Lohner for supporting my decision to study at Brown University. Martin Hunt helped me to use PUQ for the third part of my thesis. I truly thank to him.

I thank to my friends Cagatay Demiralp, Hatice Sahinoglu and Momotazur (Shantu) and Mila Rahmnan for their true friendship. I worked on my dissertation and lived in West Lafayette, IN for last three years. I thank to Selma and Turkay Yolcu, Derya and Rafet Malta for their support during my work at Purdue university. I also thank to Francesca Randazzo for her friendship.

I would like to thank to my parents Zeynep and Celil Yildirim for their unconditional support and love. I dedicated this dissertation to my parents. I thank to my sister Mehtap Yildirim. My wife Yesim Yildirim was patient with me and kept me
going on, without her support I would not be able to finish this dissertation work. I have baby daughter Elif Ecrin born in September 2010, she tried all tricks to stop my dissertation. In spite of this, this thesis is dedicated to her along with my parents. I also thank to parents of my wife, especially for Fatma Serin who was taking care of Elif and us during the last part of my study.
We investigate two primary aspects of numerical simulations for the oceans. The first is the degree of spatial complexity of the dynamics in regional ocean models. Using simulation results from three different regional ocean models (HOPS, ROMS and FVCOM) we show that just a few spatio-temporal POD (proper orthogonal decomposition) modes are sufficient to describe the most energetic ocean dynamics. In particular, we demonstrate this with the simulated ocean dynamics for the New Jersey coast, Massachusetts Bay and Gulf of Maine. Moreover, the extrema of the POD spatial modes are very good locations for sensor placement and accurate field reconstruction. We employ a modified POD theory to incorporate a limited number of measurements in reconstructing the velocity and temperature fields, and we study systematically the corresponding reconstruction errors as a function of the sensor location, number of sensors, and number of POD modes. This new approach is quite accurate in short-term simulation, and hence it has the potential of accelerating the use of real-time adaptive sampling in data assimilation for ocean forecasting.

The second theme is the accurate prediction of surface gravity waves in the ocean. We develop a new, high-order, hybrid discretization of the phased-averaged (action balance) equation to simulate ocean waves. We employ Discontinuous Galerkin (DG) discretization on an unstructured grid of the geophysical space and Fourier-collocation for the wave number direction and frequency coordinates. The original action balance equation is modified to facilitate absorbing boundary conditions in the frequency domain. This modification enforces periodicity at the frequency boundaries so that the fast convergence of Fourier-collocation still holds. In addition, a mapping of the wave number directional coordinate is introduced to cluster the collocation points about the sharply defined, directional spectra. Time-discretization
is accomplished by a TVD Runge-Kutta scheme. The overall convergence of the scheme is exponential (spectral). We successfully verified and validated the method against several analytical solutions, observational data, and experimental results.

The last part of this study uses a generalized polynomial chaos (gPC) framework to quantify the uncertainty of the ocean wave simulations. To this end, we adopted the collocation stochastic method. The collocation points in the random space were chosen from the Smolyak grid. This assured us substantial reduction in the number of samples for the large random variables. We considered two random inputs: (1) the activated source term parameters and (2) the current field provided from the experiment. We introduced a random perturbation current field from a Karhunen-Loeve expansion of the Gauss correlation function. The uniform distribution probability function was chosen for all random inputs. We finally quantified the uncertainty first for each individual contribution (from either the random variables associated with source term parametrizations or the random process of the current field) and then evaluated the uncertainty with both random sources.
## Contents

Acknowledgments

1 Motivation and Objective

2 Data simulation and POD
   2.1 Ocean Circulation Models: ROMS, HOPS, and FVCOM
   2.2 POD Formulation
   2.3 POD-Method of Snapshot
   2.4 Simulations and Low-dimensionality
      2.4.1 New Jersey and Middle Atlantic Bight
      2.4.1.1 The Snapshot Length Effect
      2.4.2 Massachusetts Bay
      2.4.3 Gulf of Maine
      2.4.4 The “numerical” ocean is low-dimensional
   2.5 POD-based Reconstruction
      2.5.1 Unsteady flow past a cylinder
   2.6 Massachusetts Bay: Results and Discussion
      2.6.1 Total velocity
      2.6.2 Temperature
   2.7 Summary and Discussion

3 Numerical Ocean Wave
   3.1 Action Balance Equation
   3.2 Modeling the Source Terms
      3.2.1 Source generation by wind ($S_{in}$)
      3.2.2 Dissipation by white-capping ($S_{wc}$)
      3.2.3 Nonlinear wave-wave interactions ($S_{nl}$)
         3.2.3.1 Quadruplet wave-wave interactions ($S_{nl4}$)
3.2.3.2 Triad wave-wave interactions \((S_{nl3})\) .................................. 80
3.2.4 Depth-induced breaking \((S_{br})\) ........................................... 83
3.2.5 Bottom friction \((S_{bfr})\) .................................................. 86

4 Hybrid Spectral/DG Method for Phase-averaged Equation ........................................ 89
4.1 Space Discretizations .......................................................... 90
4.1.1 Time Discretization ......................................................... 102
4.1.2 Forward and Backward Transformations .............................. 104
4.1.3 Mapping in the Spectral Direction ..................................... 106
4.1.4 Absorbing Boundary Treatment in Frequency Direction .......... 111

5 Verification and Validation ....................................................... 115
5.1 Verification of Temporal and Spatial Accuracy .......................... 116
5.1.1 Temporal Accuracy Verification ........................................ 117
5.1.2 Spatial Accuracy Verification .......................................... 117
5.2 Verification and Validation Cases ......................................... 121
5.2.1 Case I: Numerical diffusion in deep water \((S = 0)\) ................. 122
5.2.2 Case II: Current-induced shoaling and retractions in deep wa-
ter \((S = 0)\) ................................................................................. 126
5.2.3 Case III: Depth-induced shoaling and refractions in shallow
water \((S = 0)\) ................................................................................. 130
5.2.4 Case IV: Duration-limited growth in deep water \((S = S_{in} +
S_{nl4} + S_{we})\) ............................................................................. 133
5.2.5 Case V: Fetch-limited growth in deep water \((S = S_{in} +S_{nl4} + S_{we})\) 136
5.2.6 Case VI: Depth-induced wave breaking \((S = S_{br})\) ................ 138
5.2.7 Case VII: Flume test case \((S = S_{b,fr} + S_{br} + S_{nl3})\) ............ 141
5.2.8 Case VIII: HISWA Experiment \((S = S_{b,fr} + S_{br} + S_{nl3})\) .... 144
5.3 Results and discussion ........................................................... 152

6 Uncertainty Quantification for Numerical Ocean Wave Prediction 156
6.1 Uncertainty in source terms ..................................................... 159
6.1.1 Sensitivity analysis of source term parameters ....................... 159
6.1.2 Results and discussion ...................................................... 164
6.2 Uncertainty in current field ..................................................... 171
6.2.1 Results and discussion ...................................................... 174
6.3 Uncertainty of source and current field ................................. 179
6.3.1 Results and discussion ...................................................... 181
6.4 Summary and Conclusions .................................................. 187

7 Summary and Future Works ....................................................... 190

A Spectral Methods for Discontinuous and Sharp-Gradient Problems 195
A.1 Introduction ................................................................. 196
A.2 Chebyshev Collocation Methods ........................................ 197
A.2.1 Filtering ................................................................. 199
A.2.2 Coordinate Transformation ............................................ 202
A.2.3 Time Integration ....................................................... 205
A.3 Numerical results ......................................................... 206
A.3.1 The one-dimensional viscous Burger equation ................. 206
A.3.2 The two-dimensional inviscid and viscous Burger Equations . 210

A.3.2.1 Spectral convergence test case ................................. 210
A.3.2.2 Circular shock test case ........................................ 212
A.3.3 Euler equations test case (Shock tube problem) .............. 214
A.4 Summary ......................................................................... 217

B Sea Spectrums ..................................................................... 224
B.1 Pierson-Moskowitz Spectrum ........................................... 225
B.2 JONSWAP (Joint North Sea Wave Project) Spectrum .......... 225
B.3 Breugem and Holthuijsen Spectrum ................................ 226

C The Integrated (Mean) Parameters ...................................... 227
C.1 Significant Wave Height ($H_s$) ......................................... 228
C.2 The Mean Absolute Wave Period ($T_{m01}$) ......................... 228
C.3 The Mean Direction ($\bar{\theta}$) .......................................... 228
List of Tables

2.1 Mean (First) and subsequent eight eigenmodes for New Jersey Coast ($^1 = 2.5$ days, $^2 = 25$ days), Massachusetts Bay ($^1 = 8$ days, $^2 = 47$ days) and Gulf of Maine simulations. ................................. 36
2.2 First eight eigenmodes for New Jersey Coast ($^1 = 2.5$ days, $^2 = 25$ days), Massachusetts Bay ($^1 = 8$ days, $^2 = 47$ days) and Gulf of Maine simulations. ........................................ 37
2.3 Sensor location for 12 sensors. In case $1 \mathbf{x}_i^j$ denotes the coordinate of mode $j$ with $i^{th}$ extrema, with 1 corresponding to largest extremum; the contours of the POD modes of total velocity $U = \sqrt{u^2 + v^2}$ are employed for identifying the extrema. ............................................. 45
2.4 Errors $e_1$ (w.r.t. full simulation) and $e_2$ (w.r.t. POD reconstruction) of the total velocity $U = \sqrt{(u^2 + v^2)}$. ......................................................... 46
2.5 Sensor configurations for Mass Bay. In the table-boxes, N x means N times x, with x a multiplier adjusted for the total number of sensors. 48

5.1 $L_2$ error for RK2, RK3, and RK4 at time = 1.0. $12^{th}$ order Jacobi polynomials (in geographical space) and 8 Fourier-collocations (in spectral space) are used. ......................................................... 117
5.2 $L_2$ and $L_\infty$ errors and convergence rates ($N_\theta = 24$, $N_\sigma = 24$, $\Delta t = 0.001$) between meshes (h/2) and (h/4). The errors are integrated over simulation time $t \in [0, 1]$ using the RK4 scheme. The rates hit either/time-related or/spectral space-related discretization. 121
5.3 Case IV: Activated source parameters in Duration-limited growth(left) and Fetch-limited growth(right) test cases. ......................................................... 135

6.1 The required number of random space dimension for the various correlation lengths 30, 25, and 20. ................................................................. 174
6.2 The required runs from the Smolyak grids for the random process current field. The runs 8801 and 30801 (red circled) did not take place in any cases and they are given here for completeness. .......... 174
List of Figures

2.1 New Jersey coast: Contours of the first POD mode (dimensionless temperature) and domain. ............................................. 23
2.2 New Jersey coast: Energy spectra of horizontal velocity vector \((u, v)\), temperature and salinity for 2.5-days (left) and 25-days (right) simulations. ......................................................... 24
2.3 New Jersey coast: Time resolution effects on the energy spectra of horizontal velocity \((u, v)\) vector for 2.5-days (left) and 25-days (right) simulations. ................................................................. 24
2.4 Time resolution effect on POD temporal modes of horizontal velocity \((u, v)\) vector for a 2.5 day simulation of the New Jersey coast. . . . . 25
2.5 Time resolution effect on POD temporal modes of horizontal velocity \((u, v)\) vector for a 25 day simulation of the New Jersey coast. . . . . 26
2.6 The simulation length (in time) effect on energy spectra of horizontal velocity \((u, v)\) vector for a 2.5 day and 25 day simulation of the New Jersey coast. The left figure represents short-time and the right one does long-time simulations. ......................................................... 27
2.7 The snapshot length effect on POD temporal modes of horizontal velocity \((u, v)\) vector for a 2.5 day simulation of the New Jersey coast. 28
2.8 The snapshot length effect on POD temporal modes of horizontal velocity \((u, v)\) vector for a 25 day simulation of the New Jersey coast. 29
2.9 Mass Bay: Contours of the first POD mode (dimensionless temperature) and domain. .......................................................... 30
2.10 Mass Bay: Energy spectra of horizontal velocity vector \((u, v)\), temperature and salinity for 8-days (left) and 47-days simulations. . . . 31
2.11 Gulf of Maine: Contours of the first POD mode (temperature) and domain. ............................................................... 32
2.12 Gulf of Maine. Left: Energy spectra of horizontal velocity vector \((u, v)\), temperature and salinity for Gulf Of Maine. Right: Energy spectra of horizontal velocity vector at two different grid resolutions. 33
2.13 Comparison of eigenspectra for the turbulent wake and the long-term ocean dynamics of three regions. ................................. 35
2.14 Comparison of eigenspectra for the turbulent wake and the short-term and long-term ocean dynamics of Mass Bay. ....................... 38
2.15 Computational domain for flow past a cylinder. ............................ 42
2.16 Contours of first four POD modes of the total velocity $U$ and sensor locations for case 1. Mode 1: upper-left; Mode 2: upper-right; Mode 3: lower-left; Mode 4: lower-right. 

2.17 Contours of streamwise velocity $u$ and sensor locations for case 3. 

2.18 Mass Bay: Schematic of best sensor locations for Case 1 (see Table 2.5: 4 modes, 12 sensors). Only a slice at the surface of the first POD mode is shown; different symbols correspond to different modes. The extrema are located at some depth from the surface. The contours represent the second POD mode of the temperature. 

2.19 Short-term simulation, 4-modes: Time averaged errors for total velocity. The total of four modes contain 92.6% of total energy and the corresponding truncation error is $\tilde{e}_1 = 8.5\%$. For 12 sensors $\kappa(M) = 14.82, 10.54, 9.55$ for Cases 1, 2 and 3, respectively. 

2.20 Short-term simulation, 6-modes: Time averaged errors for total velocity. The total of six modes contain 96.1% of total energy and the corresponding truncation error is $\tilde{e}_1 = 6.2\%$. For 24 sensors $\kappa(M) = 18.92, 17.74, 16.54$ for Cases 1, 2 and 3, respectively. 

2.21 Short-term simulation, 8-modes. Time averaged errors for total velocity. The total of eight modes contain 98.1% of total energy and the truncation error is $\tilde{e}_1 = 4.3\%$. For 40 sensors $\kappa(M) = 19.15, 20.37, 19.94$ for Cases 1, 2 and 3, respectively. 

2.22 Long-term simulation, 4-modes. Time averaged errors for total velocity. The total of four modes contain 53.2% of total energy and the truncation error is $\tilde{e}_1 = 31.7\%$. For 12 sensors $\kappa(M) = 20.73, 11.33, 7.10$ for Cases 1, 2 and 3, respectively. 

2.23 Long-term simulation, 16-modes. Time averaged errors are for total velocity. The total of sixteen modes contain 83.1% of total energy and the truncation error is $\tilde{e}_1 = 19.3\%$. For 144 sensors $\kappa(M) = 13.42, 13.92, 11.78$ for Cases 1, 2 and 3, respectively. 

2.24 Comparison of original (left) and reconstructed (right) total velocity field for the 24th snapshot in the long-term simulation. 

2.25 Time averaged errors for temperature; 8-modes. Both short-term and long-term results are shown using 40 sensors. 

2.26 Comparison of original (left) and reconstructed (right) temperature field for the 36th snapshot in the short-term simulation. 

3.1 Linear wave theory basic equation and boundary conditions taken from [82]. 

3.2 The classification of waves by Kinsman (reproduced with permission from [92]). 

3.3 A description of random sea by summation of many harmonic waves (from Pierson et al. [118]). 

3.4 (a) An illustration of geophysical and spectral spaces. The frequency direction ($\sigma = 2\pi f$) is truncated from $[0, \infty)$ to $[0, 2.5]$ Hz and the directional coordinate ($\theta$) covers full circle $[0, 2\pi]$. (b) An illustration of coordinates $m$ and $s$. 

3.5 Wind source term for the JONSWAP spectrum. 

3.6 The whitecapping source term ($C_{wc} = 2.36 \times 10^{-5}$, $n = 0$ and $p = 4$) for the JONSWAP spectrum. 

3.7 The nonlinear (Quadruplet) source term ($\lambda = 0.25$ and $C_{nl4} = 3.0 \times 10^7$) computed via DIA method for the JONSWAP spectrum.
3.8 The nonlinear interaction (Triad) term ($\alpha_{BE} = 1.0$) for the JONSWAP spectrum. .......................................................... 84
3.9 The depth-induced breaking source term ($\alpha_{BJ} = 1.0, \lambda = 0.73$) for the JONSWAP spectrum. .......................................................... 87
3.10 The bottom friction source term ($C_{bfr} = 0.067$) for the JONSWAP spectrum. .......................................................... 88

4.1 Transformation from the standard triangle to rectangular region from [91]. .......................................................... 96
4.2 (a) The derivative of a $\cos^{500}(\theta)$ directional spreading ($A_1 = 1$) where the clustered region is confined to the interval $[-7.5^\circ, 7.5^\circ]$ which only makes up 4% of whole domain $2\pi$. (b) Pointwise errors for standard Fourier-collocation, (c) Mapped Fourier-collocation with ($L = 2\sigma_\theta$), and (d) with ($L = 4\sigma_\theta$) (d) for various resolutions. The spectral width for $\cos^{500}(\theta)$ is $2.5^\circ$. .......................................................... 110
4.3 The evolution of the JONSWAP spectrum for non-reflecting boundary conditions. The analytical solution (—) and numerical solution (− − −) are compared at $t = 0, 1.5, 1.625, 1.75, 1.8, 1.9$. .......................................................... 113

5.1 (a) Mesh used in Runge-Kutta (RK) discretization verifications. (b)(c)(d) The sequence of meshes used in verification of space discretizations, respectively, (h,h/2,h/4) where h denotes the side of triangle and is the length of $\pi$. .......................................................... 118
5.2 $L_2$ errors for for the sequence of meshes (h,h/2,h/4) for domain $[-\pi, \pi] \times [-\pi, \pi]$. Frequency ($\sigma$) and Directional ($\sigma$) collocation resolutions 4, 8, 16, 24 are tested against polynomial order P. The errors are integrated over simulation time $t \in [0, 1]$ with the SSP-RK4 scheme ($\Delta t = 0.001$). .......................................................... 120
5.3 Case I: Diffusion test meshes. .......................................................... 123
5.4 Significant wave heights (a) SWAN first order scheme (FD scheme, 10,000 grid points) (b) SWAN third order scheme (FD scheme, 10,000 pts) (c) FE-WAVE (about 20,000 triangles) (d) FVCOM-SWAVE (about 20,000 triangles) (reproduced with permission from [125]). .......................................................... 124
5.5 Case I: Significant heights ($H_s$) for a mesh (122 triangles) for two different polynomial orders. .......................................................... 124
5.6 Case I: Significant height ($H_s$) for a mesh (966 triangles) for three different polynomial orders. .......................................................... 125
5.7 Case II: A sketch of wave-current interaction setups for four test cases. The current velocity vector $V_C$ has a linear distribution from 0 to 2 m/s. .......................................................... 127
5.8 Case II: The computational mesh used for wave-current interaction tests. Fourth-order Jacobi polynomials ($p=4$) are employed in each element for all simulations. .......................................................... 128
5.9 Case II: Comparison of significant wave height $H_s$ versus distance from analytical equation and numerical simulations for all four cases of depth-induced shoaling and current-induced refraction (see the illustration of all cases in Figures 5.7a and 5.7b). .......................................................... 129
5.10 Case II: Analytical and numerical results are compared along the coordinate $x$ for cases (c) and (d) (see the illustration of the cases in Figure 5.7b). .......................................................... 129
5.11 Case III: Depth-induced shoaling and refraction test case. .......................................................... 131
5.12 Case III: The significant wave height ($H_s$) for case (a) $\bar{\theta}_0 = 0^\circ$ and case (b) $\bar{\theta}_0 = 30^\circ$. .......................................................... 131
5.13 Case III: The mean wave direction ($\bar{\theta}$) for case (a) $\bar{\theta}_0 = 0^\circ$ and case (b) $\bar{\theta}_0 = 30^\circ$.

5.14 Case IV: Dimensionless energy growth ($E^* = g^2 E_{tot}/u_s^4$) and time ($t^* = gt/u_s$) plot for the duration-limited wave growth. ECWAM (WAM modeling) and our numerical results are compared to the empirical fit (JKdV) by Jannsen et al. (1987) [89].

5.15 Case IV: Dimensionless energy ($E^* = g^2 E_{tot}/u_s^4$) growth curve versus wave age ($\chi^* = cp(t)/u_s$) are presented for duration-limited growth. Wave age from left to right show the young sea transition to old sea. The present numerical results (●●●) are compared to ECWAM results (●●●●) and the empirical data (JKdV) (——) compiled by Jannsen et al. [89].

5.16 Case V: The computational mesh plotted in logarithmic axis is used for the fetch-limited wave growth test case.

5.17 Case V: Energy growth curve versus fetch. $E^* = g^2 E_{tot}/u_s^4$ and $x^* = gx/u_s^2$. B&H (2006) empirical data fit and SWAMP Group (1985) and $u_s$ is friction velocity.

5.18 Case VI: Significant height variation in the Battjes and Janssen (1978) experiment. SWAN and the present simulation used $\gamma = 0.73$ and $\alpha_{BJ} = 1$ in the depth-induced wave breaking source term.

5.19 Case VII: The lateral view of the experimental setup (top) and computational mesh (below) is presented. The crosses (x) are experimental measurement locations. The mesh has only two elements and 18th-order of Jacobi polynomials were used over the elements.

5.20 Case VII: Significant height variations in the Wood et al. (2001) experiment; the SWAN and present code results are compared. Here $\gamma = 0.73$ and $\alpha_{BJ} = 1$ in the depth-induced wave breaking source term. The triad proportionality coefficient $\alpha_{EB} = 0.05$ and 1.0 are used if the triad source term is activated.

5.21 The computational mesh (48 triangle elements) used for the HISWA simulation. The dashed region is the experimental region.

5.22 The sensor locations and the bathymetry contour (left) and current vector field (right).

5.23 Comparison of significant heights ($H_s$) from three different polynomial orders (p=1, 3, 5) and experimental and SWAN results.

5.24 Comparison of mean wave periods ($T_{m01}$) from three different polynomial orders (p=1, 3, 5) and experimental and SWAN results.

5.25 Comparison of mean wave direction ($\bar{\theta}$) from three different polynomial orders (p=1, 3, 5) and experimental and SWAN results.

5.26 Comparison of energy spectra for three different polynomial orders (p=1, 3, 5), experimental, and SWAN results.

6.1 The elementary effect values for significant height ($H_s$) for four source parameters ($C_{bfr}$, $\alpha_{BJ}$, $\gamma$, $\alpha_{EB}$).

6.2 The elementary effect values for mean wave period ($T_{m01}$) for four source parameters ($C_{bfr}$, $\alpha_{BJ}$, $\gamma$, $\alpha_{EB}$).

6.3 The elementary effect values for mean wave direction ($\bar{\theta}$) for four source parameters ($C_{bfr}$, $\alpha_{BJ}$, $\gamma$, $\alpha_{EB}$).

6.4 The mean ($\mu$) and relative percentage of standard deviation ($\sigma$) and mean ratio of significant wave height ($H_s$) from three level expansions (2,3,4) for the four source parameters ($C_{bfr}$, $\alpha_{BJ}$, $\gamma$, $\alpha_{BJ}$).
6.5 The mean ($\mu$) and relative percentage of standard deviation ($\sigma$) and mean ratio of absolute mean wave period ($T_{m01}$) from three level expansions (2,3,4) for the four source parameters ($C_{bf}, \alpha_{BJ}, \gamma, \alpha_{BJ}$). 

6.6 The mean ($\mu$) and relative percentage of standard deviation ($\sigma$) and mean ratio of mean wave direction ($\bar{\theta}$) from three level expansions (2,3,4) for the four source parameters ($C_{bf}, \alpha_{BJ}, \gamma, \alpha_{BJ}$). 

6.7 Significant wave height $H_s$ of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the fourth order expansion for the source parameters ($C_{bf}, \alpha_{BJ}, \gamma, \alpha_{BJ}$). 

6.8 Mean Wave Period $T_{m01}$ of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the fourth order expansion for the source parameters ($C_{bf}, \alpha_{BJ}, \gamma, \alpha_{BJ}$). 

6.9 Mean Wave Direction $\bar{\theta}$ of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the fourth order expansion for the source parameters ($C_{bf}, \alpha_{BJ}, \gamma, \alpha_{BJ}$). 

6.10 Energy spectra of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the fourth order expansion for the source parameters ($C_{bf}, \alpha_{BJ}, \gamma, \alpha_{BJ}$). 

6.11 The comparison of relative percentages of standard deviation ($\sigma$) and mean ratio of significant wave height ($H_s$) for various correlation lengths (cor. len. = 30, 25, 20) used in the random process of the current field. All runs used level 3 from the Smolyak grid. 

6.12 The comparison of relative percentages of standard deviation ($\sigma$) and mean ratio of mean wave period ($T_{m01}$) for various correlation lengths (cor. len. = 30, 25, 20) used in the random process of the current field. All runs used level 3 from the Smolyak grid. 

6.13 The comparison of relative percentages of standard deviation ($\sigma$) and mean ratio of mean wave direction ($\bar{\theta}$) for various correlation lengths (cor. len. = 30, 25, 20) used in the random process of the current field. All runs used level 3 from the Smolyak grid. 

6.14 Significant wave height $H_s$ of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the third order expansion for the current parameters (1-5 modes, cor. len. = 25). 

6.15 Mean Wave Period $T_{m01}$ of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the third order expansion for the current parameters (1-5 modes, cor. len. = 25). 

6.16 Mean Wave Direction $\bar{\theta}$ of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the third order expansion for the current parameters (1-5 modes, cor. len. = 25). 

6.17 Energy spectra of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the third order expansion for the current parameters (1-5 modes, cor. len. = 25).
6.18 Significant wave height $H_s$ of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the third order expansion for the current parameters (1-5 modes, cor. len. = 25) and the source parameters ($C_{bf}$, $\alpha_{BJ}$, $\gamma$, $\alpha_{BJ}$).

6.19 Mean Wave Period $T_{m01}$ of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the third order expansion for the current parameters (1-5 modes, cor. len. = 25) and the source parameters ($C_{bf}$, $\alpha_{BJ}$, $\gamma$, $\alpha_{BJ}$).

6.20 Mean Wave Direction $\theta$ of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the third order expansion for the current parameters (1-5 modes, cor. len. = 25) and the source parameters ($C_{bf}$, $\alpha_{BJ}$, $\gamma$, $\alpha_{BJ}$).

6.21 Energy spectra of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the fourth order expansion for the current parameters (1-5 modes, cor. len. = 25).

A.1 Plots of the Chebyshev collocation solutions to the viscous Burger equation at different time stations ($\mu = 0.01, N = 128$).

A.2 Comparison of numerical and analytical steady-state solutions ($\mu = 0.1, t = 10$) with the Tal-Ezer mapping and an exponential filter (16th order) for various resolutions.

A.3 Point-wise errors of the numerical solutions with the Tal-Ezer mapping ($\mu = 0.1, t = 10$) and an exponential filter (16th order) for various resolutions.

A.4 Comparison of numerical and analytical steady-state solutions ($\mu = 0.001, t = 10$) with the Tal-Ezer mapping and an exponential filter (16th order) for various resolutions.

A.5 Point-wise errors of the numerical solutions with the Tal-Ezer mapping ($\mu = 0.001, t = 10$) and an exponential filter (16th order) for various resolutions.

A.6 A detailed comparison plot of the numerical ($\mu = 0.001, t = 10$) and exact solution.

A.7 Point-wise error plots of spectral convergence test cases ($\mu = 0.1$, $t=0.1$) along the coordinate $x(y = 0)$ axis.

A.8 Point-wise error plots of spectral convergence test cases ($\mu = 0.001$, $t=0.1$) along the coordinate $x(y = 0)$ axis.

A.9 Contours of the circular shock test case at time $t = 1.0$, (left) for a coarse ($128 \times 128$) grid and (right) for a fine ($256 \times 256$) grid. The Tal-Ezer mapping and an exponential filter (16th order) are used ($\mu = 0.0001, t=1$).

A.10 One-dimensional plot of the circular shock test case along the diagonal for different grid resolutions ($t = 0.2$).

A.11 Diagonal solutions (bottom-left to upper-right) for the circular shock test case ($\mu = 0.0001$) at different time stations ($t = 0.5, 5, 10, 100$) with the exponential filter of order 16 and the Tal-Ezer mapping.

A.12 Effect of filtering order $p$ on a fine grid ($256 \times 256$) at different times $t = 0.5, 2, 10, 20$ for the inviscid case.
A.13 The shock tube problem with the initial left state \( q_L(\rho, u, p) = \{1, 0, 1/\gamma\} \) and the initial right state \( q_R(\rho, u, p) = \{1/10, 0, 1/10\gamma\} \). Shown are comparisons of the density for numerical solutions and the exact solution at \( t = 0.25 \). The Tal-Ezer mapping and exponential filter \( (p = 16) \) are employed.

A.14 The shock tube problem with the initial left state \( q_L(\rho, u, p) = \{1, 0, 1/\gamma\} \) and the initial right state \( q_R(\rho, u, p) = \{1/10, 0, 1/10\gamma\} \). Shown are comparisons of the velocity for numerical solutions and the exact solution at \( t = 0.25 \). The Tal-Ezer mapping and exponential filter \( (p = 16) \) are employed.

A.15 The shock tube problem with the initial left state \( q_L(\rho, u, p) = \{1, 0, 1/\gamma\} \) and the initial right state \( q_R(\rho, u, p) = \{1/10, 0, 1/10\gamma\} \). Shown are comparisons of the total enthalpy for numerical solutions and the exact solution at \( t = 0.25 \). The Tal-Ezer mapping and exponential filter \( (p = 16) \) are employed.
Chapter One

Motivation and Objective
Prediction of ocean circulation is desired for many reasons such as global climate variability, marine environment protection, fishery, marine biology research, and military and rescue operations.

Ocean circulation plays an important role in the global climate and in its variability. The currents in the world’s oceans transport heat energy from one region to another to accounting for a substantial contribution to the global heat flux. The heat capacity of the earth’s oceans is much larger than that of the entire atmosphere. For example, the top approximately 3 m layer of the ocean holds as much heat as the global atmosphere.

The ocean is the largest carbon dioxide (CO\textsubscript{2}) reservoir holding 50 times more than the atmosphere. The excess of carbon dioxide is balanced between the atmosphere and the ocean through air-sea interaction. This exchange happens at a very slow rate (fully resolving CO\textsubscript{2} exchange requires time scales of a thousand years). As an example, thermocline circulation (THC), which has about a period of a millennium [80], is one of main mechanisms absorbing carbon dioxide in the cold regions where cold water sinks and releases it in the warm regions to the atmosphere. It is obvious that the climate change will significantly modify the ocean circulation as well. The ocean circulation can change the earth’s climate and also be conversely influenced by changes in the global climate in a nonlinear fashion.

Regional ocean dynamics such as wind-driven upwelling brings nutrient-rich water from the deep to the surface, which in turn leads to a locally high fish population. A significant portion of fishery occurs at these upwellings, which have very localized dynamics in the ocean [80].

In order to initialize dynamical forecasting systems for the ocean state, measure-
ments of high accuracy are required; such measurements are difficult and costly, and in many cases, e.g., in real-time adaptive sampling, they have to be done very fast. Given the spatio-temporal variability in the ocean and its intermittent dynamics, sampling a pre-determined region uniformly in time-space can be very inefficient. This is due to the fact that only a small subset of those measurements have a significant effect on the accuracy of the forecasts [132]. Adaptive sampling is an evolving method for the efficient sampling of the most energetic ocean phenomena in support of real-time nowcasting and forecasting. It has been used only recently in ocean forecasting demonstrations (e.g., [83]) and has the potential of reducing the observational requirements by orders of magnitude. However, adaptive sampling is still complicated and costly for routine observations. Moreover, truly real-time adaptive sampling of the ocean requires fast data assimilation methods and rigorous criteria to identify locations of best sensor placement. To this end, adjoint methods (that solve the inverse problem) [9] bear great promise, and a recent demonstration in a data assimilation experiment of the East Australia Current confirmed this [163]. However, adjoint methods tax computational resources more heavily than other methods and they typically require a lot of computer memory. Other methods, e.g., based on uncertainty estimation such as the ESSE system [101], have been used in certain demonstration experiments but they too are expensive for truly real-time adaptive sampling. Various nonlinear versions of the Kalman filter have been applied to simplified models, but no clear consensus on their effectiveness has been reached yet [35, 174]. A recent progress report from a joint NSF-ONR workshop [100] on Data Assimilation (DA) identified the need for improving real-time adaptive sampling and recommended the development of new economic DA without loss of accuracy based on reduced-dimension schemes that will complement adjoint- and ensemble-based methods.
In light of ocean complexities over a wide range of scales – see the “multiscale ocean” in [50] – extracting the proper hierarchy can be both valuable in physical understanding and also in developing new ways of modeling and forecasting ocean processes. Proper Orthogonal Decomposition (POD) [128, 17, 10, 143] is one such approach (also known as the method of Empirical Functions or EOF), and oceanographers have used it to analyze their data or to develop reconstruction procedures for gappy data sets, see [15, 48, 77, 56] and [164, 115, 85, 124]. In the present work, we are interested in extracting useful concepts from POD analysis that will lead to concrete guidelines for optimum and efficient sensor placement in adaptive sampling for regional ocean forecasting. Specifically, we pose the question: Can we use properties of the POD modes, computed over a specific time interval, to decide on the best and, we hope, optimum location of the sensors? Here we define optimum location as the positions that will give us the best possible reconstructed ocean field, given the limited number of measurements at these locations.

To this end, instead of tracking instantaneous special flow features in the physical domain, we are looking for special features in the POD modes, that is in the modal domain. However, for this approach to be efficient, we first need to demonstrate, from numerical simulations, that the ocean dynamics predicted by different regional models is low-dimensional, and hence that just a few dominant POD modes can reproduce the essential ocean dynamics. Clearly, the dimensionality of the ocean region we simulate depends strongly on the phenomena involved (e.g., convection, upwelling, etc.) and no general statements can be made. To address this difficult question, we have undertaken a systematic study using three different ocean models, ROMS [133], HOPS [84], and FVCOM [58] to simulate the short-term and long-term dynamics of different regions and phenomena. As we will see, this distinction between short-term and long-term dynamics is important as the effective dimensionality of
the ocean system increases with time. In Chapter 2, we will present details of the methodology and its applications in the three different ocean regions: New Jersey and Middle Atlantic Bight, Massachusetts Bay, and Gulf Of Maine.

Waves are another important phenomenon in the ocean and have important consequences for the environment and operational equipment in the sea or coastal areas. Waves can damage harbors, vessels, offshore structures, etc. and are capable of extending their damage to the structures on the land (tsunami waves). The importance of wave information is vital to harbor designers, coastal protection agencies, near-coast and offshore structures, navigation, and rescue and military operations in the sea.

Waves can be considered as the sum of many waves in different directions and frequencies. To simulate realistic ocean waves, the spectral ocean wave model (phase-averaged models) [82, 173, 11] is the only proven and feasible current model. The model includes propagation of energy density of waves in geophysical and spectral spaces. A solution to the energy density equation is only possible through numerical methods for real applications. They are many numerical schemes available to us to discretize the energy density equation. The traditional methods are finite difference, finite element, and finite volume schemes. An ideal new scheme should meet the requirements of realistic application and be more efficient than the traditional ones. The ideal numerical scheme should have low dissipation and phase errors, which might otherwise damp/shift the solution since waves travel a greater distance. The complex coastal boundaries stipulate that the ideal new scheme should handle general grids (unstructured). The conventional discretization (finite difference or finite element) for the wave equation requires huge computational resources (in memory and speed). The recent coupling effort between the ocean circulation model and wave models [109, 108] reveals that the wave models are two-orders of magnitude
more expensive than the ocean circulation model for the same mesh. The numerical ocean community critically needs an efficient scheme for ocean wave equation (private communication with Prof. C. Chen, UMass, Bedford). Accordingly, the new scheme should be *computationally efficient* (fast and low memory storage in the computer).

The high-order numerical methods are known to be superior to low-order ones for the same accuracy. In many cases, low-order methods need to employ extremely high-resolution meshes to match the accuracy obtained by high-order methods [78, 91]. The computational savings (in speed and memory), if high-order methods are employed, can be several orders of magnitude for some cases. The desired level of dissipation/phase errors can be efficiently achieved by high-order methods.

The spectral wave model has two separable spaces: (1) geophysical space ($\mathbf{X}$) and (2) spectral space ($\theta \in [-\pi, \pi], \sigma \in [\sigma_{\min}, \sigma_{\max}]$). The conspicuous feature of the geophysical aspect is the complexity of the coastal boundaries. The discretization in the geophysical space via the Discontinuous Galerkin (DG) method supplies us an easily constructed high-order scheme on general grids. The high-order methods are almost free of dissipation/phase errors. Ocean models using a first order scheme (as used by all the traditional models) for the spatial advection term are limited to coastal simulations where waves propagate over relatively short distances such that the numerical solution still does not damp out significantly. There are very few high-order discretizations of the geophysical derivatives in the literature. We only found that Tolman [153] and Bender [18] reported improving results using second and third-order schemes for the propagation term in physical spaces. On the other hand, unlike geophysical space, spectral space has no tolerance for numerical diffusion of the first-order scheme; those terms are exclusively treated with second-order discretization [23]. The discretization of the spectral space derivatives demands a higher-order
accuracy more than that of physical space derivatives. To this end, the Fourier-collocation discretization in the spectral space can be chosen due to the simplicity of the domain.

The discontinuous Galerkin (DG) method originated with Reed and Hill’s work [127] on neutron transport. They proposed a discretization that allows an approximate solution (angular neutron flux) to be discontinuous across triangular boundaries for the neutron transport equation. This method later has been shown to have convergence $O(h^{P+1/2})$ for scalar linear hyperbolic equations on general grids of cell size $h$ and with a polynomial basis of order $P$ [90] [79]. Chavent and Salzano [39] discretized space with a piecewise linear element and time using a forward Euler method for the one-dimensional conservation law. However, the stability condition of the scheme was far too restrictive to be useful in practice. Later, Chavent and Cockburn [38] relaxed this restriction by incorporating a slope limiter, but still achieved only first-order accuracy in time. Cockburn and Shu [44] [45] [46] proposed the nonlinearly stable (as well as high-order in time) Total Variation Diminishing (TVD) Runge-Kutta discontinuous Galerkin method (RKDG). We have followed this discretization in this study.

The discontinuous Galerkin method has advantages over other traditional methods (FEM, FVM, FDM) in numerous ways: has relatively easy implementation to high-order (no continuity requirement across cell boundaries), is highly parallelizable, handles complex geometries well by allowing generalized grids, and is well suited for $p$-adaptivity and inherently conservative scheme. We refer interested readers to DG historical developments and the rich literature reviews of Cockburn and Karniadakis and Shu [43], Cockburn and Shu [46] and the work by Hesthaven and Warburton [79].
The second part of this thesis in Chapter 3 introduces the action balance equation and source terms governing ocean waves. We propose a new scheme for solving the equation in Chapter 4 and verify and validate this scheme in Chapter 5.

The phase-averaged models have primary variables such as energy density and obtain important integrated wave parameters such as significant wave height \( H_s \), absolute mean wave period \( T_{m01} \), etc. through integration of the energy density. The significant wave height is related to maximum possible wave height \( H_{max} \), which is critical to ocean structures. As for illustration, the designers of extreme wave structures (wave harbours, nuclear plant) usually need to know the maximum wave height \( H_{max} \), which is expressed (using Rayleigh-distribution) by [103, 146]

\[
H_{max} = 0.707 H_s \sqrt{\log N}
\]  

(1.1)

where \( H_s \) is significant wave height and \( N \) is the number of waves. For a known significant wave height \( H_s \), the number of waves \( N \) for the duration \( T_d \) with the average wave period \( \bar{T} \) may be estimated by \( T_d/\bar{T} \). Let us say that waves have the duration of 1 year with an average period of 15 s, and the number of waves \( N \) is \( 365 \times 24 \times 3600/15 \), then we obtain \( H_{max} = 2.9 H_s \) via equation (1.1). However, this probabilistic estimation of the maximum wave height relies solely on the prediction of the deterministic value of the significant wave height from the numerical simulation. The significant wave height obtained from the numerical simulation intrinsically has many uncertainties.

The spectral wave models might predict the waves for a variety of scales, but this is still the only option for wave simulation in the large basins. Nevertheless, it is not a perfect model under extreme conditions. The spectral wave models rely on source
terms to represent generation/dissipation of waves (wave generation, white-capping, 
depth-induced breaking, and bottom friction) and wave interactions (triads, quadruplets). Among them, we know the exact mathematical expression only for nonlinear 
wave interactions (quadruplets) [69, 71, 70] but model the rest of the source terms.
However, quadruplets were also modeled by Hasselmann et. al. [75, 76] since the 
computation of the exact formulation is several orders more expensive than the solver 
itself. The numerical ocean community has mostly adopted parametrization of source 
terms from the WAMDI group’s work [160]. It is the so-called third-generation ocean 
wave prediction model. The model’s success relies heavily on the parametrization of 
the source terms. Parametrization is one of the main sources of uncertainty in spec-
tral ocean modeling. Data assimilation techniques are usually applied to estimate 
the better parametrization, which always requires real observations. Regardless of 
whether the observation is realizable or not (cheap or expensive), practical engineer-
ing is interested in the range of extreme solutions. To this end, we will do uncertainty 
quantification of the HISWA case in a generalized polynomial chaos (gPC) frame-
work [59, 168]. The source of uncertainty besides the source parameters comes from 
bathymetry, current field (if any), boundary and initial conditions, etc.

Generalized polynomial chaos (gPC) [169] is an extended version of the clas-
sical polynomial chaos (PC) method [59]. Generalization is introduced by using 
the Askey scheme for hypergeometric orthogonal polynomials [169]. The gPC ba-
sically expands the random inputs in terms of orthogonal polynomials which have 
exponential (spectral) convergence for the smooth data. The expansions will be in-
corporated into the system equations then will use the orthogonality of the basis 
functions and the Galerkin approach (gPC Galerkin method) to reduce the Partial 
Differential Equations (PDE) system to an ordinary differential equation. The other 
traditional methods such as Monte Carlo, perturbation, and moment equation tech-
niques have generally limited applications or are inefficient for complex systems. The
gPC Galerkin method is a proper way to deal with any system, but it needs to be
derived for every system applied and requires the modification of a deterministic
simulator (non-intrusive way). Moreover, it is difficult to apply the gPC Galerkin
method for nonlinear problems (i.e., nonlinear terms not in polynomial forms). The
gPC adopted many ideas from the spectral methods [60, 78], which have overcome
similar difficulties from the Galerkin approach by introducing the collocation method.
The collocation methods are simply that the residual of the governing equations is
zero at the collocations. We can similarly apply this idea to uncertainty quantifi-
cation. The stochastic governing equation is zero at the collocations which are now
chosen in random space. Each collocation point in random space is now a deter-
ministic governing equation. The collocation method requires two ingredients: (1)
a deterministic solver and (2) collocations at the random space for the uncertainty
quantification. The important statistical parameters can be post-processed from
the ensemble. This is called the stochastic collocation (SC) method. Unlike the
Galerkin approach, there is no need to derive stochastic equations for each system.
Moreover, SC methods run each simulation independently (embarrassingly parallel)
for the collocations and hence they are widely preferred in practice.

The stochastic collocations can be constructed by the tensor product of multidi-
ensional random variables. However, the number of deterministic runs (N) from
the tensor-product construction can be prohibitively expensive ($O(N^d)$) for large
dimensions ($d$). The Smolyak grid is a subset of tensor-product collocations [144, 4]
that copes with the curse of dimensionality of tensor-product construction. We here
used the sparse grid based on Clenshaw-Curtis quadratures [42]. The sparse grid
has significantly reduced the number of required collocations (about several orders)
compared to tensor-product construction or Monte Carlo sampling.
Stochastic collocation simulation requires many deterministic runs (high-order spectral ocean wave code [170] here). The number of runs depends on the level and dimensions of the sparse grid. The complex code with many runs usually requires high performance computers (HPC). Also, scheduling many runs on HPC is cumbersome for users. We used the uncertainty quantification program (PUQ) [3] for generating collocation points for random variables, scheduling job submission on HPC, and post-processing the ensemble to obtain important statistical parameters such as mean and standard deviation.

The previous works in the literature are mostly dominated by Monte-Carlo methods. The sensitivity of wind variability was studied by introducing the Gaussian random noise to surface wind velocity on the spatial grid [123]. Abdalla & Caveleri later investigated the effects of wind gustiness and air density variability on wave growth [6] using the Monte Carlo simulation with ECWAM. Bouws & Battjes applied the Monte Carlo approach for refraction of water waves using initial values of wave frequencies and directions of propagation as random variables [26]. Bonekamp et al. [22] performed Monte Carlo simulation for the Charnock [37] parameter using an atmosphere-wave coupled version of ECMWF. The ECMWF ensemble prediction system generating realizations only for initial conditions was also used by Roulston et al. [134]. They later post-processed the ensemble forecasts to improve the reliability of the wave statistics. The Bayesian network model has successfully been applied to handle boundary, bathymetry, and parameter uncertainties in Plant et al.'s work [121, 122]. Cea et al. [36] applied quasi-Monte Carlo simulation [97], which significantly accelerated the convergence rate of the traditional method for the mass transport equation in shallow estuaries. There are many studies on uncertainty quantification using Monte-Carlo based methods. Although the gPC usage entered into many areas [168], we did not come across works (or rather only a few)
on the generalized polynomial Chaos (gPC) framework in the numerical ocean wave community. The deterministic simulation of a real ocean is already a computationally demanding application. A stochastic application that at least requires several orders of magnitude larger in computation resources will reach the limits of computational power. However, stochastic collocation using the Smolyak sparse grid is currently the most promising paradigm in stochastic application for a reasonable number of uncertainty variables (< 20).

The third part of this study is uncertainty quantification of a wave application (HISWA experiment given in Chapter 5). The source term parameters for depth-induced breaking, bottom friction, and triads are taken as random variables. We also added a random process of the current field for the uncertainty quantification. We first considered source term and current field randomness individually and then all together. The results are presented in Chapter 6.
CHAPTER TWO

Data simulation and POD
We first show that the numerical ocean output from ROMS, HOPS, and FVCOM codes is low-dimensional. In this chapter, we also demonstrate numerical evidence that, by increasing the length of the snapshots, we will have convergence for the lower modes. By using the gappy POD approach, we reconstruct all the ocean data with few sensors in an extremely fast way. These few sensors are selected as max/min of POD modes as the optimum locations. The methodology is applied to three regions: (1) New Jersey and Middle Atlantic Bight (ROMS), (2) Massachusetts Bay (HOPS), and (3) Gulf of Maine (FVCOM).

We first discuss the governing equation of the ocean state in section 2.1. Then we present POD formulation in section 2.2. The snapshot POD which makes the method practical for real applications is given next in section 2.3. The low-dimensionality of the ocean output from the three codes is given in section 2.4. Subsequently, we present a recent extension of POD, termed gappy POD [57, 15, 158, 165], that leads to a reformulation of the data assimilation problem as a gappy data problem in section 2.5. First, we apply this approach to a benchmark problem (flow over sphere) in section 2.5, and then to Massachusetts Bay in section 2.6. We conclude the paper with a summary and a discussion of our findings in section 2.7.

2.1 Ocean Circulation Models: ROMS, HOPS, and FVCOM

The general ocean models solve conservation of mass, momentum, and tracers (salinity and temperature) for hydrostatic and stratified fluid on a rotating earth. The equations are called primitive equations since they are cast in terms of primitive variables rather than conserved variables. The Boussinesq approximation ignores the all
variations of density of a fluid in the momentum equation, except when associated with the gravitational term [155]. Another simplification is the hydrostatic approximation that order of vertical pressure and gravity terms are much higher than time derivative, nonlinear advection, and Coriolis terms [67]. The governing equations will be complete with the equation of state that relates density, pressure, salinity, and temperature.

The large-scale ocean circulation can be classified as a hydrostatic, stratified flow. The ratio of vertical (1 – 10 km) to horizontal lengths (5000 – 10000 km) in the ocean is so small that it is classified as shallow water. This implies that vertical dynamics are not important and leads to a hydrostatic state assumption of the ocean. The density changes with depth and it depends on temperature, salinity. The near balance of pressure gradient and Coriolis forces result in a quasi-geostrophic flow in the ocean. The implication of this phenomenon is that water flows along constant pressure isobars.

The current algorithms for the ocean circulation model stem from variations of the Bryan-Cox model [33]. The Bryan-Cox model frames the equations using a geopotential vertical coordinate and discretizes them by a finite difference scheme.

The finite volume, finite element, and finite difference discretization schemes are the most popular methods among the ocean codes. ROMS and HOPS employ finite difference methods, whereas FVCOM uses finite volume formulation. The finite volume formulation has advantages over the finite difference discretization in handling complex boundary shapes. FVCOM uses an unstructured grid and so represents complex boundaries better than the other two models used here, ROMS and HOPS.
The regional ocean models have high resolution, with horizontal resolution of about 1-10 km. However, it is known that turbulent processes in the ocean can be on the order of a few centimeters. In order to model the sum effect of the turbulent mixing, numerical ocean models incorporate subgrid-scale parametrization. The closure problem is to parametrize Reynold stress which arises in the time-averaging (Reynold averaging) of the momentum equations. The details of these parametrizations of the subgrid-scale can be found in the references for ROMS [66] and FVCOM [40] and HOPS [131].

The first ocean models that followed Bryan and Cox’s [34] [33] choice of geopotential (z-based) coordinate are also known as level models. The vertical coordinate choice has been diversified over years. The other most well-known choices are isopycnic coordinates (potential density as z coordinate) [20], terrain-following (sigma) coordinates (σ = 0 at the surface σ = −1 at the bottom) [137], and hybrid coordinates (including z, isopynic, and σ coordinates) [21]. The HOPS model is an example of level models (z-based); FVCOM and ROMS use sigma-coordinates for the vertical direction.

### 2.2 POD Formulation

Proper Orthogonal Decomposition (POD) [19] can be formulated as an optimization problem. That is, it seeks spatial functions maximizing energy under an orthogonality condition. Let us define time-average operator and inner product in the $L^2$ norm.
\begin{align*}
\langle \cdot, \cdot \rangle &= \lim_{T \to \infty} \frac{1}{T} \int_0^T \cdot dt, \\
(f, g) &= \int_\Omega fg d\Omega. \tag{2.1b}
\end{align*}

Target function \( \Phi(x) \), maximizing the energy \( \langle (\Phi(x), u)^2 \rangle \) under the normalization condition \((\Phi(x), \Phi(x)) = 1\). Using the time average and inner product operators, the energy can be written as

\[
E = \lim_{T \to \infty} \frac{1}{T} \int_0^T \left\{ \int_\Omega u(x, t)\Phi(x)dx \int_\Omega u(x', t)\Phi(x')dx' \right\} dt. \tag{2.2}
\]

Interchanging integral operators, the above equation can be regrouped as

\[
E = \int_\Omega \int_\Omega \left\{ \lim_{T \to \infty} \frac{1}{T} \int_0^T u(x, t)u(x', t)dt \right\} \Phi(x)\Phi(x') \, dx \, dx'. \tag{2.3}
\]

Defining the curly bracket term as \textit{space-correlation tensor} \( R(x, x') \), the energy \( (E) \) equation becomes

\[
E = \int_\Omega \int_\Omega R(x, x')\Phi(x)\Phi(x') \, dx \, dx'. \tag{2.4}
\]

\[
= \left( \left( R(x, x'), \Phi(x) \right), \Phi(x') \right).
\]

Let the above rhs be \( \mathcal{F}(\Phi) \) and normalization constraint condition, \(((\Phi(x), \Phi(x)) = 1\), as \( \mathcal{G}(\Phi) \) functionals. The maximization problem can easily be set using the
Euler-Lagrange equations:

$$\delta F = \lambda \delta G$$  \hspace{1cm} (2.5)

where "\(\delta\)" is a variation of the functionals and \(\lambda\) is the Lagrange multiplier. Since the variations are

$$\delta F = 2 \left( R(x,x'), \Phi(x') \right),$$  \hspace{1cm} (2.6)

$$\delta G = 2 \Phi(x),$$  \hspace{1cm} (2.7)

then the final equation to obtain \(\Phi(x)\) (maximizing the energy under the normalization condition) is:

$$\int_{\Omega} R(x,x') \Phi(x') dx' = \lambda \Phi(x).$$  \hspace{1cm} (2.8)

The above integral equation is called the Fredholm equation of the second kind [120]. And the target function \(\Phi(x)\) is an eigenfunction of the space-correlation tensor \(R(x,x')\) and is called the POD function.

Accordingly, there is no other basis (e.g., Fourier basis) more efficient than the POD basis in \(L^2\) sense. Due to the Hilbert-Schmidt theory, the integral equation has a countable infinite set of orthogonal solutions \(\Phi(x)\), which form a complete set of orthogonal functions [120]. Thus, variable \(u(x,t)\) can be expanded as

$$u(x,t) = \sum_{k=1}^{\infty} a_k(t) \Phi^k(x).$$  \hspace{1cm} (2.9)
Using the normalization of the POD function $\Phi(x)$, the POD temporal coefficients are expressed as

$$a_k(t) = \left( u(x,t), \Phi(x)^k \right).$$

(2.10)

The computation of the POD function from direct formulation (2.8) is prohibitively expensive for real applications. The size of space-correlation tensors can easily be orders of several hundred million. The eigenvalue and eigenvector computation is still challenging for this size of matrix. The Method of Snapshots using the ergodic hypothesis was proposed by Sirovich [140, 141, 142] to overcome this computational difficulty.

### 2.3 POD-Method of Snapshot

A Snapshot is an uncorrelated record of data for a discrete time

$$u^n = u(x,t_n),$$

(2.11)

where the integer ”$n$” signifies the specific time. The kernel of integral is approximated as

$$\mathcal{R}(x,x') \approx \frac{1}{M} \sum_{n=1}^{M} u^n(x)u^n(x'),$$

(2.12)

where $M$ is a sufficiently large total number of snapshots. Sirovich further argued
that the approximated kernel $R(x, x')$ is separable and corresponding eigenfunctions of the kernel can be expressed in expansion form as

$$\Phi(x) = \sum_{k=1}^{M} b_k u^k(x).$$  \hfill (2.13)$$

The original problem is now transformed to the following eigenvalue problem :

So

$$R(x, x') \approx \frac{1}{M} \sum_{n=1}^{M} u^n(x)u^n(x')$$

$$\int R(x, x')\Phi(x)dx' = \lambda \Phi(x)$$

$$CQ = \lambda Q$$  \hfill (2.14)$$

where the eigenvector $Q = (a_1, \ldots, a_M)$ contains time-dependent coefficients of the expansion and the time correlation matrix $C$, which is an $M \times M$ matrix, whose entries read

$$C_{ij} = \frac{1}{M} \int_{\Omega} u(x, t_i) \cdot u(x, t_j)dx.$$  \hfill (2.15)$$

Note that the original method solves $N \times N$ matrix and $N$ denotes the degree of freedom in space to obtain the eigenvectors directly. However, Sirovich’s method requires one to solve eigenvectors of only the time-correlation matrix $M \times M$ for time dependent coefficients and then determine the eigenvectors. The method of snapshot may have the substantial computational gain in the problems where $M << N$. The
total energy can now be defined in terms of the inner product of variable \( u(x,t) \), then

\[
E(t) = \left( < u(x,t), u(x,t) > \right) = \sum_{i=1}^{\infty} a_i(t)^2, \tag{2.16}
\]

We can further demonstrate the orthogonality of the coefficients \( a_k \) relative to the time-average operator \( \langle a_i(t)a_j(t) \rangle = \delta_{ij}\lambda^i \) [140]. As a result we get total energy as the sum of the eigenvalues.

### 2.4 Simulations and Low-dimensionality

We have employed three different regional ocean models, specifically, the Rutgers model ROMS [133], the Harvard model HOPS [84], and the University of Massachusetts at Dartmouth model FVCOM [58] to obtain simulation data for different regions and conditions (New Jersey coast, Massachusetts Bay (Mass Bay), and Gulf of Maine, respectively). We have analyzed the ocean dynamics over short-term but also for much longer periods using POD. By cross-correlating different snapshots obtained from the simulations we constructed the covariance matrix, the eigen-decomposition of which yields the POD eigenvalues and corresponding POD temporal and spatial modes. In particular, the sum of the normalized eigenvalues is representative of the energy captured by the corresponding POD modes. The POD modes are hierarchical with the lower-indexed modes containing higher energy. We used a serial code based on LAPACK for the ROMS and HOPS simulation outputs,
however, we implemented a parallel version of the POD code based on ScaLAPACK to deal with the large matrices involved in the Gulf of Maine FVCOM simulations (matrices with more than 1 billion entries).

In the following, we describe the different data sets we analyzed and present a summary of our results that provide evidence of the low-dimensionality of the “numerical” ocean.

### 2.4.1 New Jersey and Middle Atlantic Bight

The Lagrangian Transport and Transformation Experiment (LaTTE) is a coordinated program of field and numerical experiments that addresses the biological and geographic extent of contaminants along the New Jersey and Middle Atlantic Bight, see [99, 41] and Fig. 2.1. We will use the LaTTE data base in our analysis. Here, POD analysis is carried out for two cases, one short-term and one long-term simulation. The simulation results were provided by the Rutgers Ocean Modeling Group [135]. The short-term simulation is for 2.5 days; it starts at midnight of May 13, 2005 and ends at noon of May 15, 2005. The long-term simulation is for 25 days; it starts on February 04, 2006 and ends on February 28, 2006. Time resolution of snapshots and various snapshot length effects on energy spectra and POD coefficients were investigated systematically to avoid any erroneous conclusions.

The normalized energy spectra for both cases are shown in Fig. 2.2 for three variables, namely horizontal velocity vector, temperature and salinity. Several resolution checks were performed to assess the accuracy of our results. In Fig. 2.3 we show the effect of sampling at four different resolutions. Specifically, the short-term simulation was sampled for every 0.625hr, 1.25hr, 2.50hr and 3.75hr while the long-
term simulation for every 3.0hr, 6.0hr, 12.0hr and 18.0hr. The percentage of total energy ($\sum \lambda \times 100$) captured by the coarse resolutions superimposed on the finest resolution shows that the low modes of the finest simulation comprise about 99% of the total energy. The resolution study has been repeated for the time dependent POD coefficient of horizontal velocity ($u, v$) vector for a 2.5 day (short) and 25 days (long) simulations. We have presented the first four coefficients (time-dependent) of POD modes in Figures 2.4 and 2.5, respectively. The short and long simulations show that the time resolutions have significant effect on the POD coefficients. Especially, the long term simulation reveals that poor time resolution might lead large errors in the POD coefficients.
Figure 2.2: New Jersey coast: Energy spectra of horizontal velocity vector \((u,v)\), temperature and salinity for 2.5-days (left) and 25-days (right) simulations.

Figure 2.3: New Jersey coast: Time resolution effects on the energy spectra of horizontal velocity \((u,v)\) vector for 2.5-days (left) and 25-days (right) simulations.

2.4.1.1 The Snapshot Length Effect

The duration length of the simulation plays an important role in POD modes and coefficients. In complex dynamic simulation, extending the simulation time certainly adds more new features to those dynamics. The POD modes and coefficients will
Figure 2.4: Time resolution effect on POD temporal modes of horizontal velocity \((u, v)\) vector for a 2.5 day simulation of the New Jersey coast.

continually be modified depending on the new feature in the dynamics. The snapshot length effect in (LaTTE) simulation will be studied here. Figure 2.6 presents energy spectra for the short (left) and long (right) simulations. The snapshot lengths have modified the energy spectra of short simulation stronger than long simulation. In
Figure 2.5: Time resolution effect on POD temporal modes of horizontal velocity \((u, v)\) vector for a 25 day simulation of the New Jersey coast.

In the short simulation, the first few POD modes start to converge about the 2.5 days length simulation. The lower fidelity of the POD modes for the various snapshot lengths in the short simulation is a revelation of the snapshot length shortness. In the longer simulations, we see that only higher POD modes will be strongly
modified as we increase the snapshot length; conversely, lower POD modes change slightly. The first four POD coefficients in Figures 2.7 (short) and 2.8 (long) have been given for the various snapshot lengths. In the short simulation, only the first POD time coefficient converges reliably in the first mode. The second and third POD time coefficients converge to a somewhat moderate level; however, the fourth POD coefficient obviously suffers shortness of the snapshot length effect. On the contrary, the long simulation has the POD coefficients which are only slightly modified with the longer snapshot. The main motivation of this chapter is to reconstruct the entire ocean with a few sensors. The idea will be explained in section 2.5. The method uses a few lower POD modes for this construction. The underlying assumption for this method in the practical application is that the lower modes are slowly changing. We here present a strong evidence that, with enough sets of snapshots, we can use these POD modes for the reconstruction since they are in slow-manifold.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{energy_spectra_pod.png}
\caption{Energy spectra of POD for Velocity}
\end{figure}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{energy_spectra_pod_long.png}
\caption{Energy spectra of POD for Velocity}
\end{figure}

**Figure 2.6:** The simulation length (in time) effect on energy spectra of horizontal velocity \((u, v)\) vector for a 2.5 day and 25 day simulation of the New Jersey coast. The left figure represents short-time and the right one does long-time simulations.
Figure 2.7: The snapshot length effect on POD temporal modes of horizontal velocity ($u, v$) vector for a 2.5 day simulation of the New Jersey coast.

2.4.2 Massachusetts Bay

The Massachusetts Bay (Mass Bay) simulation data was provided by the Harvard Ocean Prediction System Group [101], see Fig. 2.9. The short-term simulation covers
Figure 2.8: The snapshot length effect on POD temporal modes of horizontal velocity \((u, v)\) vector for a 25 day simulation of the New Jersey coast.

eight days, starting on August 25 and ending on September 2, 1998; the data are recorded at every hour (193 snapshots). The long-term simulation covers 47 days, between August 20 to October 6, 1998; the data are recorded daily (47 snapshots). The energy spectra (excluding the mean mode) is given for both simulations in Fig.
2.10. As also seen in the New Jersey simulations, here too a few modes contain most of the energy for both simulations, see also Tables 2.1 and 2.2. The first four modes account for 93\% for the 8-days simulation, and for 78\% for the 47-days simulation, as seen in Table 2.1. Additionally, Table 2.2 shows the eigenmode energy content of the instantaneous field (excluding mean mode). The temperature and salinity variables show very low dimensionality for both simulations with the first mode only accounting for about 99\% of the total energy.

2.4.3 Gulf of Maine

The Gulf of Maine was simulated by the Unstructured Finite Volume Coastal Ocean Model (USG-FVCOM) and the simulation data were provided by the University of Massachusetts-Dartmouth Ocean Modeling Group; see Fig. 2.11. The simulation
covers all of August 1998 and the output is recorded hourly (744 snapshots); a typical result is shown in Fig. 2.12, including sensitivity of the velocity energy spectrum on two different grids. We note parenthetically here that the third POD mode shows that most of the activity is centered around Georges Bank (not shown here). Table 2.1 presents the first nine POD eigenvalues. The first mode of temperature and salinity again account for 99% of the total energy, while the sum of the first three eigenmodes of velocity accounts for about 95%.

### 2.4.4 The “numerical” ocean is low-dimensional

The physical and biological processes in the ocean are characterized by a wide range of spatio-temporal scales, from 1 mm for molecular processes to more than 10 km for fronts, eddies and filaments, and corresponding characteristic times from 1 second to several months [50]. However, the question that we address here is what range of such scales is captured in simulations using some representative regional ocean
Figure 2.11: Gulf of Maine: Contours of the first POD mode (temperature) and domain.

models, and what is the corresponding energy hierarchy. (We refer to the outputs of the simulation codes as “numerical” ocean.)

We present a summary of all our results in terms of the relative energy as represented by the normalized eigenvalues. Specifically, in Table 2.1 we list the first nine POD eigenvalues from the analysis of the total fields, i.e., including the mean mode. This mode, corresponding to the first row in the table, is responsible for about 99.9% of the total energy for temperature and salinity. Velocity, however, has a wider distribution with its first two eigenmodes containing about 90% energy for the Gulf of Maine, about 70% for Massachusetts Bay, and about 50% for the New Jersey Coast. The energy contained in the sum of the first eight eigenmodes comprises about 97% for the short-term simulation and about 91% for the long-term simulation. We also present the POD eigenvalues in Table 2.2 with a different normalization, i.e., we do not include the first (mean) mode, and hence it is easier to appreciate the energy
content in all the modes representing spatio-temporal fluctuations.

In order to put these results into the proper context we compare next with a well-studied prototype flow, namely the laminar and turbulent wake. In past work, in a series of papers [53, 106, 105], the low-dimensionality of the turbulent wake in flow past a circular cylinder has been demonstrated. These results were based on direct numerical simulations (DNS), where all spatial scales down to Kolmogorov scale were accurately resolved. We want to compare the dimensionality of the dynamics in the three ocean regions we simulated with ROMS, HOPS and FVCOM with that of the cylinder flow in order to gain some insight on its complexity. In Fig. 2.13 we plot the eigenspectra of the cylinder flow at $Re = 10,000; 3,900$; and 185 together with the ocean eigenspectra corresponding to the long-term simulations for the LaTTE, Mass Bay, and Gulf of Maine data.\footnote{We note that at $Re = 185$ the cylinder wake is three-dimensional but laminar whereas at $Re = 3,900$ it is turbulent with a small inertial range; at $Re = 10,000$ the wake is fully turbulent although the boundary layer around the cylinder is still laminar.} Based on the relative rapid decay of the ocean eigenspectra for all three cases, it is clear that the dynamics that ROMS, HOPS
and FVCOM simulate is low-dimensional and certainly much less complex than the cylinder dynamics at $Re = 10,000$ for which there is significant energetic contribution from the high modes unlike the ocean dynamics. To appreciate also the distinction between the short-term and long-term ocean dynamics, we plot in Fig. 2.14 the eigenspectra for the short- and long-term simulations of Mass Bay; at the 40th mode the energy content for the former is about two orders of magnitude smaller than the energy content of the latter. We will analyze these two cases in more detail in section 2.6.

These and other results not presented here suggest that in the current generation of mesoscopic ocean models the dynamics captured is low-dimensional, and hence reduced-order modeling is possible. This, in turn, implies that both the data assimilation problem as well as the forecasting problem can be re-formulated in order to exploit the efficiency of low-dimensional representations, specifically using the POD modes. While this finding may not be entirely surprising, the actual low value of the dimensionality, especially for the short-term dynamics compared to the periodic vortex shedding of the turbulent wake, is an intriguing result. It does not reflect the real ocean dynamics, which is clearly more complex than the cylinder dynamics; after all the Reynolds number of the flow around a buoy (used possibly for measurements) at modest current is larger than $Re = 10,000$! However, it points to the limitations of the physical modeling in the current generation of mesoscopic ocean codes and their inability to resolve accurately energetic contributions beyond a handful of modes. One could possibly argue that this is also a limitation of the coarse spatio-temporal resolution but at least for the results we presented here, see Figs. 2.3 and 2.12, this does not seem to be the case. In our view, the problem lies in the modeling of the sub-grid terms, and future reformulations of turbulence modeling in the framework of large-eddy simulations (LES) – using petaflop resources – will improve the ac-
accuracy of simulated ocean dynamics greatly.\textsuperscript{2} Such LES-based ocean models may require new data assimilation schemes.

![Figure 2.13: Comparison of eigenspectra for the turbulent wake and the long-term ocean dynamics of three regions.](image)

### 2.5 POD-based Reconstruction

A three-dimensional field $\mathbf{u}(\mathbf{x},t)$ can be represented by $\mathbf{u}(\mathbf{x},t) = \sum_{k=1}^{\infty} a_k(t) \Phi_k(\mathbf{x})$, or by employing a truncated expansion it can be approximated as

$$
\mathbf{u}(\mathbf{x},t) \approx \sum_{k=1}^{K} a_k(t) \Phi_k(\mathbf{x}),
$$

\textsuperscript{2}This observation is supported by the fact that the aforementioned cylinder eigenspectrum at $Re = 10,000$, simulated not with DNS but based on Reynolds-averaged Navier-Stokes (RANS), resembles the eigenspectrum of Fig. 2.13 at $Re = 185$. 
Table 2.1: Mean (First) and subsequent eight eigenmodes for New Jersey Coast \( (1 = 2.5 \text{ days}, 2 = 25 \text{ days}) \), Massachusetts Bay \( (1 = 8 \text{ days}, 2 = 47 \text{ days}) \) and Gulf of Maine simulations.

<table>
<thead>
<tr>
<th>Mode No:</th>
<th>NJ Coast (^1)</th>
<th>NJ Coast (^2)</th>
<th>Mass. Bay (^1)</th>
<th>Mass. Bay (^2)</th>
<th>Gulf of Maine</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Velocity</td>
<td>Temperature</td>
<td>Salinity</td>
<td>Velocity</td>
<td>Temperature</td>
</tr>
<tr>
<td>First Mode</td>
<td>30.52 %</td>
<td>99.94 %</td>
<td>99.99 %</td>
<td>31.82 %</td>
<td>99.72 %</td>
</tr>
<tr>
<td>Second Mode</td>
<td>23.34 %</td>
<td>0.02 %</td>
<td>0.01 %</td>
<td>20.26 %</td>
<td>0.14 %</td>
</tr>
<tr>
<td>Third Mode</td>
<td>12.38 %</td>
<td>0.01 %</td>
<td>&lt; 0.01 %</td>
<td>12.96 %</td>
<td>0.04 %</td>
</tr>
<tr>
<td>Fourth Mode</td>
<td>10.46 %</td>
<td>&lt; 0.01 %</td>
<td>&lt; 0.01 %</td>
<td>10.88 %</td>
<td>0.02 %</td>
</tr>
<tr>
<td>Fifth Mode</td>
<td>8.58 %</td>
<td>&lt; 0.01 %</td>
<td>&lt; 0.01 %</td>
<td>5.68 %</td>
<td>0.01 %</td>
</tr>
<tr>
<td>Sixth Mode</td>
<td>7.54 %</td>
<td>&lt; 0.01 %</td>
<td>&lt; 0.01 %</td>
<td>4.39 %</td>
<td>0.01 %</td>
</tr>
<tr>
<td>Seventh Mode</td>
<td>1.89 %</td>
<td>&lt; 0.01 %</td>
<td>&lt; 0.01 %</td>
<td>1.91 %</td>
<td>&lt; 0.01 %</td>
</tr>
<tr>
<td>Eighth Mode</td>
<td>1.54 %</td>
<td>&lt; 0.01 %</td>
<td>&lt; 0.01 %</td>
<td>1.72 %</td>
<td>&lt; 0.01 %</td>
</tr>
<tr>
<td>Ninth Mode</td>
<td>0.99 %</td>
<td>&lt; 0.01 %</td>
<td>&lt; 0.01 %</td>
<td>1.49 %</td>
<td>&lt; 0.01 %</td>
</tr>
</tbody>
</table>
Table 2.2: First eight eigenmodes for New Jersey Coast ($^1 = 2.5$ days, $^2 = 25$ days), Massachusetts Bay ($^1 = 8$ days, $^2 = 47$ days) and Gulf of Maine simulations.

<table>
<thead>
<tr>
<th>Mode No:</th>
<th>NJ Coast $^1$</th>
<th>NJ Coast $^2$</th>
<th>Mass. Bay $^1$</th>
<th>Mass. Bay $^2$</th>
<th>Gulf of Maine</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Velocity</td>
<td>Temperature</td>
<td>Salinity</td>
<td>Velocity</td>
<td>Temperature</td>
</tr>
<tr>
<td>First Mode</td>
<td>33.6 %</td>
<td>49.1 %</td>
<td>59.1 %</td>
<td>29.7 %</td>
<td>49.5 %</td>
</tr>
<tr>
<td>Second Mode</td>
<td>17.8 %</td>
<td>13.6 %</td>
<td>19.4 %</td>
<td>63.6 %</td>
<td>16.3 %</td>
</tr>
<tr>
<td>Third Mode</td>
<td>15.1 %</td>
<td>8.1 %</td>
<td>10.5 %</td>
<td>15.1 %</td>
<td>9.8 %</td>
</tr>
<tr>
<td>Fourth Mode</td>
<td>12.3 %</td>
<td>6.0 %</td>
<td>18.7 %</td>
<td>12.3 %</td>
<td>7.4 %</td>
</tr>
<tr>
<td>Fifth Mode</td>
<td>10.9 %</td>
<td>5.1 %</td>
<td>1.6 %</td>
<td>10.9 %</td>
<td>3.1 %</td>
</tr>
<tr>
<td>Sixth Mode</td>
<td>2.7 %</td>
<td>3.8 %</td>
<td>10.0 %</td>
<td>2.7 %</td>
<td>2.4 %</td>
</tr>
<tr>
<td>Seventh Mode</td>
<td>2.2 %</td>
<td>3.3 %</td>
<td>0.9 %</td>
<td>2.2 %</td>
<td>2.0 %</td>
</tr>
<tr>
<td>Eighth Mode</td>
<td>1.4 %</td>
<td>2.6 %</td>
<td>0.8 %</td>
<td>1.4 %</td>
<td>1.4 %</td>
</tr>
</tbody>
</table>
Figure 2.14: Comparison of eigenspectra for the turbulent wake and the short-term and long-term ocean dynamics of Mass Bay.

where \( K \) is the number of the basis functions \( \Phi^k(x) \), and \( a^k(t) \) are the time-dependent coefficients. Let us define the gappy vector \( \mathbf{\tilde{u}} \), which is the point-wise product of a mask vector \( \mathbf{m} \) and the complete vector \( \mathbf{u} \). Defining an intermediate solution vector \( \mathbf{\tilde{u}}_K \) that uses the existing spatial POD functions \( \Phi^k(x) \), we obtain the expansion form

\[
\mathbf{\tilde{u}}_K = \sum_{k=1}^{K} b^k(t) \Phi^k(x)
\]

defined by the unknown coefficients \( b^k(t) \). Minimization of the error defined between the gappy vector \( \mathbf{\tilde{u}} \) and the intermediate solution \( \mathbf{\tilde{u}}_K \) results in the best possible time coefficients \( b^k \) in the mean energy sense (L2-norm). The error can be defined in the gappy norm as

\[
\mathcal{E} = ||\mathbf{\tilde{u}} - \mathbf{\tilde{u}}_K||_m^2,
\]

(2.19)
where the subscript \( m \) denotes that this is not the standard L2-norm. By defining the gappy inner product \((u, v)_m = ((m \cdot u), (m \cdot v)) \) (\( \cdot \) as point-wise multiplication), the corresponding gappy norm \( ||v||_m^2 = (v, v)_m \) can be defined. We note that the POD functions \( \Phi(x) \) are no longer orthogonal with respect to the gappy inner product. We can write Eq. (2.19) as

\[
E = ||\tilde{u}||_m^2 - 2 \sum_{k=1}^{K} b^k (\tilde{u}, \Phi^k(x))_m + \sum_{i=1}^{K} \sum_{j=1}^{K} b^i b^j (\Phi^i(x), \Phi^j(x))_m ,
\]

(2.20)

and by differentiating with respect to \( b^n(t) \), the error-minimization problem leads to the equation

\[
\frac{\partial E}{\partial b^n} = -2(\tilde{u}, \Phi^n)_m + 2 \sum_{k=1}^{K} b^k (\Phi^k, \Phi^n)_m = 0.
\]

(2.21)

This is a linear system of equations in matrix form for the coefficient \( b^k \) in the form

\[
Mb = f
\]

(2.22)

where the matrix entries are defined as \( M_{ij} = (\Phi^i, \Phi^j)_m \) and the right-hand-side matrix can be computed from \( f_i = (\tilde{u}, \Phi^i)_m \), so \( f_i \) represents the projection of the measured data onto the available eigenmodes. We note that the matrix \( M \) is non-singular – as it may be the case in the original method ([57]) – because the eigenmodes are not obtained from gappy snapshots but rather from the output of the ocean model code. Having obtained the time coefficients \( b^k \), we can find the temporary solution through Eq. (2.18) and then repair the gappy data \( \tilde{u} \) with new information in \( \tilde{u}_K \) for the missing points.

We need, however, to determine the criteria for the optimum placement of the
sensors available to sample the POD modes. In ([47]) this problem was considered for unsteady flow past a circular cylinder and the sensors were placed at the extrema of the POD modes. This, of course, assumes that one knows these modes in advance or in practice one can approximate (e.g., by extrapolation) these modes from another (e.g., near-by) state. An alternative approach in selecting optimum locations was presented in ([165]), where the properties of the linear system in the modified POD formulation were involved; a similar approach was also used in ([111]). To make this point more clear, we consider the linear system of equations (equation (2.22)), and we solve for the unknown coefficients $b^k$ with the forcing $f_i$ obtained from the measured data. We note that the matrix $M$ is the identity matrix in the case of complete data due to orthonormality and its condition number $\kappa(M) = 1$ in that case. However, for gappy data $\kappa(M) > 1$, and hence an “optimization” problem can be set up, where the locations of sensors are obtained so that $\kappa(M)$ is minimized, hence approaching closer the complete data case ([165]). The hypothesis here is that the closer the value of $\kappa(M)$ is to unity the more accurate the reconstructed field will be. Specifically, ([165]) proposed the following greedy algorithm to solve this (non-trivial) optimization problem:

- (i) Consider placing the first sensor: loop over all possible placement points, evaluate $M$ for each point, and choose the point that minimizes $\kappa(M)$.
- (ii) With the first location set, loop over all possible remaining placement points. For each point, update the mask vector, evaluate $M$, and choose the point that minimizes $\kappa(M)$.
- (iii) Repeat step (ii) for all remaining sensor locations.

We have modified the above algorithm in two ways: First, we have changed step (i) so that instead of a single sensor we always start with $K$ number of sensors,
i.e., equal to the number of retained POD modes in the expansion. This is required as the matrix \(M\) is rank-deficient when the number of sensors is smaller than \(K\) and hence the condition number will take an infinite value. Specifically, we place \(K\) sensors at the extrema of the POD modes as initial conditions and solve for all possible combinations, with the objective of minimizing \(\kappa(M)\). Second, after step (iii) we pursue a further optimization by re-locating one-at-a-time all the sensors, successively starting from the first one. We found that this additional step may lead to even smaller values of condition number, see results below. We note that we search all the grid points for this optimization problem. Other approaches are also possible, e.g. using a combined steepest descent-simulated annealing employed in ([111]).

### 2.5.1 Unsteady flow past a cylinder

To investigate which criterion we should be using in placing the sensors for sampling the POD modes, we consider a benchmark problem, namely two-dimensional flow past a circular cylinder, and perform extensive tests. In particular, we present a comparative study on the sensor locations using two different criteria: (1) the locations of the extrema of POD modes according to [47], and (2) the locations that minimize the condition number \(\kappa(M)\) of the matrix in equation (2.22) according to [165]. To this end, we consider two-dimensional unsteady flow past a circular cylinder with diameter \(D = 1\) in a computational domain as shown in Fig. 2.15. We performed high-resolution spectral/hp element simulations at \(Re = 100\) for which a time-periodic state is achieved. The POD modes are extracted from snapshots taken over one time period; details can be found in [105].

We assume that we have 12 sensors for the limited number of measurements,
and the question is which POD modes to sample and how. For simplicity, we also assume that we will employ only four POD modes for the reconstruction. We have investigated more than 20 cases to determine the best possible sensor configurations but here we present only the 10 most representative ones. In general, for each configuration we start with the extrema of the POD modes (using different variables, e.g., \( u, v, U = \sqrt{u^2 + v^2} \) or \( T \) (temperature)). We then search for locations that minimize \( \kappa(M) \) and evaluate the error of the reconstructed field using as reference solution the results from the full simulation. We summarize the 10 cases in Table 2.3. In case 1 we do not use the greedy algorithm but instead we place the sensors at the extrema of the POD modes of the total velocity \( U \) with the following sensor distribution \((2,2,4,4)\) for the modes \((1,2,3,4)\), respectively. In other words, here we place more sensors in the higher modes \((3 \& 4)\). However, we have investigated other configurations, e.g. \((4,4,2,2)\) or \((3,3,3,3)\). Specifically, we examined another nine cases for which the sensors are placed at the extrema of different POD modes and with different number of sensors per mode; the results are somewhat similar to case 1 so we do not present them here. In case 2 all the sensors are placed on the centerline. For the other cases (3-10) the aforementioned modified greedy algorithm was used to find the best possible locations. For Cases 3 and above we have used different initial configurations as follows. Case 3 uses Case 2 as initial configuration
and then a greedy algorithm is run (i.e., the last part only of the relocation part of the algorithm). Similarly, Case 4 uses Case 3 as initial configuration, and so on. We identified 100 extrema in the POD modes and starting with these as possible initial locations we had to perform extensive searches. Subsequently, we re-located all sensors, one-at-a-time in an iterative fashion, to look for possibly better sensor locations.

In order to compare the various cases we define the reconstruction error as

\[ e_j = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \frac{\int_{\Omega} (\tilde{u}_i - u_i)^2}{\int_{\Omega} u_i^2}}, \]

where \( N \) is the number of snapshots, and \( \tilde{u} \) and \( u \) denote reconstructed and reference fields, respectively. We should also distinguish between the measurement error and the total error (measurement plus truncation). The latter reflects the inaccuracies committed due to the fact that we only employed four POD modes instead of the total number of POD modes available. Hence, we use the subscript \( j \) that takes the value of 1 or 2 for the total and measurement errors, respectively.

In Table 2.4 we summarize the results for the aforementioned 10 cases. Overall, we have observed that a small condition number \( \kappa(M) \) corresponds to small reconstruction errors \( e_2 \) (and hence \( e_1 \)) but the absolutely smallest \( \kappa(M) \) does not correspond to absolutely smallest \( e_2 \). We note that in case 2, with all 12 sensors along the centerline, we obtain a very large condition number and correspondingly a very large reconstruction error. Cases 4, 7 and 10 have different \( \kappa(M) \) but about the same error \( e_2 \). In case 1 (sensors on the extrema of POD modes) – for which no optimization solution is required for \( \kappa(M) \) - we have the second smallest error \( e_2 \) although its condition number is well above 1. We have observed similar trends by considering the POD modes of other variables (e.g., temperature or vorticity)
and placing the sensors as above but following the contour levels of these state variables. We have concluded that configurations corresponding to low condition number should be preferred for low reconstruction error. However, placing the sensors at the extrema of the POD modes of the most important state variable leads to both low reconstruction errors and low condition number without the extra overhead associated with the extra search/optimization algorithm.

In Fig. 2.16 we plot the first four POD modes along with the location of the sensors for case 1; the sensors are placed at the extrema of the POD modes. In Fig. 2.17 we plot the locations of 12 sensors for case 3, the coordinates of which were computed based on the greedy algorithm. Comparing the results shown in the two figures, we see that the sensors are placed in different positions although the reconstruction error is of about the same magnitude, i.e., 6-7%. This, of course, indicates that there is no unique solution to this problem and hence the most efficient method should be used in practice, i.e., the extrema of the POD modes.

![Figure 2.16](image)

**Figure 2.16:** Contours of first four POD modes of the total velocity U and sensor locations for case 1. Mode 1: upper-left; Mode 2: upper-right; Mode 3: lower-left; Mode 4: lower-right.

In summary, the general conclusion we draw from these results is that the extrema of POD modes are very good locations, if not optimum, to place the sensors. At
Table 2.3: Sensor location for 12 sensors. In case 1 $x^i_j$ denotes the coordinate of mode $j$ with $i^{th}$ extrema, with 1 corresponding to largest extremum; the contours of the POD modes of total velocity $U = \sqrt{u^2 + v^2}$ are employed for identifying the extrema.

<table>
<thead>
<tr>
<th>Sensor No:</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case No:</td>
<td>x = $x^1_1$</td>
<td>x = $x^2_1$</td>
<td>x = $x^3_1$</td>
<td>x = $x^4_1$</td>
<td>x = $x^5_1$</td>
<td>x = $x^6_1$</td>
</tr>
<tr>
<td></td>
<td>y = $y^1_1$</td>
<td>y = $y^2_1$</td>
<td>y = $y^3_1$</td>
<td>y = $y^4_1$</td>
<td>y = $y^5_1$</td>
<td>y = $y^6_1$</td>
</tr>
<tr>
<td>1</td>
<td>x = 1.0</td>
<td>x = 1.0</td>
<td>x = 1.0</td>
<td>x = 1.0</td>
<td>x = 1.0</td>
<td>x = 1.0</td>
</tr>
<tr>
<td></td>
<td>y = 1.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
</tr>
<tr>
<td>2</td>
<td>x = 0.62</td>
<td>x = 0.62</td>
<td>x = 0.62</td>
<td>x = 0.62</td>
<td>x = 0.62</td>
<td>x = 0.62</td>
</tr>
<tr>
<td></td>
<td>y = -0.11</td>
<td>y = -0.11</td>
<td>y = -0.11</td>
<td>y = -0.11</td>
<td>y = -0.11</td>
<td>y = -0.11</td>
</tr>
<tr>
<td>3</td>
<td>x = 2.00</td>
<td>x = 2.00</td>
<td>x = 2.00</td>
<td>x = 2.00</td>
<td>x = 2.00</td>
<td>x = 2.00</td>
</tr>
<tr>
<td></td>
<td>y = 0.00</td>
<td>y = 0.00</td>
<td>y = 0.00</td>
<td>y = 0.00</td>
<td>y = 0.00</td>
<td>y = 0.00</td>
</tr>
<tr>
<td>4</td>
<td>x = 1.25</td>
<td>x = 1.25</td>
<td>x = 1.25</td>
<td>x = 1.25</td>
<td>x = 1.25</td>
<td>x = 1.25</td>
</tr>
<tr>
<td></td>
<td>y = -0.2</td>
<td>y = -0.2</td>
<td>y = -0.2</td>
<td>y = -0.2</td>
<td>y = -0.2</td>
<td>y = -0.2</td>
</tr>
<tr>
<td>5</td>
<td>x = 2.06</td>
<td>x = 2.06</td>
<td>x = 2.06</td>
<td>x = 2.06</td>
<td>x = 2.06</td>
<td>x = 2.06</td>
</tr>
<tr>
<td></td>
<td>y = -2.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
</tr>
<tr>
<td>6</td>
<td>x = 0.87</td>
<td>x = 0.87</td>
<td>x = 0.87</td>
<td>x = 0.87</td>
<td>x = 0.87</td>
<td>x = 0.87</td>
</tr>
<tr>
<td></td>
<td>y = -0.48</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
</tr>
<tr>
<td>7</td>
<td>x = 3.12</td>
<td>x = 3.12</td>
<td>x = 3.12</td>
<td>x = 3.12</td>
<td>x = 3.12</td>
<td>x = 3.12</td>
</tr>
<tr>
<td></td>
<td>y = 0.74</td>
<td>y = -0.74</td>
<td>y = 0.74</td>
<td>y = -0.74</td>
<td>y = 0.74</td>
<td>y = -0.74</td>
</tr>
<tr>
<td>8</td>
<td>x = 3.24</td>
<td>x = 3.24</td>
<td>x = 3.24</td>
<td>x = 3.24</td>
<td>x = 3.24</td>
<td>x = 3.24</td>
</tr>
<tr>
<td></td>
<td>y = -0.5</td>
<td>y = 0.5</td>
<td>y = 0.5</td>
<td>y = 0.5</td>
<td>y = 0.5</td>
<td>y = 0.5</td>
</tr>
<tr>
<td>9</td>
<td>x = 0.38</td>
<td>x = 0.38</td>
<td>x = 0.38</td>
<td>x = 0.38</td>
<td>x = 0.38</td>
<td>x = 0.38</td>
</tr>
<tr>
<td></td>
<td>y = 1.17</td>
<td>y = 1.17</td>
<td>y = 1.17</td>
<td>y = 1.17</td>
<td>y = 1.17</td>
<td>y = 1.17</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sensor No:</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case No:</td>
<td>x = $x^1_2$</td>
<td>x = $x^2_2$</td>
<td>x = $x^3_2$</td>
<td>x = $x^4_2$</td>
<td>x = $x^5_2$</td>
<td>x = $x^6_2$</td>
</tr>
<tr>
<td></td>
<td>y = $y^1_2$</td>
<td>y = $y^2_2$</td>
<td>y = $y^3_2$</td>
<td>y = $y^4_2$</td>
<td>y = $y^5_2$</td>
<td>y = $y^6_2$</td>
</tr>
<tr>
<td>1</td>
<td>x = 1.0</td>
<td>x = 1.0</td>
<td>x = 1.0</td>
<td>x = 1.0</td>
<td>x = 1.0</td>
<td>x = 1.0</td>
</tr>
<tr>
<td></td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
</tr>
<tr>
<td>2</td>
<td>x = 0.53</td>
<td>x = 0.53</td>
<td>x = 0.53</td>
<td>x = 0.53</td>
<td>x = 0.53</td>
<td>x = 0.53</td>
</tr>
<tr>
<td></td>
<td>y = -0.53</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
</tr>
<tr>
<td>3</td>
<td>x = 0.61</td>
<td>x = 0.61</td>
<td>x = 0.61</td>
<td>x = 0.61</td>
<td>x = 0.61</td>
<td>x = 0.61</td>
</tr>
<tr>
<td></td>
<td>y = -0.44</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
<td>y = 0.0</td>
</tr>
<tr>
<td>4</td>
<td>x = 1.15</td>
<td>x = 1.15</td>
<td>x = 1.15</td>
<td>x = 1.15</td>
<td>x = 1.15</td>
<td>x = 1.15</td>
</tr>
<tr>
<td></td>
<td>y = 0.41</td>
<td>y = -0.41</td>
<td>y = 0.41</td>
<td>y = -0.41</td>
<td>y = 0.41</td>
<td>y = -0.41</td>
</tr>
<tr>
<td>5</td>
<td>x = 3.15</td>
<td>x = 3.15</td>
<td>x = 3.15</td>
<td>x = 3.15</td>
<td>x = 3.15</td>
<td>x = 3.15</td>
</tr>
<tr>
<td></td>
<td>y = -0.77</td>
<td>y = 0.77</td>
<td>y = -0.77</td>
<td>y = 0.77</td>
<td>y = -0.77</td>
<td>y = 0.77</td>
</tr>
<tr>
<td>6</td>
<td>x = 3.24</td>
<td>x = 3.24</td>
<td>x = 3.24</td>
<td>x = 3.24</td>
<td>x = 3.24</td>
<td>x = 3.24</td>
</tr>
<tr>
<td></td>
<td>y = -0.5</td>
<td>y = 0.5</td>
<td>y = -0.5</td>
<td>y = 0.5</td>
<td>y = -0.5</td>
<td>y = 0.5</td>
</tr>
<tr>
<td>7</td>
<td>x = 2.24</td>
<td>x = 2.24</td>
<td>x = 2.24</td>
<td>x = 2.24</td>
<td>x = 2.24</td>
<td>x = 2.24</td>
</tr>
<tr>
<td></td>
<td>y = -0.52</td>
<td>y = 0.52</td>
<td>y = -0.52</td>
<td>y = 0.52</td>
<td>y = -0.52</td>
<td>y = 0.52</td>
</tr>
<tr>
<td>8</td>
<td>x = 2.42</td>
<td>x = 2.42</td>
<td>x = 2.42</td>
<td>x = 2.42</td>
<td>x = 2.42</td>
<td>x = 2.42</td>
</tr>
<tr>
<td></td>
<td>y = -0.82</td>
<td>y = 0.82</td>
<td>y = -0.82</td>
<td>y = 0.82</td>
<td>y = -0.82</td>
<td>y = 0.82</td>
</tr>
<tr>
<td>9</td>
<td>x = 11.7</td>
<td>x = 11.7</td>
<td>x = 11.7</td>
<td>x = 11.7</td>
<td>x = 11.7</td>
<td>x = 11.7</td>
</tr>
<tr>
<td></td>
<td>y = 0.52</td>
<td>y = -0.52</td>
<td>y = 0.52</td>
<td>y = -0.52</td>
<td>y = 0.52</td>
<td>y = -0.52</td>
</tr>
<tr>
<td>10</td>
<td>x = -0.78</td>
<td>x = -0.78</td>
<td>x = -0.78</td>
<td>x = -0.78</td>
<td>x = -0.78</td>
<td>x = -0.78</td>
</tr>
<tr>
<td></td>
<td>y = -1.44</td>
<td>y = -1.44</td>
<td>y = -1.44</td>
<td>y = -1.44</td>
<td>y = -1.44</td>
<td>y = -1.44</td>
</tr>
</tbody>
</table>

these locations, the condition number $\kappa(M)$, becomes small but not minimum. In fact, minimization of $\kappa(M)$ does not necessarily lead to the smallest reconstruction field.

Next, we present results for the Mass Bay using the most efficient method, i.e., selecting the extrema of POD modes as sensor locations.
Table 2.4: Errors $e_1$ (w.r.t. full simulation) and $e_2$ (w.r.t. POD reconstruction) of the total velocity $U = \sqrt{u^2 + v^2}$.

<table>
<thead>
<tr>
<th>Case No:</th>
<th>Cond. Number $\kappa(M)$:</th>
<th>$e_1$</th>
<th>$e_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4.1932</td>
<td>15.79 %</td>
<td>7.42 %</td>
</tr>
<tr>
<td>2</td>
<td>2.3e+08</td>
<td>66283 %</td>
<td>66939.4 %</td>
</tr>
<tr>
<td>3</td>
<td>1.0359</td>
<td>15.33 %</td>
<td>6.37 %</td>
</tr>
<tr>
<td>4</td>
<td>3.0940</td>
<td>18.66 %</td>
<td>12.49 %</td>
</tr>
<tr>
<td>5</td>
<td>1.0276</td>
<td>17.32 %</td>
<td>10.33 %</td>
</tr>
<tr>
<td>6</td>
<td>1.5024</td>
<td>19.17 %</td>
<td>13.25 %</td>
</tr>
<tr>
<td>7</td>
<td>1.0152</td>
<td>18.38 %</td>
<td>12.05 %</td>
</tr>
<tr>
<td>8</td>
<td>1.4815</td>
<td>15.95 %</td>
<td>7.76 %</td>
</tr>
<tr>
<td>9</td>
<td>1.0327</td>
<td>16.10 %</td>
<td>8.07 %</td>
</tr>
<tr>
<td>10</td>
<td>1.0249</td>
<td>18.16 %</td>
<td>11.71 %</td>
</tr>
</tbody>
</table>

Figure 2.17: Contours of streamwise velocity $u$ and sensor locations for case 3.

2.6 Massachusetts Bay: Results and Discussion

We now apply the modified gappy POD framework in sampling Mass Bay. We will investigate the level of reconstruction errors for a different number of retained POD modes (e.g., different truncations) and also for different number of sensors for both short-term (8-days) and long-term (47-days) simulations. Following the findings from our cylinder investigation, we will place the sensors at the extrema of the POD modes; a typical result is shown in Fig. 2.18. We still have many choices regarding the sensor distribution per mode; here, we will only show a subset of our results. We use as guiding modes the ones corresponding to the total velocity $U(= \sqrt{u^2 + v^2})$
and also the temperature. The sensor configurations for all cases are given in Table 2.5. Each box in the table contains information on how the sensors are distributed per mode; for example, $2x - 2x - 4x - 4x$ for the four-modes in Case 1 implies that $(2,2,4,4)$ sensors are used for modes $(1,2,3,4)$, respectively, if $x = 1$ or $(4,4,8,8)$ if $x = 2$, depending on the number of the available sensors (12 and 24, respectively).

![Figure 2.18: Mass Bay: Schematic of best sensor locations for Case 1 (see Table 2.5: 4 modes, 12 sensors). Only a slice at the surface of the first POD mode is shown; different symbols correspond to different modes. The extrema are located at some depth from the surface. The contours represent the second POD mode of the temperature.](image)

We will measure the time-averaged reconstruction errors using the following definition:

$$e_j = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \frac{\int_{\Omega} (\tilde{\mathbf{u}}_i - \mathbf{u}_i)^2}{\int_{\Omega} (\mathbf{u} + \mathbf{u}_{avg})^2}}, \quad \text{for } j = 1, 2,$$

where the subscript “$i$” denotes snapshot number. The reference state $\mathbf{u}$ is meant to be a complete HOPS solution for $j = 1$ and reconstructed POD solution for $j = 2$ with as many modes $(4, 6, 8, 16)$ as we indicate in each case in Table 2.5. We note
that in the definition of the errors $e_1, e_2$ we normalize by the total magnitude, i.e.,
we include the velocity $\hat{\mathbf{u}}_{\text{avg}}$.

Table 2.5: Sensor configurations for Mass Bay. In the table-boxes, N x means N times x, with x
a multiplier adjusted for the total number of sensors.

<table>
<thead>
<tr>
<th>Case No:</th>
<th>Total Modes:</th>
<th>Four Modes</th>
<th>Six Modes</th>
<th>Eight Modes</th>
<th>Sixteen Modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case 1</td>
<td>2x-2x-4x-4x</td>
<td>2x-2x-4x-4x-6x-6x</td>
<td>2x-2x-4x-4x-6x-6x-8x-8x</td>
<td>2x-2x-4x-4x-6x-6x-8x-8x-10x-10x-12x-12x-14x-14x-16x-16x</td>
<td></td>
</tr>
<tr>
<td>Case 2</td>
<td>3x-3x-3x-3x</td>
<td>4x-4x-4x-4x-4x-4x</td>
<td>5x-5x-5x-5x-5x-5x</td>
<td>9x-9x-9x-9x-9x-9x-9x-9x-9x-9x-9x-9x-9x-9x</td>
<td></td>
</tr>
<tr>
<td>Case 3</td>
<td>4x-4x-2x-2x</td>
<td>6x-6x-4x-4x-2x-2x</td>
<td>8x-8x-6x-6x-4x-4x-2x-2x</td>
<td>16x-16x-14x-14x-12x-12x-10x-10x-8x-8x-6x-6x-4x-4x-2x-2x</td>
<td></td>
</tr>
</tbody>
</table>

2.6.1 Total velocity

We first present errors for the short-term simulation using 4, 6 and 8 POD modes in
the expansion. For the 4-modes case, the results are given in Fig. 2.19. The error $e_1$
is about 10.5% for 12 sensors and 9.5% for 24 sensors. The corresponding truncation
error $\tilde{e}_1$ is about 8.5% and hence it dominates. In Fig. 2.20, we show results for the
6-modes case and we see that the $e_1$ error is reduced to about 7.2% for 24 sensors
and 6.8% for 48 sensors. Clearly, using twice as many sensors is not an effective way
to reduce the error since the truncation error dominates. In fact, we can use only
40 sensors and instead increase the number of modes to 8 leading to a greater error
reduction, i.e., the error $e_1$ decreases to about 4.5%, see Fig. 2.21. More specifically,
the error $e_2$ that excludes the truncation error is only 2% in this case; however,
doubling the number of sensors from 40 to 80 decreases the error slightly to about 1.5% (for Cases 1 and 2 only).

What we have seen so far from the short-term simulation results is that we have to look for the appropriate combination of number of modes and number of sensors for effective error reduction and efficient utilization of the resources. We will apply the same approach to results from long-term simulations, where we will now use 4, 8, and 16 modes to reconstruct new fields and we also assume that we can use (12,24) (40, 80) and (144,288) sensors, respectively.

In Fig. 2.22, reconstruction errors for the 4-modes long-term integration are shown. The corresponding truncation error is \( \tilde{e}_1 = 31.7\% \), which is a lower limit for the reconstruction errors \( e_1 \). The error \( e_1 \) for all three cases is about 36% for 12 sensors; increasing the total sensors to 24, \( e_1 \) is lowered by only 2%. These are large errors although the corresponding values of the condition number are low, e.g., 7.10 and 8.12 for Case 3 corresponding to 12 and 24 sensors. For the 8-modes case (results not shown here), the difference between \( e_1 \) and the truncation error \( \tilde{e}_1 \) has dropped to about 3% with 40 sensors and about 2% with 80 sensors. In comparison with the 4-modes results, the error dropped about 16% more for the 8-modes case. Doubling the number of modes to 16, see Fig. 2.23, yields an \( e_1 \) error of about 21% which is only 2% above the truncation error.

In summary, comparing the short-term and long-term simulation we can appreciate how the reconstruction errors are dependent on the dimensionality of the system. For adaptive sampling in mesoscopic regions, the short-term results are more relevant, however we include the long-term results as an extreme case. However, even in this case, the reconstructed field has all the salient features of the original field as it is shown in the comparison in figure 2.24.
Figure 2.19: Short-term simulation, 4-modes: Time averaged errors for total velocity. The total of four modes contain 92.6% of total energy and the corresponding truncation error is $\tilde{e}_1 = 8.5\%$. For 12 sensors $\kappa(M) = 14.82, 10.54, 9.55$ for Cases 1, 2 and 3, respectively.

Figure 2.20: Short-term simulation, 6-modes: Time averaged errors for total velocity. The total of six modes contain 96.1% of total energy and the corresponding truncation error is $\tilde{e}_1 = 6.2\%$. For 24 sensors $\kappa(M) = 18.92, 17.74, 16.54$ for Cases 1, 2 and 3, respectively.

2.6.2 Temperature

We have also applied the gappy POD method in reconstructing the temperature field for short-term and long-term simulations. The reconstructed fields for temperature are much more accurate than the velocity due to the lower dimensionality of the temperature field. Specifically, in the short-term simulation, only four modes contain 97.9% of total energy (truncation error about 0.7%) whereas in the long-term simulation four modes contain 65.7% of the total energy (truncation error 5.0%); this is a larger percentage than the corresponding 53.2% total velocity energy content. Using 12 sensors, the reconstruction error $e_1$ is about 1% and 6% for the short-term
Figure 2.21: Short-term simulation, 8-modes. Time averaged errors for total velocity. The total of eight modes contain 98.1% of total energy and the truncation error is $\tilde{e}_1 = 4.3\%$. For 40 sensors $\kappa(M) = 19.15, 20.37, 19.94$ for Cases 1, 2 and 3, respectively.

Figure 2.22: Long-term simulation, 4-modes. Time averaged errors for total velocity. The total of four modes contain 53.2% of total energy and the truncation error is $\tilde{e}_1 = 31.7\%$. For 12 sensors $\kappa(M) = 20.73, 11.33, 7.10$ for Cases 1, 2 and 3, respectively.

and long-term simulations, respectively. On the other hand, if we employ 16 POD modes, the truncation error $e_1$ for the long-term simulation drops to 2.4% and the reconstruction error (using 144 sensors) drops to about 2.6%; it is negligible for the short-term simulation.

Here we present detailed results for eight modes as shown in Fig. 2.25 for the short-term and long-term simulations. In the former, the truncation error is 0.3% while in the latter is 3.6%. We use 40 sensors for each simulation distributed according to the aforementioned three different cases. We see that for the short-term simulation, the reconstruction error in the temperature field is slightly above the
truncation error and no greater than 0.5% for all three cases whereas for the long-term simulation the reconstruction error is about 4% due to the larger truncation error. The condition number $\kappa(M)$ is smallest for Case 3 for both short-term and long-term simulations (5.37 and 10.59, respectively).

While the previous plots provide a quantitative assessment of the reconstruction errors, in Fig. 2.26 we present a more global but qualitative comparison of the temperature field between the original data (“the ground truth”) and the reconstructed field at one time instant. The agreement is very good and it is typical for all other time instants as well.

### 2.7 Summary and Discussion

In this paper we have addressed the question of how to identify efficiently the “best” locations for sampling the ocean state for the most accurate reconstruction of the velocity, temperature and salinity fields from a limited number of measurements, in the context of fast adaptive sampling. To this end, we first performed simulations.
of three different regions using the mesoscale ocean models ROMS, HOPS and FV-COM and analyzed their outputs using proper orthogonal decomposition (POD). Specifically, we investigated both short-term and long-term simulation outputs and obtained the corresponding POD eigenvalues and eigenmodes. Our results indicate that the “numerical” ocean is low-dimensional and hence a handful of modes is sufficient to describe the essential dynamics. In particular, we demonstrated that even the long-term dynamics simulated with HOPS for Mass Bay has dimensionality lower than the dynamics of the turbulent wake at \( Re = 10,000 \) exhibiting periodic shedding. This, in turn, suggests that in reconstructing an ocean state it is advantageous
to capture the most energetic modes, i.e., perform sampling in modal space rather than in physical space. We employed POD and investigated two different strategies for sensor placement. In the first approach, the extrema of the POD modes are selected as the sensor locations whereas in the second one, an iterative procedure is set up that aims to minimize the condition number of a matrix involved in the POD approach. We found that the latter approach is more expensive than the former and does not lead to any significant accuracy gains. These are not the only possible optimization approaches, and one can use the reconstruction error directly as alternative cost function or set up different minimization procedures using other sets of sampling points, e.g. see ([113]). As stated in ([113]), there can be several sets of “best” points as uniqueness is not guaranteed. For adaptive sampling in forecasting the ocean state, in particular, the simplest and most efficient approach is the desirable one.

Based on the extrema of POD modes, we reconstructed the velocity field for the Mass Bay for 8-days and 47-days simulations. We found that for the 8-days simulation even a relatively small number of sensors (e.g., 12) can give an accurate reconstructed velocity field. However, for a 47-days simulation even for a large number of sensors the reconstruction errors are large. The latter case is not representative.
of adaptive sampling but we included it in order to investigate how the POD approach behaves in that extreme limit. It also points to the connection we attempted to make between the dimensionality of the system and the particular scheme of data assimilation that needs to be employed.

The accuracy of our method is shown to strongly depend on the number of employed POD modes, and hence it is important how to distribute the available sensors among the POD modes to be sampled. Another question is which state variables to use in a multi-physics simulation involving velocity, temperature, salinity and
possibly chemistry or biology. We have found that the total velocity \( U = \sqrt{u^2 + v^2} \) (employed in this paper) but also the vorticity are the best variables. In particular, we simulated scenarios where the extrema of the temperature POD modes were selected as the measurement locations, and we found that the errors in all other fields increased compared to the cases where the extrema of the total velocity or vorticity POD modes were employed. This is expected given that the POD eigenspectra of the velocity field show a broader distribution of the energy (i.e., higher dimensionality) compared to the temperature or the salinity eigenspectra, and hence higher dimensionality. From the practical standpoint, given the currently available resources for multi-field measurements, e.g. CTD (Conductivity-Temperature-Depth), it is therefore better to take measurements at the extrema of the total velocity POD modes even though these may not be the theoretically guaranteed best locations for the other fields.

Another open question is which POD modes to use in the reconstruction. Here in order to focus on the reconstruction error only, we employed POD modes from the full simulation. In practice, such modes will not be available and hence some approximations are required. In the adaptive sampling context, as we advance the ocean computer model in time, such POD modes can be constructed from the snapshots obtained at previous time steps, hence there will be a time lag due to such extrapolation. In preliminary tests we performed we found differences less than 5%, in agreement also with the results of ([111]). In general, given the small time steps involved in the mesoscale ocean models we do not anticipate any major error contributions from this time extrapolation. Specifically, let us assume that we have a time window of two to three hours to deploy our sensors; using snapshots from the previous hour to obtain the POD modes will not significantly increase the reconstruction errors. This is also justified by our results for short-term integration that
show extremely small errors in the short-term dynamics runs.

Another issue that requires a physical explanation is why the extrema of POD modes are such important locations, and specifically how are these locations related to the uncertainty fields employed, e.g., in HOPS via the ESSE (Error Subspace Statistical Estimation) system ([101]). In work not presented here, we performed direct comparisons with the ESSE approach for the Mass Bay, and we found the POD extrema coincide or are very close to the locations of maximum uncertainty employed in the ESSE data assimilation scheme. From the approximation theory standpoint, one can draw an analogy with non-Fourier spectral theory where the maxima of the Chebyshev polynomials are the collocation points (the so-called Gauss-Lobatto points), based on which the most accurate approximations are guaranteed – we refer here to the so-called “minimax theorem”, see ([61]). Clearly the POD modes are not necessarily polynomials but they can be approximated by spectral polynomials.

Finally, we emphasize that the “measured data” in this study are of “infinite” accuracy, however, in practice we have to incorporate the uncertainty in our measurements and propagate it through the computer ocean model along with other parametric uncertainties of the model. Clearly, further systematic studies are required to resolve these issues before the use of our proposed approach in future adaptive sampling experiments.
CHAPTER THREE

Numerical Ocean Wave
Waves are well-known important phenomena in the ocean. The classification of waves [92] is done in many different ways according to (1) the length of the periods, (2) the restoring forces, and (3) the depth of the water. Kinsmann [92] sketched energy scales versus frequency (see Figure 3.2) for all ranges of waves in the ocean to illustrate the various groups of waves. Capillary waves, gravity waves, and long period waves are three important main categories. The smallest period of waves are capillary waves ($T < 0.1 \text{ s}$), the next level waves are the gravity waves ($0.1 \text{s} < T < 30 \text{ s}$), then the waves are long period waves ($5 \text{ min} < T < 24 \text{ hr}$) (including tsunami, tidal waves), and finally the greatest wave periods of all are transtidal waves (Rossby waves) ($24 \text{ hr} < T$).

The previous classifications were based on the length of the period. A similar categorization can be done if we consider restoring force into the system. The surface tension is primary force (as restoring forces) in the capillary waves. The gravitational and pressure forces are leading forces in the gravity waves. The wave length of long and transtidal waves can easily be of the order of 100 km, which will induce a large mass of the water motion; therefore, the Coriolis force is no longer negligible. The Coriolis force becomes an important driving force in the long and transtidal waves. This classification has some gray areas such that surface/gravity forces, gravity/Coriolis forces are in the same order.

Depth is another classification of waves. Waves induce Lagrangian motion of water particles in the vertical plane and these motions die out. Otherwise, they will modify the surface forces (shallow-water). The criteria for the shallow water or deep water waves (they are characteristically different waves) are based on the ratio of the wave length $L$ and the water depth $d$. If the waves have small values of the ratio $L/d$, we call them deep-water waves and if the ratios are large, we classify them as shallow-water waves. The water depth $d$ alone might not be adequate for this
classification. As an example, tsunami waves in the ocean might be perceived as deep-water waves since the ocean is very deep. However, tsunami waves have very large wave length $L$, which makes the ratio $L/d$ large, then tsunamis are classified as shallow water waves. As another example, the ripples in a very shallow pond, which might be perceived as shallow-water, are classified as deep-water waves since the small values of the ratio $L/d$.

The wind generated gravity waves are interesting to us in this study. The amount of energy transferred from wind (the largest energy source in the sea and much higher than seismic or sun/moon activities) is done through the gravity waves (see Figure 3.2). The gravity waves have neglected Coriolis (time scales too short for Coriolis) and capillary forces; however, the gravity and pressure forces are kept. We have also the assumption that the wave amplitudes are small. These assumptions lead us to the linearized wave theory for the gravity waves. This assumption is the basis of linear wave theory (Airy wave theory [7]). The linear wave theory is based on assumptions such that two-dimensional inviscid and incompressible flow with very small amplitudes (the ratio of wave height $H$ to the wave length $L$ ($H/L << 1$) and the ratio of the wave height $H$ to the water depth $d$ ($H/d << 1$)). The governing equation is obtained as follows. Using velocity potential ($\phi(x, z, t)$), we can define x-velocity component ($u(x, z, t) = -\frac{\partial \phi}{\partial x}$) and z-velocity component ($w(x, z, t) = -\frac{\partial \phi}{\partial z}$) (see Figure 3.1). Then the mass conservation law ($\nabla \cdot u = 0$) is turned into the Laplace equation $\nabla^2 \phi = 0$. We rewrite the momentum equation in terms of velocity potential $\phi$ and apply zero pressure $p$ boundary conditions at the free surface $\eta$ ($-\frac{\partial \phi}{\partial t} + g\eta = 0$, at $z = 0$). Applying boundary conditions at $z = 0$ rather than $z = \eta$ is justified from the small amplitude assumption ($\eta \to 0$). There are two more kinematic boundary conditions to apply: (1) vertical velocity of water particles ($w$) and the wave ($d\eta/dt$) must be equal at the free surface ($w = \frac{d\eta}{dt} = \frac{\partial \eta}{\partial t} + u \frac{\partial \eta}{\partial x}$, at $z = 0$)
and (2) inviscid wall boundary condition at the bottom \((z = -d)\) \((w = -\frac{\partial \phi}{\partial z} = 0, \text{ at } z = -d)\), and then it follows that the solution \(\phi(x, z, t)\) is:

\[
\phi(x, z, t) = \frac{ag \cosh (k(d + z))}{2\omega \cosh (kd)} \cos \left(kx - \omega t\right),
\]

(3.1)

where \(a = H\) is the wave amplitude, \(\omega = 2\pi f\) is the radian frequency, and \(k = 2\pi / L\) is the wave number. The details can be found in these works [146, 171, 82]. The combination of dynamic boundary condition \((-\frac{\partial \phi}{\partial t} + g\eta = 0, \text{ at } z = 0)\) and kinematic boundary condition \((-\frac{\partial \phi}{\partial z} = \frac{\partial \eta}{\partial t})\) yields

\[
-\frac{\partial \phi}{\partial z} = \frac{1}{g} \frac{\partial^2 \phi}{\partial t^2}
\]

(3.2)

We can insert the solution in equation (3.1) into the above equation, then we obtain

\[
\omega^2 = gk \tanh (kd).
\]

(3.3)
The above equation is called dispersion relation, which is an important concept in the wave theory. The dispersion relation stipulates that any one of $\omega$, $k$, and $d$ becomes dependent on the other two variables. Another important definition is the group velocity, which is computed by differentiating the dispersion relation (3.3) in respect to the wave number $k$, that is,

$$c_g = \frac{1}{2} \left[ 1 + \frac{2kd}{\sinh(2kd)} \right] \frac{\omega}{k} \quad (3.4)$$

Waves propagating with the group velocity behave differently in deep and in shallow waters. Waves starting out from deep water will slow down their propagation as soon as they enter in shallow water. This wave transformation is called depth-induced wave shoaling and is simply a direct result of energy conservation; that is, the energy with fast waves in the deep water will be converted to potential energy (by increasing the wave height) due to decelerated waves in the shallow water.

Shoaling also occurs if waves encounter a current field. The current against the waves will decrease the wave propagation velocities. This will subsequently induce the wave height increase (current-induced shoaling). The current along with the waves have an adverse effect such that they increase the wave propagation velocities and thus decrease the wave height.

Bathymetry might change along the wave crest, which is another cause of wave transformation called wave refraction (depth-induced). Through phase velocity relation ($c = \omega/k$ and $c = \sqrt{gd}$ in shallow water), the phase velocities will vary along the wave crest in the case of existing depth gradient. Each point along the wave crest having a different speed will cause refraction of the waves. This can be observed on a coast. One will always see a wave that is parallel to the coastline even if it
originated from an arbitrary direction in the deep ocean. All waves will bend with
the bathymetry counters.

The current can also induce the \textit{wave refraction} (current-induced) in the same
way that depth does. That is, in the presence of the gradient of the current field,
the phase speed will change along the wave crests. The nonuniformity of the phase
speed along the wave crests will bend the wave counters towards with the current
gradient counters.

Other well-known solutions for nonlinear waves are Stokes [148, 49] and solitary
waves [94]. Spectral ocean description is used in this study to simulate the ocean
waves. They are a summation of many linear waves with different directions and
frequencies. Spectral ocean wave description is currently applied to linear waves.
Although not impossible, the spectral description is not easy to extend to nonlin-
ear waves. The assumption of linear waves sounds contradictory to real observation
(ocean wave structure seems quite chaotic) in the sea where many nonlinear pro-
cesses are in place such as nonlinear wave-wave interactions and highly nonlinear
process (locally) white-capping. Fortunately, those nonlinear effects on a large scale
are small; hence, spectral ocean description with harmonic waves still maintains its
legacy in slowly varying a weakly nonlinear field.

Wave prediction models fall into two categories: phased-averaged and phase-
resolving. The phased-averaged wave models solve the energy density (or action
density) to obtain important parameters, such as the significant wave height, through
integration of the energy density. On the other hand, the phase-resolving models
simulate directly the wave height in space and time.

Phase-averaged models are well suited for slowly varying wave fields such as ocean
Figure 3.2: The classification of waves by Kinsman (reproduced with permission from [92]).
Figure 3.3: A description of random sea by summation of many harmonic waves (from Pierson et al. [118]).

waves in deep water, and they are based on the spectral representation of ocean waves. The idea of spectral representation is to superimpose many different linear harmonic waves with different directions and frequencies in order to simulate realistic ocean waves. In this model, the basic quantity is energy density \( E(t, X, \theta, \sigma) \), which is a function of time \( t \), physical space \( X \), and the direction \( \theta \) and frequency \( \sigma \) of the waves. In water waves, the energy density can be shown to relate to energy flux \( F \) through the group velocity \( C \) as \( F = CE \) [162]. Replacing the flux term in the energy conservation equation \( (\partial_t E + \nabla \cdot F = 0) \) will yield an energy density equation \( (\partial_t E + \nabla \cdot (CE) = 0) \). Bretherton and Garett [31] extended the wave energy
equation to the *wave action*, which was first established by Whitham [162, 161] through variational formulation, equation \( \partial_t E/\sigma + \nabla \cdot c_g E/\sigma = 0 \). The new variable wave action \( (E/\sigma) \) is conserved in the presence of nonuniformly moving water (such as the current field) unlike the energy \( (E) \) and it will be more convenient for working with the wave action equation. This equation is the ancestor of phase-averaged models, which were later modified to take into account shoaling and refraction in shallow water and physical sources (wind generation) and sink (white-capping, depth-induced breaking...), as well as conservative wave transformation mechanisms (triplet, quadruplet interactions), which neither add nor extract energy from the system but only distribute energy among frequencies. We will refer to these as, *source terms*. The source terms have been continuously modified over the years either by including a new mechanism or improving the already available source terms. These modifications over the years have been named first, second, third, and recently, fourth generation models. The third generation model, established by the WAMDI group [160], is still today the most widely accepted model in operational ocean codes. Phase-averaged models have been widely used from large oceans to coastal regions with great success. The most well-known are from the European Centre for Medium-Range Weather Forecast (ECMWF) [1], Simulating Waves Near-shore (SWAN) from Delft University [5], and NOAA’s WAVEWATCH [2] models.

Phase-resolving models, unlike phase-averaged, have diversified into many different equations based on depth characteristics, strength of nonlinearities, and bottom slopes such as Boundary Integral models, mild-slope equation models, Boussinesq models, and shallow water equations. The phase-resolving model has advantages over the phase-averaged ones in that it contains inherently nonlinear wave-interaction (triads) and better resolves the wave height in a strongly varying wave field. However, its prohibitive computation cost (much more expensive than phase-averaged
models per unit area) limits it to only small regions (such as near-field of wave-structure interactions). We refer interested readers to Battjes’ [12] for a review of both the phase-averaged and phase-resolving models.

### 3.1 Action Balance Equation

The evolution of the energy equation can be represented (in Lagrangian form) as

$$\frac{dE(\theta, \sigma; x, y, t)}{dt} = S(\theta, \sigma; x, y, t),$$  \hspace{1cm} (3.5)

where the source term is to take account of generation, wave-wave interaction, and dissipation processes to simulate the real ocean. The right hand side is a kind of closure equation which will later be shown, which requires the modeling expressed in terms of energy density $E$. As mentioned before, the wave action ($N = E/\sigma$) which is a conserved quantity unlike energy ($E$) in a nonuniform current field is a more favorable variable than energy. The evolution of the wave action variable (using chain rule), the action balance equation [162, 23] for ocean waves in the Eulerian framework, can be written as (see detailed derivation of group velocities form Holthuijsen [82])

$$\frac{\partial N(\theta, \sigma; x, y, t)}{\partial t} + \frac{\partial c_x N(\theta, \sigma; x, y, t)}{\partial x} + \frac{\partial c_y N(\theta, \sigma; x, y, t)}{\partial y} + \frac{\partial c_\theta N(\theta, \sigma; x, y, t)}{\partial \theta} + \frac{\partial c_\sigma N(\theta, \sigma; x, y, t)}{\partial \sigma} = \frac{S(\theta, \sigma; x, y, t)}{\sigma},$$  \hspace{1cm} (3.6)
where \( N(\theta, \sigma, x, y, t) \) is the action density defined as the ratio of energy \( E(\theta, \sigma, x, y, t) \) to relative frequency \( \sigma \) (\( N = E/\sigma \)), \( c_x \) and \( c_y \) are the propagation velocities of wave energy in physical \((x - y)\) space (see Figure 3.4), and \( c_\theta \) and \( c_\sigma \) are the propagation velocities in spectral \((\theta \in [-\pi, \pi], \sigma \in [0, \infty])\) space (see Figure 3.4). The source term \( S(\theta, \sigma; x, y, t) \) takes into account wave generation, dissipation, and interaction mechanisms. The energy density is not conserved but the action density is conserved in the presence of a spatially variable current. The action density propagation velocities can be derived by differentiating the dispersion relation

\[ \sigma^2 = g |\vec{k}| \tanh(|\vec{k}| d), \quad \text{where} \ d = d(x, y) \ \text{is depth,} \ \vec{k} = (k_x, k_y) \quad (3.7) \]

with respect to wave number \( k \), gives

\[ \frac{d\vec{k} = (x, y)}{dt} = (c_x, c_y) = \vec{c}_g + \vec{u} = \frac{1}{2} \left( 1 + \frac{2|\vec{k}|d}{\sinh(2|\vec{k}|d)} \right) \frac{\sigma \vec{k}}{|\vec{k}|^2} + \vec{u}. \quad (3.8) \]

The propagation velocities in spectral space are

\[ \frac{d\theta}{dt} = c_\theta = -\frac{1}{k} \left( \frac{\partial \sigma}{\partial d} \frac{\partial d}{\partial m} + \vec{k} \cdot \frac{\partial \vec{u}}{\partial m} \right), \ \text{and} \quad (3.9a) \]
\[ \frac{d\sigma}{dt} = c_\sigma = \frac{\partial \sigma}{\partial d} \left( \frac{\partial d}{\partial t} + \vec{u} \cdot \nabla d \right) - c_g \vec{k} \cdot \frac{\partial \vec{u}}{\partial s}. \quad (3.9b) \]

where \( s \) is the space co-ordinate in the wave propagation direction of \( \theta \), \( m \) is a coordinate perpendicular to \( s \), sketched in Figure 3.4, and \( \vec{u} \) is a current velocity vector \((\vec{u} = (u_x, u_y)). \) Note that the propagation of action density in frequency
space only occurs if there is a current in the domain (assuming that the bathymetry does not change with time).

The propagation velocities in spectral space are indeed useful to obtain in terms of Cartesian coordinates for the implementation. The derivatives in respect to wave propagation direction \( (s) \) and their perpendicular coordinate \( (m) \) are related to Cartesian coordinate derivatives as follows:

\[
\begin{bmatrix}
\frac{\partial (\bullet)}{\partial s} \\
\frac{\partial (\bullet)}{\partial m}
\end{bmatrix}
= \begin{bmatrix}
\cos(\theta) & \sin(\theta) \\
-\sin(\theta) & \cos(\theta)
\end{bmatrix}
\begin{bmatrix}
\frac{\partial (\bullet)}{\partial x} \\
\frac{\partial (\bullet)}{\partial y}
\end{bmatrix}.
\]

The inner product of a wave vector \( \vec{k} \) with partial derivatives of current \( \vec{u} \) in respect to the coordinate \( s \) :

\[
\vec{k} \cdot \frac{\partial \vec{u}}{\partial s} = k_x \frac{\partial u_x}{\partial s} + k_y \frac{\partial u_y}{\partial s}.
\]

In a similar way, the inner product for the current with a derivative of coordinate \( m \) can be obtained.

The dispersion relationship between frequency \( (\sigma) \) and wave number \( k \) is

\[
\sigma^2 = g k \tanh(k d).
\]

The differentiation of the dispersion relation Eq. (3.12) in respect to depth \( (d) \) will give us frequency change in depth as
\[
\frac{\partial \sigma}{\partial d} = \frac{k \sigma}{\sinh(2k d)}.
\] (3.13)

The propagation velocity \( (c_\theta) \) in Eq (3.9a) using Eqs. (3.10), (3.11), and (3.13) will take the following form in the Cartesian coordinate:

\[
c_\theta = \frac{\sigma}{\sinh(2k d)} \left( \sin(\theta) \frac{\partial d}{\partial x} - \cos(\theta) \frac{\partial d}{\partial y} \right) + \sin^2(\theta) \frac{\partial u_y}{\partial x} + \sin(\theta) \cos(\theta) \left( \frac{\partial u_x}{\partial x} - \frac{\partial u_y}{\partial y} \right) - \cos^2(\theta) \frac{\partial u_x}{\partial y}. \] (3.14)

And, the propagation velocity in a frequency direction will be:

\[
c_\sigma = \frac{k \sigma}{\sinh(2k d)} \left( \frac{\partial d}{\partial t} + u_x \frac{\partial d}{\partial x} + u_y \frac{\partial d}{\partial y} \right) - c_g |\vec{k}| \left[ \cos^2(\theta) \frac{\partial u_x}{\partial x} + \sin(\theta) \cos(\theta) \left( \frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x} \right) + \sin^2(\theta) \frac{\partial u_y}{\partial y} \right]. \] (3.15)

The derivatives of depth and current velocity in respect to Cartesian coordinates have been computed using collocation differentiation [91] to ensure that a high-order of scheme is preserved.

The action balance equation (3.6) is expressed in the Cartesian coordinate system, which might not be accurate for large scale \( (> 10^6m) \) wave simulations due to the curvature of the earth’s surface. The correction can be easily done by transforming
Figure 3.4: (a) An illustration of geophysical and spectral spaces. The frequency direction \((\sigma = 2\pi f)\) is truncated from \([0, \infty]\) to \([0, 2.5]\) Hz and the directional coordinate \((\theta)\) covers full circle \([0, 2\pi]\). (b) An illustration of coordinates \(m\) and \(s\).

3.2 Modeling the Source Terms

Phase-averaged models represent wind generation; white-capping; nonlinear wave interaction; depth-induced breaking, and bottom friction mechanisms as source terms in an action balance equation (3.6) to predict ocean waves. Wind generation, nonlinear wave interaction (quadruplet), and white-capping (dissipation) are wave mechanisms in deep water only. The role of each of the physical source mechanisms differs according to the field of application such as shelf sea, near-shore, or harbor. We refer interested readers to Battjes’ review [12] on physical processes for a broad class of applications. We will next briefly mention the source terms in this study.
3.2.1 Source generation by wind \((S_{in})\)

Most people have observed the phenomenon of wave generation by either throwing pebble stones into a pond or blowing air over water in a pot. Though these are two distinct ways of wave generation waves, they are the same in essence; namely, both actions uniquely change the stress distribution on the water surface [171]. Ocean waves are of interest here because they are generated by exchanging energy between air and water (mostly air to water). The third generation of wind input is based on Philips (initial wave growth) and Miles theories [110] predicting the exchange of energy rate from wind [160] as in the following form:

\[
S_{in}(\theta, \sigma) = A + B\; E(\theta, \sigma),
\]

where \(A\) and \(B\) are, respectively, linear and exponential growth terms. We used the WAMDI group for the parametrization of the wind source term [93]. The linear wave growth \(A\) reads

\[
A = \frac{1.5 \times 10^{-3}}{2\pi g^2} (U_*\max[0, \cos(\theta - \theta_w)])^4 H,
\]

where \(\theta_w\) is wind direction, and \(H\) is an exponential filter \((\exp(-\sigma/\sigma^*)^{-4})\). The peak frequency of Pierson & Moskowitz \((\sigma_{PM}^*)\) [119] may be computed by using friction velocity \(U_*^2 = C_D U_{10}^2\) as

\[
\sigma_{PM}^* = \frac{0.13g}{28U_*} \frac{2\pi}{2},
\]

(3.17)
with drag coefficient [167] defined as

\[
C_D(U_{10}) = \begin{cases} 
1.2875 \times 10^{-3}, & \text{for } U_{10} < 7.5 \text{ m/s} \\
(0.8 + 0.065 \text{s/m} \times U_{10}) \times 10^{-3}, & \text{for } U_{10} < 7.5 \text{ m/s}.
\end{cases}
\]  (3.18)

The exponential growth by wind is formulated by WAMDI [160] using the results from Snyder et al. [145] and the friction velocity definition of Komen [93] in the third generation model as follows:

\[
B = \max \left[ 0, 0.25 \frac{\rho_a}{\rho_w} \left( \frac{28}{c_{ph}} U_* \cos(\theta - \theta_w) - 1 \right) \right] \sigma,
\]  (3.19)

where \( \rho_a \) and \( \rho_w \) denote, respectively, the densities of air and water. We have plotted wind source terms computed by our code for a JONSWAP spectrum (see equation (B.3)) [72] with \( H_s = 3.5 \text{ m} \) and \( T_{\text{peak}} = 7 \text{ s} \) in Figure 3.5. As seen in the figure, the wind source term feeds the positive source term into the system. The wind source term sustains energy more to peak-around frequencies and keeps giving energy to higher frequencies but at a slower rate. We will use this JONSWAP spectrum to illustrate all the other source terms in this section.

### 3.2.2 Dissipation by white-capping (\( S_{wc} \))

The energy growth by wind becomes unstable and makes waves which eventually break. This version of breaking other than depth-induced is called white-capping.
and is a major source of energy loss in deep water wave evolution. The highly nonlinear process of white-capping is the least understood mechanism among the source terms [171]. The third generation model proposes a source function [160], which is consistent with the pulse-based model of Hasselmann [73].

The white-capping in the third generation model [160] [73] [93] [82] is modeled as

$$S_{wc}(\theta, \sigma) = -\mu k E(\sigma, \theta),$$  \hspace{1cm} (3.20)

where $\mu$ is
\[ \mu = C_{wc} \left( (1 - n) + \frac{k}{\tilde{k}} \right) \left( \frac{\tilde{s}}{\tilde{s}_{PM}} \right)^{p\tilde{\sigma}} \tilde{k}, \]  

(3.21)

and tilde \( \tilde{\cdot} \) refers to a mean quantity of variables. The mean wave steepness \( \tilde{s} \) is defined as \( \tilde{s} = \tilde{k}\sqrt{E_{tot}} \) and it is normalized by the Pierson-Moskowitz spectrum [119] with value \( \tilde{s}_{PM} = \sqrt{3.02 \times 10^{-3}} \). The remaining parameters \( C_{wc}, n \) and \( p \) are tunable coefficients. The mean frequency and wave number is, respectively,

\[ \tilde{\sigma} = \left( E_{tot}^{-1} \int_{0}^{2\pi} \int_{0}^{\infty} \sigma^{-1} E(\theta, \sigma) d\theta d\sigma \right)^{-1}, \quad (3.22a) \]

\[ \tilde{k} = \left( E_{tot}^{-1} \int_{0}^{2\pi} \int_{0}^{\infty} k^{-1/2} E(\theta, \sigma) d\theta d\sigma \right)^{-2}. \quad (3.22b) \]

An example of a whitecapping source term is plotted in Figure 3.6 along with the JONSWAP spectrum. The figure confirms that the whitecapping source term will introduce dissipation into a wave such that the dissipation is proportional to the energy level of the frequency of the wave. Alternative formulations are also given in the SWAN technical manual (www.fluidmechanics.tudelft.nl/swan).

### 3.2.3 Nonlinear wave-wave interactions \( (S_{nl}) \)

Spectral ocean wave equation has superimposed many different frequencies and directional linear waves to represent the ocean surface. It is well known that some waves may interact non-linearly with another group of waves in physics, and this is no exception for ocean waves. The interaction among ocean waves was first described
Figure 3.6: The whitecapping source term ($C_{we} = 2.36 \times 10^{-5}$, $n = 0$ and $p = 4$) for the JONSWAP spectrum.

by Hasselmann [69] through perturbation analysis under the assumption that waves are weakly non-linear. His analysis revealed that the nonlinear wave-wave interaction has the same order of magnitude as wind generation and white-capping mechanisms. Hasselmann’s [69] work could not be verified numerically at that time because his description of nonlinear wave-waves was a six-dimensional Boltzmann equation and is computationally very demanding to solve exactly, even today. The Joint North Sea Wave Project (JONSWAP) experiment [72] may be the first validation of his theory. They at least observed a qualitative agreement with the theory. Later, Tanaka [150] verified Hasselmann’s formulation by using a direct numerical simulation of gravity waves up to third-order interactions (three and four wave-waves). The set of wave-wave interactions has been mostly limited to three and four since they are relatively more important than further higher-order resonances, of which
the derivation is too involved and likewise the computation [68]. Contrary to the wind input and white-capping dissipation source mechanisms, nonlinear wave-wave interactions neither add nor subtract energy from the system but only redistribute energy among all the interacting waves by transferring energy from the high to the lower part of the frequency spectrum. Nonlinear wave-wave interactions have spectrum shape stabilization ability in the case of a sudden change in one of the other source terms [172].

3.2.3.1 Quadruplet wave-wave interactions ($S_{nl4}$)

Hasselmann [69] found that nonlinear wave-wave interaction would be comparable to other source terms (wind input and white-capping dissipation) if the following resonant conditions are satisfied:

\begin{align}
  k_1 + k_2 &= k_3 + k_4, \quad (3.23a) \\
  \omega_1 + \omega_2 &= \omega_3 + \omega_4. \quad (3.23b)
\end{align}

Accordingly, not every subset of four waves interacts but only the ones that satisfy the above conditions. Evolution integral for nonlinear wave-wave interaction is six-dimensional integral and its computational cost is several orders higher than that of ocean simulation at the operational level. The computation of the Boltzmann integral has been an active research area for three decades [172]. The third generation model employs discrete interaction approximation (DIA) [76] for evaluating the Boltzmann Integral. The basis of this method relies on findings from
Hasselman’s work [71]. He showed that neighboring wave number interactions have the strongest effect on the evolution of waves. The parametrization of nonlinear wave-wave interaction selects only one configuration satisfying the above resonant conditions. Despite the fact that this is a very crude approximation, its behavior meets practical applications successfully in realistic applications. The DIA has the same number of free parameters as used in spectral space; hence, its computation is as cheap as other source terms. However, this approximation has some deficiencies such as generating a larger negative lobe than the exact one, predicting a spectral width that is too large, and transferring too much energy to higher frequencies (there is too much irregularity in the source) [156].

This study used the Laplacian filter in spectral space, which is the only difference from the original DIA computation.

Hasselmann [69] has derived a six-dimensional Boltzmann integral to describe nonlinear wave-wave interaction. The computation of the Boltzmann equation is prohibitively expensive to use in operational sea models. Hasselmann [76] has proposed various approximations to the Boltzmann equation. Among them, Discrete Interaction Approximation (DIA), which considers only two of the most significant configurations, is adapted in the third generation models. The first quadruplet configuration is

\[
\begin{align*}
\sigma_1 &= \sigma_2 = \sigma, \\
\sigma^+ &= \sigma(1 + \lambda), \\
\sigma^- &= \sigma(1 - \lambda).
\end{align*}
\]
The energy contributions due to the wave interactions among the three resonant waves at $\sigma$, $\sigma^+$ and $\sigma^-$ will be

$$
\begin{pmatrix}
\delta_{nl4}^-
\delta_{nl4}^+
\delta_{nl4}^+
\end{pmatrix} = R \begin{pmatrix}
-2 \\
+1 \\
+1
\end{pmatrix} C_{nl4} (2\pi)^2 g^{-4} \left( \frac{\sigma}{2\pi} \right)^{11} \left[ E^2(\theta, \sigma) \left( \frac{E(\theta^+, \sigma^+)}{(1 + \lambda)^4} + \frac{E(\theta^-, \sigma^-)}{(1 - \lambda)^4} \right) - \frac{E(\theta, \sigma) E(\theta^+, \sigma^+) E(\theta^-, \sigma^-)}{(1 - \lambda)^4} \right],
$$

(3.24)

where the angle ($\theta^+$) between wave numbers $\vec{k}$ and $\vec{k}^+$ is $-11.5^\circ$; the angle ($\theta^-$) between wave numbers $\vec{k}$ and $\vec{k}^-$ is $33.6^\circ$. The second configuration is only a mirror of the above configuration ($k \angle k^+ = 11.5^\circ$ and $k \angle k^- = -33.6^\circ$). The parameter $R$ is a depth scaling parameter [138] for shallow water and $C_{nl4}$ is a constant parameter taken as $3 \times 10^7$. Note that the summation of individual contributions would sum up to zero, so the whole process conserves the total energy. The computation of the above contribution requires energy at locations either out of the spectral domain or off-grid (such that $\theta^+, \sigma^+$). We have assumed zero energy if the directional locations are out of domain and the frequency locations are less than minimum frequency $f_{min}$. The high-frequency energy tail is assumed as $f^{-4}$ to compute energy beyond cut-off frequency $f_{max}$ ($E(\sigma) = E(\sigma_{max}) \ast \left( \frac{\sigma}{\sigma_{max}} \right)^{-4}$). Inside the domain, energy requiring off-grid collocation has been interpolated from the nearest grid collocations. The contribution among low frequencies might fall very close to each other; therefore, it leads to irregular shapes in the source term. We have used a simple filter just averaging a specified neighborhood after the DIA computation. This helped smooth the quadruplet source term.
Figure 3.7 shows the DIA computation of the JONSWAP spectrum. The promise of the DIA is to (try to replicate nonlinear wave-wave interaction) take energy from the lower end of the spectrum to give away at the higher frequency part. The DIA generates three region of lobes, that is, positive, negative, and again positive lobes from left to right in the frequency axis. The negative lobe is exceedingly represented in the DIA approximation, which is much smaller than the one computed exact computation [156].

![Graph](image)

**Figure 3.7**: The nonlinear (Quadruplet) source term ($\lambda = 0.25$ and $C_{nl4} = 3.0 \times 10^7$) computed via DIA method for the JONSWAP spectrum.

### 3.2.3.2 Triad wave-wave interactions ($S_{nl3}$)

Nonlinear wave-wave interactions among three waves (triads) play an important role in shallow water. Triads only occur in shallow water but not in deep water[116]. The resonant conditions among the three waves are
\[ k_1 \pm k_2 = k_3 \]  \hspace{1cm} (3.25a)

\[ \omega_1 \pm \omega_2 = \omega_3. \]  \hspace{1cm} (3.25b)

Triad interactions transfer energy from lower frequency to higher frequency, which is reverse in the quadruplet effect of deep water. Triad interactions generate sub-harmonic (subtraction among the three waves) and super-harmonic (summation among the three waves) waves. These large sub-harmonic waves (infra gravity waves) are excluded in wind generated waves. Primary waves interact with each other and generate first super-harmonic near frequency \( 2f_p \) (\( f_p \) peak frequency), and first super-harmonic waves lead to second super-harmonic, around \( 4f_p \), and so on.

The energy transfer depends on the phase difference of the interacting three wave components. This difference is called bi-phase. The true solution would be that one needs to obtain bi-phase evolution from Boussinesq equations, which describe inherently triad interactions, together with an action equation.

It is desirable to seek an alternative to getting triad interactions without appealing to Boussinesq equations. Eldeberky’s work has proposed Lumped-triad-approximation (LTA) [55] to model triad interactions. The approximation is often used in SWAN code to represent triad source terms. The LTA parametrization for bi-phase used spectral energy and water depth (through Ursell number) to match Beji & Battjes’ laboratory experiments [16]. However, we found that LTA generates too much dissipation in depth-induced breaking validation cases and might be better if turned off in some cases.
The Lumped Triad Approximation (LTA) by Eldeberky [55] is the simplest expression for triad wave interactions. The approximation is

\[ S_{nl3}(\theta, \sigma) = S_{nl3}^{-}(\theta, \sigma) + S_{nl3}^{+}(\theta, \sigma). \]  

(3.26)

And the positive term contribution is computed as

\[ S_{nl3}^{+} = \max\left[0, \alpha_{EB} 2\pi cc g J^2 |\sin \beta| \{E^2(\theta, \sigma/2) - 2E(\theta, \sigma/2)E(\theta, \sigma)\}\right], \]

(3.27)

where \(\alpha_{EB}\) is a control parameter. A negative contribution of source terms can be defined in terms of the positive one as

\[ S_{nl3}^{-} = -2S_{nl3}^{+}(\theta, 2\sigma). \]

(3.28)

The bi-phase parameter \(\beta\) is approximated by using Ursell number \(U_r\):

\[ \beta = -\frac{\pi}{2} + \frac{\pi}{2} \tanh \left(\frac{0.2}{U_r}\right) \]

(3.29)

and

\[ U_r = \frac{g}{8\sqrt{2\pi}} \frac{H_s T_{n01}^2}{d^2}, \]

(3.30)
where the significant wave height \( (H_s) \) and mean wave period \( (T_{m01}) \) are, respectively, defined in Eqs. (C.1) and (C.2).

We have not computed the interaction for very small values of Ursell numbers that occur in deep water (where no triad wave-wave interactions are possible). In a very large Ursell number which occurs in shallow water \( (d << H) \), triad computations will also drop out. The coefficient \( J \) [107] can now be defined as

\[
J = \frac{k^2 \sigma / 2 (g d + 2 c^2 / \sigma)}{k \sigma d (g d + \frac{2}{15} g d^3 k^2 \sigma - \frac{2}{5} \sigma^2 d^2)}.
\]  (3.31)

The computation of the positive and negative terms requires frequency bins beyond the discretization bins. We have assumed zero energy if the energy is out of the spectral domain and interpolated energy inside the domain for off-grid frequency locations \( (\sigma/2) \). However, it will be more appropriate to use shallow water high-frequency tail \( (\approx f^{-3}) \) after cut-off frequency \( f_{\text{cut-off}} \).

The LTA takes energy from a peak-around \( (f_p) \) frequency and gives to higher frequencies (around \( f_{2p}, f_{4p}, \ldots \) ) as shown in Figure 3.8. In the figure, energy is taken from the peak frequency, which is \( 1/7 \) Hz, and feeds the frequencies around \( f_{2p} = 2/7 \) Hz. It is readily seen that the overall contribution of the LTA approximation neither adds energy to the system or extracts the energy from it.

### 3.2.4 Depth-induced breaking \( (S_{br}) \)

Waves approaching coastlines will interact with the bottom in many different ways such as percolation, wave-induced bottom motion, scattering, bottom-friction, and
depth-induced breaking [138] in addition to wave shoaling and refraction. In very shallow water, depth-induced breaking is dominant among other dissipation mechanisms. Wave breaking (depth-induced) is a random process in wind-generated waves and is little understood. We have followed modeling of depth-induced breaking due to Battjes and Janssen (BJ) [13]. BJ modeled the first single breaking wave and extended this result to random fields by applying the probability of breaking waves for a given depth [13][138]. Eldeberky and Battjes [55] extended the bore model to spectral modeling after Beji and Battjes’ experiments (BB) [16], which observed the evolution of a spectral shape to be almost the same for breaking and non-breaking waves. They concluded that breaking waves do not interfere with other mechanisms (such as triad interactions), and total energy energy dissipation does not influence the spectral shape but only the magnitude of total energy [55]. This led them to model depth-induced wave breaking for spectral wave modeling. We next present
this modeling in detail.

The mean rate of energy dissipation due to depth-induced breaking is:

\[
D_{tot} = \frac{1}{4} \alpha_{BJ} Q_b \left( \frac{\tilde{\sigma}}{2\pi} \right) H^2_{max},
\]

where the value of \( \alpha_{BJ} \) is in the order of 1. The fraction of breaking waves is represented by \( Q_b \), which is derived by assuming cumulative probability distributions of all waves [13] [14] from the following relation:

\[
\frac{1 - Q_b}{\log Q_b} = -8 \frac{H_{rms}}{H^2_{max}},
\]

where \( H_{rms} \) is the root-mean-square of wave height and \( H_{max} \) is the maximum wave height for a given depth. The maximum wave height is defined as \( H_{max} = \gamma d \). The breaker parameter \( \gamma \) is chosen as an average value of 0.73 from the work of Battjes and Stive [14]. The characteristic frequency in the original formulation has been chosen as the mean frequency (\( \tilde{\sigma} \)) (see Eq. (3.22a)) in the dissipation rate formula.

The fraction of breaking waves \( Q_b \) can be numerically computed through Eq. (3.33). However, the computation of \( Q_b \) we followed from the SWAN technical manual such that

\[
Q_b = \begin{cases} 
0, & \text{for } \beta \leq 0.2 \\
Q_0 - \beta^2 \frac{Q_0 - \exp(-[(Q_0 - 1)/\beta]^2)}{\beta^2 - \exp(-[(Q_0 - 1)/\beta]^2)}, & \text{for } 0.2 < \beta < 1 \\
1, & \text{for } \beta \geq 1 
\end{cases}
\]
where parameter $\beta$ is the ratio of root-mean-square $H_{rms}$ height and maximum wave height $H_{max}$, and $Q_0$ reads

$$Q_0 = \begin{cases} 
0, & \text{for } \beta \leq 0.5 \\
(2\beta - 1)^2, & \text{for } 0.5 < \beta \leq 1
\end{cases} \tag{3.35}$$

The depth-induced wave breaking source term [55] can be modeled as

$$S_{br}(\theta, \sigma) = \frac{D_{tot}}{E_{tot}} E(\theta, \sigma). \tag{3.36}$$

Note that $S_{br}$ will be a negative sign due to $D_{tot}$. The BJ model for the depth-induced breaking source term has been plotted along with the JONSWAP spectrum in Figure 3.9. As seen in the figure, the model extracts the energy strongly around the peak frequency as the white-capping model did.

### 3.2.5 Bottom friction ($S_{bfr}$)

Bottom friction is modeled from the WAMDI group [160] based on the further simplification of the bottom friction model of the JONSWAP experiment [72] as

$$S_{bfr}(\theta, \sigma) = -C_b \frac{\sigma^2}{g^2 \sinh^2(kd)} E(\theta, \sigma), \tag{3.37}$$

where $C_b$ is the tunable bottom friction coefficient. The modeling of the bottom friction source term assumes that the tangential stress on the bottom can be modeled by
Figure 3.9: The depth-induced breaking source term ($\alpha_{Bf} = 1.0, \lambda = 0.73$) for the JONSWAP spectrum.

A quadratic friction law [74], which has interpretation in terms of the bottom friction source term, leading to the above formulation [72]. The JONSWAP experiment [72] has conducted 10 swell cases and found that the bottom friction coefficient changes from 0.03 to 0.16, with a mean value of 0.038 $\text{m}^2\text{s}^{-3}$. The bottom friction coefficient $C_b = 0.067 \text{m}^2\text{s}^{-3}$ has been used by Bouws and Komen [24] for fully developed waves in shallow water. The coefficient $C_b$ is not constant and depends on the bottom amplitude Reynolds number and the bed roughness. We refer interested readers to Young’s book [171] for further discussion and references.

The source term due to bottom friction plot for the JONSWAP spectrum is presented in Figure 3.10. The source term plot shows that this model extracts energy around the peak frequency and has a similar pattern which we have seen in
the whitecapping and in the depth-induced breaking terms.

Figure 3.10: The bottom friction source term \((C_{bfr} = 0.067)\) for the JONSWAP spectrum.
Chapter Four

Hybrid Spectral/DG Method for Phase-averaged Equation
We will use Fourier-collocation for the spectral space and discontinuous Galerkin (DG) spectral element discretization for the geographical space with Strong Stability Preserving Runge-Kutta (SSP-RK) time discretization. Absorbing boundary conditions are considered for both coastal and frequency domain boundaries. The choice of upwind flux will eliminate having to do any special treatment for absorbing boundary condition in geographical space. Using the absorbing boundary layer (ABL) approach [87] in the frequency direction results in the solution leaving the truncated frequency domain with negligible reflections. We will present the ABL formulation for an action balance equation and also demonstrate its effectiveness. The directional spreading might be so narrow that evenly distributed Fourier-collocation points might be wasteful for a slowly changing region in the spectral direction. Using a mapping technique allows for the local refinement in the directional axis. Arc-tan mapping [28] is used here for local refinement for the spectral directional axis.

4.1 Space Discretizations

To achieve a high-order DG scheme for geographical and spectral spaces, we must deal with effectively four-dimensional basis vectors which may render the computational cost prohibitive. At this point, it is useful to distinguish between the spectral and geographical spaces. Even if we have a four-dimensional problem, spectral ocean wave modeling simply superimposes many waves of different directions and frequencies on the geographical space. It is helpful to think of a fixed geographic location having a corresponding spectral space (such that $f(\theta, \sigma, X_0)$) or a fixed point in spectral space with associated geographical space (e.g. $f(\theta_0, \sigma_0, X)$) (see Figure 3.4). The geographical space demands a complicated boundary in most practical applications (e.g. coastal lines) whereas a spectral space requires a very regular grid. In spectral
spaces, the directional domain extends from $-\pi$ to $\pi$ and the frequency from 0 to $\infty$. A frequency domain is usually truncated to $(\sigma_{\text{min}}, \sigma_{\text{max}})$. Here, we propose a hybrid scheme with Fourier-collocation for spectral space derivatives and DG discretization for the geographical space. In this way, we can exploit the flexibility of unstructured spatial grid dealing with complex coastal lines using two-dimensional Jacobi polynomials (tensor-product expansion in a triangle [91]) and the efficiency of Fourier-collocation in spectral space.

Let us now define the following fluxes with the transformation of the frequency coordinate $\sigma \rightarrow \zeta = \pi \left[ \frac{2(\sigma - \sigma_{\text{min}})}{\sigma_{\text{max}} - \sigma_{\text{min}}} \right]$:

$$E = c_x N(\zeta, \theta, x, y, t), \quad F = c_y N(\zeta, \theta, x, y, t),$$
$$G = c_\theta N(\zeta, \theta, x, y, t), \quad H = c_\zeta N(\zeta, \theta, x, y, t).$$

With the above definitions, the action balance equation is written in flux form, i.e.,

$$\frac{\partial N}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} + \frac{\partial G}{\partial \theta} + J_{\zeta \sigma} \frac{\partial H}{\partial \zeta} = \frac{S}{\sigma},$$

(4.1)

where $J_{\zeta \sigma} = 2\pi/(\sigma_{\text{max}} - \sigma_{\text{min}})$. We use Fourier-collocation in a directional domain extending from $-\pi$ to $\pi$ due to periodicity. However, a straightforward Fourier-collocation for the frequency direction is not possible since we are considering a truncated domain, which is non-periodic. The truncation ends should not reflect the solution back to a computational domain. Even if initial conditions are in general periodic (almost zero at $\sigma = \sigma_{\text{min}}$ and $\sigma_{\text{max}}$), the converged solutions will not necessarily be periodic due to source terms and propagation velocities. Fortunately, using the
absorbing boundary layer approach [87] still allows us to employ Fourier-collocation in a frequency direction (see section 4.1.4) without degrading the exponential spectral accuracy. Let us discretize the wave direction $\theta$ as

$$
\theta_k = -\pi + \frac{2\pi}{N} k \quad \forall k \in [0, 1, ..., N - 1]
$$

$$
\zeta_l = -\pi + \frac{2\pi}{N} l \quad \forall l \in [0, 1, ..., N - 1]
$$

Then using Lagrangian interpolation [60, 78], we can compute

$$
\frac{\partial G}{\partial \theta} \bigg|_{\theta = \theta_k} = \sum_{j=0}^{N-1} D_{kj} G(\zeta, \theta_j, x, y, t), \quad (4.2)
$$

where $D_{kj}$ is the differentiation matrix for Fourier-collocation. Similarly, we obtain

$$
\frac{\partial H}{\partial \zeta} \bigg|_{\zeta = \zeta_l} = \sum_{j=0}^{N-1} D_{lj} H(\zeta_j, \theta, x, y, t). \quad (4.3)
$$

Note that the derivative of both fluxes $G$ and $F$ can be more efficiently computed by Fast Fourier Transform (FFT) ($N \log(N)$) than the matrix-vector product ($N^2$). We have implemented both methods for flux differentiations; however, we will observe a faster code if we use the matrix method until the number of collocations is above 256. We have found that the matrix method still works faster than FFT in the range of a practical resolution. This study used the matrix method to differentiate spectral space derivatives for all computations throughout the paper. The differentiation matrix [78] for an even number of collocation points (Lagrangian polynomial if $q = 0$)
reads in a dummy coordinate \( z \) (either \( \theta \) or \( \zeta \) to obtain the above flux derivatives)

\[
D_{ij}^q = \frac{2}{N} \sum_{n=0}^{N/2} \frac{1}{cn} \begin{cases} 
(i n)^q \cos[n(z_i - z_j)], & q \text{ even} \\
(i n)^q \sin[n(z_i - z_j)], & q \text{ odd} 
\end{cases}
\]  

(4.4)

where \( c_0 = c_{N/2} = 2 \) and \( c_n = 1 \), otherwise. For an odd number of collocation points, the same differentiation matrix can be used by setting \( c_n = 1 \) for all values of \( n \).

Inserting the collocation differentiation equations (4.2) and (4.3) into the flux form of action equation (4.1), we obtain

\[
\frac{\partial N}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} + L(\theta_k, \zeta_l, x, y, t) = 0,
\]

(4.5)

where \( L \) is

\[
L(\theta_k, \zeta_l, x, y, t) = \frac{\partial G}{\partial \theta} \bigg|_{\theta=\theta_k} + J_{\zeta} \frac{\partial H}{\partial \zeta} \bigg|_{\zeta=\zeta_l} - S(\sigma_k, \theta_l, x, y, t).
\]

(4.6)

The discontinuous Galerkin statement now reads: Find an approximate solution \( \tilde{N}_h \) in the piecewise polynomial space \( V_h(\Omega) \), defined as \( \left( V_h(\Omega) \in P^k(\Omega) \right) \) where \( P^k(\Omega) \) denotes the collection of polynomials of degree up to \( k \) in the element \( \Omega \) such that
\[
\int_{\Omega} \frac{\partial N_h}{\partial t} v_h d\Omega + \int_{\Omega} \frac{\partial E_h}{\partial x} v_h d\Omega + \int_{\Omega} \frac{\partial F_h}{\partial y} v_h d\Omega + \int_{\Omega} L(\theta_k, \zeta_l, x, y, t) v_h d\Omega = 0, \quad \forall v_h \in V_h, \tag{4.7}
\]

where \( \Omega \) is an arbitrary triangle element, and \( L \) is sum of spectral space derivatives and physical source terms which is defined above. Applying integration by parts, we obtain

\[
\int_{\Omega} \frac{\partial N_h}{\partial t} v_h d\Omega - \int_{\Omega} E_h \frac{\partial v_h}{\partial x} d\Omega - \int_{\Omega} F_h \frac{\partial v_h}{\partial y} d\Omega + \int_{\partial \Omega} E_h v_h \cdot n dS + \int_{\partial \Omega} F_h v_h \cdot n dS + \int_{\Omega} L(\theta_k, \zeta_l, x, y, t) v_h d\Omega = 0. \tag{4.8}
\]

We next present details of the steps. The transformation from \((x, y)\) can be expressed in terms of the standard coordinate \((\xi_1, \xi_2)\):

\[
\begin{pmatrix}
    x \\
    y
\end{pmatrix}
= -\frac{\xi_1 + \xi_2}{2} \begin{pmatrix}
    x_1 \\
    y_1
\end{pmatrix}
+ \frac{1 + \xi_1}{2} \begin{pmatrix}
    x_2 \\
    y_2
\end{pmatrix}
+ \frac{1 + \xi_2}{2} \begin{pmatrix}
    x_3 \\
    y_3
\end{pmatrix}, \tag{4.9}
\]

where the subscripts \(1, 2, 3\) denote triangle vertices. The Jacobian of transformation is
\[
[J_{x\xi}] = \begin{bmatrix}
\frac{\partial x}{\partial \xi_1} & \frac{\partial y}{\partial \xi_1} \\
\frac{\partial x}{\partial \xi_2} & \frac{\partial y}{\partial \xi_2}
\end{bmatrix}.
\] (4.10)

The spatial derivatives can be computed in terms of transformed ones such that
\[
\begin{bmatrix}
\frac{\partial (\bullet)}{\partial x} \\
\frac{\partial (\bullet)}{\partial y}
\end{bmatrix} = \frac{1}{|J_{x\xi}|} \begin{bmatrix}
\frac{\partial y}{\partial \xi_2} & -\frac{\partial y}{\partial \xi_1} \\
-\frac{\partial x}{\partial \xi_2} & \frac{\partial x}{\partial \xi_1}
\end{bmatrix} \begin{bmatrix}
\frac{\partial (\bullet)}{\partial \xi_1} \\
\frac{\partial (\bullet)}{\partial \xi_2}
\end{bmatrix}.
\] (4.11)

The discontinuous Galerkin formulation in the standard region \(\Omega_S\) now reads
\[
\int_{\Omega_S} \frac{\partial N_h v_h}{\partial t} |J_{x\xi}| d\Omega_S - \int_{\Omega_S} E_h \left[ \frac{\partial y}{\partial \xi_2} \frac{\partial v_h}{\partial \xi_1} - \frac{\partial y}{\partial \xi_1} \frac{\partial v_h}{\partial \xi_2} \right] d\Omega_S - \\
\int_{\Omega_S} F_h \left[ -\frac{\partial x}{\partial \xi_2} \frac{\partial v_h}{\partial \xi_1} + \frac{\partial x}{\partial \xi_1} \frac{\partial v_h}{\partial \xi_2} \right] d\Omega_S + \int_{\partial\Omega_S} E_h v_h \cdot ndS + \\
\int_{\partial\Omega_S} F_h v_h \cdot ndS + \int_{\Omega_S} \mathcal{L}(\theta_k, \theta_l, x, y, t) v_h |J_{x\xi}| d\Omega_S = 0,
\] (4.12)

where the standard region area is equal to the product of differential sides of the element as \(d\Omega_S = d\xi_1 d\xi_2\). The construction of two-dimensional expansion functions are obtained readily by the tensorial product of two one-dimensional expansion functions in a standard structured domain \((Q^2 = \{-1 \leq \xi_1, \xi_2 \leq 1\})\). Two-dimensional tensorial constructed expansions allow dimensional factorization to compute integration of the terms efficiently, which was first recognized by [114]. A new coordinate system will be necessary for a triangle where the local coordinates are not orthogonal. Introducing collapsed coordinate transformation allows us to construct two-dimensional
expansion functions for a triangle region the same as in a structured region. The collapsed transformation is defined as [91]

\[ \eta_1 = \frac{2(1 + \xi_1)}{1 - \xi_2} - 1; \quad \eta_2 = \xi_2, \quad (4.13) \]

where \( \eta_1 \) and \( \eta_2 \) are new coordinates \(-1 \leq \eta_1, \eta_2 \leq 1\), see Figure 4.1, removing coordinate dependency of bounds in a triangle domain \((-1 \leq \xi_1 + \xi_2 \leq 1)\).

The Jacobian of transformation \( J_{\xi\eta} \) and its partial derivatives on a standard triangle coordinate \((\xi_1, \xi_2)\) can now be written in terms of a new coordinate system \((\eta_1, \eta_2)\)

\[ J_{\xi\eta} = \frac{1 - \eta_2}{2}, \quad (4.14) \]

\[ \begin{pmatrix} \frac{\partial}{\partial \xi_1} \\ \frac{\partial}{\partial \xi_2} \end{pmatrix} = \frac{1}{J_{\xi\eta}} \begin{pmatrix} \frac{\partial}{\partial \eta_1} \\ \frac{\partial}{\partial \eta_2} \end{pmatrix} \begin{pmatrix} 1 + \frac{\eta_1}{2} \frac{\partial}{\partial \eta_1} + \frac{1 - \eta_2}{2} \frac{\partial}{\partial \eta_2} \end{pmatrix}. \quad (4.15) \]

The discontinuous Galerkin formulation in equation (4.12) becomes in the new coordinate systems:
\[
\int_{\Omega_C} \frac{\partial N_h}{\partial t} v_h J_{\xi_2} J_{\xi_1} d\eta_1 d\eta_2 - \int_{\Omega_C} E_h \left[ \left( \frac{\partial y}{\partial \xi_2} - \frac{\partial y}{\partial \xi_1} \frac{1 + \eta_1}{2} \right) \frac{\partial v_h}{\partial \eta_1} - \frac{\partial y}{\partial \xi_1} \frac{1 - \eta_2}{2} \frac{\partial v_h}{\partial \eta_2} \right] d\eta_1 d\eta_2 - \\
\int_{\Omega_C} F_h \left[ \left( - \frac{\partial x}{\partial \xi_2} + \frac{\partial x}{\partial \xi_1} \frac{1 + \eta_1}{2} \right) \frac{\partial v_h}{\partial \eta_1} + \frac{\partial x}{\partial \xi_1} \frac{1 - \eta_2}{2} \frac{\partial v_h}{\partial \eta_2} \right] d\eta_1 d\eta_2 + \int_{\partial \Omega_C} E_h v_h \cdot n dS + \\
\int_{\partial \Omega_C} F_h v_h \cdot n dS + \int_{\Omega_C} L(\theta_k, \theta_l, x, y, t) v_h J_{\xi_2} J_{\xi_1} d\eta_1 d\eta_2 = 0, \quad (4.16)
\]

where a standard triangle region is defined in terms of new coordinates \((\eta_1, \eta_2)\) as

\[T^2\{(\eta_1, \eta_2) | -1 \leq \eta_1, \eta_2 \leq 1\}.\]

We seek approximate solution \(N_h\) expanded by orthogonal expansional functions and choose test function \(v_h\) as the same orthogonal expansion functions (Galerkin projection), respectively,

\[
N_h = \sum_{p=0}^{P_1} \sum_{q=0}^{P_2} u_{pq}(t) \psi_p(\eta_1) \psi_{pq}(\eta_2) \quad \text{and} \quad (4.17a) \\
v_h = \psi_m(\eta_1) \psi_{mn}(\eta_2). \quad (4.17b)
\]

The complete polynomial space for a triangular region will be \(\{(pq) \mid 0 \leq p, q; p \leq P_1; p + q \leq P_2; P_1 \leq P_2\}\). The orthogonal expansion functions in terms of Jacobi polynomials are [91]:
\[
\psi_p(\eta_1) = P_{p}^{0,0}(\eta_1); \quad \psi_m(\eta_1) = P_{m}^{0,0}(\eta_1)
\]
\[
\psi_{pq}(\eta_2) = \left(1 - \frac{\eta_2}{2}\right)P_{q}^{2p+1,0}(\eta_2); \quad \psi_{mn}(\eta_2) = \left(1 - \frac{\eta_2}{2}\right)^m P_{n}^{2m+1,0}(\eta_2)
\]

(4.18)

Inserting the approximate solution \( N_h \) \((N_h \in V_h)\) and test function \( v_h \) in equations (4.17) using Jacobi polynomials \( \psi_p \) and \( \psi_{pq} \) defined in equation (4.18) into equation (4.16), we obtain

\[
J_{x\xi} \sum_{p=0}^{P_1} \sum_{q=0}^{P_2} \frac{d u_{pq}(t)}{dt} \int_{-1}^{1} \int_{-1}^{1} \psi_p(\eta_1)\psi_m(\eta_1)\psi_{pq}(\eta_2)\psi_{mn}(\eta_2) J_{\xi\eta} d\eta_1 d\eta_2 - \int_{-1}^{1} \int_{-1}^{1} E(\theta_k, \sigma_l; \eta_1, \eta_2, t) \left( \frac{\partial y}{\partial \xi_2} - \frac{\partial y}{\partial \xi_1} \frac{1 + \eta_1}{2} \right) \psi_{mn}(\eta_2) \frac{d \psi_m(\eta_1)}{d \eta_1} \right) d\eta_1 d\eta_2 - \int_{-1}^{1} \int_{-1}^{1} F(\theta_k, \sigma_l; \eta_1, \eta_2, t) \right) \left\{ \left( - \frac{\partial x}{\partial \xi_2} + \frac{\partial x}{\partial \xi_1} \frac{1 + \eta_1}{2} \right) \psi_{mn}(\eta_2) \frac{d \psi_m(\eta_1)}{d \eta_1} + \frac{\partial x}{\partial \xi_1} \frac{1 - \eta_2}{2} \psi_m(\eta_1) \frac{d \psi_{mn}(\eta_2)}{d \eta_2} \right\} d\eta_1 d\eta_2 + \int_{\partial \Omega} E_h v_h \cdot n dS + \int_{\partial \Omega} F_h v_h \cdot n dS + J_{x\xi} \int_{-1}^{1} \int_{-1}^{1} L(\theta_k, \theta_l; \eta_1, \eta_2, t) \psi_m(\eta_1)\psi_{mn}(\eta_2) J_{\xi\eta} d\eta_1 d\eta_2 = 0, \quad (4.19)
\]

where \( u_{pq} \) are the basis coefficients to be determined. The boundary flux term in equation (4.8) can be replaced by various numerical fluxes [154]. The upwind flux is chosen in here to replace the term.

The discretized equations are now reduced to an Ordinary Differential Equation (ODE) for the expansion coefficients \( u_{pq} \).
\[
\frac{du_{pq}(t)}{dt} = \frac{1}{J_{x\xi}} M^{-1}_{pq,pp} \left( S_{pq} + T_{pq} - E_{pq} - F_{pq} - L_{pq} \right),
\]
(4.20)

where the mass matrix \( M \), the stiffness vectors \( S \) and \( T \), the edge fluxes \( E \) and \( F \), and numerical source term \( L \) are, respectively, the result of integrated terms in equation (4.19).

The mass matrix \( M \) is the result of the following quadrature integration from Eq. (4.19) by using the orthogonality of Jacobi polynomials \( \int P_m(x) P_n(x) dx = 0, \ m \neq n \):

\[
M_{mn,pq} = \int_{-1}^{1} \int_{-1}^{1} P^0_p P^0_m \left( \frac{1 - \eta_2}{2} \right)^p P^{2p+1,0}_q \left( \frac{1 - \eta_2}{2} \right)^m P^{2m+1,0}_n \frac{1 - \eta_2}{2} d\eta_1 d\eta_2
\]
\[
M_{pm,pq} = \frac{2}{1 + 2p} \int_{-1}^{1} \left( \frac{1 - \eta_2}{2} \right)^{2p+1} P^{2p+1,0}_q \frac{1 - \eta_2}{2} P^{2m+1,0}_n d\eta_2
\]
\[
M_{pq,pq} = \frac{2}{(1 + 2p)(1 + p + q)}.
\]
(4.21)

The derivative of expansion functions Eqs. (4.18) in respect to \( \eta_1 \) and \( \eta_2 \) directions are

\[
\frac{d\psi_m(\eta_1)}{d\eta_1} = \frac{dP^0_m(\eta_1)}{d\eta_1},
\]
\[
\frac{d\psi_{mn}(\eta_2)}{d\eta_2} = -\frac{m}{2} \left( \frac{1 - \eta_2}{2} \right)^{m-1} P^{2m+1,0}_n(\eta_2) + \left( \frac{1 - \eta_2}{2} \right)^m \frac{dP^{2m+1,0}_n(\eta_2)}{d\eta_2}.
\]
(4.22)
The stiffness vectors $S$ and $T$ are the result of integration-by-parts, which can be computed by using the relation (4.22) as follows:

$$S_{pq} = \sum_{i=0}^{Q_1-1} w_i \left( \frac{\partial y}{\partial \xi_2} - \frac{\partial y}{\partial \xi_1} \right) \frac{d\psi_p(\eta_{1i})}{d\eta_1} \sum_{j=0}^{Q_2-1} w_j E(\theta_k, \sigma_l, \eta_{1i}, \eta_{2j}, t) \psi_{pq}(\eta_{2j})$$

$$- \sum_{i=0}^{Q_1-1} w_i \frac{\partial y}{\partial \xi_1} \psi_p(\eta_{1i}) \sum_{j=0}^{Q_2-1} w_j E(\theta_k, \sigma_l, \eta_{1i}, \eta_{2j}, t) \frac{1 - \eta_{2j}}{2} \frac{d\psi_{pq}(\eta_{2j})}{d\eta_2}, \quad (4.23a)$$

$$T_{pq} = \sum_{i=0}^{Q_1-1} w_i \left( - \frac{\partial x}{\partial \xi_2} + \frac{\partial x}{\partial \xi_1} \right) \frac{d\psi_p(\eta_{1i})}{d\eta_1} \sum_{j=0}^{Q_2-1} w_j F(\theta_k, \sigma_l, \eta_{1i}, \eta_{2j}, t) \psi_{pq}(\eta_{2j})$$

$$+ \sum_{i=0}^{Q_1-1} w_i \frac{\partial x}{\partial \xi_1} \psi_p(\eta_{1i}) \sum_{j=0}^{Q_2-1} w_j F(\theta_k, \sigma_l, \eta_{1i}, \eta_{2j}, t) \frac{1 - \eta_{2j}}{2} \frac{d\psi_{pq}(\eta_{2j})}{d\eta_2}, \quad (4.23b)$$

where $\eta_{1i}, \eta_{2j}$ are the quadrature points in the $\eta_1$ and $\eta_2$ directions and, whose corresponding weights are, respectively, $w_i$ and $w_j$. We have still not mentioned which quadrature rule will be used in the above integrations. We have only one restriction which is not to choose quadratures as Gauss-Lobatto points in the one end of $\eta_2$ direction due to singularity; otherwise, we are free to choose any quadrature points. The integration terms in Eqs. (4.23) containing polynomial $P_{2p+1}$ are numerically exact if the quadrature points $Q$ will be chosen $Q = p + 1$ for Gauss-Lobatto, $Q = p + 1$ (total quadrature points $Q$ must be integer and $Q = p + 1$ is exact for $P_{2p+2}$) for Gauss-Radau, and $Q = p$ for Gauss-Jacobi quadratures. The Gauss-Lobatto quadrature includes both end points $[-1, 1]$ and the Gauss-Radau excludes one end point at $+1 [-1, 1)$. The Gauss-Jacobi quadratures are inside the interior domain $(-1, 1)$. 
It is obvious that choosing Gauss-Jacobi quadrature points is better if accuracy is the only consideration; namely, for a given polynomial order $p$, Gauss-Jacobi would use the least number of total quadrature points $Q$ among the others. However, for ease in applying boundary conditions, it is advantageous to employ Gauss-Lobatto in $\eta_1$ and Gauss-Radau quadratures (excluding a singular point introduced by the collapsed coordinate transformation) in $\eta_2$ directions at the expense of a slightly increased computation cost. Gauss-Lobatto and Gauss-Radau quadratures are used for numerical integration of the terms in this study.

The integration by parts has yielded two boundary terms called edge flux terms in Eq. (4.19) which need to be integrated over the edge of the elements. The numerical solutions are allowed to be discontinued across elements in the domain so that multiple physical solutions occur at the element boundaries. This will be resolved by replacing fluxes $E_h$ and $F_h$ with numerical fluxes such as $H_h$ and $G_h$, respectively. The fluxes in a normal direction can be replaced as

\[
E_h \cdot n_e = H_e(N_{h}^{int}, N_{h}^{ext}) \quad \text{and} \quad \quad (4.24a)
\]
\[
F_h \cdot n_f = G_e(N_{h}^{int}, N_{h}^{ext}), \quad \quad (4.24b)
\]

where interior "$int$" and exterior "$ext$" will be interpreted differently according to a specific edge of element and subscript "$e$" denotes an element. We have chosen upwind fluxes with the justification that it is the simplest to implement and has no need of any special treatment in the code for absorbing boundary condition treatment in coastal boundaries. The more advanced flux treatments will be considered for a future study. The computation of edge integrals is approximated as
\[ \int_{e} H_e v_h dS \approx E_{pq} = \frac{1}{2} \sum_{i=0}^{Q-1} w_i H_e(x_{ei}, y_{ei}, t) v_h(x_{ei}, y_{ei}) |d\epsilon|, \tag{4.25a} \]

\[ \int_{e} G_e v_h dS \approx F_{pq} = \frac{1}{2} \sum_{i=0}^{Q-1} w_i G_e(x_{ei}, y_{ei}, t), v_h(x_{ei}, y_{ei}) |d\epsilon|, \tag{4.25b} \]

where the letter "\( e \)" denotes an arbitrary element. Remember that the trial function was defined as \( v_h = \psi_p(\eta_1) \psi_{pq}(\eta_2) \).

The numerical source term integration in Eq. (4.20) is

\[ L_{pq} = J_{\xi} \sum_{i=0}^{Q_1-1} \sum_{j=0}^{Q_2-1} w_i \psi_p(\eta_{1i}) w_j \mathcal{L}(\theta_k, \theta_l, \eta_{1i}, \eta_{2j}, t) \psi_{pq}(\eta_{2j}) \frac{1 - \eta_{2j}}{2}. \tag{4.26} \]

### 4.1.1 Time Discretization

We employ explicit Strong Stability Runge-Kutta (SSP-RK) time discretization (also called TVD-RK methods), which is inherently stable, to discretize equation (4.20) in time. Total Variation Diminishing (TVD) schemes prevent nonlinear instability, which occurs in hyperbolic problems containing shocks or any discontinuities. Standard stable RK methods (non-TVD) of orders higher than one is shown oscillatory solutions even for spatial discretizations having the TVD property [65]. Implicit high-order RK methods (> 1) do not satisfy TVD conditions [64]; however, they might not show oscillatory in many practical applications of interest to us. We will narrow our scope to explicit schemes in this paper and leave implicit schemes for future study. This study employed the second (2-stages) and third order (3-stages) SSP-RK methods proposed in [65], and the fourth order SSP-RK (5-stages) given
Gottlieb and Shu [65] showed that it is not possible to construct fourth order TVD scheme in 4-stages. Note that the fourth-order SSP-RK schemes require larger storage and higher computation due to the scheme becoming 5-stages but this is partially compensated for allowing a time step 50% larger than in the first-order Euler method.

Eq. (4.19) can be rewritten as

$$\frac{du(t)}{dt} = r(u, t), \quad (4.27)$$

where the unknown $u$ is alias to $u_{pq}$ to shorten the notation.

The 2nd and 3rd order SSP-RK schemes are, respectively, [65] [79]

$$v^{(1)} = u^n + \Delta t \ r(u^n, t^n)$$
$$u^{n+1} = \frac{1}{2} \left( u^n + v^{(1)} + \Delta t \ r(v^{(1)}, t^n + \Delta t) \right), \quad (4.28)$$

and

$$v^{(1)} = u^n + \Delta t \ r(u^n, t^n)$$
$$v^{(2)} = \frac{1}{4} \left( 3u^n + v^{(1)} + \Delta t \ r(v^{(1)}, t^n + \Delta t) \right)$$
$$u^{n+1} = \frac{1}{3} \left( u^n + 2v^{(2)} + 2\Delta t \ r(v^{(2)}, t^n + \frac{1}{2} \Delta t) \right). \quad (4.29)$$
The 4th order RK (5-stage) scheme is \[147][79]\]

\[
v^{(1)} = u^n + 0.39175222700392\Delta t \, r(u^n, t^n)
\]
\[
v^{(2)} = 0.44437049406734u^n + 0.55562950593266v^{(1)} + 0.36841059262959\Delta t \, r(v^{(1)}, t^n + 0.39175222700392\Delta t)
\]
\[
v^{(3)} = 0.62010185138540u^n + 0.37989814861460v^{(2)} + 0.25189177424738\Delta t \, r(v^{(2)}, t^n + 0.58607968896780\Delta t)
\]
\[
v^{(4)} = 0.17807995410773u^n + 0.82192004589227v^{(3)} + 0.5449745021237\Delta t \, r(v^{(3)}, t^n + 0.47454236302687\Delta t)
\]
\[
u^{n+1} = 0.0068325884039u^n + 0.51723167208978v^{(2)} + 0.12759831133288v^{(3)} + 0.348333675773694v^{(4)} + 0.08460416338212\Delta t \, r(v^{(3)}, t^n + 0.47454236302687\Delta t) + 0.22600748319395\Delta t \, r(v^{(4)}, t^n + 0.93501063100924\Delta t).
\]

\[(4.30)\]

### 4.1.2 Forward and Backward Transformations

The initial condition for equation in Eq. (4.20) (only available in physical space) needs to be transformed from physical to modal space, using a forward transformation. Similarly, the modal coefficients \(u_{pq}(t)\) need to be transformed back to the physical variable of interest at various computation stages, using a backward transformation.

The approximate solution given in equation (4.17a) is expanded in terms of modal coefficients. We define the Vandermonde transformation matrix as
where $Q_1$ and $Q_2$ are, respectively, the total quadrature points in the direction of $\eta_1$ and $\eta_2$. In the triangle region, the complete space can be defined by only $N_p = \frac{(P_1+1)(P_2+2)}{2}$ polynomials rather than $N_P = (P_1 + 1)(P_2 + 2)$ as in the quadrilateral region, and the total number of quadrature points is a product of $Q_1$ and $Q_2$ as $N_Q = Q_1 \ast Q_2$. As mentioned earlier, $Q_1$ and $Q_2$ should be chosen $(P_1+1)$ and $(P_2+1)$ for exact integration of the terms, respectively. The above Vandermonde matrix has row and column dimensions of $N_Q$ and $N_P$ and has a rectangular shape (not square matrix, $N_Q \neq N_P$) for the triangular region. The computation of the Vandermonde matrix is convenient to compute once and store it at the initial provided that the quadrature points have been fixed (p-adaptivity certainly will force us to adjust the quadrature points along the computation) for the entire simulation.

The backward transformation can now be represented as matrix-vector multiplication:

\[
V = \begin{bmatrix}
\psi_0(\eta_0^0)\psi_00(\eta_2^0) & \psi_0(\eta_1^0)\psi_01(\eta_2^0) & \cdots & \psi_1(\eta_1^0)\psi_10(\eta_2^0) & \cdots & \psi_p(\eta_1^0)\psi_pq(\eta_2^0) \\
\psi_0(\eta_0^1)\psi_00(\eta_2^1) & \psi_0(\eta_1^1)\psi_01(\eta_2^1) & \cdots & \psi_1(\eta_1^1)\psi_10(\eta_2^1) & \cdots & \psi_p(\eta_1^1)\psi_pq(\eta_2^1) \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\psi_0(\eta_0^Q_1)\psi_00(\eta_2^Q_1) & \psi_0(\eta_1^Q_1)\psi_01(\eta_2^Q_1) & \cdots & \psi_1(\eta_1^Q_1)\psi_10(\eta_2^Q_1) & \cdots & \psi_p(\eta_1^Q_1)\psi_pq(\eta_2^Q_1) \\
\end{bmatrix}_{[(i\eta_0),(pq)]}
\]

\[
N_h = V \ u_{pq}. \tag{4.32}
\]
The forward transformation (evaluating $u_{pq}$ from $N_h$) is not obvious from equation (4.32) because the Vandermonde matrix is not a square matrix and hence is not invertible in a traditional way. Using Singular Value Decomposition (SVD), the Vandermonde matrix in equation (4.32) is factorized as

$$ V = U S V^T, \quad (4.33) $$

where $U$ is an $N_Q \times N_P$ unitary matrix, $S$ is a min$(N_Q, N_P)$ diagonal matrix, and $V^T$ is a transpose of $N_P \times N_P$ unitary matrix. We invert the above equation by using a property of unitary matrices ($A^{-1} = A^T$), and get

$$ V^{-1} = VS^{-1}U^T. \quad (4.34) $$

Inversion of the diagonal matrix $S$ and transpose of unitary matrix $U$ is a trivial task after computation of SVD; In this study, we have used the linear algebra library LAPACK [8]. The forward transformation can be now computed by using the inverse Vandermonde matrix as follows:

$$ u_{pq} = V^{-1}N_h. \quad (4.35) $$

### 4.1.3 Mapping in the Spectral Direction

The directional spectrum is not known a priori for many cases and it is usually proposed in a specific form. The form suggested in [82] is
where the amplitude $A_1$ is defined in terms of Gamma functions $A_1 = \Gamma\left(\frac{1}{2}m + 1\right)/\left[\Gamma\left(\frac{1}{2}(m + 1)\right)\sqrt{\pi}\right]$ and the parameter ‘$m$’ is the width of the directional spectrum; the larger (smaller) values of the width parameter $m$ generate a narrower (wider) width. The parameter ‘$m$’ might vary from 2 to 800 in practical applications. In this work, we use the narrow directional spectrum $\cos^{500}(\theta)$ to describe incoming waves for test cases in the verification and validation chapter 5.

The spectral width can be defined by an analogy to the standard deviation definition

$$
\sigma^2_\theta = \int_{-\pi}^{\pi} \theta^2 D(\theta) d\theta,
$$

and is used as a guide to determine the value of the directional width. For example, if the width parameter $m = 500$ then the spectral width $\sigma_\theta = 2.5^\circ$. For this specific example, the region $[-7.5^\circ, 7.5^\circ]$ would cover 99% of the total energy in Figure 4.2a. It is obvious that using the Fourier-collocation points on regions $[-180^\circ, -7.5^\circ)$ and $(7.5^\circ, 180^\circ]$ would not improve the solution significantly. However, we have no choice if we are using standard Fourier expansions, which require evenly distributed collocations in the domain $[-\pi, \pi]$. The mapping technique is an implicit way of modifying the expansion functions and allowing a collocation grid clustered in a specific region.

The atan mapping [28, 30] can be defined in $[-\pi, \pi]$ for the spectral direction

$$
\theta = g(\chi, L) = 2 \arctan[L \tan(0.5\chi)] + \chi_0,
$$

(4.38)
where \( \chi \) denotes computational space which permits the standard Fourier-collocation method for differentiation, and the constant parameter \( L \) (in general \( L(t) \)) controls the amount of clustering around the peak center \( \chi_0(\chi_0(t)) \). Using a change of derivative coordinate transformation for the mapping relation in equation (4.38), we obtain the transformation between the physical \( \theta \) and computational coordinates \( \chi \) as follows:

\[
g' = \frac{d\theta}{d\chi} = \frac{L[1 + \tan^2(\frac{\chi}{2})]}{1 + L^2\tan^2(\frac{\chi}{2})}, \tag{4.39}\]

and the directional differentiation of flux is

\[
\frac{\partial G}{\partial \theta} = \frac{d\chi}{d\theta} \frac{\partial G}{\partial \chi} = \frac{1}{g'} \frac{\partial G}{\partial \chi}, \tag{4.40}
\]

where the differentiation matrix in equation (4.4) can be used to differentiate \( G \) with respect to \( \chi \). If we define the diagonal matrix \( \mathcal{M} \) as

\[
\mathcal{M}_{jj} = \frac{1}{g'} \tag{4.41}
\]

then, the flux derivative \( \frac{\partial G}{\partial \theta} \) in which the collocation grids are clustered is now computed by a new differentiation matrix, \( \mathcal{D} \):
The clustering collocation grids in $\theta$, and the transformation metrics, $g'$, may be computed by using a user-specified clustering parameter $L$ and standard Fourier-collocation grids (equi-spaced distribution, $-\pi + \frac{2\pi}{N} k$, $k = [0, \cdots, N - 1]$). The mapped differentiation matrix is evaluated once and it is stored for all computations in fixed collocation grids (non-adaptive cases).

We specified the directional spreading of incoming waves as $\cos^{500}(\theta)$ (see Figure 4.2a) for some verification and validation cases later in this work. In this section, we will present derivative results of this specific directional spreading form. The standard Fourier expansion wastes collocation points in a very smooth region of this example, such that very smooth region will require about 96% of all the collocation grids.

In Figure 4.2, we compare equi-spaced standard Fourier differentiation with the modified Fourier differentiation with mapping. It is worth mentioning that the standard method also shows Gibbs-like oscillations if the high-gradient region is poorly resolved for this specific example. We have tried uniform collocation grids with $N = 64, 128, 256, 512$ and examine the pointwise errors ($u - u_N = \frac{dD(\theta)}{d\theta} - \frac{dD(\theta)N}{d\theta}$, see Figure 4.2b). This clearly shows that once the solution is resolved, the Fourier method achieves excellent spectral convergence behavior. Figures 4.2c and 4.2d present pointwise errors for two different constant parameters, $L$ as $L = 2\sigma_\theta$ and $L = 4\sigma_\theta$. We see that the pointwise errors are decreasing exponentially for both cases. These comparisons reveal that precise value of mapping parameter $L$ does not seem to be playing an important role (see $N = 16$ resolutions from Figures 4.2c and
Figure 4.2: (a) The derivative of a $\cos^{500}(\theta)$ directional spreading ($A_1 = 1$) where the clustered region is confined to the interval $[-7.5^\circ, 7.5^\circ]$ which only makes up 4% of whole domain $2\pi$. (b) Pointwise errors for standard Fourier-collocation, (c) Mapped Fourier-collocation with ($L = 2\sigma_\theta$), and (d) with ($L = 4\sigma_\theta$) for various resolutions. The spectral width for $\cos^{500}(\theta)$ is $2.5^\circ$. 
4.2d) for the coarse collocation grids but this might become important as the grid resolution increases. The choice of the parameter $L$ should still be based on the spectral width $\sigma_\theta$; we suggest that at least $L = 3\sigma_\theta$. The differentiation with mappings have achieved *comparable error levels* of the standard method with $N = 128$ (see Figures 4.2b, 4.2c, and 4.2d) in the region where a high-gradient solution occurs and *much lesser errors* away from this region, with *eight-fold* reduction in resolution.

### 4.1.4 Absorbing Boundary Treatment in Frequency Direction

Our present problem has a semi-infinite ($[0, \infty]$) domain, and solutions decay exponentially fast as $\sigma \to \infty$ and $\sigma \to 0$.

The initial conditions at the truncated boundaries $(N(\theta, \sigma_{\min}, X)$ and $N(\theta, \sigma_{\max}, X)$) for real applications after the truncation is made are very small ($N(\sigma_{\min}) \neq N(\sigma_{\max}) \ll 1$). The periodicity of a function is in the literal sense violated, and ideally we should use for this case Chebyshev method instead of Fourier-collocation. However, Boyd’s work [29] has shown that Fourier-collocation is more efficient than the Chebyshev method even in truncated domains in the case of very small domain truncation errors if there is exponential decay of the solution. Ocean wave solutions truly satisfy both conditions such that there are very small truncation errors and exponentially decaying solutions. This is the main reason to still use Fourier-collocation rather than Chebyshev methods. The true boundary conditions for a truncated domain would be non-reflecting boundary conditions permitting incoming waves without any reflection back to the computational domain. We have used the Absorbing Boundary Layer(ABL) approach [87] to satisfy non-reflective boundary
conditions in the frequency direction. This approach will enforce periodicity for even a non-periodic developing solution at the truncated domain boundaries. The combination of Fourier-collocation and ABL would be our preferred way of imposing non-reflective boundary condition in a frequency direction.

In order to impose the ABL approach, we add the term $g(\sigma) N(\theta; x, y, t)$ to the original equation (3.6), and hence the new modified action balance equation is

$$\frac{\partial N(\theta, \sigma; x, y, t)}{\partial t} + \frac{\partial c_{g,x} N(\theta, \sigma; x, y, t)}{\partial x} + \frac{\partial c_{g,y} N(\theta, \sigma; x, y, t)}{\partial y} + \frac{\partial c_{g,\theta} N(\theta, \sigma; x, y, t)}{\partial \theta} + \frac{\partial c_{g,\sigma} N(\theta, \sigma; x, y, t)}{\partial \sigma} + g(\sigma) * N(\theta, \sigma; x, y, t) = \frac{S(\sigma, \theta; x, y, t)}{\sigma} \arg \text{modification term} \quad (4.43)$$

where $g(\sigma)$ is the absorbing function, getting zero values inside the domain and non-zero values in the absorbing layers, hence damping the numerical solution in these regions. The absorbing function can be defined in the computational domain $\zeta$ in $[-\pi, \pi]$ as follows:

$$g(\zeta) = \gamma_L \left(1 - \tanh[\alpha_L(\zeta + \pi)]\right) + \gamma_R \left(1 + \tanh[\alpha_R(\zeta - \pi)]\right), \quad (4.44)$$

where $\gamma$ controls the magnitude of the absorbing functions, $\alpha$ determines the effective width of the absorbing layers, and $L$ and $R$ denote, respectively, the left and right layers. The term $\mathcal{L}$ in equation (4.6) is now modified as
\[ \mathcal{L}(\theta_k, \zeta_l, x, y, t) = \frac{\partial G}{\partial \theta} \bigg|_{\theta=\theta_k} + J_{\zeta} \frac{\partial H}{\partial \zeta} \bigg|_{\zeta=\zeta_l} + g(\sigma_l)N(\theta_k, \sigma_l, x, y, t) - S(\theta_l, \sigma_k, x, y, t). \] (4.45)

(a) Initial JONSWAP spectrum with \( \gamma = 1.0, f_{\text{peak}} = 0.5 \text{Hz}, \sigma_a = 0.07 \) and \( \sigma_b = 0.09 \).

(b) Absorbing Boundary Layer Function

(c) \( t = 1.25 \)

(d) \( t = 1.5 \)

(e) \( t = 1.625 \)

(f) \( t = 1.75 \)

(g) \( t = 1.8 \)

(h) \( t = 1.9 \)

Figure 4.3: The evolution of the JONSWAP spectrum for non-reflecting boundary conditions. The analytical solution (—) and numerical solution (---) are compared at \( t = 0, 1.5, 1.625, 1.75, 1.8, 1.9 \).

The parameter couples \( \alpha_L \) and \( \alpha_R \) control the ABL widths. The computation of controlling parameters is possible by introducing arbitrary cut-off values \( \epsilon \). For certain cut-off values, we can relate the \( \alpha_L \) values to the absorbing boundary layer
width $\Delta \zeta_L$ as
\[
\alpha_L = \frac{\text{atanh}(1 - \epsilon/\gamma_L)}{\Delta \zeta_L}.
\] (4.46)

In a similar way for $\alpha_R$
\[
\alpha_R = \frac{\text{atanh}(1 - \epsilon/\gamma_R)}{\Delta \zeta_R},
\] (4.47)

where $\Delta \zeta_L$ and $\Delta \zeta_R$ are user-defined and represent the left and right-absorbing boundary layer width.

Next we will demonstrate propagation of the JONSWAP spectrum (enforced as initial condition) [72] at a truncated domain. The boundary conditions will be simulated by the ABL method. The propagation velocities $c_x, c_y, c_\theta$ are set to zero, and $c_\sigma$ is set to $2\pi$. The physical domain (now 1D frequency direction) extends from 0.2 Hz to 2.0 Hz. The left and right absorbing layers ($\Delta \sigma_L, \Delta \sigma_R$) are specified, respectively, as 0.1 Hz and 0.2 Hz. Adding the layers that extend the physical domain from [0.2, 2.0] Hz to an effective domain [0.1, 2.2] Hz increases the cost of the computation up to 15%. The numerical simulation is compared to analytical solutions at various time points ($t = 0, 1.5, 1.625, 1.75, 1.8, 1.9$). We are only interested in the solutions inside the physical domain, i.e. we are not concerned with the solutions in the ABL layers. Figure 4.3 confirms that no reflection is occurring at the boundaries. As soon as the numerical solution enters the absorbing layers, it is damped severely.
Chapter Five

Verification and Validation
We have proposed a new scheme for spectral ocean wave equation in the previous chapter. We here do verification and validation of the scheme in this chapter. We will first present time and spatial (both physical and spectral) accuracy verification of the scheme in section 5.1. After that, we will validate our implementation against analytical solutions, observation data, and experimental results in section 5.2.

5.1 Verification of Temporal and Spatial Accuracy

We present errors norms $L_2$ and $L_\infty$ used in the verification. The $L_2$ error norm is defined as:

$$||e||^2_{L_2} = \int_t^\infty \int_\sigma \int_\Omega (u - \hat{u})^2 \ d\Omega \ d\sigma \ d\theta \ dt,$$

(5.1)

where $u$ and $\hat{u}$ are, respectively, the exact and numerical solutions. The maximum norm $L_\infty$ is defined as

$$||e||_{L_\infty} = \int_t^\infty \max |u - \hat{u}| \ dt.$$

(5.2)

We first give temporal verification accuracy of the Runge-Kutta scheme in section 5.1.1. Next we present $p$ convergence results for physical and spectral discretization and $h$ convergences for the only physical space discretization in section 5.1.2.
5.1.1 Temporal Accuracy Verification

We simplify the equation by dropping the source term and setting the propagation velocities ($c_x, c_y, c_{\theta}, c_{\sigma}$) to one in Eq. (3.6). We propose the analytical function

$$N(\sigma, \theta; x, y, t) = \sin(x - t) \cos(y - t) \sin(\theta - t) \cos(\sigma - t), \quad (5.3)$$

which satisfies the action balance equation after the above simplifications. We have generated boundary and initial conditions from the manufactured solution Eq. (5.3). The geographical space discretizations using the mesh (see Figure 5.1a) employs $12^{th}$ order of Jacobi polynomials to eliminate the domination of spatial discretization errors with eight collocation grid points in both spectral directions ($\theta$ and $\sigma$). Our simulation runs up to time unit 1.0 for second, third, and fourth-order SSP-RK schemes. The numerical tests, see Table 5.1, confirm the order of accuracy expected by SSP-RK time discretizations.

<table>
<thead>
<tr>
<th>$\Delta t$</th>
<th>error in RK2</th>
<th>rate</th>
<th>error in RK3</th>
<th>rate</th>
<th>error in RK4</th>
<th>rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td>8.13787e-02</td>
<td>2.0</td>
<td>2.48741e-03</td>
<td>3.02</td>
<td>4.92365e-05</td>
<td>4.13</td>
</tr>
<tr>
<td>0.005</td>
<td>2.03336e-02</td>
<td>2.0</td>
<td>3.07691e-04</td>
<td>3.01</td>
<td>2.82155e-06</td>
<td>3.87</td>
</tr>
<tr>
<td>0.0025</td>
<td>5.08308e-03</td>
<td>-</td>
<td>3.83024e-05</td>
<td>-</td>
<td>1.92636e-07</td>
<td>-</td>
</tr>
</tbody>
</table>

5.1.2 Spatial Accuracy Verification

The spatial errors have contributions from both the Fourier-collocation and DG spectral element discretization. We will show that the proposed scheme has spectral accuracy in both spaces. We also want to investigate the effective convergence
order in spectral space. The manufactured solution proposed before in Eq. (5.3) will converge too rapidly to very low levels of errors as we increase the number of collocations in spectral space. Hence, we introduce another manufactured solution to better quantify the exponential convergence in spectral space, i.e.,

$$N(\theta, \sigma; x, y, t) = e^{\sin(x-t)} + e^{\cos(y-t)} + e^{\sin(\theta-t)} + e^{\cos(\sigma-t)}. \quad (5.4)$$

We run the code for a sequence of meshes presented in Figures 5.1b, 5.1c, and
5.1d for a varied order of Jacobi polynomials (p-convergence test) with the SSP-RK4 scheme in a simulation using time step size $\Delta t = 0.001$ up to final time $t = 1$. The fourth-order time scheme yields the $L_2$ error of time discretization about $1e-9$. Each curve in Figure 5.2 represents $L_2$ error for a specific resolution of Fourier-collocation and mesh with a varying order of Jacobi polynomials. We made the time discretization errors sufficiently low that we can expect exponential convergence in the geographical scheme by increasing the order of Jacobi polynomials, provided that we have sufficiently low Fourier-collocation discretization errors in spectral space. If we look at Figure 5.2a, we observe exponential convergence of the DG scheme for a specific spectral space discretizations ($N_\theta = 24, N_\sigma = 24$). All other figures have consistently confirmed the exponential accuracy of the DG related discretization scheme if the number of spectral space discretization errors is kept low. We see that the $L_2$ error curves eventually hit a plateau in Figure 5.2. This indicates another error barrier due to Fourier-collocation discretizations. The DG discretization errors are certainly still decreasing but are so small compared to the Fourier-collocation errors that they are no longer dominant in the cumulative $L_2$ error.

The exponential accuracy for Fourier-collocation (spectral space discretization) can be examined on these plateau regions. The errors of these plateau levels can be used to quantify the convergence behavior of spectral space discretizations (Fourier-collocation).

Another way to quantify convergence is to study h-convergence. The convergence rate of the DG scheme can be established by evaluating the rate of error with respect to the refined meshes (see Figures 5.1b, 5.1c, and 5.1d). We run simulations up to final time 1.0 with the time step $\Delta t = 0.001$ by using the SSP-RK4 time scheme and 24 Fourier-collocations in spectral space ($N_\theta = 24, N_\sigma = 24$). The convergence rates given in Table 5.2 confirm the accuracy ($(p + 1)$th order) of a DG scheme up
Frequency ($\sigma$) and Directional ($\sigma$) collocation resolutions 4, 8, 16, 24 are tested against polynomial order P. The errors are integrated over simulation time $t \in [0,1]$ with the SSP-RK4 scheme ($\Delta t = 0.001$).

polynomial order 9. It is also clear that we reach either time or Fourier-collocation discretization errors for the Jacobi polynomial of order 11 ($p=11$).
Table 5.2: $L_2$ and $L_\infty$ errors and convergence rates ($N_\theta = 24$, $N_\sigma = 24$, $\Delta t = 0.001$) between meshes ($h/2$) and ($h/4$). The errors are integrated over simulation time $t \in [0, 1]$ using the RK4 scheme. The rates * hit either/both time-related or/spectral space-related discretization.

<table>
<thead>
<tr>
<th>p</th>
<th>$L_\infty$</th>
<th>$L_2$</th>
<th>(\Delta t) = 0.001</th>
<th>$L_\infty$</th>
<th>$L_2$</th>
<th>(\Delta t) = 0.001</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.18308e+01</td>
<td>4.36525e+00</td>
<td>1.44</td>
<td>3.00660e+00</td>
<td>9.70100e-01</td>
<td>1.63</td>
</tr>
<tr>
<td>3</td>
<td>8.80192e-01</td>
<td>7.60808e-02</td>
<td>3.53</td>
<td>1.79565e-01</td>
<td>1.35042e-02</td>
<td>3.73</td>
</tr>
<tr>
<td>5</td>
<td>4.08564e-02</td>
<td>8.71108e-04</td>
<td>5.55</td>
<td>8.20653e-03</td>
<td>1.23181e-04</td>
<td>6.06</td>
</tr>
<tr>
<td>7</td>
<td>1.71219e-03</td>
<td>7.40512e-06</td>
<td>7.85</td>
<td>2.68193e-04</td>
<td>9.08277e-07</td>
<td>8.21</td>
</tr>
<tr>
<td>9</td>
<td>6.23888e-05</td>
<td>7.24080e-08</td>
<td>9.75</td>
<td>7.36509e-06</td>
<td>7.72374e-09</td>
<td>9.90</td>
</tr>
</tbody>
</table>

5.2 Verification and Validation Cases

We will present the test cases used in the verification and validation of other spectral ocean models. We mainly followed the test cases for SWAN verification and validation in Ris’s work [130] except for the duration-limited wave growth test case (taken from [88]). The first section 5.2.1 addresses the diffusion behavior of the current scheme. The next two sections 5.2.2 and 5.2.3 present the results of shoaling and refraction, respectively, from current and depth variations. Those test cases assumed that there is no source term in the equation. The test cases in sections 5.2.4 and 5.2.5 validate, respectively, the duration-limited wave growth and fetch-limited wave growth in the sea. These test cases activated relevant source terms in the deep water (generation by wind, white-capping, and quadruplets). The final three test cases in sections 5.2.6, 5.2.7, and 5.2.8 validate the scheme against the laboratory experiments. We also consider a wave blocking test case [130, 98]. This test case has a sharp gradient in the energy along the frequency direction. This generated Gibbs-like oscillations and caused our scheme to be unstable. The first remedy (the costly one) is to employ fine collocations around the sharp gradient solution. Without considering an adaptive scheme along the frequency direction, this solution increases the collocation points about several orders higher. Filtering is another remedy for
this issue. We presented a filter application to represent how the scheme is stabilized and recovers exponential convergence for discontinuous or sharp-gradient problems arising from nonlinear inviscid/viscous Burger and inviscid nonlinear Euler equations in appendix A. The filtering given in appendix A can also be applied to the other derivatives such as directional and physical derivatives to have a more robust scheme without losing exponential convergence. This will help the scheme to be stable for possible sharp gradients in all directions. This integration is left for future study.

Finally, we summarize the results in section 5.3.

5.2.1 Case I: Numerical diffusion in deep water (\(S = 0\))

We adopt the deep water diffusion test from Booij et al. [23] and compare our scheme to several other spectral wave models using first- and third-order finite difference, finite element (FEM), and finite volume (FV) schemes. Incoming long-crested waves through the bottom-left corner with a gap 150\(\sqrt{2}\) m have been prescribed at an angle of 45° in a domain 10 km \(\times\) 10 km (see Fig. 5.3a). The mean wave direction \(\bar{\theta}\) is 45° for this specific test problem. The incoming wave is assumed to be a Gaussian-shape in frequency direction and directional spectra form as \(\cos^m \theta\). The two-dimensional energy spectrum prescribed at the corner of the bottom-left boundary is

\[
E(\theta, \sigma) = A_1 \cos^m(\theta - \bar{\theta}) \frac{E_{\text{tot}}}{16\sqrt{2\pi}\sigma_0} \exp\left[-\frac{(\sigma - \sigma_{pk})^2}{2\sigma_0^2}\right], \tag{5.5}
\]

where we specified the standard deviation \(\sigma_0\) as 0.01 Hz and the directional parameter \(m\) as 500 (the resultant spectral width \(\sigma_\theta = 2.5^\circ\)) for significant wave height \(H_s\) as 1 m \((E_{\text{tot}} = 4\sqrt{H_s})\) at a peak frequency \(\sigma_{pk} = 0.1\) Hz.
We used two different resolution meshes, as shown in Fig. (5.3) with a varying order of Jacobi polynomials over the elements \((p = 2, 4, 6, 8)\). We employed fifteen collocations in \([-\pi, \pi]\) but mainly clustered around the directional interval \((\bar{\theta} - 3 \sigma_\theta)\) and \((\bar{\theta} + 3 \sigma_\theta)\). For a frequency domain in \([0.07 - 0.13]\) Hz (with Gaussian distribution, \(\sigma_{pk} \pm 3 \sigma_0\)), we used 16 collocation points.

This same study was conducted for a test of first- and third-order SWAN finite-difference ([23]), second-order Finite element method (FEM) [86], and second-order finite volume (FV) [125] in the past. We present here comparisons from Qi et al.’s work [125] in Figure 5.4. Among them, it is clear that the first-order SWAN code is the most diffusive scheme for even a relatively short distance (10 km). The other schemes’ relatively higher-order than the first-order SWAN scheme generated much less diffusion, outperforming the first-order scheme (see Figure 5.4). We note that this comparison is not fair since it used the about same grid resolution but different order-methods for the comparison. In this way, the computational cost of the
Figure 5.4: Significant wave heights (a) SWAN first order scheme (FD scheme, 10,000 grid points) (b) SWAN third order scheme (FD scheme, 10,000 pts) (c) FE-WAVE (about 20,000 triangles) (d) FVCOM-SWAVE (about 20,000 triangles) (reproduced with permission from [125]).

Figure 5.5: Case I: Significant heights (H_s) for a mesh (122 triangles) for two different polynomial orders.
methods would have been discarded.

Our simulation results are given for the coarse and fine meshes in Figures 5.5 and 5.6. We have obtained a steady-state solution using the integration scheme SSP-RK2. The relative computational work load can be approximately computed between two schemes A and B by the relation $\eta_{AB} = (P_A + 1)^2/(P_B + 1)^2$, where $P$ denotes the polynomial order of a scheme used on an element. Our cases, either in coarse or fine mesh, performed better than all the other schemes in Figure 5.4. The computational work of our scheme in the runs are either comparable or much less (up to 5 times cheaper in spatial space) than other schemes.

Figure 5.6: Case I: Significant height ($H_s$) for a mesh (966 triangles) for three different polynomial orders.
Among the results in Figure 5.4, the third-order scheme is considered superior to all other schemes since it is the highest-order accuracy scheme in Qi et al.’s work [125]. The third-order scheme will be much less dissipative and more accurate than the first or second-order schemes. None of the contour patterns in Figure 5.4 exactly matched each other (by focusing on a specific counter). However, looking at a specific counter level (0.4), our present results closely agree with those of the third-order SWAN scheme. The first-order scheme introduced serious numerical dissipation. As for the FV scheme, corresponding contour level is slightly higher than 0.4.

5.2.2 Case II: Current-induced shoaling and retractions in deep water \( (S = 0) \)

These cases are indented to test the scheme for waves of variable currents for deep water. We specified Gaussian shape energy spectra (5.5) as incoming waves (with \( T_{\text{peak}} = 10\text{s}, \ H_s = 1\text{m} \)) on the west boundary of a rectangular domain \( 4 \times 10 \text{ km} \) (see Figure 5.8). A current-induced shoaling will occur if incoming waves encounter a current going in the same or opposite directions but not a refraction (mean wave direction \( \bar{\theta} = 0 \)), which is schematically shown in Figure 5.7a. A current-induced refraction additionally happens with rotating both incoming waves with \( \pm 30^\circ \) and the current vectors with \( \pm 90^\circ \) (up and down) with respect to positive coordinate \( x \), which is also illustrated in Figure 5.7b. The computational grid given in Figure 5.8 used 70 triangles and fourth-order Jacobi polynomials \( (p=4) \) over the elements along with 16 directions in \([-15^\circ, 15^\circ]\) and 16 frequencies in \([0.05, 0.25]\) Hz. Our results will be directly compared to the analytical results obtained from kinematic conservation of a wave number \( (\sigma + kU = \text{const}) \) [117] in the case of no refraction (cases (a) and (b)).
\[
\frac{c}{c_0} = \frac{1}{2} + \frac{1}{2} \left( 1 + \frac{4}{c_0} \right)^{1/2},
\]
\[
\frac{H}{H_0} = \frac{c_0}{[c (c + 2 U)]^{1/2}},
\]

where \(c\) and \(U\) denote deep water propagation and current velocities, respectively. The quantities subscripted with 0 represent the incoming waves. The current velocity profiles shown in Figures 5.7a and 5.7b vary linearly from 0 to 2 m/s for all cases. Higgins et al. [104] extended the above analytical results for waves on a shearing current (cases (c) and (d)) as

\[
\sin \theta = \frac{\sin \theta_0}{\left[ 1 - \frac{U}{c_0} \sin \theta_0 \right]^{1/2}},
\]
\[
\frac{H}{H_0} = \left[ \frac{\sin 2\theta_0}{\sin 2\theta} \right]^{1/2}.
\]

\(\theta\) is the wave angle relative to the current direction. Higgins et al. found that in the cases of wave-current interaction, \(\theta \approx 0\) or \(\theta \approx 90°\), the net change in wave height was negligible.

We present significant wave heights \(H_s\) for all four cases in Figure 5.9. The
Figure 5.8: Case II: The computational mesh used for wave-current interaction tests. Fourth-order Jacobi polynomials (p=4) are employed in each element for all simulations.

Numerical results for the significant wave heights are in good agreement with the above analytical results. The mean wave direction changes along the coordinate x only for cases (c) and (d) but not for the other cases ($\bar{\theta}(x) = 0^\circ$). Both the analytical and numerical results of mean wave direction are plotted in Figure 5.10 and match very well along the coordinate x.
Figure 5.9: Case II: Comparison of significant wave height \( H_s \) versus distance from analytical equation and numerical simulations for all four cases of depth-induced shoaling and current-induced refraction (see the illustration of all cases in Figures 5.7a and 5.7b).

Figure 5.10: Case II: Analytical and numerical results are compared along the coordinate \( x \) for cases (c) and (d) (see the illustration of the cases in Figure 5.7b).
5.2.3 Case III: Depth-induced shoaling and refractions in shallow water \((S = 0)\)

Shoaling and refractions may occur due to depth changes in coastal waters; we model these phenomena next. We impose the incoming waves specified in the previous section on the west boundary of a rectangular domain \(4 \times 10\) km which has a linearly varying depth (from 20 m to 0.5 m) perpendicular to the coastal line. We consider two cases: (a) incident waves moving perpendicular to the coastal line, and (b) turning incident wave direction to \(30^\circ\) in respect to positive coordinate \(x\). Both test cases are sketched in Figure 5.11. The same computational grid (both spatial, as shown in Figure 5.8, and spectral space resolutions) as in the previous section is used but expanded with the Jacobi polynomial of the seventh-order over the triangles.

The numerical simulation results are compared to the solution of the linear wave theory for wave height as

\[
\frac{H}{H_0} = \frac{c_0}{c} \cos \theta_0 \cos \theta,
\]

where the wave direction can be calculated by using Snell’s law

\[
\theta = \arcsin[\frac{c}{c_0} \sin \theta_0].
\]

The significant wave heights are accurately predicted for incident waves moving to perpendicular to the coastal line (case a) and turning incident waves to \(30^\circ\) with positive \(x\) (case b) in Figure 5.12. The incoming wave directions perpendicular to the
coastal lines (case a) are not be refracted since there is no variation in coordinate y. The numerical prediction presented in Figure 5.13 showed no refraction happening for case (a), which agrees well with our expectations. The depth-induced refraction happens in case (b) where the incident wave turns over $30^\circ$. In the same figure, the mean wave direction deflecting toward the coastal lines is also predicted accurately by the numerical simulation.

**Figure 5.11:** Case III: Depth-induced shoaling and refraction test case.

**Figure 5.12:** Case III: The significant wave height ($H_s$) for case (a) $\theta_0 = 0^\circ$ and case (b) $\theta_0 = 30^\circ$
Figure 5.13: Case III: The mean wave direction ($\overline{\theta}(\circ)$) for case (a) $\overline{\theta}_0 = 0^\circ$ and case (b) $\overline{\theta}_0 = 30^\circ$. 
5.2.4 Case IV: Duration-limited growth in deep water ($S = S_{\text{in}} + S_{\text{nl4}} + S_{\text{we}}$)

Duration-limited wave growth will be simulated over a single physical point at which a constant wind blows for a very long time. Deep water has three source terms: wind generation, white-capping, and nonlinear wave-wave interactions, which contribute the evolution of the wave. The wave energy is generated and sustained by wind and dissipated by white-capping and redistributed energy among the frequencies by nonlinear wave-wave interactions. The physical space propagation velocities ($c_x$ and $c_y$) are set to zero to mimic our solver as if it is behaving as a single point solver. Spectral space propagation velocities would be computed as zero since the spatial derivatives of depth and current are zero (there is no current in this case).

The duration-limited test case adopted from Jannsen [88] assumes initial sea conditions as the JONSWAP spectrum, which has a peak frequency of $f_p = 0.34$ Hz, the scale parameter $\alpha = 0.025$, and shape parameters $\gamma = 3$, $\sigma_a$ and $\sigma_b$ as 0.1. The directional spreading of $\cos^2 \theta$ is specified with a directional resolution of $10^\circ$. The frequency distribution is specified by a geometric growth factor 0.05 ($f_i = 0.05 f_{i+1}$) with an initial frequency of 0.0418 Hz ($f_0$) and a final frequency of 1.0 Hz ($f_N$). Note that the frequency domain collocation points are not evenly distributed because Fourier-collocation is not active for this problem (there is no need to differentiate among the spectral space derivatives). The wind speed ($U$) over the single point is chosen as 18.45 m/s.

We will compare the simulation results of our implementation to those of a single-grid-point version of ECWAM (European Center Wave Model) [1]. In order to do this, we use only one element with zero order of Jacobi polynomial (there is no need
to resolve any spatial variation) and set the spatial propagation velocities as zero, which makes the code a single-point-grid solver if we interpret the solution on any point inside the single element (the solution will be constant over the element). The action balance equation is integrated from time $t = 0.0$ to $10^6$ s using an SSP-RK4 integration scheme with time step size $\Delta t = 100$ s. The parameters of source terms wind generation $S_{\text{in}}$, white-cap dissipation $S_{\text{wc}}$, and nonlinear-wave interaction $S_{\text{nl}}$ used in this case are given in Table 5.3.

The non-dimensionalized energy growth is compared to the results of ECWAM, and the empirical data fit for duration-limited growth from the North Sea [89] in Figure 5.14 along non-dimensionalized time. Energy and time are non-dimensionalized with respect to gravity $g$ and friction velocity $u_*$. Wave age non-dimensionalization also uses peak-propagation velocity $c_p(t)$. In Figure 5.14, our numerical simulation has resolved the initial growth region much better because we have used a fourth-order accurate scheme with a much smaller time step size. The ECWAM code has an implicit integration capability allowing to perform time integration with a time step of 15 mins.

Toba [151, 152] derived the 3/2 power law between significant wave height $H_s$ and significant wave period $T_p$, which led to the dimensionless energy being proportional to the third power of the wave age. The dimensionless energy and wave age results are plotted in Figure 5.15. Both numerical results from our implementation and ECWAM code are in agreement with empirical fits obtained by Jansen et al. [89].
Table 5.3: Case IV: Activated source parameters in Duration-limited growth (left) and Fetch-limited growth (right) test cases.

<table>
<thead>
<tr>
<th></th>
<th>Duration-limited</th>
<th>Fetch-limited</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_{in}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$u_{10}$</td>
<td>18.45 m/s</td>
<td>20 m/s</td>
</tr>
<tr>
<td>$A$</td>
<td>0.8</td>
<td>0.8</td>
</tr>
<tr>
<td>$B$</td>
<td>0.065</td>
<td>0.065</td>
</tr>
<tr>
<td>$S_{wc}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$C_{wc}$</td>
<td>3.33e-5</td>
<td>2.36e-5</td>
</tr>
<tr>
<td>$n$</td>
<td>2.0</td>
<td>0.0</td>
</tr>
<tr>
<td>$p$</td>
<td>4.0</td>
<td>4.0</td>
</tr>
<tr>
<td>$\tilde{s}_{PM}$</td>
<td>0.0676</td>
<td>0.0549</td>
</tr>
<tr>
<td>$S_{nl4}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\lambda$</td>
<td>0.25</td>
<td>0.25</td>
</tr>
<tr>
<td>$C_{nl4}$</td>
<td>3e7</td>
<td>3e7</td>
</tr>
<tr>
<td>$C_{sh1}$</td>
<td>5.5</td>
<td>5.5</td>
</tr>
<tr>
<td>$C_{sh2}$</td>
<td>0.85714</td>
<td>0.85714</td>
</tr>
<tr>
<td>$C_{sh3}$</td>
<td>-1.25</td>
<td>-1.25</td>
</tr>
</tbody>
</table>

Figure 5.14: Case IV: Dimensionless energy growth ($E^* = g^2 E_{tot} / u_*^4$) and time ($t^* = gt / u_*$) plot for the duration-limited wave growth. ECWAM (WAM modeling) and our numerical results are compared to the empirical fit (JKdV) by Jannsen et al. (1987) [89].
Figure 5.15: Case IV: Dimensionless energy ($E^* = g^2E_{tot}/u^4$) growth curve versus wave age ($\chi^* = cp(t)/u_\ast$) are presented for duration-limited growth. Wave age from left to right show the young sea transition to old sea. The present numerical results (●●●) are compared to ECWAM results (••••) and the empirical data (JKeV) (——) compiled by Janssen et al. [89].

5.2.5 Case V: Fetch-limited growth in deep water ($S = S_{ln} + S_{nl4} + S_{we}$)

A steady wind blows perpendicular to the coastal line generating waves growing gradually over distance. The energy steadily grows until it reaches a balance between atmospheric wind input and dissipation through white-capping and energy redistribution through nonlinear wave-wave interactions. We impose specific spectral waves on the west boundary of mesh given in Figure 5.16. Note that Figure 5.16 is in logarithmic axis; the first element edge (in coordinate $x$) is 1km and then the next edge grows as multiples of tenth of the initial edge (10km, 100km,...), which
results in a triangulation mesh of a total of 10 elements. The directional spectra spread out between $-90^\circ$ and $90^\circ$ with the frequency domain over $[0.04-1]$ Hz, which has a frequency resolution due to geometrical growth factor $0.1 (f_i = 0.1f_{i+1})$. The initial state of the domain is assumed to be a calm sea (zero energy). We have a coarse resolution in the spectral domain due to the absence of current in deep water ($c_\theta = 0$ and $c_\sigma = 0$). The wind input $S_{in}$, nonlinear wave-wave interaction $S_{nl4}$, and white-capping dissipation $S_{wc}$ parameters of this study are presented in Table 5.3.

The non-dimensional energy growth $E^* = g^2Etot/u^4_*$ over fetch $x^* = gx/u_*^2$ is plotted against the generalized energy growth formulas (B&H) [32] obtained from data sets over Lake George, Australia [173] and the data sets of the SWAMP Group [149] in Figure 5.17. In the present simulations we ran implementation for cases using second, fourth, and eighth-orders of Jacobi polynomials over the elements. The energy growth is captured reasonably well and agrees with both the observed data set (B&H) [32] and the (SWAMP) envelope results of various numerical ocean wave modeling codes [149].

Using a higher polynomial clearly resolved the wave growth for smaller fetches, as seen in Figure 5.17. We used a constant polynomial order over elements which vary greatly from one to another in size. It would be better to use variable polynomial orders based on the size of the element to obtain optimal errors. It is worth mentioning that the SWAN code [23], FVCOM-SWAVE [125] and FEM SWAN [86] used about three to four orders of magnitude finer meshes than the mesh of the current simulation in order to predict accurately the fetch-limited energy growth.
5.2.6 Case VI: Depth-induced wave breaking \((S = S_{br})\)

Depth-induced wave breaking is a dominant dissipation mechanism in shallow water. Here, we have used the modeling by Battjes & Janssen [13]. This study will compare the results from both SWAN and our simulation to those of the laboratory experiments conducted by Battjes & Janssen [13].

Incoming incident waves are generated on the left boundary of the experimental setup. The schematic representation is given in Figure 5.18b. We have assumed a JONSWAP spectrum with significant wave height \((H_s)\) 0.202 m and a peak frequency \((f_p)\) 0.53 Hz for the incident wave. The directional spreading is cos\(^{500}(\theta)\) narrow spectra distribution. A relatively coarse mesh (see Figure 5.18a) with seventh-
order Jacobi polynomials is used in geographical space; the directional and frequency derivatives, respectively, employed 32 collocation points in the domains $[-15^\circ, 15^\circ]$ and $[0.25, 3.125]$ Hz. In our implementation we only activated the depth-induced wave breaking source term $S_{br}$; however, the SWAN code has additionally activated triad-wave interactions, which apparently led to slightly more dissipation, which will be investigated in the next test case. We used the breaker parameter $\gamma$ as 0.73, which corresponds to an average value for different types of bathymetry [14].

We have simulated this case and compared the significant wave height $H_s$ with SWAN and the experiment along coordinate $x$ in Figure 5.18c. The significant wave height is slightly underpredicted for the first quarter of the submerged bar (0-5 m) due to the deactivated triad-wave interactions in our simulation (triad dissipations
Figure 5.18: Case VI: Significant height variation in the Battjes and Janssen (1978) experiment. SWAN and the present simulation used $\gamma = 0.73$ and $\alpha_{BJ} = 1$ in the depth-induced wave breaking source term.
are not homogeneous). The remaining part of the experimental results is in good agreement with both SWAN and our simulation. The significant wave heights have not changed much in the trough region (9.92-17 m) where almost no breaking happens in the experiment (observed visually). Towards the end of the region, the experimental results show again an increase in significant wave height, which clearly shows that the rate of dissipation due to wave-breaking is slower than that of the potential energy (results in shoaling) of the waves [13] in very shallow water. Both numerical simulations using the same wave-breaking modeling introduce more dissipation toward the end of the experiment, unlike the results of the experiment in which they started rising again. In the limit of shallow water, the fraction of breaking waves \( Q_b \) will approach one and the wave breaking dissipation will decrease proportional to the depth, \( h^2 \) [13]) (a dissipation decrease paves the way for shoaling again).

\[ S = S_{br} + S_{nl3} \]

5.2.7 Case VII: Flume test case

Next we consider another experimental work by Wood at al [166] to compare our method along with the SWAN code in case that the triad term is activated. The triad effects in depth-induced wave breaking are briefly mentioned in [130]. According to Ris’ work [130], the activation of the triad term tends to slightly decrease the significant wave heights. Our results support a decrease of significant wave height (more explicitly, energy) that is not homogeneous in the domain. We observed that the total energy has been more dissipated when the triads are active. However, we have additionally observed that the dissipation is not uniform along the coordinate \( x \) such that it might increase the significant wave height locally even though it introduced more dissipation in total.
Triad source term modeled by lumped-triad-approximation (LTA) has been recommended not to be activated in the SWAN model in Wood et al.'s work [166] due to additional dissipation (other than depth-induced breaking) in the prediction of waves in a flume. We will simulate waves of Wood et al.'s experimental setup, whose side view is given in Figure 5.19a, where crosses (x) are the measurement gauges with three different bottom slopes, respectively, 1:12, 1:20, and 1:30. We use an extremely coarse mesh with only two triangular elements (see Figure 5.19b), and Jacobi polynomials (p=18) are expanded over the elements. The spectral space resolution has 32 collocation points for directional \([-15^\circ, 15^\circ]\) and frequency \([0.2, 2.0]\) Hz domains.

Energy frequency distribution based on spectra from experimental data is noisy (the blue dots in Figure 5.20a), hence it needs to be smoothed through a filtering. This filtered data is enforced at measurement location 1 as a boundary condition for the SWAN code in Wood et al. [166]. We specified a boundary spectrum different from Wood et al.'s work (the original data is not available to us) such that the JONSWAP spectrum is enforced on the west boundary with a peak frequency 0.6 Hz, scale parameter \(\alpha 0.0137\), and shape parameters \(\gamma, \sigma_a\) and \(\sigma_b\) 3.3, 0.7 and 0.09, respectively. This way of imposing boundary conditions resulted in smooth energy spectra unlike those of SWAN in Figures 5.20a and 5.20b.

The energy spectra at wave gauges 1 and 12 are presented in Figures 5.20a and 5.20b. The energy spectra of both numerical simulations (the SWAN and our code) agreed best with the spectra of the experiment in Figure 5.20b if triads source terms (LTA) are turned off. It seems also that the experimental data does not show any first \((f = 2f_p = 1.2\) Hz\) or second \((f = 4f_p = 2.4\) Hz\) super-harmonics. However, the LTA model generates super-harmonics which are apparent in our numerical and SWAN results in Figure 5.20b. Energy spectra for activated triad interactions in Figure 5.20b have shown to be underestimating the energy spectra by both our sim-
ulation and the SWAN codes. The triad interactions transform energy distribution by extracting energy from the primary peak frequency and adding to its super-harmonic [82, 54]. Our code simulates the lumped-triad approximation (LTA) of Eldeberky [54] (see blue or clay line results in Figure 5.20b). The coefficients of LTA ($\alpha_{EB}$) used in the SWAN input are not clear in Wood et al. [166]. We used two different values of ($\alpha_{EB}$) 0.05 and 1.0. The energy spectra showed no significant difference between the two different values of ($\alpha_{EB}$) in Figure 5.20b. The larger values of coefficient $\alpha_{EB}$ are apparently generating slightly more dissipation.

Figure 5.20c presents comparisons of significant wave heights for both numerical simulation and experimental measurements. SWAN simulation without triad interactions matched better with experimental results than our simulation does after the point of the flume where the slope becomes zero. It seems that our depth-induced wave breaking introduced more dissipation after that point. However, once the triads are activated, the SWAN result is more dissipative in the ramp region than our results. As for the triad coefficients, the case of the largest value of ($\alpha_{EB}$) 1.0 has shown increasing significant wave height from the bottom of the ramp to the middle of the slope. The rise of significant wave height in this portion of the experiment will cause further wave-breaking dissipation. We see that this case is more dissipated toward end of the flume.

Our numerical experiments agree with Wood at al. that lumped-triad approximation (LTA) is better to be excluded in the numerical model if depth-induced wave breaking dominates; otherwise, it will introduce more dissipation than necessary.
5.2.8 Case VIII: HISWA Experiment \((S = S_{b,fr} + S_{br} + S_{nla})\)

The HISWA experiment [52, 51] is a laboratory experiment conducted for random, short-crested waves to validate the HISWA numerical spectral ocean model which is second-generation model [81]. The experiments served two purposes: (1) to have the comparison of the computations and the measurements for future use (2) and to provide the detailed measurements if there is a need for future validations. The experiment is one of the most comprehensive works for wave propagation in a laboratory. They were run for the three different bathymetry shapes as (1) a flat basin, (2) a simple one (fully cylindrical bar), and (3) a complicated one (semi-cylindrical bar with a rounded head), as well as for many different input and boundary conditions. We chose the complex bathymetry (semi-cylindrical with rounded head; see its depth counters in Figure (left) 5.22) for this study. The shape of the submerged breakwater can be exactly generated by using the transformations given in the reference work.
Figure 5.20: Case VII: Significant height variations in the Wood et al. (2001) experiment; the SWAN and present code results are compared. Here $\gamma = 0.73$ and $\alpha_{BJ} = 1$ in the depth-induced wave breaking source term. The triad proportionality coefficient $\alpha_{EB} = 0.05$ and 1.0 are used if the triad source term is activated.
In this study we chose the case me35 where ‘3’ and ‘5’ denote, respectively, bathymetry of semi-cylindrical bar with round over and a specific input in the reference work [51]. The input case 5 has specified incoming waves (perpendicular to $y$-axis) with a significant wave height of 10 cm, with a peak period of 1.25 s and with directional spreading of $25^\circ$. The case 5 is considered to be the most commonly occurring one in nature since it includes a complicated bathymetry, which induces refraction and shoaling as well as a complex current field which induces refraction and shoaling. The current field has been measured at 81 points on a grid of 3 by 3 m at half the water depth.

This experiment was later used to validate the SWAN model [130]. The experiment is done on a relatively large rectangular basin 26 m $\times$ 34 m. The wave maker generates waves from left at the shorter edge (along $x = 0$ line, see Figure 5.22) to right.

We have simulated the HISWA tank experiment by using the high-order ocean wave solver. The physical computational domain (see Figure 5.21) $[(0,30) \, m \times (-45,45) \, m]$ is discretized into 48 triangular elements accompanied with a spectral space mesh whose frequency axis ranges from 0.4 Hz to 3.0 Hz with the frequency resolution 0.04 Hz and the directional domain lying in $[-60^\circ, 60^\circ]$ with a resolution about $4^\circ$ around the center and about $20^\circ$ around the tails of the directional domain. The directional collocation points are generated by the local refinement using the arctan mapping as given in Eq. (4.38); the mapping clusters the collocation grids around the center. The equi-spaced collocation grids are employed in the frequency direction. The directional domain size $[-60^\circ, 60^\circ]$ is enough to cover most of the energy for the me35 case that has the directional spreading width given as $25^\circ$. We
have used the left absorbing boundary layer as 0.2 Hz and the right one as 1.0 Hz, which effectively extended the frequency domain from [0.4, 3.0] Hz to [0.2, 4.0] Hz.

![Figure 5.21](image)

**Figure 5.21:** The computational mesh (48 triangle elements) used for the HISWA simulation. The dashed region is the experimental region.

We applied the incoming energy spectrum on the left boundary by the JONSWAP spectrum with directional spreading model $\cos^m(\theta - \theta_0)$ [82], where $\theta_0$ is the reference wave direction and ‘$m$’ is chosen as four based on the given directional width of 25°. The JONSWAP spectrum [72, 82] of peak frequency $f_p$ used here of 0.8 Hz, scale parameter $\alpha$ 0.0154, peak-width parameters $\sigma_a$ 0.07 and $\sigma_b$ 0.09 are chosen to represent the significant height of 10 cm for the incoming waves on the left boundary condition. The centered reference direction ($\theta_0$) in the directional distribution is interpolated from measurement locations 1-2-3 (along the $y$ axis, see Figure 5.22) and projected on the left boundary. The computational boundary outside the ray A and ray D lines (see Figure 5.22) assume the reference direction (\(\theta_0\)) zero. The lateral boundaries are naturally reflective, but our implementation does not take this into account, which is not significant in this case (me35) since, in the experiment,
wave generators send waves perpendicular (not obliquely) to the left boundary. We specified zero energy along these boundaries. This is inherently a dissipative process and pollutes the numerical solution inside the domain. We extended the lateral boundaries $y_{min}$ to -45 and $y_{max}$ to 45 for minimizing the numerical boundary pollution in the region of interest (see dashed region in Figure 5.21). The experiment was designed to minimize any reflection on the right end and hence we set non-reflecting boundary conditions at this end. The initial condition was set to zero energy in the domain for all runs.

The interpolated current field from the experiment was provided by N. Booij at Digital Hydraulics Holland (private communication). This current field is interpolated to the quadrature points in the simulation. The current field of the computational grid outside the experimental area is assumed to be zero. The vector field of the current is shown in Figure (right) 5.22.

**Figure 5.22:** The sensor locations and the bathymetry contour (left) and current vector field (right).

The bottom friction and depth-induced wave breaking are the important dissipation processes with nonlinear wave-wave interactions (only triads). The bottom friction process ($S_{b,fr}$) with bottom friction coefficient $C_{bfr} = 0.067$, the depth-induced
breaking process $S_{br}$ with $\gamma = 0.81$ and the scaling coefficient (after Battjes and Janssen) $\alpha_{BJ} = 1.5$, triad interactions ($S_{nl3}$) with the scale factor $\alpha_{EB} = 0.5$ are all represented in our model. The source parameters are chosen by the suggestion of Ris’ work [130] whose results we used here to compare our results.

The computation is carried out by marching in time (the implementation currently supports only unsteady computation) to reach a steady state solution. To this end, we march the unsteady solution with the time step 0.025 s up the final time 35 s, which at a time, we see conclusively that further solutions no longer change. The Runge-Kutta 2nd order time integration scheme is employed for all HISWA cases. The first, third, and fifth order of the Jacobi polynomial are expanded over the triangles in the computation of this section.

The measurement locations of 26 stations are given in Figure (left) 5.22. We have measurements for the significant wave height ($H_s$) and mean wave period ($T_{m01}$) for all 26 locations, but mean wave direction ($\bar{\theta}$) for only seven locations (1, 2, 3, 17, 24, 25, 26) from the HISWA experiment. The comparison is made on four horizontal lines (Rays A, B, C, D) rather than whole measurement locations to simplify the further discussion on the comparison. The four rays cover 24 stations and exclude only two stations 12 and 17. The energy spectra are compared for only eight locations (5, 3, 13, 15, 16, 19, 24, 26) in this study.

We used the same directional and frequency resolutions while employing the three different Jacobi polynomial orders expanded over the triangles. The plots of integrated parameters ($H_s$, $T_{m01}$ and $\bar{\theta}$) and energy spectra $E(f)$ presented, respectively, in Figures 5.23, 5.24, 5.25, and 5.26 have shown significant improvement from the first-order to the third-order polynomials. The high-order polynomial results ($p=3, 5$) in Figures 5.23, 5.24 and 5.25 have clearly improved the integrated parameters
(\(H_s\), \(T_{m01}\) and \(\bar{\theta}\)) on the ray B, C, and D locations.

Energy spectra are shown in a two set group of measurement stations on Ray B (5, 13, 16, 24) and Ray D (3, 15, 19, 26). The rays B and D are the most interesting lines due to complex bathymetry and current field. The stations on ray A are exempted from the comparison because there is no change in the depth of the gradient along this ray. There is no comparison also made on the stations of ray C since the waves on ray C have similar features to the waves on ray D. Energy spectra plots given in Figure 5.26 have shown a relatively large improvement at stations 13, 15, 16 and 19 than other stations 5, 3, 24 and 26 for the higher polynomial orders (\(p=3, 5\)). The first set of groups shows better improvement over the increasing polynomial order since they are located in a strong varying field of the depth and the current. The results of 5 and 3 stations do not improve with increasing Jacobi polynomial orders. They are clearly in the least interesting region where the depth does not change around those stations. The other stations 24 and 26 show some improvements for the higher order polynomials than those of 5 and 3 since they are exposed to a stronger gradient of current field. Moreover, our new high-order scheme seems always to generate secondary harmonic peaks. The new scheme predicted the secondary peaks better than SWAN at stations 13, 24, 19, and 26. The experiment has shown that second peak frequency does not always occur at a harmonic of peak wave frequency (see stations 16 and 15 in Figure 5.26). This discrepancy between experiment and numerical simulation can be attributed to inadequate modeling of triad source terms used here [55] in the numerical simulation.

The results given in Figures 5.23 and 5.24 agree well with the experiment for the significant wave height and mean wave period for the third and fifth orders of Jacobi polynomials. The reference SWAN simulation is taken from Ris’ work [130]. The high-order method predicted these two statistical parameters better than SWAN.
specifically on Rays B, C, and D. The experimental results of the mean wave direction have only seven stations where the measurements are done. We have a limited data for the experimental measurement of mean wave direction (7 out of 26). Along Ray A, both SWAN and our codes agree well with each other and match experimental only available data at station 1. The results can be seen in Figure 5.25. However, neither code predicted mean wave direction accurately on the stations behind the bar (stations 24, 25, 26).

![Figure 5.23](image-url): Comparison of significant heights ($H_s$) from three different polynomial orders ($p=1, 3, 5$) and experimental and SWAN results.
Figure 5.24: Comparison of mean wave periods ($T_{m01}$) from three different polynomial orders ($p=1, 3, 5$) and experimental and SWAN results.

5.3 Results and discussion

We have presented a high-order numerical method to solve phase-averaged ocean wave equations. Accuracy verification has shown that the proposed scheme has exponential convergences in both spaces (physical and spectral). The high-order scheme results in minimal dissipation in the diffusion test case (case I) even when using a coarser mesh than other available codes. Results from current-induced shoaling/refraction (case II) and depth-induced shoaling/refraction (case III) tests agreed well with corresponding available analytical results. Again, these simulations only
used a few high-order triangular elements. The duration-limited growth (case IV) case tested our solver as a single point solver and the result of this test matched well with available observations in the literature. The fetch-limited growth test (case V) admittedly was the toughest among the test cases because we have only an explicit time integration scheme to reach a steady-state solution with a relatively very small time step. We ran unsteady simulation to reach a steady case, which is extremely long with a CFL restricted small time step. We have been able to run cases for various orders of polynomial order. The solutions in general agree with available observation data. Higher order polynomials improve the solutions significantly.
However, we have observed an instability, which we attribute to quadruplet interactions [153], which are filtered out to stabilize the scheme. The depth-inducing break
(case VI) and flume test (case VII) validate our implementation against laboratory tests. Comparisons are also made to SWAN codes. One should not expect this implementation to be more accurate than other ocean models in a realistic simulation because modeling errors in the source terms may dominate overall the simulation. However, if we exemplify a depth-induced test (case VI) using very coarse mesh of 24 triangular high-order elements (p=7) and a flume test (case VII) employing extremely coarse mesh of 2 high-order elements (p=18), we captured the wave height and energy spectra as accurately as in the SWAN code, which needs at least to use three orders of magnitude finer resolution meshes, pointing to the efficiency of the scheme proposed here.

We have presented the validation of a new high-order scheme proposed in Chapter 4 for the HISWA experiment. The validation used a coarse triangular mesh in which different Jacobi polynomial orders (p = 1, 3, 5) are expanded over the triangular elements. The first order of Jacobi polynomial expansion was too low to capture the important wave parameters of the HISWA experiment. The third and fifth orders of the Jacobi polynomials significantly improved the result on the same coarse mesh. The new high-order scheme predicted slightly better for significant wave height and mean wave period parameters (see Figures 5.23 and 5.24) than SWAN did. The scarcity in the measurement hindered us in comparing the results of the new scheme and SWAN for the mean wave direction. The energy spectra results (see Figure 5.26) show that the new scheme captured the secondary peaks better than SWAN, which underpredicted the secondary peaks.
Chapter Six

Uncertainty Quantification for Numerical Ocean Wave Prediction
The source of uncertainty comes from many inputs such as modeling errors, numerical discretizations, initial and boundary conditions, geometry and medium properties. In this study, we will restrict the uncertainty inputs to the tune-up source parameters and current field.

The spectral ocean equation represents the important wave physics (generation, dissipation, and nonlinear wave interactions) through source terms. The available code can predict the statistical parameters (significant wave height, mean wave period,...) of ocean waves. The prediction relies on source term modeling. The numerical ocean wave community has been actively doing research to improve them [160] for the last three decades. Over the years, improvements in the source terms or adding new physics into the source terms have been called first, second, and third generation, and more recently, fourth generation ocean wave model. Among them, the third generation is still the most common model. The reliability of ocean wave prediction, in most cases, depends on tune-up parameters of the source terms. These parameters can be tune up with each problem to improve the prediction. Among the source terms, we have exact mathematical expression for only quadruplet wave-wave interactions [69, 70, 71]. As mentioned before, this is also modeled by Hasselmann [75, 76] due to prohibitively expensive computation of the exact expression. The rest of the important parameters including dissipation (white-capping, depth-induced breaking, and bottom friction) and triads which no explicit mathematical expression (not well understood) is modeled [160] into the source terms. Aside from the modeling errors in the source terms, the errors are also attributed to the uncertainty of the source term parametrization.

In this chapter, we will do uncertainty quantification of HISWA experiment [51, 52, 130]. To this end, we will treat the parameters $\gamma$, $\alpha_{BJ}$ of depth-induced wave breaking source term, the bottom friction coefficient $C_{bfr}$ of the bottom friction term
and the triad coefficient $\alpha_{EB}$ of triad interaction modeling as random variables.

The uncertainty comes from other sources than the source term parametrization such as the current field and the boundary and initial conditions. The latter ones are called random processes since they are now a function of space. In our case, we have calm water at the initial (no uncertainty). We also ignore the uncertainty associated with boundary conditions and bathymetry. But the current will be incorporated into our stochastic simulation as random process.

The collocation stochastic simulation requires many runs to predict the important statistical quantities such as mean and standard deviation. The deterministic solvers of the spectral ocean wave model, which is already computationally very demanding, might need a great number of runs in the stochastic simulation. They certainly need to run on high-performance computers (HPC). Most HPC administration limits users for run time, the number of nodes, etc. This typically forces stochastic simulators to submit their jobs by several installments, or a node can fail and may need to resubmit the job to the system, and so on. One can easily find the overall process burdensome. This study has used a new python program PUQ [3] to generate the Smolyak grids, support a variety of probability distribution of inputs, and schedule all job submission with great ease. The PUQ can easily work with any solver (solver assumed as black box) and any job scheduler (PBS, MOAB, ...).

All runs of the deterministic solver (here high-order ocean wave code) employed 3rd order Jacobi polynomial expansion on the triangles of a fixed mesh (see the mesh in Figure 5.21). The time integration scheme, boundary condition, and spectral space size and resolution are all kept the same as before in the previous chapter.

We first deal with uncertainty in source terms in section 6.1 and then with current
field in section 6.2. We combined both stochastic inputs in section 6.3. Each section has its own discussion in the sections. We finally summarize findings in section 6.4.

6.1 Uncertainty in source terms

The source terms depth-induced, bottom friction, and triads are activated in the code for the HISWA simulation. The depth-induced breaking has two parameters treated as random variables: the scaling factor $\alpha_{BJ} \in [1.0, 2.0]$ with the center value of 1.5 and the wave breaking coefficient $\gamma \in [0.6, 1.0]$ with the center value of 0.8. The bottom friction coefficient $C_{bfr} \in [0.038, 0.096]$ with the center value of 0.067 is the only random variable in the bottom friction term. The value of 0.038 is recommended for swell conditions by the JONSWAP experiment [72] and 0.067 by Bouws&Komen [25] for fully developed wave conditions in the shallow water. The tune-up parameter of the triad source term $\alpha_{EB} \in [0.0, 1.0]$ with the center value 0.5 is the final random variable in the source term.

The probability distributions of the tune-up parameters are not available to us. We will assume that the tune-up parameters are characterized by the uniform probability distributions in this study.

6.1.1 Sensitivity analysis of source term parameters

The elementary effect method [112] gives the sensitivity result from a small number of model evaluations. Let’s consider a function $\mathcal{F}$ that has $k$ independent input parameters such that
\[ F = f(X_1, X_2, \ldots, X_k), \]

where \( \mathcal{X} = (X_1, X_2, \ldots, X_k) \) varies in \( k \) dimensional cube with \( p \) level which selects sampling locations on each dimension. The elementary effect is defined as:

\[
d(X_i) = \frac{F(X_1, X_2, \ldots, X_{i-1}, X_i + \triangle, \ldots, X_k) - F(X_1, X_2, \ldots, X_k)}{\triangle} \quad (6.1)
\]

where \( \triangle \) is a value chosen equal to \( p/(2(p - 1)) \) for even \( p \). The method randomly chose \( r \) sample points among \( p \) levels \( (r < p) \). The number of \( r \) samples can be considerable smaller than \( p \) and hence the method needs a few \( k \times r \) output evaluations to compute the sensitivity. It is obvious that the sparse grid is well suited to the space definition of the elementary effect method \( (k \) dimensional cube and \( p \) level). The random \( r \) sample points can be taken from the Smolyak grid. The elementary effect method uses two measures for mean \( (\mu) \) and standard deviation \( (\sigma) \) of \( r \) elementary effects of each input. The mean \( (\mu) \) measures the relative influence factor of each input. The mean of \( r \) sampling elementary effects is defined \( (\text{for input } X_i) \) as

\[
\mu_i = \frac{1}{r} \sum_{j=1}^{r} d(X_i^{(j)}). \quad (6.2)
\]

The standard deviation \( \sigma \) assesses non-linearity with other factors \([136]\). The software PUQ has the capability of post-processing output functions to compute the mean and standard deviations of \( r \) sampled elementary effects for the input parameters.
Figure 6.1: The elementary effect values for significant height \((H_s)\) for four source parameters \((C_{bfr}, \alpha_{BJ}, \gamma, \alpha_{EB})\).

We plot the mean values of input elementary effects for significant wave height, mean wave period, and mean wave direction, respectively, in Figures 6.1, 6.2, 6.3. The values of the elementary effects signify the degree of importance of each input sensitivity for a specific output. The values of the effects are only meaningful to be compared inside the same figure. We present results for all 24 locations (rays A-D) already given in Figure (left) 5.22. The first seven stations (1-7) are the front measurement locations before the submerged water breaker with the flat bottom. As seen in Figures 6.1, 6.2, and 6.3, the elementary effect values are very low for the three outputs at these stations. This clearly shows that the numerical wave solution
is not sensitive to the tune-up parameters there.

The elementary effect values of the significant wave height ($H_s$) in Figure 6.1 shows that the most important parameter at all locations is $\gamma$, the parameter of depth-induced breaking. The importance ranking among other remaining inputs $C_{bfr}, \alpha_{BJ}, \alpha_{EB}$ varies along with the different locations for this output.

The triad coefficient ($\alpha_{EB}$) becomes the dominant tune-up parameters if we look at the elementary effect values of mean wave period ($T_{m01}$) in Figure 6.2. The depth-induced parameter ($\gamma$) follows the triad coefficient in importance. It follows then that
Figure 6.3: The elementary effect values for mean wave direction ($\bar{\theta}$) for four source parameters ($C_{bfr}$, $\alpha_{BJ}$, $\gamma$, $\alpha_{EB}$).

The bottom friction coefficient ($C_{bfr}$) and the other depth-induced parameter ($\alpha_{BJ}$) have in the same order of sensitivity for mean wave period output.

The depth-induced coefficient ($\gamma$) takes the lead again for the sensitivity of the mean wave direction output ($\bar{\theta}$) as seen in Figure 6.3. The remaining parameters $C_{bfr}$, $\alpha_{EB}$, and $\alpha_{BJ}$ have about the same order of influence on the mean wave direction.

The important conclusion from the sensitivity of the three outputs here is that the dominance of elementary effect of each input for three outputs ($H_s, T_{m01}, \bar{\theta}$) is not uniform in the space and, therefore, we shall not exclude any of the random
variables for reducing the dimensions of random space in this study.

6.1.2 Results and discussion

We will use the four parameters of the source terms as the random variables. The mean ($\mu$) and standard deviation ($\sigma$) are evaluated by using the second, third, and fourth order gPC expansion. The four random variables using the Smolyak sparse grid yield 41, 137, and 401 deterministic runs, respectively, for the second, third, and fourth order expansions. We present mean and standard deviations for the three important parameters ($H_s, C_{bf}, \bar{\theta}$) in Figures 6.4, 6.5, and 6.6. The mean values ($\mu(\cdot)$) of the three parameters (the left-column of the figures) did not change at all for increasing the order of expansion. We preferred to use the relative percentage of the mean ($\mu(\cdot)$) and the standard deviation ($\sigma(\cdot)$) ratio ($\frac{\sigma(\cdot)}{\mu(\cdot)} \times 100$) for the significant wave height in Figure 6.4 and the mean wave period in Figure 6.5 for the comparison. But we used the standard deviation for the mean wave direction in Figure 6.6. The third and fourth order expansions did not improve the statistical quantities (the mean and the standard deviation) in the significant wave height (see Figure 6.4). However, the third order improves the standard deviation in the mean wave period and the mean wave direction (see Figures 6.5 and 6.6). The fourth order expansion slightly changes the statistical quantities of all three wave parameters. This confirms the convergence of the expansion.

Figures 6.4, 6.5, and 6.6 also confirm that the stations of ray A have the lowest standard deviation values. This is somewhat expected due to the flat bottom and small variation of the current field. Another observation from the standard deviation of all three outputs is that the most uncertainty is pertinent to the region behind the bar. The standard deviation values start rising at $x > 12$ m as seen in the
right-column figures of Figures 6.4, 6.5, and 6.6.

Figure 6.4: The mean ($\mu$) and relative percentage of standard deviation ($\sigma$) and mean ratio of significant wave height ($H_s$) from three level expansions (2,3,4) for the four source parameters ($C_{bf}, \alpha_{BF}, \gamma, \alpha_{BF}$).

We present the error bar plots with a confidence interval about 95% ($\mu \pm 2\sigma$)
around the mean value for the stations on rays A-D in Figures 6.7, 6.8, and 6.9. Figure 6.7 shows that the confidence interval does not cover the experimental results
Figure 6.6: The mean ($\mu$) and relative percentage of standard deviation ($\sigma$) and mean ratio of mean wave direction ($\bar{\theta}$) from three level expansions (2,3,4) for the four source parameters ($C_{bfr}$, $\alpha_{BJ}$, $\gamma$, $\alpha_{BJ}$).

for the stations on ray A. As mentioned before, the standard deviations have low values along ray A. The real issue is that the numerical solution does not match well
with that of the experiment due to the fact that we have current field data available for only the physical region but not for the computational grid. As mentioned earlier, we extended our lateral boundaries from $y \pm 13$ to $y \pm 45$. The current field is assumed to be zero in this extended region ($-45 \leq y < -13; 45 \geq y > 13$). This discrepancy might be reason of the numerical solution not matching well along the stations of ray A. The error bars on the remaining stations on rays B-D for significant wave height and mean wave period are given, respectively, in Figures 6.7 and 6.8. They confirm that the confidence interval of 95% covers well the experiment results.

Furthermore, we observed that the confidence intervals vary greatly at the region ($x \geq 15$) behind the submerged breaker. The scarcity of measurement locations (only 7 out of 26) for the mean wave direction makes the judgment harder as to whether or not the confidence intervals (see Figure 6.9) cover the experimental measurements. The confidence interval only covers the measurement location on Ray B (see ray B subplot in Figure 6.9). We see again that the submerged behind breaker is the most uncertain region for the mean wave direction. The stations on rays B and C have the largest standard deviation values of the mean wave direction in this region due to relatively complex bathymetry and the strong current gradient along the B and C.

We finally present the error bar plots within 95% confidence intervals for the energy spectra of the selected stations on rays B and D in Figure 6.10. The numerical simulation and experimental results are compared for the energy spectra of stations on ray B (5, 13, 16, 24) and ray D (3, 15, 19, 26). The left-column and right-column subplots are arranged such that the left plots are for the stations on ray B and the right plots are for the stations on ray D in Figure 6.10.

The mean value of the energy spectra of stations (red dots) matched experimental spectra better on ray B than the stations on Ray D for the primary peaks. The
Figure 6.7: Significant wave height $H_s$ of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the fourth order expansion for the source parameters ($C_{bf,fr}$, $\alpha_{Bfr}$, $\gamma$, $\alpha_{Bf}$).

secondary peaks arise in the numerical simulation but are not produced as strongly in the measurement. The numerical simulation on the primary peak is overestimated for station 13 but the peak of station 16 is underestimated. Additionally, the numerical solution overestimated the secondary peaks for these two stations. The error plots (left-columns) of energy spectra in Figure 6.10 show that the confidence intervals mostly capture the experimental spectra. The more the disagreement between the matching numerical and experimental results, the larger the error bars that show up.

As for the stations on ray D, the mean value of the numerical simulation predicted that the first peaks would be slightly off along the frequency axis for the stations (15, 19) (see right-column plots in Figure 6.10). The standard deviation values are particularly very large around the secondary peaks for these stations. The confidence
The controversial parameters of the source term are treated as the random variables in this section. Most of the experimental results have been contained in the confidence interval within 95% of numerical results with the stochastic simulation. The stochastic simulation of the spectral ocean wave equation with random inputs of source terms gives us an invaluable tool to quantify the degree of confidence of our numerical results.
Figure 6.9: Mean Wave Direction $\bar{\theta}$ of simulation results (confidence interval within 95\% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the fourth order expansion for the source parameters ($C_{bf}r$, $\alpha_{BJ}$, $\gamma$, $\alpha_{BJ}$).

6.2 Uncertainty in current field

We take the two-dimensional current field $\vec{u} = (u_x, u_y)$ as the stochastic input in this section. The Karhunen-Loeve (KL) decomposition [102] of a Gaussian correlation function is used to represent the stochastic current field. The KL decomposition represents a low dimensional description of the Gaussian correlation function. The random process, which is a function of the space, is obtained by adding perturbation of KL expansion terms to the experimental realization $\bar{u}(x,y)$. The random process of the current field now reads
Figure 6.10: Energy spectra of simulation results (confidence interval within 95% (μ ± 2σ)) and experimental data are compared. The sparse grid used the fourth order expansion for the source parameters ($C_{bfr}$, $α_{BJ}$, $γ$, $α_{BJ}$).

\[ u(x, y, ω) = \bar{u}(x, y) + \delta \sum_{i=1}^{∞} \sqrt{λ_i}ψ_i(x, y)ξ_i(ω), \quad (6.3) \]
where \( u(x, y, \omega) \) denotes the random processes, \( \bar{u}(x, y) \) is the current mean, \( \{\xi_i(\omega)\} \) is a set of uncorrelated random variables with the zero mean and the unit variance \( (\sigma^2 = 1) \), and \( \delta \) is the scalar to control the magnitude of the perturbation. The \( \{\psi_i(x)\} \) and \( \{\lambda_i\} \) are the eigenfunctions and eigenvalues of the covariance kernel \( R_{uu}(x_1, x_2) \), that is

\[
\int R_{uu}(x_1, x_2)\psi_i(x_2)dx_2 = \lambda_i\psi_i(x_1). \tag{6.4}
\]

We use here the Gaussian correlation function:

\[
R_{uu}(x_1, x_2) = \exp \left[ -3\frac{(x_1 - x_2)^2}{l_c^2} \right], \tag{6.5}
\]

where \( l_c \) denotes the correlation length. This length \( l_c \) determines the dimension of random space. The random perturbation in equation (6.3) is in practice truncated to the \( P \) finite dimension. The desired number of finite dimension \( M \) can be chosen by a specified percentage of the perturbation energy \( \left( \sum_{i=0}^{P} \lambda_i / \sum_{i=0}^{M} \lambda_i \times 100 \right) \). We use here three different correlation lengths relative to experiment domain length: 30, 25, and 20. The resultant number of dimensions of the random space to capture 90% of the energy for the three correlation lengths can be seen in Table 6.1.

The current field vector components \( u_x \) and \( u_y \) are treated as the two separate independent random processes. This doubles the size of the random space dimension given in Table 6.1. The number of runs from the Smolyak grid is presented in Table 6.2 for the three different correlation lengths and three different levels.

The perturbation scale \( \delta \) in Karhunen-Loeve expansion (6.3) is chosen to have
**Table 6.1:** The required number of random space dimension for the various correlation lengths 30, 25, and 20.

<table>
<thead>
<tr>
<th>corr. len. ($l_c$)</th>
<th>30</th>
<th>25</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>num. of dimension</td>
<td>4</td>
<td>5</td>
<td>7</td>
</tr>
</tbody>
</table>

**Table 6.2:** The required runs from the Smolyak grids for the random process current field. The runs 8801 and 30801 (red circled) did not take place in any cases and they are given here for completeness.

<table>
<thead>
<tr>
<th>corr. len. ($l_c$)</th>
<th>Level 2</th>
<th>Level 3</th>
<th>Level 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>$l_c = 30$ (ndim=4 × 2)</td>
<td>145</td>
<td>849</td>
<td>3937</td>
</tr>
<tr>
<td>$l_c = 25$ (ndim=5 × 2)</td>
<td>221</td>
<td>1581</td>
<td>8801</td>
</tr>
<tr>
<td>$l_c = 25$ (ndim=7 × 2)</td>
<td>421</td>
<td>4117</td>
<td>30801</td>
</tr>
</tbody>
</table>

perturbation in a magnitude of 25% percent to the experimental realization $\bar{u}(x, y)$. To this end, we took the $\delta$ values as 0.0225 for the velocity $x$-component ($u_x$) and 0.0175 for the velocity $y$-component ($u_y$). The three different correlation lengths are run for the level 2 and the level 3 accuracy in this section. We do not present these results here. We observed that the mean and the standard deviation values along the stations have quite close values for level 2 and level 3. We chose level 3 in the Smolyak grid for all the runs in this section.

### 6.2.1 Results and discussion

We have used a relatively large values of correlation length to keep the random space dimensions low enough so that we can realize the deterministic runs in the computer resources available to us. We first study the influence of the correlation length effects on the standard deviation deviations of the three wave parameters in Figures 6.11, 6.12, and 6.13. The values of standard deviation of significant wave (see Figure 6.11),
mean wave period (see Figure 6.12), and mean wave direction (see Figure 6.13) do not change much for the three correlation effects. However, there is an exception for plotting ray D as seen in Figure 6.12.

The normalized standard deviation (dividing standard deviation by the mean) for the significant wave height and the mean wave direction take the values mostly below 2%. This indicates that introducing a perturbation of magnitude 25%, the current random inputs induced very low uncertainty for these two quantities. This is a rather surprising result since the current has important effects on the shoaling and the refraction of the waves.

Figure 6.11: The comparison of relative percentages of standard deviation ($\sigma$) and mean ratio of significant wave height ($H_s$) for various correlation lengths (cor. len. = 30, 25, 20) used in the random process of the current field. All runs used level 3 from the Smolyak grid.

We will next show the error plots for the three statistical wave parameters ($H_s$, $T_{m01}$, $\bar{\theta}$) in Figures 6.14, 6.15, and 6.16. All the error bar plots have confi-
Figure 6.12: The comparison of relative percentages of standard deviation ($\sigma$) and mean ratio of mean wave period ($T_{m01}$) for various correlation lengths (cor. len. = 30, 25, 20) used in the random process of the current field. All runs used level 3 from the Smolyak grid.

dence intervals within 95% ($\mu \pm 2\sigma$). The significant wave height results in Figure 6.14 show that the confidence intervals are hardly noticeable for the stations on rays B, C, and D. The very small error bars (green) may be superimposed by the mean (red dots) in these plots. The stations on ray A have shown larger confidence intervals and we think that improper boundary conditions have amplified this result. As mentioned earlier, the physical boundaries have been extended along the y axis, and we assumed the current field to be zero in this extended region. We introduced randomness only in the physical region in the stochastic simulation. Therefore, this ad-hoc assumption might cause a relatively larger uncertainty on the stations of Ray A.

The random current process has produced very small error bars (see Figure 6.15)
Figure 6.13: The comparison of relative percentages of standard deviation ($\sigma$) and mean ratio of mean wave direction ($\bar{\theta}$) for various correlation lengths (cor. len. = 30, 25, 20) used in the random process of the current field. All runs used level 3 from the Smolyak grid.

on all the stations for the mean wave period. Ris’ work [130] had run the same HISWA simulation with the SWAN code, and with the current deactivated. They observed that the effects of current on the significant wave height and mean wave period are much less pronounced than their expectation. We confirm this surprising result in our stochastic simulation. The perturbation of the current field in a magnitude of 25% percent slightly changed these two statistical wave parameters ($H_s$, $T_{m01}$). On the other hand, the mean wave direction has shown a larger confidence interval for almost all stations on rays A-D. Ris [130] also stated the importance of the current field to get the correct mean wave direction. Ris [130] experimented with turning off the current field in their numerical simulation and observed that the results of mean wave direction were largely off from the experimental results. The mean wave direction seems sensitive more to the stochastic perturbation of the
current than to the significant wave height and mean wave period.

**Figure 6.14:** Significant wave height $H_s$ of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the third order expansion for the current parameters (1-5 modes, cor. len. = 25).

The energy spectra with the error bars from the stochastic simulation are plotted for the stations of ray B (5, 13, 16, 24) and of ray D (3, 15, 19, 26). The plots are arranged such that the left-column of Figure 6.17 accommodates the stations of Ray B and the right-column of the figure is reserved for the Ray D stations. The left-column plots in Figure 6.17 showing the stations on the ray B have significantly larger error bars around the primary peaks. The right column plots (the stations on ray D) have showed smaller error bars around the primary peaks than those of the left-column plots. The stochastic simulation results suggest that, by introducing the current random process, the confidence interval is larger around ray B (the larger uncertainty), which happens to be a well rounded body over the cylinder bar. These results confirmed our intuition that the most uncertain region is around the stations
Figure 6.15: Mean Wave Period $T_{m01}$ of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the third order expansion for the current parameters (1-5 modes, cor. len. = 25).

of ray B since there is the complex bathymetry and current field.

6.3 Uncertainty of source and current field

We presented stochastic simulation with the random inputs from the source and the current field in the previous two sections. We combine both these stochastic inputs here. The four random variables ($\alpha_{BJ}$, $\gamma$, $C_{br}$, and $\alpha_{EB}$) with the uniform probability distribution within the range has been defined already in section 6.1. The current field is perturbed around $\pm 25\%$ of the current velocity of the experimental realization. The correlation length of 25 was also chosen as in the previous section 6.2. The dimension of random space for capturing 90% of the energy for the correlation
Figure 6.16: Mean Wave Direction $\bar{\theta}$ of simulation results (confidence interval within 95\% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the third order expansion for the current parameters (1-5 modes, cor. len. = 25).

length of 25 (see Table 6.1) will be five. The $x - y$ component velocities are taken as the two independent random processes and therefore the dimension of the random space yield is 10 for the current random process. This together with random variables of the source term parametrization raise the dimension of random space to 14. The third order gPC expansion is used here. As seen in the previous sections, the results from level 3 are very close those of level 4. Level 4 from the Smolyak grid requires a large number of deterministic runs (30801, see Table 6.2) for 14 random variables. We sacrificed the small accuracy in the convergence to reduce the number of deterministic runs in this section. The number of deterministic runs for level 3 and the fourteen variables is 4117 from the Smolyak grid (see Table 6.2).
(a) (left column) stats. on ray B  
(b) (right column) stats. on ray D

Figure 6.17: Energy spectra of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the third order expansion for the current parameters (1-5 modes, cor. len. = 25).

6.3.1 Results and discussion

We presented the error bar plots for the significant wave height, the mean wave period, and the mean wave direction, respectively, in Figures 6.18, 6.19, and 6.20.
The significant wave height results given in Figure 6.18 are slightly different from those of the stations on ray A in Figure 6.7 (source term uncertainty). This clearly shows that the additional current randomness influenced those stations. However, the results of other stations on Ray B, C, and D have had been little influence from the current random input. These are apparently the same as those of the stations in Section 6.1.

The mean wave period results are shown in Figure 6.19. The change of the mean wave period is negligible compared to that of the mean wave period in Figure 6.8. The source random input is clearly dominant for the magnitude of the confidence intervals for the mean wave period.

The confidence intervals of the mean wave direction in Figure 6.20 have noticeable differences from those of the mean wave direction shown in Figures 6.9 and 6.16. The confidence intervals of mean wave direction were already shown in Figure 6.9 for the source random input and in Figure 6.16 for the random current field input. They individually contributed to the confidence intervals of the mean wave direction. We rather expected that the combination of both uncertainty inputs will influence much more than just the summation of the individual effect for the mean wave direction. But we observed from the results of Figure 6.20 that the combination of both random inputs have mildly increased the confidence intervals over the results of the source term random inputs.

We finally show the error bar plots of the energy spectra (see Figure 6.21) for the selected stations (5, 13, 16, and 24) on ray B and (3, 15, 19, and 26) on ray D. We arranged plots in Figure 6.21 such that the left column accommodated plots of the stations of ray B and the right column plots of stations of ray D. The ray D stations did not show much different results in energy spectra from those of the source
term random input already given in Figure 6.10. The source random inputs clearly dominate in the confidence intervals of energy spectra on the stations along ray D. On the other hand, the confidence intervals of the energy spectra on the stations of ray B (left column of Figure 6.21) have a slightly larger variation (comparing the results for the source random inputs in Figure 6.10) along the frequency axis. These stations are exposed to the strong current gradient (see the current field in Figure 5.22). The ray B stations that show the larger variations can be explained by the component of random current input. We finally notice that the energy around the peak has significantly been altered by adding a random current field for station 5.

**Figure 6.18:** Significant wave height $H_s$ of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the third order expansion for the current parameters (1-5 modes, cor. len. = 25) and the source parameters ($C_{f\alpha}, \alpha_{BJ}, \gamma, \alpha_{BJ}$).
Figure 6.19: Mean Wave Period $T_{m01}$ of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the third order expansion for the current parameters (1-5 modes, cor. len. = 25) and the source parameters ($C_{bf}, \alpha_{BJ}, \gamma, \alpha_{BJ}$).
Figure 6.20: Mean Wave Direction $\bar{\theta}$ of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the third order expansion for the current parameters (1-5 modes, cor. len. = 25) and the source parameters ($C_{bfr}$, $\alpha_{BJ}$, $\gamma$, $\alpha_{BJ}$).
Figure 6.21: Energy spectra of simulation results (confidence interval within 95% ($\mu \pm 2\sigma$)) and experimental data are compared. The sparse grid used the fourth order expansion for the current parameters (1-5 modes, cor. len. = 25).
6.4 Summary and Conclusions

We have performed sensitivity analysis of the source term parameters used in this simulation. We looked at three different outputs: (1) significant wave height, (2) mean wave period, (3), and mean wave direction. The depth-breaking parameters $\gamma$ and $\alpha_{BJ}$, bottom friction coefficient $C_{bfr}$, and triad parameters $\alpha_{EB}$ all seems important either for some region of the domain and/or for one of the three wave outputs. The sensitivity results suggested to keep the parametrization of the three source terms as random inputs for the uncertainty quantification.

We have conducted stochastic simulation to quantify the uncertainty in the HISWA simulation. We considered only randomness in source term parametrization and current field in this study. The source term uncertainty and random current field were individually treated first and then combined. We used the second, the third, and the fourth order expansion of gPC for stochastic simulation with random inputs of source term parametrization. We compared the mean and standard deviations of the three wave parameters in Figures 6.4, 6.5, and and 6.6. The results suggested that the third-order expansion was reasonably converged. We have shown the plots of confidence intervals within 95% in Figures 6.7, 6.8, and 6.9. We see that the confidence intervals within 95% captured well the experimental results for the significant wave height and mean wave period. As for the mean wave direction, the stations after the wave breaker have shown larger confidence interval variations. But the confidence intervals within 95% did not contain the experimental results at ray C or ray D. The rightmost measurement on ray C (see Figure 6.9) can be covered by the confidence intervals 99% $(\mu \pm 3\sigma)$. However, we see that the confidence intervals of the rightmost station on ray D are still far from the experimental results (see the Figure 6.9). We remark that source term parametrization has little or no influence
on the three wave parameters on the stations of ray A or on the stations before the wave breaker \((x < 12)\).

The confidence intervals within 95% of the energy spectra are shown in Figure 6.10. The intervals are significantly larger around the primary and secondary peaks for all stations. The energy spectra from the experiment are also well contained by the confidence intervals from the stochastic simulation for the source random input.

The current field has an important influence on depth-induced and current-induced wave refraction and shoaling. We introduced random current field by using the Karhunen-Loeve expansion of the exponential Gaussian correlation function. The correlation length \((l_c)\) determines the dimension of random space. We have tested three different correlation lengths: 30, 25, and 20. The results (see Figures 6.11, 6.12, and 6.13) show that the standard deviation of the three important wave parameters \((H_s, T_{m01}, \theta)\) did not make a significant difference for level 3 expansion from the Smolyak grid. We chose the correlation length as 25 and used the level 3 from the Smolyak grid for stochastic simulation of the random current input. The magnitude of perturbation of about 25% was introduced to \(x\) and \(y\) velocity components of the experimental current field. Although it is a relatively large perturbation of the current field, the confidence intervals (see Figures 6.14 and 6.15) did not induce any noticeable differences in the significant wave height or the mean wave period. This is a rather unexpected result also observed by Ris [130]. The current random input, however, caused a relatively large variation in the confidence intervals for the mean wave direction (see Figure 6.16). The variation extended even on the stations in front of the wave breaker. Although there were some variations in the mean wave direction from the current random field input, the confidence intervals within 95% still did not contain the experimental results.
The error bars of energy spectra were slightly changed around just the primary peaks at the stations (see the Figure 6.17) where strong current gradients were present. For this perturbation level, the uncertainty of source term parametrization still has a much larger effect than the random current field input on the wave parameters and energy spectra.

We finally combined two random sources to perform stochastic simulation. We again used level 3 from the sparse grid of Smolyak. The error plots are very close to the ones from the uncertainty of source parametrization. The current field caused slightly larger error bars on the mean wave direction (see Figure 6.20) than those in Figure 6.9. The current random input also modified the confidence intervals on the energy spectra especially for station 5. The simulation using the source term parametrization as the random variable did not change at all for this station. We expected an interesting pattern of confidence intervals from the stochastic simulation using both the source and current random inputs due to the complex interaction of current and source terms. The combination of both random inputs did not give a surprising result in that the two individual uncertainty sources seemed only to be superimposed on each other. The patterns in the combination of both uncertainties are very similar to those from the uncertainty of source terms alone.
Chapter Seven

Summary and Future Works
The first part of this thesis showed that, using POD formulation, numerical ocean output is low-dimensional. We applied the gappy POD approach to reconstruct all ocean states from a few measurement locations. The methodology relied on the assumption that computed lower POD modes, which are in slow-manifold, can be used in the adjacent times. We demonstrated numerical evidence from real numerical ocean output that lower POD modes change slightly as we added new snapshots. We have a proposition based on the findings of this thesis so that we should create a POD database available to the public. The database should be updated as the new simulations generate the snapshots. Therefore, our methodology will generate more accurate results as the convergence of lower modes is achieved.

The second main contribution of this thesis is a new scheme for spectral ocean wave equation. The new scheme has shown that it can be efficient. As mentioned earlier, the wave simulation is currently bottlenecked in the coupling of ocean and wave codes. The new scheme is a first attempt in higher-order discretization in the numerical ocean wave community. We next have several recommendations along with this study to improve the new scheme and to fix the drawbacks that we have seen with this study and to make it more efficient than other operational models.

Although we have good success in a range of problems, we have a limitation too. We tried to validate our implementation against the wave-blocking test case [130, 98], and initially smooth runs eventually blew up due to instability. The reason for this instability comes from the very instant change in energy-density profile (also observed in Lai at al.’s work [98]) in frequency direction. This sharp profile generates Gibbs-like oscillations in spectral space if not resolved and it eventually renders the simulation unstable. Employing such a fine grid in frequency direction (now CFL restricted by Fourier-collocation discretization) is extremely costly. This will be remedied if we use local refinement (as we did along directional axis) but with
adaptivity. Note that if these oscillations are a result of discontinuities, which we do not have here, filtering would be the only effective way to stabilize the scheme [78] (see appendix A). These oscillations are indeed a friendly warning of poor resolution. We currently have fixed collocation points (not local refinement and adaptivity) in frequency directions. Future work should include adaptivity in the frequency direction.

The coastal application has been reported [23] to require very small time steps due to CFL conditions which are more restricted by spectral space discretization than physical space discretization. The stability of Fourier-collocation is indirectly (eigenvalues of the differential operator appear directly in stability analysis) related to minimal spacings ($\Delta \sigma_{\text{min}}$ or $\Delta \theta_{\text{min}}$). Resolving a sharp profile in a frequency direction (after local refinement) would yield very dense distribution of collocation points around the clustering region, which will strictly limit the scheme to very small time steps. We recommend that implicit treatment of spectral space discretizations or a semi-Lagrangian scheme should be investigated.

All applications here have very simple boundaries. Coastal applications will need very complicated boundaries to be gridded. This implementation currently supports only fixed polynomial order elements. So as to exploit high-order efficiency, we can use as large an element but with very high-order polynomials. This is possible for cases with simple boundaries, as already seen in case VII (simulated with only two very high-order elements). The implementation with a fixed polynomial order can still use very large elements on a very complex boundary by using curved boundaries such that the boundary element is represented as accurately as possible. Future implementation should handle high-order curvature on coastal boundaries. Another way to resolve complex boundaries is to use unstructured grid technology for generating very fine mesh on coastal lines, which eventually forces us to consider an
implicit scheme in physical space as well as $p$-adaptivity. The realistic applications will have triangles with a large disparity in sizes. High-order polynomials will be expanded over large elements to resolve the simulation. Without $p$-adaptivity, the fixed high-order polynomials may over-resolve the solution (increasing computational cost) on the small elements.

More advanced flux terms which enhance accuracy [154, 126] will be incorporated in a future study.

The third main contribution of this thesis is that uncertainty quantification through collocation method is introduced to the ocean wave application. We employed stochastic simulation using the Smolyak grid rather than estimating better values of tune-up source term parameters. This way certainly increases the computational cost to several orders of magnitude higher. However, it still requires much fewer deterministic runs than Monte Carlo does. The source term tune-up parameters are taken as random variables. If we only consider third generation models, we can take the following parameters as random variables: (1) density of air ($\rho_a$), density of water ($\rho_w$), drag coefficient ($C_d$) in source generation by wind; (2) the parameters of whitecapping ($C_{wc}$, $n$, $p$, $\tilde{s}_{PM}$); (3) quadruplet parameters ($C_{nl4}$, $\lambda$); (4) $\alpha_{EB}$ parameter in triad term; (5) the breaking wave parameter ($\gamma$); and (6) the bottom friction coefficient ($C_{bfr}$). This makes the random variables total up to 12 parameters. The total of 12 random variables results in several thousand deterministic runs for the second or fourth order of gPC expansion, which is usually adequate for the convergence, from the Smolyak grid. These runs are independent of each other (embarrassingly parallel) and can be simulated on the current average HPC clusters. Knowledge of important wave parameters with confidence intervals rather than single values is certainly preferable for making better engineering decisions.
We have also treated the current field as a random process. Although the effects of uncertainty in the current field are suppressed by those of the source term in HISWA simulation, there are many situations where this might not be true. We proposed a Gaussian correlation with a specified correlation length. Then using KL expansion, we perturbed the current field. The value of the correlation length determines the size of the random space dimension. The random space dimensionality can easily reach up to several hundreds for a single random process. We can not currently handle several hundred random dimensions. Moreover, the challenges are even greater for a real ocean application since there are many other random processes. As an example, bathymetry, boundary and initial conditions, wind direction ($\theta_w$), wind velocity at 10 m ($U_{10}$), depth-inducing parameter ($\gamma$), etc. can all be taken as random processes. Future works should also aim at this grandiose challenge.
Appendix A

Spectral Methods for Discontinuous and Sharp-Gradient Problems
A.1 Introduction

Spectral methods achieve spectral accuracy for smooth solutions. The problems with a discontinuity (or a sharp viscous shock layer) cripple the spectral methods in two ways: undermining stability and losing spectral accuracy. Recovery of spectral accuracy away from discontinuities and stability may have been provided by filtering [63] [62].

Our aim is largely to show the viscous-boundary layer interaction for the Chebyshev collocation methods with filtering and Tal-Ezer mapping. Therefore, our first case is the one-dimensional viscous Burger equation with discontinuous initial conditions under a specific boundary condition such that it forms a viscous shock layer at the beginning and then impinges on the boundary layer in the steady-state. Based on available analytical solutions of the steady-state case, we will investigate filtering for spectral accuracy away from the viscous shock layer. Later, we will look at the two-dimensional inviscid and viscous Burger equations that develop a shock which hits on the wall boundaries. We will comment on the effects of the viscosity term and of the order of filter on stability. Lastly, we apply the Chebyshev collocation method (filtered and Tal-Ezer mapped) to the shock tube problem in order to see the fast removal of Gibbs-like oscillations away from the expansion, contact, and shock discontinuities.

The outline of this appendix is as follows. The first section introduces the Chebyshev collocation method, with filtering and Tal-Ezer mapping. Numerical results follow in the next section, and we summarize in the last section.
A.2 Chebyshev Collocation Methods

Collocation methods require that the residuals vanish on the grid points. To illustrate the method, let us assume that we want to solve the following model equation

\[ \frac{\partial u}{\partial t} = \frac{\partial f(u)}{\partial x} + \frac{\partial^2 u}{\partial^2 x} \] (A.1)

with the initial condition \( u(x, 0) = s(x) \), and boundary conditions \( u(-1, t) = h(t) \) and \( u(1, t) = g(t) \).

We seek a solution

\[ u_N(x, t) = \sum_{j=0}^{N} u_N(x_j, t) l_j(x), \] (A.2)

with the flux defined as

\[ f_N(u_N(x, t)) = \sum_{j=0}^{N} f(u_N(x_j, t)) l_j(x) \] (A.3)

where, for easy implementation of boundary conditions, the collocation points, \( x_j \), are chosen as Gauss-Lobatto quadrature points and defined as

\[ x_j = -\cos(\frac{\pi j}{N}) \quad \forall \in [0, \ldots, N]. \]
The Lagrange polynomials $l_j(x)$ may always be expressed in terms of the ultraspherical polynomials, $P_n^{(a)}(x)$. The Lagrange polynomials ($\alpha = 0.5$ for the Chebyshev polynomials here) are

$$l_j(x) = w_j \sum_{n=0}^{N} \frac{P_n^{(a)}(x)P_n^{(a)}(x_j)}{\tilde{\gamma}_n}, \quad (A.4)$$

with the Chebyshev quadrature weights

$$w_j = \frac{\pi}{c_j N}, \quad c_j = \begin{cases} 2 & j = 0, N, \\ 1 & j \in [1, \ldots, N-1] \end{cases},$$

and the coefficient $\tilde{\gamma}_n$ for the Chebyshev Gauss-Lobatto quadratures

$$\tilde{\gamma}_n = \begin{cases} \pi & n \in [1, \ldots, N-1] \\ \frac{\pi}{2} & n = 0, N. \end{cases}$$

The collocation methods require the residual

$$R_N(x,t) = \frac{\partial u_N}{\partial t} - \frac{\partial f_N}{\partial x} - \frac{\partial^2 u_N}{\partial x^2} \quad (A.5)$$

to vanish at the collocation points which here are the Chebyshev Gauss-Lobatto points. Defining the interpolation operator $I_N$, the interpolation of the residual reads
The system is closed with boundary conditions

\[ u_N(x_0, t) = h(t) \quad u_N(x_N, t) = g(t), \]

and initial condition

\[ u_N(x_j, 0) = s(x_j). \]

The spatial derivatives may be computed by using differentiation matrices. In the following subsection, we will derive the differentiation matrices and apply a filter operation for derivatives of arbitrary order.

### A.2.1 Filtering

Nonlinear hyperbolic equations may develop a discontinuous solution for smooth initial conditions. A discontinuity in a domain poses a serious challenge for spectral methods due to the global nature of these methods. Since the methods fail to achieve spectral convergence and only obtain first-order accuracy, one may question the legitimacy of spectral methods in problems which have such discontinuities. It would seem that spectral methods are inherently impotent for discontinuous prob-
lems. However, one way to recover spectral accuracy away from a discontinuity is to employ a filtering technique [63]. The loss of spectral accuracy in the discontinuity problem is one aspect of the whole problem and which can be overcome by filtering; however, there is another problem pertinent to nonlinear hyperbolic systems, that is, the stability issue. Approximation of a non-smooth function with orthogonal functions generates Gibbs-like oscillations. These oscillations interact with the nonlinear term causing instability in time-dependent problems. Fortunately, the filtering approximations are also useful for stabilizing a scheme. Filtering in spectral methods is shown to be tantamount to adding dissipation into the system [62].

The accuracy of filtered approximations depends on the filter order and the smoothness of the solution away from the discontinuity. In a time-dependent problem including a moving discontinuity, we want to maintain both high-accuracy and stability, which may cause a conflict of interest. The higher accuracy stipulates use of a higher order filter; on the other hand, the high-order filter might shatter stability, which usually has a final jurisdiction about a scheme. Although the use of a lower order filter stabilizes the scheme, the discontinuity might be substantially smeared out, which degrades the spectral accuracy of the scheme away from the discontinuity. It is obvious that the best choice is applying the highest filter order permitted by stability. Here, we choose an exponential filter and define it as

\[ \sigma(\eta) = e^{-\alpha (\frac{\eta}{\pi})^p}, \] (A.7)

where \( \alpha \) is a coefficient of influence and \( p \) is the order of the filter. However, the above exponential filter does not strictly satisfy the filtering properties (\( \sigma(\eta) = 0, |\eta| \geq 1 \)) [157]. The influence factor \( \alpha \) can be chosen to make the filter machine zero \( \epsilon_m \) at
\[ \eta = 1 \ (\sigma(1) = e^{-\alpha} = \epsilon_m). \]  
This study chooses the influence coefficient \( \alpha \) as 32.2319.

Using the Lagrangian interpolation, we can approximate \( u(x) \) as

\[ u(x) = \sum_{j=0}^{N} u(x_j) l_j(x) \]  
(A.8)

where the \( x_j \) are interpolation points, and the \( l_j \) are the Lagrangian polynomials. The Lagrangian polynomials can be expressed in terms of the ultraspherical polynomials;

\[ l_j(x) = w_j \sum_{n=0}^{N} \frac{P_n^\alpha(x) P_n^\alpha(x_j)}{\gamma_n}, \]  
(A.9)

the filtered approximation \( u^\sigma(x) \)

\[ u^\sigma(x) = \sum_{j=0}^{N} u(x_j) l^\sigma_j(x), \]  
(A.10)

and the filtered Lagrangian polynomials, \( l^\sigma_j(x) \), are defined as,

\[ l^\sigma_j(x) = w_j \sum_{n=0}^{N} \sigma \left( \frac{n}{N} \right) \frac{P_n^\sigma(x) P_n^\sigma(x_j)}{\gamma_n}, \]  
(A.11)

the filtered differentiation matrices, \( D^{(q),\sigma} \), can be readily obtained now by differentiating the filtered Lagrangian polynomials \( q \) times.
Here, we have used the Chebyshev polynomial \( \alpha = -0.5 \), which is a special ultraspherical polynomial. A derivative of arbitrary order of \( u_N \) is

\[
\frac{d^q}{dx^q} u_N(x_i) = \sum_{j=0}^{N} D_{ij}^{(q),\sigma} u(x_j) \quad (A.13)
\]

The filter used here is a single variable function of \( \eta \) (a fixed filter). One can choose to filter with a spatially variable order, that is, using a higher order filter away from a discontinuity and a lower order filter near it. The latter has been shown to be superior to fixed filters [27]. Spatially-variable filters require knowledge of the exact location of a discontinuity, so they have to incorporate edge-detection methods. Moreover, we will present unstable cases when the discontinuity hits on the boundary might be recovered if lower order filters are used.

### A.2.2 Coordinate Transformation

The extremum eigenvalues of the Chebyshev differential operator for the \( k \)th derivative scales with \( O(N^{2k}) \), which in turn constrains the time step size as \( \Delta t \approx O\left(\frac{1}{N^{2k}}\right) \) in explicit time integration schemes. The Chebyshev quadratures cluster around both boundaries such that the minimal space scales as \( O\left(\frac{1}{N^2}\right) \). A popular solution will be to transform the Chebyshev points to a new set of points which are more uniformly distributed. Fortunately, this kind of transformation alleviates the severity of the time constraint in spectral methods. However, one should not mix finite differ-
ence grids and spectral collocations. More explicitly, eigenvalues of the differential operators appear directly in stability analysis but not in minimal spacing (a rather indirect way). Nonphysical and severe stability constraints make the Chebyshev (or Legendre) polynomials unattractive candidates for equations containing derivatives of an order higher than one. To reduce this strict constraint, we might need to use a non-polynomial basis. A one-to-one and onto mapping has been proposed for the Chebyshev polynomials [95], called the Tal-Ezer mapping. The Tal-Ezer mapping reduces the growth of the eigenvalues of the differential operators from $O(N^{2k})$ to $O(N^{k})$, and that of the minimal spacing from $O(1/N^{2k})$ to $O(1/N^{k})$.

The Tal-Ezer mapping is defined as

$$x = g(\xi, \alpha) = \frac{\sin^{-1}(\alpha \xi)}{\sin^{-1}\alpha},$$

(A.14)

where $\xi_j = -\cos(\pi j/N), j = 0, ..., N$, and the choice of the coefficient $\alpha$ can be justified using resolution [95] and stability and accuracy [95] [159] considerations. We choose the coefficient $\alpha$

$$\alpha = \text{Sech}\left(\frac{\ln|\epsilon|}{N}\right)$$

(A.15)

to keep the mapping error about machine precision $\epsilon$. The accuracy consideration should be taken as a top priority for higher-order derivatives; otherwise, the computation of the derivatives yields unacceptably inaccurate values [159].

The first derivative of $u^\sigma(x)$ is then
\[
\frac{du^\sigma}{dx} = 1 \frac{du^\sigma}{g' \, d\xi}, \quad (A.16)
\]

where \( g' = \frac{dx}{d\xi} \). If we define the diagonal matrix \( M \) as

\[
M_{jj} = \frac{1}{g'(\xi_j, \alpha)}, \quad (A.17)
\]

then, the first derivatives of \( u^\sigma(x) \) are written in the following form

\[
\frac{du^\sigma}{dx} = MD^{(1),\sigma}u. \quad (A.18)
\]

After applying one more differentiation,

\[
\begin{align*}
\frac{d}{dx} \left( \frac{du^\sigma}{dx} \right) &= \frac{1}{g'} \frac{d}{d\xi} \left( \frac{1}{g'} \frac{du^\sigma}{d\xi} \right) \\
\frac{d^2 u^\sigma}{dx^2} &= \frac{1}{g'^2} \frac{d^2 u^\sigma}{d\xi^2} - \frac{g''}{g'^3} \frac{du^\sigma}{d\xi} \quad (A.19)
\end{align*}
\]

we can compute the second-order derivatives as

\[
\frac{d^2 u^\sigma}{dx^2} = \left( S \, D^{(2),\sigma} - R \, D^{(1),\sigma} \right) u, \quad (A.20)
\]

where the diagonal matrices \( R \) and \( S \) are given by
\[ R_{jj} = \frac{g''(\xi_j, \alpha)}{g'^3(\xi_j, \alpha)} \quad S_{jj} = \frac{1}{g'^2(\xi_j, \alpha)}. \]  
(A.21)

If we define the matrix \( D^{2,\sigma} \) as

\[ D^{2,\sigma} = S D^{(2),\sigma} - R D^{(1),\sigma}, \]  
(A.22)

the second-order derivatives can be rewritten as

\[ \frac{d^2 u^\sigma}{dx^2} = D^{2,\sigma} u. \]  
(A.23)

### A.2.3 Time Integration

The time integration scheme (a kind of Runga-Kutta scheme) is adopted from Total Variation Diminishing (TVD) schemes [139]

\[ u^{(1)} = u^n + \Delta t \mathcal{L}(u^n) \]  
(A.24a)

\[ u^{(2)} = \frac{3}{4} u^n + \frac{1}{4} u^{(1)} + \frac{1}{4} \Delta \mathcal{L}(u^{(1)}) \]  
(A.24b)

\[ u^{n+1} = \frac{1}{3} u^n + \frac{2}{3} u^{(2)} + \frac{2}{3} \Delta \mathcal{L}(u^{(2)}) \]  
(A.24c)

with a CFL condition \( \leq 1 \).
The operators $\mathcal{L}$, containing the spatial derivatives, are computed with the filtered differentiation operators. Our filtering strategy is to compute the operators $\mathcal{L}(\cdot)$ first and then filter the corresponding data of that operator. The filtering of data first and then applying the filtered differentiation operators might be more excessive than the projected one. Specifically, in the first stage, we compute the operator $\mathcal{L}(u^n)$ (with the filtered differentiation operators) and then filter the data $u^n$ and compute $u^{(1)}$. For the intermediate stage, we compute $\mathcal{L}(u^{(1)})$ first (again with only the filtered differentiation operators) and then filter the $u^{(1)}$ to get $u^{(2)}$. We repeat the same steps in the final stage.

### A.3 Numerical results

#### A.3.1 The one-dimensional viscous Burger equation

In many physical phenomena, we observe advection-diffusion processes coexisting together. The Burger equation is the simplest model to simulate nonlinear advective and diffusive behavior simultaneously, and it is well suited to numerical experiments for these problems.

The one-dimensional viscous Burger equation can be written as

$$
\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = \mu \frac{\partial^2 u}{\partial x^2} \tag{A.25}
$$

where $\mu$ is the coefficient of viscosity.

Here, we are interested in the steady-state solution of the Burger equation. It has
been shown that its solution converges to a unique steady state in time [96]. The steady-state equation is obtained by dropping the time derivative, and the analytical solution to the nonlinear ODE yields

\[ u(x) = -\sqrt{2C_1} \Tanh\left[\frac{\sqrt{C_1}(x - 2\mu C_2)}{\sqrt{2\mu}}\right], \]  

(A.26)

where the coefficients \( C_1 \) and \( C_2 \) need to be computed numerically for the given boundary conditions and the viscosity coefficient \( \mu \).

In our numerical test cases, we initialized the system with a discontinuity such that

\[
\begin{cases}
  u(x,0) : & \begin{cases} 
  2 & x \leq 0 \\
  1 & x > 0.
  \end{cases}
\end{cases}
\]

Note that the viscous Burger equation does not allow a discontinuity in its solution due to its parabolic characteristic of the equation. The boundary conditions are fixed to be consistent with the initial conditions such that

\[
\begin{cases}
  u(x,t) : & \begin{cases} 
  2 & x = -1 \\
  1 & x = +1,
  \end{cases}
\end{cases}
\]

where the solution forms a viscous shock layer, and which is advected to the right boundary and reaches a steady-state solution depending on the viscous coefficient \( \mu \) for these specific boundary conditions. A steady-state solution depends greatly on
boundary conditions. It is worth mentioning the choice of anti-symmetric boundary conditions \(( u(-1) = -\alpha, u(+1) = \alpha )\) with a discontinuous initial condition at arbitrary locations. In this case, the speed of the shock layer formed \((\approx O(e^{-1/\mu}))\) can be very slow, even for moderately low values of viscosity \(\mu\). This is also confirmed in our numerical experiments (not shown here). It might be numerically very challenging to get steady-state solutions under anti-symmetric boundary conditions and low viscosity \(\mu\) values [129].

Figure A.1 shows typical time-dependent behavior for a solution of the Burger equation for the above initial and boundary conditions, here specifically for the viscosity \(\mu = 0.01\). The shock layer forms instantly and moves to the right boundary; its thickness depends on the viscosity. The Chebyshev collocation method with exponential filtering and the Tal-Ezer mapping are used to capture thick \((\mu = 0.1)\) and thin \((\mu = 0.001)\) shock layers of the steady-state solution for the given initial and boundary conditions. A steady-state solution is achieved down to machine precision through a time accurate scheme such that the solution no longer changes in time. The numerical solutions stop changing after five time units \((t = 5)\) in the test cases. Therefore, we can safely assume that steady-state is achieved at \(t = 10\). Figure A.2 compares numerical steady-state solutions for various collocation resolutions \((N = 16, 32, 64, 128)\) to the analytic solution for viscosity value \(0.1\) \((\mu = 0.1)\). All resolutions capture the shock profile, and there are no visually detectable differences among them. However, it is worth checking the point-wise error plot given in Figure A.3, which confirms the spectral accuracy in the domain, including near the shock layer. For this somewhat high viscosity value, the solution forms a thick shock layer and the usage of a filter might be excessive. Nonetheless, we have employed a high-order exponential filter \((p=16)\), introducing very low dissipation to the solution.

We next consider relatively low viscosity \((\mu = 0.001)\) cases for the same initial
and boundary conditions. The equation develops a sharp shock layer immediately after the start, then moves to the right boundary. Contrary to the previous test cases ($\mu = 0.1$), without filtering, one will observe two phenomena: the Gibbs-like oscillatory solution (destroying the spectral accuracy) in the smooth region and the instability in the scheme. As mentioned before, filtering will cure both issues. Figure A.4 presents the effect of filtering away from the discontinuity. The Gibbs-oscillations are now confined to a region near the discontinuity and are completely removed from the smooth region away from the discontinuity. Figure A.4 shows the results for various resolutions. At resolution $N = 512$, there is no visible error; however, in fact there are still oscillations very near the discontinuity. In Figure A.6, a detailed view still shows small oscillations around the shock layer next to the right boundary (only shown $x \in [0.98, 1.0]$). The Point-wise error plot, Figure A.5, confirms that filtering recovers the spectral accuracy away from the sharp shock layer.

**Figure A.1:** Plots of the Chebyshev collocation solutions to the viscous Burger equation at different time stations ($\mu = 0.01, N = 128$).
Figure A.2: Comparison of numerical and analytical steady-state solutions ($\mu = 0.1, t = 10$) with the Tal-Ezer mapping and an exponential filter (16th order) for various resolutions.

A.3.2 The two-dimensional inviscid and viscous Burger Equations

The two-dimensional viscous Burger equation can be written as

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = \mu \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right)$$  \hspace{1cm} (A.27)

Here, we apply our scheme to two test cases: those of spectral convergence and circular shock. All test cases run on a domain ($x \in [-1, 1] \times y \in [-1, 1]$).

A.3.2.1 Spectral convergence test case

The exact solution $u(x, t)$ to Eq. (A.27) is taken to be $e^{-t} \cos((x - y)/\sqrt{2\mu})$. The initial and boundary conditions are generated from this solution for our numerical
implementation. The fluxes in the $x$ and $y$ directions cancel each other out so that the equation does not develop any shock front, but only smooth solutions for this test case. However, filtering is still effective for the case in each run besides the mapping. The overall purpose of this test case is to strengthen the legacy of two dimensional implementation before the next circular shock case, which has no available analytical solution.

Figures A.7 and A.8, respectively, plot pointwise errors for viscosity values $\mu = 0.1$ and $\mu = 0.001$ at time unit $t = 0.1$ along the coordinate $x(y = 0)$ line. We increase the collocation grid resolution and decrease the time step resolution simultaneously in order to keep the CFL number fixed around 0.25 for both cases. According to the plots, the Chebyshev collocation implementation, filtering, and the mapping confirms the spectral accuracy.
A.3.2.2 Circular shock test case

Initially, the circular shock is located in the center of the domain as

\[
u(x, y, 0) : \begin{cases}
1 & x^2 + y^2 \leq 0.25 \\
0 & x^2 + y^2 > 0.25.
\end{cases}
\]

Wall boundary conditions \((u = 0)\) are imposed on all of the boundaries.

We consider the inviscid \((\mu = 0)\) and low viscosity \((\mu = 0.0001)\) cases. Both cases show, qualitatively, very similar results to each other. However, we observe that the inviscid case might be unstable when the developed discontinuous solution hits on the boundary especially for higher-order filters; in contrasts, the viscous solutions are stable for even fairly low viscosity values.

The first circular shock test cases are run for low viscosity \((\mu = 0.0001)\). Figure
Figure A.5: Point-wise errors of the numerical solutions with the Tal-Ezer mapping \((\mu = 0.001, t = 10)\) and an exponential filter (16th order) for various resolutions.

A.9 shows sample contours at time \(t = 1.0\) for two different grid resolutions \((128 \times 128\) and \(256 \times 256)\). Accordingly, the initial discontinuity begins to develop a sharp viscous layer, advecting toward the upper right and eventually hitting the wall. The two-dimensional solution is well characterized by its behavior along the diagonal of the domain. Therefore, the different grid resolutions are run and compared along the \(x = y\) line \(([0, 2\sqrt{2}])\). Figure A.10 shows the four different grid resolution results at \(t = 0.2\). As seen in the figure, the solution quickly converges away from the discontinuities if the grid resolution doubles. The highest resolution \((N = 512)\) does an excellent job of capturing the profile without any significant oscillations. However, we are more concerned about the stability of the scheme after this viscous layer hits on the wall-boundary. Figure A.11 presents numerical evidence that the scheme is stable. The solution is plotted for grid resolution \(256 \times 256\) at times 0.5, 1, 5, 10, and 100. Despite this lengthy simulation time, the shock impinges on the wall-boundary without showing instability. We conclude that the scheme seems stable with very high-order filtering \((p = 16)\) in the case of the viscous Burger equation for fairly low viscosity values.
The inviscid cases are run for the same initial conditions. Here, the wall-boundary condition applies only to the left and bottom boundaries, while the rest are treated as open boundaries. Figure A.12 shows the results for different orders of the exponential filters \((p = 4, 8, 12)\). The important observation is that, for \(\mu = 0\), results using higher-order filters \((p = 12, 16)\) show instability when the shock hits the boundaries. These results suggest using lower-order filters only around the boundaries to stabilize the shock-boundary interaction for the scheme.

\section*{A.3.3 Euler equations test case (Shock tube problem)}

The one-dimensional Euler equations can be written in vector form as

\[
U_t + F(U)_x = 0, \quad \text{(A.28)}
\]

with
Figure A.7: Point-wise error plots of spectral convergence test cases ($\mu = 0.1$, t=0.1) along the coordinate $x(y = 0)$ axis.

Here, $E$ is the total energy

$$E = \rho \left( \frac{1}{2} u^2 + e \right)$$  \hspace{1cm} (A.29)

where the specific internal energy is defined as ($e = \frac{p}{(\gamma - 1) \rho}$). The ratio of specific heats $\gamma$ can be taken to be 1.4 for ideal gases.

The shock tube problem is chosen to test the filtering scheme with the Tal-Ezer mapping for the Euler equations. Initial conditions are for the case given as:
where the diaphragm separating the left ($L$) and right ($R$) states is located at $x = 0$ and $t < 0$ in the domain $x \in [-1, 1]$. The diaphragm instantly bursts at initial time ($t = 0$), and then the system develops a shock, contact discontinuity, and expansion waves inside the tube. The characteristic variable boundary condition is used for both ends of the shock tube.

Figures A.13, A.14, and A.15 show the solution of the shock tube problem for density, velocity, and total enthalpy ($(E + p)/\rho$, respectively, at time $t = 0.25$. We use four different resolutions ($N = 256, 512, 1024, 2048$). Note that all simulations
Figure A.9: Contours of the circular shock test case at time $t = 1.0$, (left) for a coarse $(128 \times 128)$ grid and (right) for a fine $(256 \times 256)$ grid. The Tal-Ezer mapping and an exponential filter ($16^{th}$ order) are used ($\mu = 0.0001$, $t=1$).

shown here have purposefully not been post-processed so we can see the fast decay of the Gibbs-like oscillations. These oscillations are again removed quickly away from the discontinuities as we double the resolution. This spectral method with filtering recovers the spectral accuracy away from the discontinuities, but still does not do a better job than low-order methods for localized errors at the discontinuities. However, the method is still superior to lower-order methods once it captures the profile since it almost guarantees to keep the shock and its speed away from errors of dissipation and dispersion.

A.4 Summary

We have used the Chebyshev spectral methods with exponential filtering and the Tal-Ezer mapping for solving the inviscid $\propto$ viscous Burger and Euler equations. All computations were practical on a desktop computer since the severe time constraint condition is relaxed by the Tal-Ezer mapping. The one-dimensional results of the viscous Burger equation show that filtering effectively recovers the spectral accuracy.
away from the discontinuity and stabilizes the scheme. We also observed that the viscous Burger solution with high-order filtering \((p = 16)\) remained stable when the discontinuity hit the boundary even for relatively low values of viscosity \((\mu = 0.0001)\). On the other hand, after the shock hit the boundary, the inviscid Burger equation yielded unstable solutions for high-order filtering \((p = 12, 16)\). We note that we had no instability issue as long as the discontinuities stayed in the domain, and instability occurred only if the discontinuity waves hit the boundaries. Accordingly, hyperbolic systems containing discontinuity wave-boundary interactions may be stabilized either by a low-order filter (ideally only around boundaries where the interactions occur) or by explicitly adding weak viscous terms to the equation. We see again in the results for the Euler equation that filtering stabilized the scheme and quickly removed the Gibbs-like oscillations away from the expansion, contact, and shock discontinuities.
Figure A.11: Diagonal solutions (bottom-left to upper-right) for the circular shock test case ($\mu = 0.0001$) at different time stations ($t = 0.5, 5, 10, 100$) with the exponential filter of order 16 and the Tal-Ezer mapping.
Figure A.12: Effect of filtering order $p$ on a fine grid ($256 \times 256$) at different times $t = 0.5, 2, 10, 20$ for the inviscid case.
Figure A.13: The shock tube problem with the initial left state $q_L(ρ, u, p) = \{1, 0, 1/\gamma\}$ and the initial right state $q_R(ρ, u, p) = \{1/10, 0, 1/10\gamma\}$. Shown are comparisons of the density for numerical solutions and the exact solution at $t = 0.25$. The Tal-Ezer mapping and exponential filter ($p = 16$) are employed.
Figure A.14: The shock tube problem with the initial left state $q_L(\rho, u, p) = \{1, 0, 1/\gamma\}$ and the initial right state $q_R(\rho, u, p) = \{1/10, 0, 1/10\gamma\}$. Shown are comparisons of the velocity for numerical solutions and the exact solution at $t = 0.25$. The Tal-Ezer mapping and exponential filter ($p = 16$) are employed.
Figure A.15: The shock tube problem with the initial left state $q_L(\rho, u, p) = \{1, 0, 1/\gamma\}$ and the initial right state $q_R(\rho, u, p) = \{1/10, 0, 1/10\gamma\}$. Shown are comparisons of the total enthalpy for numerical solutions and the exact solution at $t = 0.25$. The Tal-Ezer mapping and exponential filter ($p = 16$) are employed.
APPENDIX B

Sea Spectrums
B.1 Pierson-Moskowitz Spectrum

Pierson and Moskowitz [119] obtained a fully developed sea spectrum (deep water) as

\[ E_{PM} = \alpha_{PM} g^2 (2\pi)^{-4} f^{-5} \exp \left[ -\frac{5}{4} \left( \frac{f}{f_{PM}} \right)^{-4} \right], \quad (B.1) \]

where the scaling parameter \( \alpha_{PM} \) is found as 0.0081.

B.2 JONSWAP (Joint North Sea Wave Project) Spectrum

Hasselman et al. [72] conducted a sea experiment to obtain a fully developed sea spectrum. Accordingly, they found the spectrum to have a sharper peak than that of Pierson-Moskowitz. They assumed the Pierson and Moskowitz spectrum with enhancement function \( G(f) \) [82], that is:

\[ G(f) = \gamma \exp \left[ -\frac{1}{2} \left( \frac{f/f_{\text{peak}} - 1}{\mu} \right)^2 \right], \quad (B.2) \]

where \( \gamma \) is a peak-enhancement factor and \( \mu \) is a controlling parameter of spectrum width. The JONSWAP spectrum is proposed as follows
\[ E_{\text{JONSWAP}}(f) = E_{PM}(f). \]  

(B.3)

### B.3 Breugem and Holthuijsen Spectrum

The nondimensional energy \( E^* \) and \( x^* \) are defined as follows

\[ E^* = \frac{g^2 E_{\text{tot}}}{u_*^4}, \quad X^* = \frac{gX}{u_*^4}, \]  

(B.4)

where the friction velocity \( u_* \) is defined in section 3.2.1. The nondimensional energy growth from a young sea to a fully developed sea [173, 32] (unified fit) can be expressed as follows

\[ E^* = \frac{U_4^{10}}{16U_4^2} \tilde{H}_\infty^2 \left[ \tanh \left\{ k_1 \left( X^* \frac{u_*^4}{U_{10}^2} \right)^{m_1} \right\} \right]^{2p}. \]  

(B.5)

where \( \tilde{H}_\infty = 0.24, k_1 = 4.14e - 14, m_1 = 0.79 \) and \( p = .572 \).
APPENDIX C

The Integrated (Mean) Parameters
In the phase-averaged models, important mean statistical parameters such as Significant Wave Height ($H_s$), Mean Absolute Wave Period ($T_{m01}$) and Mean Wave Direction ($\bar{\theta}$) are obtained by integration of weighted energy variance $E(\theta, \sigma)$.

### C.1 Significant Wave Height ($H_s$)

The significant wave height is defined as

$$H_s = 4\sqrt{\frac{1}{\sigma_{\text{max}} - \sigma_{\text{min}}} \int_0^{2\pi} E(\theta, \sigma) d\theta d\sigma}.$$  \hspace{1cm} (C.1)

### C.2 The Mean Absolute Wave Period ($T_{m01}$)

The mean absolute wave period can be computed as

$$T_{m01} = 2\pi \frac{\int_{\sigma_{\text{min}}}^{\sigma_{\text{max}}} \int_0^{2\pi} \sigma^{-1} E(\theta, \sigma) d\theta d\sigma}{\int_{\sigma_{\text{min}}}^{\sigma_{\text{max}}} \int_0^{2\pi} E(\theta, \sigma) d\theta d\sigma}.$$ \hspace{1cm} (C.2)

### C.3 The Mean Direction ($\bar{\theta}$)

The mean wave direction is:

$$\bar{\theta} = \arctan \left[ \frac{\int_{\sigma_{\text{min}}}^{\sigma_{\text{max}}} \int_0^{2\pi} \sin(\theta) E(\theta, \sigma) d\theta d\sigma}{\int_{\sigma_{\text{min}}}^{\sigma_{\text{max}}} \int_0^{2\pi} \cos(\theta) E(\theta, \sigma) d\theta d\sigma} \right].$$ \hspace{1cm} (C.3)
Bibliography

[1] ECMWF. http://www.ecmwf.int/.


[99] URL. http://marine.rutgers.edu/cool/latte/.


