Development of a Phase-resolving Computer Model for Operational Nearshore Wave Assessment Développement d'un modèle de vagues littorales dispersives pour l'évaluation opérationnelle des risques côtiers

Fatima-Zahra Mihami

Development of a Phase-resolving Computer Model for Operational Nearshore Wave Assessment

Thèse

Présentée à
L'UNIVERSITÉ DE PAU ET DES PAYS DE L'ADOUR

Pour l'obtention du grade de
DOCTEUR

par
Fatima-Zahra Mihami

ÉCOLE DOCTORALE :
SCIENCES EXACTES & LEURS APPLICATIONS (ED211)

Spécialité :
GÉNIE CIVIL

Composition du jury :

Michel Benoit	Pr.	EDF R&D LNHE	Rapporteur
Henrik Kalisch	Pr.	University of Bergen	Rapporteur
Douglas S. Luther	Pr.	University of Hawai'i	Examinateur
María Pilar García Navarro	Pr.	University of Zaragoza	Examinatrice
Denys Dutykh	Pr.	Khalifa University	Examinateur
Jean-François Filipot	Dr.	France Energies Marines	Examinateur
Matthias Delpey	Dr.	Rivage Pro Tech (SUEZ)	Invité
Denis Morichon	MC	UPPA	Co-directeur de thèse
Volker Roeber	Chaire	UPPA	Directeur de thèse

Université de Pau et des Pays de l'Adour
E2S-UPPA, chair HPC-Waves, SIAME, Anglet, France

ذِكْراكِ سَتَبْقَى خالِدَةً يَا خَالَتِي الحَبيبَةُ صَباحُ

Acknowledgements

I would like to start by thanking my supervisor, Volker Roeber, for his invaluable guidance, encouragement, and feedback throughout these years. This research work would not have been the same without his broad expertise and insights into Boussinesq-type models. I would also like to thank him for giving me the opportunity to work on such a unique and exciting topic and for supporting me through each step of the process. I am also very grateful to my co-supervisor, Denis Morichon, for all the help and guidance he provided during my Ph.D. I would also like to thank him for all the resources and facilities he made available to me, which were essential to the completion of my research.

I would like to express my sincere thanks to all the jury members for their time and valuable feedback on my work. A special thanks goes to Michel Benoit and Henrik Kalisch for their diligent review of my thesis and their constructive comments, which helped refine my research and improve the quality of my work. I would also like to give a special thanks to Denys Duthyk for his tremendous help and support during my Ph.D. journey. I am especially grateful for his profound expertise in Boussinesq-type equations, which he generously shared with me.

I would like to give a special thanks to Matthias Delpey and Nikola Danglade from Rivages Pro Tech for their help and support with the model validation. I would also like to thank them for all the interesting discussions we had, which have greatly improved the quality of the new model and the thesis. I would like to express my sincere gratitude to the oceanography team at the University of Hawaii, particularly Douglas Luther, Assaf Azouri, Martin Guiles, and Camilla Tognacchini, for generously providing the field data for the North Shore of Oahu as well as the computer hardware to run the tests. Their support, expertise, and collaborative spirit have been crucial to the success of this work.

Furthermore, I would like to extend my gratitude to Pierre-Henri Cocquet and Yann Moguen for their collaborative work on the improved SGN equations. I am truly grateful for their invaluable contributions and the opportunity to work with them on such an interesting topic. A special thanks goes to Maria Bjørnestad, my esteemed colleague and dear friend, for her invaluable assistance in the model validation and our collaborative work on the EU project Alpheus.

I would also like to thank my office mates and fellow Ph.D. students at the SIAME laboratory: Solène, Jonas, Florian, Ange, Ximun, Amir, Mohamed, Jannik, Erwan, Aritz, Kindro, and Alexandre for their support and for making this Ph.D. journey enjoyable. It has been an absolute delight working alongside them and spending time with them. Many thanks to the university's staff, Caroline Hanin and Isabelle Aguillé, for helping with the administrative paperwork. Their hard work and support are greatly appreciated.

Lastly, I would like to express my deepest appreciation and gratitude to my beloved family and friends for their support and belief in me. I am indebted to my parents for their unconditional love, encouragement, and guidance, which have been the driving force behind all my academic achievements. I am truly blessed to have them by my side.

Abstract

Numerical modeling of nearshore waves has been of high interest for the quantitative assessment of ocean-borne hazards and the design of coastal infrastructure. An accurate calculation of waves in the littoral environment requires numerical models to account for phase-dependent processes that can pose challenges not only to the quality but also to the computational complexity of the numerical solution.

This Ph.D. research presents the strategic development of a numerical model for a set of Boussinesq-type equations with the objective of building a tool for practical applications and operational forecasting systems. The effort has led to a computer code where multiple fundamental features were optimized for both computational efficiency and accuracy to achieve fast and reliable solutions of nearshore waves. It is highlighted that the choice of the numerical framework for the underlying Shallow Water Equations is of critical importance for the quality of the full Boussinesq-type solution. Counter-intuitively, the numerical scheme combines shock-capturing capabilities for breaking waves and run-up with low diffusion for the propagation of multi-directional periodic waves. These properties are obtained with an explicit second-order finite-volume scheme over a staggered grid and without the use of the method of characteristics or computationally expensive Riemann solvers. Consequently, the streamlined algorithm caters to numerical efficiency and grid nesting, which are crucial for fast turnaround times over large domains.

The model was verified and validated with standard benchmark tests for problems involving commonly encountered processes such as bore propagation, run-up, shoaling, refraction/diffraction, and energy transfer to infra-gravity waves. A particular quality of the new model is its consistent calculation of the challenging wave transformations in the surf zone dominated by breaking waves. Here, the additional solution of turbulent kinetic energy produces time- and space-varying dissipation to mimic physically realistic surf zone processes.

Finally, the performance of the model was tested with two energetic swell scenarios at the Northshore of Hawai'i and validated with field data for its applicability in handling complex wave transformations over a large irregular bathymetry with thousands of input waves. The optimized numerical structure in combination with efficient parallelization methods on CPU and GPU supports high-performance computing and offers the possibility to carry out intensive computations on commodity hardware.

Keywords: Nearshore waves, phase-resolving models, Boussinesq approach, numerical methods, operational computations, high-performance computing, GPU computation, run-up

Contents

List of Figures 8

List of Tables 9

List of Acronyms 10

List of Symbols 12

1 Introduction 14

1.1 General Context . . . 15
1.2 Nearshore Wave Assessment . . . 16
1.2.1 Mathematical Framework . . . 16
1.2.2 Numerical Methods . . . 18
1.3 Model-based Forecast . . . 19
1.4 Objectives . . . 20
1.5 Outline . . . 21

2 Shallow Water Solver 23

2.1 Introduction . . . 24
2.2 Efficient Numerical Computations of Long-Wave Run-Up and Their Sensitivity to Grid Nesting . . . 24
2.3 Malpasset Dambreak . . . 57
2.4 Conclusions . . . 59

3 Boussinesq-type Equations: Nonlinearity and Dispersion 60

3.1 Introduction . . . 61
3.2 SGN 1D Model . . . 62
3.2.1 Governing Equations . . . 62
3.2.2 Numerical Scheme . . . 63
3.2.3 Verification . . . 66
3.3 Nwogu 1D model . . . 68
3.3.1 Governing Equations . . . 68
3.3.2 Numerical Scheme . . . 69
3.3.3 Verification . . . 71
3.4 Frequency Dispersion . . . 73

3.4.1 Linear Dispersion Relation . . . 73
3.4.2 Numerical Tests . . . 75
3.5 Nonlinearity . . . 79
3.5.1 Numerical Tests . . . 79
3.6 Conclusions . . . 81

4 Sensitivity of Periodic Wave Problems to Numerical Diffusion 82
4.1 Introduction . . . 83
4.2 Hybrid FV/FD Solver . . . 84
4.2.1 Governing Equations . . . 84
4.2.2 Godunov-type Scheme . . . 84
4.2.3 Flux Reconstruction . . . 86
4.3 Numerical Results . . . 87
4.3.1 Monochromatic Wave Propagation . . . 87
4.3.2 Irregular Wave Propagation . . . 90
4.4 Conclusions . . . 93

5 Wave Breaking in Boussinesq-type Models 94
5.1 Introduction . . . 95
5.2 Methodology . . . 96
5.2.1 Identification of Wave Breaking Onset . . . 96
5.2.2 Wave Breaking Closure . . . 97
5.3 Numerical Tests . . . 99
5.3.1 Laboratory Experiment . . . 99
5.3.2 Model Settings . . . 100
5.3.3 Numerical Results . . . 101
5.4 Conclusions . . . 103

6 2D Dispersive Wave Model: Implementation and Validation 105
6.1 Introduction . . . 106
6.2 Numerical Solution . . . 106
6.2.1 Governing Equations . . . 106
6.2.2 Numerical Scheme . . . 109
6.2.3 Wave Generation . . . 114
6.2.4 Absorbing Boundaries . . . 117
6.2.5 Wave Breaking . . . 118
6.3 Implementation Strategies for Computational Efficiency . . . 119
6.3.1 Floating-point Division . . . 121

6.3.2 Tri-diagonal Solver . . . 121
6.3.3 Wavemaker . . . 121
6.4 Numerical Tests . . . 122
6.4.1 Gaussian Drop Test . . . 122
6.4.2 Evolution of an Undular Bore . . . 124
6.4.3 Solitary Wave Runup on a Plane Beach . . . 125
6.4.4 HIreef - Solitary Wave over 2D Reef . . . 128
6.4.5 Breaking Solitary Wave over 3D Reef with a Cone . . . 131
6.4.6 Wave Refraction-Diffraction over an Elliptic Shoal . . . 136
6.5 Conclusions . . . 138

7 GPU-based Implementation 139
7.1 Introduction . . . 140
7.2 GPU Programming Strategies . . . 140
7.2.1 Heterogeneous Computing . . . 141
7.2.2 Memory Hierarchy . . . 142
7.2.3 Data Transfer . . . 142
7.3 GPU Implementation . . . 143
7.3.1 Timestep Reduction Algorithm . . . 144
7.3.2 Tri-diagonal Solver . . . 146
7.3.3 Wavemaker Source Term . . . 147
7.4 Parallelization Performance . . . 148
7.5 Conclusions . . . 150

8 Applicability for Operational Scenarios 151
8.1 Introduction . . . 152
8.2 Study Site and Field data . . . 152
8.2.1 Swell Events . . . 152
8.2.2 Bathymetry and Model Settings . . . 153
8.2.3 Input Spectrum . . . 155
8.3 Numerical Results . . . 157
8.3.1 Accuracy of the Computed Results . . . 157
8.3.2 Grid Sensitivity . . . 160
8.3.3 Model Performance . . . 163
8.4 Conclusions . . . 164

9 Conclusions and Perspectives 165
9.1 Main Results . . . 166

9.2 Perspectives . . . 168
9.2.1 Implementation into a Forecast System . . . 168
9.2.2 Mesh Refinement in Boussinesq Models . . . 168
9.2.3 Wave Breaking Closure . . . 168
9.2.4 Fully Dispersive Model . . . 168
9.2.5 Further Extensions and Improvements . . . 169

References **170**

Appendices **181**

A Governing Equations Derivation **181**
A.1 Base Model Derivation . . . 182
A.2 Serre–Green–Naghdi Equations . . . 188
A.3 Nwogu Equations . . . 188
A.3.1 Weakly-nonlinear equations . . . 190

B Derivation of the SGN Elliptic System **192**

C Modified SGN Equations with Improved Dispersion **196**
C.1 Improved SGN Equations . . . 197
C.2 Optimized Equations . . . 198
C.2.1 Optimized phase celerity for a single kb . . . 199
C.2.2 Group celerity . . . 200
C.3 Preliminary Numerical Results . . . 201
C.3.1 Sine wave on flat bottom . . . 201
C.3.2 Periodic wave propagation over a submerged bar . . . 202

D Numerical Discretization of the 2D SGN equations **204**
D.1 Governing Equations . . . 205
D.2 Elliptic Equation Discretization . . . 206
D.3 Reflective Boundary condition . . . 209

E Empirical Wave Spectra **210**
E.1 Pierson-Moskowitz Spectrum . . . 211
E.2 JONSWAP Spectrum . . . 211
E.3 TMA Spectrum . . . 212
E.4 Illustration . . . 212

List of Figures

2.1 Propagation stages of the Malpasset dambreak after 200, 700, 1300, and 2300 sec. 58

2.2 Influence of Strickler friction coefficient on the wave front arrival time (left). Maximum water elevation compared with field data and results from Hervouet's model (right) . . . 58

3.1 Definition sketch . . . 62

3.2 Collocated and Staggered grid . . . 63

3.3 Elliptic solver verification. . . . 67

3.4 Solitary wave solution at $t = 100\,\mathrm{s}$ for different grid resolutions $\zeta = \frac{d}{\Delta x}$ . . . 67

3.5 Staggered layout for Nwogu's variables . . . 70

3.6 Solitary wave propagation $\left(A/d = 0.1\right)$ at time $t = 145s$ . . . 73

3.7 Error in phase speed between Airy wave theory and linearized SGN equations . . . 75

3.8 Computation of monochromatic wave propagation in a long flat channel for different values of kh. The lines denote the reference and computed free surface at $t = 500\,\mathrm{s}$. The blue line denotes the solution with Nwogu's equations, the red line represents SGN equations, and the black line is the reference solution from Airy wave theory. . . . 76

3.9 Definition sketch of wave transformation over a submerged bar. (a) Laboratory setup from Beji and Battjes [1993]. (b) Numerical model setup. Circles denote gauge locations. . . . 77

3.10 Computation of wave transformation over a submerged bar. The blue line is the free surface at 74.5 s computed with Nwogu's model. The red line is from SGN model . . . 78

3.11 Computation of wave transformation over a submerged bar. Black circles denote laboratory data from Beji and Battjes [1993]. The blue lines are time series from Nwogu's model ($\alpha = -0.39$) and red lines represent the results SGN model . . . 79

3.12 Nonlinear shoaling of a solitary wave: Numerical model setup . . . 80

3.13 Nonlinear shoaling. Comparison between computed wave crest evolution and data from Grilli et al. [1994] . . . 81

4.1 Sine wave propagation of $kh = 0.5$: Comparison between HLLC and Kurganov schemes for a grid size $\lambda/\Delta x = 30$ and fifth order-reconstruction . . . 88

4.2 Sine wave propagation of $kh = 0.5$: Comparison between different orders of flux reconstructions for a grid size $\lambda/\Delta x = 30$ . . . 88

4.3 Sine wave propagation of $kh = 0.5$: Comparison between Present model and HLLC scheme for different grid sizes . . . 90

4.4 Case 1: Godunov-type model with fifth-order flux reconstruction . . . 91

4.5 Case 1: Present model with second-order flux reconstruction 91
4.6 Case 2: Godunov-type model with fifth-order flux reconstruction 92
4.7 Case 2: Present model with second-order flux reconstruction 92
4.8 Case 3: Godunov-type model with fifth-order flux reconstruction 92
4.9 Case 3: Present model with second-order flux reconstruction 92

5.1 Bathymetry layouts with wave gauge positions for the three cases 100
5.2 Comparison between the observed (o) and computed (x) significant wave heights at nine wave gauges for different wave and topography conditions. The blue, green, and red markers denote the solutions with the TKE closure, the hybrid approach, and no wave-breaking treatment, respectively. 102
5.3 Comparisons between the observed (black lines) and computed (colored lines) spectra and time series for different grid sizes and wave-breaking treatments at **Gauge 7** for **Case 2**. The blue, green, and red lines illustrate the solutions with the TKE closure, the hybrid approach, and no wave-breaking treatment, respectively. 103

6.1 Definition sketch: State variables of Nwogu's equations 107
6.2 Definition sketch: Nwogu's equations state variables 109
6.3 The creation of an artificial wave group as a result of coarse frequency binning. . 116
6.4 Gaussian drop test: Free-surface evolution at different time intervals. 123
6.5 Gaussian drop test: Comparison of free-surface evolution along two diagonal transects: Transect $y = x$ (red line) and Transect $y = -x$ (blue line) to verify the symmetry of the model implementation. 124
6.6 Evolution of an Undular Bore: Free surface elevation at gauges. Colored lines denote the numerical results with different grid sizes. The dashed line represents the laboratory data. 125
6.7 Definition sketch of solitary wave runup on a plane beach. 126
6.8 Surface profiles of solitary wave transformation on a 1:19.85 plane beach with $A/h = 0.3$. Solid lines and circles denote computed and measured data. 127
6.9 Solitary wave runup on a plane beach. (a) 1:19.85 (Synolakis [1987]). (b) 1:15 (Li and Raichlen [2002]). (c) 1:5.67 (Hall et al. [1953]). Solid lines and circles denote computed and measured data. 128
6.10 HIreef Benchmark test. Snapshots of free surface profiles for the propagation of solitary wave with $A/h = 0.3$ over 1:12 slope and exposed reef crest. Blue lines and red lines denote the water free surface and the TKE. The TKE vertical axis is zoomed in to highlight the shape of the TKE solution. During the computation, the TKE quantity reaches a maximum value of $33\,\mathrm{m}^2\,\mathrm{s}^{-2}$ 130
6.11 HIreef Benchmark test. Time series of the free surface at different wave gauges for the propagation of solitary wave with $A/h = 0.3$ over 1:12 slope and exposed reef crest. Solid lines and circles denote computed and measured data. 131

6.12 Breaking solitary wave over 3D reef. Layout with locations for resistance-type wire wave gauges and ADVs (positions 2, 3, and 10 only) . . . 132
6.13 Breaking solitary wave over 3D reef. Free surface elevation at the gauges. The blue line denotes the computed result and the black circles represent the laboratory data. 133
6.14 Breaking solitary wave over 3D reef. Horizontal flow velocities at the gauges. The blue line denotes the computed result, the black circles represent the laboratory data. . . . 134
6.15 Breaking solitary wave over 3D reef. Free surface after 8 s. Overtopping of the cone with strong vortex formation at lee side. Wave breaking front is described as flow discontinuity. . . . 134
6.16 Breaking solitary wave over 3D reef. Free surface after 17 s. Run-up on an initially dry concrete bed. The flow adjusts to the small roughness elements over the dry bed resulting from imperfections in the concrete pouring. . . . 135
6.17 Breaking solitary wave over 3D reef. Free surface after 35 s. Sloshing continues in the basin with small vortices advected around the cone. Some water is dripping down the slope. . . . 135
6.18 Wave refraction-diffraction over an elliptic shoal. Three-dimensional view of the free surface and the topography elevation at $t = 50$ s . . . 136
6.19 Wave refraction-diffraction over an elliptic shoal. Berkhoff et al. [1982] experimental layout and position of the transects of the collected data. . . . 137
6.20 Wave refraction-diffraction over an elliptic shoal. Wave height comparison between the model and the experimental data along different transects (1)-(8). The blue line denotes the computed result, and the black circles represent the laboratory data. . . . 137

7.1 CUDA heterogeneous computing Guide [2013] . . . 141
7.2 GPU Implementation Flow Chart . . . 144
7.3 GPU Global Reduction Harris et al. [2007] . . . 145
7.4 Ratios of CPU and GPU computational speedup times for different domain sizes (total number of cells) with System 1 hardware . . . 149

8.1 Computational domain covering the inner portion of the Northshore of Oahu as denoted by the solid rectangle. The inner dashed rectangle denotes the extent of the sponge layer for wave absorption. The offshore boundary is pointing towards 315° (-45° rotation from true North). The gauges at Mokuleia are denoted by 8a, 8b, and 8c. Haleiwa Harbor is shown in Figure 8.2 in detail. Map adapted from West Maui Coastal Resilience Group report Phase I. . . . 154
8.2 Close-up view of Haleiwa Harbor (small rectangle in Figure 8.1. Only gauges 2, 3, and 4 are selected for model-data comparison. Map adapted from West Maui Coastal Resilience Group report Phase I. . . . 155
8.3 Original CDIP spectrum from swell 2. . . . 156

8.4 PSD of free surface elevation at Mokuleia array with a grid size of $\Delta x = \Delta y = 7\,\mathrm{m}$. Results from the Present model and from BOSZ for comparison. 158
8.5 PSD of free surface elevation at Haleiwa Harbor with a grid size of $\Delta x = \Delta y = 7\,\mathrm{m}$. Results from the Present model and from BOSZ for comparison. 159
8.6 PSD of free surface elevation at Mokuleia array with a grid size of $\Delta x = \Delta y = 21\,\mathrm{m}$. Results from the Present model and from BOSZ for comparison. 161
8.7 PSD of free surface elevation at Haleiwa Harbor with a grid size of $\Delta x = \Delta y = 21\,\mathrm{m}$. Results from the Present model and from BOSZ for comparison. 162

C.1 Variation of $\delta_{\min}$ and $J(\delta_{\min})$ for differnt $x_{\max}$. 199
C.2 $C_{g,SGN,\delta_{\min}}/C_{g,Airy}$ for several values of $x_{\max}$. Up Left: $x_{\max} = \pi$, Up Right: $x_{\max} = 2\pi$, Bottom Left: $x_{\max} = 4\pi$, Bottom Right: $x_{\max} = 8\pi$. 201
C.3 The black line describes the analytical solution from Airy wave theory and the blue line represents the modified SGN solution 202
C.4 Computation of wave transformation over a submerged bar. Black circles denote laboratory data from Beji and Battjes [1993]. The red lines are time series from the original SGN solution and blue lines represents the results of the improved SGN solution ($\delta_{Nwogu} = 0.17$) 203

D.1 Definition sketch. 206
D.2 Stencil dependency in the elliptic solver 208

E.1 Empirical spectra comparison 213

List of Tables

3.1 Solitary wave solution: error and convergence rate 68

4.1 Generation of JONSWAP wave spectrum: model configurations 91

5.1 Wave and topography inputs for each case configuration 99

7.1 Performance of Thomas Algorithm and *cusparseSgtsvStridedBatch* in Solving Nwogu's tri-diagonal Systems 147

7.2 Hardware specifications 149

8.1 Wave conditions and tide levels for the two test scenarios 153

8.2 Summary of all model runs. 154

8.3 Present model performance compared to BOSZ on CPU hardware. 163

8.4 Performance and cost: CPU vs. GPU 164

List of Acronyms

CFL	Courant-Friedrichs-Lewy
CPU	Central Processing Unit
CUDA	Compute Unified Device Architecture
FD	Finite Difference
FE	Finite Element
FV	Finite Volume
GPU	Graphics Processing Unit
IG waves	Infragravity waves
MPI	Message Passing Interface
OpenMP	Open Multi-Processing
PDE	Partial Differential Equation
PSD	Power Spectral Density
SGN	Serre-Green-Naghdi
SWE	Shallow Water Equations (Saint-Venant Equations)
SWL	Still Water Level
TDMA	TriDiagonal Matrix Algorithm (Thomas algorithm)
TKE	Turbulent Kinetic Energy
TVD	Total Variation Diminishing

List of Symbols

Constants

ρ	Density of sea water	1035 $kg.m^{-3}$
g	Gravitational acceleration	9.81 $m.s^{-2}$

Variables

x, y	Cartesian coordinates	m
z	Vertical elevation	m
t	time	s
i, j	Spatial indices (subscript)	–
n	Temporal index (superscript)	–
h	Still water depth	m
η	Free surface elevation	m
H	Total water depth	m
u	Reference horizontal velocity x-direction	$m.s^{-1}$
v	Reference horizontal velocity y-direction	$m.s^{-1}$
w	Vertical velocity	$m.s^{-1}$
$\wp$	Depth-integrated non-hydrostatic pressure	$m^3.s^{-2}$
ϱ	Non-hydrostatic pressure trace at the bottom	$m^2.s^{-2}$
ψ_{wm}	Wavemaker source term	$m^2.s^{-2}$
ψ_c	Dispersion source term in Nwogu's continuity	$m^2.s^{-2}$
ψ_u	Dispersion source term in Nwogu's momentum equation (x-direction)	$m^2.s^{-2}$
ψ_v	Dispersion source term in Nwogu's momentum equation (y-direction)	$m^2.s^{-2}$
Z_α	Nwogu's reference elevation	m
k	Turbulence kinetic energy	$m^2.s^{-2}$
ν_t	Eddy Viscosity	$m^2.s^{-1}$

Variables

f	Frequency	Hz
λ	Wave length	m
ψ	Wave phase	rad
ω	Angular frequency	$rad.s^{-1}$
C	Wave celerity	$m.s^{-1}$
C_g	Group speed	$m.s^{-1}$
H_s	Significant wave height	m
T_p	Peak period	s
ε	Nonlinearity parameter	$-$
μ	Dispersion parameter	$-$
H_{min}	Minimum water depth	m

Chapter 1

Introduction

Contents

1.1 General Context **15**
1.2 Nearshore Wave Assessment **16**
1.2.1 Mathematical Framework 16
1.2.2 Numerical Methods 18
1.3 Model-based Forecast **19**
1.4 Objectives **20**
1.5 Outline **21**

1.1 General Context

The Sixth IPCC Report on Climate (Pörtner et al. [2022]) is warning that the world's average temperature will exceed the critical threshold of 1.5 degrees Celsius over the next few decades unless drastic actions are taken. It is uncertain whether these measures will be implemented, as to date no significant reduction in global anthropogenic greenhouse gas emissions has been noticeable. The consequences of a continuation of global warming are widely understood and fairly undisputed. With a focus on the coastal zone, it is foreseeable that sea level rise will build up excessive pressure on natural and engineered beaches since the elevated water table will allow for wave action to be carried closer to shore and, at the same time, impair beaches' ability to act as a dynamic buffer for the absorption of the waves' energy.

In a broader context, it is clear that the landfall of extra-tropical and tropical storms from the open ocean can result in massive onshore surges of seawater that flood low-lying coastal areas, endangering life and property. Tsunamis can have a similar destructive nature; hence, early warning systems and evacuation plans are in place and are constantly improved for these extreme events. Less noticed by policymakers and the general public, large swell events and associated wave-driven flooding occur much more frequently than catastrophic storm surges or tsunamis, and when they coincide with spring tides, they become more than just "nuisance flooding". These events are also much more difficult to quantify than extreme scenarios, and their impact will grow more acute as the number and duration of such events are bound to increase (Emanuel [2005]). In addition, the continuation of sea level rise due to global warming will soon push the water table of regular high tides into the range of spring high tides and exacerbate the severity of the impact that large swell waves have on the coastal zone. Therefore, it is necessary from a practical point of view to plan ahead with protective measures in anticipation of the consequences linked with sea level rise and nearshore waves. While the effects of large swells on coastal erosion, freshwater aquifers, and coastal infrastructure are already significant (Storlazzi et al. [2015]), the effects of large swell waves in combination with sea level rise are still poorly understood. Particularly for complex seabeds, wave breaking and shoreline erosion do not necessarily occur in the area of the largest wave activity. Especially infra-gravity (IG) oscillations, which are the driving factor for locally large run-up and nearshore currents, are often generated by nonlinear wave-by-wave processes and therefore require expensive numerical solutions of adequate governing equations to be properly quantified.

While traditional wave forecasting methods provide average or integrated wave conditions near the coast, future climate conditions will call for operational phase-resolving wave modeling to provide a more complete picture of potentially hazardous conditions along entire stretches of coastline. Therefore, new models must be developed that can provide reliable solutions over large domains with potentially millions of grid cells and that can be computed with standard hardware in a relatively short amount of time.

This Ph.D. research is tightly embedded into the E2S chair HPC-Waves, and it is a crucial component of one of the two chair's axes—namely, numerical model development with a focus on practical applications. This research work is part of the chair's effort to provide novel solutions to today's coastal problems, such as hazard mitigation associated with flooding, and to strengthen collaborative initiatives in the future.

1.2 Nearshore Wave Assessment

Based on the outline above, it is clear that coastal engineers, emergency managers, and scientists have been gaining interest in quantitative assessment of swell waves and their behavior at the coast. Today, nearshore wave forecasts and hindcasts are indispensable for the construction of coastal structures, the development of ports, and flood risk assessment. To better understand nearshore conditions, a large number of physical wave measurements and field observations were used to calibrate and validate numerical (*e.g.*, Booij et al. [1999]) and empirical models (*e.g.*, Komar and Gaughan [1973]; Caldwell and Aucan [2007]). Though it is possible to obtain rough approximations of wave run-up based on empirical formulae (*e.g.*, Stockdon et al. [2006]), such estimates are often of limited applicability in areas with complex shorelines and bathymetries, such as those encountered along the Basque coast or in tropical areas with fringing reefs. In the absence of applicable formulations, the transformation of swells in nearshore areas can be analyzed by constructing a physical scaled model. Obviously, these models require a significant investment of resources, and their applicability is quite restricted to certain wave and terrain conditions. That is why, in recent years, laboratory experiments have primarily been used to gain data for benchmarking and validation of mathematical and numerical models.

1.2.1 Mathematical Framework

Due to the substantial increase in computing power, numerical solutions for coastal wave problems are now reliable, cost-effective, and time-saving tools for engineering designs. The advantage of numerical models lies in their simplicity, with which different layouts can be constructed and tested compared to a physical model. Open ocean swell propagation can be fairly well described with wave action balance equations as implemented in spectral wave models, since it is the overall wave energy that matters rather than the details of individual waves in a swell. In recent years, a number of advanced wave spectral models have been developed, (*e.g.*, WAM model (WAMDI [1988]), Simulating WAves Nearshore (SWAN [2015]) and WAVEWATCH III (Tolman et al. [2009]). These models, also known as the "third-generation spectral models", are able to account for some key nearshore processes, such as refraction, shoaling, white capping, and depth-induced breaking. Spectral models are based on the assumption that phase-dependent quantities vary slowly over wavelength-scale distances (Rusu and Soares [2013]). The fact that individual waves are not resolved allows for the use of coarse meshes and consequently fast computations of wave propagation over large open seas. However, closer to shore, where complex wave processes occur over relatively short distances, such as inside harbors, across the surf zone, or over steep sloping beaches, these models are limited in producing an accurate description of the complex and time-dependent wave field. Moreover, as the nearshore waves feel the bottom, nonlinear phenomena gain weight, and especially in the surf zone, wave-by-wave action takes over as the driving factor for secondary wave processes such as wave set-up and surf beat as well as infra-gravity waves and recirculation. This has led to the development of phase-resolving models to compute individual waves and their interactions, along with run-up and coastal inundation. Obviously, a phase-resolving computation calls for high grid resolutions of only a few meters, and thus, it is computationally much more demanding than a phase-averaged solution by a spectral model, where often mesh sizes of tens or even hundreds of meters are sufficient. Since

a complete phase-resolving model requires the solution of highly nonlinear, time-dependent, dispersive wave transformation processes, the details necessary for the solutions on a wave-by-wave basis can pose significant challenges to numerical solvers. And while recent advancements in computing power have paved the way for a wide and mainstream usage of numerical models, the computations of large-scale and three-dimensional wave problems are still not feasible in a timely manner, especially in the context of an operational system—not even counting expensive post-processing procedures. Therefore, efforts to lower the computation time are necessary and have been sought by reducing the dimensions of the numerical problem. A simplification can be achieved by integrating over the water column, thereby reducing the spatial dimensions from 3D to 2D. Such models, in which the vertical structure is not directly solved but only parametrically modeled, are known as two-dimensional in the horizontal plane models (*i.e.*, 2DH models).

Models based on the depth-averaged hydrostatic Shallow Water Equations, have been particularly popular for the phase-resolving computation of long waves, such as tsunamis and tidal waves. Due to the simplified structure of the governing equations, this mathematical framework has been extensively used in operational tsunami systems (*e.g.*, MOST (Titov and Gonzalez [1997]), TUNAMI (Imamura [1989]) and COMCOT (Wang [2009])) and has proven to be cost-effective, especially when dealing with long-distance wave propagation. Nevertheless, the accurate prediction of nearshore wave processes requires both nonlinearity and dispersive effects to be taken into account. Consequently, the SWE equations with their hydrostatic assumptions are expected to fail and become invalid for an application to nearshore swells. Boussinesq-type equations, although based on long-wave assumptions, resolve waves in intermediate water depths and have proven to be applicable to nearshore waves. This is because, unlike the SWE, they include some effects of frequency dispersion in the governing equations. The original Boussinesq equations (*i.e.*, Peregrine [1967]) have been extended to accurately compute wave propagation in deeper water (*i.e.*, $\lambda = 2h$). Equations such as the ones derived by Madsen and Sørensen [1992]; Nwogu [1993]; Wei and Kirby [1995]; Zou [1999, 2000] have become very powerful tools for nearshore wave assessment and have paved the way for the rise of new and improved dispersive wave models (*e.g.*, FUNWAVE-TVD (Shi et al. [2011]), BOSZ (Roeber and Cheung [2012]), COULWAVE (Kim et al. [2009]), TUCWave (Kazolea and Delis [2013])). While these extended equations have helped improve the dispersion properties required to compute waves in intermediate water, they still have limited applicability to deeper water. Several researchers have made attempts to practically eliminate water depth limitations and treat highly dispersive and nonlinear waves. Inspired by Nwogu [1993], Gobbi et al. [2000] presented a new form of high-order Boussinesq equations based on a linear combination of the velocities at two arbitrary vertical levels. This led to a dispersion relation that is valid up to $\lambda = h$. Madsen et al. [2002] extended the applicability of Boussinesq-type equations up to $\lambda = h/6$ by including higher-order derivative operators (*i.e.*, up to fifth-order). Madsen et al. [2003] achieved similar dispersion properties along with an improved velocity profile by using Padé approximants as an alternative to the Taylor series. This resulted in a much higher order of accuracy without increasing the order of the derivatives. This highly dispersive formulation was later extended to include rapidly varying bathymetry (*i.e.*, Madsen et al. [2006]). Along with the high-order Boussinesq-type equations, fully-dispersive formulations have also been proposed, which achieve the exact dispersion characteristics with no limitation on the water depth, *i.e.*, $h/\lambda < \infty$ (*e.g.*, Tsutsui et al. [1998]; Schäffer [2005]; Karambas and Memos [2009]; Klonaris et al. [2013]). Karambas and Memos [2009] derived a fully-dispersive model

through an inverse Fourier transform that offers significant advantages due to the small number of terms involved in the equations. Most recent improvements in Boussinesq-type equations aim to improve the nonlinear-dispersion properties of the model by splitting the vertical domain into a few layers (typically two to five). Lynett and Liu [2004] proposed a two-layer formulation with separate velocity polynomials within the layers and additional interface conditions. Multi-layer models such as Chazel et al. [2009]; Liu and Fang [2016]; Liu et al. [2018], provide a high-accuracy representation of the vertical structure, while allowing for lower-order operators in each layer. Other developments focused on improving the embedded nonlinear properties of the model as well as its performance for highly nonlinear and steep waves. This was typically achieved by allowing the nonlinear scaling parameter to be of order 1, thus retaining nonlinear dispersive terms in the formulations (see *e.g.*, Wei and Kirby [1995]; Madsen and Schäffer [1998]; Gobbi et al. [2000]). While fully-nonlinear Boussinesq equations have become long-accepted tools in nearshore wave modeling, some instabilities related to the fully-nonlinear and weakly-dispersive approximations have been recently reported by Madsen and Fuhrman [2020]. Madsen and Fuhrman [2020] has also shown that these potential trough instabilities are less pronounced when the accuracy of the linear dispersion relation is improved and are completely resolved when the dispersion relation is exact, such as in the case of High-Order Spectral (HOS) methods (*e.g.*, Dommermuth and Yue [1987]; Yates and Benoit [2015]; Raoult et al. [2016]; Zhang and Benoit [2021]). It is pretty evident that a fully-dispersive and fully-nonlinear formulation provides a complete and high-quality mathematical framework for nearshore wave computation; however, the inherent numerical complexity of such models significantly restricts their use, especially in an operational context.

As Boussinesq-type equations handle the effects of non-hydrostatic pressure variations, they solve essentially for the same processes as the so-called "non-hydrostatic" models. The two approaches differ from each other as the non-hydrostatic equations require the solution of an additional governing equation for the vertical pressure, which is used for the correction of the hydrostatic pressure computed by the SWE. The Boussinesq approach includes the vertical variation in pressure directly in the horizontal momentum equations through additional source terms. Since extended Boussinesq-type equations such as Nwogu [1993] include enhanced dispersion properties, non-hydrostatic models (*e.g.*, SWASH Zijlema et al. [2011]) require two or more layers to provide dispersion properties equivalent to those of standard Boussinesq-type models.

Boussinesq-type equations have evolved into powerful tools for studying waves in the nearshore environment and have found their way into reliable and operational wave modeling. However, models based on these equations contain two main limitations: they do not per se hold for wave breaking, and they are still considered to be computationally expensive, especially for large computational domains (*i.e.*, millions of grid cells). This means that as the waves approach the shore, wave-breaking effects, and run-up computations have to be dealt with while maintaining the stability and consistency of the high-order dispersive equations.

1.2.2 Numerical Methods

From a numerical standpoint, Boussinesq-type equations have been solved by using different numerical techniques such as Finite Difference (FD), Finite Element (FE), and Finite Volume (FV). Since these dispersive equations contain both hyperbolic and elliptic components, their

solution should account for the "wave-like" properties of the equations along with an accurate description of the non-hydrostatic elliptic terms. In the context of nearshore wave modeling, important processes such as wave propagation, nonlinear wave interactions, bore formation, and moving shorelines should all be adequately described within the numerical framework. This requires the numerical scheme to fulfill some key requirements:

- Mass and momentum conservation,
- Low diffusive and dispersive numerical errors,
- Well-balanced computations,
- Shock-capturing,
- Stable wet-dry transitions.

The numerical solutions of Boussinesq-type have been subject to many trends. Earlier solutions were based on traditional FD schemes solved on a staggered grid (Arakawa and Lamb [1981]). This approach has been successfully employed in many first-generation Boussinesq models, such as BOUSS-2D (Nwogu and Demirbilek [2001]), earlier versions of FUNWAVE (Wei and Kirby [1995]), and COULWAVE (Lynett et al. [2002])). While FD offers a simple and lean approximation for high-order derivatives, earlier implementations were not suitable for the computation of flow discontinuities, which are necessary for bore propagation and run-up. To overcome these limitations, recent Boussinesq models are based on a hybrid approach, in which the hyperbolic portion of the equations is computed with well-established FV schemes (*e.g.*, Toro [1989]; Roe [1986]; Kurganov et al. [2001]), while the dispersive terms are approximated using high-order FD. It is important to note that the vast majority of existing Boussinesq models are now based on hybrid FV-FD methods (Erduran et al. [2005]; Tonelli and Petti [2009]; Roeber et al. [2010]; Kazolea and Delis [2013]). This shift was motivated by the stability and shock-capturing capabilities of FV computations. A similar change in trend was also reported for hydrostatic tsunami models. High-order flux reconstructions in combination with TVD limiters were successfully used to reduce the inherent diffusivity of shock-capturing methods (*e.g.*, Jiang and Shu [1996]; Kim and Kim [2005]; Choi et al. [2018]). The use of a Cartesian grid for these high-order approximations provides more robust and stable computations due to consistent discretization errors. For this reason, the structured grid has been widely used for the implementation of operational Boussinesq models. To obtain a high-resolution solution for strategic sites, the uniform Cartesian grid has recently been combined with a nested grid approach for the solution of Nwogu's equations (Choi et al. [2022]). This approach has proven to be cost-effective and does not alter the solver's main structure. A different technique is based on the unstructured FE method. It has been successfully implemented for the solution of Boussinesq-type equations (*e.g.*, Walkley and Berzins [2002]; Panda et al. [2014]; Ricchiuto and Filippini [2014]). While this approach provides more grid flexibility and is applicable to unstructured mesh, the accuracy of the solution itself is similar to FD schemes.

1.3 Model-based Forecast

The rise of the global sea level has the potential to increase both the frequency and the magnitude of coastal hazards. This future scenario has prompted a great deal of effort to build

accurate and reliable forecasts of nearshore waves and flooding hazards. Wave forecasts are available almost globally; however, they are based on spectral wave models with relatively coarse resolutions and do not provide the necessary details and features for nearshore wave assessments. A new generation of phase-resolving wave forecasts has been implemented recently, such as the Wave Run-up Forecast for West Maui (PacIOOS [2022]) and the Imperial Beach Flood Forecast System (Fiedler et al. [2020]). These operational systems combine spectral models for offshore energy propagation with phase-resolving computations to model nearshore wave transformations. While phase-resolving models are reliable tools for providing high-resolution metrics of surf zone dynamics, coastal flooding, and overtopping, the high computational effort required by these models can limit their use in an operational setting. One main limitation is the long computation required to solve a substantial 2D nearshore domain. For this reason, many operational systems replace the 2D domain with 1D cross-shore transects. This obviously compromises the validity of the numerical solution, especially in the case of complex topography where refraction effects are a controlling factor. Another workaround to achieve timely results is based on using a coarse grid ($\Delta x > 10\,\mathrm{m}$) for the phase-resolving simulation combined with empirical formulae to extract the run-up. While computer clusters with multiple processors can reduce the computation time of these phase-resolving simulations, the cost of the hardware as well as their high energy consumption remain major issues.

Some of the main challenges for operational phase-resolving models are associated with

- Stability of the nonlinear and dispersive solution,
- Efficiency and robustness of numerical schemes,
- Reduced numerical errors (especially diffusive errors),
- Robust and reliable wave-breaking treatments,
- Localized and targeted mesh-refinement,
- Efficiency of the numerical algorithm with respect to implementation and parallelization.

It is obvious that the above items cannot be seen independently but rather present a complex and compound ensemble. Addressing one particular item might impair the solution of another. Nevertheless, a structured and dynamic approach with a focus on all problems can help find new avenues in the numerical modeling of nearshore waves.

1.4 Objectives

This thesis describes the systematic development of a computer code for the solution of nearshore waves with a focus on the improvement of accuracy and computational speed in comparison to established models. A clear focus of this work lies in the development of a model for practical applications. This implies that priority is given to solutions that are widely applicable rather than specific solutions with limited use for real-world challenges.

Further, it is the objective that the novel computer code can be used on consumer-grade hardware and still render fast and reliable solutions. This requires the development of a numerical code that is not only well parallelizable but also, and this is most important, provides solutions in an

efficient way so that accurate results can be obtained without excessively fine meshes or a large cell count.

1.5 Outline

The thesis outlines the step-by-step development of a new Boussinesq-type model for nearshore waves with a focus on computational efficiency. This research work is presented in the form of seven chapters:

Chapter 2 describes the development of a new Shallow Water model with the objective of being used as the foundation for the final dispersive model. We demonstrate the robustness and accuracy of this new numerical framework for shock and run-up computation. We also discuss the implementation and validation of the nested grid approach for the hydrostatic model. The sensitivity and efficiency of this mesh-refinement technique are highlighted using standard tsunami benchmark tests.

Chapter 3 outlines the 1D extension of the Shallow Water model into two dispersive models: one based on the fully nonlinear SGN equations (*i.e.*, Serre [1953]; Green and Naghdi [1976]) and another on extended Boussinesq equations (Nwogu [1993]). The two equations are rewritten to obtain an elliptic-hyperbolic decoupling, and their numerical solutions are verified using analytical tests. The numerical solutions of these equations are compared to standard benchmark tests to illustrate the effects of the nonlinear and dispersion properties of the governing equations.

In **Chapter 4,** we discuss how the properties of the hyperbolic solver can impact the numerical solution, particularly in the case of long-distance wave propagation. We focus on one important and often overlooked issue: numerical diffusion. We propose an adequate numerical scheme with only low numerical diffusivity and compare its performance to that of well-established numerical schemes. For that, we implement the solution of Nwogu's equation using the hybrid FV/FD approach. Simple periodic wave propagation tests are then conducted to assess the accuracy of each approach and its sensitivity to grid resolution.

Chapter 5 details the implementation of a turbulent kinetic energy (TKE)-based eddy viscosity approach to account for wave breaking effects in the Present Boussinesq-type model. We also examine the accuracy and grid-convergence of this method compared to other standard wave breaking closures.

Chapter 6 describes the development of the 2D dispersive wave model along with some key features such as internal wave generation, wave-absorbing boundaries, and eddy viscosity wave-breaking closure. The Present model is based on the conservative solution of Nwogu's equations and incorporates several strategic techniques to improve the efficiency of the 2D computations. The new model is then validated using a series of benchmark tests that include a wide range of wave transformation processes such as refraction-diffraction, shoaling, wave-breaking, and run-up.

Chapter 7 outlines the GPU-based implementation of the new model into the CUDA C/C++ framework. While the explicit nature of the Present model caters to massive data parallelism and thus enables GPU implementation, the inherent data-dependent elliptic solver along

with the wavemaker source term must be specifically addressed to achieve improved performance without alteration of the solution. The speed-up of the GPU parallelization is then compared to a standard CPU parallelization on conventional hardware of comparable cost.

In **Chapter 8** we evaluate the model's applicability to a real-world case that involves energetic nearshore waves over complex reef-dominated bathymetry. The test case uses a large computational domain and computationally intensive wave generation with over 40,000 waves, which are ideal conditions for testing the accuracy and performance of this new implementation. We also compare the quality of the results from the Present model with the solutions of a well-established model (BOSZ) and finally highlight some key differences.

Chapter 2

Shallow Water Solver

Contents

2.1 Introduction 24
2.2 Efficient Numerical Computations of Long-Wave Run-Up and Their Sensitivity to Grid Nesting 24
2.3 Malpasset Dambreak 57
2.4 Conclusions 59

2.1 Introduction

The Shallow Water Equations (SWE) have gained massive popularity for modeling a wide range of free surface problems, including open channel flows, tides, dam-break induced waves, and long waves such as tsunamis. Since SWE represent the backbone of many depth-integrated wave models such as Boussinesq-type models (*e.g.*, Roeber and Cheung [2012]; Shi et al. [2011]), non-hydrostatic models (*e.g.*, Zijlema et al. [2011]) and sediment transport models (*e.g.*, Audusse et al. [2012]), significant effort has been made to improve the numerical solutions regarding accuracy and computational efficiency. In this section, we present the rigorous development of a robust and accurate numerical framework for the solutions of the SWE, with the underlying objective of building the foundation for a depth-integrated dispersive wave model. The choice for the staggered grid approach was motivated by the robustness and accuracy demonstrated in many related papers (*e.g.* Zijlema et al. [2011], Yamazaki et al. [2011], and Stelling and Zijlema [2003]). The discretization used for the model is adapted from Stelling and Duinmeijer [2003] and extended in some aspects. This implementation has proven to satisfy key properties of a hyperbolic solver: shock-capturing, well-balanced computations, and water depth non-negativity preservation. Such properties are necessary for accurate computations of long-wave propagation and run-up. The implementation is then combined with a nested grid approach to achieve efficient local run-up computations. The performance of the numerical model is evaluated by running a series of benchmark tests presenting a wide range of free surface flow problems, such as flows with shocks, transcritical flows, flows over a frictional bed and with moving boundaries.

2.2 Efficient Numerical Computations of Long-Wave Run-Up and Their Sensitivity to Grid Nesting

The implementation of the hydrostatic model is described and validated with analytical and laboratory tests for the computation of long wave run-up. The efficiency of the new SWE solver is enhanced with the implementation of the nested grid technique, which can achieve high-resolution run-up computations at low computational cost. This work takes the form of a published article (Mihami et al. [2022]), inserted here in its original form.

Water Waves
https://doi.org/10.1007/s42286-022-00070-8

ORIGINAL ARTICLE

Efficient Numerical Computations of Long-Wave Run-Up and Their Sensitivity to Grid Nesting

Fatima-Zahra Mihami[1] · Volker Roeber[1,3] · Denis Morichon[2]

Received: 16 October 2022 / Accepted: 25 October 2022

Abstract

Computation of long-wave run-up has been of high interest in the fields of ocean sciences and geophysics—particularly for tsunami and river flood modeling. An accurate calculation of run-up and inundation requires the numerical model to account for a sequence of critical processes—each of them posing a different challenge to the numerical solution. This study presents the strategic development of a numerical solution technique for shallow water equations with a focus on accuracy and efficiency for long-wave run-up. The present model is based on an explicit second-order finite-volume scheme over a staggered grid that efficiently achieves fundamental properties. The scheme is well-balanced and preserves shock fronts without the need for computationally expensive solvers. The streamlined code serves as a foundation for the implementation of nested grids. Computations of commonly used long-wave benchmark tests showcase that accurate predictions of local extreme run-up can often be achieved with highly refined yet spatially focused nested grids. Strategic grid nesting can lead to stable and accurate solutions of run-up at locations of interest and reduce the computational load to a fraction of what is usually necessary for a comparable solution over a single grid.

Keywords Shallow water equations · Explicit staggered grid · Grid nesting · Run-up · Long-waves · Numerical methods

1 Introduction

The estimation of run-up from long waves is crucial for the assessment and prediction of hazardous flooding scenarios associated with tsunamis and storm surges. As wave

Volker Roeber
volker.roeber@univ-pau.fr

1 Université de Pau et des Pays de l'Adour, E2S-UPPA, chair HPC-Waves, SIAME, Anglet, France

2 Université de Pau et des Pays de l'Adour, E2S-UPPA, SIAME, Anglet, France

3 Department of Oceanography, University of Hawai'i at Mānoa, Honolulu, HI, USA

Published online: 17 November 2022

run-up is the final stage a water wave undergoes when it reaches the shore, it depends on multiple processes such as wave transformation, breaking, and interaction with dry land. Consequently, a substantial and continuous effort has been made to better understand and compute the run-up processes of long waves [1]. This includes studies with respect to the derivation of analytical solutions for simplified geometries (e.g., [2–5]), laboratory experiments (e.g., [6–8]), and development of new numerical methods (e.g., [9, 10]). The latter provides approximate yet valid run-up solutions in more general settings suitable for the reconstruction of past events, forecasting, and practical engineering applications.

Numerical models for long waves, such as tides, storm surges, and tsunamis, have traditionally been based on shallow-water equations (SWE). Despite their simplistic hydrostatic assumptions, the SWE provide a valid basis for many long-wave problems and are often preferred over more complete equations thanks to their hyperbolic nature in which shocks can form as part of the solution. These depth-averaged equations have proven to give a reasonable balance between the accuracy and numerical cost [11] and serve by far as the most commonly used baseline for run-up calculations (e.g., [12–14]). Various numerical techniques have been proposed for the discretization of the SWE, ranging from conventional mesh-based methods such as finite difference (FD), finite volume (FV), or finite element (FE) to unconventional mesh-free methods such as smooth particle hydrodynamics (SPH) [15]. The numerical solutions of SWE have been subject to many trends. Earlier solutions were based on traditional FD schemes solved on a staggered grid [16]. This approach has been successfully employed in many first-generation tsunami models (e.g., TUNAMI [17], COMCOT [18]). Several wetting–drying techniques have been proposed to achieve a reasonable representation of the run-up heights. Shuto and Goto [19] used a staggered scheme with a Lagrangian description for the moving boundaries. Another approach has been based on the Neumann-type technique, which has been used to extrapolate the velocity at the wet-dry fronts [20], while Liu et al. [21] modeled the run-up based on water-level changes through flooding and drying of the cells.

FD methods offer a simplified solution for hyperbolic equations. However, they are known to exhibit deficiencies when dealing with flow discontinuities [22, 23], which particularly require local conservation of both mass and momentum. These conservation properties are necessary for the transport of breaking waves toward the shore and, hence, are important for the accuracy of the run-up computation. FV methods, on the other hand, solve the integral form of the SWE and directly benefit from conservation and shock-capturing capabilities. For this reason, FV methods such as Godunov [24], and Roe [25] solvers, which were previously used in gas dynamics, have become increasingly popular for the solution of long-wave problems. A new generation of tsunami and flooding models has been developed [26–29] based on a finite-volume interpretation of the equations, where the in-going and out-going fluxes over a control volume are computed with approximate Riemann solvers (e.g., [30–32]). These solvers are designed to preserve the hyperbolicity of the governing equations to allow for the formation of discontinuities in the numerical system. However, hyperbolicity can be a source of problems for the solution of the SWE. One drawback of this property is the well-balance between flux gradient, and source terms [33]. This means that models based on the FV approach often require computationally expensive techniques

to ensure the scheme is well-balanced—especially in the presence of dry cells [22, 34–38]. With respect to run-up and as a way to deal with the numerical problems of the moving shoreline, many FV schemes employ an artificial bed-wetting algorithm. These work through the definition of a minimum value of the water depth in the dry cells adjacent to the wet cells for computation of the numerical flux [37, 39]. Another difficulty for these schemes lies in the conservation of the non-negativity of the water depth—especially in the case of run-down [40]. Nevertheless, several operational models such as FUNWAVE ([41], COULWAVE [42], and BOSZ ([43]) successfully utilize these schemes.

Another approach for solving the SWE is linked to the use of conservative staggered schemes. These methods benefit from the efficiency and robustness of the FD approximations while achieving conservative and shock-capturing properties. Such schemes have been successfully applied to flows at high Froude numbers, including hydraulic jumps and inundation of dry areas (e.g., [44–48]). These schemes are based on specific FD approximations, which satisfy the Rankine–Hugoniot jump condition at a discrete level [33], and achieve valid solutions for rapidly varying flows. The concept from [45] has been widely used in many operational wave and run-up models (e.g., SWASH [49], NEOWAVE [50] and Xbeach [51]). This scheme guarantees the positivity of the water depth under the standard Courant–Friedrichs–Lewy (CFL) condition and, therefore, is very efficient for the computation of large-scale inundation problems.

The design of the numerical solutions of long-wave run-up requires taking the multi-scale nature of the problem into account, i.e., large-scale long-wave propagation in combination with the small-scale run-up and inundation processes. High spatial resolution is necessary for a detailed representation of the run-up process. However, computing a high-resolution grid over the entire domain is often unnecessarily expensive and can hinder the applicability of the model to real problems. With the objective of achieving efficient long-wave run-up computations, it is, therefore, desirable to utilize different grid sizes—each appropriate for the particular problems in the propagation and the run-up stages. Different approaches have been used to obtain local mesh refinement. For example, traditional nested grid methods have been implemented in tsunami models [17, 18, 52]. These techniques are usually built into structured grids where the refinement arises from the insertion of a sub-grid with higher resolution. The exchange of information between the grids is achieved either with one-way or two-way interactions. On the other hand, for unstructured grids, an adaptive mesh refinement technique has been successfully implemented in a number of long-wave models (e.g., [26, 53, 54]). The adaptive mesh refinement generates locally refined cells adapted to the flow condition without the need to use fixed sub-grid [55]. The refined region is, therefore, able to move with the area of interest, and unnecessary refinement is avoided. The disadvantage of these methods lies mainly in the complexity of the grid generation techniques, which require intensive data storage. In addition, the time step constraint is bound to the smallest grid cell that can hinder the efficiency of the implementation for explicit schemes [56].

This paper presents the rigorous development of a stable and accurate numerical framework for the computation of long-wave run-up. We address the details of the numerical scheme and outline the strategy for grid nesting to achieve a fast and

low-cost numerical tool for run-up computations. The model is based on a conservative staggered scheme in which the shock-capturing capabilities are achieved by satisfying the Rankine–Hugoniot at a discrete level. The scheme avoids the splitting of the free surface gradient into a pressure flux and topography term, resulting in direct non-negativity preserving and well-balanced computations. The verification process checks off the fundamental properties necessary for the computation of run-up: shock-capturing capabilities, moving boundaries with bottom friction, and exchange of information across nested grids. Two standard tsunami benchmark datasets are then employed to demonstrate the sensitivity of long-wave run-up to the overall grid resolution as well as to the extent of the nested grid and the refinement factor.

2 Methodology

2.1 Governing Equations

The present study considers the two-dimensional, depth-averaged Shallow Water equations (SWE). These equations provide a powerful baseline for long-wave modeling thanks to their wave-like hyperbolic structure. Moreover, the SWE serve as the backbone for many numerical models that address nearshore wave propagation and inundation. This is the case for dispersive Boussinesq-type and non-hydrostatic models in which the governing equations contain the SWE as a subset.

The SWE are derived from the Navier–stokes equations under the following assumptions: (a) the pressure is hydrostatic, and (b) the vertical distribution of the horizontal velocity is uniform (no variation). Under these assumptions, the equations take the following differential form in Cartesian coordinates:

$$\frac{\partial h}{\partial t} + \frac{\partial hu}{\partial x} + \frac{\partial hv}{\partial y} = 0 \tag{1}$$

$$\frac{\partial hu}{\partial t} + \frac{\partial hu^2}{\partial x} + \frac{\partial huv}{\partial y} + gh\frac{\partial \eta}{\partial x} = -gn^2\frac{u\sqrt{u^2+v^2}}{h^{1/3}} \tag{2}$$

$$\frac{\partial hv}{\partial t} + \frac{\partial huv}{\partial x} + \frac{\partial hv^2}{\partial y} + gh\frac{\partial \eta}{\partial y} = -gn^2\frac{v\sqrt{u^2+v^2}}{h^{1/3}} \tag{3}$$

We define t as the time variable, x and y are the space variables, h is the water depth, u and v are the depth-averaged velocities in the x- and y-directions, respectively. η refers to the free surface elevation: $\eta\,(x, y, t) = h\,(x, y, t) - d\,(x, y)$, where d is the positive bottom topography (Fig. 1). The constant g is the gravitational acceleration and n is the Manning roughness coefficient [s $m^{-1/3}$].

We write the SWE, Eqs. (1)–(3) in a conservative form to ensure the conservation of mass and momentum across discontinuities. The conserved variables, in this case, are the total water depth h and its product with the velocity components: hu and hv. In this form of the equations, we avoid the splitting of the free surface gradient into an artificial flux gradient and a source term that includes the effect of bed slope. The free surface gradient is, therefore, computed independently of the numerical flux, and

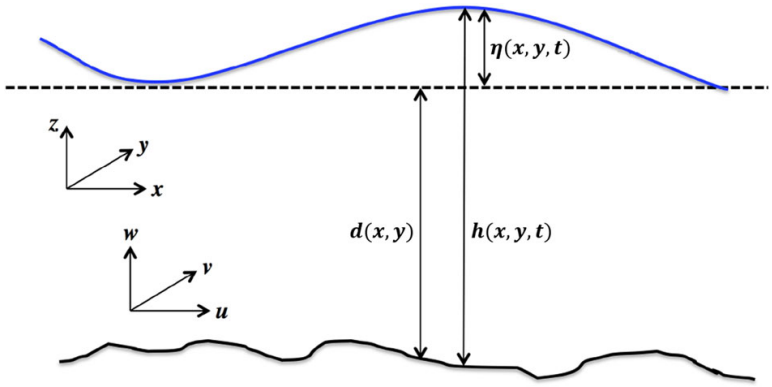


Fig. 1 Definition sketch for the free surface flow problem with key variables

no additional treatments are required to ensure that the scheme is well-balanced. The preservation of shocks and discontinuities will consequently depend on the numerical approximations of the scheme, which have to satisfy the Rankine–Hugoniot jump condition at the discrete level [33].

We introduce the auxiliary variables p and q, which denote the mass fluxes:

$$p = hu \qquad q = hv \tag{4}$$

We rewrite the SWE in the following form:

$$\frac{\partial h}{\partial t} + \frac{\partial p}{\partial x} + \frac{\partial q}{\partial y} = 0 \tag{5}$$

$$\frac{\partial hu}{\partial t} + \frac{\partial pu}{\partial x} + \frac{\partial qu}{\partial y} + gh\frac{\partial \eta}{\partial x} = -gn^2\frac{u\sqrt{u^2+v^2}}{h^{1/3}} \tag{6}$$

$$\frac{\partial hv}{\partial t} + \frac{\partial pv}{\partial x} + \frac{\partial qv}{\partial y} + gh\frac{\partial \eta}{\partial y} = -gn^2\frac{v\sqrt{u^2+v^2}}{h^{1/3}} \tag{7}$$

It is worth mentioning that in the momentum equations Eqs. (6) and (7), the variables hu and hv in the local acceleration and the variables p and q in the convective acceleration play different roles. The former is a storage quantity, while the latter is a transport quantity. Consequently, these terms are approximated differently, and in order to avoid confusion, we avoid using the same symbols.

2.2 Conservative Staggered Scheme

As detailed in the introduction, a variety of numerical schemes have previously been developed for the solution of the SWE. The choice of the numerical scheme depends mainly on the problem being addressed, which defines the requirements for the scheme properties. For the computation of long-wave run-up, a conservative shock-capturing scheme is crucial for the preservation of momentum and propagation of shocks at the correct speed and height. Other important properties are the well-balanced approximations of the topography variations, along with the non-negativity of the water depth

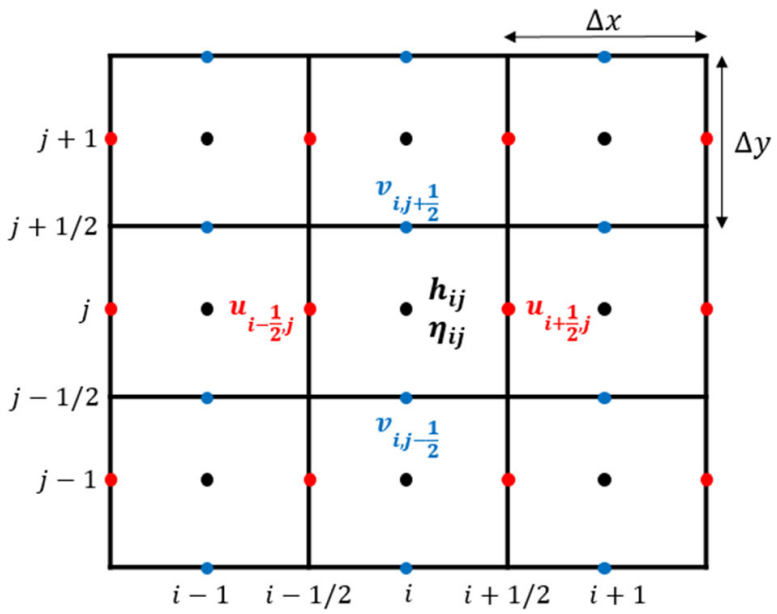


Fig. 2 Schematic of the 2D staggered grid

to ensure mass conservation across wet/dry transitions without parasitic waves. This adds to the stability and robustness of the numerical model—particularly over irregular bathymetry. Consequently, a scheme that provides these features is suitable for computing wave-breaking processes and, subsequently, wave run-up estimations.

In this study, the objective is to develop a lightweight yet accurate and stable solution structure that keeps the computational expenses at a low level. For these reasons, we utilize a conservative scheme on a staggered grid where the numerical fluxes are computed with simple FD approximations instead of Riemann solvers. The SWE variables, in this case, are approximated on a staggered C-grid: the total water depth h and the bed topography d are defined at the cell center, and the depth-averaged velocities (u, v) are stored at the cell interfaces (see Fig. 2).

For the discretization, we consider a 2D rectangular computational domain with a uniform grid spacing of Δx and Δy in the x- and y-directions, respectively. The variables stored at the cell center are expressed as $x_{i,j}$, where i and j are the spatial indices in the x- and y-directions. The variables stored at the cell interface are denoted by $x_{i\pm\frac{1}{2},j}$ or $x_{i,j\pm\frac{1}{2}}$ in the x- and y-directions, respectively. The time stepping is based on discrete, non-uniform time intervals $t^n = n\Delta t$, where n is the time index and Δt is the adaptive time step. The value of each variable a at the time level t^n is denoted with a^n. The water depth h is evaluated at each time step level $t = n\Delta t$, whereas, the depth-averaged velocities u and v are evaluated halfway between the present and the following time step $t = \left(n + \frac{1}{2}\right)\Delta t$. This leads to the staggering of spatial and temporal information and facilitates consistent second-order accuracy in space and time.

The present scheme first requires the solution of the continuity equation, which is subsequently used in the momentum equation. The discretization of the continuity

equation, Eq. (5), is expressed as

$$\frac{h_{i,j}^{n+1} - h_{i,j}^{n}}{\Delta t} + \frac{p_{i+\frac{1}{2},j}^{n} - p_{i-\frac{1}{2},j}^{n}}{\Delta x} + \frac{q_{i,j+\frac{1}{2}}^{n} - q_{i,j-\frac{1}{2}}^{n}}{\Delta y} = 0 \tag{8}$$

where

$$p_{i+\frac{1}{2},j}^{n} = \hat{h}_{i+\frac{1}{2},j}^{n} u_{i+\frac{1}{2},j}^{n+\frac{1}{2}} \qquad q_{i,j+\frac{1}{2}}^{n} = \hat{h}_{i,j+\frac{1}{2}}^{n} v_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} \tag{9}$$

$\hat{h}_{i\pm\frac{1}{2},j}^{n}$ and $\hat{h}_{i,j\pm\frac{1}{2}}^{n}$ are the water depths at the cell interfaces computed with an upwind approximation:

$$\hat{h}_{i+\frac{1}{2},j}^{n} = \begin{cases} h_{i,j}^{n} & \text{if } u_{i+\frac{1}{2},j}^{n+\frac{1}{2}} \geq 0 \\ h_{i+1,j}^{n} & \text{if } u_{i+\frac{1}{2},j}^{n+\frac{1}{2}} < 0 \end{cases} \qquad \hat{h}_{i,j+\frac{1}{2}}^{n} = \begin{cases} h_{i,j}^{n} & \text{if } v_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} \geq 0 \\ h_{i,j+1}^{n} & \text{if } v_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} < 0 \end{cases} \tag{10}$$

The next step is the solution of the momentum equation, Eq. (6). First, we consider the momentum equation without the friction term; the approximation of this term will be detailed later. We employ the FD approximations recommended in Zijlema [33] to achieve conservation of the momentum flux across discontinuities, as

$$\begin{aligned} &\frac{\bar{h}_{i+\frac{1}{2},j}^{n+1} u_{i+\frac{1}{2},j}^{n+\frac{3}{2}} - \bar{h}_{i+\frac{1}{2},j}^{n} u_{i+\frac{1}{2},j}^{n+\frac{1}{2}}}{\Delta t} + \frac{\hat{u}_{i+1,j}^{n+\frac{1}{2}} \bar{p}_{i+1,j}^{n} - \hat{u}_{i,j}^{n+\frac{1}{2}} \bar{p}_{i,j}^{n}}{\Delta x} \\ &\quad + \frac{\hat{u}_{i+\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} \bar{q}_{i+\frac{1}{2},j+\frac{1}{2}}^{n} - \hat{u}_{i+\frac{1}{2},j-\frac{1}{2}}^{n+\frac{1}{2}} \bar{q}_{i+\frac{1}{2},j-\frac{1}{2}}^{n}}{\Delta y} \\ &= -g \bar{h}_{i+\frac{1}{2},j}^{n+1} \frac{\eta_{i+1,j}^{n+1} - \eta_{i,j}^{n+1}}{\Delta x} \end{aligned} \tag{11}$$

Regarding the free surface gradient term, the use of the updated variable $\bar{h}_{i+\frac{1}{2},j}^{n+1}$ is necessary for the scheme to guarantee the entropy inequality as demonstrated in Doyen and Gunawan [47]. Further, it is necessary to approximate the convective acceleration with an upwind scheme, where the mass fluxes p and q are the criteria for upwinding and the velocities u and v are the upwinded quantities:

$$\hat{u}_{i,j}^{n+\frac{1}{2}} = \begin{cases} u_{i-\frac{1}{2},j}^{n+\frac{1}{2}} & \text{if } \bar{p}_{i,j}^{n} \geq 0 \\ u_{i+\frac{1}{2},j}^{n+\frac{1}{2}} & \text{if } \bar{p}_{i,j}^{n} < 0 \end{cases} \qquad \hat{u}_{i+\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} = \begin{cases} u_{i+\frac{1}{2},j}^{n+\frac{1}{2}} & \text{if } \bar{q}_{i+\frac{1}{2},j+\frac{1}{2}}^{n} \geq 0 \\ u_{i+\frac{1}{2},j+1}^{n+\frac{1}{2}} & \text{if } \bar{q}_{i+\frac{1}{2},j+\frac{1}{2}}^{n} < 0 \end{cases} \tag{12}$$

It is important to note, that a reversed approach where the upwinded quantities are p and q, leads to errors in the computation of the momentum fluxes across discontinuities as demonstrated in Zijlema [33].

Since the mass fluxes p and q are continuous quantities, an averaged approximation of these quantities can be applied in the computation of the convective acceleration terms:

$$\bar{p}^n_{i,j} = \frac{1}{2}\left(p^n_{i+\frac{1}{2},j} + p^n_{i-\frac{1}{2},j}\right) \qquad \bar{q}^n_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{1}{2}\left(q^n_{i,j+\frac{1}{2}} + q^n_{i+1,j+\frac{1}{2}}\right) \tag{13}$$

The flow depth, originally defined at the cell centroid, is approximated at the cell interface with arithmetic averaging to be used in the computation of the local acceleration:

$$\bar{h}^{n+1}_{i+\frac{1}{2},j} = \frac{1}{2}\left(h^{n+1}_{i,j} + h^{n+1}_{i+1,j}\right) \tag{14}$$

Finally, the momentum equation, Eq. (7), in the y-direction is solved in an analogous way as

$$\begin{aligned} &\frac{\bar{h}^{n+1}_{i,j+\frac{1}{2}} v^{n+\frac{3}{2}}_{i,j+\frac{1}{2}} - \bar{h}^{n}_{i,j+\frac{1}{2}} v^{n+\frac{1}{2}}_{i,j+\frac{1}{2}}}{\Delta t} + \frac{\hat{v}^{n+\frac{1}{2}}_{i,j+1}\bar{q}^n_{i,j+1} - \hat{v}^{n+\frac{1}{2}}_{i,j}\bar{q}^n_{i,j}}{\Delta y} \\ &\quad + \frac{\hat{v}^{n+\frac{1}{2}}_{i+\frac{1}{2},j+\frac{1}{2}}\bar{p}^n_{i+\frac{1}{2},j+\frac{1}{2}} - \hat{v}^{n+\frac{1}{2}}_{i-\frac{1}{2},j+\frac{1}{2}}\bar{p}^n_{i-\frac{1}{2},j+\frac{1}{2}}}{\Delta x} \\ &= -g\bar{h}^{n+1}_{i,j+\frac{1}{2}} \frac{\eta^{n+1}_{i,j+1} - \eta^{n+1}_{i,j}}{\Delta y} \end{aligned} \tag{15}$$

where

$$\hat{v}^{n+\frac{1}{2}}_{i,j} = \begin{cases} v^{n+\frac{1}{2}}_{i,j-\frac{1}{2}} & \text{if } \bar{q}^n_{i,j} \geq 0 \\ v^{n+\frac{1}{2}}_{i,j+\frac{1}{2}} & \text{if } \bar{q}^n_{i,j} < 0 \end{cases} \qquad \hat{v}^{n+\frac{1}{2}}_{i+\frac{1}{2},j+\frac{1}{2}} = \begin{cases} v^{n+\frac{1}{2}}_{i,j+\frac{1}{2}} & \text{if } \bar{p}^n_{i+\frac{1}{2},j+\frac{1}{2}} \geq 0 \\ v^{n+\frac{1}{2}}_{i+1,j+\frac{1}{2}} & \text{if } \bar{p}^{n\,\frac{1}{2}}_{i+\frac{1}{2},j+\frac{1}{2}} < 0 \end{cases} \tag{16}$$

and

$$\bar{q}^n_{i,j} = \frac{1}{2}\left(q^n_{i,j+\frac{1}{2}} + q^n_{i,j-\frac{1}{2}}\right) \qquad \bar{p}^n_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{1}{2}\left(p^n_{i+\frac{1}{2},j+1} + p^n_{i+\frac{1}{2},j}\right) \tag{17}$$

The flow depth in this case is approximated as

$$\bar{h}^{n+1}_{i,j+\frac{1}{2}} = \frac{1}{2}\left(h^{n+1}_{i,j} + h^{n+1}_{i,j+1}\right) \tag{18}$$

2.2.1 Second-Order Numerical Accuracy

Staggering of the variables both in space and time and utilization of the Leapfrog scheme lead to second-order accuracy for both the continuity and the momentum equations, except for the advection terms, [45, 49]. The flux terms are responsible for transporting the conserved quantities, and consequently, the construction of the advection terms with upwind differencing is necessary for the robustness and stability of the computed solution. However, first-order upwind methods are diffusive, and it is, therefore, useful to target second-order accuracy for all terms in the equations. One way to counter unnecessary numerical dissipation is based on extending the upwind scheme to the second order in combination with a slope limiter.

The approximations in Eqs. (10), (12), (16), and (17) can be improved by including two neighboring data points instead of only one, as is the case in the first-order upwind approach. The second-order upwind discretization is shown for Eq. (10) and applied to Eqs. (12), (16), and (17) in the same way:

$$\hat{h}^n_{i+\frac{1}{2},j} = \begin{cases} h^n_{i,j} + \frac{1}{2}\psi\left(r^+_{i+\frac{1}{2},j}\right)\left(h^n_{i,j} - h^n_{i-1,j}\right), & \text{if } u^n_{i+\frac{1}{2},j} \geq 0 \\ h^n_{i+1,j} + \frac{1}{2}\psi\left(r^-_{i+\frac{1}{2},j}\right)\left(h^n_{i+1,j} - h^n_{i+2,j}\right), & \text{if } u^n_{i+\frac{1}{2},j} < 0 \end{cases} \tag{19}$$

$r^+_{i+\frac{1}{2},j}$ and $r^-_{i+\frac{1}{2},j}$ are, respectively, the left and right gradients of the flow depth:

$$r^+_{i+\frac{1}{2},j} = \frac{h^n_{i+1,j} - h^n_{i,j}}{h^n_{i,j} - h^n_{i-1,j}} \quad , \quad r^-_{i+\frac{1}{2},j} = \frac{h^n_{i+1,j} - h^n_{i,j}}{h^n_{i+2,j} - h^n_{i+1,j}} \tag{20}$$

$\psi(r)$ is the slope limiter function, which locally reduces the solution from second to first order. The slope limiter is necessary to ensure the stability of the second-order upwind scheme at locations with opposite slopes, zero gradients, or sharp transitions. Here, a Generalized MinMod slope limiter is used:

$$\phi(r, \theta) = \max\left(0, \min\left(\theta r, \frac{1+r}{2}, \theta\right)\right) \tag{21}$$

θ is a parameter that controls the diffusivity. The generalized MinMod limiter is most dissipative for $\theta = 1$ when it reduces to the traditional MinMod limiter, and it is least diffusive for $\theta = 2$.

A predictor–corrector method can be used to improve the temporal accuracy of the advection terms to retain second-order accuracy in time. Here, we employ the Total Variation Diminishing (TVD) Runge–Kutta method. This method enhances the accuracy of the scheme in time while maintaining the strong stability property of the first-order Euler integration [57]. It is worth mentioning that other time integration methods can be combined with the above-described spatial discretization. For example, [49] used a MacCormack approach for the second-order integration in time.

We split the SWE equations into a convective acceleration term $\mathscr{F}$ and a free surface gradient term $\mathscr{G}$, which simplifies the description of the multi-step method:

$$\frac{\partial U}{\partial t} + \mathscr{F}(U) + \mathscr{G}(U) = 0 \tag{22}$$

where

$$U = \begin{bmatrix} h \\ hu \\ hv \end{bmatrix} \qquad \mathscr{F}(U) = \begin{bmatrix} \frac{\partial p}{\partial x} + \frac{\partial q}{\partial y} \\ \frac{\partial pu}{\partial x} + \frac{\partial qu}{\partial y} \\ \frac{\partial pv}{\partial x} + \frac{\partial qv}{\partial y} \end{bmatrix} \qquad \mathscr{G}(U) = \begin{bmatrix} 0 \\ gh\frac{\partial \eta}{\partial x} \\ gh\frac{\partial \eta}{\partial y} \end{bmatrix} \tag{23}$$

The discretization described in Eqs. (8), (11), and (15) can be summarized in the following expression:

$$U_{ij}^{n+1} = U_{ij}^{n} - \Delta t \mathscr{F}\left(U_{ij}^{n}\right) - \Delta t \mathscr{G}\left(U_{ij}^{n+1}\right) \tag{24}$$

where

$$U_{ij}^{n} = \begin{bmatrix} h_{ij}^{n} \\ \bar{h}_{i+\frac{1}{2},j}^{n} u_{i+\frac{1}{2},j}^{n+\frac{1}{2}} \\ \bar{h}_{i,j+\frac{1}{2}}^{n} v_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} \end{bmatrix} \tag{25}$$

At each time step, the variables (h, hu, hv) are solved using a two-stage time integration with an intermediate solution obtained by the predictor step. In the first step, we solve the equations with only the advection terms on the right-hand side:

$$U_{ij}^{*} = U_{ij}^{n} - \Delta t \mathscr{F}\left(U_{ij}^{n}\right) \tag{26}$$

This leads to a predictor solution of first-order accuracy for the complete continuity equation and incomplete momentum equations due to the lack of source terms. In the second step, the surface gradient terms are added to the momentum equations, and the predicted variables are corrected to full second-order accuracy in time by

$$U_{ij}^{n+1} = \frac{\Delta t}{2}\left(U_{ij}^{n} + U_{ij}^{*}\right) - \frac{\Delta t}{2}\mathscr{F}\left(U_{ij}^{*}\right) - \Delta t \mathscr{G}\left(U_{ij}^{n+1}\right) \tag{27}$$

It is important to emphasize that the predictor step of the time integration should only involve the convective acceleration terms. The source terms attain second-order accuracy by staggering the flow speed variables in time, and an application of Eq. (26) to the source terms would lead to inaccurate results.

The last term on the right-hand side of Eq. (27) applies only to the momentum equations and involves the corrected flow depth value h^{n+1}. This completes the fully

explicit time integration where no system of equations with data dependencies has to be solved.

2.2.2 Flooding and Drying

The wetting and drying process requires the model's performance for two fundamental processes. The scheme has to be well-balanced and has to preserve the positivity of the water depth across wet/dry boundaries.

The model is based on an explicit time integration. Although implicit methods lead to unconditionally stable computations with no restriction on the time step, their solution is admittedly complex and requires the solution of systems of equations [58–60]. This solution structure can pose a bottleneck, especially in parallelized implementations due to data dependencies.

The explicit scheme is stable under the Courant–Friedrichs–Lewy (CFL) condition given by

$$Cr = \Delta t \max \left(\frac{|u^{n+\frac{1}{2}}_{i+\frac{1}{2},j}| + \sqrt{g\hat{h}^{n}_{i+\frac{1}{2},j}}}{\Delta x}, \frac{|v^{n+\frac{1}{2}}_{i,j+\frac{1}{2}}| + \sqrt{g\hat{h}^{n}_{i,j+\frac{1}{2}}}}{\Delta y} \right) < 1 \qquad (28)$$

Under this condition, the scheme preserves the non-negativity of the water depth [33, 61]. This has the advantage that the run-up and inundation limits are inherent solutions of the numerical scheme and are not subject to additional ad-hoc flooding and drying treatments or require particular restructuring of the flux and source terms.

Since the flow depth can become arbitrarily small at the wet-dry transitions and, therefore, can lead to excessively high-velocity values, it makes sense to limit the minimum flow depth at the run-up front to a physically and numerically meaningful level. For efficiency reasons, the velocity values can be set to zero when the local water level falls below a threshold value $h_{\min}$, and the calculation of the momentum equations can be skipped:

$$u^{n+\frac{3}{2}}_{i+\frac{1}{2},j} = 0 \qquad \text{if} \qquad \frac{h^{n+1}_{i,j} + h^{n+1}_{i+1,j}}{2} < h_{\min} \qquad (29)$$

$$v^{n+\frac{3}{2}}_{i,j+\frac{1}{2}} = 0 \qquad \text{if} \qquad \frac{h^{n+1}_{i,j} + h^{n+1}_{i,j+1}}{2} < h_{\min} \qquad (30)$$

The value of $h_{\min}$ should be chosen as small as possible to accurately resolve the wet-dry front [37], but large enough to avoid physically questionable values in the local flow speed, which can cause excessively small time steps as shown in Eq. (28). It should be noted that the present scheme is not particularly sensitive to this threshold, and values between 10^{-8} and 10^{-4} m lead to virtually identical results. For the sake of quality verification and validation, we are using $h_{\min} = 10^{-8}$ m in the subsequent examples.

2.2.3 Friction Term

The friction terms added to the momentum equations are discretized as

$$gn^2 \frac{u^{n+\frac{3}{2}}_{i+\frac{1}{2},j}\sqrt{\left(u^{n+\frac{1}{2}}_{i+\frac{1}{2},j}\right)^2+\left(\bar{v}^{n+\frac{1}{2}}_{i+\frac{1}{2},j}\right)^2}}{\left(\bar{h}^{n+1}_{i+\frac{1}{2},j}\right)^{1/3}} \quad \text{and} \quad gn^2 \frac{v^{n+\frac{3}{2}}_{i,j+\frac{1}{2}}\sqrt{\left(\bar{u}^{n+\frac{1}{2}}_{i,j+\frac{1}{2}}\right)^2+\left(v^{n+\frac{1}{2}}_{i,j+\frac{1}{2}}\right)^2}}{\left(\bar{h}^{n+1}_{i,j+\frac{1}{2}}\right)^{1/3}} \tag{31}$$

where

$$\begin{aligned} \bar{v}^{n+\frac{1}{2}}_{i+\frac{1}{2},j} &= \frac{1}{4}\left(v^{n+\frac{1}{2}}_{i,j-\frac{1}{2}} + v^{n+\frac{1}{2}}_{i,j+\frac{1}{2}} + v^{n+\frac{1}{2}}_{i+1,j-\frac{1}{2}} + v^{n+\frac{1}{2}}_{i+1,j+\frac{1}{2}}\right) \\ \bar{u}^{n+\frac{1}{2}}_{i,j+\frac{1}{2}} &= \frac{1}{4}\left(u^{n+\frac{1}{2}}_{i-\frac{1}{2},j} + u^{n+\frac{1}{2}}_{i+\frac{1}{2},j} + u^{n+\frac{1}{2}}_{i-\frac{1}{2},j+1} + u^{n+\frac{1}{2}}_{i+\frac{1}{2},j+1}\right) \end{aligned} \tag{32}$$

and n, with units [s $m^{-1/3}$], is the Manning roughness coefficient representing the bottom property. Using the variables $\left(h^{n+1}_{i,j+\frac{1}{2}}, u^{n+\frac{3}{2}}_{i+\frac{1}{2},j}, v^{n+\frac{3}{2}}_{i+\frac{1}{2},j}\right)$ from the next time step in the friction terms improves the accuracy and the robustness of the solution [49].

Some of the following numerical tests, presented in Sect. 3, are computed with the Darcy–Weisbach formulation, which requires replacement of the term $\frac{gn^2}{h^{1/3}}$ by $\frac{f}{8}$. f is the dimensionless Darcy–Weisbach coefficient.

2.3 Nested Grid Method

The accuracy and applicability of a numerical model for free surface flows can substantially benefit from an efficient mesh refinement technique. Here, we concentrate on the nested grid method, which provides a reasonable trade-off between computational complexity and the gain in accuracy of the numerical solution for long-wave run-up. A simplified approach for mesh refinement involves the insertion of a high-resolution Child grid into a surrounding Parent grid of coarser resolution. The grids are herein fixed in space and predefined before the computation is executed. The SWE are solved independently in each grid. Consequently, the overall solution structure of the governing equations remains untouched as the exchange between the grids only requires interpolation of the key variables.

The staggered C-grid has been widely used in combination with embedded grid models due to its simplicity and conservative properties [56, 62, 63]. In this study, we build the nested grid approach on some of the techniques used and validated by several previously developed tsunami models [17, 18]. Several features are expected from a functioning grid nesting technique:

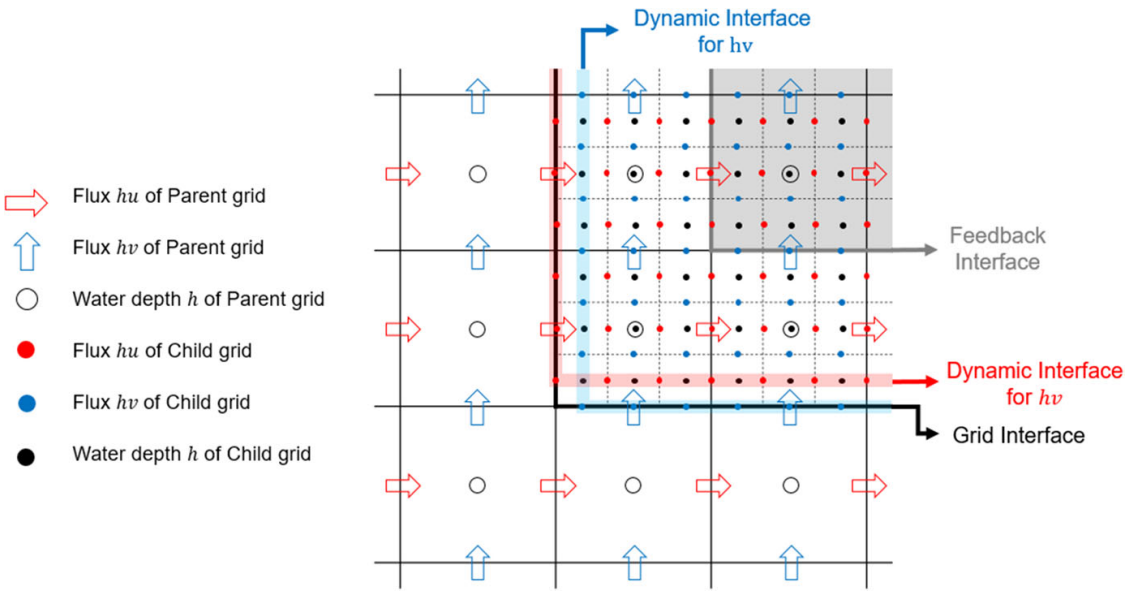


Fig. 3 Schematic illustrating the two-way nesting process on a Arakawa C grid

Data exchange The exchange of information between the grids occurs along the boundary of the inner grid. The Parent grid provides the boundary conditions to the Child grid in a one-way interaction. The flux variables (i.e., hu and hv) from the coarse grid are linearly interpolated in time and space and then dynamically imposed in each time step as boundary conditions to the solution of the Child grid (see Fig. 3). For a two-way interaction, the high-resolution free surface elevation from the Child grid is used to update the information in the Parent grid via an averaging operator. The update of the free surface only occurs inside the feedback interface in the Parent grid (see Fig. 3), rather than in the domain occupied by the Child grid. Several authors have proposed separating the feedback interface from the dynamic interface where the boundary values are interpolated [64–66]. This separation helps to avoid inconsistencies between the solutions and stability problems that often arise from forcing the solution of the Parent grid with the updated values of the inner grid [56].

Time synchronization The use of an explicit time integration means that the model needs to verify the CFL stability condition, and the ratio $\Delta t/\Delta x$ must be kept smaller than a given value on the whole grid hierarchy. Consequently, a temporal refinement must be applied in addition to the spatial mesh refinement. The integration algorithm for a time refinement of 3 is depicted in Fig. 4. The model is first integrated on the Parent grid Ω_p with a time step equal to Δt_{p1}, the model is then advanced multiple times on the Child grid Ω_c to reach the same physical time as the outer grid. To synchronize the two solutions, the last time step in the inner grid is imposed: $\Delta t_{c3} = \Delta t_{p1} - \sum \Delta t_{ci}$.

- 1: Model integration on the Parent grid Ω_p
- 2: Model integration on the Child grid Ω_c
- 3: Time and space interpolation of the boundary values
- 4: Update of the Parent Grid in feedback domain

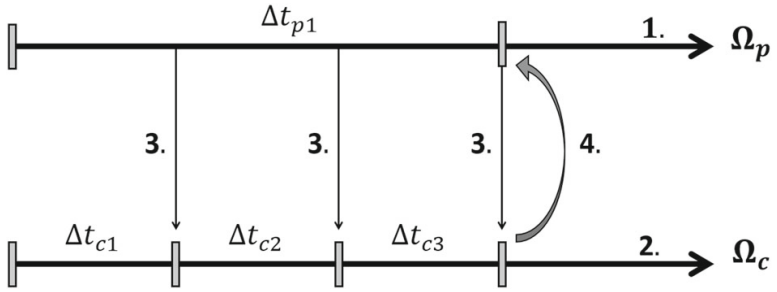


Fig. 4 Schematic illustrating the two-way nesting process

3 Verification

A systematical analysis of a numerical solution for long-wave run-up requires benchmarking. Since the model was developed from scratch and involves a combination of adapted numerical features, it is required to first verify its performance for idealized flow problems, for which analytical solutions have been derived. These tests examine the model's ability to handle important flow processes, such as flow discontinuities and wet/dry transitions. These features are particularly critical for the quality of the computed run-up and can often pose numerical challenges. The implementation of the nested grid approach is then verified with a 2D moving boundaries problem.

3.1 Shock-Capturing Capabilities

Shock-capturing schemes refer to numerical methods that can directly solve wave propagation with large gradients and rapid changes in the free surface and velocity regimes. Such nonlinear phenomena are present in many wave problems (e.g., wave breaking, dambreak wave propagation, and propagation of wet/dry fronts). Consequently, a lot of effort is made to compute shock waves as part of the complete solution [37]. A stable numerical solution for shock waves targets the generation and propagation of an oscillation-free discontinuity without excessive smearing across the shock front.

In the following, we examine the solution of the present model in handling discontinuities and assess the accuracy and quality of the results. Since many shock-capturing flow models are built around Riemann solvers, we compare the solution from the presented scheme, referred to as "Present Scheme", with the solution obtained by a 1D HLLC Riemann solver ("HLLC Scheme"). The HLLC scheme used for comparison was coded based on the techniques given by Toro [37]. For consistency with the presented scheme, the first-order HLLC scheme is extended to second-order accuracy through a MUSCL reconstruction [67] combined with a generalized MinMod limiter and a predictor–corrector Runge–Kutta time integration.

The dambreak problem is a widely used test to demonstrate the shock-capturing capabilities of numerical schemes. We consider a one-dimensional dambreak over a

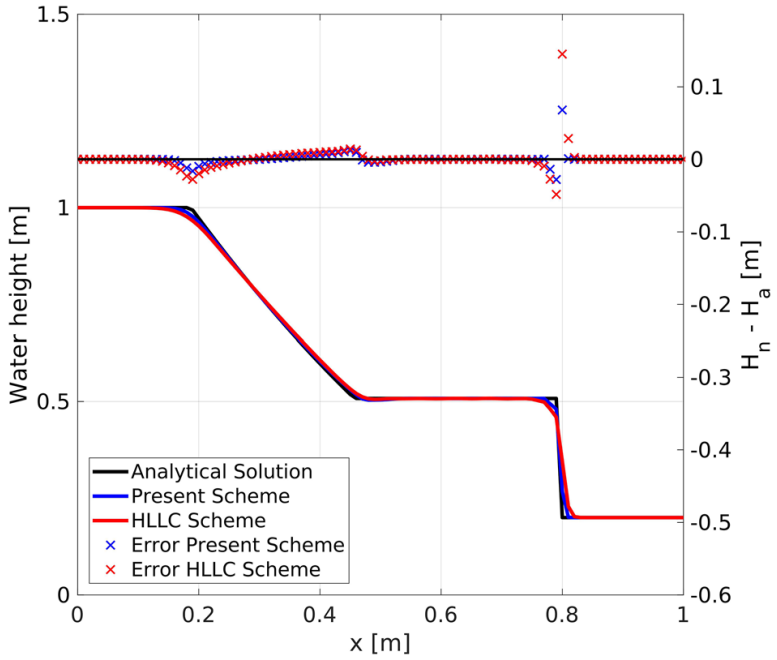


Fig. 5 Dambreak over wet bed: water height profiles from the Present and HLLC schemes at $t = 0.1$ s for a cell size Δx of 1 cm

wet bed of uniform depth. The domain is 1 m long and the initial condition is

$$h\,(x, 0) = \begin{cases} 1\,\text{m} & \text{if } x \leq 0.5\,\text{m} \\ 0.2\,\text{m} & \text{otherwise} \end{cases} \qquad u\,(x, 0) = 0\,\text{m/s}$$

The analytical solution for this test was derived by Stoker [68] and consists of a shock and a rarefaction wave moving in opposite directions from the center of the domain. The solutions of the dambreak test for 100 grid cells ($\Delta x = 1$ cm) and at $t = 0.1$ sec are shown in Fig. 5. For both schemes, we use a constant Courant number of $CN = 0.7$ and a diffusion parameter in the generalized MinMod limiter $\theta = 1.5$.

Both numerical schemes correctly capture the rarefaction and shock waves despite small discrepancies in comparison to the analytical solution. This small mismatch can be reduced significantly with a reduction in grid size. In general, the Present scheme achieves slightly sharper solutions around the flow transitions compared to the HLLC scheme. Consequently, the Present scheme contains smaller L1-norm errors than the HLLC scheme, as listed in Table 1, albeit the fact that both solutions converge towards the exact solution with mesh refinement. The presented model is able to compute the propagation of shocks with the correct wave speed and height, proving its powerful shock-capturing capability without the need for the computationally expensive sampling of the solution as it is necessary for the HLLC scheme.

3.2 Moving Boundaries

An essential feature of shallow-water models used for flood and inundation mapping is the ability to compute wet-dry transitions and track moving boundaries. The biggest

Table 1 Dambreak over a wet bed: L1-norm error

Number of cells	h		hu	
	Present	HLLC	Present	HLLC
100	3.69×10^{-3}	5.20×10^{-3}	6.37×10^{-3}	1.24×10^{-2}
200	1.85×10^{-3}	2.56×10^{-3}	3.17×10^{-3}	6.22×10^{-3}
400	7.90×10^{-4}	1.29×10^{-3}	1.90×10^{-3}	3.22×10^{-3}
800	4.44×10^{-4}	6.36×10^{-4}	7.76×10^{-4}	1.52×10^{-3}

challenges are associated with the definition of the numerical fluxes and source terms in the presence of dry cells. A clean and stable representation of the moving boundary is essential for the correct description of run-up and inundation limits independent of the previous stages of wave propagation and breaking.

We investigate the performance of the present model in describing fast sheet flows induced by a dambreak over a dry bed with and without frictional resistance. This test is also used to verify the implementation of the friction term.

The test case involves a 2000 m long horizontal channel of uniform depth with Δx = 5 m grid spacing and the following initial condition:

$$h\,(x,0) = \begin{cases} 6\,\mathrm{m} & \text{if } x \leq 1000\,\mathrm{m} \\ 0\,\mathrm{m} & \text{otherwise} \end{cases} \qquad u\,(x,0) = 0\,\mathrm{m}$$

The test is computed with a minimum water depth of $h_{\min} = 10^{-8}$ m and a Courant number of 0.7. Two cases are taken into account:

1. Dambreak without friction: The numerical results are compared with the Ritter solution. The solution involves a wet-dry front propagating downstream and a rarefaction wave moving upstream into the reservoir.
2. Dambreak with friction: In this case, the Darcy–Weisbach friction law with a coefficient $f = 8g/40^2$ is utilized in the friction source term of the momentum equations. The reference solution is based on the Dressler/Whitham/Chanson conceptual model [69–71], which is based on the assumption that near the wavefront, frictional resistance controls the fluid motion. The exact shape of the wavefront can be found in [71]. In contrast to the process at the downstream wave, the frictional resistance in the rarefaction regime is neglected, and the solution at the front can be described by a modified Ritter's solution as presented in Delestre et al. [72].

In both cases, good agreement between the reference and the numerical solutions is obtained (Fig. 6). In the case of bottom friction, the model accurately captures the deceleration of the wavefront, which verifies its capability of correctly handling bottom roughness. As before, the results can be improved through mesh refinement but not through further reduction of the predefined minimum water depth $h_{\min}$.

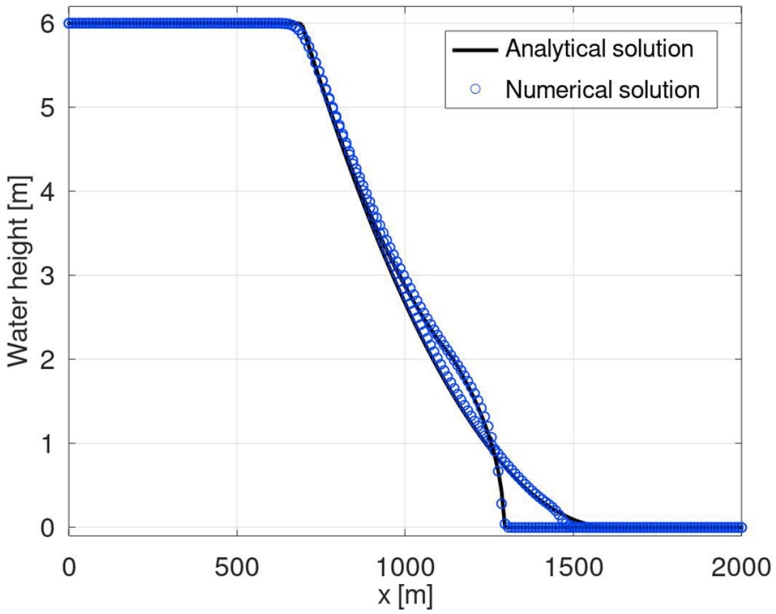


Fig. 6 Dambreak on dry bed: water height profiles after $t = 40$ s for grid spacing $\Delta x = 10$m. The wave front around 1250 m corresponds to the solution with a friction coefficient of $f = 8g/40^2$

3.3 Nested Grid Implementation

In this section, we examine the accuracy of the nested grid implementation. This step is important to scrutinize the model performance with respect to the information exchange across different grid resolutions, especially in the presence of wet/dry transitions. For applications related to long-wave run-up, the nested grid approach is expected to deal with moving boundaries and fast flows over varying topography in two-dimensional settings. A few analytical solutions of the SWE exist for problems in the 2D horizontal plane. The oscillation in a parabolic basin is one of them, as it addresses a two-dimensional run-up problem, which helps examine the validity of the numerical structure in the combined xy-directions.

The water oscillation is induced inside a $[0, L] \times [0, L]$ parabolic basin given by

$$z(r) = -h_0 \left(1 - \frac{r^2}{a^2}\right) \quad \text{where} \quad r = \sqrt{\left(x - \frac{L}{2}\right)^2 + \left(y - \frac{L}{2}\right)^2}$$

The value of h_0 represents the still-water depth at the basin center, and a is the radius of the wetted perimeter. The exact solution for this test was derived by Thacker [4]. For a smooth bed with no friction, the analytical solution for the water depth is described as

$$h(r, t) = h_0 \left(\frac{\sqrt{1 - A^2}}{1 - A\cos(\omega t)} - 1 - \frac{r^2}{a^2} \left(\frac{1 - A^2}{(1 - A\cos(\omega t))^2} - 1 \right) \right) - z(r)$$

where $\omega = \sqrt{8gh_0/a}$ is the frequency of the oscillation, and the coefficient $A = \left(a^2 - r_0^2\right) / \left(a^2 + r_0^2\right)$ with r_0 the radius of the initial shoreline. For the setup of the

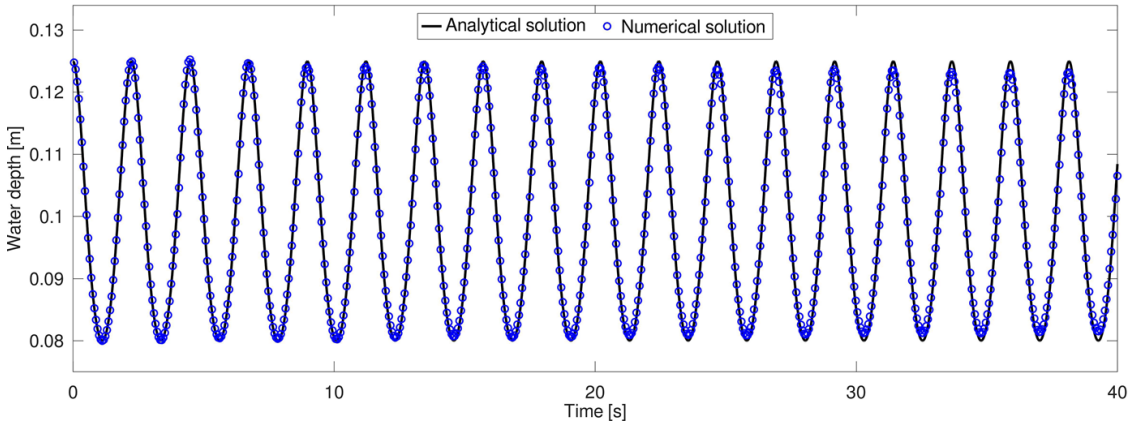


Fig. 7 Time series of water height at center point of parabolic basin ($x = L/2$, $y = L/2$). Grid spacing $\Delta x = \Delta y = 2$ cm. The numerical solution is of low diffusion without requiring excessively fine mesh sizes

dimensions of the parabola and the initial condition of the free surface, we use $a = 1$ m, $r_0 = 0.8$ m, $h_0 = 0.1$ m and $L = 4$ m.

The analytical solution is used to verify the symmetry and accuracy of the nested grid implementation. Here, we place a nested domain off-center, including the moving waterline, with a refinement factor of 4. The use of an off-center nested grid is critical to verifying the grid exchange for both the normal and cross fluxes. The Parent grid is computed with a quadratic cell size of 2 cm by 2 cm. The inner Child grid is computed with $\Delta x = \Delta y = 0.5$ cm.

The water height evolution at the center of the basin is shown in Fig. 7 after 40 s corresponding to over 17 full cycles in the Parent grid. The present solution convinces through the maintenance of amplitude and phase over multiple oscillation cycles the quality of the second-order numerical scheme. These results confirm not only the low numerical diffusion but also the smooth transition across the wet/dry boundary inherent to the model without the need for excessively small grid sizes.

Figure 8 depicts the free surface transect across the basin center line at several stages, $t = T$, $t = T + T/4$, and $t = T + T/2$, where $T = 2\pi/\omega$ denotes the oscillation period. The run-up is well described, and no numerical artifact arises from the exchange between the Parent and Child grid. In addition, Fig. 9 gives a visual impression of the three-dimensional problem and showcases that the definition of the run-up outline benefits from mesh refinement.

4 Effect of Grid Nesting on Wave Run-Up

Previous verification efforts have ensured that the present model correctly handles the fundamental features that are essential for the accurate computation of long-wave run-up. The following tests examine the sensitivity of the computed results to grid nesting for efficient computation of local run-up problems. For this purpose, we utilize two standard experimental benchmark tests that have been widely used in the tsunami community and that highlight the complexity of the local long-wave run-up. The two tests present common long-wave features such as the increase in local wave run-up

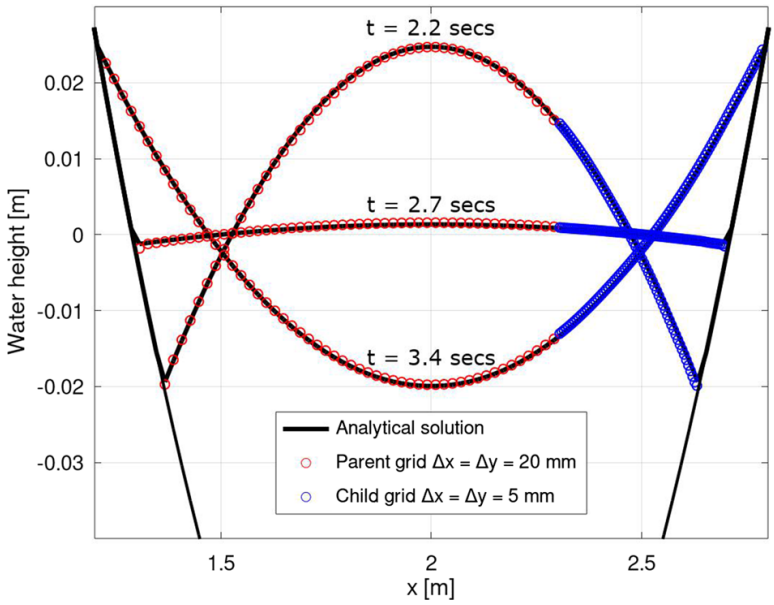


Fig. 8 Oscillation in a parabolic basin: cross-section of water height after one full oscillation. Each circle represents the solution from the numerical model at each grid cell across the transect. The solution from the nested grid is indicated with blue circles

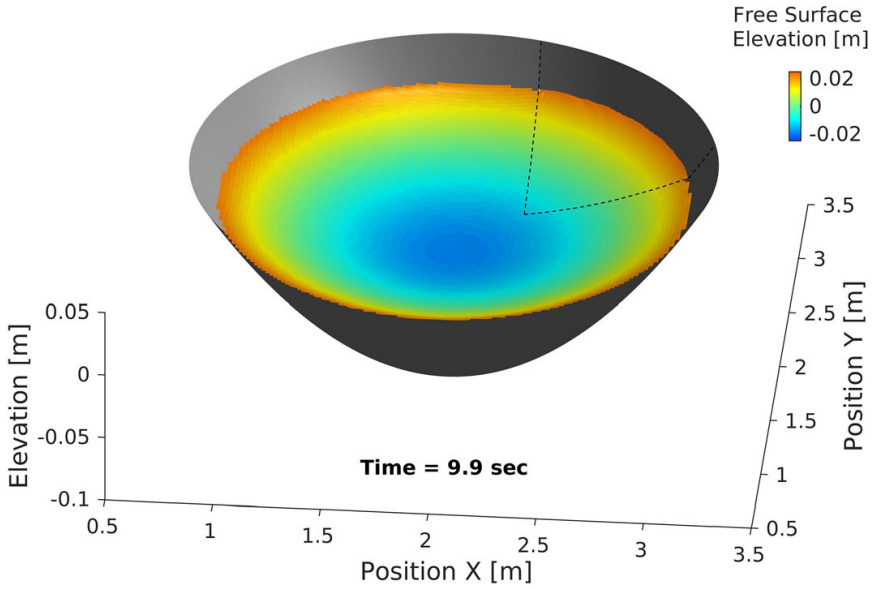


Fig. 9 Free surface elevation of oscillation in parabolic basin with grid nesting after 4.5 cycles (9.9 s). The refined Child grid is denoted by the dashed line and shows a more detailed run-up limit than the coarse Parent grid

from the collision of two or more waves as well as extreme run-up over highly detailed terrain. We will present an analysis of the sensitivity of the computed run-up to the general mesh size and further investigate the sensitivity of the maximum run-up extent to the size of the nested grid and the refinement factor.

4.1 Solitary Wave Run-Up Around a Conical Island

The transformation of long waves around islands has attracted a lot of attention in the past—especially among tsunami researchers. A common observation is that long waves can refract and diffract around an island from both sides and collide in the back. In some cases, the maximum run-up occurs counter-intuitively at the island's lee side due to a superposition effect when the refracted/diffracted waves from both sides run into each other and double up. The problem of the conical island is exemplary since the high run-up and inundation at the lee side cannot be approximated with empirical formulae or computationally cheap 1D calculations. Instead, the problems require a full 2D solution that naturally exhibits a substantial computational effort.

Briggs et al. [7] conducted a large-scale laboratory experiment to investigate solitary wave transformation around a conical island. The basin is 25 m by 30 m with a circular island in the shape of a truncated cone constructed of concrete with a diameter of 7.2 m at the bottom and 2.2 m at the top. The island is 0.625 m high and has a side slope of 1:4. A 27.4-m-long directional wavemaker consisting of 61 paddles generated the input solitary waves for three laboratory tests. Wave absorbers at the three remaining sidewalls reduced reflection in the basin. Further details about the laboratory model setup, the location of the wave gauges, and the numerical setup can be found in NTHMP [73].

The present study focuses on experiments with a water depth $h = 0.32$ m and solitary wave heights of $A/h = 0.1$. Consistent with NTHMP benchmark problem 6 [73], our numerical test uses the measured wave heights of $A/h = 0.096$ from the laboratory experiment instead of the target wave heights as they better represent the recorded data and thus the incident wave conditions to the conical island. A reflective

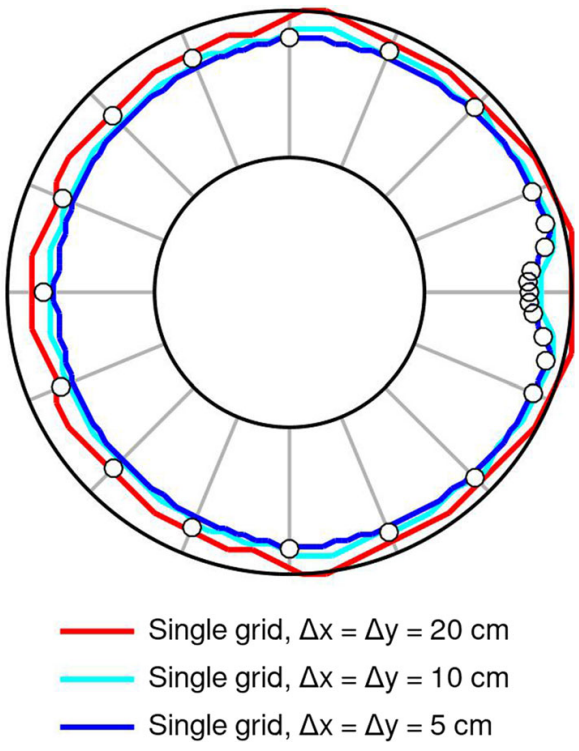


Fig. 10 Grid size sensitivity of maximum run-up outlines for the test with $A/h = 0.096$ of Briggs et al. [7]. Black dots denote experimental data, solid lines represent results from the present model

boundary condition is imposed at the lateral sides. The wave absorbers from the laboratory layout are not considered since their absorbing performance is unknown. The model is set up with a reference grid of $\Delta x = \Delta y = 5$ cm. A Manning roughness coefficient of $n = 0.012$ s/m$^{1/3}$ accounts for the smooth concrete finish according to Chaudhry [74]. The Courant number is set to Cr = 0.5. The model setup is comparable to earlier work and will be used as a reference as it is expected to return solutions of similar and comparable quality to previously published studies. The results from the free surface elevation observed at five wave gauges are omitted here as they are comparable to previously published results.

Sensitivity to grid resolution In view of sensitivity to the grid size, Fig. 10 shows the run-up limits for the reference scenario with $\Delta x = \Delta y = 5$ cm, as well as the solutions of the model with coarser grid sizes of $\Delta x = \Delta y = 10$ cm and $\Delta x = \Delta y = 20$ cm, respectively. The reference mesh size of 5 cm returns the closest agreement overall with the run-up data—particularly at the lee side of the island. Nevertheless, a numerical domain with four times fewer cells, i.e., uniform 10 cm grid spacing, still provides a decent estimate of the run-up, albeit with less precision at the lee side. It is not really surprising that a grid size of $\Delta x = \Delta y = 20$ cm is too coarse to represent the details of the run-up outline, and virtually no run-up is recorded in the lee of the island.

Careful examination of the temporal evolution of the wave field at $\Delta x = \Delta y = 5$ cm resolution shows that the colliding waves in the back of the island locally and momentarily augment the water level, but then pass through each other and continue the refraction/diffraction process. The locally high run-up in the back of the island results to a great extent from the two waves that shoal and spill up on either flank of the leeward side. It often goes unnoticed that the steepened refracted waves then meet head-on over the leeward topography, i.e., the initially dry beach, from where a substantial portion of maximum run-up and inundation originates. For a relatively steep slope, this wrapping process requires rather fine resolution to properly account for the flooding process, and insufficient grid cells over the beach can lead to an under-representation of the run-up.

Sensitivity to grid nesting It is understood that any reduction in the total cell count will reduce the computational load. A nested grid approach caters to lowering the computational effort without compromising too much on the quality of the results. A question of practical interest is whether the overall wave transformation around the island could potentially be computed over a coarse grid, from which information is fed into a nested inner grid of higher resolution that is placed only over a local area of interest. Figure 10 demonstrates that a grid resolution of 5 cm is an adequate choice for the resolution of the wave run-up along the beach of the conical island and that coarser mesh sizes, in particular the 20 cm resolution, are insufficient to resolve most of the run-up.

The solitary input wave has a length of several meters. As shown in the previous benchmark tests, e.g. Sect. 3.3, the present model computes long waves with minimal numerical diffusion and hence is expected to handle the general processes of the solitary wave transformation around the island even over a rather coarse mesh. Inspection of the full free surface evolution has shown that even a grid of 20 cm mesh size can account for the overall wave processes in the vicinity of the conical island and that it only fails in computing the detailed run-up.

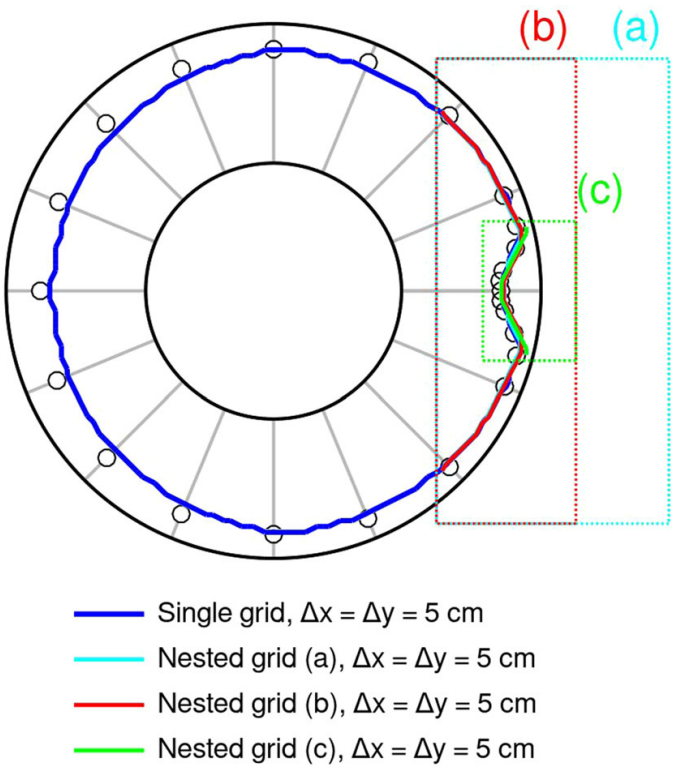


Fig. 11 Maximum run-up outlines for the test with $A/h = 0.096$ of Briggs et al. [7]. Each model run uses a Parent grid with $\Delta x = \Delta y = 20$ cm resolution and one Child grid of $\Delta x = \Delta y = 5$ cm. The three individual nested grid set-ups (a), (b), and (c) and their corresponding maximum run-up limits are color-coded and denoted by the dashed rectangle and the solid lines within

Figure 11 shows the results from three nested grid approaches—all with an inner grid of 5 cm resolution placed into an outer grid of 20 cm resolution. The individual inner domains are color-coded and of 2 m by 4 m, 1.2 m by 4 m and 0.8 m by 1.2 m. The additional computational load arising from the inner grid is associated with 3200, 1920, or 384 cells, respectively. It can be seen that the run-up outline in the nested grid (a) denoted by the cyan line in Fig. 11 is nearly identical to the outline of the uniform 5 cm reference grid. This implies that the overall wave processes are sufficiently resolved by the coarse outer grid up to the boundary of the nested grid (a), which subsequently takes care of the detailed wave transformation and run-up processes at the back side of the conical island. The domain size of the nested grid is then reduced behind the island, as illustrated by the red dashed rectangle. The corresponding run-up limit (red line) remains nearly identical to the run-up outline from the largest nested grid setup. The run-up at the lee side, therefore, depends only minimally on the higher resolution in the area behind the island where the wave collision process occurs. The nested grid extends to a very small area just around the hotspot of run-up, as denoted by the green dashed rectangle. Surprisingly, the run-up along the center lee side remains qualitatively very similar to the run-up computed by the larger nested grids.

The effect of the nested grid approaches can be seen in Fig. 12 in more detail. Row 1 shows the free surface evolution over the 5 cm uniform reference grid. The corresponding alternate solutions, denoted by the black dashed rectangles, illustrate the nested grid solutions. As the wave is moving around the island, the nested grid (a) (second row) picks up its energy and resolves the wrap of the run-up tongue in detail, though with slightly less steepness at the leading edge compared to the reference solution. The maximum run-up after 9.2 s in the nested grid is nearly identical to the uniform reference solution. The third row shows the free surface elevation from

the red rectangle and the run-up limit from Fig. 11. The high-resolution inner grid extends only marginally over the bathymetry behind the island and mostly covers the topography. The detailed solution of the colliding waves behind the island is less critical for the maximum run-up than a high-resolution computation of the two refracted run-up tongues that meet each other over the dry slope. The last row shows that a representative run-up limit is achievable even by only using an extremely small inner grid of high resolution at the location where the refracted waves collide over the beach.

The long-wave refraction and collision processes do not necessarily require high grid resolution given that a low-diffusive coarse solution captures the main energy flux. Counter-intuitively, the locally high run-up of long waves, as illustrated in this example, is often driven by wave processes in the immediate vicinity of the shoreline and over the beach. Accurate run-up results can potentially be obtained with locally very small nested grids as long as they cover the entire run-up zone over the beach. This is particularly true for locations with steep beach slopes.

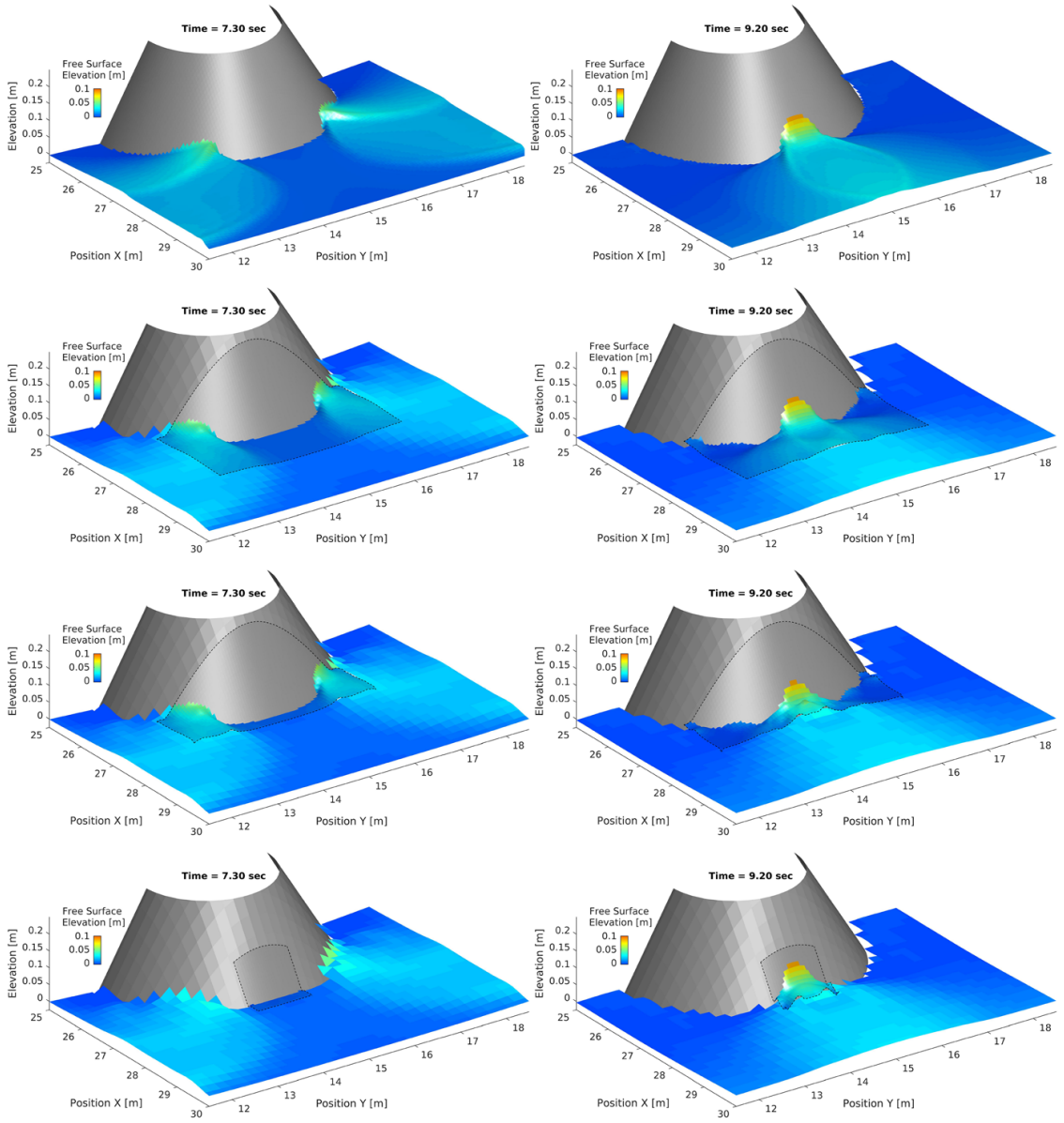
As for the results from Figs. 10, 11 and 12, the computed wave field is symmetric to machine precision with respect to the horizontal center line at 15 m in the y-direction. This supports the quality of the numerical results as any instability arising from the interface at the boundary of the nested grids would have eliminated the perfect symmetry.

4.2 Long-Wave Run-Up at Monai Valley

The second benchmark is testing the sensitivity of the present model to the mesh size and refinement of the solution with a nested grid for the computation of non-linear wave processes over an irregular terrain that favors extreme run-up. The 1993 Hokkaido Nansei-Oki tsunami is a well-studied event thanks to the laboratory experiments conducted by Matsuyama and Tanaka [75] at the Central Research Institute for Electric Power Industry (CRIEPI) in Japan. The down-scaled laboratory test examined the extreme run-up of over 30 m at Monai Valley, located between two headlands and sheltered by the small Muen Island. The area around Monai Valley was reconstructed with a plywood model at 1:400 scale based on bathymetric and topographic data as shown in Fig. 13.

A wave gauge near the wavemaker recorded the initial low amplitude N-wave used in the present numerical model as boundary input with the free surface elevation interpolated from the data according to the model time step. As in the previous test, we first examine the sensitivity of the numerical solution to the grid size over a single domain with uniform resolution. Again, the Courant number is kept constant at $Cr = 0.5$. A Manning coefficient of $n = 0.012\ sm^{-1/3}$ accounts for the surface roughness of the plywood model [74].

Sensitivity to grid resolution Figure 14 shows the comparison between the computed and recorded data at the wave gauges placed in the numerical and experimental setup between Muen Island and Monai Valley. The computed results are of similar quality as the solutions from previous studies. The wave regime at the locations of the gauges is still reasonably well resolved with a rather coarse mesh. Even with a 10 cm grid size,



(a) Solution after 7.3 sec **(b)** Solution after 9.2 sec

Fig. 12 Free surface elevation at lee side of conical island computed over the single reference grid of $\Delta x = \Delta y = 5$ cm (first row) and with three separate grid nesting approaches each combining a coarse outer grid of $\Delta x = \Delta y = 20$ cm mesh size with inner fine grids of $\Delta x = \Delta y = 5$ cm resolution (2nd, 3rd, 4th row). First column: refraction/diffraction of solitary wave around flank of conical island. Second column: maximum run-up from superposition of refracted/diffracted waves. The extent of the nested grid is outlined by the black dashed line in row 2 to 4

the general shape of the free surface time series is captured, and the overall energy of the wave field behind Muen island is accounted for.

Figure 15A illustrates the sensitivity of the computed maximum run-up to different uniform grid sizes of 1.25 cm, 2.5 cm, 5 cm, and 10 cm. The local run-up in Monai Valley is more sensitive to the grid resolution than the nearshore wave field in front of the beach. Since the terrain is steep and narrow, the computations with the present model show that a rather fine grid of 1.25 cm is necessary to obtain a proper outline of the run-up envelope. NTHMP [73] confirms that most previous numerical studies

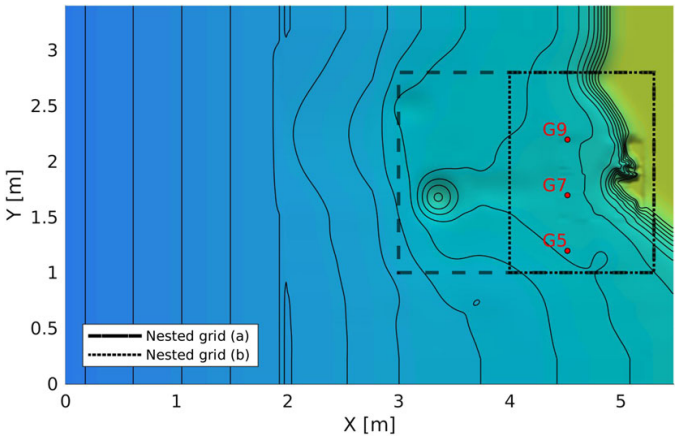


Fig. 13 Outline of the bathymetry from the 1:400 scaled model used by Matsuyama and Tanaka [75]. The black dashed and dotted lines denotes the boundaries of two individual inner nested grids (a) and (b)

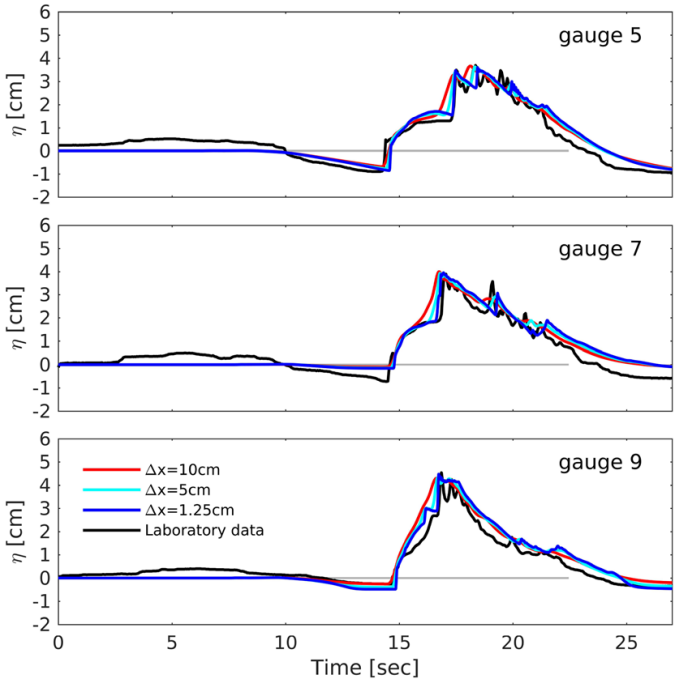


Fig. 14 Free surface time series at the gauges shown in the left panel from computations over the entire domain with different uniform mesh sizes

utilized a mesh size of $\Delta x = \Delta y < 1.5$ cm to obtain a consistent definition of the wave run-up in the narrow and steep valley. The fine grid of 1.25 cm in the second row of Fig. 16 resolves the details of wave refraction and collision in front of the steep cliff, whereas a coarser option of 10 cm resolves neither the flow details nor the small-scale flow features over the topography and consequently leads to a significant underestimation of the run-up in the Monai Valley.

Sensitivity to grid nesting Similar to the previous benchmark test, the question arises whether it is possible to utilize a coarse mesh for the overall flow field in combination with a fine nested grid for the detailed run-up in an area of interest like Monai Valley. Knowing that the run-up over terrain with irregular and steep slopes requires small grid sizes, we utilize a 1.25 cm nested grid (a) inside a Parent grid as outlined in Fig. 13. The inner nested grid starts offshore of Muen island, similar to what Yamazaki et al. [52] have used. The Parent grid is of 5.5 m by 3.4 m size. It contains only 1870 cells with a 10 cm resolution. The two nested grid options (a) and (b) have dimensions of 2.3 m by 1.8 m and 1.3 m by 2.3 m and consequently add 26,496 or 14,976 grid

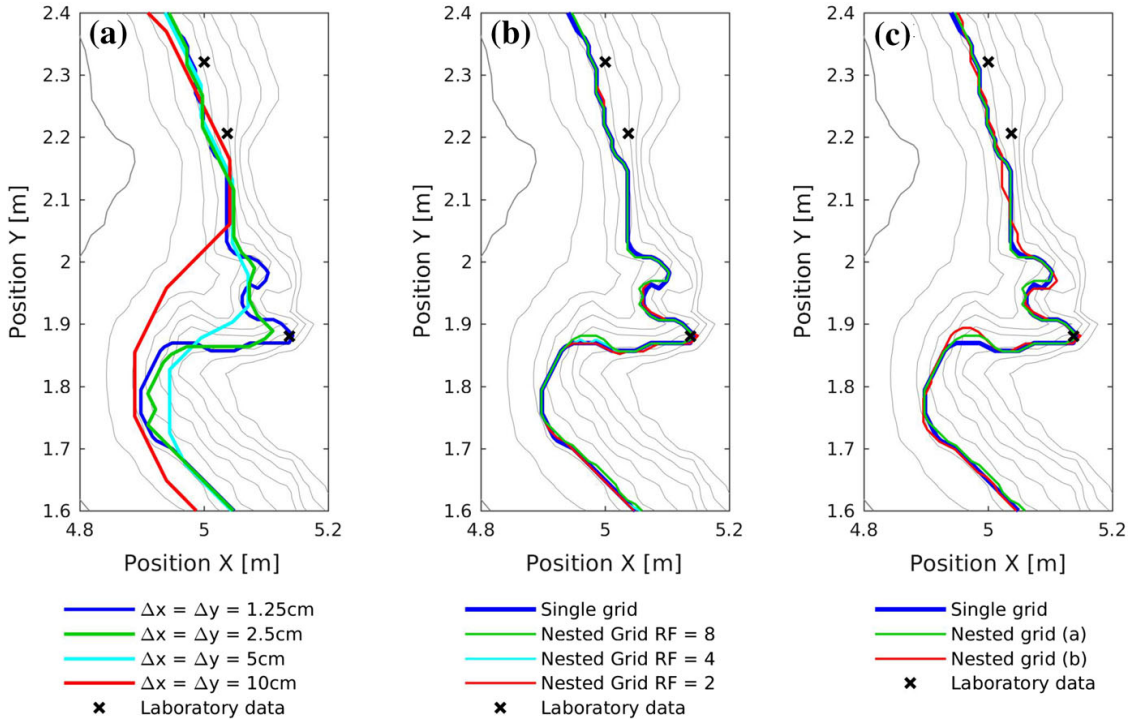
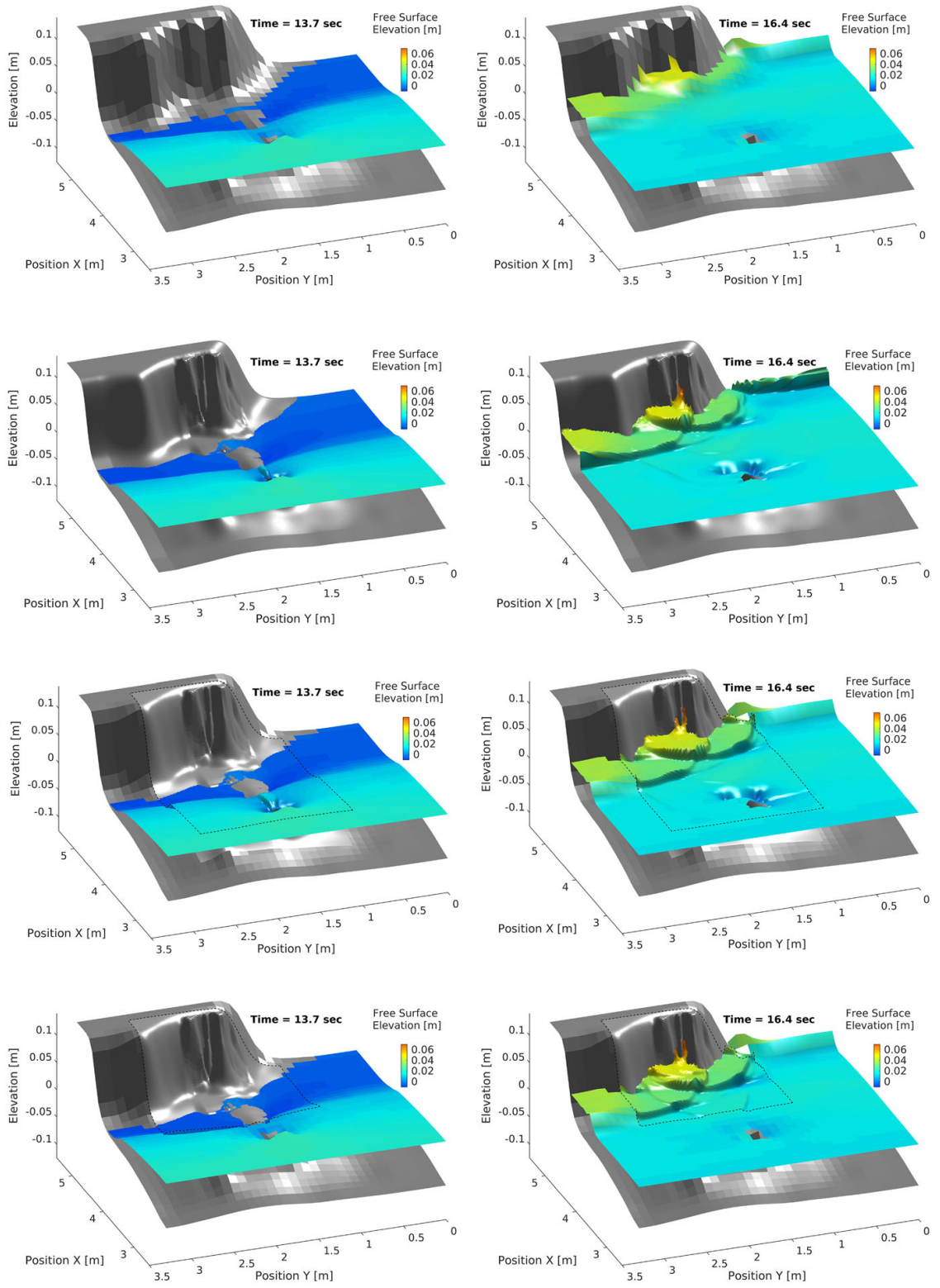


Fig. 15 Maximum run-up limits around Monai Valley. Left **A** Uniform grid with different resolutions. Center **B** Nested grid (a) with 1.25 cm resolution and different Parent grid resolutions of 2.5 cm, 5 cm, and 10 cm leading to refinement factors of RF = 2, RF = 4, and RF = 8. Right **C** Nested grid (a) and (b) with 1.25 cm resolution and Parent grid resolution 10 cm

cells, respectively, to the computation. Hence, the two nested grid options reduce the total cell count by 76% and 86% in comparison to a single grid of uniform 1.25 cm resolution with 119,680 cells.

The sensitivity of the results with respect to the refinement factor is analyzed by increasing the Parent grid resolution by factors of 2, 4, and 8 with respect to the nested grid. Consequently, the individual run-up limits of Fig. 15B refer to the results from a 1.25 cm nested grid in combination with different Parent grids of 2.5 cm, 5 cm, and 10 cm. The refinement factor hardly influences the run-up limit with a nested grid domain that covers most of the nearshore area (dashed line of the domain (a) in Fig. 13). Again, a basic requirement for the utilization of a coarse Parent grid is a low diffusivity of the numerical scheme. It is understood that the interpolation in the nesting process between the individual grids can lead to small discrepancies in comparison to a uniform grid with high resolution. This can be seen in Fig. 15. The grid nesting strategy should, therefore, always be seen as method to primarily reduce the computational load by still retaining an acceptable quality of the solution.

It is finally shown how the computed results are sensitive to the nested domain size. This is analyzed through reduction of the area covered by the nested grid (see dotted line (b) in Fig. 13). The resolutions of the Parent and Child grid are identical to the setup with nested grid (a). The two scenarios only differ in the domain size of the nested grids. Figure 15C highlights that the run-up limit from the two scenarios varies only at some locations. Though the flow details of the overtopping and refraction processes around Muen island are resolved in detail with a fine grid as shown in rows two and



(a) Solution after 13.7 sec **(b)** Solution after 16.4 sec

Fig. 16 Free surface elevation in front of Monai Valley with single Parent grid of $\Delta x = \Delta y = 10$ cm (first row) and $\Delta x = \Delta y = 1.25$ cm (second row) mesh size. Results from embedded nested Child grids **a** and **b** of $\Delta x = \Delta y = 1.25$ cm in a Parent grid of $\Delta x = \Delta y = 10$ cm are shown in the second row and third row. The extent of the respective Child grid is outlined by the black dashed lines. First column: drawdown from leading depression of N-wave and approaching wave crest upstream of Muen island. Second column: maximum run-up from superposition of refracted and reflected waves

three of Fig. 16, they do not have a substantial influence on the run-up. It is sufficient that the outer grid resolves the overall wave energy and the inner grid accounts for the run-up process.

5 Conclusions and Perspectives

We have shown the performance of a newly developed model for long-wave run-up with respect to standard analytical solutions and laboratory experiments. The model was demonstrated to be shock-capturing, well-balanced, and water-depth positivity preserving, which are crucial properties for the correct estimation of long-wave-driven run-up. The model was proven to be stable and efficient in dealing with wet/dry transitions without the need for computationally expensive treatment of the moving boundary. The numerical scheme is based on a finite-volume staggered approximation with second-order accuracy in space and time. The accuracy in time arises from a combination of the Runge–Kutta method for the convective acceleration and the Leapfrog method for the surface gradient and friction terms. Similarly, the spatial accuracy comes from a second-order upwinded advection along with a second-order central difference scheme for the remaining terms.

The model performs consistently for shock-driven problems and compares to established Riemann solver-based TVD methods. The wet-dry interface is stable and well-defined without the need for additional treatment of the moving boundary. The model contains a two-way grid nesting scheme that allows for local refinement of the solution. The implementation has been verified and proven to be accurate and stable for moving boundaries and was shown to be applicable to long-wave run-up problems.

The performance and sensitivity of long-wave run-up was then investigated in dependence of the nested grid's domain size and the level of its refinement. Two standard benchmark tests from the tsunami community were chosen for the investigation. Though there are no universal rules for the size, position, and refinement factor of nested grids, our results from the two benchmark tests reveal that computations of high quality can be achieved with small nested grids placed strategically at a location of interest such as in areas where locally high run-up occurs. The refinement factor was found to have only small influence on the run-up limit, if the solution in the Parent grid is representative of the wave envelope and the grid nesting method accounts for the correct exchange of the total wave energy flux.

Further, it was demonstrated that it is possible to place a nested grid rather close to the initial still-water level as long as the long-wave flow regime prevails across nested grid's offshore boundary. Long-wave run-up is often more subject to the resolution of the local topography than it is influenced by the detailed wave processes over the bathymetry. This is line with commonly used empirical run-up formulae for swell waves where the maximum run-up envelope is controlled by the overall wave energy and the slope.

The quality of the computed results encourages to expand the development of the model with respect to frequency dispersion. This will allow for a further investigation of how grid nesting can affect the run-up from swell waves. In the same context, the model can be optimized through implementation of massive parallelization techniques

commonly used to reduce the computation time associated with large flow problems. It is evident that a low overall cell count reduces the model's computation time and that the insights gained from the present study can be used to efficiently decrease the computational load for computations of long waves by retaining accuracy and quality of the solutions.

Acknowledgements The authors acknowledge financial support from the I-SITE program Energy & Environment Solutions (E2S), the Communauté d'Agglomération Pays Basque (CAPB), and the Communauté Région Nouvelle Aquitaine (CRNA) for the chair position HPC-Waves. Additional support comes from the European Union's Horizon 2020 research and innovation programme under grant agreement No 883553.

Author Contributions Conceptualization: [F-ZM, VR]; methodology: [F-ZM, VR]; software: [F-ZM]; formal analysis: [F-ZM]; validation: [F-ZM, VR]; visualization: [F-ZM, VR]; writing—original draft: [F-ZM, VR]; writing—review and editing: [F-ZM, VR, DM]; project administration: [VR]; funding acquisition: [VR]; supervision: [VR, DM].

Data Availability The routines for replication of the numerical results can be provided upon reasonable request.

Declarations

Conflict of Interest The authors declare no conflict of interest. The funding agencies had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Liu, P.L.F., Synolakis, C.E., Yeh, H.H.: Report on the international workshop on long-wave run-up. J. Fluid Mech. **229**, 675–688 (1991)
2. Carrier, G.F., Greenspan, H.P.: Water waves of finite amplitude on a sloping beach. J. Fluid Mech. **4**, 97–109 (1958)
3. Synolakis, C.E.: The runup of solitary waves. J. Fluid Mech. **185**, 523–545 (1987)
4. Thacker, W.C.: Some exact solutions to the nonlinear shallow-water wave equations. J. Fluid Mech. **107**, 499–508 (1981)
5. Mayer, R., Kriebel, D.: Wave runup on composite-slope and concave beaches. Coast. Eng. **1995**, 2325–2339 (1994)
6. Hall, J.V., Watts, G.M., et al.: Laboratory Investigation of the Vertical Rise of Solitary Waves on Impermeable Slopes. Army Coastal Engineering Research Center, Washington DC (1953)
7. Briggs, M.J., Synolakis, C.E., Harkins, G.S., Green, D.R.: Laboratory experiments of tsunami runup on a circular island. Pure Appl. Geophys. **144**, 569–593 (1995)
8. Briggs, M.J., Synolakis, C.E., Kanoglu, U., Green, D.R.: Runup of solitary waves on a vertical wall. Long Wave Runup Models: Proceedings of International Workshop, pp. 375–383 (1996)
9. Liu, P.L.F., Woo, S.B., Cho, Y.S.: Computer Programs for Tsunami Propagation and Inundation, vol. 25. Cornell University, Ithaca (1998)
10. Titov, V.V., Synolakis, C.E.: Modeling of breaking and nonbreaking long-wave evolution and runup using VTCS-2. J. Waterw. Port Coast. Ocean Eng. **121**, 308–316 (1995)
11. Brocchini, M., Dodd, N.: Nonlinear shallow water equation modeling for coastal engineering. J. Waterw. Port Coast. Ocean Eng. **134**, 104–120 (2008)
12. Titov, V., Kânoğlu, U., Synolakis, C.: Development of MOST for real-time tsunami forecasting. Ph.D. thesis, American Society of Civil Engineers (2016)
13. George, D.L., LeVeque, R.J.: Finite volume methods and adaptive refinement for global tsunami propagation and local inundation. Science of Tsunami Hazards (2006)
14. Hervouet, J.M.: Hydrodynamics of Free Surface Flows: Modelling with the Finite Element Method, vol. 360. Wiley Online Library, New York (2007)

15. Wei, Z., Dalrymple, R.A., Hérault, A., Bilotta, G., Rustico, E., Yeh, H.: SPH modeling of dynamic impact of tsunami bore on bridge piers. Coast. Eng. **104**, 26–42 (2015)
16. Arakawa, A., Lamb, V.R.: A potential enstrophy and energy conserving scheme for the shallow water equations. Mon. Weather Rev. **109**, 18–36 (1981)
17. Imamura, F.: Tsunami Numerical Simulation with the Staggered Leap-frog Scheme (Numerical code of TUNAMI-N1), School of Civil Engineering. Tohoku University, Asian Inst. Tech. and Disaster Control Research Center (1989)
18. Wang, X.: User Manual for COMCOT Version 1.7 (first draft), vol. 65. Cornel University, Ithaca (2009)
19. Shuto, N., Goto, T.: Numerical simulation of tsunami run-up. Coast. Eng. Jpn. **21**, 13–20 (1978)
20. Titov, V.V., Synolakis, C.E.: Numerical modeling of tidal wave runup. J. Waterw. Port Coast. Ocean Eng. **124**, 157–171 (1998)
21. Liu, P.L.F., Cho, Y.S., Briggs, M.J., Kanoglu, U., Synolakis, C.E.: Runup of solitary waves on a circular island. J. Fluid Mech. **302**, 259–285 (1995)
22. Wei, Y., Mao, X.Z., Cheung, K.F.: Well-balanced finite-volume model for long-wave runup. J. Waterw. Port Coast. Ocean Eng. **132**, 114–124 (2006)
23. Olabarrieta, M., Medina, R., Gonzalez, M., Otero, L.: C3: a finite volume-finite difference hybrid model for tsunami propagation and runup. Comput. Geosci. **37**, 1003–1014 (2011)
24. Godunov, S.: Different Methods for Shock Waves. Moscow State University, Moscow (1954)
25. Roe, P.L.: Characteristic-based schemes for the Euler equations. Ann. Rev. Fluid Mech. **18**, 337–365 (1986)
26. Berger, M.J., George, D.L., LeVeque, R.J., Mandli, K.T.: The GeoClaw software for depth-averaged flows with adaptive refinement. Adv. Water Resour. **34**, 1195–1206 (2011)
27. Macías, J., Castro, M.J., Ortega, S., Escalante, C., González-Vida, J.M.: Performance benchmarking of tsunami-HySEA model for NTHMP's inundation mapping activities. Pure Appl. Geophys. **174**, 3147–3183 (2017)
28. Dutykh, D., Poncet, R., Dias, F.: The VOLNA code for the numerical modeling of tsunami waves: generation, propagation and inundation. Eur. J. Mech. B Fluids **30**, 598–615 (2011)
29. Yuan, Y., Shi, F., Kirby, J.T., Yu, F.: FUNWAVE-GPU: multiple-GPU acceleration of a Boussinesq-type wave model. J. Adv. Model. Earth Syst. **12**, e2019MS001957 (2020)
30. Roe, P.L.: Approximate Riemann solvers, parameter vectors, and difference schemes. J. Comput. Phys. **135**, 250–258 (1997)
31. Harten, A., Lax, P.D., Leer, Bv.: On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. SIAM Rev. **25**, 35–61 (1983)
32. Toro, E.: A weighted average flux method for hyperbolic conservation laws. Proc. R. Soc. Lond. Math. Phys. Sci. **423**, 401–418 (1989)
33. Zijlema, M.: The role of the Rankine–Hugoniot relations in staggered finite difference schemes for the shallow water equations. Comput. Fluids **192**, 104274 (2019)
34. LeVeque, R.J.: Balancing source terms and flux gradients in high-resolution Godunov methods: the quasi-steady wave-propagation algorithm. J. Comput. Phys. **146**, 346–365 (1998)
35. Zhou, J.G., Causon, D.M., Mingham, C.G., Ingram, D.M.: The surface gradient method for the treatment of source terms in the shallow-water equations. J. Comput. Phys. **168**, 1–25 (2001)
36. Brufau, P., Vázquez-Cendón, M., García-Navarro, P.: A numerical model for the flooding and drying of irregular domains. Int. J. Numer. Methods Fluids **39**, 247–275 (2002)
37. Toro, E.F.: Shock-Capturing Methods for Free-Surface Shallow Flows, vol. 868. Wiley, New York (2001)
38. Audusse, E., Chalons, C., Ung, P.: A simple well-balanced and positive numerical scheme for the shallow-water system. Commun. Math. Sci. **13**, 1317–1332 (2015)
39. Dodd, N.: Numerical model of wave run-up, overtopping, and regeneration. J. Waterw. Port Coast. Ocean Eng. **124**, 73–81 (1998)
40. Audusse, E., Bouchut, F., Bristeau, M.O., Klein, R., Perthame, Bt.: A fast and stable well-balanced scheme with hydrostatic reconstruction for shallow water flows. SIAM J. Sci. Comput. **25**, 2050–2065 (2004)
41. Shi, F., Kirby, J.T., Harris, J.C., Geiman, J.D., Grilli, S.T.: A high-order adaptive time-stepping TVD solver for Boussinesq modeling of breaking waves and coastal inundation. Ocean Model. **43–44**, 36–51 (2012). https://doi.org/10.1016/j.ocemod.2011.12.004
42. Kim, D.H., Lynett, P.J., Socolofsky, S.A.: A depth-integrated model for weakly dispersive, turbulent, and rotational fluid flows. Ocean Model. **27**, 198–214 (2009)

43. Roeber, V., Cheung, K.F.: Boussinesq-type model for energetic breaking waves in fringing reef environments. Coast. Eng. **70**, 1–20 (2012)
44. Zhou, J., Stansby, P.: 2D shallow water flow model for the hydraulic jump. Int. J. Numer. Methods Fluids **29**, 375–387 (1999)
45. Stelling, G.S., Duinmeijer, S.A.: A staggered conservative scheme for every Froude number in rapidly varied shallow water flows. Int. J. Numer. Methods Fluids **43**, 1329–1354 (2003)
46. Madsen, P.A., Simonsen, H.J., Pan, C.H.: Numerical simulation of tidal bores and hydraulic jumps. Coast. Eng. **52**, 409–433 (2005)
47. Doyen, D., Gunawan, P.H.: An explicit staggered finite volume scheme for the shallow water equations. In: Finite Volumes for Complex Applications VII-Methods and Theoretical Aspects, pp. 227–235. Springer (2014)
48. Yamazaki, Y., Kowalik, Z., Cheung, K.F.: Depth-integrated, non-hydrostatic model for wave breaking and run-up. Int. J. Numer. Methods Fluids **61**, 473–497 (2009)
49. Zijlema, M., Stelling, G., Smit, P.: SWASH: an operational public domain code for simulating wave fields and rapidly varied flows in coastal waters. Coast. Eng. **58**, 992–1012 (2011)
50. Yamazaki, Y.,Cheung, K.F., Kowalik, Z., Lay, T.;,Pawlak, G.:Neowave. Proceedings and results of the 2011 NTHMP model benchmarking workshop, Boulder: US Department of Commerce/NOAA/NTHMP (NOAA Special Report), pp. 239–302 (2012)
51. Roelvink, D., McCall, R., Mehvar, S., Nederhoff, K., Dastgheib, A.: Improving predictions of swash dynamics in XBeach: the role of groupiness and incident-band runup. Coast. Eng. **134**, 103–123 (2018)
52. Yamazaki, Y., Cheung, K.F., Kowalik, Z.: Depth-integrated, non-hydrostatic model with grid nesting for tsunami generation, propagation, and run-up. Int. J. Numer. Methods Fluids **67**, 2081–2107 (2011)
53. Sætra, M.L., Brodtkorb, A.R., Lie, K.A.: Efficient GPU-implementation of adaptive mesh refinement for the shallow-water equations. J. Sci. Comput. **63**, 23–48 (2015)
54. Donat, R., Martí, M.C., Martínez-Gavara, A., Mulet, P.: Well-balanced adaptive mesh refinement for shallow water flows. J. Comput. Phys. **257**, 937–953 (2014)
55. Liang, Q.: A structured but non-uniform Cartesian grid-based model for the shallow water equations. Int. J. Numer. Methods Fluids **66**, 537–554 (2011)
56. Debreu, L., Blayo, E.: Two-way embedding algorithms: a review. Ocean Dyn. **58**, 415–428 (2008)
57. Gottlieb, S., Shu, C.W., Tadmor, E.: Strong stability-preserving high-order time discretization methods. SIAM Rev. **43**, 89–112 (2001)
58. Stelling, G.S.: Boosted robustness of semi-implicit subgrid methods for shallow water flash floods in hills. Comput. Fluids **247**, 105645 (2022)
59. Casulli, V.: Semi-implicit finite difference methods for the two-dimensional shallow water equations. J. Comput. Phys. **86**, 56–74 (1990)
60. Wilders, P., Van Stijn, T.L., Stelling, G., Fokkema, G.: A fully implicit splitting method for accurate tidal computations. Int. J. Numer. Methods Eng. **26**, 2707–2721 (1988)
61. Gunawan, H.P.: Numerical simulation of shallow water equations and related models. Ph.D. thesis, Paris Est (2015)
62. Liu, P.L.F., Cho, Y.S., Yoon, S., Seo, S.: Numerical simulations of the 1960 Chilean tsunami propagation and inundation at Hilo, Hawaii. In: Advances in Natural and Technological Hazards Research pp. 99–115. Springer (1995) Hazards Research pp. 99–115. Springer (1995)
63. Herzfeld, M., Rizwi, F.: A two-way nesting framework for ocean models. Environ. Model. Softw. **117**, 200–213 (2019)
64. Phillips, N.A., Shukla, J.: On the strategy of combining coarse and fine grid meshes in numerical weather prediction. J. Appl. Meteorol. Climatol. **12**, 763–770 (1973)
65. Zhang, D.L., Chang, H.R., Seaman, N.L., Warner, T.T., Fritsch, J.M.: A two-way interactive nesting procedure with variable terrain resolution. Mon. Weather Rev. **114**, 1330–1339 (1986)
66. Oey, L.Y., Chen, P.: A nested-grid ocean model: With application to the simulation of meanders and eddies in the Norwegian Coastal Current. J. Geophys. Res. Oceans **97**, 20063–20086 (1992)
67. Van Leer, B.: Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method. J. Comput. Phys. **32**, 101–136 (1979)
68. Stoker, J.: Water Waves, The Mathematical Theory with Applications. Interscience Publ. Inc., New York (1957)
69. Dressler, R.F.: Hydraulic Resistance Effect upon the Dam-break Functions, vol. 49. National Bureau of Standards, Washington DC (1952)

70. Whitham, G.B.: The effects of hydraulic resistance in the dam-break problem. Proc. R. Soc. Lond. Ser. Math. Phys. Sci. **227**, 399–407 (1955)
71. Chanson, H.: Application of the method of characteristics to the dam break wave problem. J. Hydraul. Res. **47**, 41–49 (2009)
72. Delestre, O., Lucas, C., Ksinant, P.A., Darboux, F., Laguerre, C., Vo, T.N.T., James, F., Cordier, S.: SWASHES: a compilation of shallow water analytic solutions for hydraulic and environmental studies. Int. J. Numer. Methods Fluids **72**, 269–300 (2013)
73. NTHMP, National Tsunami Hazard Mitigation Program.: Proceedings and Results of the 2011 NTHMP Model Benchmarking Workshop, Boulder: U.S. Department of Commerce/NOAA/NTHMP, (NOAA Special Report) pp. 1–436 (2012)
74. Chaudhry, M.H.: Open-channel Flow. Springer Science & Business Media, Berlin (2007)
75. Matsuyama, M., Tanaka, H.: An experimental study of the highest run-up height in the 1993 Hokkaido Nansei-oki earthquake tsunami. National Tsunami Hazard Mitigation Program Review and International Tsunami Symposium (ITS), pp. 879–889 (2001)

2.3 Malpasset Dambreak

As a final test, we examine the applicability of the model to solutions of energetic shallow water flow problems over a large domain by validating the presented model results with measured field data obtained after the Malpasset dambreak event. This test is complementary to the evaluation of the numerical model's efficiency and robustness over very complex terrain with friction. It should be noted that the proposed model was not particularly developed with the intention of computing river and dambreak flows. Instead, the test has been included to demonstrate the consistency of the numerical solutions with other studies that were specifically devoted to river flooding problems.

The Malpasset catastrophe is a rare example of a complete and instantaneous collapse of a vault dam. The dam was built between 1952 and 1954 in the Reyran River in southern France near the Italian border to support irrigation and drinking water supplies. On December 2, 1959, after a period of heavy rainfall that quickly filled the reservoir, the dam suddenly collapsed and released 50 million m^3 of water into a narrow 12 km-long river valley (Hervouet [2003]). After the dam failure, a field survey was conducted by the local police and returned measurements of the maximum water elevation at 9 locations along the river valley. In addition, a physical model was built to 1:400 scale by the Laboratoire National d'Hydraulique (LNH) to examine the arrival time of the dambreak bore at the 9 locations in the valley. The availability of the data has made the Malpasset case a very popular benchmark test for the validation of shallow water models (*e.g.*, Hervouet [2003], Valiani et al. [2002]). Due to the drastic changes in the terrain induced by the dambreak flow, the topography data has been reconstructed by the Institut Géographique National (IGN) based on a map from 1931, which was later digitized by EDF. The available topography covers a 17 km x 9 km domain, where the bottom elevation ranges from 20 m below sea level to 100 m above sea level. Based on this data, the numerical domain is defined with a grid spacing of $\Delta x = \Delta y = 20\,\mathrm{m}$. At its initial state, the still water level of the reservoir upstream of the dam has an elevation of 100 m. The area along the river valley downstream of the dam is assumed to be dry. These conditions lead to a 55 m high wall of water at the dam site. The Courant number is set to $Cr = 0.5$ and the minimum water depth to $h_{min} = 10^{-4}$ m. This helps to deal with the sheet flow along the leading edge, where non-physical values of the velocities can occur. These artifacts can cause excessive reductions in time steps and consequently longer computation times than necessary.

The results from the physical model test suggest a range of Strickler friction coefficients between $30\,\mathrm{m}^{1/3}\mathrm{s}^{-1}$ and $40\,\mathrm{m}^{1/3}\mathrm{s}^{-1}$. The effect of the friction coefficient was analyzed by computing the same test with different Strickler friction coefficients of $K = 30\,\mathrm{m}^{1/3}\mathrm{s}^{-1}$, $K = 35\,\mathrm{m}^{1/3}\mathrm{s}^{-1}$, and $K = 40\,\mathrm{m}^{1/3}\mathrm{s}^{-1}$. The test cases are computed for 4000 s, and the results are shown in Figures 2.1 and 2.2.

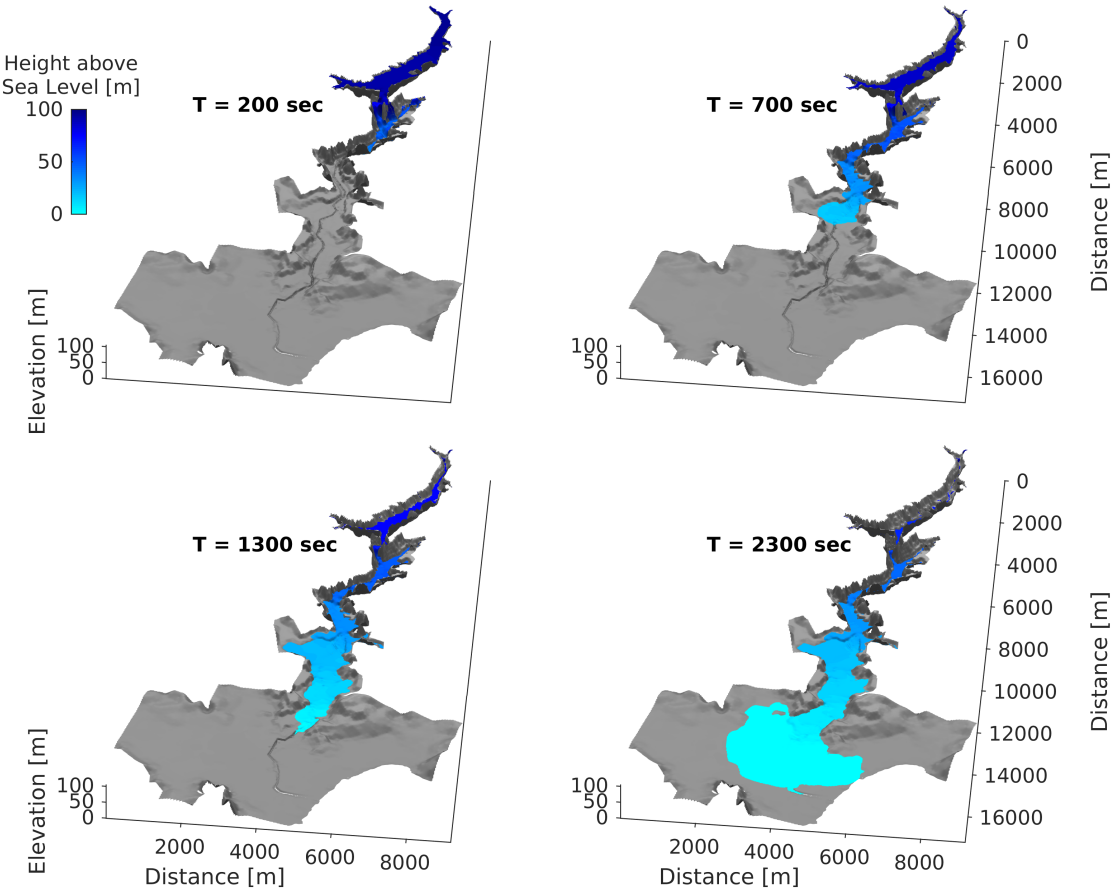


Figure 2.1: Propagation stages of the Malpasset dambreak after 200, 700, 1300, and 2300 sec.

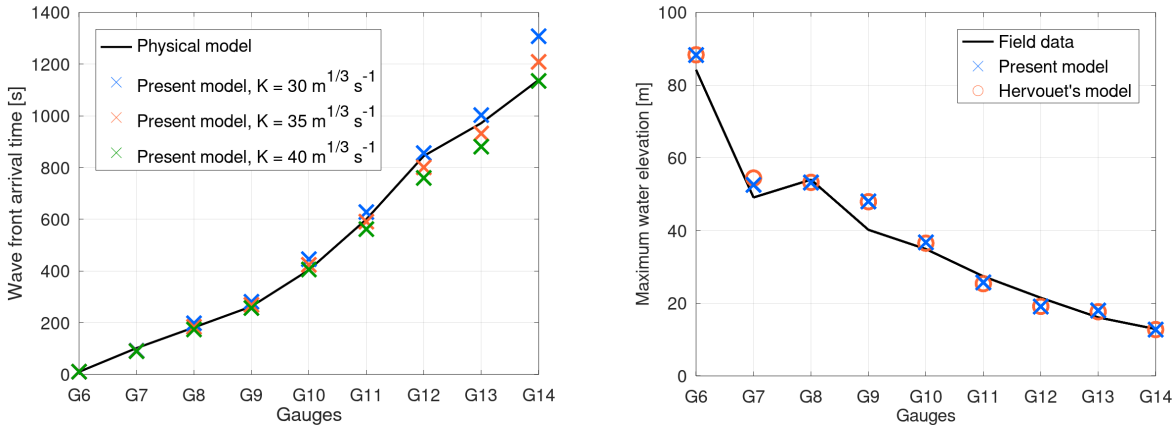


Figure 2.2: Influence of Strickler friction coefficient on the wave front arrival time (left). Maximum water elevation compared with field data and results from Hervouet's model (right)

Summary of findings:

- Figure 2.2 presents the arrival times at the 9 measurement points, labeled G6 to G14, for the three different Strickler coefficients. As expected, the friction coefficient has an impact on the flow speed and consequently leads to slightly different arrival times.
- For $K = 30\,\mathrm{m}^{1/3}\mathrm{s}^{-1}$ the combined results from the computed arrival times and maximum flow elevations show the best overall agreement with the measurements.
- The quality of the results from the present model is comparable to other studies (*e.g.*, Hervouet and Petitjean [1999]), which show a very similar evolution of the flow field and trends in the differences between measured data and computed solutions. The same differences and sensitivity of the numerical solutions have been reported in other studies, *e.g.*, Brodtkorb et al. [2012], Zhang and Wu [2011], and Sætra [2014].

2.4 Conclusions

This section outlines the strategic development of an efficient and robust numerical model for the solution of the SWE. The choice of the staggered grid approach was motivated by its lean structure as well as its robustness when dealing with wet-dry transitions. The numerical solution was proven to be shock-capturing and compares well to long-established techniques such as Riemann solvers for standard tests with flow discontinuities. A series of benchmark tests confirms the accuracy and stability of the model for a variety of hydrostatic flow scenarios. Further, the low diffusivity of the numerical scheme combined with the nested grid technique leads to stable and accurate solutions of run-up at locations of interest in a fraction of the time usually necessary for a comparable solution over a uniform grid. The level of numerical efficiency and consistency of the computed results favor the usage of this approach as a base model and support its extension in multiple directions, such as the inclusion of frequency dispersion terms in the governing equations.

Chapter 3

Boussinesq-type Equations: Nonlinearity and Dispersion

Contents

3.1 Introduction **61**
3.2 SGN 1D Model **62**
3.2.1 Governing Equations 62
3.2.2 Numerical Scheme 63
3.2.3 Verification 66
3.3 Nwogu 1D model **68**
3.3.1 Governing Equations 68
3.3.2 Numerical Scheme 69
3.3.3 Verification 71
3.4 Frequency Dispersion **73**
3.4.1 Linear Dispersion Relation 73
3.4.2 Numerical Tests 75
3.5 Nonlinearity **79**
3.5.1 Numerical Tests 79
3.6 Conclusions **81**

3.1 Introduction

Phase-resolving nearshore wave models are mostly based on Boussinesq-type and non-hydrostatic approximations, which are applicable to intermediate and shallow water regimes. The primary objective of these mathematical formulations is to reduce the three-dimensional flow problem to two dimensions while retaining some of the non-hydrostatic effects of the vertical flow structure. The derivation of the simplified equations will therefore depend on assumptions involving two main parameters:

- **Nonlinearity**: accounts for the deviation of the free surface with respect to the still water level, $\varepsilon = \frac{a}{d}$.
- **Dispersion**: relates the wavelength to the mean water depth, $\mu = \frac{d}{l}$.

Since Peregrine [1967] first proposed the classical Boussinesq equations by retaining the lowest-order effects of nonlinearity and frequency dispersion (*i.e.*, $\mathcal{O}(\varepsilon) = \mathcal{O}(\mu^2)$), considerable efforts have been devoted to improving the range of applicability of the Boussinesq models with respect to small amplitudes and long wave restrictions. Dingemans [1973] included higher-order dispersion and nonlinearity by retaining terms of order $\mathcal{O}(\mu^4)$ and $\mathcal{O}(\varepsilon\mu^2)$, while others (*e.g.*, Serre [1953]; Wei and Kirby [1995]; Madsen and Schäffer [1998]) derived fully-nonlinear models that include all nonlinear terms up to the retained order of dispersion. While introducing high-order nonlinear and dispersive terms into the equations provides more consistent and accurate mathematical models, the numerical solution of these additional terms is rather challenging and prone to instabilities. A different approach to improving the equation properties, notably dispersion, is by manipulating terms that are normally zero to the order of approximation and optimizing these terms to achieve a better match with Airy wave theory regarding dispersion. Madsen and Sørensen [1992] used differential operators on leading order terms to construct enhanced terms that had no physical meaning but did not alter the validity of the equations. The choice of several free parameters was made by considering linear dispersion and shoaling properties. Nwogu [1993], on the other hand, derived a new set of improved Boussinesq-type equations by considering the velocity at an arbitrary elevation z_α as the reference velocity (see Appendix A). Both approaches yield a dispersion relation that is Padé [2, 2] to the exact Airy theory and thus improving the applicability of the equations to deeper water.

The choice of a mathematical framework for our new dispersive nearshore wave model is a challenging one. While a fully-dispersive and fully-nonlinear model would be the obvious choice from an accuracy and completeness point of view, the complex numerical solution required by such a model would undermine its speed and robustness, and thus its applicability to operational settings. Therefore, a balance between accuracy and computational cost has to be achieved in order to efficiently solve wave processes from intermediate water to the swash zone. In the context of coastal wave modeling, two sets of equations have been particularly attractive for the computation of nearshore waves: the standard Nwogu's equations due to their enhanced dispersion and the Serre-Green-Naghdi equations thanks to their fully nonlinear properties. The derivation of the two equations is detailed in Appendix A. While both equations retain terms to $\mathcal{O}(\mu^2)$, the SGN equation simplifies the system by setting the reference velocity as the depth-averaged velocity, yielding a fully nonlinear model. Nwogu's equation, on the other hand, uses

the velocity at z_α as the reference velocity to improve the dispersion, and the system is simplified by adopting the weakly nonlinear assumption. In this section, we solve the two sets of equations (*i.e.*, SGN and Nwogu) in 1D using the same numerical scheme and examine the effects of dispersive and nonlinear properties on the quality of the solution.

3.2 SGN 1D Model

3.2.1 Governing Equations

The SGN equations are commonly used in the modeling of fully nonlinear, weakly dispersive waves. We consider the following conservative form of SGN equations proposed by Khakimzyanov et al. [2017]. The equations describe the incompressible fluid flow in a layer bounded from below by the impermeable steady bottom $y = -h(x)$ and the free surface $y = \eta(x,t)$:

$$H_t + \left[Hu\right]_x = 0, \tag{3.1}$$

$$\left[Hu\right]_t + \left[Hu^2\right]_x + gH\eta_x = \wp_x - \varrho h_x. \tag{3.2}$$

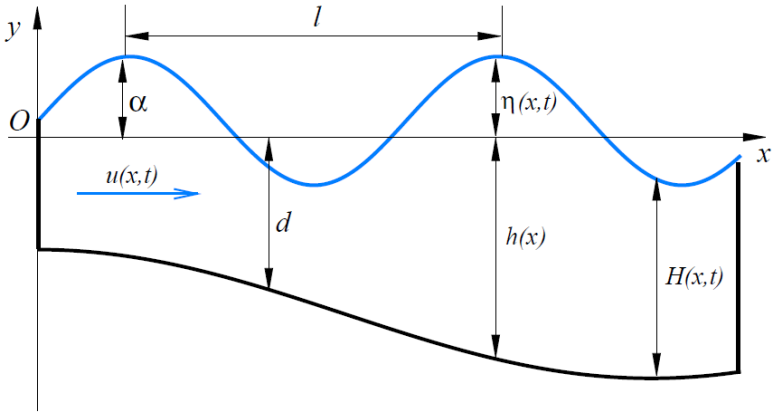


Figure 3.1: Definition sketch

We solve for the velocity variable $u(x,t)$, which represents the depth-averaged velocity, similar to the SWE. The variable $\wp(x,t)$ denotes the non-hydrostatic component of the depth-integrated pressure and is expressed as:

$$\wp = \frac{H^3}{3}\mathscr{R}_1 + \frac{H^2}{2}\mathscr{R}_2, \tag{3.3}$$

where:

$$\mathscr{R}_1 = u_{xt} + uu_{xx} - (u_x)^2, \qquad \mathscr{R}_2 = u_t h_x + u\left[uh_x\right]_x. \tag{3.4}$$

$\varrho(x,t)$ is the non-hydrostatic contribution of the pressure trace at the bottom and can be written as:

$$\varrho = \frac{H^2}{2}\mathscr{R}_1 + H\mathscr{R}_2. \tag{3.5}$$

Since the equations above contain mixed derivatives up to the third order, they are more complex to solve compared to the classical SWE. From a numerical standpoint, the high-order derivatives pose challenges to the solution and can lead to instability problems. Khakimzyanov et al. [2017] simplified the system by extracting a second-order sub-problem for the non-hydrostatic pressure term. It is important to note that this form of the SGN equations is quite unique and was first proposed by Khakimzyanov et al. [2017]. The derivation of this special form of SGN equations in 1D is described in Appendix B. The non-hydrostatic part of the pressure is computed by solving the following linear elliptic equation with variable coefficients:

$$4\left[\frac{\wp_x}{H\mathscr{Y}}\right]_x - 6\left[\frac{2}{H^3}.\frac{\mathscr{Y}-3}{\mathscr{Y}} + \left[\frac{h_x}{H^2\mathscr{Y}}\right]_x\right]\wp = \mathscr{F}, \tag{3.6}$$

where: $\mathscr{Y} = 4 + h_x^2$. For non-moving topography, (*i.e.*, $h_t = 0$), $\mathscr{F}$ and $\mathscr{R}$ are expressed as:

$$\mathscr{F} = \left[g\eta_x + \frac{\mathscr{R}h_x}{\mathscr{Y}}\right]_x - \frac{6\mathscr{R}}{H\mathscr{Y}} + 2u_x^2, \tag{3.7}$$

$$\mathscr{R} = -g\eta_x h_x + u^2 h_{xx}. \tag{3.8}$$

The non-hydrostatic pressure at the bottom $\varrho\,(x,t)$ can be expressed through $\wp$ in the following way:

$$\varrho\,(x,t) = \frac{1}{\mathscr{Y}}\left[\frac{6\wp}{H} + H\mathscr{R} + \wp_x h_x\right]. \tag{3.9}$$

Since the hyperbolic and elliptic terms are split in this form of the SGN equations, each system can be solved independently with the appropriate numerical operators.

3.2.2 Numerical Scheme

The solution of the SGN equations requires the solution of the elliptic system (*i.e.*, Eq. (3.6)). Here we present the numerical solution of the elliptic problem on two types of grids: Collocated grid and Staggered grid.

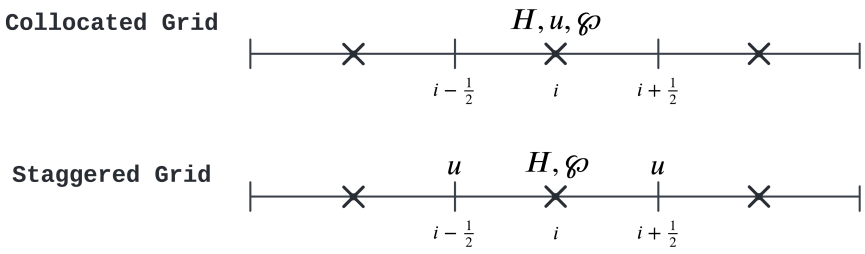


Figure 3.2: Collocated and Staggered grid

In order to discretize the elliptic problem, we integrate Eq. (3.6) over one cell $\left[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\right]$:

$$\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} 4\left[\frac{\wp_x}{H\mathcal{Y}}\right]_x dx - 6\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \left[\frac{2}{H^3}.\frac{\mathcal{Y}-3}{\mathcal{Y}} + \left[\frac{h_x}{H^2\mathcal{Y}}\right]_x\right]\wp\, dx = \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathcal{F}\, dx. \tag{3.10}$$

To simplify the notation we introduce the variable $K = \frac{4}{H\mathcal{Y}}$ and we discretize the elliptic problem terms one by one:

$$\begin{aligned}
\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} 4\left[\frac{\wp_x}{H\mathcal{Y}}\right]_x dx &= \left[K\wp_x\right]_{i+\frac{1}{2}} - \left[K\wp_x\right]_{i-\frac{1}{2}} \\
&= \frac{K_{i+1}+K_i}{2}\frac{\wp_{i+1}-\wp_i}{\Delta x} - \frac{K_i+K_{i-1}}{2}\frac{\wp_i-\wp_{i-1}}{\Delta x} \\
&= \left(\frac{K_{i+1}+K_i}{2\Delta x}\right)\wp_{i+1} - \left(\frac{K_{i+1}+2K_i+K_{i-1}}{2\Delta x}\right)\wp_i + \left(\frac{K_i+K_{i-1}}{2\Delta x}\right)\wp_{i-1},
\end{aligned}$$

where:

$$K_i = \frac{4}{H_i\left(4+\frac{\left(h_{i+1}-h_{i-1}\right)^2}{4\Delta x^2}\right)}. \tag{3.11}$$

Similarly, the second term can be discretized as:

$$\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} 6\left[\frac{2}{H^3}.\frac{\mathcal{Y}-3}{\mathcal{Y}} + \left[\frac{h_x}{H^2\mathcal{Y}}\right]_x\right]\wp\, dx = 6\left(\frac{\Delta x}{H_i^3}\frac{\mathcal{Y}_i-3}{\mathcal{Y}_i} + \left[\frac{h_x}{H^2\mathcal{Y}}\right]_{i+\frac{1}{2}} - \left[\frac{h_x}{H^2\mathcal{Y}}\right]_{i-\frac{1}{2}}\right)\wp_i, \tag{3.12}$$

where:

$$\mathcal{Y}_i = 4 + \frac{\left(h_{i+1}-h_{i-1}\right)^2}{4\Delta x^2}, \tag{3.13}$$

and:

$$\left[\frac{h_x}{H^2\mathcal{Y}}\right]_{i+\frac{1}{2}} = \frac{\Delta x\left(h_{i+1}-h_i\right)}{\left(\frac{H_{i+1}+H_i}{2}\right)^2\left(4\Delta x^2+\left(h_{i+1}+h_i\right)^2\right)}. \tag{3.14}$$

The source term of the elliptic problem is computed with:

$$\mathcal{F}_i = \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathcal{F}\, dx = \left[g\eta_x + \frac{\mathcal{R}h_x}{\mathcal{Y}}\right]_{i+\frac{1}{2}} - \left[g\eta_x + \frac{\mathcal{R}h_x}{\mathcal{Y}}\right]_{i-\frac{1}{2}} + 2\Delta x\left[u_x\right]_i^2 - \Delta x\frac{6\mathcal{R}_i}{H_i\mathcal{Y}_i}, \tag{3.15}$$

where:

$$R_{i+\frac{1}{2}} = -g\left[\eta_x\right]_{i+\frac{1}{2}}\left[h_x\right]_{i+\frac{1}{2}} + u_{i+\frac{1}{2}}^2\left[h_{xx}\right]_{i+\frac{1}{2}}, \tag{3.16}$$

$$R_i = -g\,[\eta_x]_i\,[h_x]_i + u_i^2\,[h_{xx}]_i\,. \tag{3.17}$$

We use a central difference scheme to compute the space derivatives of h and η:

$$[\eta_x]_{i+\frac{1}{2}} = \frac{\eta_{i+1}-\eta_i}{\Delta x}, \qquad [h_x]_{i+\frac{1}{2}} = \frac{h_{i+1}-h_i}{\Delta x}, \qquad [h_{xx}]_{i+\frac{1}{2}} = \frac{h_{i+2}-h_{i+1}-h_i+h_{i-1}}{2\Delta x^2}, \tag{3.18}$$

$$[\eta_x]_i = \frac{\eta_{i+1}-\eta_{i-1}}{2\Delta x}, \qquad [h_x]_i = \frac{h_{i+1}-h_{i-1}}{2\Delta x}, \qquad [h_{xx}]_i = \frac{h_{i+1}-2h_{i+1}+h_i}{\Delta x^2}. \tag{3.19}$$

In the case of a collocated grid, we get the following discretization of the velocity derivatives:

$$[u_x]_i = \frac{u_{i+1}-u_{i-1}}{2\Delta x}, \qquad u_{i+\frac{1}{2}} = \frac{u_{i+1}+u_i}{2}. \tag{3.20}$$

For the staggered grid we will have:

$$[u_x]_i = \frac{u_{i+\frac{1}{2}}-u_{i-\frac{1}{2}}}{\Delta x}, \qquad u_i = \frac{u_{i+\frac{1}{2}}+u_{i-\frac{1}{2}}}{2}. \tag{3.21}$$

In order to solve the elliptic problem for the values of $\wp$, we need to solve the following tridiagonal system:

$$\begin{pmatrix} \beta_1 & \gamma_1 & & 0 \\ \alpha_2 & \ddots & \ddots & \\ & \ddots & \ddots & \gamma_{N-1} \\ 0 & & \alpha_N & \beta_N \end{pmatrix} \begin{pmatrix} \wp_1 \\ \wp_2 \\ \vdots \\ \wp_N \end{pmatrix} = \begin{pmatrix} \mathscr{F}_1 \\ \mathscr{F}_2 \\ \vdots \\ \mathscr{F}_N \end{pmatrix}, \tag{3.22}$$

where:

$$\gamma_i = \frac{K_{i+1}+K_i}{2\Delta x}, \tag{3.23}$$

$$\beta_i = -\frac{K_{i+1}+2K_i+K_{i-1}}{2\Delta x} - 6\left(\frac{1}{H_i^3}\frac{\mathscr{Y}_i-3}{\mathscr{Y}_i} + \left[\frac{h_x}{H^2\mathscr{Y}}\right]_{i+\frac{1}{2}} - \left[\frac{h_x}{H^2\mathscr{Y}}\right]_{i-\frac{1}{2}}\right), \tag{3.24}$$

$$\alpha_i = \frac{K_i+K_{i-1}}{2\Delta x}. \tag{3.25}$$

The tridiagonal system is solved with the Thomas algorithm, and the non-hydrostatic pressure at the bottom ϱ is then computed at the cell center in accordance with Eq. (3.9):

$$\varrho_i = \frac{1}{\mathscr{Y}_i}\left[\frac{6\wp_i}{H_i} + H_i\mathscr{R}_i + \frac{\wp_{i+1}-\wp_{i-1}}{2\Delta x}\frac{h_{i+1}-h_{i-1}}{2\Delta x}\right]. \tag{3.26}$$

Finally, the non-hydrostatic corrections are added to the momentum equation as source terms:

$$\left[\wp_x - \varrho h_x\right]_{i+\frac{1}{2}} = \frac{\wp_{i+1} - \wp_i}{\Delta x} + \frac{\varrho_{i+1} + \varrho_i}{2}\frac{h_{i+1} - h_i}{\Delta x}. \tag{3.27}$$

3.2.3 Verification

Solitary wave solution

In the case of a flat bathymetry $h(x,t) = d$, the SGN equations admit an analytical solitary wave solution, given by the following expressions:

$$\eta(x,t) = \alpha \operatorname{sech}^2\left(\frac{\sqrt{3\alpha g}}{2dv}(x - x_0 - vt)\right), \tag{3.28}$$

$$u(x,t) = \frac{v\eta(x,t)}{d + \eta(x,t)}, \tag{3.29}$$

where α is the wave amplitude, x_0 is the initial wave position and v is the wave celerity defined as:

$$v = \sqrt{g(d+\alpha)}. \tag{3.30}$$

The non-hydrostatic pressure under the solitary wave can be written as:

$$\wp(x,t) = \frac{g}{2}\left[H(x,t)^2 - d^2\right] - dvu(x,t). \tag{3.31}$$

This analytical solution is used to validate the SGN numerical solver.

Numerical results

We take a sufficiently large domain $[0, l]$ with $l = 500\,\mathrm{m}$ and a water depth $d = 1\,\mathrm{m}$. The solitary wave is initially located at $x_0 = 35\,\mathrm{m}$ with an amplitude $\alpha = 0.4\,\mathrm{m}$. We compute the soliton propagation for $t = 100\,\mathrm{s}$ using a Courant number $Cr = 0.8$. The SGN solution approximates the full Euler solution fairly well up to $2\alpha \leqslant d$ (Duran et al. [2013]).

We first verify the elliptic solver by comparing the computed $\wp$ to the exact solution given in Eq. (3.31). Both the collocated and staggered grid solutions converge to the analytical solution, which verifies the elliptic solver for this test. The solutions with the staggered and collocated grids are identical.

Figure 3.4 shows the computed free surface profile at $t = 100\,\mathrm{s}$ for different grid resolutions. The numerical solution converges to the exact solution with mesh refinement.

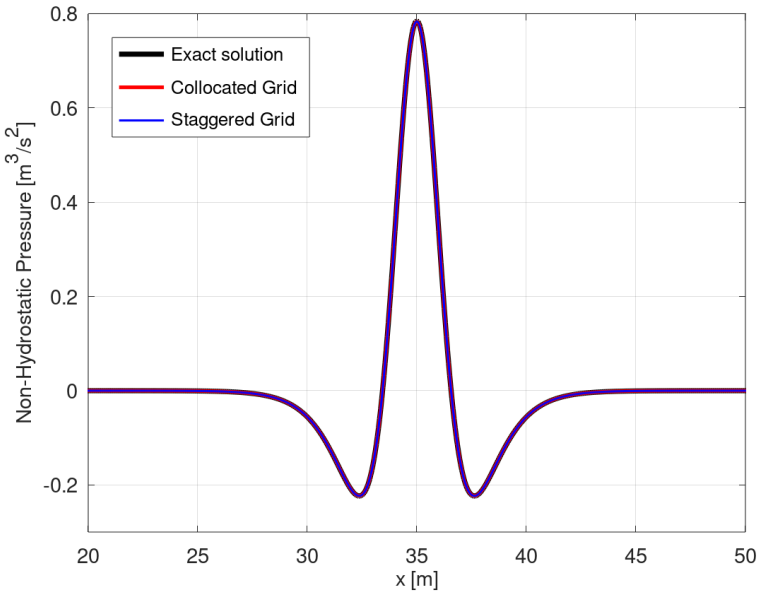


Figure 3.3: Elliptic solver verification.

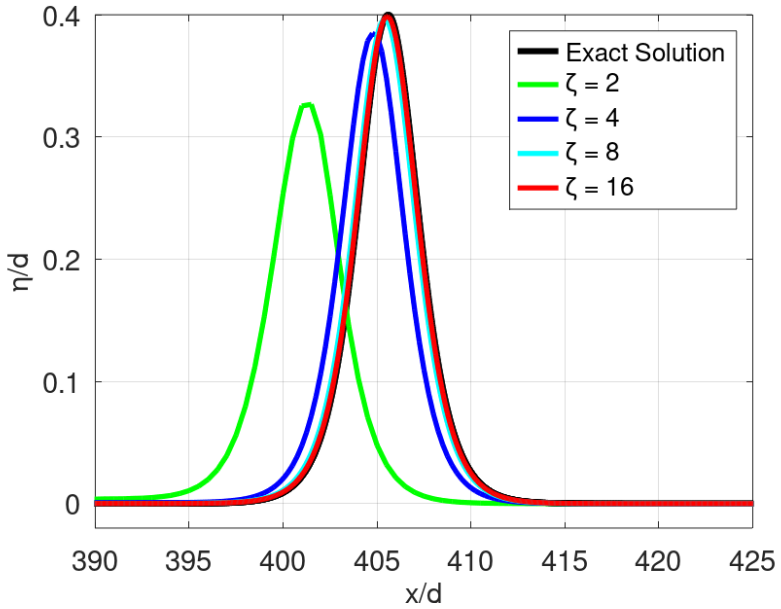


Figure 3.4: Solitary wave solution at $t = 100\,\mathrm{s}$ for different grid resolutions $\zeta = \frac{d}{\Delta x}$

In order to quantify the accuracy of the numerical solution, we measure the L_∞-error discrete error for different grid resolutions ($\zeta = \frac{d}{\Delta x}$) :

$$\left\|\varepsilon_\zeta\right\|_\infty = \left\|\eta_\zeta - \eta_0\right\|_\infty , \tag{3.32}$$

where η_ζ stands for the numerical and η_0 for the exact free surface profiles. The convergence

rate $\mathbb{K}$ is computed as:

$$\mathbb{K} = \log_2 \left\{ \frac{\left\| \varepsilon_\zeta \right\|_\infty}{\left\| \varepsilon_{2\zeta} \right\|_\infty} \right\}. \tag{3.33}$$

The numerical results in Table 3.1 indicate the error and convergence rate for different grid resolutions. The convergence rate is around 2 which validates the second-order accuracy of our numerical solver.

Table 3.1: Solitary wave solution: error and convergence rate

Δx [m]	ζ	$\left\| \varepsilon_\zeta \right\|_\infty$	$\mathbb{K}$
0.5	2	0.4471	-
0.25	4	0.1177	1.93
0.125	8	0.0301	1.97
0.0625	16	0.0076	1.98

3.3 Nwogu 1D model

3.3.1 Governing Equations

The Boussinesq-type equations of Nwogu [1993] consist of a continuity and a momentum equation in the form:

$$\eta_t + \Big[(\eta + h)\, u \Big]_x + \left[\left(\frac{z_\alpha^2}{2} - \frac{h^2}{6} \right) h u_{xx} + \left(z_\alpha + \frac{h}{2} \right) h \left(hu \right)_{xx} \right]_x = 0, \tag{3.34}$$

$$u_t + u u_x + g \eta_x + \left[\frac{z_\alpha^2}{2} u_{xx} + z_\alpha \left(hu \right)_{xx} \right]_t = 0. \tag{3.35}$$

where z_α is the reference depth, at which the velocity variable is evaluated (*i.e.*, $u = u|_{z=z_\alpha}$). Nwogu [1993] defines this level as a fixed proportion of the local water depth with a position near mid-depth: $z_\alpha = \sigma h = -0.531 h$.

We define the dispersion variables:

$$\psi_c = \left[\left(\frac{z_\alpha^2}{2} - \frac{h^2}{6} \right) h u_{xx} + \left(z_\alpha + \frac{h}{2} \right) h \left(hu \right)_{xx} \right]_x, \tag{3.36}$$

$$\psi_m = \frac{z_\alpha^2}{2} u_{xx} + z_\alpha \left(hu \right)_{xx}. \tag{3.37}$$

We derive the conservative form of Nwogu's equations. Since the bathymetry is stationary, the continuity equation becomes:

$$H_t + [Hu]_x + \psi_c = 0. \tag{3.38}$$

To write the momentum equation in a conservative form, we multiply the equation by H. The momentum acceleration terms, $u_t + uu_x$, are expressed in the conservative form by replacing the flow velocity u with Hu:

$$H\Big(u_t + uu_x\Big) = \Big[Hu\Big]_t + \Big[Hu^2\Big]_x - u\Big(H_t + [Hu]_x\Big) = \Big[Hu\Big]_t + \Big[Hu^2\Big]_x + u\psi_c. \tag{3.39}$$

The dispersion term in the momentum equation can be written in the conservative form, as follows:

$$H\,[\psi_m]_t = [H\psi_m]_t - H_t\psi_m. \tag{3.40}$$

The momentum equation becomes:

$$\Big[Hu + H\psi_m\Big]_t + \Big[Hu^2\Big]_x + gH\eta_x + u\psi_c - H_t\psi_m = 0. \tag{3.41}$$

The resulting set of governing Eqs. (3.38) and (3.41) contain the nonlinear shallow-water equations and second-order dispersion terms based on the flow velocity. The governing equations are thus expressed in a conservative form:

$$U_t + F_x + S = 0, \tag{3.42}$$

where:

$$U = \begin{bmatrix} H \\ Hu + H\psi_m \end{bmatrix}, \qquad F = \begin{bmatrix} Hu \\ Hu^2 \end{bmatrix}, \qquad S = \begin{bmatrix} \psi_c \\ gH\eta_x + u\psi_c - H_t\psi_m \end{bmatrix}. \tag{3.43}$$

3.3.2 Numerical Scheme

In this section, we describe the extension of the SWE model to the solution of Nwogu's equations. As previously mentioned, and similar to the SGN solution, the hyperbolic component of the dispersive equation is computed with the SWE solver described in Chapter 2 while the dispersive terms will be computed with central FD approximations. The layout of Nowgu's variables on a staggered grid is shown in Figure 3.5.

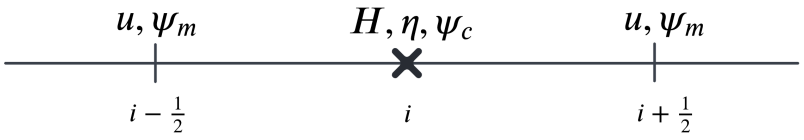


Figure 3.5: Staggered layout for Nwogu's variables

The continuity source term is computed as:

$$\begin{aligned}\left[\psi_c\right]_i = &\left(\frac{z^2_{\alpha(i+\frac{1}{2})}}{2} - \frac{1}{6}\right) h_{i+\frac{1}{2}} \frac{u_{i+\frac{3}{2}} - 2u_{i+\frac{1}{2}} + u_{i-\frac{1}{2}}}{\Delta x^3} + \left(z_{\alpha(i+\frac{1}{2})} + \frac{1}{2}\right) h_{i+\frac{1}{2}} \frac{(hu)_{i+\frac{3}{2}} - 2\,(hu)_{i+\frac{1}{2}} + (hu)_{i-\frac{1}{2}}}{\Delta x^3} \\ - &\left(\frac{z^2_{\alpha(i-\frac{1}{2})}}{2} - \frac{1}{6}\right) h_{i-\frac{1}{2}} \frac{u_{i+\frac{1}{2}} - 2u_{i-\frac{1}{2}} + u_{i-\frac{3}{2}}}{\Delta x^3} + \left(z_{\alpha(i-\frac{1}{2})} + \frac{1}{2}\right) h_{i-\frac{1}{2}} \frac{(hu)_{i+\frac{1}{2}} - 2\,(hu)_{i-\frac{1}{2}} + (hu)_{i-\frac{3}{2}}}{\Delta x^3}.\end{aligned}$$

Similarly, the dispersion variable in the momentum equation ψ_m is approximated as:

$$\left[\psi_m\right]_{i+\frac{1}{2}} = \frac{z^2_{\alpha(i+\frac{1}{2})}}{2} \frac{u_{i+\frac{3}{2}} - 2u_{i+\frac{1}{2}} + u_{i-\frac{1}{2}}}{\Delta x^2} + z_{\alpha(i+\frac{1}{2})} \frac{(hu)_{i+\frac{3}{2}} - 2\,(hu)_{i+\frac{1}{2}} + (hu)_{i-\frac{1}{2}}}{\Delta x^2}. \tag{3.44}$$

In this case, the conserved variables for the hyperbolic solver are H and $P = Hu + H\psi_m$. The momentum conserved variable P is expressed in terms of the flow variables as:

$$P_{i+\frac{1}{2}} = H_{i+\frac{1}{2}} u_{i+\frac{1}{2}} + H_{i+\frac{1}{2}} z_{\alpha(i+\frac{1}{2})} \left[\psi_m\right]_{i+\frac{1}{2}}. \tag{3.45}$$

The central difference scheme only involves cells to the left and right of each grid cell and thus forms a linear system of equations in the form:

$$\begin{pmatrix} \beta_1 & \gamma_1 & & 0 \\ \alpha_2 & \ddots & \ddots & \\ & \ddots & \ddots & \gamma_{N-1} \\ 0 & & \alpha_N & \beta_N \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{pmatrix} = \begin{pmatrix} P_1 \\ P_2 \\ \vdots \\ P_N \end{pmatrix}. \tag{3.46}$$

To simplify the expressions, we use the index k to replace the index at the cells interface $k = i + \frac{1}{2}$

$$\alpha_k = H_k z_{\alpha k} \left(\frac{z_{\alpha k}}{2\Delta x^2} + \frac{h_{k-1}}{\Delta x^2}\right), \tag{3.47}$$

$$\beta_k = H_k z_{\alpha k}\left(\frac{1}{z_{\alpha k}} - \frac{z_{\alpha k}}{\Delta x^2} - \frac{2h_k}{\Delta x^2}\right), \tag{3.48}$$

$$\gamma_k = H_k z_{\alpha k}\left(\frac{z_{\alpha k}}{2\Delta x^2} + \frac{h_{k+1}}{\Delta x^2}\right). \tag{3.49}$$

The Thomas algorithm is used to solve the tridiagonal system, and the horizontal velocity at the interface (*i.e.*, $u_{i+\frac{1}{2}}$) is then computed.

3.3.3 Verification

Solitary wave solution

The solitary wave solution is one of the standard tests to verify the accuracy of Boussinesq-type solvers. While propagating in a flat frictionless channel, a solitary wave is expected to maintain its shape, speed, and amplitude due to the exact balance of dispersive and nonlinear terms. In the case of Nwogu's equations, Hamdi et al. [2005] have derived an exact solution to the dispersive equations. However, the non-physical aspect of the solution (*i.e.*, the total water depth is negative for $\sigma = -0.531$) makes it impossible to compute with a conservative numerical implementation, which relies on the assumption of the non-negativity of the total water depth. Wei and Kirby [1995], on the other hand, derived an approximate analytical solitary wave solution to Nwogu's equations. For a flat topography (*i.e.*, $h = d = \text{const}$), the 1D Nwogu's equations are expressed as:

$$\eta_t + du_x + [\eta u]_x + \left(\alpha + \frac{1}{3}\right) d^3 u_{xxx} = 0, \tag{3.50}$$

$$u_t + uu_x + g\eta_x + \alpha d^2 u_{xxt} = 0. \tag{3.51}$$

with $\alpha = \frac{\sigma^2}{2} + \sigma = -0.39$. Wei and Kirby [1995] proposed an approximate ordinary differential equation to Eq. (3.50)- (3.51) for the wave potential ϕ:

$$-\phi_{tt} + \phi_{xx} - \varepsilon\left(2\phi_x\phi_{xt} + \phi_t\phi_{xx}\right) + \mu^2\left[\left(\alpha + \frac{1}{3}\right)\phi_{xxxx} - \alpha\phi_{xxtt}\right] = \mathcal{O}\left(\varepsilon^2, \varepsilon\mu^2\right). \tag{3.52}$$

Due to the truncated terms on the right-hand side, the analytical solution of Eq. (3.52) differs slightly from the exact solitary wave solution of the original equations (*i.e.*, Eqs. (3.50) and (3.51)). Eq. (3.52) admits the following exact solutions for η and u:

$$u = A\,\mathrm{sech}^2\left[B\left(x - x_0 - Ct\right)\right], \tag{3.53}$$

$$\eta = A_1\,\mathrm{sech}^2\left[B\left(x - x_0 - Ct\right)\right] + A_2\,\mathrm{sech}^4\left[B\left(x - x_0 - Ct\right)\right]. \tag{3.54}$$

The corresponding dimensional expressions for A, B, A_1 and A_2 are:

$$A = \frac{C^2 - gd}{C}, \tag{3.55}$$

$$B = \left\{ \frac{C^2 - gd}{4\left[\left(\alpha + \frac{1}{3}\right) gd^3 - \alpha d^2 C^2\right]} \right\}^{\frac{1}{2}}, \tag{3.56}$$

$$A_1 = \frac{C^2 - gd}{3\left[\left(\alpha + \frac{1}{3}\right) gd - \alpha C^2\right]} d, \tag{3.57}$$

$$A_2 = -\frac{\left(C^2 - gd\right)^2}{2gdC^2} \frac{\left[\left(\alpha + \frac{1}{3}\right) gd + 2\alpha C^2\right]}{\left[\left(\alpha + \frac{1}{3}\right) gd - \alpha C^2\right]} d. \tag{3.58}$$

The parameter C is computed by solving the following third-order polynomial equation:

$$2\alpha \left(\frac{C^2}{gd}\right)^3 - \left(3\alpha + \frac{1}{3} + 2\alpha\delta\right) \left(\frac{C^2}{gd}\right)^2 + 2\delta \left(\alpha + \frac{1}{3}\right) \left(\frac{C^2}{gd}\right) + \left(\alpha + \frac{1}{3}\right) = 0. \tag{3.59}$$

Numerical results

We propagate the solitary wave described in Eq. (3.53), (3.54) in a $d = 0.45\,\mathrm{m}$ water depth for $t = 150\,\mathrm{s}$ and we compare the numerical results to the analytical solution proposed by Wei and Kirby [1995]. The wave amplitude for this test is $A/d = 0.1$. The converged results with grid size $\zeta = \frac{d}{\Delta x} = 16$ are shown in Figure 3.6 .

The numerical results show good agreement between the computed and the analytical solutions. The converged solitary solution has a slightly higher amplitude than the approximate analytical solution, along with a small dispersive tail that is consistent with numerical findings from other Nwogu's models (*e.g.*, Kazolea and Delis [2013]; Wei and Kirby [1995]).

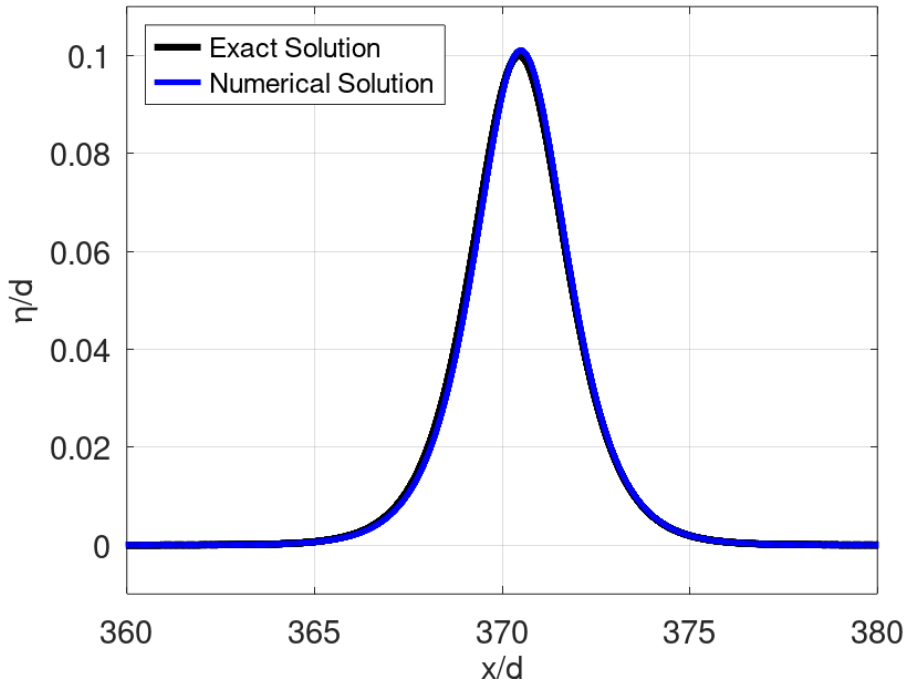


Figure 3.6: Solitary wave propagation $(A/d = 0.1)$ at time $t = 145s$

3.4 Frequency Dispersion

The quality of the numerical computations depends on the accuracy of the governing equations and the discretization techniques. Both exhibit numerical errors that can offset or reinforce each other. It is important to understand the limitations of numerical models so that they are used correctly and to their best potential for wave fore- and hindcasting. An inherent source of error related to Boussinesq models results from the approximate nature of the governing equations. Both the SGN and Nwogu equations are truncated to $\mathcal{O}(\mu^2)$ order, resulting in derivatives up to the third order. While these approximations reduce the numerical complexity of the equations, they can lead to high dispersive errors due to the limited representation of the flow's vertical structure. In this section, we derive the linear dispersion relation for both the Nwogu and SGN equations and discuss the impact of dispersive errors on the accuracy of the results.

3.4.1 Linear Dispersion Relation

The linearized standard SGN equations (*i.e.*, Eq.(3.1)-(3.4)) take the following form for a constant water depth h:

$$\begin{aligned} \eta_t + hu_x &= 0, \\ u_t + g\eta_x - \frac{h^3}{3}u_{xxt} &= 0. \end{aligned} \tag{3.60}$$

Similarly, the linearized Nwogu's equations can be expressed as:

$$\begin{aligned} \eta_t + hu_x + \left(\alpha + \frac{1}{3}\right) h^3 u_{xxx} = 0, \\ u_t + g\eta_x + \alpha h^2 u_{xxt} = 0. \end{aligned} \tag{3.61}$$

It is worth noting that when linearized, the SGN equations are equivalent to Nwogu's equation for $\alpha = -\frac{1}{3}$.

We consider a steady periodic wave solution of the following form:

$$\eta = \eta_0 \, \mathrm{e}^{\mathrm{i}(kx-\omega t)}, \qquad u = u_0 \, \mathrm{e}^{\mathrm{i}(kx-\omega t)}. \tag{3.62}$$

where k is wave number and ω is the wave frequency. By substituting Eq. (3.62) into Eq. (3.61) we get:

$$\begin{pmatrix} -\mathrm{i}\omega & \mathrm{i}kh - \mathrm{i}\left(\alpha + \frac{1}{3}\right)(kh)^3 \\ \mathrm{i}gk & -\mathrm{i}\omega + \mathrm{i}\alpha\omega(kh)^2 \end{pmatrix} \cdot \begin{pmatrix} \eta \\ u \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{3.63}$$

Non-zero solutions are obtained if:

$$\omega^2\left[1 - \alpha(kh)^2\right] = k^2\left[gh - gh\left(\alpha + \frac{1}{3}\right)(kh)^2\right]. \tag{3.64}$$

As a result, the dispersion relation is written as:

$$C = \frac{\omega}{k} = \left\{ gh \frac{1 - \left(\alpha + \frac{1}{3}\right)(kh)^2}{1 - \alpha\,(kh)^2} \right\}^{\frac{1}{2}}. \tag{3.65}$$

By taking $\alpha = -0.39$ we find the dispersion relation proposed by Nwogu [1993]:

$$C_{Nwogu} = \left\{ gh \frac{1 - \left(\alpha + \frac{1}{3}\right)(kh)^2}{1 - \alpha\,(kh)^2} \right\}^{\frac{1}{2}}. \tag{3.66}$$

For the SGN equations, we take $\alpha = -\frac{1}{3}$:

$$C_{SGN} = \left\{ gh \frac{1}{1 + \frac{1}{3}(kh)^2} \right\}^{\frac{1}{2}}. \tag{3.67}$$

The SGN equations share the same linear dispersion relation with other well-established equations (*e.g.*, Peregrine [1967]; Zheleznyak and Pelinovsky [1985]). The nonlinear properties of these equations, meanwhile, can be different.

The linear dispersion relation of each equation is then compared to the fully dispersive Airy wave theory:

$$C_{Airy} = \left\{ gh \frac{\tanh(kh)}{kh} \right\}^{\frac{1}{2}}. \tag{3.68}$$

Figure 3.7 shows the dispersion error in the form of the ratio between the phase celerity of Airy wave theory and Boussinesq-type equations. The dispersive error for SGN equations increases quite considerably for $kh > 1$, limiting their applicability for high-frequency waves. Nwogu's equations, on the other hand, match the Airy theory up to $kh \approx \pi$, making the equations very applicable to shallow and intermediate water. In addition, the error sign is also different between the two equations. The SGN model has a negative error and tends to underpredict the wavelength and celerity. Although small, the error in Nwogu's formulation ($\alpha = -0.39$) is negative around $kh \in [1, 2]$, but it quickly changes to a positive error for large values of kh.

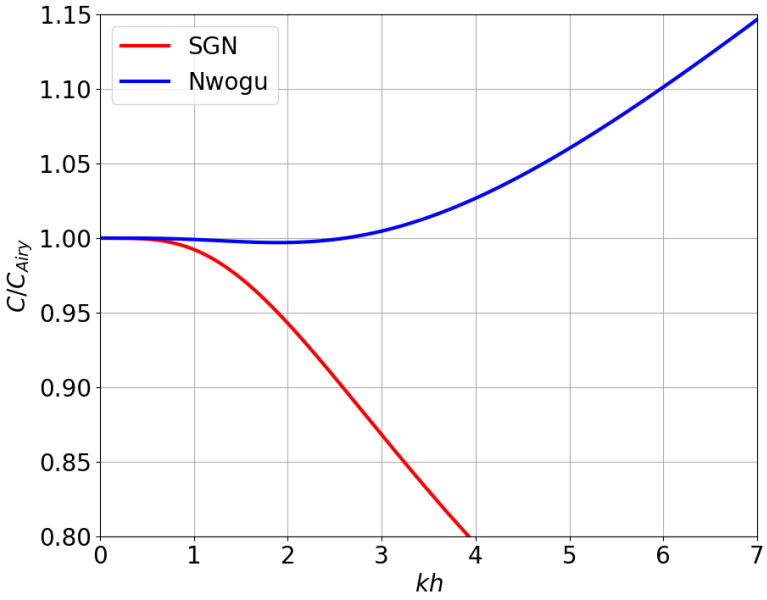


Figure 3.7: Error in phase speed between Airy wave theory and linearized SGN equations

3.4.2 Numerical Tests

Periodic wave propagation in a flat channel

To better understand the effect of the dispersion properties of the governing equations on the numerical model's overall quality, we run a simple numerical test with the two sets of equations. We generate and propagate linear monochromatic waves ($\frac{a}{h} = 0.01 \ll 1$) in a very long flat channel using two models: the SGN model and the Nwogu model, and we compare the computed results to the Airy theory.

The generation of the waves inside the computational domain is achieved through the internal wavemaker approach described in Chapter 6.2.3. For each run, we input different monochromatic

waves with varying kh values (*i.e.*, $kh = 0.5, 1, 2,$ and 3). The grid size is determined for each test such that $\zeta = \frac{\lambda}{dx} = 20$. The results are shown in Figure 3.8

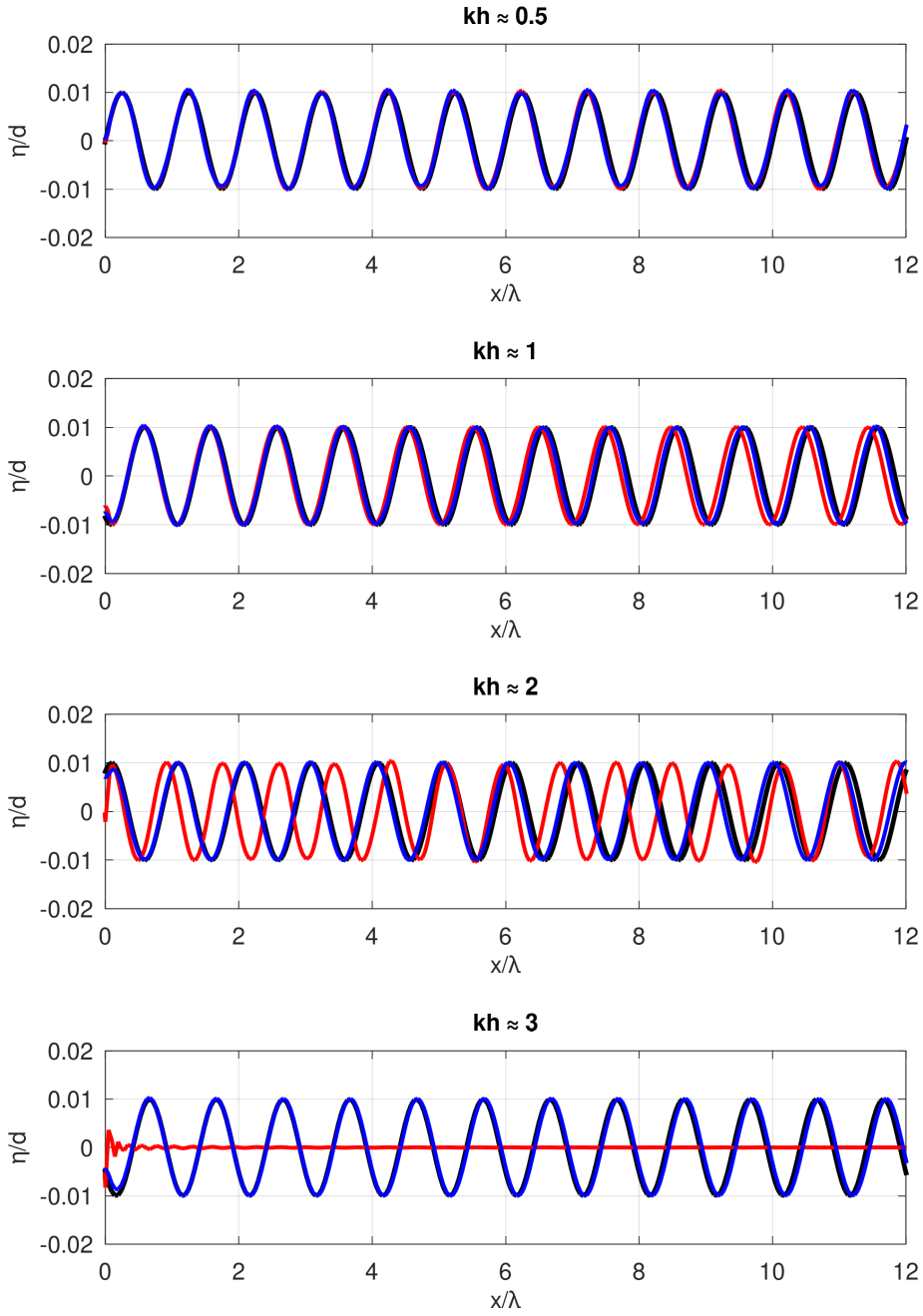


Figure 3.8: Computation of monochromatic wave propagation in a long flat channel for different values of kh. The lines denote the reference and computed free surface at $t = 500\,\mathrm{s}$. The blue line denotes the solution with Nwogu's equations, the red line represents SGN equations, and the black line is the reference solution from Airy wave theory.

For $kh = 0.5$, the two equations can generate and propagate waves at the correct speed and

wavelength, which is consistent with the small dispersion error for this range of kh. For $kh = 1$, we notice a slight mismatch between the SGN model and the exact solution. This mismatch becomes more apparent for higher values of kh, where the wavelength is significantly smaller than in Airy wave theory. At $kh = 3$, the dispersion error of the SGN model is too big for the wavemaker to handle, and the wavelength reaches a singularity.

The overall results with the Nwogu model show a good agreement with Airy wave theory up to $kh = 3$. The model results are consistent with the dispersion relations shown in Figure 3.7. We can notice a slightly smaller wavelength for $kh \leqslant 2$ and a longer wavelength for $kh = 3$.

Periodic wave propagation over a submerged bar

Beji and Battjes [1993] reported a series of experimental investigations on the propagation of waves over a submerged bar. Figure 3.9 depicts a schematic of the 37.7 m long and 0.75 m tall flume with a 0.30 m tall trapezoidal bar 6 m away from a piston-type wavemaker. The front slope of the bar is 1:20, followed by a 2-meter crest and a 1:10 rear slope. At the end of the flume, a gravel beach is installed, which functions as a wave absorber. Waves shoal when propagating up the slope of the bar, forcing the development of bound higher-harmonics, which are then released from the carrier frequency on the left side of the bar as the water depth becomes rapidly deeper. Sand bars are quite common in coastal waters making this experiment an important test case for nearshore models. Many Boussinesq-type or non-hydrostatic models do not fully resolve the super-harmonics due to the low-order approximation of dispersion and nonlinearity in the governing equations. We use the experimental data from case A, which involves a sinusoidal wave with a 1 cm amplitude and 2.02 s period. The test was conducted in 0.4 m water depth, resulting in a $kh \approx 0.67$ incident wave. For both models, we run the test with a $\Delta x = 0.02$ m grid size, and a Courant number $Cr = 0.5$.

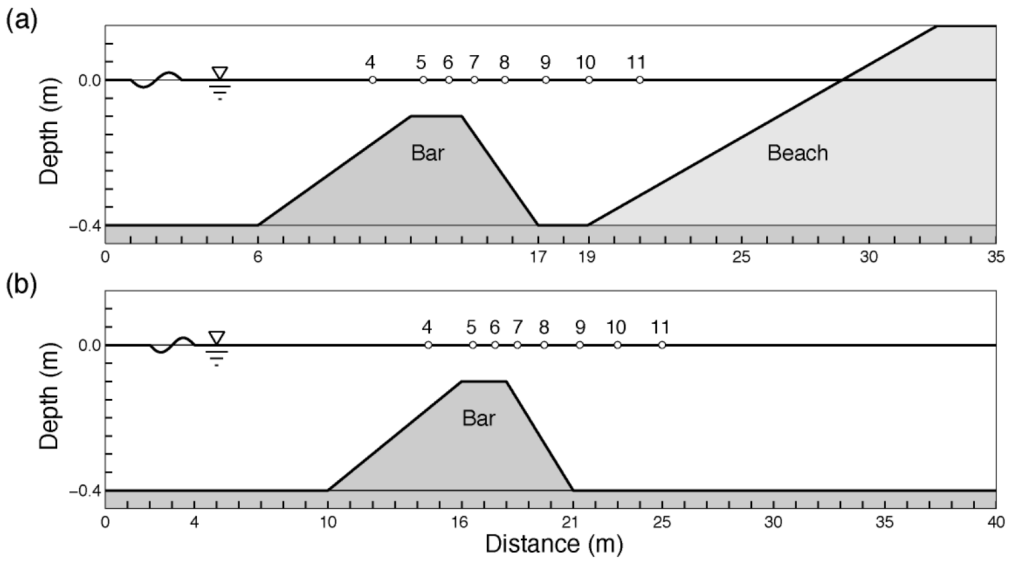


Figure 3.9: Definition sketch of wave transformation over a submerged bar. (a) Laboratory setup from Beji and Battjes [1993]. (b) Numerical model setup. Circles denote gauge locations.

Figure 3.10 shows a snapshot of the free surface computed with the two models. As the incident wave propagates over the upward slope, the wave shoals and becomes steeper, resulting in the generation of super-harmonics. These high-frequency waves are then released behind the bar as the wave travels from shallow to deeper water. While the Nwogu's model is able to compute these processes, the SGN model becomes unsable when dealing with the generation of super-harmonics. The recorded time series at different wave gauges showcase the difference between the two solutions (Figure 3.11). Both the Nwogu and SGN models maintain good agreement with experimental data at stations 4, 5, and 6 over the bar, proving the quality of numerical solutions regarding wave propagation and shoaling. The solutions of the two models significantly diverge behind the bar at stations 8–11, where super-harmonics are released. While Nwogu's model is able to capture the overall wavefield behind the bar with some minor discrepancies, the SGN fails at computing the released short waves due to the poor dispersive properties of the governing equations. This test case clearly demonstrates the impact of high-dispersive errors on the accuracy and stability of the solution. This has been confirmed by the implementation of a new model that solves SGN equations with enhanced dispersion, where the results show a significant improvement (Appendix C).

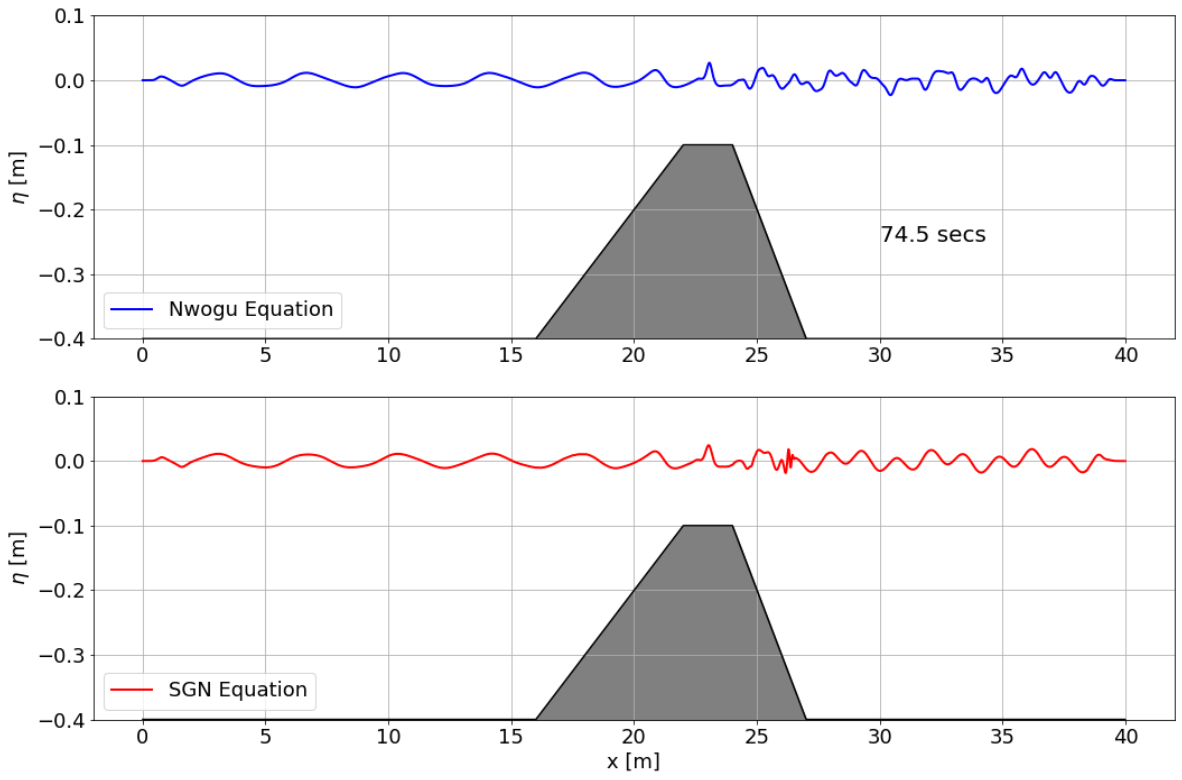


Figure 3.10: Computation of wave transformation over a submerged bar. The blue line is the free surface at 74.5 s computed with Nwogu's model. The red line is from SGN model

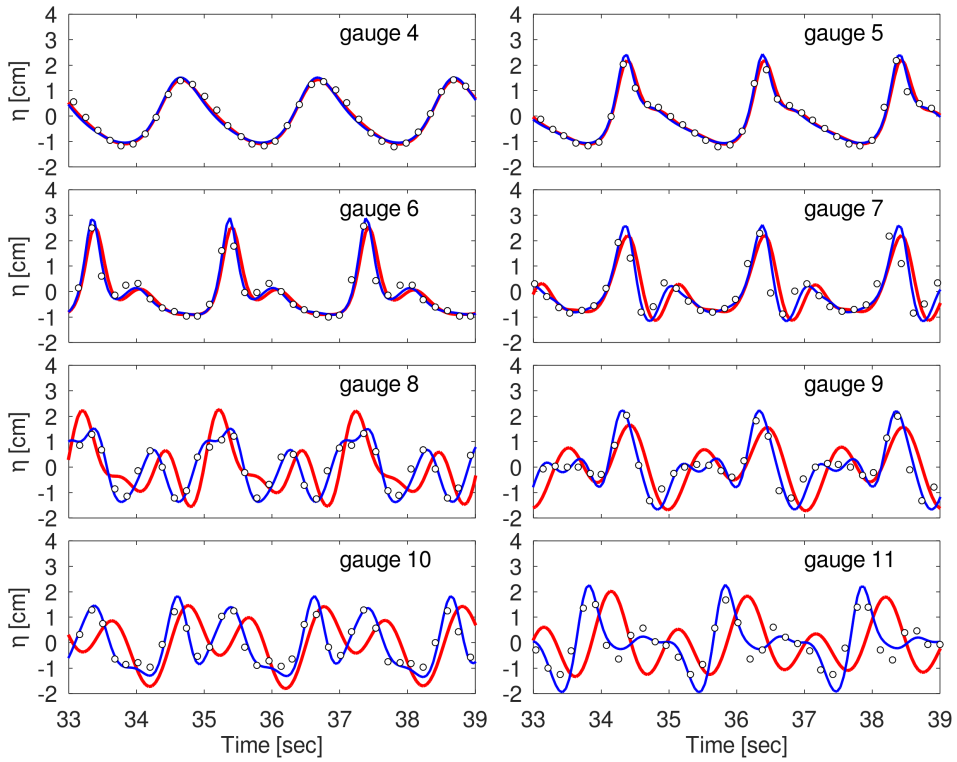


Figure 3.11: Computation of wave transformation over a submerged bar. Black circles denote laboratory data from Beji and Battjes [1993]. The blue lines are time series from Nwogu's model ($\alpha = -0.39$) and red lines represent the results SGN model

3.5 Nonlinearity

Despite their improved dispersion properties, many extended Boussinesq equations such as Nwogu's are still based on weakly nonlinear assumptions in which the dispersion terms are linearized. These assumptions help simplify the equations and their numerical solutions. In some cases, however, the small amplitude assumption might limit the model's applicability to problems where the effects of nonlinearity are too significant. In this section, we examine the effects of the weak nonlinear assumption on the numerical solution by modeling a highly nonlinear wave process, which is wave shoaling. As waves approach the shore, wave height increases, and wave height-to-water depth ratios become too large for the small amplitude assumption to be valid.

3.5.1 Numerical Tests

Nonlinear shoaling of a solitary wave on a plane beach

We consider the test presented by Grilli et al. [1994] of a solitary wave shoaling over a plane slope. The test consists of a solitary wave with amplitude $a/h_0 = 0.2$ propagating in a $h_0 = 0.44\,\mathrm{m}$ water depth, and shoaling on a constant slope $1:35$. The wave height prior to breaking approaches high

values of $\varepsilon \in [0.2, 2.2]$, and thus the benchmark is suitable for testing the nonlinear properties of the model.

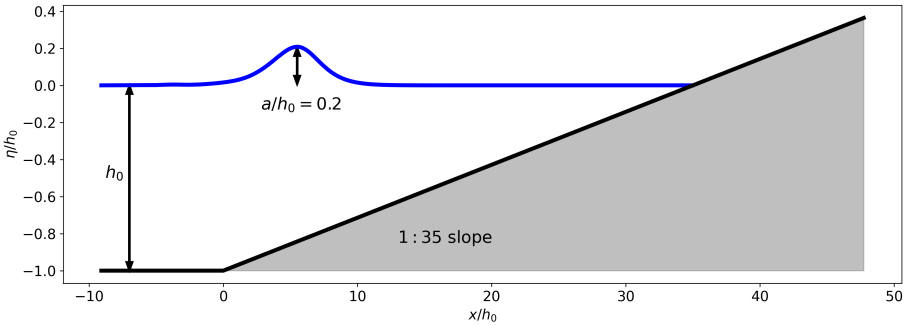


Figure 3.12: Nonlinear shoaling of a solitary wave: Numerical model setup

Figure 3.12 shows the computational domain of the test. The coordinates were set so that the slope's toe corresponds to $x = 0$. We compare the results from four numerical models, each solving a different set of governing equations with an identical numerical scheme. The results fully converge for a grid size of $\Delta x = 0.01\,\mathrm{m}$ and a Courant Number $Cr = 0.7$. The solitary waves are generated at the leftward boundary and propagated to the right. The input solitary wave profile takes the following standard form:

$$\eta\left(x,t\right) = a\,\mathrm{sech}^2\left(\sqrt{\frac{3a}{4h_0^3}}\left(x - x_0 - t\sqrt{g\left(a+h_0\right)}\right)\right),$$

$$u\left(x,t\right) = \frac{\eta\left(x,t\right)}{h_0 + \eta\left(x,t\right)}\sqrt{g\left(a+h_0\right)}.$$

For each run, we record the maximum elevation, and we compare the results with the laboratory data from Grilli et al. [1994]. The improved SGN equations are a set of SGN equations with improved dispersion (see Appendix C).

The results in Figure 3.13 show a distinct behavior of the different equations. While weakly nonlinear models (*i.e.*, Nwogu and Peregrine) tend to overestimate the shoaling wavefront, the fully nonlinear models (*i.e.*, SGN and improved SGN) have the opposite effect. This is consistent with the already-established findings, where the nonlinearity effects tend to stretch the propagating wave and reduce its amplitude (Dalrymple et al. [1984]). Since wave shoaling is a process that requires the nonlinearity and dispersion effects to balance each other, it is expected that equations with improved linear dispersion perform better than their low-dispersive counterparts. As a result, improved nonlinear and dispersion properties of the equations are both important for accurate wave shoaling computations. A fully-nonlinear potential flow model has been shown to produce a remarkable match with the laboratory experiment, which is in line with the current findings (see Grilli et al. [1994]). The pronounced oveshoaling of the weakly nonlinear Nwogu's solution has also been shown and discussed in detail in Wei and Kirby [1995].

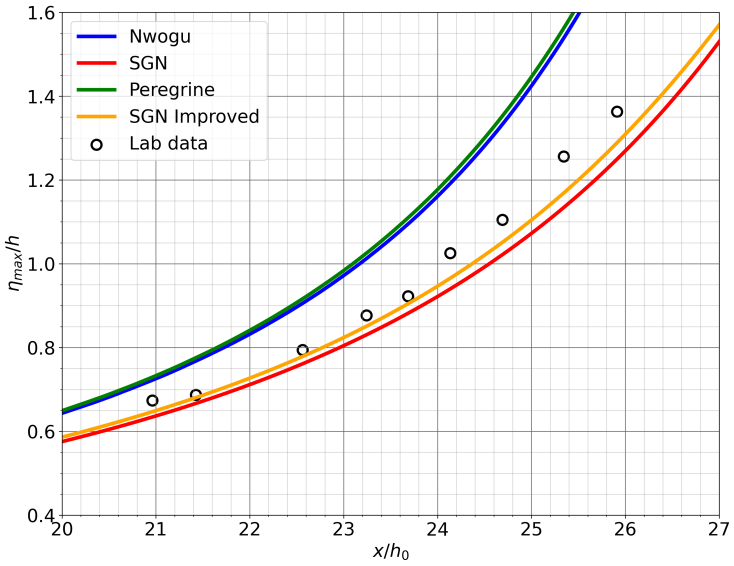


Figure 3.13: Nonlinear shoaling. Comparison between computed wave crest evolution and data from Grilli et al. [1994]

3.6 Conclusions

Thanks to the common numerical framework implemented to solve different Boussinesq-type equations, we can objectively compare the effects of the governing equation assumptions on the accuracy of the solution. We choose to compare primarily two sets of equations: the SGN equations, which are a fully nonlinear extension to the standard Boussinesq equations (Peregrine [1967]), and Nowgu's equations with their improved dispersion. With regard to dispersion, the low dispersive properties in SGN equations affect not only the accuracy but also the stability of the numerical solution, especially when dealing with short waves. The negative dispersion error in the equations, which has the tendency to shorten the wave and thus increase the error, even more, can lead to singularity in the solution. Moreover, the long-wave restriction ($kh < 0.5$) is very limiting for our applications. Nonlinearity properties, on the other hand, have been shown to be critical when dealing with steep waves, particularly during shoaling processes. The weakly nonlinear models exhibit a visible flaw when dealing with this process, resulting in an overshoaled wave and an overestimation of the wave height just before breaking. Similarly, the weakly-dispersive, fully-nonlinear models are not able to correctly reproduce this process due to the lack of dispersion and tend to underestimate the wave height. The tests presented in this chapter are essential since they highlight the limitations imposed by the assumptions in the governing equations. It is important to consider all these inherent shortcomings when working with numerical approximations. Taking into account all these findings and the numerical complexity of the 2D SGN solver (Appendix D), we build our new numerical model based on the weakly nonlinear Nwogu's equations, which offer good quality results with respect to the numerical effort involved in the solution.

Chapter 4

Sensitivity of Periodic Wave Problems to Numerical Diffusion

Contents

4.1 Introduction 83
4.2 Hybrid FV/FD Solver 84
4.2.1 Governing Equations 84
4.2.2 Godunov-type Scheme 84
4.2.3 Flux Reconstruction 86
4.3 Numerical Results 87
4.3.1 Monochromatic Wave Propagation 87
4.3.2 Irregular Wave Propagation 90
4.4 Conclusions 93

4.1 Introduction

Over the recent decades and due to the increasing technological advancement in computing power, many Boussinesq models have been developed to achieve a phase-resolving solution for nearshore wave processes. As a result, many numerical techniques for solving Boussinesq equations have been examined and considerably improved (*e.g.*, Kirby [2016]; Choi et al. [2018]; Shi et al. [2012]). Efforts to develop stable and accurate numerical implementations for the Boussinesq equations include FD, FV, and FE formulations. The first Boussinesq models were based on higher-order FD (Wei and Kirby [1995]; Nwogu and Demirbilek [2001]) and FE schemes (*e.g.*, Sørensen et al. [2004]). These strategies have been shown to be sensitive to spurious oscillations and require numerical filters and artificial viscosity to stabilize the solution near the shoreline and for breaking waves. Recently, a significant number of Boussinesq models have adopted a hybrid approach that combines FV and FD methods. The hyperbolic component of the equations is solved using conservative Godunov-type methods paired with Riemann solvers (*i.e.*, HLLC, HLLC, Roe, Osher), while the high-order dispersive terms are computed using FD approximations. These hybrid solvers have shown reliable performance in modeling breaking waves and coastal inundation thanks to their shock-capturing properties(*e.g.*, Erduran et al. [2005]; Shiach and Mingham [2009]; Tonelli and Petti [2009]; Roeber et al. [2010]; Roeber and Cheung [2012]; Kazolea and Delis [2013]; Roeber and Bricker [2015]).

The standard Godunov-type formulation has only first-order accuracy (Toro [2013]). To achieve improved accuracy, high-order reconstruction techniques along with TVD slope limiters have been proposed (*e.g.*, Jiang and Shu [1996]; Kim and Kim [2005]; Choi et al. [2018]). In recent years, the use of high-order techniques in the computation of dispersive wave propagation has acquired immense popularity. Several Boussinesq models have employed high-order shock-capturing methods to achieve low-dissipative solutions with fewer grid nodes. The fourth-order MUSCL-TVD scheme proposed by Yamamoto and Daiguji [1993] has been used in many Boussinesq implementations and remains a popular choice for approximating the dispersive conservative equations (*e.g.*, Erduran et al. [2005]; Tonelli and Petti [2009]; Shi et al. [2012]). Another technique, based on the WENO scheme (*i.e.*, Jiang and Shu [1996]) has been used successfully to solve the weakly nonlinear Nwogu's equations (Zhou et al. [2016]). The Multi-dimensional Limiting Process (MLP) proposed by Kim and Kim [2005] offers an alternative shock-capturing strategy to control spurious oscillations in a two-dimensional setting. A fifth-order MLP reconstruction has been effectively implemented by Roeber and Cheung [2012] in their 2D Boussinesq model.

In this section, we look at how the hyperbolic solver affects the quality of the Boussinesq models. We focus on a fundamental yet often overlooked issue, which is numerical diffusion. Nowadays, properties such as low dissipation have become increasingly crucial in phase-resolving wave computations. Due to the enhanced dispersion properties of the new Boussinesq equations, their applicability has been extended to deeper water, allowing the models to account for far-field wave propagation. As a result, the accumulative effects of numerical diffusion can have a significant impact on the quality of the results.

4.2 Hybrid FV/FD Solver

We contrast the numerical approach used in our model (Chapter 2) with other well-established techniques adopted by several Boussinesq solvers. The hybrid approach, where shock-capturing FV methods are used to compute numerical fluxes has become increasingly popular in recent years. These schemes are based on the characteristic decomposition of the governing equations and have proven stable in computing nearshore processes such as wave breaking and run-up.

We solve 1D Nwogu's equations with two FV methods: HLLC Riemann solver and Central Upwind scheme (*i.e.*, Kurganov et al. [2001]). To attain high-order accuracy, we implement a second, third, and fifth-order reconstruction based on the MLP method (*i.e.*, Kim and Kim [2005]).

4.2.1 Governing Equations

The 1D Nwogu's equations can be written in the following conservative form:

$$U_t + F_x + S = 0, \tag{4.1}$$

where:

$$U = \begin{bmatrix} H \\ Hu + H\psi_m \end{bmatrix}, \qquad F = \begin{bmatrix} Hu \\ Hu^2 + \frac{1}{2}gH^2 \end{bmatrix}, \qquad S = \begin{bmatrix} \psi_c \\ -gHh_x + u\psi_c - H_t\psi_m \end{bmatrix}, \tag{4.2}$$

$$\psi_c = \left[\left(\frac{z_\alpha^2}{2} - \frac{h^2}{6} \right) hu_{xx} + \left(z_\alpha + \frac{h}{2} \right) h \left(hu \right)_{xx} \right]_x, \tag{4.3}$$

$$\psi_m = \frac{z_\alpha^2}{2} u_{xx} + z_\alpha \left(hu \right)_{xx}. \tag{4.4}$$

The method of characteristics requires the homogeneous part of the equation to be isolated. Therefore, to preserve the hyperbolicity of the hydrostatic component, the pressure term $gH\eta_x$ is split into an artificial flux gradient and a source term.

4.2.2 Godunov-type Scheme

The Finite Volume Method divides the computational domain into discrete volumes that store the average values of each conserved variable. These values are then updated in time based on the numerical flux coming in and out of the cell. It's important to note that most FV schemes are based on the collocation of the flow variables. The integrated form of the conserved equations (Eq. (4.1)) is written as:

$$\frac{U_i^{n+1} - U_i^n}{\Delta t} + \frac{1}{\Delta x} \left[F_{i+\frac{1}{2}}^n - F_{i-\frac{1}{2}}^n \right] + S_i^n = 0. \tag{4.5}$$

The numerical flux $F^n_{i+\frac{1}{2}}$ approximates the solution of the local Riemann problem that occurs at each cell interface.

HLLC approximate Riemann solver

Harten et al. [1983] proposed an approximate solution to the Riemann problem, which was later corrected by Fraccarollo and Toro [1995] to account for the influence of intermediate waves. The HLLC numerical flux is computed as follows:

$$F^{hllc}_{i+\frac{1}{2}} = \begin{cases} F_L & \text{if} \quad 0 \leqslant S_L \\ F^*_L & \text{if} \quad S_L \leqslant 0 \leqslant S^* \\ F^*_R & \text{if} \quad S^* \leqslant 0 \leqslant S_R \\ F_R & \text{if} \quad S_R \leqslant 0 \end{cases}, \qquad \text{where} \qquad \begin{cases} F^*_L &= F_L + S_L \left(U^*_L - U_L\right) \\ F^*_R &= F_R + S_R \left(U^*_R - U_R\right) \end{cases}. \tag{4.6}$$

The state U^*_L and U^*_R are given by:

$$U^*_K = H_K \left(\frac{S_K - u_K}{S_K - S^*}\right) \begin{bmatrix} 1 \\ S^* \end{bmatrix}, \qquad K = L, R. \tag{4.7}$$

We compute the wave speeds S_L, S_R, and S^* based on the formulations given in Toro [2001].

Central-upwind scheme

The numerical flux $F^n_{i+\frac{1}{2}}$ can also be computed with a Riemann-free solver such as central-upwind scheme (*i.e.*, Kurganov and Petrova [2007]):

$$F^{cu}_{i+\frac{1}{2}} = \frac{a_R F_L - a_L F_R}{a_R - a_L} + \frac{a_R a_L}{a_R - a_L} \left(U_R - U_L\right), \tag{4.8}$$

where:

$$a_R = \max\left(0, u_R + \sqrt{gH_R}, u_L + \sqrt{gH_L}\right), \tag{4.9}$$

$$a_L = \min\left(0, u_R - \sqrt{gH_R}, u_L - \sqrt{gH_L}\right). \tag{4.10}$$

This family of central schemes avoids the complex and time-consuming calculation of the Eigen system and hence provides a much simpler alternative to upwind approaches.

Source terms

The dispersion terms in Nwogu's equations are computed with central FD approximations. It's important to note that in the case of a non-flat bottom, the computation of the topography variation term gHh_x is not straightforward and requires the use of specialized techniques to

achieve a well-balanced and positivity-preserving model (*e.g.*, Audusse et al. [2015]; Chertock et al. [2015]; Liang and Marche [2009]).

4.2.3 Flux Reconstruction

To compute the numerical flux, we need to approximate the state variables at the cell interface. A first-order approximation can be achieved by taking the values from the left and right cells as input into the local Riemann solver:

$$\left[U_L\right]_{i+\frac{1}{2}} = U_i, \qquad \left[U_R\right]_{i+\frac{1}{2}} = U_{i+1}. \tag{4.11}$$

For improved accuracy, several high-order schemes have been proposed. These methods employ flux limiters to achieve oscillation-free computations:

Second-order reconstruction - MUSCL

The second-order reconstruction of the state variables at the cell interface can be calculated with:

$$\left[U_L\right]_{i+\frac{1}{2}} = U_i + \frac{1}{2}\phi\left(r_L\right)\left(U_i - U_{i-1}\right), \tag{4.12}$$

$$\left[U_R\right]_{i+\frac{1}{2}} = U_{i+1} - \frac{1}{2}\phi\left(r_R\right)\left(U_{i+2} - U_{i+1}\right), \tag{4.13}$$

where:

$$r_L = \frac{U_{i+1} - U_i}{U_i - U_{i-1}}, \qquad r_R = \frac{U_{i+1} - U_i}{U_{i+2} - U_{i+1}}. \tag{4.14}$$

ϕ is a TVD slope limiter. Here we use Van-Leer's one-parameter MinMod limiter, which is defined as follows:

$$\phi\left(r, \theta\right) = \max\left(0, \min\left(\theta r, \frac{1+r}{2}, \theta\right)\right). \tag{4.15}$$

The slope limiter is most dissipative for $\theta = 1$ and is least dissipative for $\theta = 2$. We set $\theta = 2$ for all the following numerical tests.

Third-order reconstruction

Kim and Kim [2005] proposed a TVD third-order interpolation for the flux computation:

$$\left[U_L\right]_{i+\frac{1}{2}} = U_i + \frac{1}{2}\max\left(0, \min\left(2, 2r_{L,i}, \beta_L\right)\right)\left(U_i - U_{i-1}\right), \tag{4.16}$$

$$\left[U_R\right]_{i+\frac{1}{2}} = U_{i+1} - \frac{1}{2}\max\left(0, \min\left(2, 2r_{R,i+1}, \beta_R\right)\right)\left(U_{i+2} - U_{i+1}\right). \tag{4.17}$$

The values of β_L and β_R are calculated as follows:

$$\beta_L = \frac{1 + 2r_{L,i}}{3}, \qquad \beta_R = \frac{1 + 2r_{R,i+1}}{3}, \tag{4.18}$$

where:

$$r_{L,i} = \frac{U_{i+1} - U_i}{U_i - U_{i-1}}, \qquad r_{R,i+1} = \frac{U_{i+1} - U_i}{U_{i+2} - U_{i+1}}. \tag{4.19}$$

Fifth-order reconstruction

To achieve fifth-order accuracy, the state variables are reconstructed as:

$$\left[U_L\right]_{i+\frac{1}{2}} = U_i + \frac{1}{2}\max\Big(0, \min\left(2, 2r_{L,i}, \beta_L\right)\Big)\left(U_i - U_{i-1}\right), \tag{4.20}$$

$$\left[U_R\right]_{i+\frac{1}{2}} = U_{i+1} - \frac{1}{2}\max\Big(0, \min\left(2, 2r_{R,i+1}, \beta_R\right)\Big)\left(U_{i+2} - U_{i+1}\right). \tag{4.21}$$

The values of β_L and β_R are calculated as follows:

$$\beta_L = \frac{-2/r_{L,i-1} + 11 + 24r_{L,i} - 3r_{L,i}r_{L,i+1}}{30}, \qquad \beta_R = \frac{-2/r_{R,i+2} + 11 + 24r_{R,i+1} - 3r_{R,i}r_{R,i+1}}{30}, \tag{4.22}$$

$$r_{L,i} = \frac{U_{i+1} - U_i}{U_i - U_{i-1}}, \qquad r_{R,i} = \frac{U_i - U_{i-1}}{U_{i+1} - U_i}. \tag{4.23}$$

4.3 Numerical Results

When solving time- and space-dependent PDEs such as the Boussinesq-type equations, it's important to account for the effects of numerical dissipation. The diffusion error stems from the discretized form of the equations, which behaves differently than the original continuous equations. This discrepancy has a cumulative impact and can, in some cases, compromise the validity of the numerical solution. While reducing the grid size does effectively minimize this problem, it can significantly increase the computational cost and therefore affect the efficiency of the model. A low diffusion scheme is thus essential to counter this issue.

In the context of Boussinesq-type models, the effects of numerical dissipation are more important when dealing with periodic waves which involve recurring extrema (wave crest and through). We compare the diffusion effects of different numerical schemes for long-distance wave propagation.

4.3.1 Monochromatic Wave Propagation

In a long flat channel, we generate and propagate a 4 s monochromatic wave. The water depth is $d = 1\,\mathrm{m}$, resulting in $kh = 0.5$.

First, we investigate the effects of the flux approximation on the quality of the solution. For that, we compute the numerical solution of Nwogu's equations with two hyperbolic solvers: the

HLLC scheme and the Central Upwind scheme (*i.e.*, Kurganov scheme). dispersion terms. Both approaches use identical elliptic solvers along similar FD approximations for the dispersion terms. For this test, we use a fifth-order reconstruction and we compute the wave propagation with a grid size $\lambda/\Delta x = 30$, where λ is the wavelength.

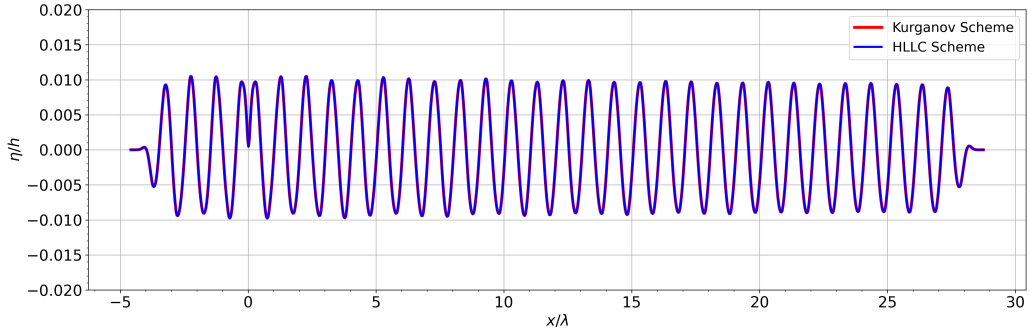


Figure 4.1: Sine wave propagation of $kh = 0.5$: Comparison between HLLC and Kurganov schemes for a grid size $\lambda/\Delta x = 30$ and fifth order-reconstruction

The numerical results from the two Godunov-type schemes are identical. Proving that for a smooth wave and constant water depth, these two approaches collapse into the same solution.

In this second part, we examine the effect of the flux reconstruction on the accuracy of the results. We run the same test as before with the HLLC scheme and compare the wave propagation with different orders of reconstructions. The results are shown in Figure 4.2.

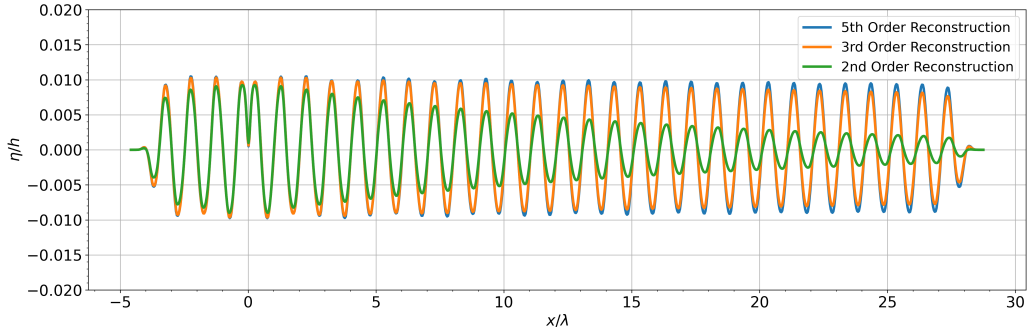


Figure 4.2: Sine wave propagation of $kh = 0.5$: Comparison between different orders of flux reconstructions for a grid size $\lambda/\Delta x = 30$

The effect of numerical diffusion is more pronounced with lower-order reconstruction, where the wave height drastically decreases and the wave completely vanishes after a few cycles. High-order methods, such as the fifth-order reconstruction, help conserve the wave height and, therefore, the wave energy. In practice, the use of Godunov-type models requires at least a fifth-order reconstruction to conserve the wave energy after 25 cycles for a $\lambda/\Delta x = 30$ grid resolution.

The grid size is generally recognized to have a significant influence on numerical dissipation. In this section, we compare the results from the Present model with the standard HLLC imple-

mentation for different grid resolutions. Since the Present model is not based on the method of characteristics for the solutions of the hyperbolic equations, the free surface gradient (*i.e.*, hydrostatic pressure) is retained as one term (not split) and computed with a central difference approximation. This is consistent with the approximation of the dispersion variables (*i.e.*, non-hydrostatic pressure). On the other hand, the Godunov-type solvers require the splitting of the hydrostatic pressure into an artificial flux and a source term and therefore compute the pressure flux with approximate Riemann solvers. It is important to note that the Present model uses a second-order reconstruction to approximate the convective acceleration, while the HLLC scheme employs a fifth-order reconstruction to compute the numerical flux.

On a very fine grid $\lambda/\Delta x = 60$, the two schemes converge to the same solution. The wave height, in this case, is well computed for both models. With a coarser grid ($\lambda/\Delta x = 15$), we can notice a significant discrepancy between the two solutions. The present model is able to maintain the correct wave height even after 25 cycles, whereas, at this resolution, the HLLC scheme fails to propagate the wave with the correct amplitude, and the dissipative effects are more important.

The sine wave propagation is an important test, since it showcases not only the effects of the diffusive errors on the numerical solution but also the dispersive errors. The diffusion error has a tendency to decrease the wave amplitude after each cycle, whereas the dispersion error alters the wavelength. For this test, a wave with 4 s period results in $kh = 0.5$. Nwogu's dispersion error for this value of kh is extremely small, and thus we expect a near-perfect match with Airy wave theory. As a result, any changes in the wavelength are mainly due to the numerical discretization. As expected, with a very fine grid (*i.e.*, $\lambda/\Delta x = 60$), the wave travels not only with the correct wave amplitude but also with the correct wavelength. The current model can maintain the correct wavelength with $\lambda/\Delta x = 30$ grid resolution, whereas the numerical errors of the HLLC scheme have a tendency to stretch the wave and overestimate its length. The dispersion errors of the HLLC scheme are more noticeable with the coarse grid (*i.e.*, $\lambda/\Delta x = 15$) where the generated wave becomes much longer than the exact solution. The Present scheme also contains dispersive errors due to the second-order central difference approximation of the pressure terms. However, these errors tend to rather shorten the wave and the magnitude of the errors is still minor in comparison to the error induced by the HLLC scheme.

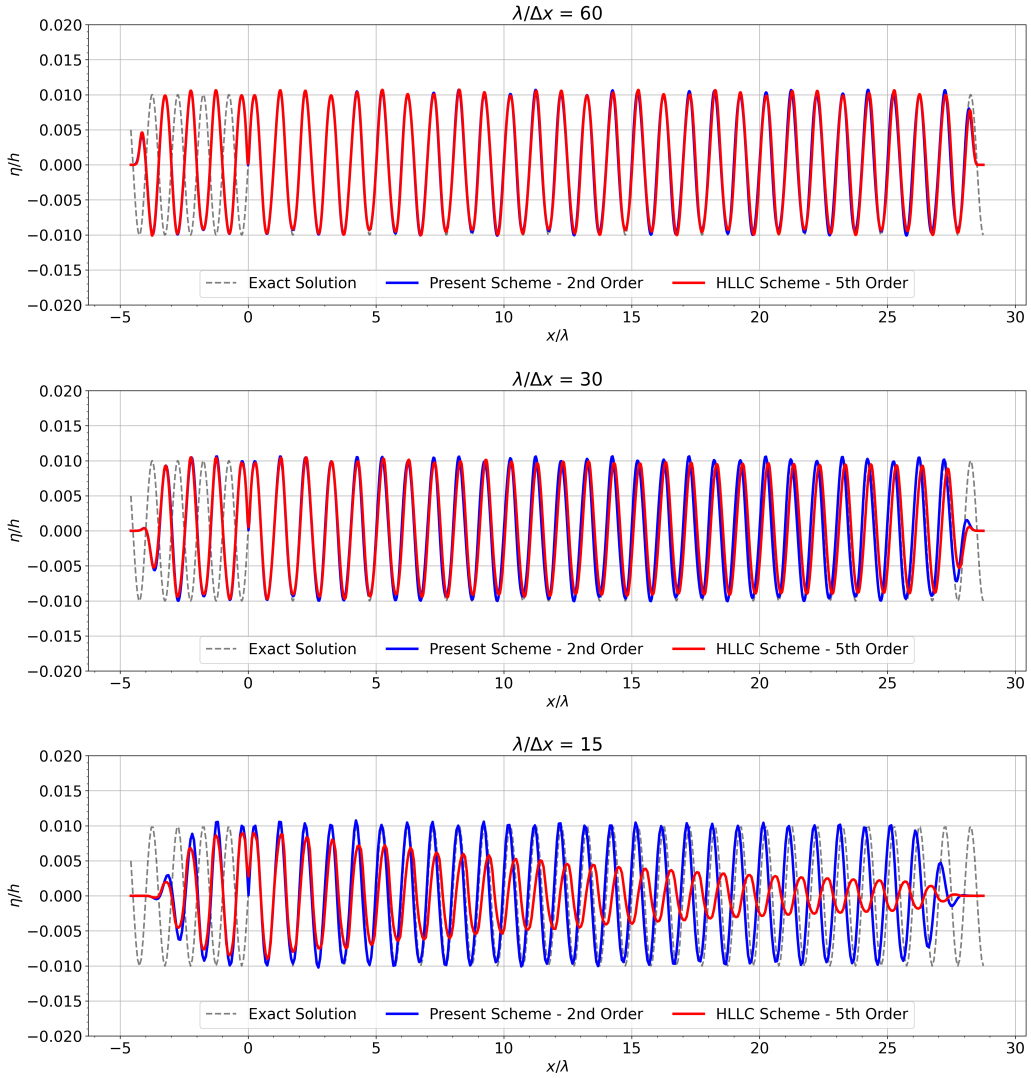


Figure 4.3: Sine wave propagation of $kh = 0.5$: Comparison between Present model and HLLC scheme for different grid sizes

4.3.2 Irregular Wave Propagation

In the previous section, we have tested the effect of numerical diffusion on the propagation of monochromatic waves. In this following section, we investigate how the numerical diffusion translates into a more practical case: a wave time series from a spectrum. We generate an irregular wave field based on the JONSWAP empirical spectrum (Appendix E) with a significant wave height of $H_s = 1\,\mathrm{m}$ and a peak period of $T_p = 15.7\,\mathrm{s}$ and we examine three configurations:

Table 4.1: Generation of JONSWAP wave spectrum: model configurations

	Case 1	Case 2	Case 3
Water depth h	50 m	25 m	25 m
Grid size Δx	10 m	10 m	5 m
Peak wavelength λ_p	300 m	230 m	230 m
$\lambda_p/\Delta x$	30	23	46
High-frequency cut λ_{min}	100 m	50 m	50 m
$\lambda_{min}/\Delta x$	10	5	10

In a 25 km long flat channel, we record the free-surface at three different locations: at 2 km, 10 km, and 20 km from the wavemaker. For each run, the Welch method is used to extract the power spectral density from the time-series.

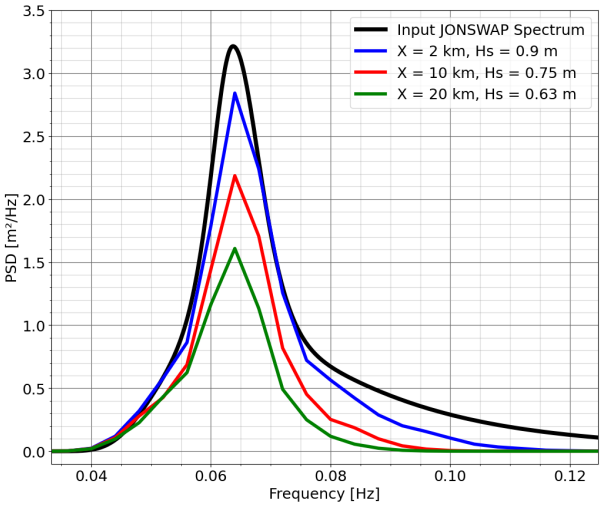


Figure 4.4: Case 1: Godunov-type model with fifth-order flux reconstruction

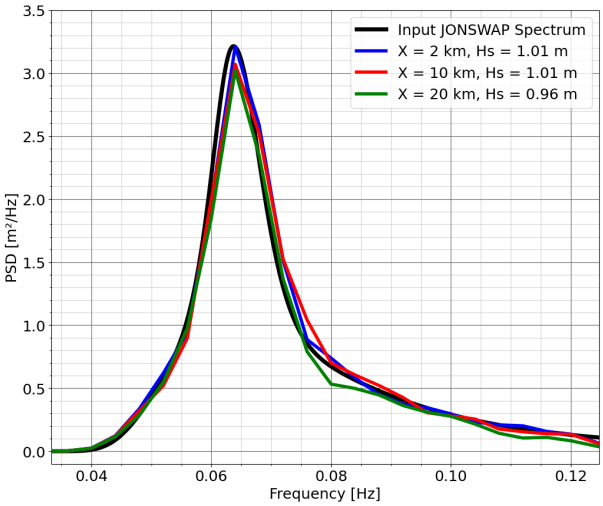


Figure 4.5: Case 1: Present model with second-order flux reconstruction

As shown in Figures 4.4 and 4.5, and for a 50 m water depth (*i.e.*, Case 1), the overall energy of the wave field decays drastically with the HLLC Riemann solver. At a distance of 20 km from the wavemaker, the loss of total wave energy is 47 %. The high-frequency tail is disproportionately impacted by the numerical dissipation, and the spectrum's shape is strongly distorted. In contrast, with the conservative staggered scheme, the energy and the spectral shape are well preserved, which proves the low diffusivity of the Present approach.

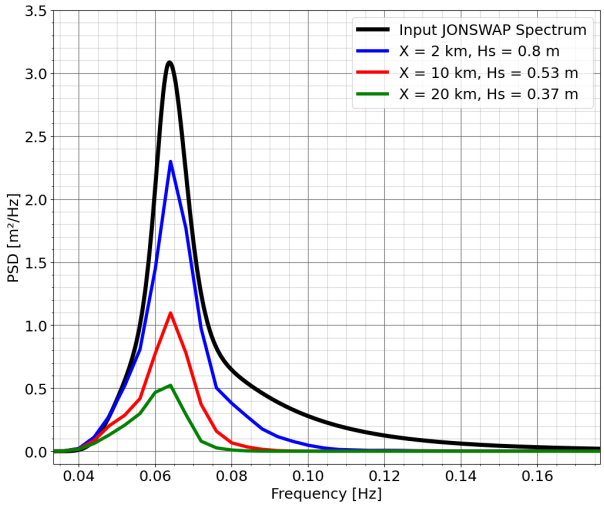


Figure 4.6: Case 2: Godunov-type model with fifth-order flux reconstruction

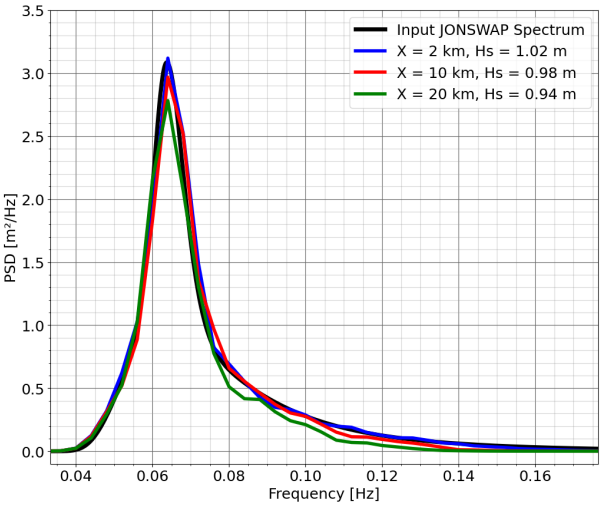


Figure 4.7: Case 2: Present model with second-order flux reconstruction

In shallower water, at a depth of 25 m, the effect of the energy dissipation is more pronounced, as the spectra now include shorter waves than in the example above. The high-frequency cut at $h = 25\,\mathrm{m}$ is $\lambda_{min} = 50\,\mathrm{m}$, whereas in $h = 50\,\mathrm{m}$ water depth $\lambda_{min} = 100\,\mathrm{m}$. Since in this case the short waves contain more energy than in the previous example, we can see that the energy loss is more substantial. With an HLLC solver, 63% of the total wave energy is lost after 20 km of propagation. In contrast, the staggered scheme experiences only minor diffusion and the overall shape of the spectrum is still well preserved. (see Figures. 4.6, 4.7)

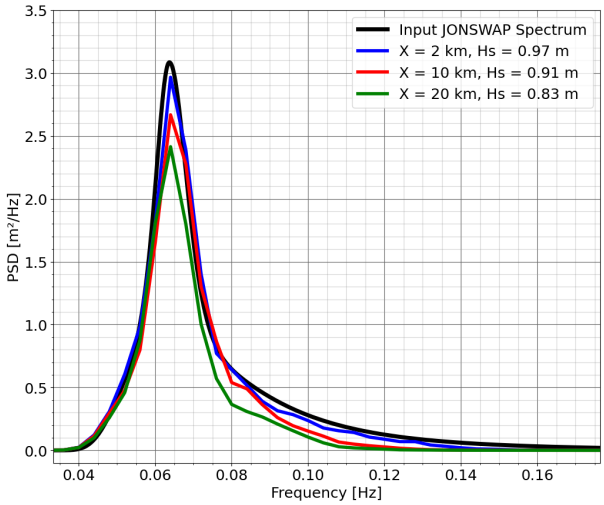


Figure 4.8: Case 3: Godunov-type model with fifth-order flux reconstruction

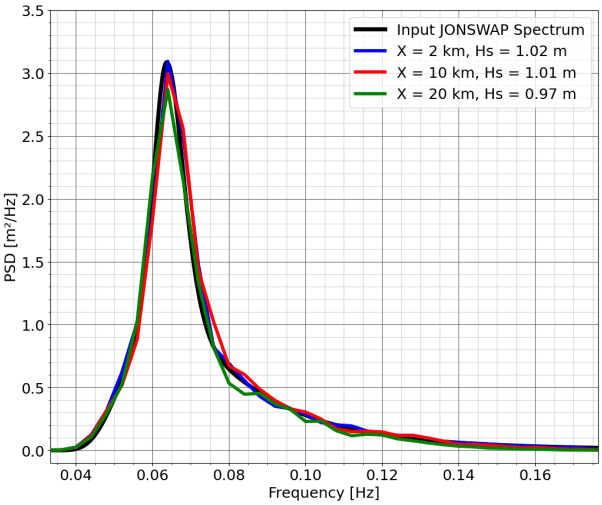


Figure 4.9: Case 3: Present model with second-order flux reconstruction

When the mesh is refined to $\Delta x = 5\,\mathrm{m}$, we can notice a significant improvement in the results for the HLLC scheme, which proves that the energy dissipation observed is numerical and is due to diffusion errors of the particular scheme. The results with the Present model are also slightly

improved but less noticeable than for the HLLC. In general, the HLLC scheme seems to be much more sensitive to the grid size for problems with periodic wave propagation.

4.4 Conclusions

We have compared the standard FV-FD hybrid method based on an HLLC Riemann solution to the conservative staggered scheme (*i.e.*, Present model) in solving Nwogu's equations. The main difference between these two techniques is the computation of the hydrostatic flux, *i.e.*, essentially the computation of the subset of SWE in the Boussinesq-type system. While the hybrid method computes the flux using a Godunov-type scheme along with high-order reconstruction, the Present model uses FD approximations that satisfy the Rankine-Hugoniot jump condition to estimate the convective acceleration (see Chapter 2). Both techniques are shock-capturing and provide stable and robust solutions for breaking waves, bores, and hydraulic jumps. Although Godunov-type schemes are well-established for the computation of flow discontinuities and shocks, the performance of these methods has never been tested for periodic waves—especially in the context of long-distance propagation. Compared to the conservative staggered scheme, these characteristics-based methods are very sensitive to the grid size and require a rather fine grid size ($\lambda/\Delta x \approx 60$) to be able to correctly propagate the wave energy to shore.

The use of low-dissipation solvers, such as the Present scheme, improves not only the accuracy of the results but also the computational efficiency. The scheme is able to conserve wave energy over a full spectrum of frequencies and captures essential wave processes with a rather coarse grid. Moreover, the lean structure of the FD approximations significantly reduces the computation time, making it an ideal choice for use in an operational model.

It should be noted that Godunov-type schemes based on Riemann solvers are not per se prone to errors. The scheme is well-established, and it works for the purposes it was designed for—namely, the computation of flow discontinuities. However, we have shown that it is not that suitable for the computation of periodic waves where wave breaking only occurs very locally and momentarily and where the wave field is mostly characterized by smooth solutions. The prevalence of HLLC schemes in nearshore wave models can likely be explained by the fact that methods from long-wave and river flooding models were ported over to the coastal engineering community where the method's applicability is simply more limited and where alternative schemes such as the one in the Present model are of superior quality.

The low numerical diffusivity of the Present scheme opens the door to many applications of a model in an operational framework. It is trivial that a low total number of cells reduces the overall computational load, independent of parallelization efforts. Nevertheless, it is even more important that the wave properties are maintained over the course of the computation so that the results remain trustworthy.

Chapter 5

Wave Breaking in Boussinesq-type Models

Contents

5.1 Introduction 95
5.2 Methodology 96
5.2.1 Identification of Wave Breaking Onset 96
5.2.2 Wave Breaking Closure 97
5.3 Numerical Tests 99
5.3.1 Laboratory Experiment 99
5.3.2 Model Settings 100
5.3.3 Numerical Results 101
5.4 Conclusions 103

5.1 Introduction

Wave breaking is a highly complex 3D process and a common challenge for Boussinesq-type models since the depth-integrated structure of the equations does not allow for the overturning of the free surface. With only one cell per grid point, the closest one can get to overturning is a vertical shock in the case of an ideally infinitely small grid cell. Discontinuous flow problems cater naturally to hyperbolic equations such as the Shallow Water Equations (SWE) (Toro [2001]). Inconveniently, the SWE are based on hydrostatic pressure assumptions and therefore have limited applicability for dispersive nearshore waves. A Boussinesq-type or non-hydrostatic equation involves elliptic terms for the non-hydrostatic pressure correction and does not directly allow for discontinuous profiles. As a result, the converged solution of the dispersive equation along the breaking wave's leading edge is a sharp spike on the verge of singularity. Since diffusive processes are not per se part of the equations, this mathematically correct behavior is not accurate from a physical perspective.

Multiple methods have been proposed to handle wave breaking in Boussinesq-type and non-hydrostatic models and have proven to provide reasonable solutions for breaking waves and subsequent second-order processes such as IG-wave generation (*e.g.*, Yamazaki et al. [2011]; Roeber and Cheung [2012]; Shi et al. [2011]). In general, they utilize two opposing strategies: (a) addressing the potentially arising singularity through diffusive terms and (b) eliminating the singularity by locally and momentarily allowing for a hyperbolic solution through the deactivation of the dispersion terms in the governing equations.

Option (b) has become a standard procedure in several operational models (*e.g.*, Roeber [2010], Roeber and Cheung [2012], Kazolea and Delis [2013]). However, the locally sudden change between dispersive and non-dispersive solutions can pose problems and instabilities when the grid size is small and a breaking wave is described over multiple grid cells. Stable solutions therefore often rely on some degree of grid diffusion; hence, it is difficult to obtain converging results with a refined mesh.

The eddy viscosity concept (a), on the other hand, is based on applying an additional diffusion term to the cells across the wave-breaking zone. This is somewhat attractive since the governing equations remain intact and no incompatibility arises from the local deactivation of dispersion terms. The eddy viscosity term counter-balances the potentially arising instability along the wavefront. Though the diffusive source term is relatively straightforward, the assessment of its magnitude remains challenging. One standard approach involves expressing the eddy viscosity term as a function of the time derivative of the free surface elevation (Kennedy et al. [2000]). A more detailed wave-breaking closure stems from the Kolmogorov-Prandtl equation, which relates the eddy viscosity to turbulent kinetic energy (TKE). This method requires the solution of an additional transport equation to compute the temporal and spatial variation of the TKE quantity.

In this section, we detail the implementation of a TKE-based eddy viscosity approach to handle wave breaking in the present Boussinesq-type model. The quality of grid convergence with this method is then examined through laboratory data and compared to other standard approaches, such as the hybrid method. It is further discussed under what condition the usage of this approach is recommended and where standard established techniques are sufficient.

5.2 Methodology

5.2.1 Identification of Wave Breaking Onset

In the past years, special workshops and seminars have addressed the definition and quantification of the onset of wave breaking based on observations and numerical modeling (*e.g.*, B'Waves workshops). It has been notoriously challenging to formulate a unified approach, in particular for depth-integrated models. Recent advances based on potential flow solutions have shown that kinematic and dynamic criteria provide more consistent information on the definition of the breaking onset than formerly used geometric criteria, which mostly use the free surface slope as an indicator for wave breaking (*e.g.*, Varing et al. [2021], Barthelemy et al. [2018]). Here we present two common wave-breaking onset criteria:

Surface elevation to depth ratio

One of the fundamental criteria for defining the onset of wave breaking is the wave height to water depth ratio. The value of 0.78 for A/h is a widely accepted threshold, though it was originally based on the breaking of solitary waves (Russell [1845]). For periodic waves, this criterion requires the computation of the total water height and thus the tracking of the crest and trough positions, which can be a tedious effort in numerical models. A simplified version of the wave height to water depth ratio is the relation of the free surface to water depth, which essentially only takes the wave crest but not the trough into account. By calibrating the threshold parameter, the hybrid approach, which involves switching back and forth between the hydrostatic and non-hydrostatic equations, performed reasonably well under this criterion (Shi et al. [2012]; Tonelli and Petti [2009]). The numerical cell is flagged if

$$\frac{\eta}{h} > C_b, \qquad \text{with} \quad C_b = 0.78. \tag{5.1}$$

Froude number

This criterion involves the Froude number values at the free surface, which can be determined from the vertical variation of the flow velocity. The quadratic assumption of the velocity profile is inherent to the governing equations and allows for calculation of the flow velocity at any position in the water column. For Nwogu's equations, the flow velocity is expressed as:

$$\mathbf{u}|_z = \mathbf{u}_\alpha + \left(\frac{z_\alpha^2}{2} - \frac{z^2}{2}\right) \nabla\left(\nabla.\mathbf{u}_\alpha\right) + \left(z_\alpha - z\right) \nabla\left(\nabla.(h\mathbf{u}_\alpha)\right). \tag{5.2}$$

Here, $\mathbf{u}_\alpha$ is the horizontal flow speed at the reference depth z_α (around mid-depth), and h is the local water depth.

In 1D setting, the equation obviously reduces to

$$u|_z = u_\alpha + \left(\frac{z_\alpha^2}{2} - \frac{z^2}{2}\right) \left(u_\alpha\right)_{xx} + \left(z_\alpha - z\right) \left(hu_\alpha\right)_{xx}. \tag{5.3}$$

By replacing $z = \eta$ into Eqs. (5.2) and (5.3), we obtain the horizontal flow velocity at the free surface based on Nwogu's equations. This approximation is reasonably valid as long as the waves are not strongly nonlinear and dispersive. Other governing equations can lead to similar approximations (see Appendix A).

The free surface Froude number can then be determined from

$$Fr|_\eta = \frac{\sqrt{(u|_\eta)^2 + (v|_\eta)^2}}{\sqrt{gh}}. \tag{5.4}$$

We know that for $Fr|_\eta > 1$ the flow becomes supercritical, *i.e.*, the particle speed overtakes the wave's celerity, and wave breaking starts. Varing et al. [2021] and others have shown that for $Fr|_\eta > 0.85$, waves are inevitably approaching the stage of breaking; *i.e.* a point of no return is passed and the wave will sooner or later break. However, it should be noted that celerity refers to shallow-water celerity under hydrostatic conditions. In two spatial dimensions, the actual celerity under moderately dispersive waves is difficult to find and usually requires a Lagrangian post-processing method. While $C = \sqrt{gh}$ is not exact, it is a very reasonable estimate.

5.2.2 Wave Breaking Closure

Deactivation of dispersion, hybrid

Depth-integrated models do not account for free surface overturning and thus cannot fully reproduce wave-breaking processes. The use of conserved variables in the governing equations along with shock-capturing schemes allows for the approximate description of breaking waves as discontinuous flows. Even though the flow is flux-dominated, the dispersive governing equations balance the high linearity with frequency dispersion. This leads to local anomalies that can result in numerical instability depending on the order of the dispersion terms, the numerical scheme, and most importantly, the grid size. The hybrid approach relies on the local and momentary deactivation of the dispersion terms to allow the hyperbolic solver to describe the breaking wave as a bore or hydraulic jump. The dispersion is therefore deactivated in all cells where the breaking criteria hold (Section 5.2.1).

Eddy viscosity closure, TKE

At the wave scale, the main effect of wave breaking is a strong energy dissipation at the wavefront. Zelt [1991] employed this property and proposed a wave-breaking closure by introducing a dissipation term into the momentum equation:

$$R_b = \begin{bmatrix} 0 \\ 2\Big[\nu_t H u_x\Big]_x + \Big[\nu_t H (u_y + v_x)\Big]_y \\ 2\Big[\nu_t H v_y\Big]_y + \Big[\nu_t H (u_y + v_x)\Big]_x \end{bmatrix}. \tag{5.5}$$

The computation of this source term requires the estimation of the eddy velocity coefficient ν_t, which controls the strength of the dissipation. The magnitude of the dissipative term depends

not only on the local variables (H, u, v) but also on the entire flow field, which is evolving over time. It is therefore necessary to provide a physical basis for ν_t to better estimate the turbulent effects. Kolmogorov [1941] and independently Prandtl et al. [1945] proposed an approximation which relate the eddy viscosity ν_t to the turbulent kinetic energy k and the mixing length l_t:

$$\nu_t = C_\nu \sqrt{k} \ell_t. \tag{5.6}$$

where, C_ν is a constant coefficient which is set to 0.55 according to many authors (Zhang and Wu [2011]; Kazolea and Ricchiuto [2018]; Pope [2000]). The mixing length l_t is set to the local water depth h. Pope [2000] proposed a transport equation for k to determine its spatial and temporal variation. The eddy viscosity term (Eq. (5.6)) is then computed and applied to the entire computational domain. In contrast to previous work, such as Kennedy et al. [2000], the eddy viscosity can have non-zero values even when the wave-breaking criterion is not met locally. It is possible that TKE remains in the system for some time or that it is advected outside the surf zone. The term can be seen as referring to the whitewater region, which is intense and very turbulent at the location of wave breaking but quickly settles down.

The governing equation for k is given by:

$$k_t = -\mathscr{A} - \mathscr{E} + \mathscr{P} + \mathscr{D}. \tag{5.7}$$

where $\mathscr{A}$, $\mathscr{E}$, $\mathscr{P}$, $\mathscr{D}$ denote advection, elimination/destruction, production, and diffusion of the turbulent kinetic energy. These terms have been detailed in Nwogu [1996]; Kazolea and Ricchiuto [2018] and involve mean flow quantities and several empirical coefficients.

The advection term $\mathscr{A}$ is defined as:

$$\mathscr{A} = \mathbf{u}.\nabla k. \tag{5.8}$$

The TKE elimination/destruction terms is denoted as:

$$\mathscr{E} = C_d \frac{k^{3/2}}{\ell_t}, \tag{5.9}$$

where, following Nwogu [1996]; Zhang and Wu [2011]; Kazolea and Ricchiuto [2018], we set this coefficient to: $C_d = C_v^3$.

The TKE diffusion term is small and given by

$$\mathscr{D} = \nabla.\left(\nu_t \nabla k\right). \tag{5.10}$$

The last term $\mathscr{P}$ denotes the production of turbulent kinetic energy. Since TKE production only occurs in cells with wave breaking, the formulation is multiplied by a coefficient B which works as a flag and takes on either 0 or 1. B is set to 1 when the breaking criterion holds. Similar to Nwogu [1996], we express the production term as:

$$\mathscr{P} = B \frac{\ell_t^2}{\sqrt{C_d}} \left[(u_z|_\eta)^2 + (v_z|_\eta)^2 \right]^{3/2}, \tag{5.11}$$

where $u_z|_\eta$ and $v_z|_\eta$ are the vertical gradients of the horizontal velocities at the free surface. These variables can be approximated using the irrotationality condition ($w_x = u_z, w_y = v_z$),

which yields:

$$\begin{aligned} u_z|_\eta &= -\eta \left[u_{xx} + v_{xy}\right] - \left[(hu)_{xx} + (hv)_{xy}\right], \\ v_z|_\eta &= -\eta \left[u_{xy} + v_{yy}\right] - \left[(hu)_{xy} + (hv)_{yy}\right]. \end{aligned} \tag{5.12}$$

Once the wave breaking criterion is flagged, the vertical gradients of the horizontal velocities in Eq. (5.11) are the primary contributors to the magnitude of $\mathscr{P}$. Energetic breaking, such as that seen in plunging breakers, increases the vertical gradient of horizontal velocities under the wave crest, resulting in an increase in u_z and v_z.

Do-nothing approach, rely on numerical diffusion

It is important to highlight that, without any wave-breaking treatment, the solution of the Boussinesq-type equations can approach a singularity. This is essentially dependent on the grid resolution. It was correctly stated by Kazolea and Ricchiuto [2018] that most previously published solutions in the context of wave breaking computed with dispersive depth-integrated equations were subject to at least some degree of numerical diffusion. The stability of the Boussinesq solution thus relies on the numerical discretization in clipping or diffusing the dispersion terms locally. In the absence of the latter, the do-nothing approach can be risky, particularly when dealing with fine grids and low-diffusion numerical schemes.

5.3 Numerical Tests

5.3.1 Laboratory Experiment

To understand the effects of the wave-breaking treatment on the composition of the wave field, we perform several numerical tests using the lab data collected during the DynaRev physical experiment that was carried out in the GWK flume in Hanover. A series of experimental tests were conducted in a 309 m long, 7 m deep, and 5 m wide flume equipped with a combined piston-flap type wavemaker to generate different wave conditions. A more detailed description of the laboratory experiments is provided in Blenkinsopp et al. [2019].

In this study, we focus on three experimental tests, each with different wave conditions and topography profiles. All the generated waves follow the distribution of the JONSWAP spectrum with varying significant wave height H_s and peak period T_p (see Table 5.1).

Table 5.1: Wave and topography inputs for each case configuration

Cases	**H_s [m]**	**T_p [s]**	**Slope**	**Iribarren**	**WL [m]**
Case 1	0.6	12	1/15	0.74	4.9
Case 2	0.8	6	1/15	0.56	4.5
Case 3	1.2	8	1/15	0.41	4.9

Outside the surf zone, the free surface time variation was measured by four gauges located at $x = 50$ m, $x = 160$ m, $x = 170$ m, and $x = 180$ m, respectively. The water elevation in the entire

surf and swash zones was measured at a high resolution using an array of three SICK LMS511 2D LiDAR scanners placed on the experimental roof. With a resolution of 10 cm, the scanners provide measurements of the free surface elevation throughout a continuous section of the surf zone, starting from the cross-shore position $x = 215\,\mathrm{m}$ up to the upper part of the beach at $x = 280\,\mathrm{m}$. For a more detailed comparison between the model results and measurements in the surf zone, we extracted six evenly spaced virtual gauges from the LiDAR data (gauges 4–9). The distance between two adjacent gauges is 5 m (Figure 5.1).

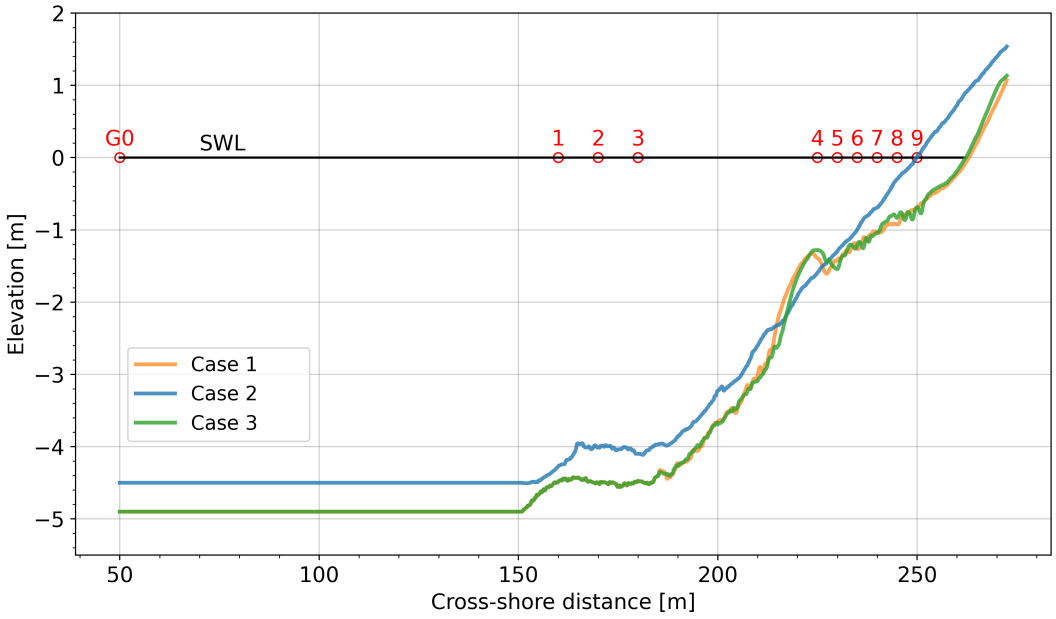


Figure 5.1: Bathymetry layouts with wave gauge positions for the three cases

5.3.2 Model Settings

We set up the Present model so that the upstream boundary lines up with the first gauge (*i.e.*, G0) in the physical experiment. The recorded free surface is then used as an input for the offshore wave boundary condition, allowing the model to be forced with the exact incident wave phases. The incident wave velocity at the offshore boundary is computed using the shallow water approximation, which is based on Airy wave theory and relates the wave speed to the local water depth:

$$u = \eta \sqrt{\frac{g}{h}}. \tag{5.13}$$

u denotes the depth-integrated horizontal velocity. A study of the time series signal shows that the incident wave contains mostly low frequencies ($kh < 0.25\pi$), which justifies the long wave assumption in this case. We run the three cases with a constant Manning friction coefficient of $n = 0.02\,\mathrm{s/m^{-1/3}}$ to account for the medium-grain sand roughness. We compute each test case with different wave-breaking methods (*i.e.*, TKE closure, hybrid approach, and no wave-breaking treatment) and grid sizes. The free surface time series is then recorded at each gauge for 20 min

with 25 Hz frequency sampling. A one-minute spin-up time is considered to allow for the wave field to be fully saturated.

In the eddy viscosity approach, the TKE production term is triggered depending on the Froude number criterion ($Fr|_\eta > 1$). A different criterion was chosen for the hybrid approach, where the local switching on and off of the dispersion terms is based on the water elevation to depth ratio ($a/h > 0.78$). Under this criterion, the hybrid method appears to perform better and produce more stable solutions (see Shi et al. [2011]; Tonelli and Petti [2009]).

5.3.3 Numerical Results

The time series is recorded at nine gauges for the three cases. Figure 5.2 shows a comparison between the computed significant wave height with different wave-breaking treatments and grid sizes. With a coarse grid (*i.e.*, $\Delta x = 100$ cm), the type of breaker seems to have little effect on the results and the overall wave energy. This shows that at a certain grid resolution, the numerical solution is able to mimic the turbulent dissipation effects by clipping the free surface at the over-shoaled wavefront. Even though the solution without a wave-breaking treatment matches the lab data quite well at this resolution, the accuracy of the result depends on the grid size and diverges quickly as the mesh gets finer, especially for more energetic waves (*e.g.*, **Case 3**). When no breaking treatment is used, the Boussinesq solution tends to overestimate the wave height and can become unstable (*e.g.*, **Case 3**, $\Delta x = 10$ cm). The model solution with local deactivation of dispersion exhibits similar tendencies, where the energy in the surf zone is over-predicted and computational instabilities arise when the mesh is refined. The instabilities are triggered by the mismatch between the hydrostatic and dispersive solutions, which becomes more pronounced with fine grid resolution. It is well understood that the added dissipation from the eddy viscosity terms counterbalances the dispersion variables, which tend to overshoot at the wavefront. The use of TKE as a parameter to estimate the magnitude of turbulent dissipation provides a more physical approach to dealing with wave-breaking effects in Boussinesq models. The convergence property of this approach is highlighted by the quality of the numerical results (blue marker), which illustrates its applicability to different wave conditions and grid sizes.

To gain more insight into the effects of the wave-breaking closure on the wave composition, we plot the free surface time series and spectrum at gauge 7 during case 2 computation (Figure 5.3). Other gauges and test cases show similar trends. The computed results with the TKE-based eddy viscosity treatment compare well to the laboratory data in terms of wave height and phase for various grid sizes. In the absence of a wave-breaking closure, the model overestimates the water elevation at the wavefront, especially with fine grids. This overshoot is offset by the additional dissipation introduced by the eddy viscosity term. Regarding the spectral composition, the eddy viscosity closure retains consistent results across various grid resolutions. This is not true for the other methods, where the solution diverges with mesh refinement. Along with the instability problems, the solution with the hybrid and no-breaking closures leads to an energy shift in the spectrum, notably the generation of false IG signals.

It should also be noted that the TKE dissipation affects smaller wave components that are not directly triggering the TKE production term. The presence of TKE in the system mimics the turbulent white water, which has dissipative properties without actually corresponding to an actively breaking wavefront. This can be seen in Figure 5.3 for frequencies higher than 0.2 Hz.

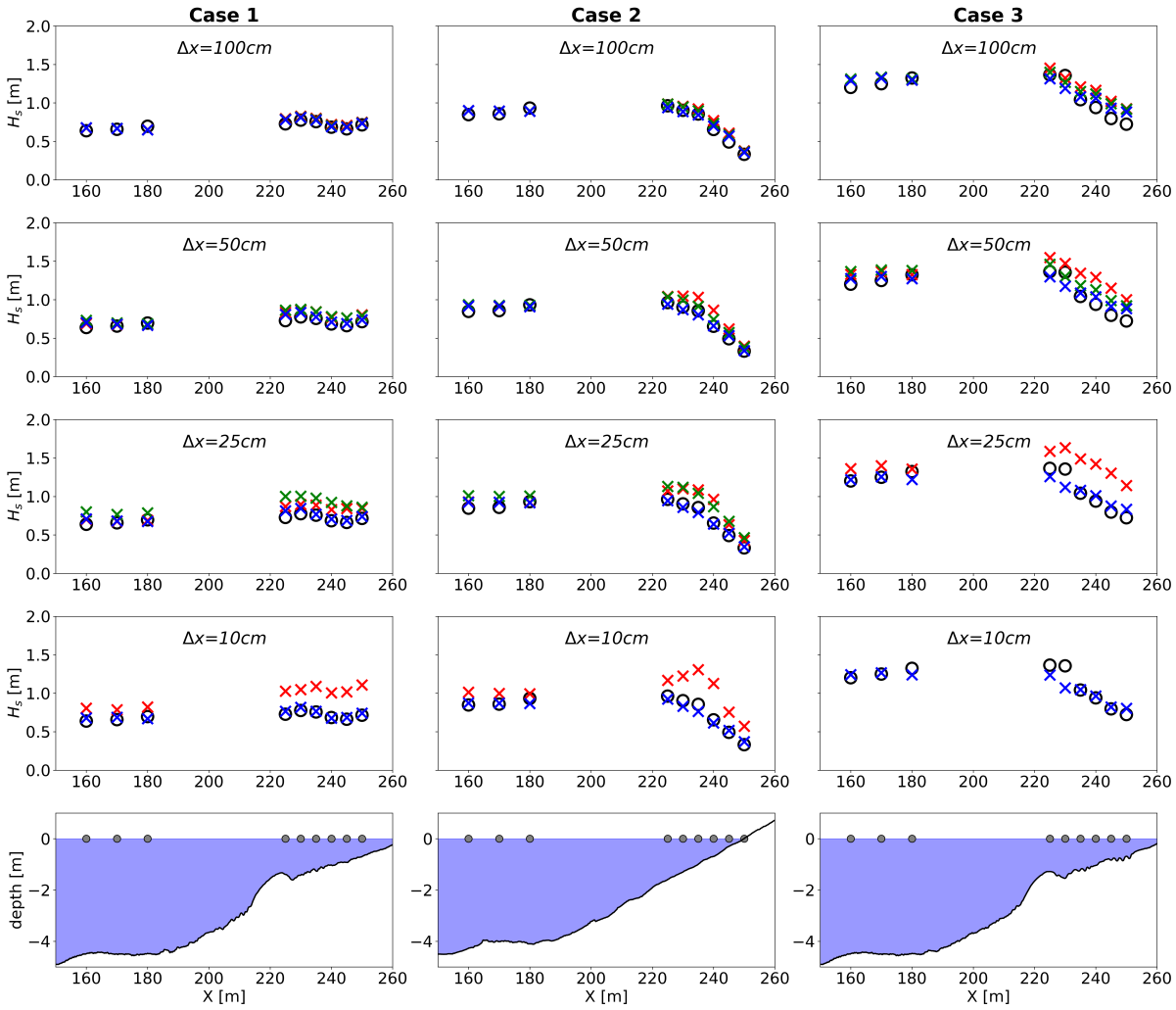


Figure 5.2: Comparison between the observed (o) and computed (x) significant wave heights at nine wave gauges for different wave and topography conditions. The blue, green, and red markers denote the solutions with the TKE closure, the hybrid approach, and no wave-breaking treatment, respectively.

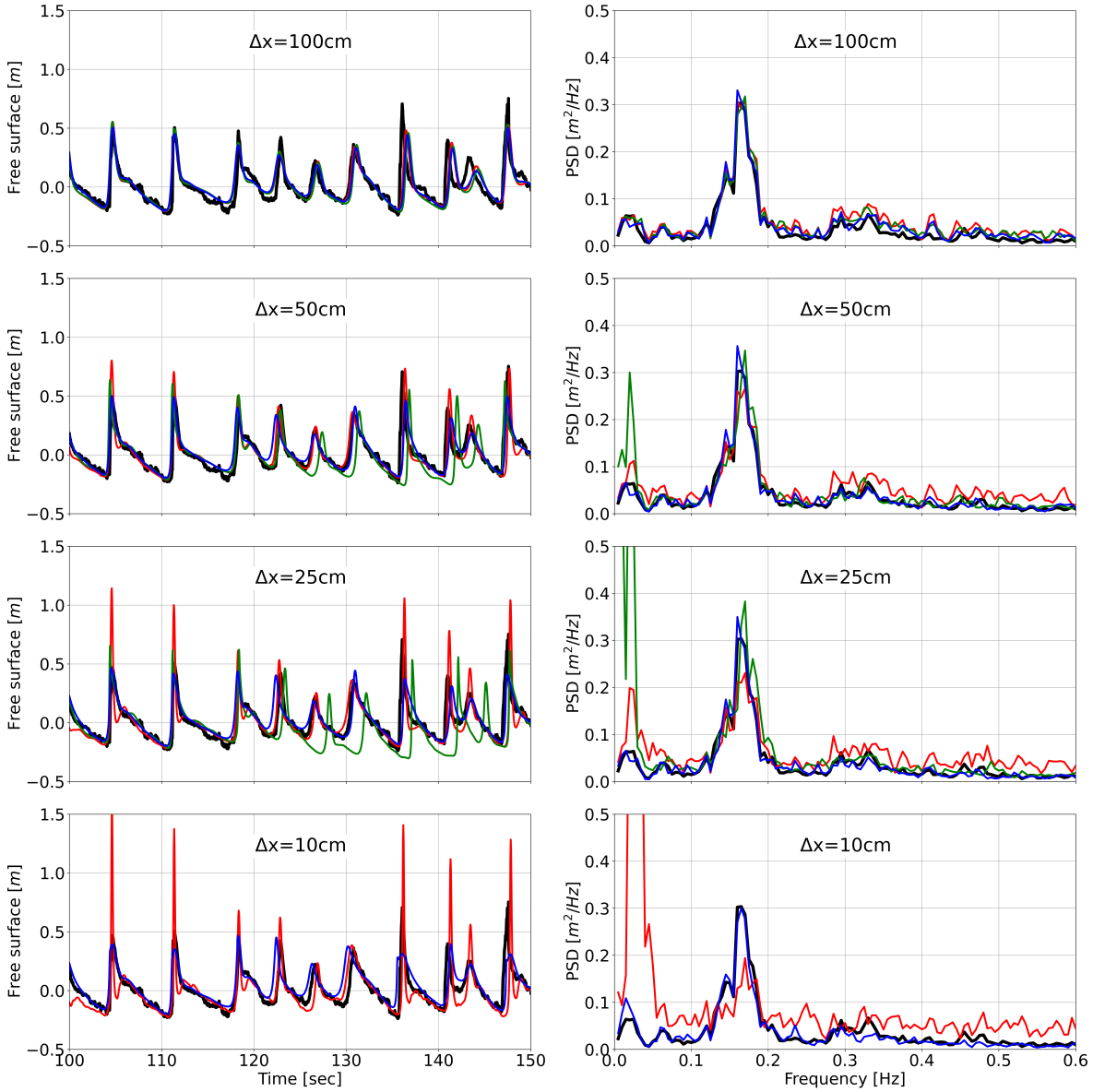


Figure 5.3: Comparisons between the observed (black lines) and computed (colored lines) spectra and time series for different grid sizes and wave-breaking treatments at **Gauge 7** for **Case 2**. The blue, green, and red lines illustrate the solutions with the TKE closure, the hybrid approach, and no wave-breaking treatment, respectively.

5.4 Conclusions

The performance of the TKE formulation for the wave-breaking closure has been examined with various numerical tests, including different wave and topographic conditions. The approach itself is a more complete closure since it mimics the physical dissipation that arises from the turbulence and mixing of the breaking wavefront. The results show good grid convergence, which supports its applicability to various problems with various grid resolutions.

Since the magnitude of the eddy viscosity is determined by the underlying TKE field, the dissipation effects are active in every cell where TKE is present. Even though the production of the TKE only occurs when the breaking criterion is met, the remaining terms in Eq. (5.7) distribute the quantity of TKE across an area of the surf zone. This particular feature is different from the conventional wave-breaking methods, where the treatments (deactivation of dispersion or eddy viscosity (Kennedy et al. [2000])) are only applied to the local wavefront but not beyond. This is more in line with the effects of the turbulent white water, which dissipates the wave energy over a large area of the surf zone and not just locally.

As demonstrated in the previous section (Chapter 4) the current model is built around a numerical scheme of low numerical diffusion. This is advantageous when investigating wave-breaking methods due to minimal numerical dissipation effects. Although the present analyses might vary with different numerical schemes, the observed trends should be representative of other models built around similar governing equations. A more diffusive numerical scheme could eventually provide more stable solutions with a refined mesh size, but would eventually also face stability issues and benefit from a TKE closure approach.

Chapter 6

2D Dispersive Wave Model: Implementation and Validation

Contents

6.1 Introduction . . . 106
6.2 Numerical Solution . . . 106
6.2.1 Governing Equations . . . 106
6.2.2 Numerical Scheme . . . 109
6.2.3 Wave Generation . . . 114
6.2.4 Absorbing Boundaries . . . 117
6.2.5 Wave Breaking . . . 118
6.3 Implementation Strategies for Computational Efficiency . . . 119
6.3.1 Floating-point Division . . . 121
6.3.2 Tri-diagonal Solver . . . 121
6.3.3 Wavemaker . . . 121
6.4 Numerical Tests . . . 122
6.4.1 Gaussian Drop Test . . . 122
6.4.2 Evolution of an Undular Bore . . . 124
6.4.3 Solitary Wave Runup on a Plane Beach . . . 125
6.4.4 HIreef - Solitary Wave over 2D Reef . . . 128
6.4.5 Breaking Solitary Wave over 3D Reef with a Cone . . . 131
6.4.6 Wave Refraction-Diffraction over an Elliptic Shoal . . . 136
6.5 Conclusions . . . 138

6.1 Introduction

In this section, we describe the numerical solution of the 2D Boussinesq-type equation from Nwogu [1993]. The implementation is based on an approach where the state variables are staggered in space and time. The model combines the efficiency of the staggered FD approximation with the conservation properties of the scheme that ensure accurate and stable near-shore wave computations (*e.g.*, wave breaking and run-up). Nwogu's equations are solved in their conservative form, and the hydrostatic part of the equations is computed with the SW solver described in Chapter 2. The implementation of essential model features such as wave generation, wave breaking, and wave absorption are also detailed. Several benchmark tests are then used to validate the new dispersive model.

6.2 Numerical Solution

6.2.1 Governing Equations

The long-established dispersive Boussinesq-type equations by Nwogu [1993] contain a continuity equation and two momentum equations. They express the motion of the free surface and its two orthogonal velocity components in the 2D horizontal plane while retaining a pseudo-3D structure through the incorporation of additional terms that account for non-hydrostatic pressure effects. The equations are derived under the assumptions of small amplitude and long period waves and are expressed as:

$$\eta_t + \nabla\left[(h+\eta)\mathbf{U}\right] + \nabla\cdot\left\{\left(\frac{z_\alpha^2}{2}-\frac{h^2}{6}\right)h\nabla\left(\nabla\cdot\mathbf{U}\right)+\left(z_\alpha+\frac{h}{2}\right)h\nabla\left[\nabla\cdot(h\mathbf{U})\right]\right\}=0, \tag{6.1}$$

$$\mathbf{U}_t + (\mathbf{U}.\nabla)\,\mathbf{U} + g\nabla\eta + \left\{\frac{z_\alpha^2}{2}\nabla(\nabla\cdot\mathbf{U}) + z_\alpha\nabla\left[\nabla\cdot(h\mathbf{U})\right]\right\}_t = 0. \tag{6.2}$$

h denotes the still water depth, which is assumed to be stationary (*i.e.*, $h_t = 0$). η is the free-surface elevation, and $\mathbf{U}$ is the horizontal flow velocity at z_α. The reference depth z_α is the vertical position within the still water column, h, at which the velocities are evaluated (see Figure 6.1). It is just below mid-depth and varies linearly with the bathymetry.

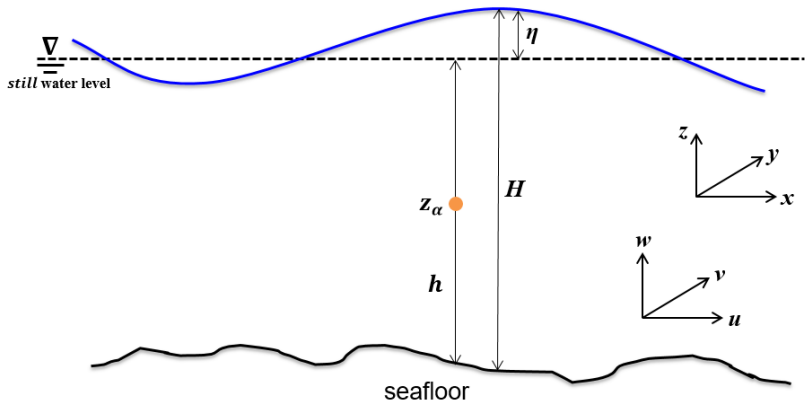


Figure 6.1: Definition sketch: State variables of Nwogu's equations

Conserved variable formulation:

Most Boussinesq-type and non-hydrostatic equations are derived from the Euler equations of motion and consequently lead to a formulation of the momentum equations in non-conserved form. The equations derived by Nwogu [1993] fall into this category. With directional derivatives, the equations are written as:

$$H_t + [Hu]_x + [Hv]_y + \psi_c = 0, \tag{6.3}$$

$$u_t + uu_x + vu_y + g\eta_x + [\psi_u]_t = 0, \tag{6.4}$$

$$v_t + uv_x + vv_y + g\eta_y + [\psi_v]_t = 0, \tag{6.5}$$

where $H = \eta + h$ is the total water depth. The dispersion terms are expressed as:

$$\begin{aligned}\psi_c = &\left[\left(\frac{z_\alpha^2}{2} - \frac{h^2}{6}\right) h\left(u_{xx} + v_{xy}\right) + \left(z_\alpha + \frac{h}{2}\right) h\left((hu)_{xx} + (hv)_{xy}\right)\right]_x \\ &+ \left[\left(\frac{z_\alpha^2}{2} - \frac{h^2}{6}\right) h\left(u_{xy} + v_{yy}\right) + \left(z_\alpha + \frac{h}{2}\right) h\left((hu)_{xy} + (hv)_{yy}\right)\right]_y,\end{aligned} \tag{6.6}$$

$$\psi_u = \frac{z_\alpha^2}{2}\Big[u_{xx} + v_{xy}\Big] + z_\alpha\Big[(hu)_{xx} + (hv)_{xy}\Big], \tag{6.7}$$

$$\psi_v = \frac{z_\alpha^2}{2}\Big[u_{xy} + v_{yy}\Big] + z_\alpha\Big[(hu)_{xy} + (hv)_{yy}\Big]. \tag{6.8}$$

The conservative form of the momentum equations can be derived by multiplying the Eqs. (6.4) and (6.5) by H:

$$\Big[Hu\Big]_t + \Big[Hu^2\Big]_x + \Big[Huv\Big]_y - u\left(H_t + \Big[Hu\Big]_x + \Big[Hv\Big]_y\right) + gH\eta_x + H\Big[\phi_u\Big]_t = 0, \tag{6.9}$$

$$\left[Hv\right]_t + \left[Huv\right]_x + \left[Hv^2\right]_y - v\left(H_t + \left[Hu\right]_x + \left[Hv\right]_y\right) + gH\eta_y + H\left[\psi_v\right]_t = 0. \tag{6.10}$$

By using the continuity equation (*i.e.*, Eq. (6.1)), we simplify the momentum equations into:

$$\begin{aligned} \left[Hu\right]_t + \left[Hu^2\right]_x + \left[Huv\right]_y + u\phi_c + gH\eta_x + H\left[\psi_u\right]_t = 0, \\ \left[Hv\right]_t + \left[Huv\right]_x + \left[Hv^2\right]_y + v\psi_c + gH\eta_y + H\left[\psi_v\right]_t = 0. \end{aligned} \tag{6.11}$$

It should be noted that the hydrostatic SWE in conserved form is a complete subset of the Boussinesq-type equation. Since the dispersion terms in the momentum equations involve temporal derivatives, the conserved variables in this case will vary from the standard SW conserved variables: H, Hu and Hv.

We divide the momentum dispersion variables into cross- and normal-derivative terms:

$$\psi_u = \psi_u^n + \psi_u^c, \qquad \text{where} \quad \begin{cases} \psi_u^n &= \frac{z_\alpha^2}{2} u_{xx} + z_\alpha \left(hu\right)_{xx} \\ \psi_u^c &= \frac{z_\alpha^2}{2} v_{xy} + z_\alpha \left(hv\right)_{xy} \end{cases}, \tag{6.12}$$

$$\psi_v = \psi_v^n + \psi_v^c, \qquad \text{where} \quad \begin{cases} \psi_v^n &= \frac{z_\alpha^2}{2} v_{yy} + z_\alpha \left(hv\right)_{yy} \\ \psi_v^c &= \frac{z_\alpha^2}{2} u_{xy} + z_\alpha \left(hu\right)_{xy} \end{cases}. \tag{6.13}$$

We rewrite the dispersion terms using Eqs. (6.12)-(6.13) and the product rule:

$$H\left[\psi_u\right]_t = H\left[\psi_u^n\right]_t + H\left[\psi_u^c\right]_t = \left[H\psi_u^n\right]_t - H_t\psi_u^n + H\left[\psi_u^c\right]_t, \tag{6.14}$$

$$H\left[\psi_v\right]_t = H\left[\psi_v^n\right]_t + H\left[\psi_v^c\right]_t = \left[H\psi_v^n\right]_t - H_t\psi_v^n + H\left[\psi_v^c\right]_t. \tag{6.15}$$

The dispersion terms $\left[H\psi_u^n\right]_t$ and $\left[H\psi_v^n\right]_t$ can be grouped with the local acceleration into evolution variables P and Q as:

$$P = Hu + H\psi_u^n, \qquad Q = Hv + H\psi_v^n. \tag{6.16}$$

The 2D Nwogu's equations can be cast in a conservative vector form as:

$$U_t + F\left(U\right)_x + G\left(U\right)_y + S\left(U\right) = 0, \tag{6.17}$$

where

$$U = \begin{bmatrix} H \\ P \\ Q \end{bmatrix}, \qquad F\left(U\right) = \begin{bmatrix} Hu \\ Hu^2 \\ Huv \end{bmatrix}, \qquad G\left(U\right) = \begin{bmatrix} Hv \\ Huv \\ Hv^2 \end{bmatrix}, \tag{6.18}$$

$$S(U) = \begin{bmatrix} \psi_c \\ gH\eta_x + u\psi_c - H_t\psi_u^n + H\left[\psi_u^c\right]_t \\ gH\eta_y + v\psi_c - H_t\psi_v^n + H\left[\psi_v^c\right]_t \end{bmatrix}.$$

6.2.2 Numerical Scheme

Eq. (6.18) forms a hyperbolic equation with the dispersion serving as a source term. In accordance with the previous chapters, we solve the conserved Nwogu's equation using a staggered grid. The staggering of the conserved variables is shown in Figure 6.2. Each equation is solved on a different control volume shifted from the others.

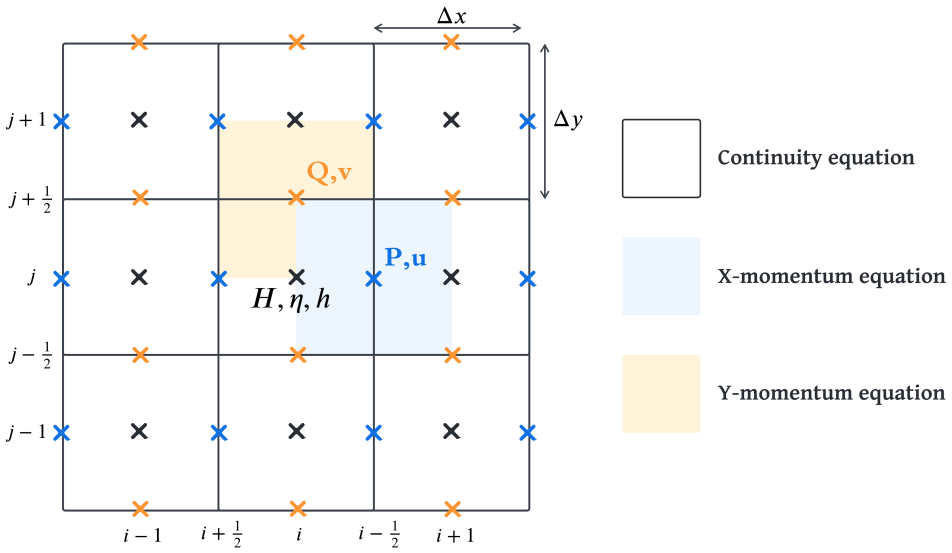


Figure 6.2: Definition sketch: Nwogu's equations state variables

The discretization of the hydrostatic portion of the equations is described in detail in Chapter 2. The SW solver uses second-order time integration along with conservative FD approximations to achieve robust and shock-capturing solutions. The dispersion terms are computed with central difference approximations.

The discretization and the solution of the hyperbolic operator are described in detail in Chapter 2. Here we focus on the numerical implementation of the dispersion terms in the model.

In the continuity equation, the source term can be approximated as follows:

$$
\begin{aligned}
\Big[\psi_c\Big]_{ij} = &\left[\left(\frac{z_\alpha^2}{2}-\frac{h^2}{6}\right)h\left(u_{xx}+v_{xy}\right)+\left(z_\alpha+\frac{h}{2}\right)h\left(\left(hu\right)_{xx}+\left(hv\right)_{xy}\right)\right]_{i+\frac{1}{2},j}\\
&-\left[\left(\frac{z_\alpha^2}{2}-\frac{h^2}{6}\right)h\left(u_{xx}+v_{xy}\right)+\left(z_\alpha+\frac{h}{2}\right)h\left(\left(hu\right)_{xx}+\left(hv\right)_{xy}\right)\right]_{i-\frac{1}{2},j}\\
&+\left[\left(\frac{z_\alpha^2}{2}-\frac{h^2}{6}\right)h\left(u_{xy}+v_{yy}\right)+\left(z_\alpha+\frac{h}{2}\right)h\left(\left(hu\right)_{xy}+\left(hv\right)_{yy}\right)\right]_{i,j+\frac{1}{2}}\\
&-\left[\left(\frac{z_\alpha^2}{2}-\frac{h^2}{6}\right)h\left(u_{xy}+v_{yy}\right)+\left(z_\alpha+\frac{h}{2}\right)h\left(\left(hu\right)_{xy}+\left(hv\right)_{yy}\right)\right]_{i,j-\frac{1}{2}}.
\end{aligned}
\tag{6.19}
$$

The second-order derivatives in Eq. (6.19) are discretized using second-order FD approximations:

$$
\Big[u_{xx}\Big]_{i+\frac{1}{2},j}=\frac{u_{i+\frac{3}{2},j}-2u_{i+\frac{1}{2},j}+u_{i-\frac{1}{2},j}}{\Delta x^2},\qquad \Big[v_{xy}\Big]_{i+\frac{1}{2},j}=\frac{\left(v_{i+1,j+\frac{1}{2}}-v_{i+1,j-\frac{1}{2}}\right)-\left(v_{i,j+\frac{1}{2}}-v_{i,j-\frac{1}{2}}\right)}{\Delta x\times\Delta y},
$$

$$
\Big[v_{yy}\Big]_{i,j+\frac{1}{2}}=\frac{v_{i,j+\frac{3}{2}}-2v_{i,j+\frac{1}{2}}+v_{i,j-\frac{1}{2}}}{\Delta y^2},\qquad \Big[u_{xy}\Big]_{i,j+\frac{1}{2}}=\frac{\left(u_{i+\frac{1}{2},j+1}-u_{i-\frac{1}{2},j+1}\right)-\left(u_{i+\frac{1}{2},j}-u_{i-\frac{1}{2},j}\right)}{\Delta x\times\Delta y}.
$$

To compute the hu and hv derivatives, the water depth h is approximated at the cell interface with average operators:

$$
h_{i+\frac{1}{2},j}=\frac{h_{i,j}+h_{i+1,j}}{2},\qquad h_{i,j+\frac{1}{2}}=\frac{h_{i,j}+h_{i,j+1}}{2}.
$$

Similarly, the dispersion terms in the momentum equations are computed at the cell interface with:

$$
\Big[\psi_u\Big]_{i+\frac{1}{2},j}=\frac{z^2_{\alpha\left(i+\frac{1}{2},j\right)}}{2}\left(\Big[u_{xx}\Big]_{i+\frac{1}{2},j}+\Big[v_{xy}\Big]_{i+\frac{1}{2},j}\right)+z_{\alpha\left(i+\frac{1}{2},j\right)}\left(\Big[\left(hu\right)_{xx}\Big]_{i+\frac{1}{2},j}+\Big[\left(hv\right)_{xy}\Big]_{i+\frac{1}{2},j}\right),
$$

$$
\Big[\psi_v\Big]_{i,j+\frac{1}{2}}=\frac{z^2_{\alpha\left(i,j+\frac{1}{2}\right)}}{2}\left(\Big[u_{xy}\Big]_{i,j+\frac{1}{2}}+\Big[v_{yy}\Big]_{i,j+\frac{1}{2}}\right)+z_{\alpha\left(i,j+\frac{1}{2}\right)}\left(\Big[\left(hu\right)_{xy}\Big]_{i,j+\frac{1}{2}}+\Big[\left(hv\right)_{yy}\Big]_{i,j+\frac{1}{2}}\right).
$$

It should be noted that the source terms for each momentum equation must be estimated at the corresponding cell interface. All source terms in the x-momentum equation are computed at $x_{i+\frac{1}{2},j}$, while the source terms in the y-momentum equation are computed at $x_{i,j+\frac{1}{2}}$.
Since the continuity equation is solved first, the temporal derivative H_t in the momentum equations can be computed explicitly using the previous and updated values of H:

$$\Big[H_t\Big]^{\mathrm{n}+\frac{1}{2}}_{i+\frac{1}{2},j} = \frac{\Big[H_{i+\frac{1}{2},j}\Big]^{\mathrm{n}+1} - \Big[H_{i+\frac{1}{2},j}\Big]^{\mathrm{n}}}{\Delta t}, \qquad H_{i+\frac{1}{2},j} = \frac{H_{i,j} + H_{i+1,j}}{2}. \tag{6.20}$$

$$\Big[H_t\Big]^{\mathrm{n}+\frac{1}{2}}_{i,j+\frac{1}{2}} = \frac{\Big[H_{i,j+\frac{1}{2}}\Big]^{\mathrm{n}+1} - \Big[H_{i,j+\frac{1}{2}}\Big]^{\mathrm{n}}}{\Delta t}, \qquad H_{i,j+\frac{1}{2}} = \frac{H_{i,j} + H_{i,j+1}}{2}. \tag{6.21}$$

Solution of System of Equations

The explicit time-stepping scheme integrates the hyperbolic equations to compute the evolution variables H, P, and Q. Since the source terms depend on the primitive variables η, u, and v, these variables have to be retrieved at each time step. The quantities P and Q are evolved in time and depend on variations of u and v, respectively. The discretization of the evolution variables can be achieved with a second-order accurate central differential scheme:

$$\Big[P\Big]_{i+\frac{1}{2},j} = H_{i+\frac{1}{2},j}\Big[u + \frac{z_\alpha^2}{2}u_{xx} + z_\alpha\,(hu)_{xx}\Big]_{i+\frac{1}{2},j},$$

$$\begin{aligned}\Big[P\Big]_{i+\frac{1}{2},j} = H_{i+\frac{1}{2},j}\Bigg(u_{i+\frac{1}{2},j} &+ \frac{z^2_{\alpha(i+\frac{1}{2},j)}}{2}\left[\frac{u_{i+\frac{3}{2},j} - 2u_{i+\frac{1}{2},j} + u_{i-\frac{1}{2},j}}{\Delta x^2}\right] \\ &+ z_{\alpha(i+\frac{1}{2},j)}\left[\frac{(hu)_{i+\frac{3}{2},j} - 2(hu)_{i+\frac{1}{2},j} + (hu)_{i-\frac{1}{2},j}}{\Delta x^2}\right]\Bigg).\end{aligned}$$

$$\Big[Q\Big]_{i,j+\frac{1}{2}} = H_{i,j+\frac{1}{2}}\Big[v + \frac{z_\alpha^2}{2}v_{yy} + z_\alpha\,(hv)_{yy}\Big]_{i,j+\frac{1}{2}},$$

$$\begin{aligned}\Big[Q\Big]_{i,j+\frac{1}{2}} = H_{i,j+\frac{1}{2}}\Bigg(v_{i,j+\frac{1}{2}} &+ \frac{z^2_{\alpha(i,j+\frac{1}{2})}}{2}\left[\frac{v_{i,j+\frac{3}{2},j} - 2v_{i,j+\frac{1}{2}} + v_{i,j-\frac{1}{2}}}{\Delta y^2}\right] \\ &+ z_{\alpha(i,j+\frac{1}{2})}\left[\frac{(hv)_{i,j+\frac{3}{2}} - 2(hv)_{i,j+\frac{1}{2}} + (hv)_{i,j-\frac{1}{2}}}{\Delta y^2}\right]\Bigg).\end{aligned}$$

The computation of the velocity components u and v requires the solution of implicit equations. Since the cross-derivative terms are approximated as source terms in the momentum equations (*i.e.*, ψ_u^c and ψ_v^c), the x- and y- systems are decoupled and u and v are computed independently with tri-diagonal solvers:

$$\underbrace{\begin{bmatrix} C^x_{1,j} & R^x_{1,j} & \vdots & \vdots & \vdots \\ L^x_{2,j} & C^x_{2,j} & R^x_{2,j} & \vdots & \vdots \\ \vdots & L^x_{3,j} & C^x_{3,j} & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & R^x_{N-1,j} \\ \vdots & \vdots & \vdots & L^x_{N,j} & C^x_{N,j} \end{bmatrix}}_{\mathscr{A}} \underbrace{\begin{bmatrix} u_{1,j} \\ u_{2,j} \\ \vdots \\ u_{N-1,j} \\ u_{N,j} \end{bmatrix}}_{\mathscr{U}} \underbrace{\begin{bmatrix} P_{1,j}/H_{1,j} \\ P_{2,j}/H_{2,j} \\ \vdots \\ P_{N-1,j}/H_{N-1,j} \\ P_{N,j}/H_{N,j} \end{bmatrix}}_{\mathscr{P}}, \tag{6.22}$$

$$\underbrace{\begin{bmatrix} C^y_{i,1} & R^y_{i,1} & \vdots & \vdots & \vdots \\ L^y_{i,2} & C^y_{i,2} & R^y_{i,2} & \vdots & \vdots \\ \vdots & L^y_{i,3} & C^y_{i,3} & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & R^y_{i,M-1} \\ \vdots & \vdots & \vdots & L^y_{i,M} & C^y_{i,M} \end{bmatrix}}_{\mathscr{B}} \underbrace{\begin{bmatrix} v_{i,1} \\ v_{i,2} \\ \vdots \\ v_{i,M-1} \\ v_{i,M} \end{bmatrix}}_{\mathscr{V}} \underbrace{\begin{bmatrix} Q_{i,1}/H_{i,1} \\ Q_{i,2}/H_{i,2} \\ \vdots \\ Q_{i,M-1}/H_{i,M-1} \\ Q_{i,M}/H_{i,M} \end{bmatrix}}_{\mathscr{Q}}, \tag{6.23}$$

where:

$$C^x_{i+\frac{1}{2},j} = z_{\alpha\left(i+\frac{1}{2},j\right)} \left[\frac{1}{z_{\alpha\left(i+\frac{1}{2},j\right)}} - \frac{z_{\alpha\left(i+\frac{1}{2},j\right)}}{\Delta x^2} - \frac{2h_{i+\frac{1}{2},j}}{\Delta x^2} \right], \tag{6.24}$$

$$C^y_{i,j+\frac{1}{2}} = z_{\alpha\left(i,j+\frac{1}{2}\right)} \left[\frac{1}{z_{\alpha\left(i,j+\frac{1}{2}\right)}} - \frac{z_{\alpha\left(i,j+\frac{1}{2}\right)}}{\Delta y^2} - \frac{2h_{i,j+\frac{1}{2}}}{\Delta y^2} \right], \tag{6.25}$$

$$L^x_{i+\frac{1}{2},j} = z_{\alpha\left(i+\frac{1}{2},j\right)} \left[\frac{z_{\alpha\left(i+\frac{1}{2},j\right)}}{2\Delta x^2} + \frac{h_{i-\frac{1}{2},j}}{\Delta x^2} \right], \tag{6.26}$$

$$R^x_{i+\frac{1}{2},j} = z_{\alpha\left(i+\frac{1}{2},j\right)} \left[\frac{z_{\alpha\left(i+\frac{1}{2},j\right)}}{2\Delta x^2} + \frac{h_{i+\frac{3}{2},j}}{\Delta x^2} \right], \tag{6.27}$$

$$L^y_{i,j+\frac{1}{2}} = z_{\alpha\left(i,j+\frac{1}{2}\right)} \left[\frac{z_{\alpha\left(i,j+\frac{1}{2}\right)}}{2\Delta y^2} + \frac{h_{i,j-\frac{1}{2}}}{\Delta y^2} \right], \tag{6.28}$$

$$R^{y}_{i,j+\frac{1}{2}} = z_{\alpha(i,j+\frac{1}{2})}\left[\frac{z_{\alpha(i,j+\frac{1}{2})}}{2\Delta y^2} + \frac{h_{i,j+\frac{3}{2}}}{\Delta y^2}\right]. \tag{6.29}$$

The dispersion processes are modulated by the reference depth parameters $\frac{z^2_\alpha}{\Delta x^2}$ and $\frac{z^2_\alpha}{\Delta y^2}$ as well as the bottom curvature computed from the $\frac{h^2}{\Delta x^2}$ and $\frac{h^2}{\Delta y^2}$ terms.
In the case of a dry cell, the source values in vectors $\mathscr{P}$ and $\mathscr{Q}$ are set to zero. A cell is considered dry if the total water height H in the cell is less or equal than the minimum water depth H_{min}:

$$\mathscr{P}_{i+\frac{1}{2},j} = \begin{cases} P_{i+\frac{1}{2},j}/H_{i+\frac{1}{2},j} & \text{if } H_{i+\frac{1}{2},j} > H_{min} \\ 0 & \text{if } H_{i+\frac{1}{2},j} \leqslant H_{min} \end{cases}, \quad \mathscr{Q}_{i,j+\frac{1}{2}} = \begin{cases} Q_{i,j+\frac{1}{2}}/H_{i,j+\frac{1}{2}} & \text{if } H_{i,j+\frac{1}{2}} > H_{min} \\ 0 & \text{if } H_{i,j+\frac{1}{2}} \leqslant H_{min} \end{cases}. \tag{6.30}$$

The systems of equations are solved both in the predictor and corrector steps. The tri-diagonal matrices $\mathscr{A}$ and $\mathscr{B}$ are diagonally dominant and can be solved with a direct tri-diagonal solver such as the Thomas Algorithm. The algorithm is described in two stages. The first stage is a forward sweep of the matrix to eliminate the lower diagonal L. The upper diagonal, R is then updated to a new value R':

$$R'_k = \begin{cases} \dfrac{R_k}{C_k}, & k = 1, \\ \dfrac{R_k}{C_k - L_k R'_{k-1}}, & k = 2, 3, \ldots, N-1. \end{cases} \tag{6.31}$$

The left-hand side is then recomputed with a forward sweep:

$$\mathscr{P}'_k = \begin{cases} \dfrac{\mathscr{P}_k}{C_k}, & k = 1, \\ \dfrac{\mathscr{P}_k - L_k \mathscr{P}'_{k-1}}{C_k - L_k R'_{k-1}}, & k = 2, 3, \ldots, N-1. \end{cases} \tag{6.32}$$

Finally, using a back substitution, we compute the solution for $\mathscr{U}$:

$$\mathscr{U}'_k = \begin{cases} \mathscr{P}'_k, & k = N, \\ \mathscr{P}'_k - R'_k \mathscr{U}'_{k+1}, & k = N-1, N-2, \ldots, 1. \end{cases} \tag{6.33}$$

Cross-terms

Similar to the approach adopted in Shi et al. [2012] and Roeber and Cheung [2012], and in order to simplify the system of equations that arise from the time-derivative dispersion terms, all cross-derivatives are moved to the right-hand side of the governing equation and are treated as source terms Eq. (6.18). The cross-terms ψ^c_u and ψ^c_v are saved at each time step, and the time-derivation is then computed with a backward difference approximation:

$$\left[(\psi_u^c)_t \right]_{i+\frac{1}{2},j}^{\mathrm{n}} = \frac{\left[\psi_u^c\right]_{i+\frac{1}{2},j}^{\mathrm{n}} - \left[\psi_u^c\right]_{i+\frac{1}{2},j}^{\mathrm{n-1}}}{\Delta t^{\mathrm{n}}}, \qquad \left[(\psi_v^c)_t \right]_{i,j+\frac{1}{2}}^{\mathrm{n}} = \frac{\left[\psi_v^c\right]_{i,j+\frac{1}{2}}^{\mathrm{n}} - \left[\psi_v^c\right]_{i,j+\frac{1}{2}}^{\mathrm{n-1}}}{\Delta t^{\mathrm{n}}}. \tag{6.34}$$

6.2.3 Wave Generation

Wave generation is an essential component of operational nearshore wave models. In a computational domain, waves can be generated from the boundaries by applying a boundary condition that introduces the wave profile into the numerical solution. Using linear wave theory, Nwogu [1993] derived the following relation between the free surface of the incident wave at the boundary η_I and the horizontal velocity $u_{\alpha,I}$:

$$u_{\alpha,I} = \frac{\omega}{kh\left[1 - (\alpha + \frac{1}{3})(kh)^2\right]}\eta_I, \tag{6.35}$$

where ω is the angular frequency and k is the wave number of the incident wave.

In practice, wave generation from the boundaries still poses some serious stability problems, especially for long simulations (Wei and Kirby [1995]). In addition, this approach prevents reflected waves from being absorbed at the input boundary, which can affect the wave generation process. To overcome these problems, Wei et al. [1999] proposed a method to generate waves inside the computational domain by introducing a source term into the continuity equation. This approach is analogous to adding and subtracting mass. To ensure a smooth transition of the wave input, the source term is applied over a range of grid points in the form of a Gaussian curve.

The wavemaker source term consists of a product of two functions $g\,(x)$ and $s\,(x)$, where g(x) is a Gaussian shape function and s(t) is a time-dependent input signal:

$$\psi_{wm}\,(x, y, t) = g\,(x)\,s\,(y, t)\,. \tag{6.36}$$

The Gaussian shape function is defined by:

$$g\,(x) = \exp\left[-\beta(x - x_s)^2\right]. \tag{6.37}$$

x_s is the wavemaker position and β is a parameter that determines the width of the source function. As β increases, the width of the wavemaker becomes narrower. To prevent instabilities that might arise from a too narrow source function, the optimized value of β is expressed as:

$$\beta \approx \frac{80}{\delta^2 L_p^2}, \tag{6.38}$$

where δ is set to 0.5 and L_p is the peak wavelength.

Multiple waves of different amplitudes, frequencies, and directions are simultaneously produced through the superposition of the individual signals:

$$s\left(y,t\right)=\sum_{i=1}^{M_\omega}\sum_{j=1}^{M_\theta}D_{ij}\sin\left[y\left(k_i\sin\theta_j\right)-\omega_i\,t+\phi_{ij}\right], \tag{6.39}$$

where ϕ_{ij} is a random phase and M_ω and M_θ denote the numbers of frequencies and directions the wave spectrum is composed of. D_{ij} is the magnitude of the source function. Wei et al. [1999] derived the following expression for D_{ij}:

$$D_{i,j}=\frac{2a_{ij}\,\cos(\theta_j)(\omega_i^2-(\alpha+\frac{1}{3})gk_i^4h^3)}{\omega_i k_i B_{ij}[1-\alpha(k_ih)^2]}, \tag{6.40}$$

where a_{ij} is the desired wave amplitude and B_{ij} is an integral given by:

$$B_{ij}=\int_{-\infty}^{\infty}\mathrm{e}^{(-\beta x^2)}\mathrm{e}^{\,(-il_{ij}x)}\,dx=\sqrt{\frac{\pi}{\beta}}\exp\left(\frac{-l_{ij}^2}{4\beta}\right), \tag{6.41}$$

$$l_{ij}=k_i\cos(\theta_j). \tag{6.42}$$

The generated waves propagate away from the center of the source in both directions, symmetrically. The source is usually placed close to the offshore boundary, with a sponge layer next to it. The sponge layer absorbs the offshore propagating components. It's important to note that the source function is not a direct alteration of the free surface, and reflected waves can travel across the wavemaker. This helps to mimic open sea conditions where reflections from the shore can freely propagate offshore until they are absorbed by the sponge layer.

Frequency binning

An irregular sea state can be accounted for in Boussinesq models through the generation of a full wave spectrum divided into multiple individual waves with different frequencies and directions. The number of waves in the frequency distribution determines the behavior and properties of the resulting time series. It should be kept in mind that it is ultimately the frequency binning that dictates the time series and, most importantly, the recycling of it. With discrete frequency binning, the width of the smallest bin Δf is related to the recycling of the resulting time series. This can be best visualized with a bichromatic wave. The wave with the lower frequency will overtake the one with the higher frequency after $1/(f_{max}-f_{min})$. In the case of a wave spectrum, the recycling process occurs in a similar fashion. For a contestant binning Δf the recycling occurs after:

$$T_{recycling}=\frac{1}{\Delta f}. \tag{6.43}$$

The following test illustrates the consequences that inadequate frequency binning can have on the wave field generated by the wavemaker. In a 1D long flat channel, we generate and propagate a Pierson-Moskowitz spectrum with a peak period $Tp=15\,\mathrm{s}$ and a significant wave height $H_s=1\,\mathrm{m}$ (see Appendix E). By taking a constant frequency binning $\Delta f=0.005\,\mathrm{Hz}$, we decompose the spectrum into 19 distinct monochromatic waves. Figure 6.3 shows the resulting time series after 1 hour of computation.

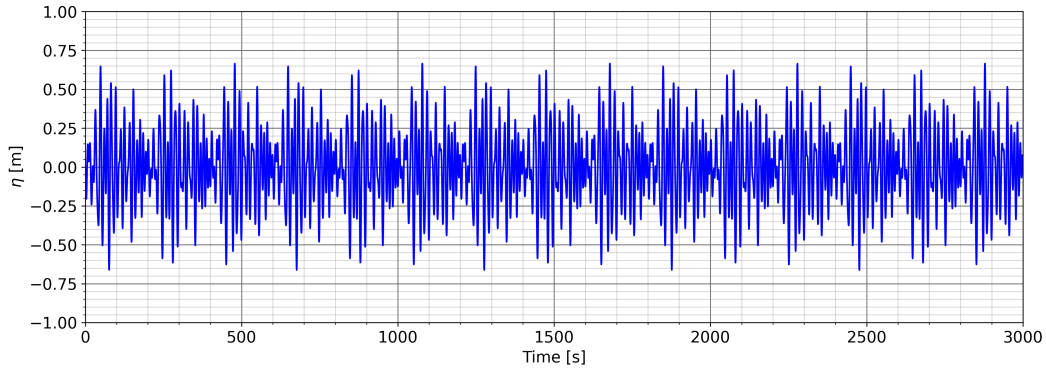


Figure 6.3: The creation of an artificial wave group as a result of coarse frequency binning.

Figure 6.3 illustrates the creation of an artificial wave grouping introduced to the solution through the recycling of the time series from the input spectrum. The waveform is repeated every $T = 200\,\mathrm{s} = 1/\Delta f$, which is a typical period of infra-gravity waves, though, it is of a purely artificial nature in this example.

For accurate wave computation, all waves composing the spectrum should ideally interact with each other once over the course of the run. To compute all wave interactions while avoiding a time series recycling, the frequency binning is set to:

$$\Delta f = \frac{1}{T_{\text{computation time}}}. \tag{6.44}$$

Frequency limits

- **Low-frequency limit:** Since the longest wave in a swell spectrum is no longer than 30 s, it is generally representative to take $f_{min} = 1/30$. We can also define f_{min} based on the peak frequency, such as $f_{min} = 0.5 f_p$

- **High-frequency limit:** This limit depends on the dispersive properties of the equations. Since the dispersion error is more significant for high frequencies (see Chapter 3), it makes sense to remove spectrum frequencies that are not compatible with the equations. In the case of Nwogu, the governing equations can handle dispersive waves up to $kh = \pi$.

After removing the high and low frequencies, it is necessary to correct the spectrum in order to preserve the correct H_s. This can be done by multiplying the energy spectrum density by the following coefficient:

$$S^{cor}(f) = C_{cor} \times S(f), \qquad \text{where} \qquad C_{cor} = \frac{Hs^2}{8 \int_{f_{min}}^{f_{max}} S(f)\, df}. \tag{6.45}$$

6.2.4 Absorbing Boundaries

By generating waves inside the computational domain with the internal wave-maker, we are often required to absorb waves when they reach the offshore boundaries. This can be done by implementing the sponge layer technique to absorb short dispersive waves coupled with a Sommerfeld radiation condition for long hydrostatic waves.

Sponge layer

The method for absorbing shortwaves is based on the sponge layer technique introduced by Larsen and Dancy [1983]. In this approach, the state variables $\phi\,(i.e, \eta, u, v)$ are directly attenuated at each time step to reach a reference solution :

$$\phi_k = \phi_{ref} + \left(\phi_k - \phi_{ref}\right)/C_s, \tag{6.46}$$

where C_s is the damping coefficient function defined by:

$$C_s = \alpha_s^{\gamma_s^{k-1}}, \qquad k = 1, 2, 3, ..., n. \tag{6.47}$$

α_s and γ_s are two free parameters. For complete absorption of the short waves, Chen et al. [1999] suggested taking $\alpha_s = 2$ and $\gamma_s = 0.88 - 0.92$. k represents the grid cell index, and n is the total number of grid cells inside the sponge layer. Chen et al. [1999] recommended using $n = 50 - 100$. Typically, the width of the sponge layer is one to two times the peak wavelength.

In addition to the L-D type sponge layer, a friction type sponge layer is applied for an efficient and noise-free absorption as suggested by Shi et al. [2011]. The source term for the friction-type sponge can be described as :

$$F_{spl} = -C_{sponge}\sqrt{u^2 + v^2}\begin{pmatrix} u \\ v \end{pmatrix} h. \tag{6.48}$$

In Eq. (6.48), the water depth h is introduced to make the source term depth-independent in the flux-type momentum equations. At the boundary, the coefficient C_{sponge} is smoothly ramped in space. C_{sponge} can be expressed as follows for a sponge layer on the left end of the domain:

$$C_{sponge} = C_{max}\left(1 - \tanh\frac{10\,(k-1)}{I_{width} - 1}\right). \tag{6.49}$$

k and I_{width} represent the grid number and the sponge layer width in points, respectively. In this case, the friction-type sponge is applied to a width of two to three times the peak wavelength. C_{max} is the maximum value of C_{sponge} which is taken equal to 1.

Radiation condition

The Sommerfeld radiation condition is then utilized to allow long waves or currents to leave the computational domain. The condition can be expressed by:

$$\eta_t + c\eta_x = 0, \tag{6.50}$$

where C is the outgoing wave celerity. For long waves $C = \sqrt{gH}$. The combination of Eq. (6.50) with the mass equation on a flat bottom yields the following boundary condition:

$$\eta = u\sqrt{\frac{H}{g}}. \tag{6.51}$$

6.2.5 Wave Breaking

Wave-breaking effects, which naturally occur in the surf zone, can be computed with a TKE-based eddy viscosity closure. This approach has proven to be very efficient and has good convergence properties compared to the hybrid approach, where the dispersion terms are locally and momentarily deactivated to allow for a cell-wise hyperbolic solution (Chapter 5). In this section, we detail the implementation of the eddy viscosity approach on a 2D staggered grid. A time- and space-dependent source term is introduced to the momentum equations:

$$R_b = \begin{bmatrix} 0 \\ 2\Big[\nu_t H u_x\Big]_x + \Big[\nu_t H (u_y + v_x)\Big]_y \\ 2\Big[\nu_t H v_y\Big]_y + \Big[\nu_t H (u_y + v_x)\Big]_x \end{bmatrix} = \begin{bmatrix} 0 \\ R_x \\ R_y \end{bmatrix}, \tag{6.52}$$

where ν_t represents the eddy viscosity that depends on the turbulent kinetic energy k (see Chapter 5). The eddy viscosity term is computed at the cell center $\nu_{t(i,j)}$ and the momentum source terms are approximated at the cell-interface :

$$\begin{aligned}
\Big[R_x\Big]_{i+\frac{1}{2},j} = {} & \frac{2}{\Delta x^2}\left[\nu_{t(i+1,j)} H_{i+1,j}\left(u_{i+\frac{3}{2},j} - u_{i+\frac{1}{2},j}\right) - \nu_{t(i,j)} H_{i,j}\left(u_{i+\frac{1}{2},j} - u_{i-\frac{1}{2},j}\right)\right] \\
& + \frac{1}{\Delta y^2}\left[\nu_{t(i+\frac{1}{2},j+\frac{1}{2})} H_{i+\frac{1}{2},j+\frac{1}{2}}\left(u_{i+\frac{1}{2},j+1} - u_{i+\frac{1}{2},j}\right) - \nu_{t(i+\frac{1}{2},j-\frac{1}{2})} H_{i+\frac{1}{2},j-\frac{1}{2}}\left(u_{i+\frac{1}{2},j} - u_{i+\frac{1}{2},j-1}\right)\right] \\
& + \frac{1}{\Delta x \Delta y}\left[\nu_{t(i+\frac{1}{2},j+\frac{1}{2})} H_{i+\frac{1}{2},j+\frac{1}{2}}\left(v_{i+1,j+\frac{1}{2}} - v_{i,j+\frac{1}{2}}\right) - \nu_{t(i+\frac{1}{2},j-\frac{1}{2})} H_{i+\frac{1}{2},j-\frac{1}{2}}\left(v_{i+1,j-\frac{1}{2}} - v_{i,j-\frac{1}{2}}\right)\right],
\end{aligned}$$

$$\begin{aligned}
\Big[R_y\Big]_{i,j+\frac{1}{2}} = {} & \frac{2}{\Delta y^2}\left[\nu_{t(i,j+1)} H_{i,j+1}\left(v_{i,j+\frac{3}{2}} - v_{i,j+\frac{1}{2}}\right) - \nu_{t(i,j)} H_{i,j}\left(v_{i,j+\frac{1}{2}} - v_{i,j-\frac{1}{2}}\right)\right] \\
& + \frac{1}{\Delta x^2}\left[\nu_{t(i+\frac{1}{2},j+\frac{1}{2})} H_{i+\frac{1}{2},j+\frac{1}{2}}\left(v_{i+1,j+\frac{1}{2}} - v_{i,j+\frac{1}{2}}\right) - \nu_{t(i-\frac{1}{2},j+\frac{1}{2})} H_{i-\frac{1}{2},j+\frac{1}{2}}\left(v_{i,j+\frac{1}{2}} - v_{i-1,j+\frac{1}{2}}\right)\right] \\
& + \frac{1}{\Delta x \Delta y}\left[\nu_{t(i+\frac{1}{2},j+\frac{1}{2})} H_{i+\frac{1}{2},j+\frac{1}{2}}\left(u_{i+\frac{1}{2},j+1} - u_{i+\frac{1}{2},j}\right) - \nu_{t(i-\frac{1}{2},j+\frac{1}{2})} H_{i-\frac{1}{2},j+\frac{1}{2}}\left(u_{i-\frac{1}{2},j+1} - u_{i-\frac{1}{2},j}\right)\right].
\end{aligned}$$

Here, $\nu_{t(i+\frac{1}{2},j+\frac{1}{2})}$ and $H_{i+\frac{1}{2},j+\frac{1}{2}}$ are obtained from the surrounding points by simple linear interpolation. It should be noted that the use of a staggered grid results in a more compact FD stencil and improved accuracy, as shown in Fornberg [1998]. The leading error ratio for a second-order central difference is four times lower in a staggered grid than in a collocated grid.

6.3 Implementation Strategies for Computational Efficiency

The model is built around a lean numerical scheme thanks to the efficient hyperbolic solver, which does not require the numerically expensive Riemann approximations. Yuan et al. [2020] showed that for their FUNWAVE model, the computation of the flux with the Riemann solver takes around 42.9% of the total model computation time on the CPU. Therefore, a simplified flux computation will substantially improve the efficiency of a Boussinesq model. Moreover, the present model does not require high-order reconstruction to deal with the inherent numerical dissipation related to TVD Riemann solutions; and a simple second-order flux reconstruction combined with two-step time integration is sufficient for our applications.

The model is written in C/C++ and the CPU-based parallelization was achieved through the use of OpenMP directives. The model's overall flowchart can be described by the following pseudo-code:

Algorithm 1:

1. **Set initial conditions**. Load wave and topography input from the steering file.

2. **Allocate memory**. Define and initialize all class objects.

3. **Pre-processing**. Compute wavemaker and dispersion source variables.

4. **Enter time loop:**

for *time step* $= 0, 1, 2, 3, ...$ **do**
- Compute Δt
- Compute wave input (either at boundary or internally –> spectral wavemaker)

for *Order* $= 1 : 2$ **do**
- Compute spatial derivatives for dispersion terms
- Compute SWE fluxes for the continuity equation
- Solve the continuity equation
- Compute SWE fluxes for the momentum equations
- Map out cells where the wave-breaking criterion holds
- Solve the turbulent kinetic equation
- Compute the eddy viscosity source term
- Compute the friction source term
- Solve the momentum equations
- Solve the tri-diagonal systems to extract u and v from P and Q
- Apply boundary conditions

end

- Save output files
- Update variables
- Proceed to next time step

end

5. **Free memory and exit program**.

In terms of computation time, the most expensive portion of the code involves the operations within the time loop. Therefore, for efficiency reasons, we avoid all heap memory allocation/deallocation inside this loop. All necessary memory allocations are performed at the start of the run, outside the time loop, and then the same memory is reused for the remaining computation. We also avoid copying large arrays; the update of the flow variables at the end of each time step is achieved by swapping their pointers instead of copying the content of the new variables into the old ones.

6.3.1 Floating-point Division

The numerical discretization of the equations involves multiple divisions by the grid size Δx and Δy as shown in Chapter 6.2.2. While most modern CPUs can perform floating-point multiplication in 1–3 clock cycles, the division requires inherent iterative subtractions that cannot be performed concurrently and thus take longer (15–25 clock cycles) (Fog et al. [2011]). This added latency is not so critical when it comes to small computations. However, in our case, and given how often these simple operations are repeated, this can be substantial. A straightforward workaround for this issue is to precompute $1/\Delta x$ and $1/\Delta y$ and then multiply all the discretization terms by these values. This eliminates many redundant divisions inside the model. We have performed some simple numerical tests where we compared the model computation time before and after reducing the number of division operations. The tests show that the particular code runs around 13% faster by using this very simple trick.

6.3.2 Tri-diagonal Solver

Assuming a constant still water depth or bathymetry ($h_t = 0$), the coefficients of the tri-diagonal matrices $\mathscr{A}$ and $\mathscr{B}$ do not vary over time and therefore can be precomputed in advance outside the time loop. Moreover, the initial forward sweep (Eq. (6.31)) can also be precomputed reducing the TDMA computations by one-third.

6.3.3 Wavemaker

The internal wave generation requires the computation of the following source term for each grid point along the trajectory of the wavemaker in the y-direction:

$$s\left(y,t\right) = \sum_{i=1}^{M_\omega} \sum_{j=1}^{M_\theta} D_{ij} \sin\left[y\left(k_i \sin\theta_j\right) - \omega_i\, t + \phi_{ij}\right]. \tag{6.53}$$

M_ω and M_θ are the numbers of frequencies and directions in the input spectrum. If we consider M_y the number of grid points in the y-direction, the computation of this source term has a $O\left(M_\omega \times M_\theta \times M_y\right)$ time complexity. For a spectrum with fine frequency and directional binning (*i.e.*, thousands of waves), the computation of this single term can be a significant bottleneck, especially since it is time-dependent and thus has to be computed inside the time loop at each time step.

Using the angle summation trigonometric identity, we can rewrite the formula in Eq. (6.53) as:

$$s\left(y,t\right) = \sum_{i=1}^{M_\omega} \Big[\mathcal{A}_i(y)\cos(\omega_i t) - \mathcal{B}_i(y)\sin(\omega_i t)\Big], \tag{6.54}$$

where:

$$\mathcal{A}_i(y) = \sum_{i=1}^{M_\theta} D_{ij} \sin\left[y\left(k_i \sin\theta_j\right) + \phi_{ij}\right], \qquad \mathcal{B}_i(y) = \sum_{i=1}^{M_\theta} D_{ij} \sin\left[y\left(k_i \cos\theta_j\right) + \phi_{ij}\right]. \tag{6.55}$$

$\mathcal{A}_i(y)$ and $\mathcal{B}_i(y)$ can be precomputed in the pre-processing step, and thus the time complexity for the wavemaker source term goes from $\mathcal{O}\left(M_\omega \times M_\theta \times M_y\right)$ to $\mathcal{O}\left(M_\omega \times M_y\right)$.

6.4 Numerical Tests

This work introduces a new Boussinseq wave model with the potential of being used in an operational system for coastal hazard assessment and prediction. The model is expected to be fast while also being reliable in modeling critical nearshore wave processes such as wave refraction-diffraction, wave breaking, wave run-up, and secondary processes such as infra-gravity waves and re-circulation. A series of benchmark tests is used to assess the accuracy of the model in both 1D and 2D settings. It is worth noting that the hydrostatic portion of the model has already been validated in Chapter 2. In this section, we focus more on the dispersive processes and wave transformations in both intermediate and shallow water.

6.4.1 Gaussian Drop Test

The Gaussian drop is a very simple test to verify the implementation of the numerical model. Since we are using a regular Cartesian grid, the numerical solution is expected to be symmetric. This means that for the solution of the 2D governing equations, all operations performed in the x-direction must be identical to their y-directional counterparts. To verify this property, we compute the free-surface evolution of a Gaussian drop over a $10\,\mathrm{m} \times 10\,\mathrm{m}$ domain with $5\,\mathrm{cm} \times 5\,\mathrm{cm}$ cell size and reflective boundaries. In 1 m water depth, the initial free surface is described by:

$$\eta_0(x, y) = a_0 \exp\left[-\left(x^2 + y^2\right)\right], \qquad a_0 = 0.3. \tag{6.56}$$

The Gaussian drop, initially at rest, collapses as gravity restores the water form. As a result, oscillations spread equally in all directions from the center of the basin (see Figure 6.4). The oscillations are due to the dispersive effects of the equations. In a hydrostatic model, a bore-type wave is generated and propagated radially instead (Delis and Katsaounis [2005]).

To assess the symmetry of the model, we compare the free-surface evolution along the two diagonal transects of the domain. We choose the diagonals so we can verify both the normal and cross terms implementation. The results are shown in Figure 6.5. The transects are identical to machine precision, which proves the symmetry of the implementation. This verification is quite valuable for spotting coding errors, which can go unnoticed even after several benchmark tests.

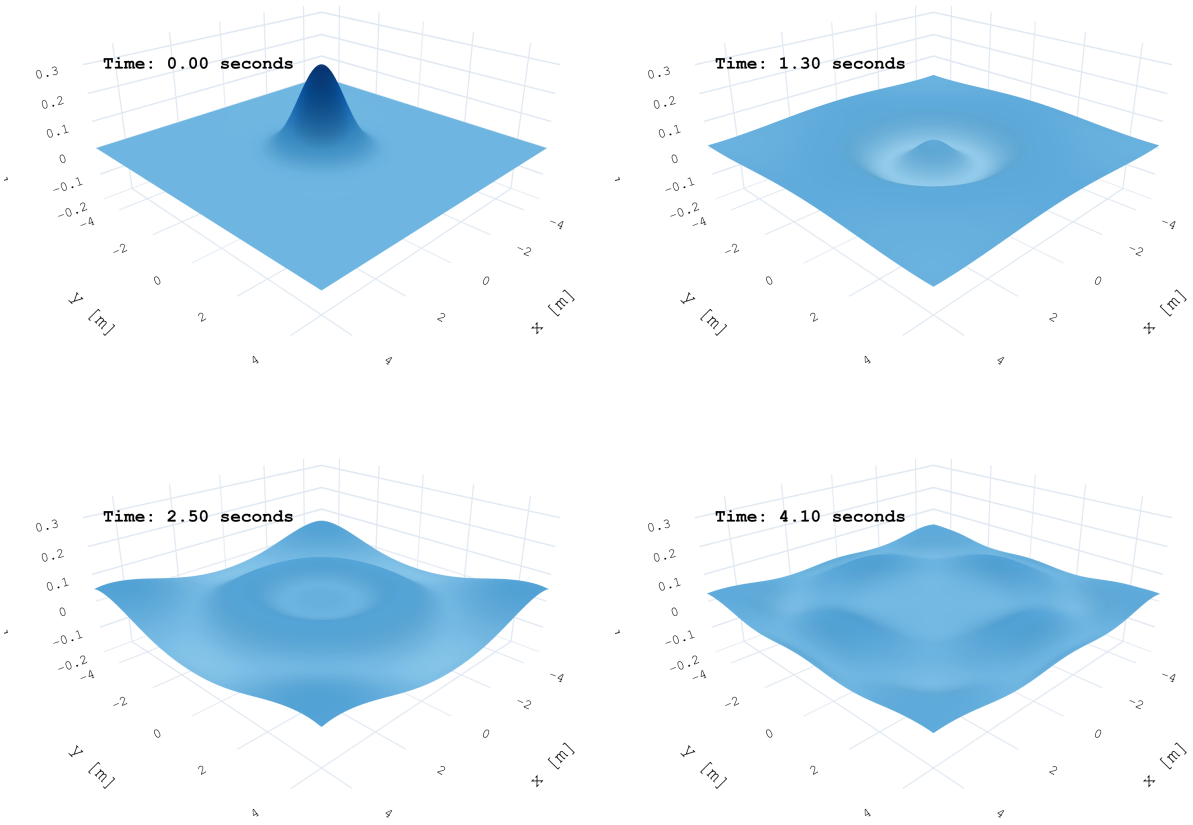


Figure 6.4: Gaussian drop test: Free-surface evolution at different time intervals.

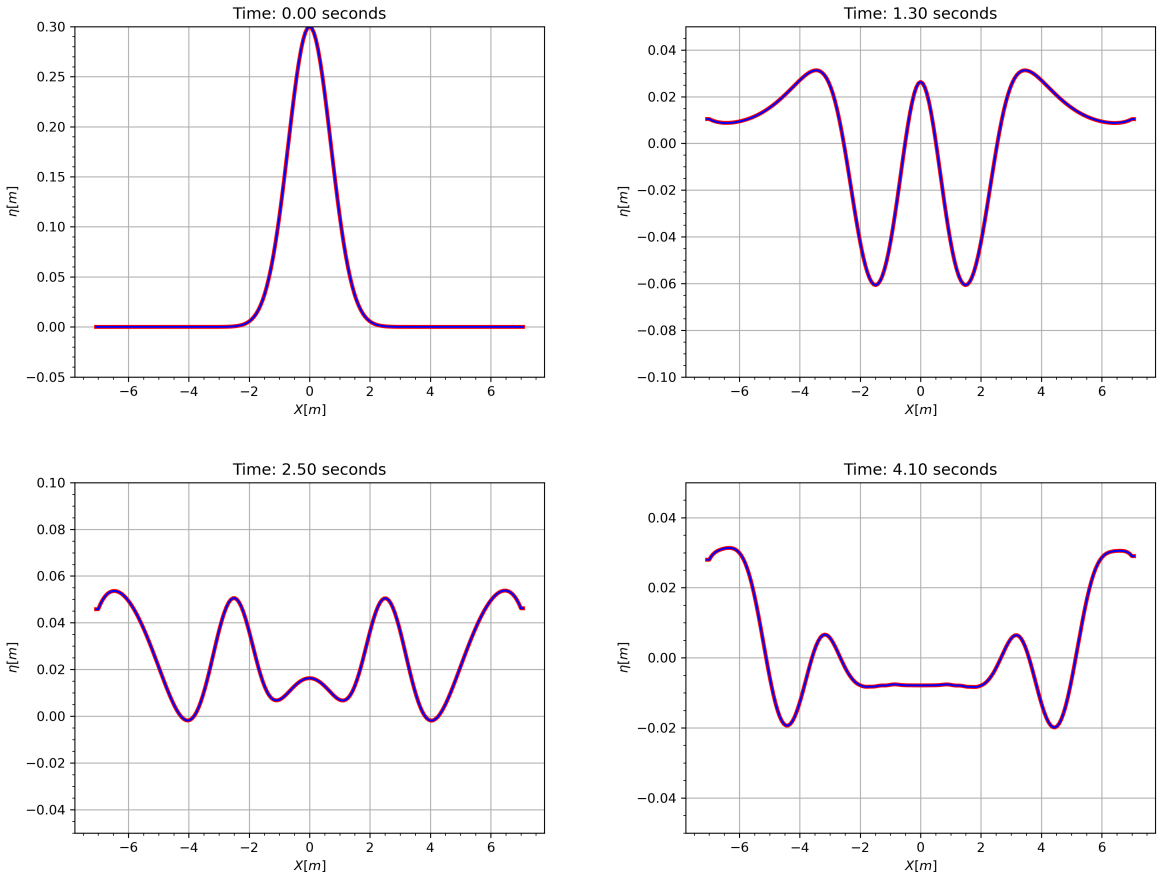


Figure 6.5: Gaussian drop test: Comparison of free-surface evolution along two diagonal transects: Transect $y = x$ (red line) and Transect $y = -x$ (blue line) to verify the symmetry of the model implementation.

6.4.2 Evolution of an Undular Bore

Undular bores, also termed dispersive shock waves, are a generic type of wave generated by an initial discontinuity in the free surface. This phenomenon has been explored by many researchers and serves as a standard illustration of the competing effects of dispersion and nonlinearity. In order to validate the non-hydrostatic properties of our Boussinesq model, we compare the numerical solution to the experimental measurements given in Barranco and Liu [2021]. The experiment is conducted in a 36 m wave flume with a dam-break gate, creating a reservoir with a water height of $h_1 = 0.157$ m. On the other side of the gate, the still-water level is fixed at $h_0 = 0.1$ m. When the gate is lifted up, an undulating bore front travels over a 11.1 m long flat bed and then over a 1/10 slope. In this test, we focus on the constant-depth propagation, and we compare the numerical results to the time series data obtained from the two wave gauges, CG3 and CG4, which are located 8.85 m from the gate and at the beach toe, respectively. The initial setup for the numerical model is similar to the laboratory configuration, where we consider an initial discontinuity in the free surface.

The time series at the two gauges is shown in Figure 6.6. The time series is lined up with the leading wave crest measured at CG3. The same time shift is used to plot the time series at

the gauge CG4. The bore velocity between CG3 and CG4 is almost a perfect match with the traveling bore front in the laboratory, proving the model's accuracy at capturing the correct speed of the undular bore. In addition, the computed bore height and period compare well with the laboratory data, especially for the leading undulations. The grid size has no effect on the frequency of the bore. However, it has some effect on the wave height, which increases for finer grids.

Since the resulting undular bore is quite nonlinear ($a/h = 0.5$), this test showcases the applicability of the weakly nonlinear model to cases where nonlinearity is important.

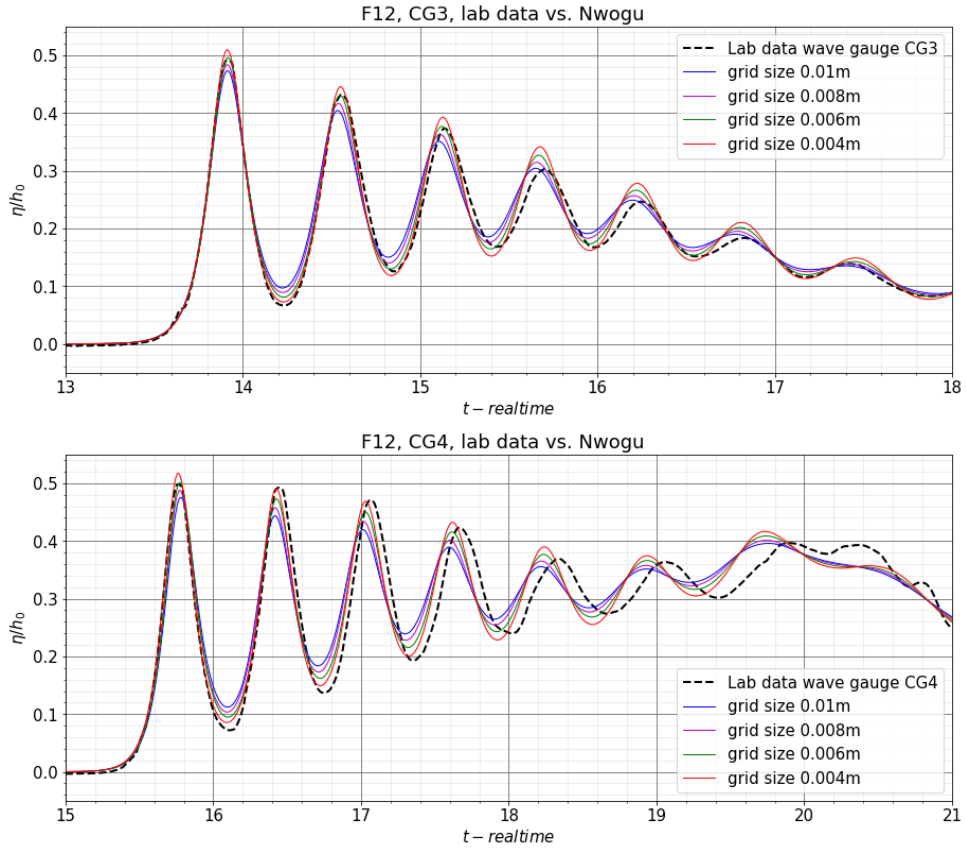


Figure 6.6: Evolution of an Undular Bore: Free surface elevation at gauges. Colored lines denote the numerical results with different grid sizes. The dashed line represents the laboratory data.

6.4.3 Solitary Wave Runup on a Plane Beach

Solitary wave runup on a plane slope is one of the most widely investigated problems in long-wave modeling. In particular, Hall et al. [1953] and Synolakis [1987] have provided important data for the validation of wave-breaking and runup models. Figure 6.7 shows a schematic of the experiment with A denoting the initial solitary wave height, R the runup, and β the beach slope. We first focus on case C from Synolakis [1987] with $A/h = 0.3$ and $\beta = 19.85$, which involves wave breaking on the beach slope. With this standard test, we assess the model's ability to describe shoreline motions and the wave shape before and after breaking. In a 25 m long computational domain with a $\Delta x = 0.025$ m grid size, we compute the benchmark test.

A Courant number of $Cr = 0.5$ is used to ensure the stability of the time integration, and a Manning coefficient of $n = 0.01\,\mathrm{s/m^{1/3}}$ defines the surface roughness of the smooth glass beach in the laboratory experiment. The initial solitary wave is located at $x/h = -20$ from the beach toe allowing the initial wave profile to adjust to the governing equations before reaching the slope. The wave-breaking process is approximated using the eddy viscosity approach, and the free-surface Froude number is chosen as a criterion to detect the onset of breaking (see Chapter 5).

Figure 6.8 compares the measured free-surface profiles to the computed results. In contrast to the Shallow Water solution, which lacks dispersion terms, the Present model reproduces the shoaling process up to $t\sqrt{g/h} = 20$ at the onset of a plunging breaker, as observed in the laboratory experiment. The breaking wave profile is well represented with the eddy viscosity approach, where the added diffusion mimics quite well the dissipative effects of breaking. The dispersive model is stable for the run-up and run-down, with the dispersion terms retained throughout the whole domain. The resulting surge reaches the maximum elevation of $R/h = 0.55$ around $t\sqrt{g/h} = 40$. The model deviates somewhat from the laboratory data around $t\sqrt{g/h} = 55$, when a hydraulic jump develops from the drawdown. Since the computed results agree with the measured data toward the end, the local discrepancies might be attributed to a mix of instrumentation and model errors. The drawdown process introduces air entrainment in the water column and splashes at the surface, which is challenging to measure by any instrument.

Figure 6.9 plots the measured and computed run-up as a function of the initial solitary wave height A/h for beach slopes of 1:19.85, 1:15, and 1:5.67. The data shows good agreement over a wide range of breaking and non-breaking events characterized by a bi-linear distribution with a distinct transition. For the 1:19.85 and 1:15 slopes, the data to the right of the transition represents plunging breakers, whereas the relatively steep slope of 1:5.67 produces surging waves without flow discontinuities or breaking. The model is able to simulate the runup for incident wave heights of up to $A/h = 0.7$, which is beyond the model's range of nonlinear properties.

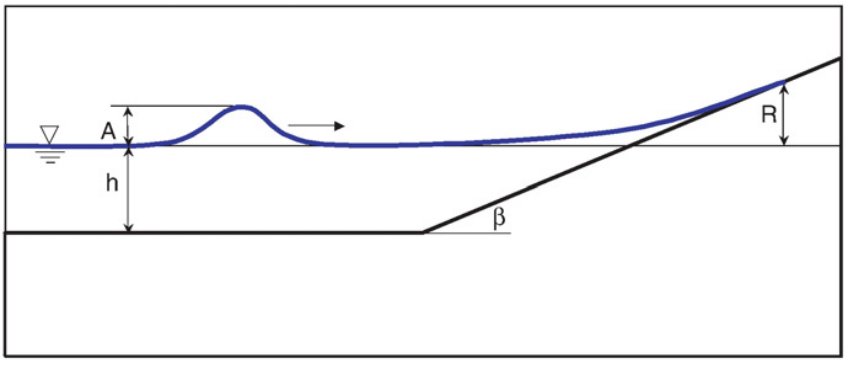


Figure 6.7: Definition sketch of solitary wave runup on a plane beach.

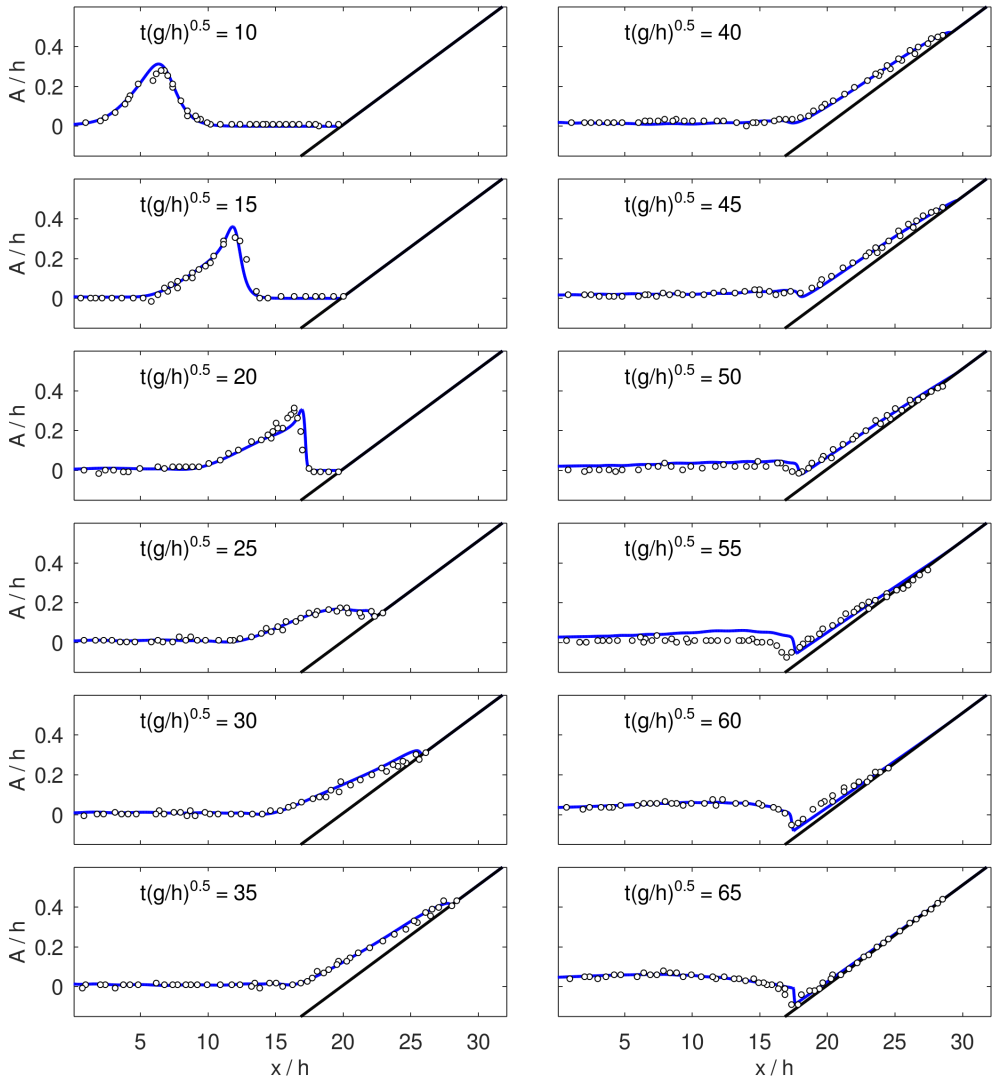


Figure 6.8: Surface profiles of solitary wave transformation on a 1:19.85 plane beach with $A/h = 0.3$. Solid lines and circles denote computed and measured data.

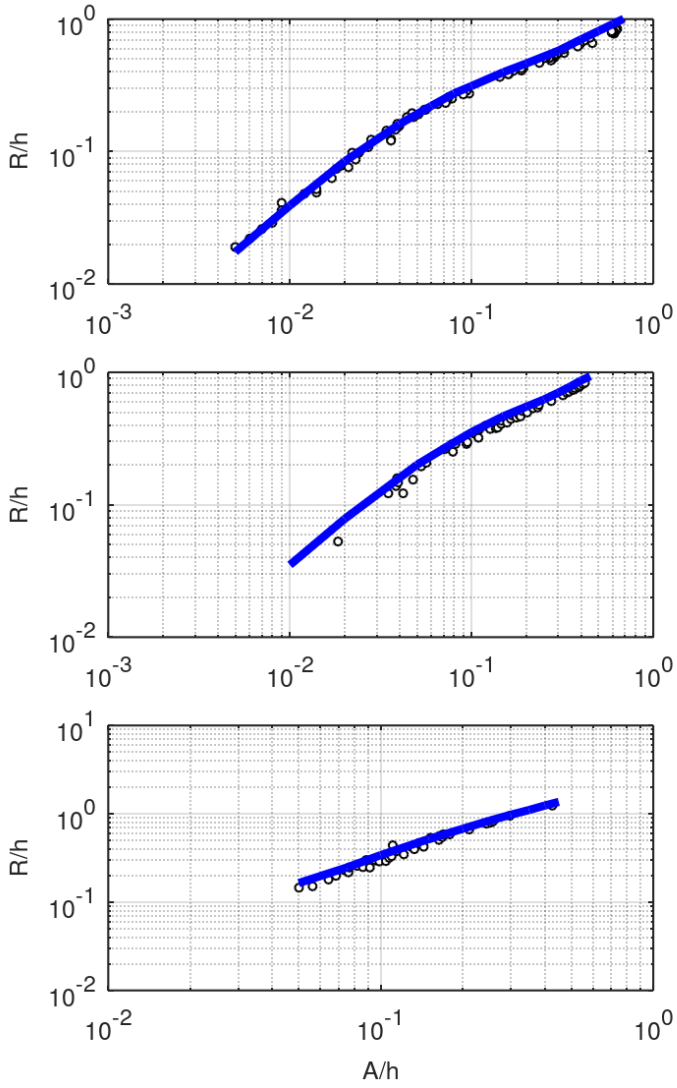


Figure 6.9: Solitary wave runup on a plane beach. (a) 1:19.85 (Synolakis [1987]). (b) 1:15 (Li and Raichlen [2002]). (c) 1:5.67 (Hall et al. [1953]). Solid lines and circles denote computed and measured data.

6.4.4 HIreef - Solitary Wave over 2D Reef

The HIreef test case describes the solitary wave transformation over an idealized fringing reef. Despite the fact that the laboratory experiment focuses on shock-related hydraulic processes such as wave breaking and bore formation, the collected data are very useful in examining other important processes such as shoaling, wave reflection, and swashing dynamics. The test was conducted by Roeber et al. [2010] at the Hinsdale Wave Research Laboratory at Oregon State University. The flume used in the experiment has an effective length of 83.7 m, with a 1:12 slope starting at 25.9 m from the wavemaker. With a water level of 2.5 m, the lagoon has a shallow depth of 0.136 m, while the reef crest remains exposed by 0.065 m. More details about the experimental procedure can be found in Roeber et al. [2010] and Roeber [2010].

We compute the benchmark test in a 1D computational domain with a fixed grid size $\Delta x = 0.05$ m

and a Courant number $Cr = 0.5$. A Manning coefficient of $n = 0.013\,\mathrm{s/m^{1/3}}$ describes the smooth, finished concrete and plywood surfaces of the flume. A 0.75 m solitary wave is generated from the left boundary, which results in a dimensionless wave height of $A/h = 0.3$.

After the generation, the solitary wave shoals over the relatively gentle slope. The free surface profile becomes nearly vertical, and the wave begins to break. Observations during the laboratory experiment indicate subsequent overturning of the free surface and the development of a plunging breaker impinging on the reef crest with a large air cavity and subsequent splash-up. The model is able to correctly account for the plunging breaker effects with the eddy viscosity approach. The values of the computed TKE increase drastically at the breaking wavefront, resulting in a local diffusion that mimics the physical turbulent dissipation. The broken wave begins to travel down the back slope of the reef crest, generating a supercritical flow and displacing the initially still water in the lagoon. The flow generates a hydraulic jump on the back reef and a propagating bore downstream. The TKE is generated and advected to follow the hydraulic jumps. This leads to a propagating discontinuity instead of a dispersive undulation, which is the correct non-breaking Boussinesq solution to this flow problem. The end wall of the flume reflects the bore back to the lagoon. The wave then overtops the reef crest as sheet flow, generating a hydraulic jump on the fore reef. As the wave propagates back from shallow to deep water, it produces a series of dispersive waves as shown in Figure 6.10.

Figure 6.11 compares the time series from the computed and recorded surface elevation. The wave gauge right next to the wall at $x = 80\,\mathrm{m}$ shows the superposition of the approaching and reflected bores propagating in opposite directions. The time series at $x = 65.2\,\mathrm{m}$ shows the approaching bore and the reflected bore from the end wall, as well as its reflection from the reef crest. The steep wave fronts recorded in the gauges reveal the propagation of the breaking wave in the flux-dominated flow. As the hydraulic jump rushes down the fore reef into deeper water, the flow transitions into a dispersion regime where the initial bore transforms into a train of short waves. The resulting undulations intensify as higher harmonics are released from the wave packet. At the same time, a long period reflected wave propagates in the onshore direction and superposes with the released higher harmonics. Wave gauges located near the toe of the slope record highly dispersive waves of $kh > 15$.

For this test, the numerical model is able to accurately reproduce important wave processes:

- Wave breaking and bore formation.
- Transitions between dispersion and flux-dominated flows
- Transitions between sub and supercritical flows.
- Mass and momentum conservation for high Froude numbers.
- Mass and momentum conservation at moving waterlines.
- Wave shoaling and reflection.

The eddy viscosity approach implemented in the model allows for an accurate and stable description of the surging and plunging waves over the reef with a relatively fine grid $\Delta x = 5\,\mathrm{cm}$. The additional diffusion term, which is dependent on the TKE quantity, effectively eliminates

potential instabilities and excessive increases in the wave height, which arise from the depth-inegrated dispersive solution. The dissipation effects have only a short-term and local impact on the numerical results. This means that the dispersion properties of the equations are not altered, and subsequent wave transformation processes, such as the release of high-order harmonics, are well accounted for.

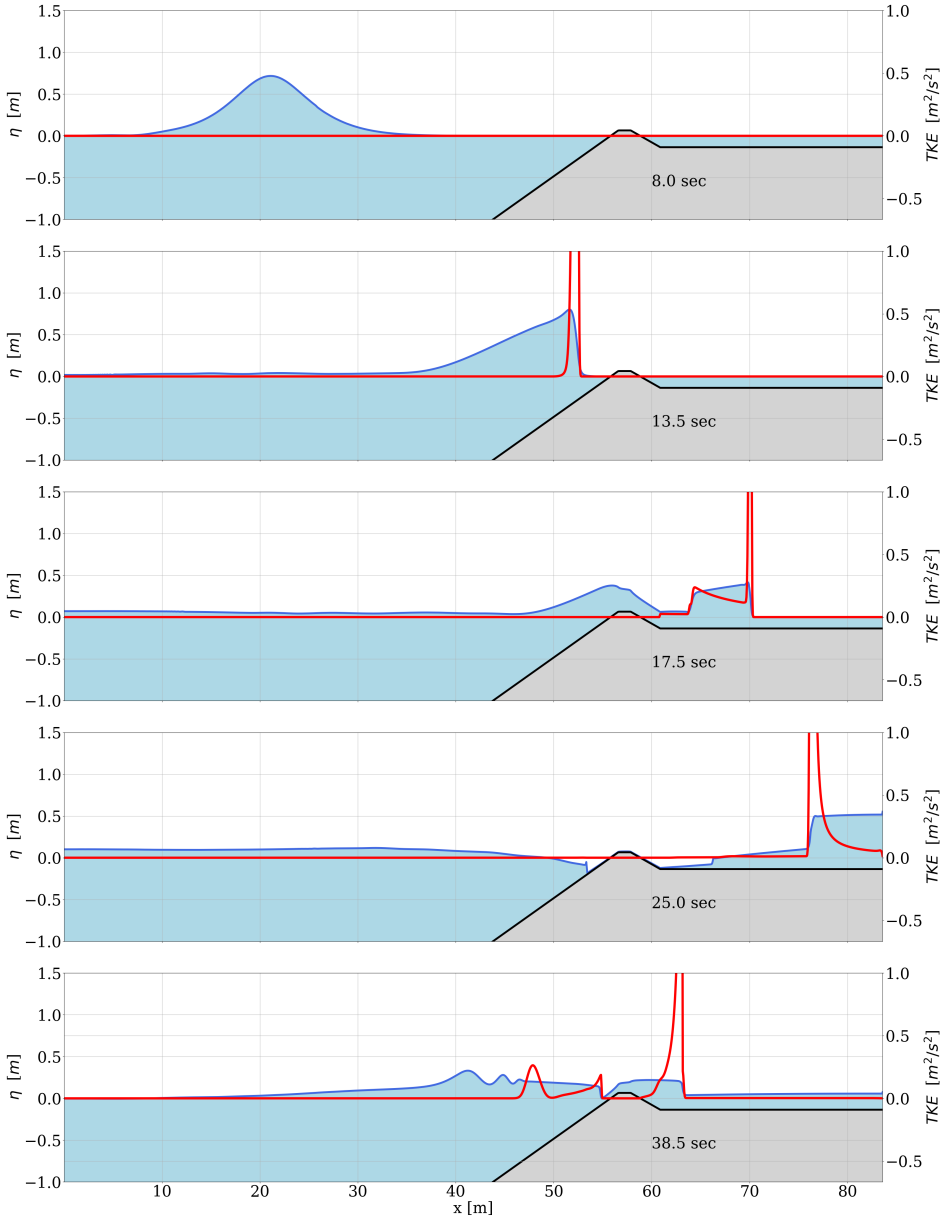


Figure 6.10: HIreef Benchmark test. Snapshots of free surface profiles for the propagation of solitary wave with $A/h = 0.3$ over 1:12 slope and exposed reef crest. Blue lines and red lines denote the water free surface and the TKE. The TKE vertical axis is zoomed in to highlight the shape of the TKE solution. During the computation, the TKE quantity reaches a maximum value of $33\,\mathrm{m}^2\,\mathrm{s}^{-2}$

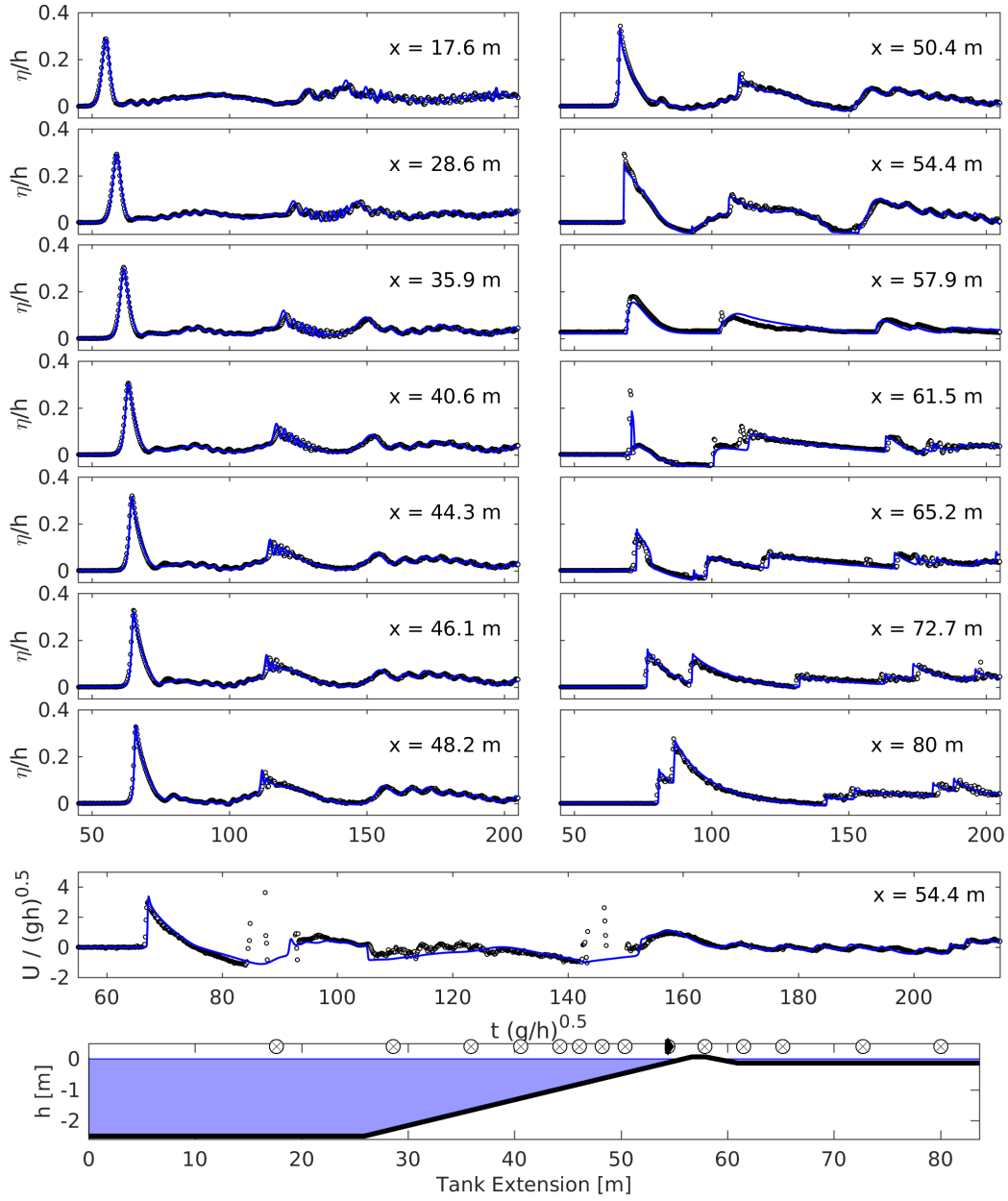


Figure 6.11: HIreef Benchmark test. Time series of the free surface at different wave gauges for the propagation of solitary wave with $A/h = 0.3$ over 1:12 slope and exposed reef crest. Solid lines and circles denote computed and measured data.

6.4.5 Breaking Solitary Wave over 3D Reef with a Cone

Swigler [2010] conducted a series of experimental tests to study the wave transformation over a three-dimensional reef. These experiments provide reliable test cases for the validation of dispersive wave models, especially when dealing with complex nearshore wave dynamics in three-dimensional settings: shoaling, wave breaking, overtopping, and run-up. Figure 6.12 shows the test case configuration produced by the laser scan and the instrumental layout for the gauges.

The model is set up with a grid size of $\Delta x = \Delta y = 0.05\,\mathrm{m}$, a Courant number of $Cr = 0.5$, and a Manning friction coefficient $n = 0.014\,\mathrm{m/s^{1/3}}$. The incident solitary wave has a height of 0.39 m, giving rise to strongly nonlinear conditions with $A/h = 0.5$. The solitary wave breaks at the apex of the reef flat, and the resulting surge completely overtops the cone. The refracted waves from the two sides of the cone and the diffracted waves converge in the back. During this process, The TKE in the x- and y-directions is produced and advected along the breaking wavefront. This leads to stable and accurate computations with the dispersive solution. The diffracted wave on the lee-side of the cone propagates up the slope, reinforcing the refracted waves from the reef edge. The flow partially reflected by the back wall of the basin recedes on the beach slope over small imperfections in the concrete surface. The drawdown then generates a bore, which collides with the reflection from the wavemaker over the reef. Small vortices are generated in the vicinity of the reef edge and are transported around the conical island. Figures. 6.15, 6.16, and 6.17 show the flow pattern of the free surface elevation over the entire domain at $t = 8\,\mathrm{s}$, $t = 17\,\mathrm{s}$, and $t = 35\,\mathrm{s}$ for further illustration.

Figure 6.13 compares the computed and recorded surface elevations over 90 s. The model reproduces the measured surface elevations in front of the cone and behind the cone where the wave collapse happens. As shown in Figure 6.14, the model matches the cross- and longshore-components of the velocity reasonably well. It should be noted that the velocities in the physical model were measured near the free surface whereas the numerical model computes the velocity close to mid-depth.

The overall agreement between the computed and recorded data demonstrate the validity of the new model in handling multiple hydraulic processes, transitions from hydrostatic to dispersion-dominated flows as well as a variety of wave-breaking scenarios in the two-dimensional horizontal plane with a moving boundary.

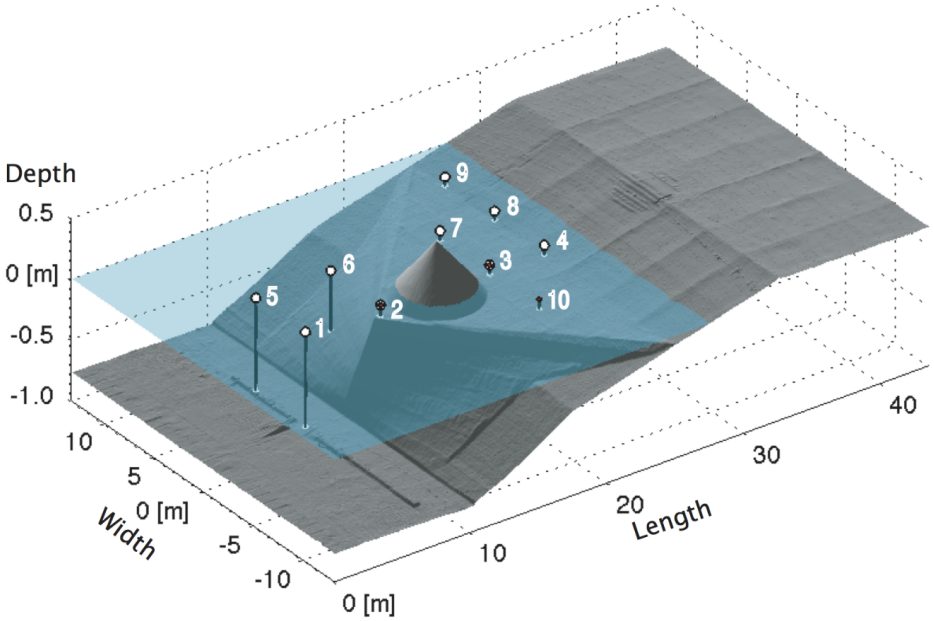


Figure 6.12: Breaking solitary wave over 3D reef. Layout with locations for resistance-type wire wave gauges and ADVs (positions 2, 3, and 10 only)

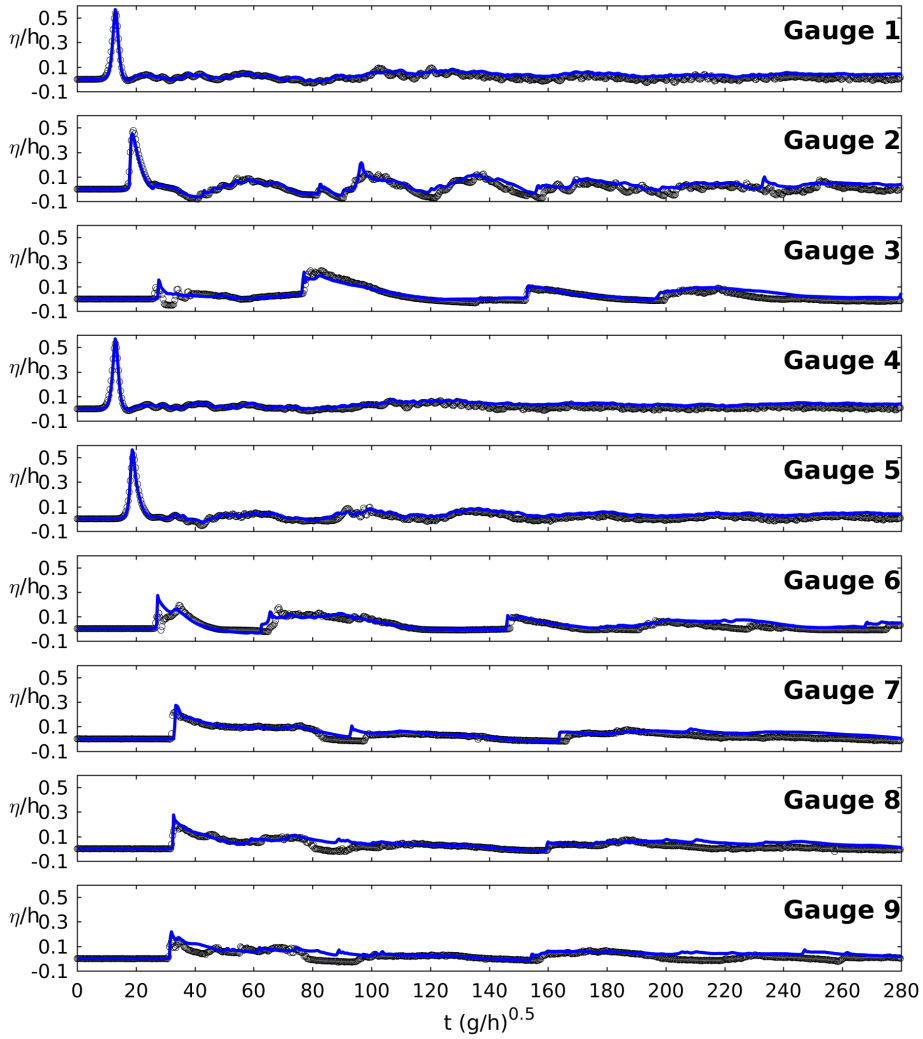


Figure 6.13: Breaking solitary wave over 3D reef. Free surface elevation at the gauges. The blue line denotes the computed result and the black circles represent the laboratory data.

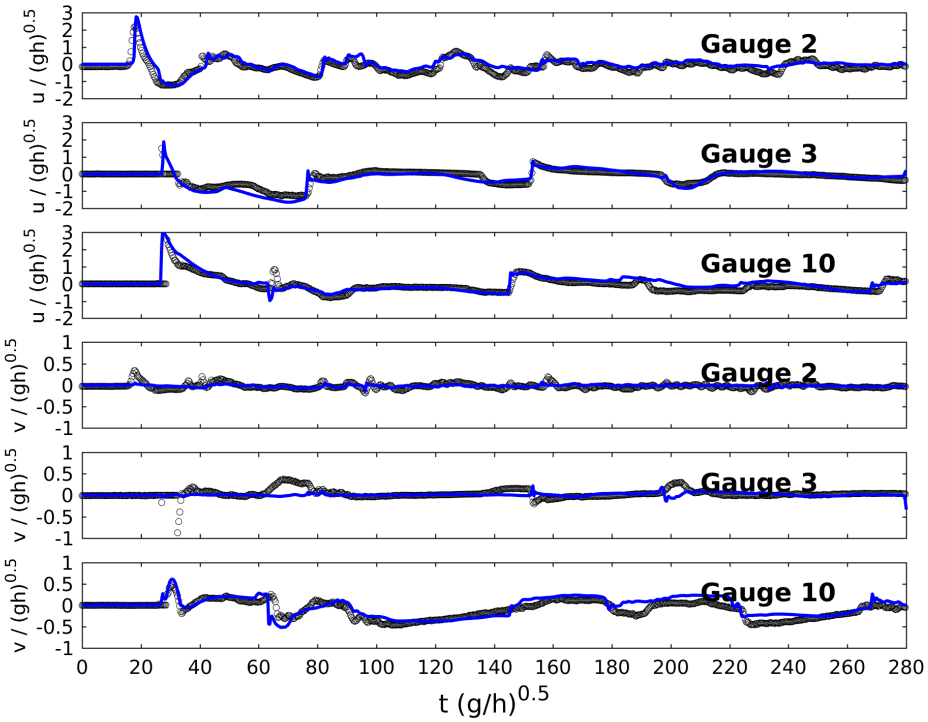


Figure 6.14: Breaking solitary wave over 3D reef. Horizontal flow velocities at the gauges. The blue line denotes the computed result, the black circles represent the laboratory data.

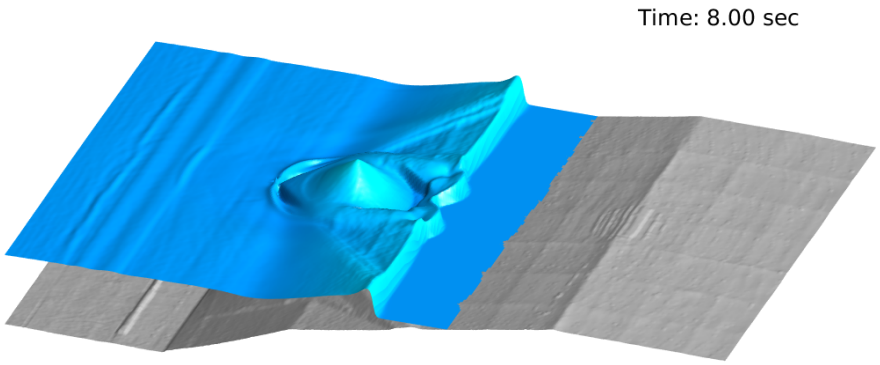


Figure 6.15: Breaking solitary wave over 3D reef. Free surface after $8\,\mathrm{s}$. Overtopping of the cone with strong vortex formation at lee side. Wave breaking front is described as flow discontinuity.

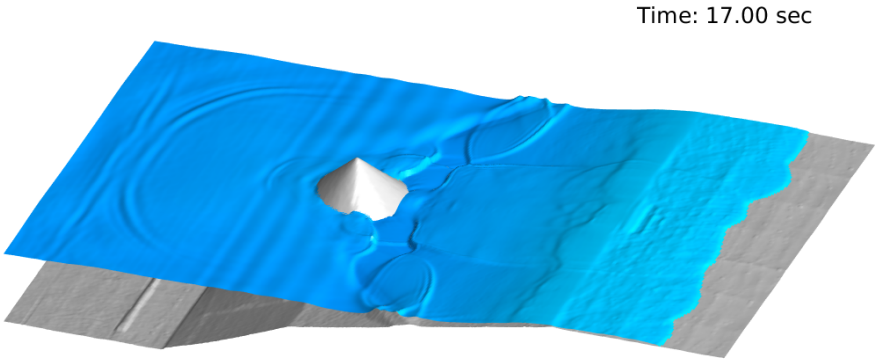


Figure 6.16: Breaking solitary wave over 3D reef. Free surface after $17\,\mathrm{s}$. Run-up on an initially dry concrete bed. The flow adjusts to the small roughness elements over the dry bed resulting from imperfections in the concrete pouring.

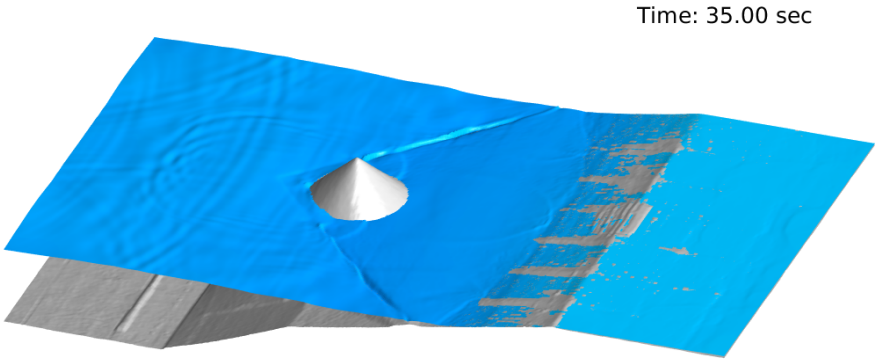


Figure 6.17: Breaking solitary wave over 3D reef. Free surface after $35\,\mathrm{s}$. Sloshing continues in the basin with small vortices advected around the cone. Some water is dripping down the slope.

6.4.6 Wave Refraction-Diffraction over an Elliptic Shoal

Berkhoff et al. [1982] conducted a laboratory experiment to study the refraction and diffraction effects of a monochromatic wave over complex bathymetry. This experiment remains a standard test for assessing the accuracy and performance of numerical models for computing wave shoaling, refraction, diffraction, and dispersive effects ($kh \simeq 1.9$). The experiment setup and the transects for collecting wave data are shown in Figure 6.19. A monochromatic wave with a period $T = 1\,\mathrm{s}$ and an amplitude $a = 2.32\,\mathrm{cm}$ is generated by the wavemaker at $x_s = 17\,\mathrm{m}$. The wave then propagates over an elliptic shoal resting on a plane beach with a constant slope of 1:50. The topographic contours on the slope are oriented at an angle of 20°. The detailed formula of the bathymetry can be found in Berkhoff et al. [1982]. The benchmark test is solved in a 36 m by 20 m computational domain with a square grid cell of $\Delta x = \Delta y = 5\,\mathrm{cm}$. Two sponge layers of a 2.5 m width are placed behind the source region and at the far end of the domain to absorb the reflected waves reaching the boundaries. We run the simulation for 50 s with a Courant number $Cr = 0.5$. The free surface at the end of the run is shown in Figure 6.18. The 3D display clearly showcases the diffraction pattern, as well as the wave focusing by refraction.

Figure 6.20 shows the comparison of wave height along all eight transects between the experimental data and the model results. The wave heights from the model are obtained by averaging the wave height of the last 10 waves (*i.e.*, from $t = 40\,\mathrm{s}$ to $t = 50\,\mathrm{s}$). A zero up-crossing method has been used to isolate the individual waves during the computation. The agreement of our numerical results with the experimental data is very satisfactory and compares very well to published results from other Boussinesq-type models (*e.g.*, Wei et al. [1999]; Tonelli and Petti [2009]; Wei and Kirby [1995]). The underestimation of the wave height in section (5) has been reported in other wave models (*e.g.*, Ricchiuto and Filippini [2014]; Tonelli and Petti [2009]; Karambas and Memos [2009]). This test demonstrates the ability of the present model to correctly account for the combined refraction-diffraction effects, resulting in a correct wave height distribution.

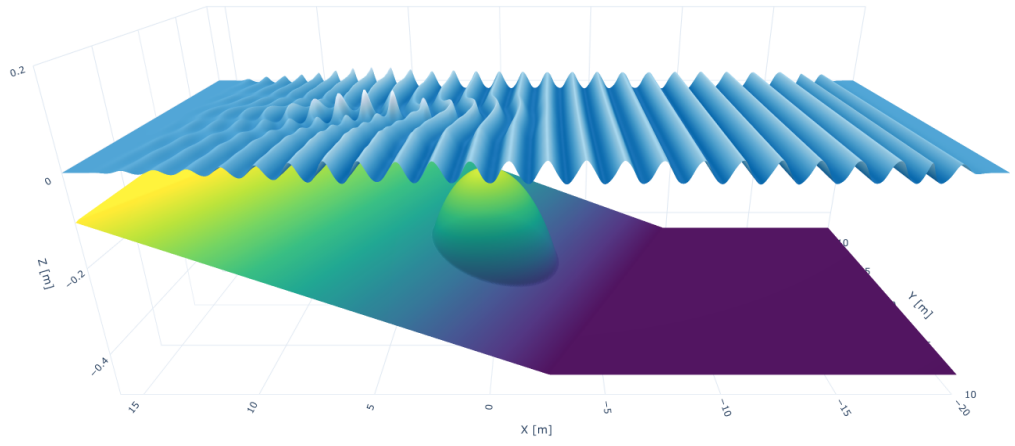


Figure 6.18: Wave refraction-diffraction over an elliptic shoal. Three-dimensional view of the free surface and the topography elevation at $t = 50\,\mathrm{s}$

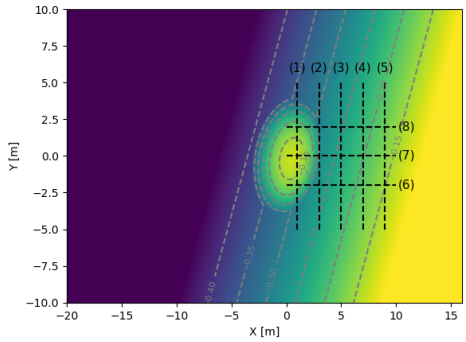


Figure 6.19: Wave refraction-diffraction over an elliptic shoal. Berkhoff et al. [1982] experimental layout and position of the transects of the collected data.

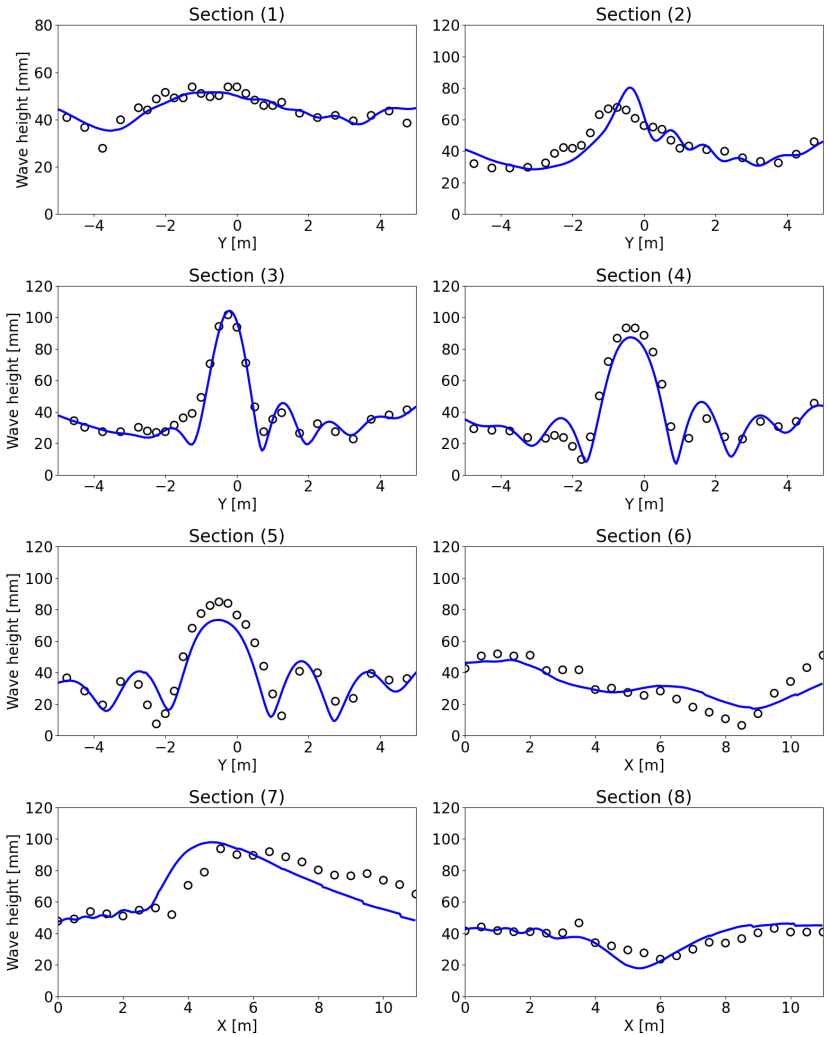


Figure 6.20: Wave refraction-diffraction over an elliptic shoal. Wave height comparison between the model and the experimental data along different transects (1)-(8). The blue line denotes the computed result, and the black circles represent the laboratory data.

6.5 Conclusions

This chapter details the implementation of the Present 2D model along with some key strategies used to reduce the computational cost and thus improve the performance of the computations. The new model combines the efficiency of the staggered grid approach with the enhanced dispersion properties of Nwogu's equations. Internal wave generation and wave-absorbing boundaries are implemented using standard and well-established techniques. A TKE-based eddy viscosity approach is used to account for the wave-breaking effects, which has proven to achieve accurate and consistent solutions across different grid resolutions. The accuracy of the Present model is evaluated using a series of standard benchmark tests describing a wide range of wave transformations such as wave-breaking, run-up, refraction-diffraction, and wave-shoaling. The overall agreement between the computed results and laboratory data demonstrates the capability of the model to compute important wave processes and validates this new numerical framework for the solution of Nwogu's equations.

Chapter 7

GPU-based Implementation

Contents

7.1 Introduction 140
7.2 GPU Programming Strategies 140
7.2.1 Heterogeneous Computing 141
7.2.2 Memory Hierarchy 142
7.2.3 Data Transfer 142
7.3 GPU Implementation 143
7.3.1 Timestep Reduction Algorithm 144
7.3.2 Tri-diagonal Solver 146
7.3.3 Wavemaker Source Term 147
7.4 Parallelization Performance 148
7.5 Conclusions 150

7.1 Introduction

GPUs have traditionally accelerated memory-intensive computer graphics tasks, including image rendering and graphic processing. Due to their multiple cores and high memory bandwidth, GPUs have become indispensable for tackling these embarrassingly parallel workloads. While GPU parallel computing began as a real-time, high-definition 3D graphics tool, it also proved effective for scientific computing applications. As a result, the technology has rapidly progressed over the past decade, supporting a wide range of general-purpose computing applications and providing a cheaper alternative to high-performance CPU clusters (Owens et al. [2007]). GPU parallelization has proved especially attractive in water wave modeling, where it has effectively been employed to boost the performance of numerous operational Tsunamis and long waves models (see Hagen et al. [2007]; Janßen et al. [2012]; de la Asunción et al. [2013]; Lacasta et al. [2014]; Sætra et al. [2015]). Several GPU implementations of Shallow Water solvers have shown significant speed gain over a traditional CPU parallelization (Lacasta et al. [2014]; Arce Acuña and Aoki [2018]). This performance increase is due in part to the hyperbolic nature of the SWE solution, which lacks data dependency and therefore maps extremely well on the GPU's highly-parallel structure.

In the case of Boussinesq type-wave models, the GPU implementations of these models have not been reported until recently. Tavakkol and Lynett [2017] presented "Celeris", a GPU-based program for interactive simulation and visualization of nearshore waves. Using a hybrid FV-FD method, the model solves the extended Boussinesq equations and is able to achieve real-time computation and visualization. Kim et al. [2018] has successfully accelerated their fully nonlinear Boussinesq model using CUDA Fortran. The new GPU implementation achieved up to a 20-fold performance increase compared to the serial CPU version. Recently, Yuan et al. [2020] presented the multiple-GPU acceleration of FUNWAVE-TVD using MPI coupled with CUDA Fortran. Efficiency analysis reveals that, compared to the CPU version operating on a 36-core HPC node, single-GPU and double-GPU runs had speedup ratios of 4-7 and over 10, respectively.

This section describes the CUDA C/C++ GPU implementation of the current Boussinesq model. We also go through the optimization strategies that were used to attain improved performance.

7.2 GPU Programming Strategies

Several GPU-based models have shown superior performance versus their CPU-based counterparts. The success of GPU computing stems from its high performance; when comparing theoretical peak bandwidth and gigaflops performance, GPUs presently outperform CPUs by a factor of seven (Brodtkorb et al. [2012]). This performance gap is the result of architectural differences between the two processors. In terms of hardware design, a CPU comprises a few cores with a significant amount of cache memory, optimized to perform sequential tasks at a very high frequency. On the other hand, a GPU is made up of thousands of smaller cores that can manage numerous threads simultaneously, making them ideal for repetitive and highly parallel computing operations. Because CPUs' most critical performance factors have traditionally been their steadily increasing frequency, these processors have been the most affected by the "Power Wall" (Brodtkorb et al. [2013]). While CPUs have reached their serial performance ceiling quite

quickly, GPUs have grown exponentially in performance due to massive parallelism. In the case of phase-resolving wave models, where a significant portion of the model exhibits embarrassingly parallel workloads, GPU parallelization has emerged as a cost-effective means of enhancing computational performance without compromising accuracy.

7.2.1 Heterogeneous Computing

GPU programming tools and libraries have evolved substantially in recent years to harness the processing power of GPUs. Several programming languages were introduced to accelerate algorithms outside the field of graphics and image rendering. The most prominent ones are: NVIDIA's CUDA, OpenCL, and Microsoft's Direct-compute (Sætra et al. [2015]). CUDA C/C++ was chosen as the interface in this study for the graphic implementation of the numerical framework outlined in previous sections. The choice of CUDA was prompted by its maturity as well as the availability of a diverse set of libraries and tools for enhanced GPU utilization.

The CUDA programming language is a heterogeneous model, in which both the CPU (*host*) and GPU (*device*) are optimized through a combination of serial (CPU) and parallel (GPU) executions. The parallel computation is expressed through kernel functions, which execute the same operation simultaneously on thousands of threads, each targeting a different set of data. Each thread is assigned a thread ID, which it uses to determine which piece of data to process. Threads are organized into blocks and grids. Threads within the same block can synchronize and collaborate by accessing fast shared memory (see Figure 7.1).

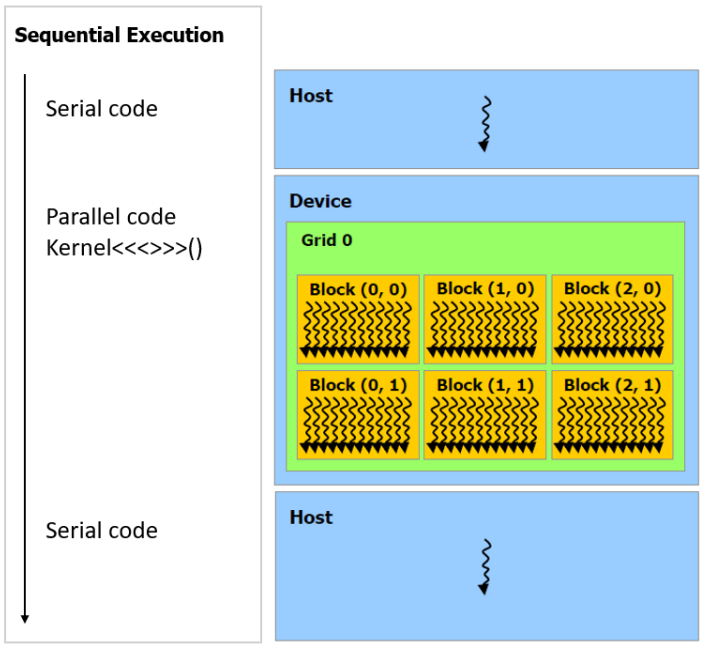


Figure 7.1: CUDA heterogeneous computing Guide [2013]

CUDA offers a variety of built-in variables and structures to define the number of blocks in a grid and the number of threads in a block. This translates to hardware in the sense that a block

runs on a single multiprocessor (SM), and a single multiprocessor can run several blocks in a time-sliced manner. As a general rule, to maximize GPU occupancy, it is required to match the Block/Grid sizes with the data and the hardware computing capability. Since kernels issue instructions in warps (32 threads), the number of threads should always be a multiple of 32. The number of blocks is then chosen accordingly so that for each launch, the GPU threads are fully occupied without exceeding the maximum number of active threads per SM.

7.2.2 Memory Hierarchy

In order to minimize memory latency, GPU programming must manage a variety of memory types. In this section, we will briefly discuss three of the most common memory types used in GPU programming:

- **Registers:** In computer science, registers are the fastest memory and the closest to the processors. For a variable to be stored as a register, it has to be locally declared inside a kernel. Despite the high amount of registers per SM, a trade-off exists between the number of active threads and register usage. In a SM with thousands of active threads, registers are a limited resource for each thread and can slow down the computation (Yuan et al. [2020]).
- **Shared memory:** Each SM has a defined amount of fast-retrieving shared memory that is shared by all threads inside the same block. Threads within the same block can thus access data in shared memory that has been loaded from global memory by other threads. This functionality, together with thread synchronization, is critical for high-performance thread-interdependent algorithms such as parallel reduction.
- **Global memory:** The global memory is the name given to the main GPU memory. This memory has the life time of the application and is accessible from all the threads. Even though GPU cards have substantially higher memory bandwidth than CPUs, they can still have memory latency issues. As a general rule, maximum memory performance should be pursued by memory coalescing, which entails a warp accessing each consecutive 128-byte (32 single precision) in a single transaction.

7.2.3 Data Transfer

Successful and efficient GPU implementation requires careful consideration of data transfer between the CPU and the GPU. The CUDA programming model assumes that both the *host* and the *device* have their own distinct global memories. As these memories are separated, data has to be occasionally transferred between the *host* and the *device* to make it accessible to both the CPU functions and the GPU kernels. Since data transfers between the CPU and the GPU pass through the PCI Express bus both ways, these transfers are costly in terms of performance and will often be a bottleneck. This latency can be kept to a minimum by using streams and asynchronous memory transfers, as well as keeping the data on the GPU as much as possible.

7.3 GPU Implementation

The explicit solution of Nwogu's equations requires stencil computations, which are embarrassingly parallel and thus ideal for massive parallelization. Apart from the solution of the tri-diagonal systems required by the dispersive equations, the numerical algorithm has no data dependency. As a result, parallelization at the grid-cell level is possible, with each GPU thread computing for one cell.

To minimize memory transfer, the computation of the numerical solution is executed solely on the GPU, the memory transfer is therefore limited to input data (copy initial conditions from *host* to *device*) and to output results (copy results from *device* to *host*). The GPU workload is divided into a set of sequential CUDA kernels to synchronize GPU threads after each operation. The general steps for the GPU implementation are illustrated in the flowchart of Figure 7.2. The numerical solution is achieved through the following sequential operations:

1. The program starts on the CPU by loading the initial and boundary conditions. In addition, several variables for bottom friction, wave breaking, and wavemaker input are defined. To avoid overloading the GPU's global memory, some preprocessing work is performed on the CPU, and only the most relevant data is then transferred to the GPU for computation.

2. On the GPU, the adaptive time step is computed for the solution of Nwogu's equations. A kernel determines the maximum wave speed in the domain using an optimized reduction algorithm (Harris et al. [2007]). The time step is then computed using the CFL condition.

3. The dispersive terms are computed first using the updated values of the velocities and the still water depth. These high-derivative terms are then efficiently stored to be used as source terms in the continuity and momentum equations, as well as the solution of TKE.

4. The continuity equation is solved by computing the SW flux terms and then adding the dispersive and wavemaker terms to the equation. It's important to note that since the output of the continuity equations is used in the momentum equations (staggering in time), the threads must wait for the continuity equations to be solved for the whole domain before computing the momentum terms.

5. The TKE equation is then solved and added to the momentum equations as a source term, along with dispersion and friction terms. Since the scheme does not require large stencils (2-point stencils to achieve second-order accuracy), threads can access the necessary information directly from the global memory. It should be noted that in the case of larger stencils, it can be beneficial to copy data to the shared memory for faster access.

6. As previously stated, resolving Nwogu's equations requires solving numerous tri-diagonal systems at each time step. For enhanced speed and accuracy, these systems of equations are solved using the Thomas algorithm, similar to the CPU version.

7. Since kernels are launched from the CPU, the variable Δt is passed to the CPU after each time step to track the overall computational progress. The transfer of one floating variable takes almost no time and doesn't affect the computation time. The CPU will stop launching kernels and exit the program once the computed time t reaches t_{end}.

8. Heavy data transfer between CPU and GPU can be avoided by processing the results directly on the GPU.

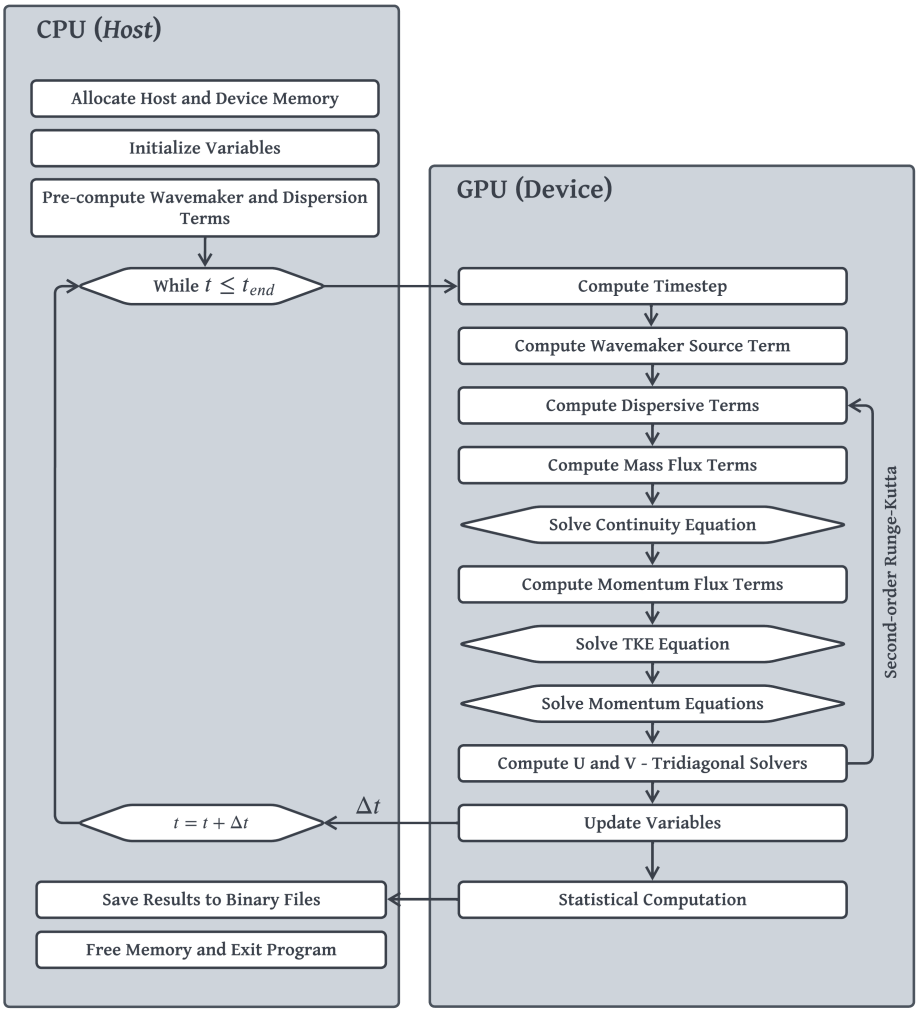


Figure 7.2: GPU Implementation Flow Chart

7.3.1 Timestep Reduction Algorithm

The computation of the adaptive time step requires finding the maximum wave speed in the entire domain at each time loop. In the case of large domains, this operation requires the implementation of an efficient GPU-based reduction algorithm. Reduction techniques are quite standard in parallel computing, although they might pose some challenges when dealing with GPU processing. The main difficulty is achieving a device-level reduction where all threads are utilized. Since global synchronization is not achievable across blocks in current GPU hardware, the problem has to be decomposed into two kernels. The first kernel generates and stores partial reduction in each block, and the second kernel reduces the partial results into one single maximum value (see Figure 7.3).

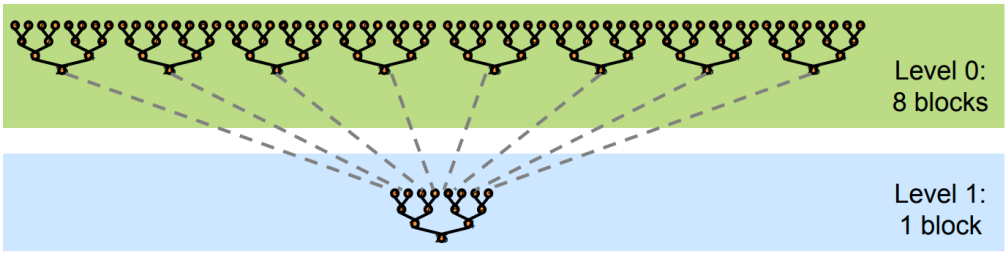


Figure 7.3: GPU Global Reduction Harris et al. [2007]

The first kernel computes the maximum value across grids using grid-stride loops, then each launched block copies one portion of the array into shared memory and then reduces it to one value, which is then written back to global memory as $maximum[blockIdx.x]$:

```
__global__ void find_maximum_wavespeed(flux& NF, float *maximum, unsigned int n)
{
        unsigned int index = threadIdx.x + blockIdx.x*blockDim.x;
        unsigned int stride = gridDim.x*blockDim.x;
        unsigned int offset = 0;

        // Allocate shared memory in each block
        __shared__ float cache[numThread_CFL];

        // Find the maximum stride
        float temp = -1.0;
        while (index + offset < n) {
                temp = fmaxf(temp, NF.WS[index + offset]);
                offset += stride;
        }
        cache[threadIdx.x] = temp;
        __syncthreads();

        // Reduction algorithm to find the maximum value in each block
        unsigned int i = blockDim.x / 2;
        while (i != 0) {
                if (threadIdx.x < i) {
                        cache[threadIdx.x] =
                            fmaxf(cache[threadIdx.x], cache[threadIdx.x + i]);
            }
                __syncthreads();
                i /= 2;
        }
```

```
        maximum[blockIdx.x] = cache[0];
}
```

Similarly, in the second kernel, one block will be used to reduce the array *maximum* to one maximum value, which will be then used to compute Δt.

7.3.2 Tri-diagonal Solver

The most challenging aspect of the GPU-parallelization of Boussinesq models is the solution of the data-dependent tri-diagonal system. The standard algorithm to solve such problems is the Thomas algorithm. This approach, which involves forward elimination and backward substitution, is commonly used in several Boussinesq-type models. However, due to the method's serial nature, various parallel algorithms that map better to the GPU environment have been adopted (*e.g.*, Cyclic Reduction, Parallel Cyclic Reduction, Recursive Doubling, see Zhang et al. [2010]).

In the case of a 2D problem with a mesh grid of $(m \times n)$, we expect to solve n diagonal systems of size m for the x-momentum equation, along with m diagonal systems of size n for the y-momentum equation. All of these systems are independent of one another and can be solved concurrently. For a straightforward GPU implementation, we may assign each tri-diagonal system to one separate thread and solve the system sequentially with a Thomas algorithm. In this case, only $n + m$ threads will be utilized with each kernel launch. Another approach is to use the optimized tri-diagonal solvers provided by NVIDIA's cuSPARSE library. Solvers such as *cusparseSgtsvStridedBatch* are able to compute the solutions of multiple tri-diagonal linear systems at once. This routine employs a combination of Cyclic Reduction (CR) and Parallel Cyclic Reduction (PCR) algorithms. To achieve improved performance, the solver does not perform any pivoting. This, however, may come at the expense of accuracy and stability.

On an Nvidia Quadro RTX 3000 card, we run a simple benchmark test to assess the performance of each approach. For different domain sizes $n \times n$, we solve the $2 \times n$ tri-diagonal systems using both the Thomas algorithm and the *cusparseSgtsvStridedBatch* solver. T_{Thomas} and T_{gtsv} are the time required to solve the systems with each approach. The values are averaged over 10,000 runs.

It's worth noting that, owing to the optimization described in Chapter 6.3.2, the forward sweep of the Thomas method is cut in half. Therefore, we only need to compute two of the three loops required by this algorithm. As shown in table 7.1, the Thomas implementation outperforms the *cusparseSgtsvStridedBatch* for large domains. Even with an intrinsically serial structure and low GPU occupancy, the algorithm's simplicity and prior optimization help achieve good performance compared to the more complex reduction solvers.

Table 7.1: Performance of Thomas Algorithm and *cusparseSgtsvStridedBatch* in Solving Nwogu's tri-diagonal Systems

	Grid Resolution			
	512×512	1024×1024	2048×2048	4096×4096
ThomasAlgorithm	3.96×10^{-4} s	1.38×10^{-3} s	4.18×10^{-3} s	1.62×10^{-2} s
cusparseSgtsv	6.47×10^{-4} s	1.40×10^{-3} s	2.99×10^{-3} s	4.67×10^{-3} s
T_{gtsv}/T_{Thomas}	0.61	0.98	1.39	3.46

7.3.3 Wavemaker Source Term

The implementation of the wavemaker source term involves computing the following sum for each grid point along the y-axis:

$$s\left(y,t\right) = \sum_{i=1}^{M_\omega} \Big[\mathcal{A}_i(y)\cos(\omega_i t) - \mathcal{B}_i(y)\sin(\omega_i t)\Big]. \tag{7.1}$$

A two-level parallelization is employed to maximize GPU-thread utilization: a *block-level* parallelization to loop over the y-direction grid points and a *thread-level* parallelization to compute the sum over the wave spectrum frequencies using a reduction algorithm.

```
__global__ void ComputeSyt(WaveMaker& WM,  float t, unsigned int M)
{
        unsigned int j = blockIdx.x;

        float sum = 0;
        for (unsigned int f = threadIdx.x; f < WM.Nf; f += blockDim.x)
        {
                float ramp = tanh(t*WM.Omg[f]/(2*d_PI));
                sum = sum
                        + ramp*WM.Ak(f,j)*cos(WM.Omg[f]*t)
                        - ramp*WM.Bk(f,j)*sin(WM.Omg[f]*t);
        }

        // Resduction algorithm
        __shared__ float cache[numThread_WM];

        cache[threadIdx.x] = sum;
        __syncthreads();

        for (int size = numThread_WM/2; size>0; size/=2) {
```

```
        if (threadIdx.x<size)
            cache[threadIdx.x] += cache[threadIdx.x+size];
        __syncthreads();
    }

        if (threadIdx.x == 0)
        WM.Syt[blockIdx.x] = cache[0];

}
```

7.4 Parallelization Performance

With the development of high-performance computing, the solution of complex water flow problems over large computational domains has become accessible through advanced parallel computing. Parallelization has become a crucial component for the development of operational numerical models for forecasting purposes. Historically, most of the phase-resolving wave models are based on CPU multi-core executions. However, as GPU computing gains popularity, numerical models are increasingly relying on the high speed of GPU cards to boost their performance. The difference in performance and scalability between CPUs and GPUs stems from their different hardware architectures. CPUs are composed of a small number of cores with very high frequency in order to execute a few tasks very quickly. In contrast, modern GPUs are composed of hundreds of cores that can handle thousands of threads simultaneously, however, at a lower clock rate. In the following, we will compare the computation speedup of the Present model from two fundamentally different types of parallelization techniques:

- A standard CPU parallelization with OpenMP.
- A GPU parallelization with CUDA.

The two model versions: a multi-core CPU parallelization with OpenMP directives and a GPU-based massive thread parallelization. To evaluate the performance of the two parallelizations, the new numerical model is executed: (1) in mono-threaded CPU sequential mode, (2) in CPU parallel mode with OpenMP, and (3) in GPU parallel mode with CUDA C/C++. The three techniques solve the same algorithm and execute identical operations. It has been confirmed that the results from the three implementations are identical to machine precision. The speedup of both parallelizations is evaluated as the ratio: $T_{serial}/T_{parallel}$, where T_{serial} and $T_{parallel}$ are the computation time needed for the execution of the serial and parallel codes, respectively.

The serial and parallelized programs are executed on the following hardware systems:

Table 7.2: Hardware specifications

Specifications	System 1	
	CPU	GPU
Processor	Intel Core i9-9880H	NVIDIA Quadro RTX 3000
Number of Cores	8	3840
Base Frequency [MHz]	2300	600
Memory bandwidth [GB/s]	41.8	288.0

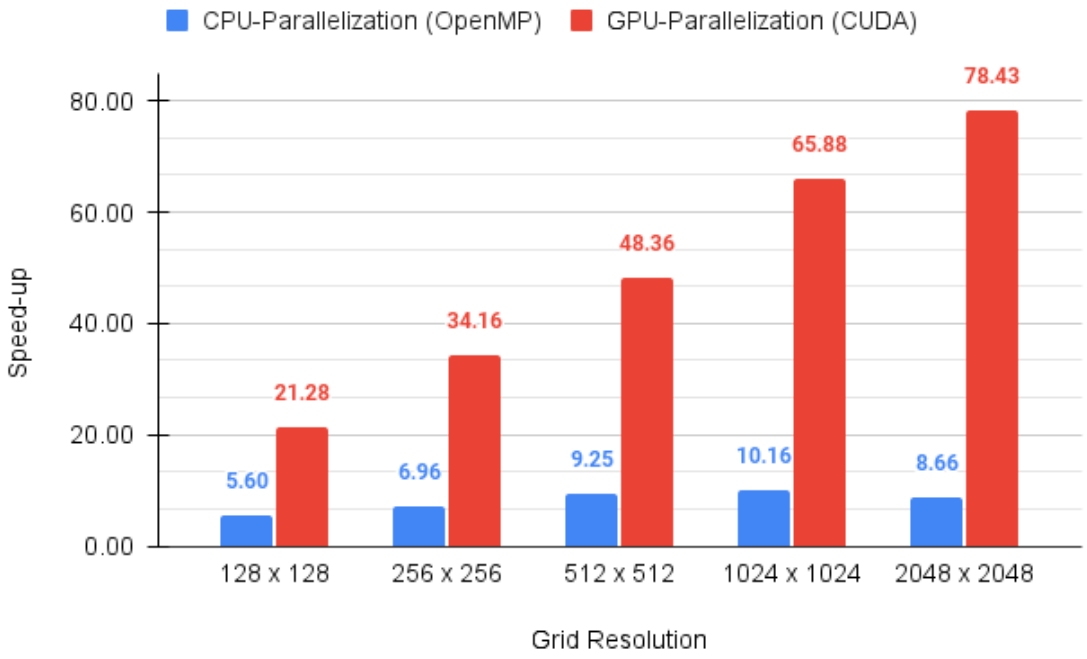


Figure 7.4: Ratios of CPU and GPU computational speedup times for different domain sizes (total number of cells) with System 1 hardware

It is important to note that the two CPU and GPU cards rate comparably regarding performance and price in their respective categories.

After multiple tests with different wave problems, it was confirmed that the speedup times of the parallel implementation depend mainly on the total number of grid cells used in the computation along with the hardware specifications. We consider the dispersive Gaussian drop simulation described in Chapter 6.4.1 as a benchmark test to evaluate the computation time for each implementation. Figure 7.4 illustrates the performance of CPU- and GPU-based runs with different grid sizes. It is not surprising that the model runs faster in parallel than in serial mode. It is also evident that even for relatively small matrices, a parallelization algorithm leads to much faster completion of the numerical solution. In the case of OpenMP parallelization on a CPU, the speedup quickly reaches a plateau with the number of cells. With GPU parallelization on an

NVIDIA graphics card, the speedup increases with the number of cells, showing high efficiency, especially for large computational domains. This is due to the high number of threads that can run concurrently. For the hardware used in this analysis, the computation on the graphics card is approximately an order of magnitude faster than a comparable CPU parallelization for a problem with over 4 million grid cells. Overall, the speedup gained from the CUDA C/C++ implementation is of significant magnitude and far superior to standard CPU parallelizations on conventional hardware of comparable cost.

7.5 Conclusions

This section outlines the GPU-based implementation of the new Boussinesq-type model using the CUDA C/C++ framework. Thanks to the explicit nature of the numerical scheme, which caters to massive data parallelism, the model's GPU implementation achieves higher performance compared to conventional multi-core CPU parallelization. The GPU's computing speed increases with the number of grid cells, confirming its high performance when dealing with large-scale domains and massive computational workloads. The high performance of the GPU solution is an essential step towards reducing the computation time of large-scale flow problems and caters to the intended use of this code for operational forecasting procedures.

Chapter 8

Applicability for Operational Scenarios

Contents

8.1 Introduction 152
8.2 Study Site and Field data 152
8.2.1 Swell Events 152
8.2.2 Bathymetry and Model Settings 153
8.2.3 Input Spectrum 155
8.3 Numerical Results 157
8.3.1 Accuracy of the Computed Results 157
8.3.2 Grid Sensitivity 160
8.3.3 Model Performance 163
8.4 Conclusions 164

8.1 Introduction

This final chapter showcases the potential of the Present model to be used in an operational setting where it is necessary to obtain accurate solutions over large computational domains in a reasonable amount of time. Two typical scenarios were chosen, for which field data are available for validation. For both applications, computational accuracy is important, however, in an operational setting where multiple scenarios have to be evaluated in a given time, the computational speed is a critical factor.

The test cases involve a range of wave transformation processes that are characteristic for energetic nearshore waves over a complex reef-dominated bathymetry, which is generally challenging for depth-integrated models. The objective is to obtain stable and accurate results of a computed time of several hours for a domain with over 2 million cells. The accuracy is validated with field data available over a transect across the reef for one test case and inside a small boat harbor basin for the other scenario.

In addition to validation with field data, the output from the Present model is compared with the results computed by BOSZ, which was previously proven to be able to handle the two swell scenarios successfully. Both models are set up in the most consistent way possible, *i.e.*, with identical boundary conditions such as the bathymetry grid, wave spectra, phase angles, and bottom friction. The differences between the models are then solely due to the internal numerical solution structures.

It is important to notice that comparisons with field data do not necessarily prove a model's accuracy, which is generally checked with laboratory benchmark tests under controlled conditions or even with analytical solutions. It is obvious that the exact wave field observed by a particular field data set cannot be reproduced, since not all boundary conditions are known in detail. Limitations range from underlying tidal fluctuations, which are not accounted for in most models, and changes in the wave conditions over the course of the observation period, to uncertainties in the local bathymetry. It is challenging to quantify the amount of expected discrepancy between field data and computed model output. Nevertheless, comparisons between field data and model output are still useful to show how closely a model is able to replicate the major wave processes or how sensitive the results are to the model setup. This information can be used as an indicator of whether a model is generally applicable to a certain problem or whether it can be used for other wave scenarios of similar nature.

8.2 Study Site and Field data

8.2.1 Swell Events

The study site is a stretch of coastline along the Northshore of Oahu. The Northshore of Oahu is frequently hit by energetic open ocean swells that transform and break over a fringing reef. Strong currents and run-up paired with substantial IG wave generation are the consequences. Since irregular bathymetry is a unique aspect that cannot be fully addressed in laboratory experiments, data from field observations can serve as validation of a model's capability to handle challenging wave conditions over large domains in a stable and efficient way.

Azouri [2016] has shown that energetic swell forcing events trigger a response at all IG frequencies; *i.e* forcing a model with such energetic input allows for analyses of the underlying physics, as nonlinear interactions push the model towards large amplitude responses. Over the reef, the wave transformation can lead to strong responses at IG frequencies, whereas in an enclosed basin, such as a harbor, specific structures with particular frequencies corresponding to the normal modes of the basin are expected.

The US Army Corps of Engineers PILOT project collected wave measurements across fringing reefs on a number of Pacific and Caribbean islands. In addition, Azouri [2016] conducted an intensive field campaign in Haleiwa Harbor for the investigation of IG-waves. Both data sets are quality-controlled and have been previously used for model-data comparison. Details of the two selected swell scenarios are shown in Table 8.1. Swell 1 is used for validation of the Present model with the data at Mokuleia, and swell 2 serves as a baseline for the comparison inside Haleiwa Harbor. For both scenarios, output data are saved at 1 Hz and converted to power spectral density to be compared with the field data.

Table 8.1: Wave conditions and tide levels for the two test scenarios

Swell	Time spectrum	Time Tide	Hs [m]	Tp [sec]	Dp [°]	Mean Tide [m]
1	2008/01/13, 22:40	20:00:24:00	6.15	18.2	310	-0.13
2	2014/01/23, 06:55	04:30:08:30	7.41	15.4	315	0.04

8.2.2 Bathymetry and Model Settings

The bathymetry files used in this study are based on a merging of three different datasets: (i) 50 m resolution multibeam data for the offshore region; (ii) 5 m resolution Lidar data for the coastal region (from USACE's 2013 survey), and; (iii) 3 m resolution echo-sounder dataset for Haleiwa Harbor. The bathymetry is shown in Figure 8.1 with the computational domain outlined by the solid rectangle. The inner dashed rectangle denotes the extent of the sponge layer for wave absorption, which is generally not used for the interpretation of the results.

The domain is 12.2 km (alongshore) x 10.2 km (cross-shore) large and stretching from Mokuleia to Laniakea, with a grid resolution of 7 m x 7 m (see Figure 8.1). This particular grid resolution is a compromise between sufficiently high resolution to achieve converging results - especially in narrow parts of Haleiwa Harbor that are on the order of tens of meters, and the need for a reasonable turnaround time of the computations given that the domain results in over 2.5 million cells. An additional test was conducted with a much coarser resolution of 21 m x 21 m to examine the sensitivity of the quality of the results to numerical diffusion and the associated speed up of the computation for only $1/9^{th}$ of the total initial cell count. The bottom friction in all grids is set to constant $0.035\,\mathrm{sm}^{-1/3}$ to account for the roughness of reef-type bathymetry (see Bretschneider et al. [1986]).

Table 8.2: Summary of all model runs.

Run	Time [hr]	Domain [km x km]	Grid Res [m]	Cell count	H_s [m]	T_p [sec]	D_p [deg]
1	4	12.2 x 10.2	7	2,539,592	6.15	18.2	310
2	4	12.2 x 10.2	7	2,539,592	7.41	15.4	315
3	4	12.2 x 10.2	21	282,177	6.15	18.2	310
4	4	12.2 x 10.2	21	282,177	7.41	15.4	315

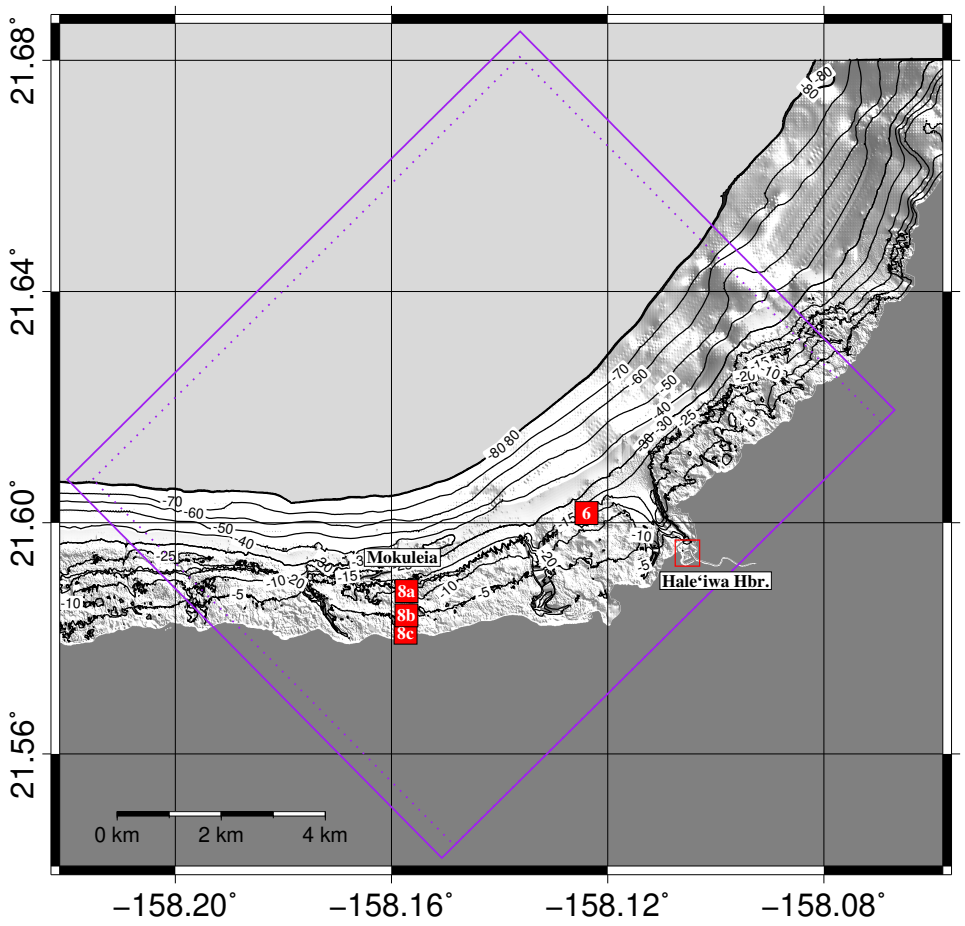


Figure 8.1: Computational domain covering the inner portion of the Northshore of Oahu as denoted by the solid rectangle. The inner dashed rectangle denotes the extent of the sponge layer for wave absorption. The offshore boundary is pointing towards 315° (-45° rotation from true North). The gauges at Mokuleia are denoted by 8a, 8b, and 8c. Haleiwa Harbor is shown in Figure 8.2 in detail. Map adapted from West Maui Coastal Resilience Group report Phase I.

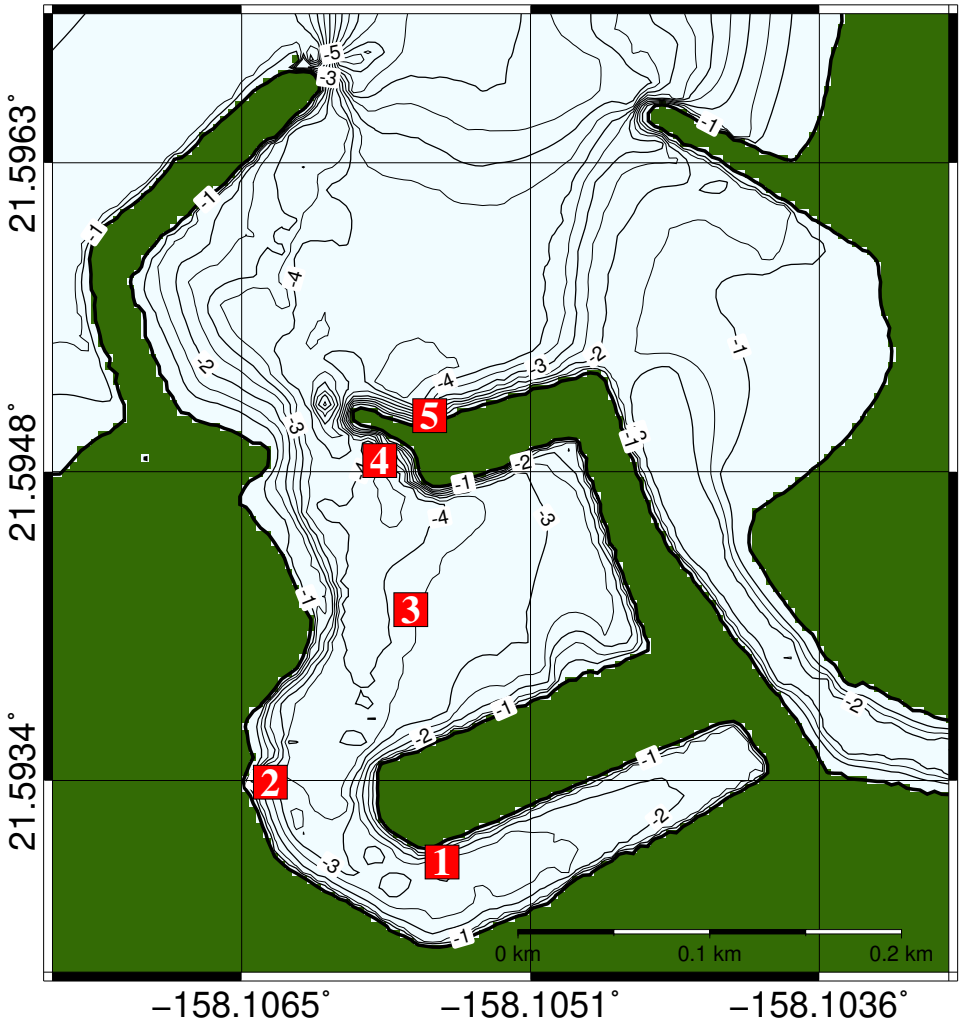


Figure 8.2: Close-up view of Haleiwa Harbor (small rectangle in Figure 8.1. Only gauges 2, 3, and 4 are selected for model-data comparison. Map adapted from West Maui Coastal Resilience Group report Phase I.

8.2.3 Input Spectrum

The wave spectra for the two swell come from CDIP and were measured by the Waimea buoy in 200 m water depth. The original spectrum from swell 2 is shown in Figure 8.3. The spectrum shows two pronounced peaks with the larger of the two centered at 315°. This is the peak direction. Another noticeable feature of the CDIP spectra is its bilinear frequency binning of 0.005 Hz up to 0.095 Hz and of 0.01 Hz above 0.11 Hz. There is one transitional frequency of 0.1013 Hz in between the two linear ranges covering 64 frequencies in total (0.025 - 0.58 Hz).
It is important to notice that usage of the original spectrum for the wavemaker would ultimately lead to a recycling of the input time series at $1/\min(\Delta f) = 1/0.005\,\text{Hz} = 200\,\text{s}$ similar to what is shown in Chapter 6.2.3. The frequency binning has to be decreased to avoid artificial wave groupings over the 4-hour computation so that $1/\min(\Delta f) \geq 4$*3600 sec. The smallest Δf in the frequency distribution should therefore be 0.000 069 4 Hz - instead of the original 0.005 Hz in the

CDIP spectrum. This requires 1104 uniformly-spaced frequencies over the range from 0.025 Hz to 0.1 Hz.

The upper-frequency limit is due to the limited dispersion properties of the governing equations as detailed in Chapter 6.2.3. Therefore the high-frequency tail of the CDIP spectrum has to be truncated. We select a cut-off threshold of $\lambda_{min} = 2h$, which means that the shortest input wave is twice as long as the offshore water depth. This returns a wavelength of 160 m or a kh-value of π, which relates to the highest input wave frequency of 0.1 Hz. Consequently, the bathymetry is limited to 80 m depth. Since the spectra from the CDIP buoy contain the vast majority of their energy in wave frequencies lower than 0.1 Hz, the truncation has little impact on the composition of the resulting time series from the superposition of all waves in the spectrum. H_s or the total energy is maintained, since the energy from the truncated tail is re-distributed over the remaining spectrum. It should be noted that the truncation only affects the input waves, but not the evolution of the wave field itself over the course of the computation; *i.e.* it is well possible that shorter waves than 0.1 Hz develop locally during the computation.

It is clear that a wave spectrum in 80 m water depth would likely be different from a spectrum obtained from a buoy in 200 m water depth. The shoaling process leads initially to a flattening of the wave shape before it causes the typical steepening before breaking. A linear wave of 18 s peak period experiences the maximum effect of the negative shoaling process in about 80 m water depth; *i.e.* it can be assumed that the peak of the gravity swell band is overestimated in the CDIP spectrum when used as input at only 80 m maximum depth offshore.

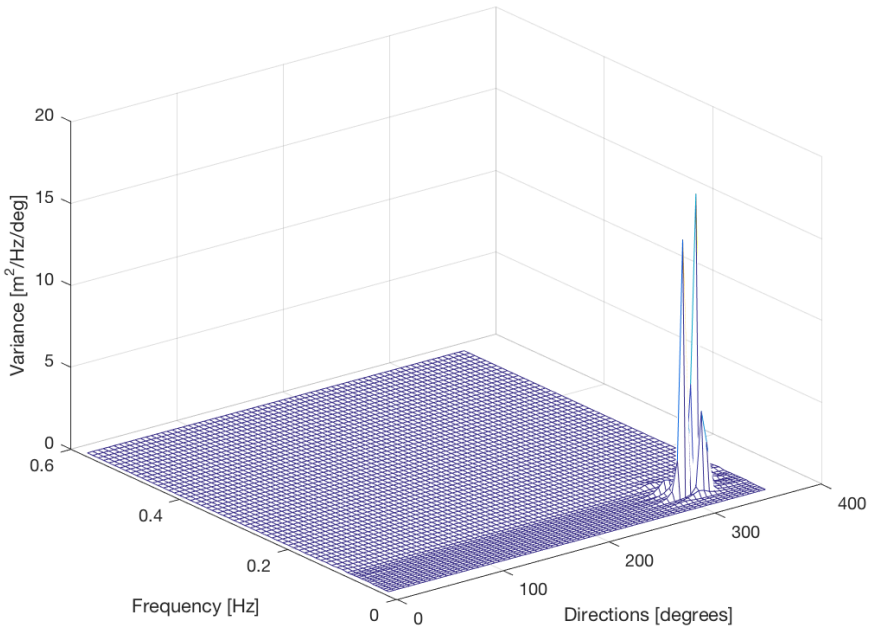


Figure 8.3: Original CDIP spectrum from swell 2.

8.3 Numerical Results

The two wave scenarios were computed for two different grid resolutions ($\Delta x = \Delta y = 7m$ and $\Delta x = \Delta y = 21m$) as shown in Table 8.2. Each instance was executed on a CPU and a GPU hardware configuration provided by the Coastal Resilience Group at the University of Hawaii:

- Intel Xeon Gold 6346 with 32 CPUs of 3.10GHz (divided into 2 sockets of 16 CPUs each)
- NVIDIA Tesla T4 TU104 with 2560 CUDA Cores of 1.25GHz

This allows for an analysis of the Present model's performance with respect to CPU and GPU parallelizations and also validates the consistency and replicability of the computed results.

8.3.1 Accuracy of the Computed Results

Figure 8.4 shows the power spectral density plots from the computed data in comparison with the observations at the Mokuleia array for swell 1. The solution from the Present model agrees reasonably well with the observed data over all frequency bands. The most offshore located gauge (8a in Figure 8.1) in about 12 m water depth shows the peak of wave energy at the peak period around 18 s. The two smaller peaks (super-harmonics) at higher frequencies are the result of the shoaling process of the incoming swell. As the waves move towards the shore, they undergo energetic breaking. The second gauge in about 6 m depth is located near the breaking point and indicates a decrease of the PSD in the gravity band and an increase in the IG band. The transfer of wave energy from gravity to IG bands is a typical process in the surf zone and the main driver of second-order phenomena such as surf beat and recirculation. Closer to shore in around 2 m depth, most of the gravity wave energy has vanished as the result of dissipation through breaking and bottom friction and transfer to the IG band. The energy level in the IG bands is over an order of magnitude higher than the energy in the gravity band. This is exactly the opposite at the gauge in 12 m depth. As expected, the Present model compares well to the solutions from BOSZ. Most visibly, the Present model shows less diffusion of short waves nearshore compared to BOSZ. Since the input swell spectrum is slightly larger than what is expected at 80 m depth, it is not surprising that the Present model predicts the remaining energy after breaking at a higher level than the observations.

The wave processes shown for swell 2 in Figure 8.5 are significantly different from swell 1. The overall PSD distribution at the three gauges inside Haleiwa harbor is relatively similar with pronounced IG energy levels over 5 min period. Though some gravity swells energy is visible inside the harbor, most of the wave energy is associated with IG waves controlled by the harbor's geometry and depth as well as the fringing reef outside the harbor.

The Present model correctly accounts for the general trend in wave energy distribution. As for swell 1, the results compare well to the solutions from BOSZ. Again, it should be noted that the differences between the two numerical models are solely due to the differences in the numerical solution as the governing equations, input waves, and boundary conditions are identical.

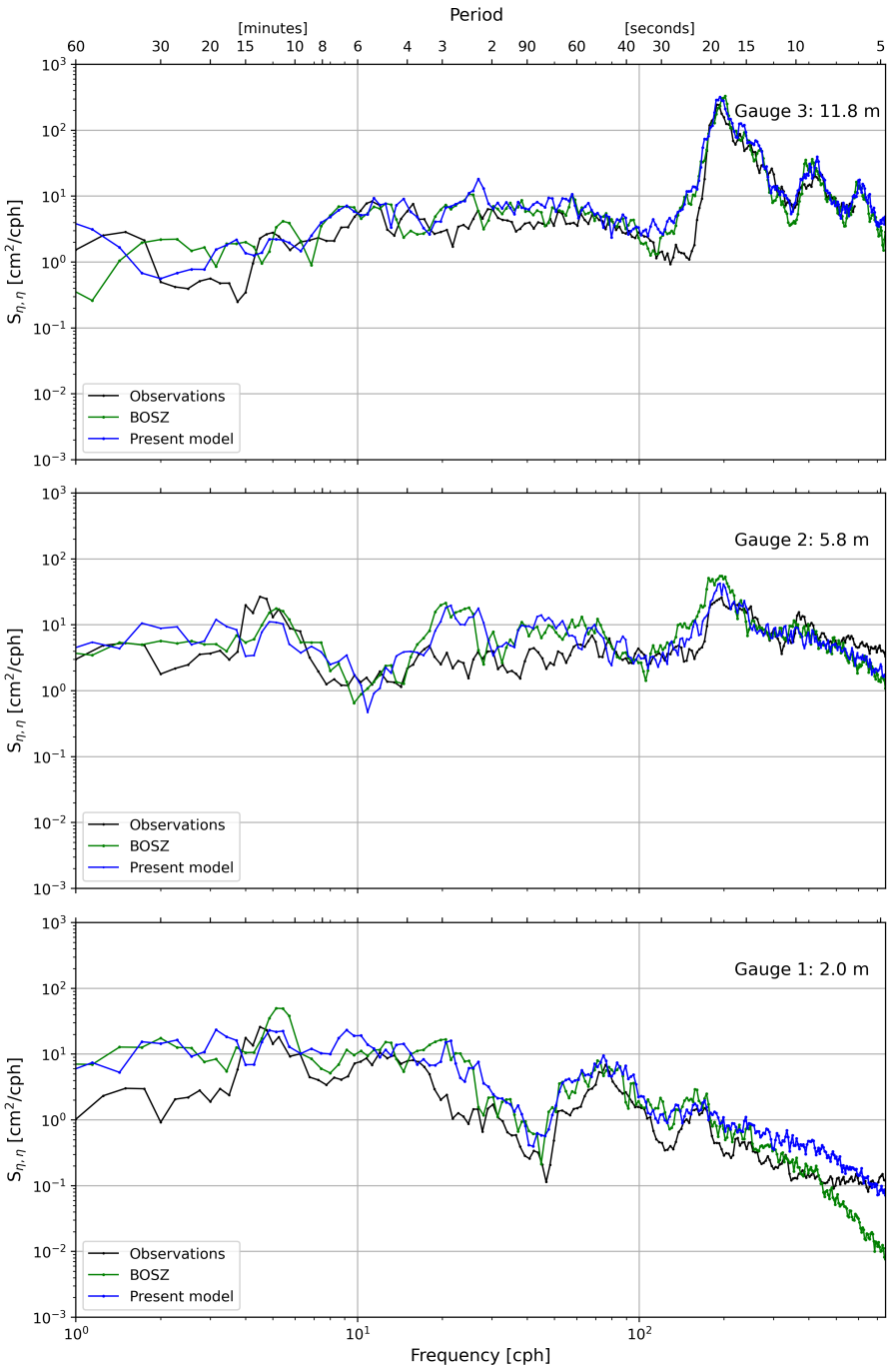


Figure 8.4: PSD of free surface elevation at Mokuleia array with a grid size of $\Delta x = \Delta y = 7\,\mathrm{m}$. Results from the Present model and from BOSZ for comparison.

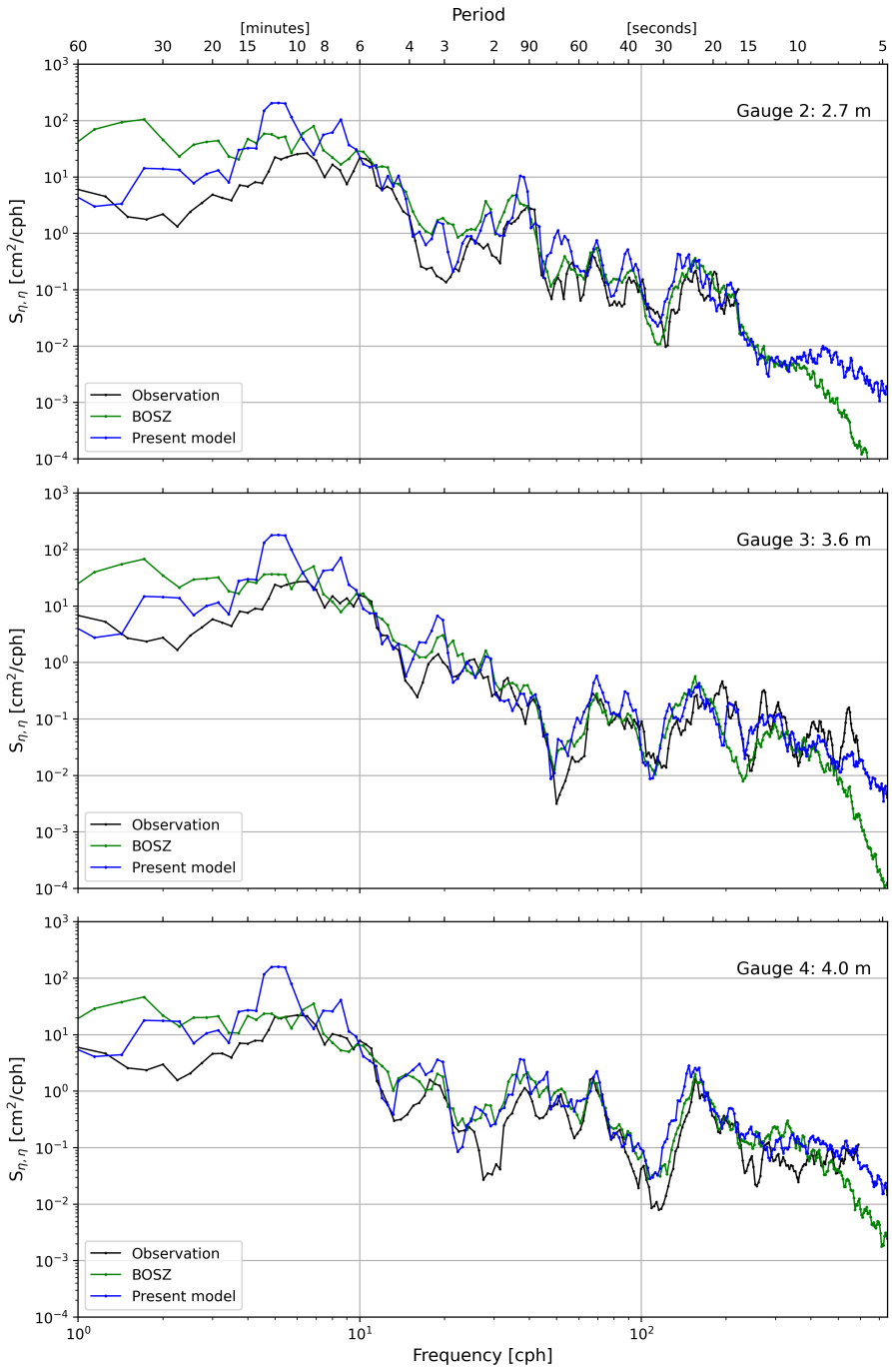


Figure 8.5: PSD of free surface elevation at Haleiwa Harbor with a grid size of $\Delta x = \Delta y = 7\,\mathrm{m}$. Results from the Present model and from BOSZ for comparison.

8.3.2 Grid Sensitivity

The two swell scenarios were computed over the same domain with two significantly different resolutions. The results from Figures 8.4 and 8.5 are based on a grid resolution of $\Delta x = \Delta y = 7\,\mathrm{m}$. This resolution is fine enough to resolve most waves in the system to sufficient detail and to account for the variations in the bathymetry. It is trivial that a coarser resolution can lead to less accurate results but on the flip side allows for fast computation due to the overall lower cell count and the accompanying increase in time stepping.

Here, we demonstrate the sensitivity of the numerical solutions to the grid resolution by using a uniform mesh size of $\Delta x = \Delta y = 21\,\mathrm{m}$, which is a grid size that an experienced user would probably not select. The objective is rather to analyze, which parts of the spectrum change the most due to the grid resolution and whether it is worth considering coarse resolution results for the sake of their low computational expenses. Figures 8.6 and 8.7 show the results at the same gauge locations as 8.4 and 8.5 for swell 1 and 2. Most noticeably, the short wave band around the super-harmonics suffers from the lower grid resolution. This is expected since the wavelength of the first super-harmonic at 12 m depth is already shorter than 100 m, which cannot be fully resolved with a 21 m grid size. The same applies to the short wave band at the two inner gauges inner the surf zone and to the results of swell 2 in Figure 8.5.

Nevertheless, the overall trends in PSD distribution are fairly similar to what was obtained with 7 m grid resolution. Especially the PSD of most IG waves is well comparable to the results from the higher resolution. Overall, the solutions from the Present model are much less sensitive to coarse meshes compared to the well-established BOSZ model. Especially inside Haleiwa harbor, which is resolved by under 100 grid cells, the coarse resolution results are still valid for a first estimate of the IG wave processes. Given the higher diffusivity of the BOSZ model, a higher grid resolution is required to obtain equivalently meaningful results. This underlines that the low diffusive numerical scheme plays a major role in the applicability of phase-resolving wave models.

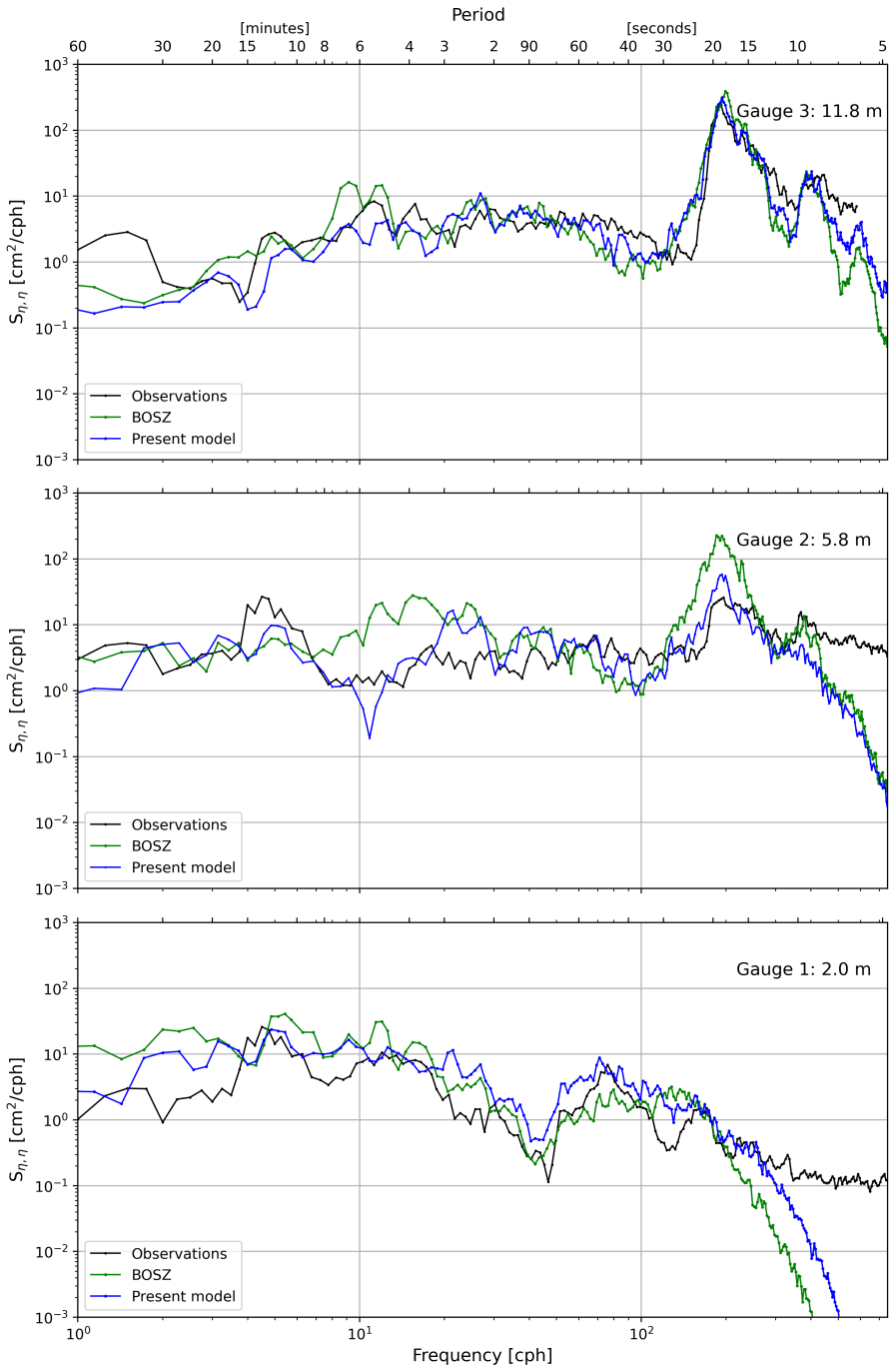


Figure 8.6: PSD of free surface elevation at Mokuleia array with a grid size of $\Delta x = \Delta y = 21\,\mathrm{m}$. Results from the Present model and from BOSZ for comparison.

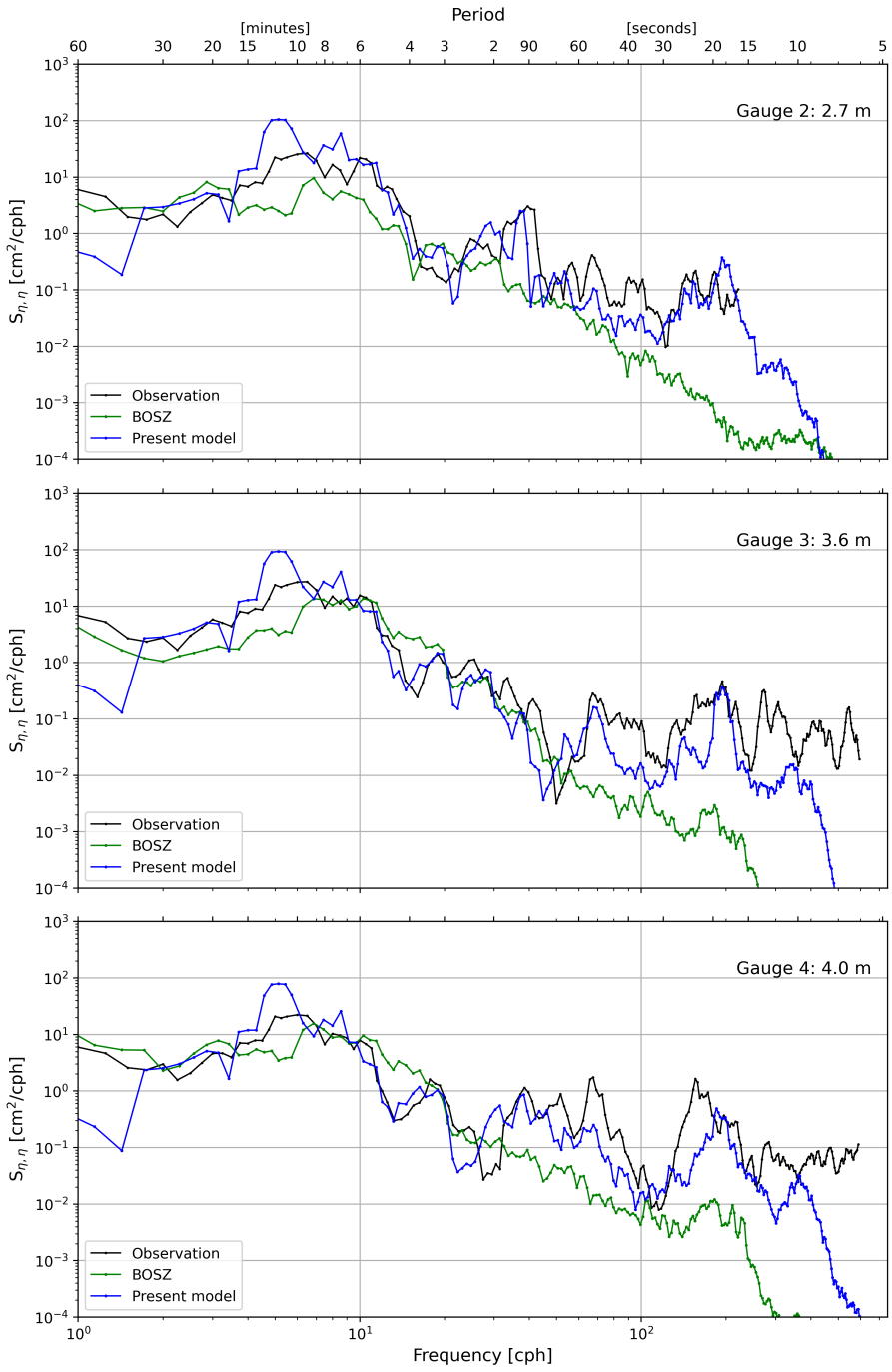


Figure 8.7: PSD of free surface elevation at Haleiwa Harbor with a grid size of $\Delta x = \Delta y = 21\,\mathrm{m}$. Results from the Present model and from BOSZ for comparison.

8.3.3 Model Performance

Finally, we evaluate the Present model's computational performance compared to the well-established BOSZ model. We computed the same wave scenario (*i.e.*, Run 1) with the two models using identical CPU hardware (*i.e.*, Intel Xeon 6346 3.1GHz). Table 8.3 shows the execution times necessary for the runs using different numbers of threads. The Present model takes 8 hours to compute the 4-hour scenario with a relatively low thread count (*i.e.*, 6 cores and 12 threads), which is the number found in most standard commodity hardware. BOSZ on the other hand takes 32 hrs for the same run. The difference in execution time between the two models is due to the lean structure of the hyperbolic solver as well as the low-order flux reconstruction (BOSZ uses a fifth-order reconstruction, whereas the Present model uses a second-order). Moreover, the optimized model implementation, as described in Chapter 6.3, adds more to the efficiency. With 24 threads, the computation time is almost real-time and about 7 times faster than the same computation in BOSZ. This proves that the CPU-parallelization speed-up scales well with the increase of computational resources in contrast to BOSZ, where the speed-up quickly reaches a plateau. On a full socket with 32 threads, the Present model accomplishes the 4-hour run in only 2 hours and 20 min, which is almost twice as fast as real-time. The use of the Present model on different sockets will require the implementation of a multi-card parallelization such as Message Passing Interface (MPI). However, the fact that it is possible to perform such extensive computations with 2.5 million cells and the generation of 40 thousand input waves on a single socket demonstrates the cost-effectiveness of the Present model.

Table 8.3: Present model performance compared to BOSZ on CPU hardware.

Model	Computed time	Hardware	Grid Res [m]	Cell count	Threads	Execution time
BOSZ	4 hrs	CPU	7	2,539,592	12	32 hrs
Present	4 hrs	CPU	7	2,539,592	12	8 hrs
BOSZ	4 hrs	CPU	7	2,539,592	24	27 hrs
Present	4 hrs	CPU	7	2,539,592	24	4 hrs 25 min
BOSZ	4 hrs	CPU	7	2,539,592	32	-
Present	4 hrs	CPU	7	2,539,592	32	2 hrs 16 min

Finally, we compare the execution times and costs of the two model implementations: CPU and GPU. It has been shown in Chapter 7.4, that the GPU implementation outperforms the computation on CPUs for comparable hardware in terms of computation time and price. This is mainly due to the GPU architecture, which caters to massive parallelism. The same trend is observed for this final real-case test, where the GPU computation is faster than the CPU computation on hardware which is four times cheaper. It is important to note that while the GPU instances require a CPU host to launch the kernels, there is no processing contribution from the CPU, and any low-end CPU can do this job without affecting the overall performance. That is the reason why in the section we compare mainly the cost and performance of the computing units of each implementation. Considering the energy consumption required to perform the model's processing operations, the GPU instance uses one-fourth of the total energy consumed by the CPU for the same model run. Here, we compare the energy dissipated by only the processing hardware. The energy used by the cooling system, which is difficult to quantify, is

not taken into account.

Table 8.4: Performance and cost: CPU vs. GPU

Model	Computed time	Hardware	Threads	Price	Execution time	Energy Used
Present	4 hrs	CPU	32	$9000	2 hrs 16 min	0.478 kWh
Present	4 hrs	GPU	-	$2300	1 hrs 48 min	0.126 kWh

8.4 Conclusions

The Present model was shown to be applicable to realistic wave scenarios as one would encounter in an operational setting. Especially for coarse grid sizes, the computed results are of superior quality than established solutions from BOSZ. This is mostly due to the implementation of the low-diffusion scheme, which is novel to Boussinesq-type models. The Present model is well parallelized and accomplishes stable solutions over a large irregular terrain for several hours in well under real-time without the need for excessive computation power.

This test clearly shows that the model is applicable in an operational setting both in terms of stability and accuracy and also with respect to its computational efficiency. The CUDA parallelization also helps to reduce the energy consumption to about 25% of the energy consumption of a CPU architecture necessary to accomplish a comparable turnaround time.

Chapter 9

Conclusions and Perspectives

Contents

9.1 Main Results **166**

9.2 Perspectives **168**

9.2.1 Implementation into a Forecast System 168

9.2.2 Mesh Refinement in Boussinesq Models 168

9.2.3 Wave Breaking Closure 168

9.2.4 Fully Dispersive Model 168

9.2.5 Further Extensions and Improvements 169

In the following, we summarize the main findings of this dissertation outlined in the previous chapters. In addition, we highlight the outstanding features of the newly developed Boussinesq model in comparison to existing wave models and specify applications, limitations, and perspectives of this work.

9.1 Main Results

This effort introduces a new phase-resolving model for operational computations of nearshore waves. The choice of the implemented features in this model has been motivated by comparative studies with the objective to find the right balance between the solution's accuracy and its associated numerical cost. The driving factor behind the rigorous development of this new numerical framework from scratch was the improvement of numerical efficiency without necessarily compromising the quality of the solution. The solver for Shallow Water Equations builds the backbone of most dispersive depth-integrated models of phase-resolving nature. Here, the choice of a conservative staggered scheme was shown to be a fundamental component of critical importance. The lean and efficient structure of the scheme, along with its robustness in the context of flow discontinuities and wet-dry transitions, have proven to be crucial properties for achieving stable and fast computations of nearshore waves. Moreover, the reduced diffusivity of the scheme compared to traditional FV methods based on characteristic decomposition leads not only to improved accuracy but also to improved efficiency since the model is able to compute important wave processes at a lower grid count. Consequently, this numerical foundation is catering to the model's extension with local mesh refinement as well as frequency dispersion. The quality of low diffusivity is critical for problems dealing with periodic waves that are solved with dispersive governing equations such as the ones by Nwogu and SGN. This property mainly ensures that the Present model is able to conserve the wave energy for propagation over long distances without the need for excessively fine grid spacing. The low diffusivity further supports the use of the nested grid method, which was proven to be very efficient in computing high-resolution local run-up at a low numerical cost. The overall wave transformation can be accurately and efficiently accounted for over a coarse grid, while the local wave run-up limit, which is mainly controlled by the details of the topography, can be described with a highly localized mesh refinement.

While the model's accuracy and efficiency greatly depend on the underlying numerical scheme, the choice of the mathematical framework can also impact its applicability, especially in the case of Boussinesq-type models. Many mathematical concepts contain strong assumptions that affect two main properties: nonlinearity and dispersion. In this thesis, a comparative study has been carried out to examine the impact of these assumptions on the final results. On one hand, the strong dispersion errors in SGN equations not only lead to inaccurate results but also to instabilities due to the negative trend of the error. On the other hand, the weak nonlinearity of equations like the one by Nwogu affects the accuracy of the shoaling process leading to an overestimation of the crest and trough amplitudes. Opposite tendencies were observed with fully nonlinear Boussinesq equations, where undershoaling originates from insufficient frequency dispersion. From an efficiency point of view and considering the complexity of the fully nonlinear equations, the weakly nonlinear Nwogu equations were selected as the base equation of the model.

The applicability and robustness of the Present model to real cases require an efficient and reliable

wave-breaking closure. Since the depth-integrated nature of the equations does not allow for the overturning of the free surface, the wave-breaking process is not fully computed as part of the main solution, but instead approximated through flow discontinuities. The formation of shocks is posing a conflict for non-hyperbolic systems and the consequence can be instabilities at the wavefront. This problem becomes mainly apparent with fine meshes where the wave front is described by multiple cells. It is therefore necessary to counter-measure a potential source of instabilities to ensure the applicability of the Present model over a wide range of grid sizes. Here, a TKE-based eddy viscosity method was implemented to account for the dissipative effects of wave breaking. The approach generally provides solutions of superior quality and shows better grid convergence for various topographies and wave conditions compared to standard methods such as the commonly applied local deactivation of dispersion.

This thesis also highlights the major impacts of simple yet efficient optimization techniques to support the computational speed of the model. Since this type of model is intended to be used for long computations over several hours and over large computational domains with millions of cells, each operation inside the main time loop impacts the overall performance, and thus multiple small optimization techniques lead to a substantial boost in performance. Moreover, the explicit nature of the solver and the massive stencil computations cater to data parallelism. The GPU hardware, in particular, provides an ideal environment to deal with such data-intensive computations. The GPU implementation of the model was proven to be not only cost-effective but also more efficient for large computations than standard CPU implementations.

The new model has been rigorously validated using a variety of standard benchmark tests. These tests involve a wide range of wave transformation processes, such as wave-breaking, run-up, refraction-diffraction, and shoaling. The overall agreement between the computed results and laboratory data demonstrates the capability of the model to accurately compute the fundamental nearshore wave processes. The model results have also been compared to field data collected on the Northshore of Oahu, Hawai'i. This test demonstrates the Present model's applicability to real-world cases involving energetic waves over complex terrain.

The main objective of this research effort was the development of a new phase-resolving model for operational nearshore wave assessment. Since it was shown that the Present model computes large domains much faster than existing models with no compromise on accuracy and applicability, it is reasonable to state that the goals were reached. The uniqueness of this development stems from a combination of pragmatic choices with optimized implementations. Since the model development started from scratch, it was possible to address several shortcomings of numerical approaches commonly used in models for nearshore waves. It also became evident that some of these techniques such as the use of characteristic decomposition methods for the underlying SWE should be reconsidered in the context of Boussinesq modeling.

9.2 Perspectives

9.2.1 Implementation into a Forecast System

The model was designed to be used in an operational framework, particularly for wave run-up forecasts. The implementation of many features in the model has been motivated by collaborative work with groups specialized in this type of forecast. While this work has helped overcome one of the main commonly encountered limitations - computational speed - the implementation of the Present model in an operational setting still requires additional validation tests, namely for run-up, with field data. Moreover, since these forecasts require input from tide and spectral wave models for adequate boundary conditions, an efficient and automated coupling of these models needs to be implemented.

9.2.2 Mesh Refinement in Boussinesq Models

The implementation of the nested grid technique into the hydrostatic solver was proven to be an efficient tool for high-resolution run-up computations. The same approach can be extended and applied to the final dispersive model with only minor changes to the main structure. While the feedback step in the two-way nesting is identical to the Shallow Water model, the input of the boundary conditions for the inner grid must be modified to include the information from the dispersive terms. A Dirichlet-type boundary condition must be imposed for the elliptic problem to enable wave dispersion across grid interfaces. The implementation of this technique will have to be later verified by analytical tests such as solitary wave propagation and further tested in the context of practical problems. It is expected that particular wave problems require specific refinement factors that can vary from those used for the nesting of the Shallow Water solver. Nevertheless, grid nesting will open another door for further reduction in computational times by maintaining locally high accuracy and quality of the results.

9.2.3 Wave Breaking Closure

It was demonstrated that the TKE-based eddy viscosity approach offers a more stable and reliable wave-breaking closure compared to standard methods. However, this method still has limitations, especially in terms of the mixing length approximation. In the Present model, the mixing length is taken to be equal to the local water depth, which is consistent with previously published models. It is quite likely that a more complete solution, such as the $k - \varepsilon$ turbulence closure, can lead to even better results, since the mixing length is directly computed by an additional equation instead of taken as a local constant value.

9.2.4 Fully Dispersive Model

There is no doubt that a higher level of frequency dispersion can help to further extend the Present model's applicability. Most obviously, the model can be applied to deeper water offshore, if the limiting ratio of λ to h was smaller. A better approximation of frequency dispersion can further help to improve the accuracy of the wave field subject to shoaling and refraction, since nonlinearity is always balanced by dispersion. Karambas and Memos [2009] has presented a

promising technique for high-order linear dispersion that could be implemented in modular form into the Present model.

9.2.5 Further Extensions and Improvements

The computed wave field from the Present model can be used as the driver for other processes such as sediment and pollutant transport. This can be achieved by solving an additional transport equation in the model (Watanabe et al. [2020]). The base model can also be extended to include other wave generation mechanisms (*e.g.*, tsunami Okada [1985], ship waves David et al. [2017]).

References

Antunes Do Carmo, J. S. (2013). Boussinesq and Serre type models with improved linear dispersion characteristics: Applications. *Journal of hydraulic research*, 51(6):719–727. (Cited on page 197.)

Arakawa, A. and Lamb, V. R. (1981). A potential enstrophy and energy conserving scheme for the Shallow Water Equations. *Monthly Weather Review*, 109(1):18–36. (Cited on page 19.)

Arce Acuña, M. and Aoki, T. (2018). Tree-based mesh-refinement GPU-accelerated tsunami simulator for real-time operation. *Natural Hazards and Earth System Sciences*, 18(9):2561–2602. (Cited on page 140.)

Audusse, E., Berthon, C., Chalons, C., Delestre, O., Goutal, N., Jodeau, M., Sainte-Marie, J., Giesselmann, J., and Sadaka, G. (2012). Sediment transport modelling: Relaxation schemes for Saint-Venant–Exner and three layer models. In *ESAIM: proceedings*, volume 38, pages 78–98. EDP Sciences. (Cited on page 24.)

Audusse, E., Chalons, C., and Ung, P. (2015). A simple well-balanced and positive numerical scheme for the shallow-water system. *Communications in Mathematical Sciences*, 13(5):1317–1332. (Cited on page 86.)

Azouri, A. (2016). *Observations, Forecast, and Modeling of 0.5-200 Min Infragravity Oscillations in Hale'iwa Harbor Region, Hawai'i*. PhD thesis, [Honolulu]:[University of Hawaii at Manoa],[December 2016]. (Cited on pages 152 and 153.)

Barranco, I. and Liu, P. L.-F. (2021). Run-up and inundation generated by non-decaying dam-break bores on a planar beach. *Journal of Fluid Mechanics*, 915:A81. (Cited on page 124.)

Barthelemy, X., Banner, M., Peirson, W., Fedele, F., Allis, M., and Dias, F. (2018). On a unified breaking onset threshold for gravity waves in deep and intermediate depth water. *Journal of Fluid Mechanics*, 841:463–488. (Cited on page 96.)

Beji, S. and Battjes, J. (1993). Experimental investigation of wave propagation over a bar. *Coastal engineering*, 19(1-2):151–162. (Cited on pages 5, 8, 77, 79 and 203.)

Berkhoff, J., Booy, N., and Radder, A. C. (1982). Verification of numerical wave propagation models for simple harmonic linear water waves. *Coastal Engineering*, 6(3):255–279. (Cited on pages 7, 136 and 137.)

Blenkinsopp, C., Bayle, P., Conley, D., Masselink, G., Gulson, E., Kelly, I., Almar, R., Turner, I., Baddock, T., Beuzen, T., et al. (2019). Dynamic coastal protection: Resilience of dynamic revetments (DYNAREV). In *Hydralab+ joint user meeting*, pages 129–139. (Cited on page 99.)

Booij, N., Ris, R. C., and Holthuijsen, L. H. (1999). A third-generation wave model for coastal regions: 1. Model description and validation. *Journal of geophysical research: Oceans*, 104(C4):7649–7666. (Cited on page 16.)

Bretschneider, C., Krock, H., Nakazaki, E., and Casciano, F. (1986). Roughness of Typical Hawaiian Terrain for Tsunami Run-up Calculations: A User's Manual. *University of Hawai'i, Honolulu*. (Cited on page 153.)

Brodtkorb, A. R., Hagen, T. R., and Sætra, M. L. (2013). Graphics Processing Unit (GPU) programming strategies and trends in GPU computing. *Journal of Parallel and Distributed Computing*, 73(1):4–13. (Cited on page 140.)

Brodtkorb, A. R., Sætra, M. L., and Altinakar, M. (2012). Efficient shallow water simulations on GPUs: Implementation, visualization, verification, and validation. *Computers & Fluids*, 55:1–12. (Cited on pages 59 and 140.)

Caldwell, P. C. and Aucan, J. P. (2007). An empirical method for estimating surf heights from deepwater significant wave heights and peak periods in coastal zones with narrow shelves, steep bottom slopes, and high refraction. *Journal of Coastal Research*, 23(5):1237–1244. (Cited on page 16.)

Chazel, F., Benoit, M., Ern, A., and Piperno, S. (2009). A double-layer Boussinesq-type model for highly nonlinear and dispersive waves. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 465(2108):2319–2346. (Cited on page 18.)

Chen, Q., Madsen, P. A., and Basco, D. R. (1999). Current effects on nonlinear interactions of shallow-water waves. *Journal of waterway, port, coastal, and ocean engineering*, 125(4):176–186. (Cited on page 117.)

Chertock, A., Cui, S., Kurganov, A., and Wu, T. (2015). Well-balanced positivity preserving central-upwind scheme for the shallow water system with friction terms. *International Journal for numerical methods in fluids*, 78(6):355–383. (Cited on page 86.)

Choi, Y.-K., Shi, F., Malej, M., and Smith, J. M. (2018). Performance of various shock-capturing-type reconstruction schemes in the Boussinesq wave model, FUNWAVE-TVD. *Ocean Modelling*, 131:86–100. (Cited on pages 19 and 83.)

Choi, Y.-K., Shi, F., Malej, M., Smith, J. M., Kirby, J. T., and Grilli, S. T. (2022). Block-structured, equal-workload, multi-grid-nesting interface for the Boussinesq wave model FUNWAVE-TVD (Total Variation Diminishing). *Geoscientific Model Development*, 15(14):5441–5459. (Cited on page 19.)

Clamond, D., Dutykh, D., and Mitsotakis, D. (2017). Conservative modified Serre–Green–Naghdi equations with improved dispersion characteristics. *Communications in Nonlinear Science and Numerical Simulation*, 45:245–257. (Cited on page 197.)

Dalrymple, R. A., Kirby, J. T., and Hwang, P. A. (1984). Wave diffraction due to areas of energy dissipation. *Journal of waterway, port, coastal, and ocean engineering*, 110(1):67–79. (Cited on page 80.)

David, C. G., Roeber, V., Goseberg, N., and Schlurmann, T. (2017). Generation and propagation of ship-borne waves-Solutions from a Boussinesq-type model. *Coastal Engineering*, 127:170–187. (Cited on page 169.)

de la Asunción, M., Castro, M. J., Fernández-Nieto, E. D., Mantas, J. M., Acosta, S. O., and González-Vida, J. M. (2013). Efficient GPU implementation of a two waves TVD-WAF method for the two-dimensional one layer shallow water system on structured meshes. *Computers & Fluids*, 80:441–452. (Cited on page 140.)

Delis, A. I. and Katsaounis, T. (2005). Numerical solution of the two-dimensional Shallow Water Equations by the application of relaxation methods. *Applied Mathematical Modelling*, 29(8):754–783. (Cited on page 122.)

Dias, F. and Milewski, P. (2010). On the fully-nonlinear shallow-water generalized Serre equations. *Physics Letters A*, 374(8):1049–1053. (Cited on page 197.)

Dingemans, M. (1973). Water waves over an uneven bottom; a discussion of long-wave equations. *Delft Hydraulics Report R729 part*, 2. (Cited on page 61.)

Dommermuth, D. G. and Yue, D. K. (1987). A high-order spectral method for the study of nonlinear gravity waves. *Journal of Fluid Mechanics*, 184:267–288. (Cited on page 18.)

Duran, A., Dutykh, D., and Mitsotakis, D. (2013). On the Galilean invariance of some nonlinear dispersive wave equations. *Studies in Applied Mathematics*, 131(4):359–388. (Cited on page 66.)

Emanuel, K. (2005). Increasing destructiveness of tropical cyclones over the past 30 years. *Nature*, 436(7051):686–688. (Cited on page 15.)

Erduran, K., Ilic, S., and Kutija, V. (2005). Hybrid finite-volume finite-difference scheme for the solution of Boussinesq equations. *International Journal for Numerical Methods in Fluids*, 49(11):1213–1232. (Cited on pages 19 and 83.)

Fiedler, J. W., Young, A. P., Ludka, B. C., O'Reilly, W. C., Henderson, C., Merrifield, M. A., and Guza, R. (2020). Predicting site-specific storm wave run-up. *Natural Hazards*, 104(1):493–517. (Cited on page 20.)

Fog, A. et al. (2011). Instruction tables: Lists of instruction latencies, throughputs and micro-operation breakdowns for Intel, AMD and VIA CPUs. *Copenhagen University College of Engineering*, 93:110. (Cited on page 121.)

Fornberg, B. (1998). *A practical guide to pseudospectral methods*. Number 1. Cambridge university press. (Cited on page 119.)

Fraccarollo, L. and Toro, E. F. (1995). Experimental and numerical assessment of the shallow water model for two-dimensional dam-break type problems. *Journal of hydraulic research*, 33(6):843–864. (Cited on page 85.)

Gobbi, M. F., Kirby, J. T., and Wei, G. (2000). A fully nonlinear Boussinesq model for surface waves. Part 2. Extension to O (kh) 4. *Journal of Fluid Mechanics*, 405:181–210. (Cited on pages 17 and 18.)

Green, A. E. and Naghdi, P. M. (1976). A derivation of equations for wave propagation in water of variable depth. *Journal of Fluid Mechanics*, 78(2):237–246. (Cited on page 21.)

Grilli, S., Subramanya, R., Svendsen, I., and Veeramony, J. (1994). Shoaling of solitary waves on plane beaches. *Journal of Waterway, Port, Coastal, and Ocean Engineering*, 120(6):609–628. (Cited on pages 5, 79, 80 and 81.)

Guide, D. (2013). Cuda c programming guide. *NVIDIA, July*. (Cited on pages 7 and 141.)

Hagen, T. R., Henriksen, M. O., Hjelmervik, J. M., and Lie, K.-A. (2007). How to solve systems of conservation laws numerically using the graphics processor as a high-performance computational engine. In *Geometric Modelling, Numerical Simulation, and Optimization*, pages 211–264. Springer. (Cited on page 140.)

Hall, J. V., Watts, G. M., et al. (1953). Laboratory investigation of the vertical rise of solitary waves on impermeable slopes. *Army Coastal Engineering Research Center Washington DC*. (Cited on pages 6, 125 and 128.)

Hamdi, S., Enright, W., Ouellet, Y., and Schiesser, W. (2005). Method of lines solutions of the extended Boussinesq equations. *Journal of Computational and Applied Mathematics*, 183(2):327–342. (Cited on page 71.)

Harris, M. et al. (2007). Optimizing parallel reduction in CUDA. *Nvidia developer technology*, 2(4):70. (Cited on pages 7, 143 and 145.)

Harten, A., Lax, P. D., and Leer, B. v. (1983). On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM review*, 25(1):35–61. (Cited on page 85.)

Hasselmann, K. F., Barnett, T. P., Bouws, E., Carlson, H., Cartwright, D. E., Eake, K., Euring, J., Gicnapp, A., Hasselmann, D., Kruseman, P., et al. (1973). Measurements of wind-wave growth and swell decay during the Joint North Sea Wave Project (JONSWAP). *Ergaenzungsheft zur Deutschen Hydrographischen Zeitschrift, Reihe A*. (Cited on page 211.)

Hervouet, J.-M. (2003). Hydrodynamique des écoulements à surface libre. (Cited on page 57.)

Hervouet, J.-M. and Petitjean, A. (1999). Malpasset dam-break revisited with two-dimensional computations. *Journal of hydraulic research*, 37(6):777–788. (Cited on page 59.)

Hughes, S. A. (1984). The TMA shallow-water spectrum description and applications. *Coastal Engineering Research Center (US)*. (Cited on page 212.)

Imamura, F. (1989). Tsunami Numerical Simulation with the Staggered Leap-frog Scheme (Numerical code of TUNAMI-N1), School of Civil Engineering, Asian Inst. *Tech. and Disaster Control Research Center, Tohoku University*. (Cited on page 17.)

Janßen, C. F., Grilli, S. T., Krafczyk, M., et al. (2012). Efficient simulations of long wave propagation and runup using a LBM approach on GPGPU hardware. In *The Twenty-second International Offshore and Polar Engineering Conference*. International Society of Offshore and Polar Engineers. (Cited on page 140.)

Jiang, G.-S. and Shu, C.-W. (1996). Efficient implementation of weighted ENO schemes. *Journal of computational physics*, 126(1):202–228. (Cited on pages 19 and 83.)

Karambas, T. V. and Memos, C. D. (2009). Boussinesq model for weakly nonlinear fully dispersive water waves. *Journal of waterway, port, coastal, and ocean engineering*, 135(5):187–199. (Cited on pages 17, 136 and 168.)

Kazolea, M. and Delis, A. (2013). A well-balanced shock-capturing hybrid finite volume–finite difference numerical scheme for extended 1D Boussinesq models. *Applied Numerical Mathematics*, 67:167–186. (Cited on pages 17, 19, 72, 83 and 95.)

Kazolea, M. and Ricchiuto, M. (2018). On wave breaking for Boussinesq-type models. *Ocean Modelling*, 123:16–39. (Cited on pages 98 and 99.)

Kennedy, A. B., Chen, Q., Kirby, J. T., and Dalrymple, R. A. (2000). Boussinesq modeling of wave transformation, breaking, and runup. I: 1D. *Journal of waterway, port, coastal, and ocean engineering*, 126(1):39–47. (Cited on pages 95, 98 and 104.)

Kennedy, A. B., Kirby, J. T., Chen, Q., and Dalrymple, R. A. (2001). Boussinesq-type equations with improved nonlinear performance. *Wave Motion*, 33(3):225–243. (Cited on page 189.)

Khakimzyanov, G., Dutykh, D., Fedotova, Z., and Gusev, O. (2020). Model Derivation on a Globally Flat Space. In *Dispersive Shallow Water Waves*, pages 1–43. Springer. (Cited on pages 182 and 188.)

Khakimzyanov, G., Dutykh, D., Gusev, O., and Shokina, N. (2017). Dispersive shallow water wave modelling. Part II: Numerical simulation on a globally flat space. *Communications in Computational Physics*. (Cited on pages 62, 63 and 209.)

Kim, B., Oh, C., Yi, Y., and Kim, D.-H. (2018). GPU-accelerated Boussinesq model using compute unified device architecture FORTRAN. *Journal of Coastal Research*, (85 (10085)):1176–1180. (Cited on page 140.)

Kim, D.-H., Lynett, P. J., and Socolofsky, S. A. (2009). A depth-integrated model for weakly dispersive, turbulent, and rotational fluid flows. *Ocean Modelling*, 27(3-4):198–214. (Cited on page 17.)

Kim, K. H. and Kim, C. (2005). Accurate, efficient and monotonic numerical methods for multi-dimensional compressible flows: Part II: Multi-dimensional limiting process. *Journal of computational physics*, 208(2):570–615. (Cited on pages 19, 83, 84 and 86.)

Kirby, J. T. (2016). Boussinesq models and their application to coastal processes across a wide range of scales. *Journal of Waterway, Port, Coastal, and Ocean Engineering*, 142(6):03116005. (Cited on page 83.)

Klonaris, G. T., Memos, C., and Karambas, T. V. (2013). A Boussinesq-type model including wave-breaking terms in both continuity and momentum equations. *Ocean engineering*, 57:128–140. (Cited on page 17.)

Kolmogorov, A. N. (1941). Equations of turbulent motion in an incompressible fluid. In *Dokl. Akad. Nauk SSSR*, volume 30, pages 299–303. (Cited on page 98.)

Komar, P. D. and Gaughan, M. K. (1973). Airy wave theory and breaker height prediction. In *Coastal engineering 1972*, pages 405–418. (Cited on page 16.)

Kurganov, A., Noelle, S., and Petrova, G. (2001). Semidiscrete central-upwind schemes for hyperbolic conservation laws and Hamilton–Jacobi equations. *SIAM Journal on Scientific Computing*, 23(3):707–740. (Cited on pages 19 and 84.)

Kurganov, A. and Petrova, G. (2007). A second-order well-balanced positivity preserving central-upwind scheme for the Saint-Venant system. *Communications in Mathematical Sciences*, 5(1):133–160. (Cited on page 85.)

Lacasta, A., Morales-Hernández, M., Murillo, J., and García-Navarro, P. (2014). An optimized GPU implementation of a 2D free surface simulation model on unstructured meshes. *Advances in engineering software*, 78:1–15. (Cited on page 140.)

Larsen, J. and Dancy, H. (1983). Open boundaries in short wave simulations—a new approach. *Coastal engineering*, 7(3):285–297. (Cited on page 117.)

Li, Y. and Raichlen, F. (2002). Non-breaking and breaking solitary wave run-up. *Journal of Fluid Mechanics*, 456:295–318. (Cited on pages 6 and 128.)

Liang, Q. and Marche, F. (2009). Numerical resolution of well-balanced Shallow Water Equations with complex source terms. *Advances in water resources*, 32(6):873–884. (Cited on page 86.)

Liu, Z. and Fang, K. (2016). A new two-layer Boussinesq model for coastal waves from deep to shallow water: Derivation and analysis. *Wave Motion*, 67:1–14. (Cited on page 18.)

Liu, Z., Fang, K., and Cheng, Y. (2018). A new multi-layer irrotational Boussinesq-type model for highly nonlinear and dispersive surface waves over a mildly sloping seabed. *Journal of Fluid Mechanics*, 842:323–353. (Cited on page 18.)

Lynett, P., Liu, P., Sitanggang, K., and Kim, D. (2002). Modeling wave generation, evolution, and interaction with depthintegrated, dispersive wave equations COULWAVE code manual. *Cornell University Long and Intermediate Wave Modeling Package*. (Cited on page 19.)

Lynett, P. and Liu, P. L.-F. (2004). A two-layer approach to wave modelling. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 460(2049):2637–2669. (Cited on page 18.)

Madsen, P. and Schäffer, H. (1998). Higher–order Boussinesq–type equations for surface gravity waves: derivation and analysis. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 356(1749):3123–3181. (Cited on pages 18 and 61.)

Madsen, P. A., Bingham, H., and Liu, H. (2002). A new Boussinesq method for fully nonlinear waves from shallow to deep water. *Journal of Fluid Mechanics*, 462:1–30. (Cited on page 17.)

Madsen, P. A., Bingham, H., and Schäffer, H. (2003). Boussinesq-type formulations for fully nonlinear and extremely dispersive water waves: derivation and analysis. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 459(2033):1075–1104. (Cited on page 17.)

Madsen, P. A. and Fuhrman, D. R. (2020). Trough instabilities in Boussinesq formulations for water waves. *Journal of Fluid Mechanics*, 889:A38. (Cited on page 18.)

Madsen, P. A., Fuhrman, D. R., and Wang, B. (2006). A Boussinesq-type method for fully nonlinear waves interacting with a rapidly varying bathymetry. *Coastal engineering*, 53(5-6):487–504. (Cited on page 17.)

Madsen, P. A., Murray, R., and Sørensen, O. R. (1991). A new form of the Boussinesq equations with improved linear dispersion characteristics. *Coastal engineering*, 15(4):371–388. (Cited on page 197.)

Madsen, P. A. and Sørensen, O. R. (1992). A new form of the Boussinesq equations with improved linear dispersion characteristics. Part 2. A slowly-varying bathymetry. *Coastal engineering*, 18(3-4):183–204. (Cited on pages 17 and 61.)

Mihami, F.-Z., Roeber, V., and Morichon, D. (2022). Efficient numerical computations of long-wave run-up and their sensitivity to grid nesting. *Water Waves*, pages 1–32. (Cited on page 24.)

Nwogu, O. (1993). Alternative form of Boussinesq equations for nearshore wave propagation. *Journal of waterway, port, coastal, and ocean engineering*, 119(6):618–638. (Cited on pages 17, 18, 21, 61, 68, 74, 106, 107, 114, 189 and 191.)

Nwogu, O. G. (1996). Numerical prediction of breaking waves and currents with a Boussinesq model. *Coastal Engineering Proceedings*, 1(25). (Cited on page 98.)

Nwogu, O. G. and Demirbilek, Z. (2001). BOUSS-2D: A Boussinesq wave model for coastal regions and harbors. Technical report, Engineer Research and Development Center Vicksburg MS Coastal and Hydraulics Lab. (Cited on pages 19 and 83.)

Okada, Y. (1985). Surface deformation due to shear and tensile faults in a half-space. *Bulletin of the seismological society of America*, 75(4):1135–1154. (Cited on page 169.)

Owens, J. D., Luebke, D., Govindaraju, N., Harris, M., Krüger, J., Lefohn, A. E., and Purcell, T. J. (2007). A survey of general-purpose computation on graphics hardware. In *Computer graphics forum*, pages 80–113. Wiley Online Library. (Cited on page 140.)

PacIOOS (2022). Wave Run-Up Forecast : West Maui. `http://www.pacioos.hawaii.edu/shoreline-category/runup-westmaui/`. (Cited on page 20.)

Panda, N., Dawson, C., Zhang, Y., Kennedy, A. B., Westerink, J. J., and Donahue, A. S. (2014). Discontinuous Galerkin methods for solving Boussinesq–Green–Naghdi equations in resolving non-linear and dispersive surface water waves. *Journal of Computational Physics*, 273:572–588. (Cited on page 19.)

Peregrine, D. H. (1967). Long waves on a beach. *Journal of fluid mechanics*, 27(4):815–827. (Cited on pages 17, 61, 74 and 81.)

Pierson Jr, W. J. and Moskowitz, L. (1964). A proposed spectral form for fully developed wind seas based on the similarity theory of SA Kitaigorodskii. *Journal of geophysical research*, 69(24):5181–5190. (Cited on page 211.)

Pope, S. B. (2000). *Turbulent Flows*, volume 1. Cambridge University Press, Cambridge. (Cited on page 98.)

Pörtner, H.-O., Roberts, D. C., Adams, H., Adler, C., Aldunce, P., Ali, E., Begum, R. A., Betts, R., Kerr, R. B., Biesbroek, R., et al. (2022). Climate change 2022: Impacts, adaptation and vulnerability. *IPCC Sixth Assessment Report*. (Cited on page 15.)

Prandtl, L. et al. (1945). Uber ein neues Formelsystem fur die ausgebildete Turbulenz. *Nacr. Akad. Wiss. Gottingen, Math-Phys. Kl*, pages 6–19. (Cited on page 98.)

Raoult, C., Benoit, M., and Yates, M. L. (2016). Validation of a fully nonlinear and dispersive wave model with laboratory non-breaking experiments. *Coastal Engineering*, 114:194–207. (Cited on page 18.)

Ricchiuto, M. and Filippini, A. G. (2014). Upwind residual discretization of enhanced Boussinesq equations for wave propagation over complex bathymetries. *Journal of Computational Physics*, 271:306–341. (Cited on pages 19 and 136.)

Roe, P. L. (1986). Characteristic-based schemes for the Euler equations. *Annual review of fluid mechanics*, 18(1):337–365. (Cited on page 19.)

Roeber, V. (2010). *Boussinesq-type model for nearshore wave processes in fringing reef environment*. PhD thesis, [Honolulu]:[University of Hawaii at Manoa],[December 2010]. (Cited on pages 95 and 128.)

Roeber, V. and Bricker, J. D. (2015). Destructive tsunami-like wave generated by surf beat over a coral reef during Typhoon Haiyan. *Nature Communications*, 6. (Cited on page 83.)

Roeber, V. and Cheung, K. F. (2012). Boussinesq-type model for energetic breaking waves in fringing reef environments. *Coastal Engineering*, 70:1–20. (Cited on pages 17, 24, 83, 95 and 113.)

Roeber, V., Cheung, K. F., and Kobayashi, M. H. (2010). Shock-capturing Boussinesq-type model for nearshore wave processes. *Coastal Engineering*, 57(4):407–423. (Cited on pages 19, 83 and 128.)

Russell, J. S. (1845). *Report on Waves: Made to the Meetings of the British Association in 1842-43*. (Cited on page 96.)

Rusu, E. and Soares, C. G. (2013). Modeling waves in open coastal areas and harbors with phase-resolving and phase-averaged models. *Journal of Coastal Research*, 29(6):1309–1325. (Cited on page 16.)

Sætra, M. L. (2014). Shallow Water Simulations on Graphics Hardware. *thesis*. (Cited on page 59.)

Sætra, M. L., Brodtkorb, A. R., and Lie, K.-A. (2015). Efficient GPU-implementation of adaptive mesh refinement for the shallow-water equations. *Journal of Scientific Computing*, 63(1):23–48. (Cited on pages 140 and 141.)

Schäffer, H. A. (2005). Another step towards a post-Boussinesq wave model. In *Coastal Engineering 2004: (In 4 Volumes)*, pages 132–144. World Scientific. (Cited on page 17.)

Serre, F. (1953). Contribution à l'étude des écoulements permanents et variables dans les canaux. *La Houille Blanche*, (6):830–872. (Cited on pages 21 and 61.)

Shi, F., Kirby, J. T., Harris, J. C., Geiman, J. D., and Grilli, S. T. (2012). A high-order adaptive time-stepping TVD solver for Boussinesq modeling of breaking waves and coastal inundation. *Ocean Modelling*, 43:36–51. (Cited on pages 83, 96 and 113.)

Shi, F., Kirby, J. T., Tehranirad, B., Harris, J. C., and Grilli, S. (2011). FUNWAVE-TVD Fully Nonlinear Boussinesq Wave Model with TVD Solver Documentation and User's Manual. *Center Appl. Coastal Res., Univ. Delaware, Newark, DE, USA, Res. Rep. No CACR-11-04*, 2:1. (Cited on pages 17, 24, 95, 101 and 117.)

Shiach, J. B. and Mingham, C. G. (2009). A temporally second-order accurate Godunov-type scheme for solving the extended Boussinesq equations. *Coastal Engineering*, 56(1):32–45. (Cited on page 83.)

Sørensen, O. R., Schäffer, H. A., and Sørensen, L. S. (2004). Boussinesq-type modelling using an unstructured finite element technique. *Coastal Engineering*, 50(4):181–198. (Cited on page 83.)

Stelling, G. and Zijlema, M. (2003). An accurate and efficient finite-difference algorithm for non-hydrostatic free-surface flow with application to wave propagation. *International journal for numerical methods in fluids*, 43(1):1–23. (Cited on page 24.)

Stelling, G. S. and Duinmeijer, S. A. (2003). A staggered conservative scheme for every Froude number in rapidly varied shallow water flows. *International journal for numerical methods in fluids*, 43(12):1329–1354. (Cited on page 24.)

Stockdon, H. F., Holman, R. A., Howd, P. A., and Sallenger Jr, A. H. (2006). Empirical parameterization of setup, swash, and runup. *Coastal engineering*, 53(7):573–588. (Cited on page 16.)

Storlazzi, C. D., Elias, E. P., and Berkowitz, P. (2015). Many atolls may be uninhabitable within decades due to climate change. *Scientific reports*, 5(1):1–9. (Cited on page 15.)

SWAN, S. (2015). technical documentation (SWAN Cycle III version 41.01 A). *Delft University of Technology*. (Cited on page 16.)

Swigler, D. T. (2010). *Laboratory study investigating the three-dimensional turbulence and kinematic properties associated with a breaking solitary wave*. PhD thesis, Texas A & M University. (Cited on page 131.)

Synolakis, C. E. (1987). The runup of solitary waves. *Journal of Fluid Mechanics*, 185:523–545. (Cited on pages 6, 125 and 128.)

Tavakkol, S. and Lynett, P. (2017). Celeris: A GPU-accelerated open source software with a Boussinesq-type wave solver for real-time interactive simulation and visualization. *Computer Physics Communications*, 217:117–127. (Cited on page 140.)

Titov, V. V. and Gonzalez, F. I. (1997). Implementation and testing of the method of splitting tsunami (MOST) model. *NOAA Technical Memorandum ERL PMEL-112*. (Cited on page 17.)

Tolman, H. L. et al. (2009). User manual and system documentation of WAVEWATCH-III TM version 3.14. Technical note. *MMAB Contribution*, 276:220. (Cited on page 16.)

Tonelli, M. and Petti, M. (2009). Hybrid finite volume–finite difference scheme for 2DH improved Boussinesq equations. *Coastal Engineering*, 56(5-6):609–620. (Cited on pages 19, 83, 96, 101 and 136.)

Toro, E. (1989). A weighted average flux method for hyperbolic conservation laws. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 423(1865):401–418. (Cited on page 19.)

Toro, E. (2001). *Shock-capturing methods for free-surface shallow flows*, volume 868. Wiley New York. (Cited on pages 85 and 95.)

Toro, E. F. (2013). *Riemann solvers and numerical methods for fluid dynamics: a practical introduction*. Springer Science & Business Media. (Cited on page 83.)

Tsutsui, S., Suzuyama, K., and Ohki, H. (1998). Model equations of nonlinear dispersive waves in shallow water and an application of its simplified version to wave evolution on the step-type reef. *Coastal Engineering Journal*, 40(01):41–60. (Cited on page 17.)

Valiani, A., Caleffi, V., and Zanni, A. (2002). Case study: Malpasset dam-break simulation using a two-dimensional finite volume method. *Journal of Hydraulic Engineering*, 128(5):460–472. (Cited on page 57.)

Varing, A., Filipot, J.-F., Grilli, S., Duarte, R., Roeber, V., and Yates, M. (2021). A new definition of the kinematic breaking onset criterion validated with solitary and quasi-regular waves in shallow water. *Coastal Engineering*, 164:103755. (Cited on pages 96 and 97.)

Walkley, M. and Berzins, M. (2002). A finite element method for the two-dimensional extended Boussinesq equations. *International Journal for Numerical Methods in Fluids*, 39(10):865–885. (Cited on page 19.)

WAMDI, G. (1988). The WAM model—A third generation ocean wave prediction model. *Journal of Physical Oceanography*, 18(12):1775–1810. (Cited on page 16.)

Wang, X. (2009). User manual for COMCOT version 1.7 (first draft). *Cornel University*, 65. (Cited on page 17.)

Watanabe, M., Yoshii, T., Roeber, V., Goto, K., and Imamura, F. (2020). Data of boulder transport experiment in super-large wave flume. *Journal of the Sedimentological Society of Japan*, 79(1):15–25. (Cited on page 169.)

Wei, G. and Kirby, J. T. (1995). Time-dependent numerical code for extended Boussinesq equations. *Journal of Waterway, Port, Coastal, and Ocean Engineering*, 121(5):251–261. (Cited on pages 17, 18, 19, 61, 71, 72, 80, 83, 114 and 136.)

Wei, G., Kirby, J. T., and Sinha, A. (1999). Generation of waves in Boussinesq models using a source function method. *Coastal Engineering*, 36(4):271–299. (Cited on pages 114, 115, 136 and 201.)

Yamamoto, S. and Daiguji, H. (1993). Higher-order-accurate upwind schemes for solving the compressible Euler and Navier-Stokes equations. *Computers & Fluids*, 22(2-3):259–270. (Cited on page 83.)

Yamazaki, Y., Cheung, K. F., and Kowalik, Z. (2011). Depth-integrated, non-hydrostatic model with grid nesting for tsunami generation, propagation, and run-up. *International Journal for Numerical Methods in Fluids*, 67(12):2081–2107. (Cited on pages 24 and 95.)

Yates, M. L. and Benoit, M. (2015). Accuracy and efficiency of two numerical methods of solving the potential flow problem for highly nonlinear and dispersive water waves. *International Journal for Numerical Methods in Fluids*, 77(10):616–640. (Cited on page 18.)

Yuan, Y., Shi, F., Kirby, J. T., and Yu, F. (2020). FUNWAVE-GPU: Multiple-GPU Acceleration of a Boussinesq-Type Wave Model. *Journal of Advances in Modeling Earth Systems*, 12(5):e2019MS001957. (Cited on pages 119, 140 and 142.)

Zelt, J. A. (1991). The run-up of nonbreaking and breaking solitary waves. *Coastal Engineering*, 15:205–246. (Cited on page 97.)

Zhang, J. and Benoit, M. (2021). Wave–bottom interaction and extreme wave statistics due to shoaling and de-shoaling of irregular long-crested wave trains over steep seabed changes. *Journal of Fluid Mechanics*, 912:A28. (Cited on page 18.)

Zhang, M. and Wu, W. (2011). A two dimensional hydrodynamic and sediment transport model for dam break based on finite volume method with quadtree grid. *Applied Ocean Research*, 33(4):297–308. (Cited on pages 59 and 98.)

Zhang, Y., Cohen, J., and Owens, J. D. (2010). Fast tridiagonal solvers on the GPU. *ACM Sigplan Notices*, 45(5):127–136. (Cited on page 146.)

Zheleznyak, M. and Pelinovsky, E. (1985). Physical and mathematical models of the tsunami climbing a beach. *Tsunami Climbing a Beach*, pages 8–34. (Cited on page 74.)

Zhou, Q., Zhan, J., and Li, Y. (2016). High-order finite volume WENO schemes for Boussinesq modelling of nearshore wave processes. *Journal of Hydraulic Research*, 54(6):646–662. (Cited on page 83.)

Zijlema, M., Stelling, G., and Smit, P. (2011). SWASH: An operational public domain code for simulating wave fields and rapidly varied flows in coastal waters. *Coastal Engineering*, 58(10):992–1012. (Cited on pages 18 and 24.)

Zou, Z. (1999). Higher order Boussinesq equations. *Ocean Engineering*, 26(8):767–792. (Cited on page 17.)

Zou, Z. (2000). A new form of higher order Boussinesq equations. *Ocean engineering*, 27(5):557–575. (Cited on page 17.)

Appendix A

Governing Equations Derivation

In this appendix, we detail the derivation of the two dispersive Boussinesq equations that are discussed in the thesis: SGN and Nwogu's equations. We follow the derivation approach given by Khakimzyanov et al. [2020]. We first derive a base model, where we maintain a general framework. We then build the two equations from this unified model by including additional assumptions specific to each set of equations.

A.1 Base Model Derivation

We consider the flow of an ideal incompressible fluid in a three-dimensional space. We assume that the fluid is homogeneous (i.e., the density is constant $\rho = 1$) and the acceleration due to gravity g is constant everywhere. To describe the equations, we consider a Cartesian coordinate system $(Oxyz)$. The horizontal plane Oxy coincides with the still water level $z = 0$. The water's free-surface elevation $\eta(x, y, t)$ propagates over a variable bottom $h(x, y, t)$.

The flow is considered to be completely determined if we solve the velocity field $U(x, y, z, t) = \big(u(x, y, z, t), v(x, y, z, t), w(x, y, z, t)\big)$ with pressure field $p(x, y, z, t)$ and the free-surface $\eta(x, y, t)$ alone, which satisfy the system of Euler equations :

$$\nabla.\mathbf{u} + w_z = 0, \tag{A.1}$$

$$\mathbf{u}_t + (\mathbf{u}.\nabla)\,\mathbf{u} + w\mathbf{u}_z + \nabla p = 0, \tag{A.2}$$

$$w_t + \mathbf{u}.\nabla w + ww_z + p_z = -g, \tag{A.3}$$

where $\mathbf{u} = (u, v)$ is the horizontal velocity and $\nabla = (\partial_x, \partial_y)$ denotes the horizontal gradient operator.

Along with the conservation of mass and momentum, the fluid has to satisfy dynamic and kinematic boundary conditions at the free-surface:

$$\eta_t + \mathbf{u}.\nabla\eta = w, \qquad z = \eta(x, y, t). \tag{A.4}$$

$$p = 0, \qquad z = \eta(x, y, t). \tag{A.5}$$

And the impermeability condition (*i.e.*, the fluid particles cannot penetrate the solid boundary) at the seabed:

$$h_t + \mathbf{u}.\nabla h + w = 0, \qquad z = -h(x, y, t). \tag{A.6}$$

Dimensionless variables

We consider the following length scales: the characteristic water depth d, wavelength l, and wave amplitude a. The dimensionless independent variables can be introduced as follows:

$$x^* = \frac{x}{l}, \qquad y^* = \frac{y}{l}, \qquad z^* = \frac{z}{d}, \qquad t^* = \frac{\sqrt{gd}}{l}t.$$

The dependent variables are scaled as:

$$u^* = \frac{u}{\sqrt{gd}}, \qquad v^* = \frac{v}{\sqrt{gd}}, \qquad w^* = \frac{l}{d\sqrt{gd}}w,$$
$$h^* = \frac{h}{d}, \qquad \eta^* = \frac{\eta}{a}, \qquad p^* = \frac{p}{gd}.$$

The scaled Euler equations are expressed as:

$$\nabla.\mathbf{u} + w_z = 0, \tag{A.7}$$
$$\mathbf{u}_t + (\mathbf{u}.\nabla)\,\mathbf{u} + w\mathbf{u}_z + \nabla p = 0, \tag{A.8}$$
$$\mu^2\,(w_t + \mathbf{u}.\nabla w + ww_z) + p_z = -1. \tag{A.9}$$

For simplicity we remove the asterisk symbol $*$. Similarly, the boundary condition at the free surface and bottom becomes:

$$\varepsilon\,(\eta_t + \mathbf{u}.\nabla\eta) = w, \qquad z = \varepsilon\eta\,(x,y,t)\,, \tag{A.10}$$
$$p = 0, \qquad z = \varepsilon\eta\,(x,y,t)\,, \tag{A.11}$$
$$h_t + \mathbf{u}.\nabla h + w = 0, \qquad z = -h\,(x,y,t)\,. \tag{A.12}$$

The parameters $\varepsilon = \frac{a}{d}$ and $\mu = \frac{d}{l}$ are measures of nonlinearity and frequency dispersion, respectively.

Long wave approximation

Under the long wave assumption, we introduce the total water depth variable $H\,(x,y,t) = \varepsilon\eta\,(x,y,t) + h\,(x,y,t)$ and the velocity variable $\bar{\mathbf{u}}\,(x,y,t)$, which is supposed to be a close approximation of the horizontal velocity of Euler equations $\mathbf{u}\,(x,y,z,t)$. The reference velocity $\bar{\mathbf{u}}$ can be taken as the depth-averaged velocity:

$$\bar{\mathbf{u}}\,(x,y,t) = \frac{1}{H\,(x,y,t)}\int_{-h}^{\varepsilon\eta}\mathbf{u}\,(x,y,z,t)\;dz. \tag{A.13}$$

Another popular choice for the velocity variable consists in taking the the horizontal velocity at certain surface $z = \mathscr{Z}\,(x,y,t)$:

$$\bar{\mathbf{u}}\,(x,y,t) = \mathbf{u}\left(x,y,\mathscr{Z}\,(x,y,t)\,,t\right). \tag{A.14}$$

Under the long wave approximations, we assume that the reference velocity $\bar{\mathbf{u}}$ approximates the exact velocity $\mathbf{u}$ to the order $\mathcal{O}\left(\mu^2\right)$:

$$\mathbf{u}\,(x,y,z,t) = \bar{\mathbf{u}}\,(x,y,t) + \mu^2\tilde{\mathbf{u}}\,(x,y,z,t)\,. \tag{A.15}$$

Continuity equation

We integrate the continuity equation (Eq. (A.7)) over the total water depth:

$$\int_{-h}^{\varepsilon\eta}\nabla.\mathbf{u}\,dz + \int_{-h}^{\varepsilon\eta}w_z\,dz = 0,$$
$$\nabla.\left[\int_{-h}^{\varepsilon\eta}\mathbf{u}\,dz\right] - \mathbf{u}|_{z=\varepsilon\eta}\nabla.\left[\varepsilon\eta\right] - \mathbf{u}|_{z=-h}\nabla.\left[h\right] + w|_{z=\varepsilon\eta} - w|_{z=-h} = 0.$$

By using the boundary conditions Eqs. (A.10) and (A.12) and replacing $\mathbf{u}$ with its expression (Eq. (A.15)), we get:

$$\nabla.\left(H\bar{\mathbf{u}}\right)+\mu^2\nabla.\left[\int_{-h}^{\varepsilon\eta}\tilde{\mathbf{u}}\,dz\right]+\varepsilon\eta_t+h_t=0.$$

The depth-integrated continuity equations becomes:

$$H_t+\nabla.\left(H\bar{\mathbf{u}}\right)=-\mu^2\nabla.\left(H\mathscr{U}\right), \tag{A.16}$$

where:

$$\mathscr{U}=\frac{1}{H}\int_{-h}^{\varepsilon\eta}\tilde{\mathbf{u}}\,dz. \tag{A.17}$$

Pressure field

We express the vertical velocity variable $w\left(x,y,z,t\right)$ by integrating the continuity equation over the vertical coordinate and applying the bottom boundary condition (Eq. (A.12)):

$$\begin{aligned}\int_{-h}^{z}\nabla.\mathbf{u}\,dz+\int_{-h}^{z}w_z\,dz&=0,\\ \nabla.\left[\int_{-h}^{z}\mathbf{u}\,dz\right]-\mathbf{u}|_{z=-h}\nabla.\left[h\right]+w\left(x,y,z,t\right)-w|_{z=-h}&=0,\\ \nabla.\Big[\left(z+h\right)\bar{\mathbf{u}}\Big]+\mu^2\nabla.\left[\int_{-h}^{z}\tilde{\mathbf{u}}\,dz\right]+w\left(x,y,z,t\right)+h_t&=0.\end{aligned}$$

We can therefore write the vertical velocity as:

$$w\left(x,y,z,t\right)=-h_t-\bar{\mathbf{u}}\nabla h-\left(z+h\right)\nabla.\bar{\mathbf{u}}+\mathcal{O}\left(\mu^2\right). \tag{A.18}$$

For the sake of simplicity, we introduce the material derivative $\mathscr{D}$ such as:

$$\mathscr{D}\left[.\right]=\left[.\right]+\bar{\mathbf{u}}.\nabla\left[.\right],\qquad \mathscr{D}^2\left[.\right]=\mathscr{D}\left[\mathscr{D}[.]\right] \tag{A.19}$$

Eq. (A.18) becomes:

$$w\left(x,y,z,t\right)=-\mathscr{D}h-\left(z+h\right)\nabla.\bar{\mathbf{u}}+\mathcal{O}\left(\mu^2\right). \tag{A.20}$$

To express the pressure field $p\left(x,y,z,t\right)$, we integrate the vertical momentum equation (Eq. (A.9)) over the vertical coordinate from an arbitrary elevation z to the free surface:

$$\begin{aligned}\mu^2\int_{z}^{\varepsilon\eta}\left[\mathscr{D}w+ww_z\right]dz-p\left(x,y,z,t\right)&=-\varepsilon\eta+z,\\ p\left(x,y,z,t\right)&=\mu^2\int_{z}^{\varepsilon\eta}\left[\mathscr{D}w+ww_z\right]dz-z+\varepsilon\eta.\end{aligned} \tag{A.21}$$

We rewrite the term $\mathscr{D}w + ww_z$ using the expression given in Eq. (A.20):

$$\begin{aligned}\mathscr{D}w &= \mathscr{D}\left[-\mathscr{D}h - (z+h)\,\nabla.\bar{\mathbf{u}} + \mathcal{O}\left(\mu^2\right)\right] \\ &= -\mathscr{D}^2 h - \mathscr{D}h\nabla.\bar{\mathbf{u}} - (z+h)\,\mathscr{D}\nabla.\bar{\mathbf{u}} + \mathcal{O}\left(\mu^2\right),\end{aligned}$$

$$\begin{aligned}ww_z &= \left(-\mathscr{D}h - (z+h)\,\nabla.\bar{\mathbf{u}} + \mathcal{O}\left(\mu^2\right)\right)\left(-\nabla.\bar{\mathbf{u}} + \mathcal{O}\left(\mu^2\right)\right) \\ &= \mathscr{D}h\nabla.\bar{\mathbf{u}} + (z+h)\,(\nabla.\bar{\mathbf{u}})^2 + \mathcal{O}\left(\mu^2\right),\end{aligned}$$

$$\begin{aligned}\mathscr{D}w + ww_z &= -(z+h)\left(\mathscr{D}\nabla.\bar{\mathbf{u}} - (\nabla.\bar{\mathbf{u}})^2\right) - \mathscr{D}^2 h + \mathcal{O}\left(\mu^2\right) \\ &= -(z+h)\,\mathscr{R}_1 - \mathscr{R}_2 + \mathcal{O}\left(\mu^2\right).\end{aligned} \tag{A.22}$$

We define the two variables $\mathscr{R}_1$ and $\mathscr{R}_2$ such as:

$$\mathscr{R}_1 = \mathscr{D}\nabla.\bar{\mathbf{u}} - (\nabla.\bar{\mathbf{u}})^2, \tag{A.23}$$

$$\mathscr{R}_1 = \mathscr{D}^2 h. \tag{A.24}$$

By substituting the last term into Eq.(A.21) and integrating over the vertical direction, we derive the following expression for the pressure field:

$$\begin{aligned}p(x,y,z,t) &= \mu^2 \int_z^{\varepsilon\eta}\left[-(z+h)\,\mathscr{R}_1 - \mathscr{R}_2 + \mathcal{O}\left(\mu^2\right)\right] dz - z + \varepsilon\eta, \\ p(x,y,z,t) &= H - (z+h) - \mu^2\left[\left(\frac{H^2}{2} - \frac{(z+h)^2}{2}\right)\mathscr{R}_1 + \big(H-(z+h)\big)\,\mathscr{R}_2\right] + \mathcal{O}\left(\mu^4\right).\end{aligned} \tag{A.25}$$

We define the non-hydrostatic pressure field $\mathscr{P}$ as:

$$\mathscr{P}(x,y,z,t) = \mu^2\left[\left(\frac{H^2}{2} - \frac{(z+h)^2}{2}\right)\mathscr{R}_1 + \big(H-(z+h)\big)\,\mathscr{R}_2\right] + \mathcal{O}\left(\mu^4\right). \tag{A.26}$$

Horizontal velocity field

In order to obtain the evolution of the approximate horizontal velocity $\bar{\mathbf{u}}(x,y,t)$ we integrated over the total water depth the horizontal momentum equation Eq. (A.8):

$$\begin{aligned}\int_{-h}^{\varepsilon\eta}\left[\mathbf{u}_t + (\mathbf{u}.\nabla)\,\mathbf{u} + w\mathbf{u}_z\right] dz + \int_{-h}^{\varepsilon\eta}\nabla p\, dz = 0, \\ \int_{-h}^{\varepsilon\eta}\left[\mathbf{u}_t + (\mathbf{u}.\nabla)\,\mathbf{u}\right] dz + \int_{-h}^{\varepsilon\eta} w\mathbf{u}_z\, dz + \nabla\int_{-h}^{\varepsilon\eta} p\, dz - p|_{z=-h}\nabla h = 0.\end{aligned}$$

The pressure variable $p\left(x,y,z,t\right)$ can be removed from the last equation using the approximation in Eq. (A.25):

$$\nabla\int_{-h}^{\varepsilon\eta}p\,dz=H\nabla H-\mu^2\left[\nabla\left(\frac{1}{3}H^3\mathscr{R}_1\right)+\nabla\left(\frac{1}{2}H^2\mathscr{R}_2\right)\right]+\mathcal{O}\left(\mu^4\right),$$

$$p|_{z=-h}\nabla h=H\nabla h-\mu^2\left[\frac{1}{2}H^2\mathscr{R}_1+H\mathscr{R}_2\right]\nabla h+\mathcal{O}\left(\mu^4\right).$$

Therefore

$$\begin{aligned}\nabla\int_{-h}^{\varepsilon\eta}p\,dz-p|_{z=-h}\nabla h&\\ =H\nabla\left(\varepsilon\eta\right)-\mu^2\left[\nabla\left(\frac{1}{3}H^3\mathscr{R}_1+\frac{1}{2}H^2\mathscr{R}_2\right)-H\left(\frac{1}{2}H\mathscr{R}_1+\mathscr{R}_2\right)\nabla h\right]+\mathcal{O}\left(\mu^4\right).&\end{aligned}\tag{A.27}$$

Then, by using Eq. (A.20) for the vertical velocity w and Eq. (A.14) for the horizontal velocity $\mathbf{u}$, we can write

$$\begin{aligned}\int_{-h}^{\varepsilon\eta}w\mathbf{u}_z\,dz&=\int_{-h}^{\varepsilon\eta}\left(-\mathscr{D}h-\left(z+h\right)\nabla.\bar{\mathbf{u}}+\mathcal{O}\left(\mu^2\right)\right)\left[\bar{\mathbf{u}}+\mu^2\tilde{\mathbf{u}}\right]_z\,dz\\&=-\mu^2\int_{-h}^{\varepsilon\eta}\left(\mathscr{D}h+\left(z+h\right)\nabla.\bar{\mathbf{u}}\right)\tilde{\mathbf{u}}_z\,dz+\mathcal{O}\left(\mu^4\right)\\&=-\mu^2\left(\tilde{\mathbf{u}}|_{z=\varepsilon\eta}-\tilde{\mathbf{u}}|_{z=-h}\right)\mathscr{D}h-\mu^2\nabla.\bar{\mathbf{u}}\int_{-h}^{\varepsilon\eta}\left[\left(z+h\right)\tilde{\mathbf{u}}\right]_z\,dz+\mu^2\nabla.\bar{\mathbf{u}}\int_{-h}^{\varepsilon\eta}\tilde{\mathbf{u}}\,dz+\mathcal{O}\left(\mu^4\right)\\&=-\mu^2\left(\tilde{\mathbf{u}}|_{z=\varepsilon\eta}-\tilde{\mathbf{u}}|_{z=-h}\right)\mathscr{D}h-\mu^2(\nabla.\bar{\mathbf{u}})H\tilde{\mathbf{u}}|_{z=\varepsilon\eta}+\mu^2(\nabla.\bar{\mathbf{u}})H\mathscr{U}+\mathcal{O}\left(\mu^4\right).\end{aligned}$$

Finally, the integral term can be written as:

$$\int_{-h}^{\varepsilon\eta}w\mathbf{u}_z\,dz=\mu^2\Big[\left(\mathscr{D}h\right)\tilde{\mathbf{u}}|_{z=-h}-\left(\mathscr{D}h+H\nabla.\bar{\mathbf{u}}\right)\tilde{\mathbf{u}}|_{z=\varepsilon\eta}+H\mathscr{U}\nabla.\bar{\mathbf{u}}\Big]+\mathcal{O}\left(\mu^4\right).\tag{A.28}$$

We then detail the local and convective acceleration terms:

$$\begin{aligned}\mathbf{u}_t+\left(\mathbf{u}.\nabla\right)\mathbf{u}&=\left[\bar{\mathbf{u}}+\mu^2\tilde{\mathbf{u}}\right]_t+\left(\left(\bar{\mathbf{u}}+\mu^2\tilde{\mathbf{u}}\right).\nabla\right)\left(\bar{\mathbf{u}}+\mu^2\tilde{\mathbf{u}}\right)\\&=\bar{\mathbf{u}}_t+\mu^2\tilde{\mathbf{u}}_t+(\bar{\mathbf{u}}.\nabla)\bar{\mathbf{u}}+\mu^2(\bar{\mathbf{u}}.\nabla)\tilde{\mathbf{u}}+\mu^2(\tilde{\mathbf{u}}.\nabla)\bar{\mathbf{u}}+\mathcal{O}\left(\mu^4\right)\\&=\mathscr{D}\bar{\mathbf{u}}+\mu^2\mathscr{D}\tilde{\mathbf{u}}+\mu^2(\tilde{\mathbf{u}}.\nabla)\bar{\mathbf{u}}+\mathcal{O}\left(\mu^4\right).\end{aligned}$$

Integrating the last expression along the total water depth yields to:

$$\begin{aligned}\int_{-h}^{\varepsilon\eta}\mathbf{u}_t+\left(\mathbf{u}.\nabla\right)\mathbf{u}\,dz&=H\mathscr{D}\bar{\mathbf{u}}+\mu^2\int_{-h}^{\varepsilon\eta}\mathscr{D}\tilde{\mathbf{u}}\,dz+\mu^2\int_{-h}^{\varepsilon\eta}(\tilde{\mathbf{u}}.\nabla)\bar{\mathbf{u}}\,dz+\mathcal{O}\left(\mu^4\right)\\&=H\mathscr{D}\bar{\mathbf{u}}+\mu^2\left[\mathscr{D}\left(H\mathscr{U}\right)-\mathscr{D}\left(\varepsilon\eta\right)\tilde{\mathbf{u}}|_{z=\varepsilon\eta}-\mathscr{D}h.\tilde{\mathbf{u}}|_{z=-h}\right]+\mu^2H\left(\mathscr{U}.\nabla\right)\bar{\mathbf{u}}+\mathcal{O}\left(\mu^4\right).\end{aligned}\tag{A.29}$$

Adding the two expressions in Eqs. (A.28) and (A.29) yields to:

$$\int_{-h}^{\varepsilon\eta} \mathbf{u}_t + (\mathbf{u}.\nabla)\,\mathbf{u} + w\mathbf{u}_z\,dz = H\mathcal{D}\bar{\mathbf{u}} + \mu^2\Big[\mathcal{D}\,(H\mathcal{U}) + H\,(\mathcal{U}.\nabla)\,\bar{\mathbf{u}} + H\mathcal{U}\nabla.\bar{\mathbf{u}}\Big] \\ - \mu^2\underbrace{\Big[\mathcal{D}H + H\nabla.\bar{\mathbf{u}}\Big]}_{(*)}\tilde{\mathbf{u}}|_{z=\varepsilon\eta} + \mathcal{O}\left(\mu^4\right). \tag{A.30}$$

Using the mass conservation equation (Eq. (A.16)), we can rewrite the term $(*)$ as:

$$\mathcal{D}H + H\nabla.\bar{\mathbf{u}} = H_t + \bar{\mathbf{u}}.\nabla(H) + H\nabla.\bar{\mathbf{u}} = H_t + \nabla\,(H\bar{\mathbf{u}}) = -\mu^2\nabla.\,(H\mathcal{U}) = \mathcal{O}\left(\mu^2\right).$$

Therefore:

$$\mu^2\Big[\mathcal{D}H + H\nabla.\bar{\mathbf{u}}\Big]\tilde{\mathbf{u}}|_{z=\varepsilon\eta} = \mathcal{O}\left(\mu^4\right).$$

By adding Eqs. (A.27) and (A.30) , we derive the depth-integrated form of the horizontal momentum equation:

$$H\mathcal{D}\bar{\mathbf{u}} + H\nabla\,(\varepsilon\eta) = \mu^2\left[\nabla\left(\frac{1}{3}H^3\mathcal{R}_1 + \frac{1}{2}H^2\mathcal{R}_2\right) - H\left(\frac{1}{2}H\mathcal{R}_1 + \mathcal{R}_2\right)\nabla h\right] \\ - \mu^2\Big[\mathcal{D}\,(H\mathcal{U}) + H\,(\mathcal{U}.\nabla)\,\bar{\mathbf{u}} + H\mathcal{U}\nabla.\bar{\mathbf{u}}\Big] + \mathcal{O}\left(\mu^4\right). \tag{A.31}$$

We divide the last expression by H and neglect terms of $\mathcal{O}\left(\mu^4\right)$

$$\bar{\mathbf{u}}_t + (\bar{\mathbf{u}}.\nabla)\bar{\mathbf{u}} + \varepsilon\nabla\eta = \frac{\mu^2}{H}\left[\nabla\left(\frac{1}{3}H^3\mathcal{R}_1 + \frac{1}{2}H^2\mathcal{R}_2\right) - H\left(\frac{1}{2}H\mathcal{R}_1 + \mathcal{R}_2\right)\nabla h\right] \\ - \frac{\mu^2}{H}\Big[\mathcal{D}\,(H\mathcal{U}) + H\,(\mathcal{U}.\nabla)\,\bar{\mathbf{u}} + H\mathcal{U}\nabla.\bar{\mathbf{u}}\Big]. \tag{A.32}$$

Using the non-hydrostatic pressure field formulation in Eq. (A.26), we introduce two new variables: the depth-integrated non-hydrostatic pressure $\wp$ and non-hydrostatic pressure trace ϱ at the bottom:

$$\wp\,(x,y,t) = \int_{-h}^{\varepsilon\eta}\mathcal{P}\,(x,y,z,t)\;dz = \mu^2\Big(\frac{1}{3}H^3\mathcal{R}_1 + \frac{1}{2}H^2\mathcal{R}_2\Big), \tag{A.33}$$

$$\varrho\,(x,y,t) = \mathcal{P}\,(x,y,-h,t) = \mu^2\Big(\frac{1}{2}H^2\mathcal{R}_1 + H\mathcal{R}_2\Big). \tag{A.34}$$

Finally, we substitute the two pressure terms into the momentum equation:

$$\bar{\mathbf{u}}_t + (\bar{\mathbf{u}}.\nabla)\bar{\mathbf{u}} + \varepsilon\nabla\eta = \frac{\nabla\wp - \varrho\nabla h}{H} - \frac{\mu^2}{H}\Big[\mathcal{D}\,(H\mathcal{U}) + H\,(\mathcal{U}.\nabla)\,\bar{\mathbf{u}} + H\mathcal{U}\nabla.\bar{\mathbf{u}}\Big].$$

Summary

We have derived a base model by depth-integrating the Euler equations under the long wave assumption (Eq. (A.14)):

$$H_t + \nabla.\left(H\bar{\mathbf{u}}\right) = -\mu^2\nabla.\left(H\mathcal{U}\right), \tag{A.35}$$

$$\bar{\mathbf{u}}_t + (\bar{\mathbf{u}}.\nabla)\bar{\mathbf{u}} + \varepsilon\nabla\eta = \frac{\nabla\mathcal{P} - \varrho\nabla h}{H} - \frac{\mu^2}{H}\Big[\mathcal{D}\left(H\mathcal{U}\right) + H\left(\mathcal{U}.\nabla\right)\bar{\mathbf{u}} + H\mathcal{U}\nabla.\bar{\mathbf{u}}\Big]. \tag{A.36}$$

It is worth noting that by setting $\mu = 0$ in Eqs. (A.35) and (A.36), we can directly obtain the well-known Shallow Water Equations. In order to close the last system of equations, we have to express the variable $\mathcal{U}$ in terms of the other dynamic variables η and $\bar{\mathbf{u}}$. Since the derivation procedure of the base model has been kept quite general, this model can be used to derive many well-established equations (see Khakimzyanov et al. [2020]). In this chapter, we focus on the derivation of SGN and Nwogu equations.

A.2 Serre–Green–Naghdi Equations

The celebrated SGN equations can be obtained by adopting the simplest closure possible:

$$\mathcal{U} \equiv 0, \tag{A.37}$$

This closure is based on the assumption that the horizontal velocity variable $\bar{\mathbf{u}}$ is chosen as the depth-averaged velocity:

$$\frac{1}{H}\int_{-h}^{\varepsilon\eta} \mathbf{u}\, dz = \bar{\mathbf{u}} + \mathcal{U} \qquad \rightarrow \qquad \mathcal{U} = 0$$

By substituting the proposed closure into Eqs. (A.35) and (A.36), we obtain the SGN equations:

$$H_t + \nabla.\left(H\bar{\mathbf{u}}\right) = 0, \tag{A.38}$$

$$\bar{\mathbf{u}}_t + (\bar{\mathbf{u}}.\nabla)\bar{\mathbf{u}} + \varepsilon\nabla\eta = \frac{\nabla\mathcal{P} - \varrho\nabla h}{H}. \tag{A.39}$$

A.3 Nwogu Equations

Another approach to specifying the variable $\mathcal{U}$ and thus closing the system of equations is based on employing the partial irrotationality condition. Namely, we assume that only two horizontal components of vorticity vanish:

$$v_z - w_y = 0, \qquad w_x - u_z = 0. \tag{A.40}$$

We can write this condition in a vector form with non-dimensional variables:

$$\mathbf{u}_z = \mu^2\nabla w. \tag{A.41}$$

We integrate the last formula over the vertical coordinate:

$$\int_{-h}^{z} \mathbf{u}_z \, dz = \mu^2 \int_{-h}^{z} \nabla w \, dz.$$

We substitute $\mathbf{u}$ and w with the expressions given in Eqs. (A.14) and (A.20):

$$\mu^2 \int_{-h}^{z} \tilde{\mathbf{u}}_z \, dz = \mu^2 \int_{-h}^{z} \nabla \Big[-\mathscr{D} h - (z+h) \nabla . \bar{\mathbf{u}} \Big] dz + \mathcal{O}\left(\mu^4\right),$$

$$\tilde{\mathbf{u}}(x,y,z,t) - \tilde{\mathbf{u}}|_{z=-h} = -(z+h) \nabla (\mathscr{D} h) - \int_{-h}^{z} \nabla h \nabla . \bar{\mathbf{u}} \, dz - \int_{-h}^{z} (z+h) \nabla (\nabla . \bar{\mathbf{u}}) \, dz + \mathcal{O}\left(\mu^2\right),$$

$$\tilde{\mathbf{u}}(x,y,z,t) = \tilde{\mathbf{u}}|_{z=-h} - (z+h) \Big[\nabla (\mathscr{D} h) + \nabla h \nabla . \bar{\mathbf{u}} \Big] - \frac{(z+h)^2}{2} \nabla (\nabla . \bar{\mathbf{u}}) + \mathcal{O}\left(\mu^2\right). \tag{A.42}$$

Consequently, the horizontal velocity can be written as:

$$\mathbf{u}(x,y,z,t) = \bar{\mathbf{u}} + \mu^2 \Big[(z+h) \mathscr{A} + \frac{1}{2} (z+h)^2 \mathscr{B} + \mathscr{C} \Big] + \mathcal{O}\left(\mu^4\right), \tag{A.43}$$

where:

$$\begin{aligned} \mathscr{A}(x,y,t) &= -\nabla (\mathscr{D} h) - \nabla h \nabla . \bar{\mathbf{u}}, \\ \mathscr{B}(x,y,t) &= -\nabla (\nabla . \bar{\mathbf{u}}), \\ \mathscr{C}(x,y,t) &= \tilde{\mathbf{u}}|_{z=-h}. \end{aligned}$$

We define the reference velocity $\bar{\mathbf{u}}$ as the trace of the horizontal velocity $\mathbf{u}$ at a certain elevation: z_α:

$$\bar{\mathbf{u}}(x,y,t) \equiv \mathbf{u}(x,y,z_\alpha,t). \tag{A.44}$$

Consequently, we have:

$$\mathscr{C}(x,y,t) = -(z_\alpha + h) \mathscr{A} - \frac{1}{2} (z_\alpha + h)^2 \mathscr{B}. \tag{A.45}$$

Thus, coefficient $\mathscr{C}$ can be removed from Eq. (A.43) to give the following representation:

$$\mathbf{u}(x,y,z,t) = \bar{\mathbf{u}} + \mu^2 \left[(z - z_\alpha) \mathscr{A} + \frac{1}{2} \Big[(z+h)^2 - (z_\alpha + h)^2 \Big] \mathscr{B} \right] + \mathcal{O}\left(\mu^4\right). \tag{A.46}$$

The variable $\mathscr{U}$ can therefore be expressed as:

$$\mathscr{U}(H, \bar{\mathbf{u}}) = \frac{1}{H} \int_{-h}^{\varepsilon\eta} \left[(z - z_\alpha) \mathscr{A} + \frac{1}{2} \Big[(z+h)^2 - (z_\alpha + h)^2 \Big] \mathscr{B} \right] dz + \mathcal{O}\left(\mu^2\right),$$

$$\mathscr{U}(H, \bar{\mathbf{u}}) = \Big[\frac{H}{2} - (z_\alpha + h) \Big] \mathscr{A} + \Big[\frac{1}{6} H^2 - \frac{1}{2} (z_\alpha + h)^2 \Big] \mathscr{B} + \mathcal{O}\left(\mu^2\right). \tag{A.47}$$

By substituting the expression of $\mathscr{U}$ into the base model (Eqs. (A.35) and (A.36)), we obtain the fully nonlinear Nwogu's equations. A time-varying reference elevation z_α is chosen to optimize the dispersive and nonlinear behavior of the equations as suggested by Kennedy et al. [2001]:

$$z_\alpha = \beta h + (1 + \beta) \varepsilon \eta, \tag{A.48}$$

where $\beta = -0.531$ (Nwogu [1993]).

A.3.1 Weakly-nonlinear equations

It is important to note that the small amplitude assumption was never used. Therefore, the base model and the derived Nwogu and SGN equations are fully nonlinear. In the present section, we derive the standard Nwogu's equations (weakly nonlinear) from the fully nonlinear model (Eqs. (A.35) and (A.36)). In this case, we assume that the non-linearity and the dispersion parameters have approximately the same order of magnitude:

$$\varepsilon = \mathcal{O}\left(\mu^2\right). \tag{A.49}$$

Under the small amplitude condition, the variable $\mathscr{U}$ becomes:

$$\mathscr{U}\left(h,\bar{\mathbf{u}}\right) = -\left[\frac{h}{2}+z_\alpha\right]\mathscr{A} + \left[\frac{h^2}{6}+\frac{(z_\alpha+h)^2}{2}\right]\mathscr{B} + \mathcal{O}\left(\mu^2\right). \tag{A.50}$$

We consider the case of non-varying water depth: $h_t = 0$ and we express the variable $\mathscr{A}$ under this assumption:

$$\begin{aligned}
\mathscr{A}\left(x,y,t\right) &= -\nabla\left(\mathscr{D}h\right) - \nabla h\nabla.\bar{\mathbf{u}} \\
&= -\nabla\left(\bar{\mathbf{u}}.\nabla h\right) - \nabla h\nabla.\bar{\mathbf{u}} \\
&= -\nabla\left(\nabla.\left(h\bar{\mathbf{u}}\right)\right) + \nabla\left(h\nabla.\bar{\mathbf{u}}\right) - \nabla h\nabla.\bar{\mathbf{u}} \\
&= -\nabla\left(\nabla.\left(h\bar{\mathbf{u}}\right)\right) + h\nabla\left(\nabla.\bar{\mathbf{u}}\right).
\end{aligned}$$

Thus, we can write $\mathscr{U}$ as:

$$\mathscr{U}\left(h,\bar{\mathbf{u}}\right) = \left[\frac{h}{2}+z_\alpha\right]\nabla\left(\nabla.\left(h\bar{\mathbf{u}}\right)\right) - \left[\frac{h^2}{6}+\frac{(z_\alpha+h)^2}{2} - h\left(\frac{h}{2}+z_\alpha\right)\right]\nabla\left(\nabla.\bar{\mathbf{u}}\right) + \mathcal{O}\left(\mu^2\right), \tag{A.51}$$

$$\mathscr{U}\left(h,\bar{\mathbf{u}}\right) = \left[\frac{h}{2}+z_\alpha\right]\nabla\left(\nabla.\left(h\bar{\mathbf{u}}\right)\right) + \left[\frac{z_\alpha^2}{2}-\frac{h^2}{6}\right]\nabla\left(\nabla.\bar{\mathbf{u}}\right) + \mathcal{O}\left(\mu^2\right). \tag{A.52}$$

Substituting the last expression into the continuity equation Eq.(A.35) and taking into account the weakly nonlinear assumption yields:

$$H_t + \nabla.\left(H\bar{\mathbf{u}}\right) + \nabla.\left\{\left[z_\alpha+\frac{h}{2}\right]h\nabla\left(\nabla.\left(h\bar{\mathbf{u}}\right)\right) + \left[\frac{z_\alpha^2}{2}-\frac{h^2}{6}\right]h\nabla\left(\nabla.\bar{\mathbf{u}}\right)\right\} = 0, \qquad \text{with } z_\alpha = \beta h. \tag{A.53}$$

By neglecting all the nonlinear dispersive terms in the momentum equations, the Nwogu momentum equation can be written as:

$$\bar{\mathbf{u}}_t + \left(\bar{\mathbf{u}}.\nabla\right)\bar{\mathbf{u}} + \varepsilon\nabla\eta = \underbrace{\frac{\mu^2}{H}\left[\nabla\left(\frac{h^3}{3}\mathscr{R}_1 + \frac{h^2}{2}\mathscr{R}_2\right) - h\left(\frac{h}{2}\mathscr{R}_1+\mathscr{R}_2\right)\nabla h - \left(h\mathscr{U}\right)_t\right]}_{(**)}, \tag{A.54}$$

where:

$$\mathscr{R}_1 = \nabla.(\bar{\mathbf{u}}_t), \qquad \mathscr{R}_2 = \bar{\mathbf{u}}_t.\nabla h. \tag{A.55}$$

By substituting the expression of $\mathscr{U}$ into the term $(**)$, we get:

$$\begin{aligned}\frac{(**)}{\mu^2} &= \frac{h}{2}\nabla\left(\nabla.(h\bar{\mathbf{u}}_t)\right) - \frac{h^2}{6}\nabla\left(\nabla.\bar{\mathbf{u}}_t\right) - \left(z_\alpha + \frac{h}{2}\right)\nabla\left(\nabla.(h\bar{\mathbf{u}}_t)\right) - \left(\frac{z_\alpha^2}{2} - \frac{h^2}{6}\right)\nabla\left(\nabla.\bar{\mathbf{u}}_t\right) \\ &= -\left[z_\alpha\nabla\left(\nabla.(h\bar{\mathbf{u}})\right) + \frac{z_\alpha^2}{2}\nabla\left(\nabla.\bar{\mathbf{u}}\right)\right]_t.\end{aligned}$$

Thus, we derive the momentum equation proposed by Nwogu [1993]:

$$\bar{\mathbf{u}}_t + (\bar{\mathbf{u}}.\nabla)\bar{\mathbf{u}} + g\nabla\eta + \left\{\frac{z_\alpha^2}{2}\nabla(\nabla\cdot\bar{\mathbf{u}}) + z_\alpha\nabla\left[\nabla\cdot(h\bar{\mathbf{u}})\right]\right\}_t = 0. \tag{A.56}$$

Appendix B

Derivation of the SGN Elliptic System

In this appendix, we derive the elliptic equation Eq. (3.6). and the relations Eq. (3.9) from the standard form of the SGN equations.

The 1-dimensional SGN momentum equation is written in a non-conservative form as:

$$u_t + uu_x + g\eta_x = \frac{\wp_x - \varrho h_x}{H}, \tag{B.1}$$

where:

$$\wp = \frac{H^3}{3}\mathscr{R}_1 + \frac{H^2}{2}\mathscr{R}_2, \tag{B.2}$$

$$\varrho = \frac{H^2}{2}\mathscr{R}_1 + H\mathscr{R}_2. \tag{B.3}$$

The terms $\mathscr{R}_1$ and $\mathscr{R}_2$ are expressed as it follows:

$$\mathscr{R}_1 = u_{xt} + uu_{xx} - (u_x)^2, \tag{B.4}$$

$$\mathscr{R}_2 = u_t h_x + u\left[uh_x\right]_x. \tag{B.5}$$

Using the definitions in Eqs. (B.2) and (B.3), we express ϱ in terms of $\wp$ and $\mathscr{R}_2$:

$$\varrho = \frac{3}{2H}\wp + \frac{H}{4}\mathscr{R}_2. \tag{B.6}$$

By taking the momentum equation into account (Eq. (B.1)), we rewrite the expression of $\mathscr{R}_2$:

$$\begin{aligned}
\mathscr{R}_2 &= u_t h_x + u\left[uh_x\right]_x \\
&= h_x\left(u_t + uu_x\right) + u^2 h_{xx} \\
&= h_x\left(-g\eta_x + \frac{\wp_x - \varrho h_x}{H}\right) + u^2 h_{xx} \\
&= \left(-g\eta_x h_x + u^2 h_{xx}\right) + \wp_x\frac{h_x}{H} - \varrho\frac{h_x^2}{H} \\
&= \mathscr{R} + \wp_x\frac{h_x}{H} - \varrho\frac{h_x^2}{H}.
\end{aligned} \tag{B.7}$$

We then replace this new expression of $\mathscr{R}_2$ (Eq. (B.7)) into Eq. (B.6):

$$\begin{aligned}
\varrho &= \frac{3}{2H}\wp + \frac{H}{4}\left(\mathscr{R} + \wp_x\frac{h_x}{H} - \varrho\frac{h_x^2}{H}\right) \\
&= \frac{3}{2H}\wp + \frac{H}{4}\mathscr{R} + \frac{h_x}{4}\wp_x - \frac{h_x^2}{4}\varrho.
\end{aligned} \tag{B.8}$$

We finally derive the expression that relates the depth-averaged non-hydrostatic pressure $\wp$ to the non-hydrostatic pressure at the bottom ϱ:

$$\varrho\left(\frac{1+h_x^2}{4}\right) = \frac{3}{2H}\wp + \frac{H}{4}\mathscr{R} + \frac{h_x}{4}\wp_x. \tag{B.9}$$

Therefore:

$$\varrho = \frac{1}{\mathcal{Y}}\left[\frac{6\wp}{H} + H\mathcal{R} + \wp_x h_x\right]. \tag{B.10}$$

The derivation of the elliptic equations Eq. (3.9) is somehow similar. From Eqs. (B.2) and (B.3) we derive another expression relating the two non-hydrostatic pressures $\wp$ and ϱ:

$$\wp = \frac{H}{2}\varrho + \frac{H^3}{12}\mathcal{R}_1. \tag{B.11}$$

By using the momentum equation (Eq. (B.1)), we derive another expression for the variable $\mathcal{R}_1$:

$$\begin{aligned}\mathcal{R}_1 &= u_{xt} + uu_{xx} - (u_x)^2 \\ &= [u_t + uu_x]_x - 2(u_x)^2 \\ &= \left[-g\eta_x + \frac{\wp_x - \varrho h_x}{H}\right]_x - 2(u_x)^2.\end{aligned} \tag{B.12}$$

We then rewrite Eq. (B.11) as follows:

$$\left[-g\eta_x + \frac{\wp_x - \varrho h_x}{H}\right]_x - 2(u_x)^2 + \frac{6}{H^2}\varrho - \frac{12}{H^3}\wp = 0. \tag{B.13}$$

By substituting the expression of ϱ (Eq. (B.10)) into the last equation, we get:

$$\begin{aligned}-g\eta_{xx} + \left[\frac{\wp_x}{H}\right]_x - \left[\frac{h_x}{H\mathcal{Y}}\left(\frac{6\wp}{H} + H\mathcal{R} + \wp_x h_x\right)\right]_x - 2(u_x)^2 + \frac{6}{H^2\mathcal{Y}}\left(\frac{6\wp}{H} + H\mathcal{R} + \wp_x h_x\right) \\ - \frac{12}{H^3}\wp = 0.\end{aligned} \tag{B.14}$$

We assemble all the terms with $\wp$ with second-order derivatives (terms in blue):

$$\left[\frac{\wp_x}{H}\right]_x - \left[\frac{h_x^2}{H\mathcal{Y}}\wp_x\right]_x = \left[\frac{(\mathcal{Y} - h_x^2)\,\wp_x}{H\mathcal{Y}}\right]_x = 4\left[\frac{\wp_x}{H\mathcal{Y}}\right]_x. \tag{B.15}$$

Similarly, we put together all the terms that contain $\wp$ with first-order derivatives (terms in blue):

$$-\left[\frac{6h_x}{H^2\mathcal{Y}}\wp\right]_x + \frac{6h_x}{H^2\mathcal{Y}}\wp_x = -\left[\frac{6h_x}{H^2\mathcal{Y}}\right]_x \wp. \tag{B.16}$$

Finally, we rewrite the terms that contain the variable $\wp$ with no space derivatives (terms in green):

$$\frac{36}{H^3\mathscr{Y}}\wp - \frac{12}{H^3}\wp - \left[\frac{6h_x}{H^2\mathscr{Y}}\right]_x \wp = -6\left(\frac{2\left(\mathscr{Y}-3\right)}{H^3\mathscr{Y}} + \left[\frac{h_x}{H^2\mathscr{Y}}\right]_x\right)\wp. \tag{B.17}$$

We then add all the terms left in Eq. (B.14) together (source terms):

$$-g\eta_{xx} - \left[\frac{h_x\mathscr{R}}{\mathscr{Y}}\right]_x - 2\left(u_x\right)^2 + \frac{6\mathscr{R}}{H\mathscr{Y}} = \left[g\eta_x + \frac{\mathscr{R}\,h_x}{\mathscr{Y}}\right]_x - \frac{6\mathscr{R}}{H\mathscr{Y}} + 2u_x^2 = -\mathscr{F}. \tag{B.18}$$

Adding all the terms in Eqs. (B.15), (B.17), and (B.18) yields the final elliptic equation:

$$4\left[\frac{\wp_x}{H\mathscr{Y}}\right]_x - 6\left[\frac{2}{H^3}.\frac{\mathscr{Y}-3}{\mathscr{Y}} + \left[\frac{h_x}{H^2\mathscr{Y}}\right]_x\right]\wp = \mathscr{F}. \tag{B.19}$$

Appendix C

Modified SGN Equations with Improved Dispersion

C.1 Improved SGN Equations

Several extensions of the classical SGN equations have been proposed to improve the dispersion properties of the equations and therefore extend their range of applicability. This can be done by introducing high-order terms into the equations (Madsen et al. [1991]). Another approach is based on introducing a free parameter into the model, which is subsequently selected to improve the desired properties (*e.g.*, Antunes Do Carmo [2013]; Clamond et al. [2017]; Dias and Milewski [2010]). Similar to the latter approach, we improve the dispersive properties of the SGN model by including a free parameter in the elliptic equation. The modified equations can be obtained by replacing the elliptic equations (*i.e.*, Eq. (3.6)) by:

$$4\,(1+\delta)\left[\frac{\wp_x}{H\mathscr{Y}}\right]_x - 6\left[\frac{2}{H^3}.\frac{\mathscr{Y}-3}{\mathscr{Y}} + \left[\frac{h_x}{H^2\mathscr{Y}}\right]_x\right]\wp = \mathscr{F}, \tag{C.1}$$

where δ is a free parameter at our disposal. It should be noted that the original SGN equations can be retrieved by taking $\delta = 0$. The modified SGN system is a regular perturbation of the original one. We thus expect the solution of the modified SGN equations to satisfy the following condition:

$$H_\delta - H = \mathcal{O}(\delta), \qquad u_\delta - u = \mathcal{O}(\delta), \qquad \wp_\delta - \wp = \mathcal{O}(\delta).$$

The linearized modified SGN equations on a flat bottom become:

$$\eta_x + bu_x = 0, \tag{C.2}$$

$$u_t + g\eta_x = \frac{\wp_x}{b}, \tag{C.3}$$

$$\wp_{xx} - \frac{3}{b^3} = gb\eta_{xx}, \tag{C.4}$$

where b is the constant water depth. We take the plane wave solutions of the form:

$$\eta = \eta_0 \mathrm{e}^{\mathrm{i}(kx-\omega t)}, \qquad u = u_0 \mathrm{e}^{\mathrm{i}(kx-\omega t)}, \qquad \wp = \wp_0 \mathrm{e}^{\mathrm{i}(kx-\omega t)}, \tag{C.5}$$

where k is wave number and ω is the wave frequency. We substitute these solutions into the linearized equations Eq. (C.4) we get:

$$A\,(-\mathrm{i}\omega, \mathrm{i}k)\begin{pmatrix}\eta\\ u\\ \wp\end{pmatrix} = \begin{pmatrix}0\\0\\0\end{pmatrix}, \tag{C.6}$$

where

$$A(-\mathrm{i}\omega, \mathrm{i}k) = \begin{pmatrix} -\mathrm{i}\omega & \mathrm{i}bk & 0 \\ \mathrm{i}gk & -\mathrm{i}\omega & -\mathrm{i}\frac{k}{b} \\ ghk^2 & 0 & -(1+\delta)k^2 - \frac{3}{b^2} \end{pmatrix}. \tag{C.7}$$

Non-zero solutions are obtained if $\det\left[A(-\mathrm{i}\omega, \mathrm{i}k)\right] = 0$, which yields the dispersion relation:

$$-\frac{b^3gk^4\delta - b^2k^2\delta\omega^2 - b^2k^2\omega^2 + 3bgk^2 - 3\omega^2}{b^2} = 0. \tag{C.8}$$

The phase celerity of the modified SGN is thus:

$$C_{SGN,\delta}(kb) = \frac{\omega}{k} = \sqrt{gb}\sqrt{\frac{(kb)^2\delta + 3}{(1+\delta)(kb)^2 + 3}}. \tag{C.9}$$

It is important to highlight that we can retrieve the exact dispersion properties of Nwogu's equations by taking $\delta = -(3\alpha + 1)$

C.2 Optimized Equations

We optimize the free parameter δ to match the phase celerity of Airy theory. C_{Airy} is given by:

$$C_{Airy}(kb) = \sqrt{gb}\sqrt{\frac{\tanh(kb)}{kb}}. \tag{C.10}$$

We set $x = kb$ and, for a given $x_{\max}$, we consider the following function:

$$J(\delta) = \left(\frac{1}{x_{\max}}\int_0^{x_{\max}} \left(C_{SGN,\delta}(x) - C_{Airy}(x)\right)^2 dx\right)^{1/2}. \tag{C.11}$$

We use a uniform subdivision of the interval $[0, x_{\max}]$ with N points and the previous function is then minimized:

$$J(\delta) = \sqrt{\frac{1}{N}\sum_{j=1}^{N}\left(C_{\mathrm{SGN},\delta}(x_j) - C_{\mathrm{Airy}}(x_j)\right)^2}. \tag{C.12}$$

The Nelder-Mead algorithm is then used with $a = 0$ as an initial guess to minimize J. Figure 1 shows the variation of the optimal free parameter $\delta_{\min}$ and the corresponding J values with respect to $x_{\max}$ used in the optimization.

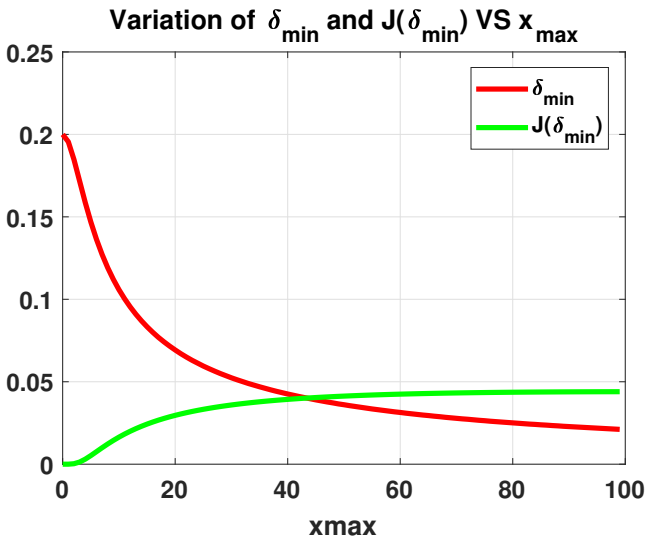


Figure C.1: Variation of $\delta_{\min}$ and $J(\delta_{\min})$ for differnt $x_{\max}$.

For small kb values, the optimal dispersion property of the modified SGN equations is achieved by taking $\delta = 0.2$

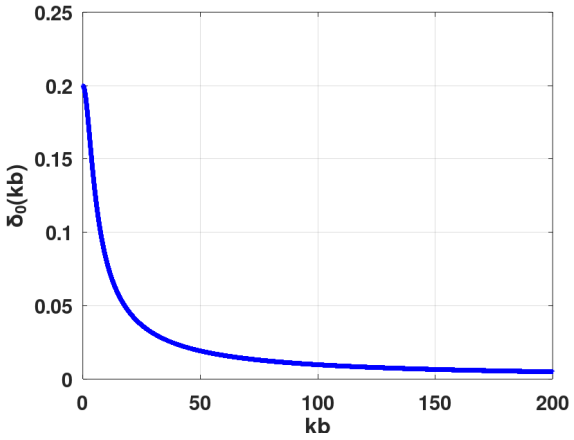
C.2.1 Optimized phase celerity for a single kb

The optimization results from the previous subsection are very useful for problems involving several spatial frequencies. In the case where only one frequency $kb = x_0$ is fixed, we can derive an optimal δ_0 such as:

$$C_{\mathrm{SGN},\delta_0}(x_0) = C_{\mathrm{Airy}}(x_0). \tag{C.13}$$

The previous equation can be analytically solved and leads to the following exact solution:

$$\delta_0 = \frac{x_0^2 \exp(2x_0) - 3x_0 \exp(2x_0) - x_0^2 + 3\exp(2x_0) - 3x_0 - 3}{((x_0 \exp(2x_0) - \exp(2x_0) + x_0 + 1)x_0^2}. \tag{C.14}$$



It is worth noting that, as $x_0 \to 0$, we obtain:

$$\delta_0 = \frac{1}{5} + \mathcal{O}(x_0). \tag{C.15}$$

We again recover that $\delta_{\min} = 1/5$ is the optimal value for small enough kb.

C.2.2 Group celerity

We investigate the group velocity of the modified SGN system. The group celerity is defined as:

$$C_g = \partial_k \omega = \partial_k (kC) = k\partial_k C + C. \tag{C.16}$$

For the Airy wave theory, the group velocity is defined as:

$$C_{g,\text{Airy}}(kb) = \frac{\sqrt{gb}}{2} \frac{\sinh(kb)\cosh(kb) + kb}{\cosh(kb)^{3/2}\sqrt{kb\sinh(kb)}}. \tag{C.17}$$

The group velocity of the modified SGN equations can be expressed as:

$$C_{g,\text{SGN},\delta} = \sqrt{gb} \frac{(kb)^4\delta^2 + (kb)^4\delta + 6(kb)^2\delta + 9}{((kb)^2\delta + (kb)^2 + 3)^{3/2}\sqrt{(kb)^2\delta + 3}}. \tag{C.18}$$

Similar to the approach previously used to optimize the phase celerity, we derive the value $\delta_{\min}$, which leads to the optimal improvement of the group velocity for values of $x = kb$ between $[0, x_{\max}]$.

Figure. C.2 shows the variation of the group velocity with respect to kb for different optimization intervals $[0, x_{\max}]$

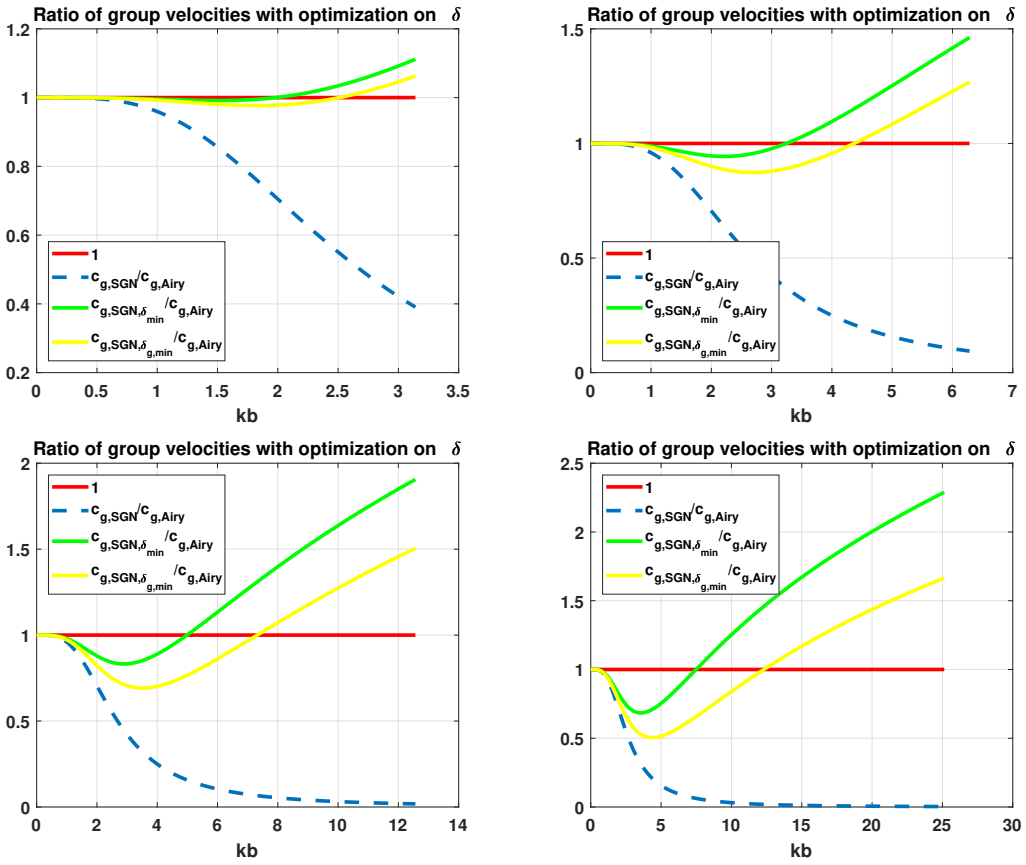


Figure C.2: $C_{g,SGN,\delta_{\min}}/C_{g,Airy}$ for several values of $x_{\max}$. Up Left: $x_{\max} = \pi$, Up Right: $x_{\max} = 2\pi$, Bottom Left: $x_{\max} = 4\pi$, Bottom Right: $x_{\max} = 8\pi$.

C.3 Preliminary Numerical Results

C.3.1 Sine wave on flat bottom

To better illustrate the effects of the improved dispersion in the modified SGN equations, we generate and propagate a linear monochromatic wave ($\frac{a}{b} = 0.01 \ll 1$) in a long flat channel. The results from the modified model are then compared to the fully dispersive Airy theory. The generation of the waves inside the computational domain is achieved through the internal wavemaker approach proposed by Wei et al. [1999]. For each run, the models are executed with a different values of $kb = 3, 5$, and 10. Based on Eq. (C.14) , we choose delta which matches Airy wave theory for each value of kb.

The grid size is determined for each test such that $\zeta = \frac{\lambda}{\Delta x} = 20$. The results are shown in Figure. C.3

The model is stable for high kb values and accurately reproduces a particular monochromatic wave with its appropriate δ value.

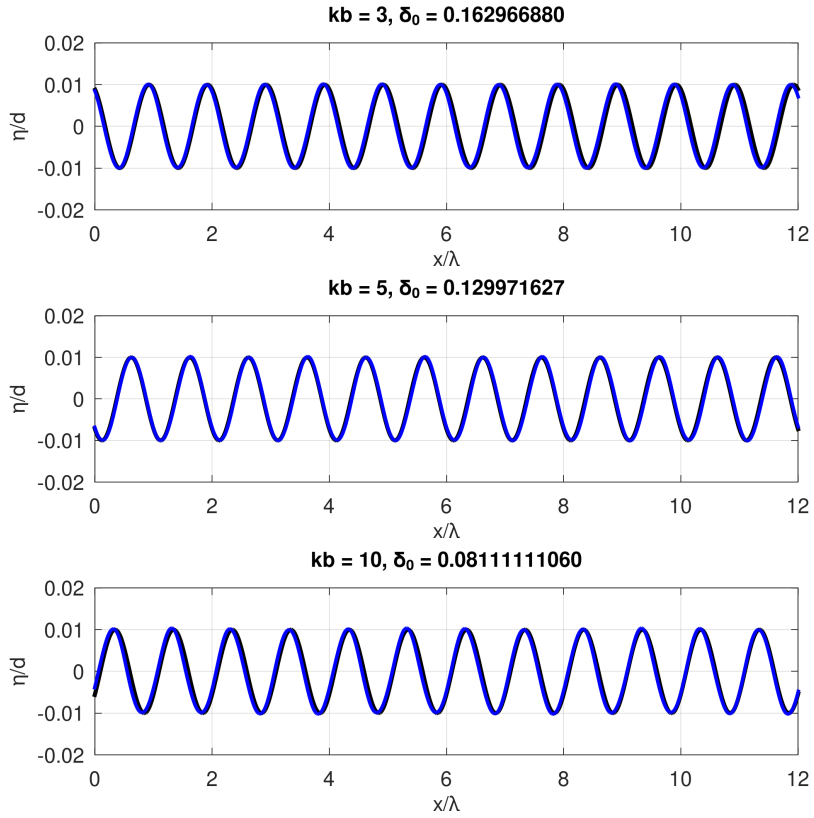


Figure C.3: The black line describes the analytical solution from Airy wave theory and the blue line represents the modified SGN solution

C.3.2 Periodic wave propagation over a submerged bar

We repeat the benchmark test described in Chapter 1. We compute the numerical solution with both the standard and modified SGN equations. This test highlights the improved numerical results with the modified SGN equations due to the enhanced linear dispersion. See Figure. C.4.

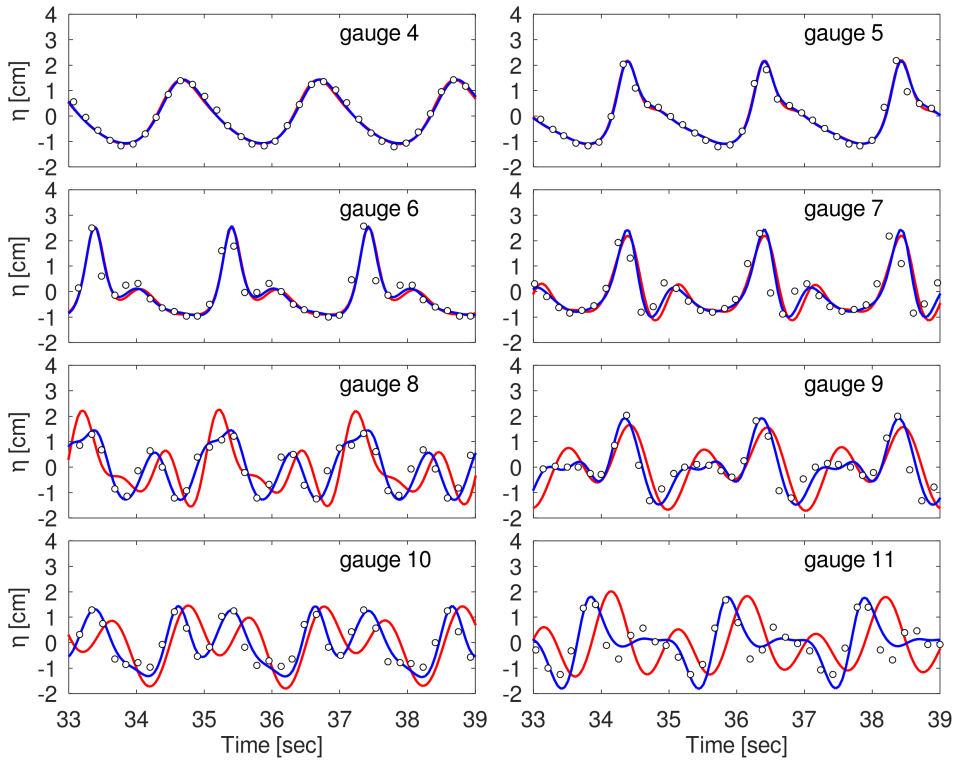


Figure C.4: Computation of wave transformation over a submerged bar. Black circles denote laboratory data from Beji and Battjes [1993]. The red lines are time series from the original SGN solution and blue lines represents the results of the improved SGN solution ($\delta_{Nwogu} = 0.17$)

Appendix D

Numerical Discretization of the 2D SGN equations

D.1 Governing Equations

We consider the following system of 2D SGN equations, which describes the incompressible homogeneous fluid flow in a layer bounded from below by the impermeable bottom $y = -h(x, y, t)$ and above by the free surface $y = \eta(x, y, t)$:

$$\eta_t + \nabla.\left[(\eta + h)\, U\right] = 0, \tag{D.1}$$

$$U_t + (U.\nabla)\, U + h\nabla\eta = \frac{\nabla\wp - \varrho\nabla h}{H}, \tag{D.2}$$

$$\nabla.\left[\frac{\nabla\wp}{H} - \frac{\left(\nabla\wp.\nabla h\right)\nabla h}{HY}\right] - 6\left[\frac{2}{H^3}.\frac{Y-3}{Y} + \nabla.\left(\frac{\nabla h}{H^2 Y}\right)\right]\wp = \mathscr{F}. \tag{D.3}$$

We rewrite the elliptic equation with directional derivatives:

$$\left[\frac{\wp_x}{H} - \frac{\left(\wp_x h_x + \wp_y h_y\right) h_x}{HY}\right]_x + \left[\frac{\wp_y}{H} - \frac{\left(\wp_x h_x + \wp_y h_y\right) h_y}{HY}\right]_y \tag{D.4}$$

$$-6\left[\frac{2}{H^3}.\frac{Y-3}{Y} + \left[\frac{h_x}{H^2 Y}\right]_x + \left[\frac{h_y}{H^2 Y}\right]_y\right]\wp = \mathscr{F}, \tag{D.5}$$

where: $Y = 4 + |\nabla h|^2 = 4 + h_x^2 + h_y^2$

$\mathscr{F}$ and $\mathscr{R}$ are defined for non-moving topography as:

$$\begin{aligned}\mathscr{F} &= \nabla.\left[g\nabla\eta + \frac{\mathscr{R}\nabla h}{Y}\right] - \frac{6\mathscr{R}}{HY} + 2\left(\nabla.\mathscr{U}\right)^2 - 2\begin{vmatrix} u_x & u_y \\ v_x & v_y \end{vmatrix} \\ &= \left[g\eta_x + \frac{\mathscr{R} h_x}{Y}\right]_x + \left[g\eta_y + \frac{\mathscr{R} h_y}{Y}\right]_y - \frac{6\mathscr{R}}{HY} + 2\left(u_x + v_y\right)^2 - 2\left(u_x v_y + u_y v_x\right),\end{aligned} \tag{D.6}$$

$$\mathscr{R} = -g\nabla\eta.\nabla h + \left[\left(\mathscr{U}.\nabla\right)\nabla h\right].\mathscr{U} \tag{D.7}$$

$$= -g\left(\eta_x h_x + \eta_y h_y\right) + u^2 h_{xx} + 2uvh_{xy} + v^2 h_{yy}. \tag{D.8}$$

Finally, The non-hydrostatic pressure at the bottom $\varrho(x, y, t)$ can be expressed through $\wp(x, y, t)$ in the following way:

$$\begin{aligned}\varrho(x, y, t) &= \frac{1}{Y}\left[\frac{6\wp}{H} + H\mathscr{R} + \nabla\wp.\nabla h\right] \\ &= \frac{1}{Y}\left[\frac{6\wp}{H} + H\mathscr{R} + \wp_x h_x + \wp_y h_y\right].\end{aligned} \tag{D.9}$$

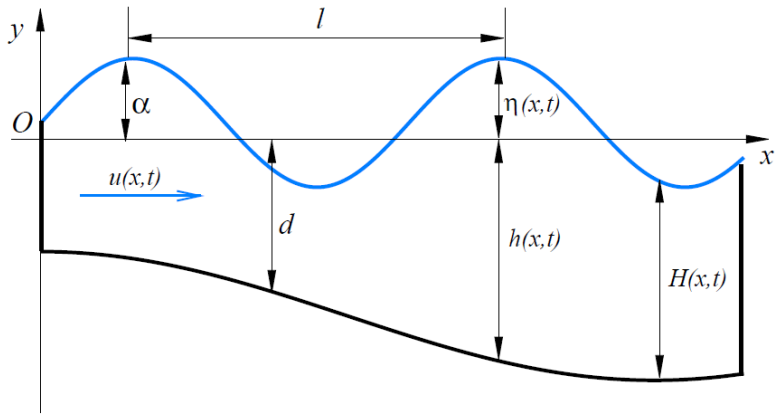


Figure D.1: Definition sketch.

D.2 Elliptic Equation Discretization

The advantage of this form of SGN equations is that the SWE is a subset of the problem and can be independently solved with efficient Shallow Water solvers.

Here we present the numerical solution of the elliptic solver on a staggered grid. To simply the discretization, we introduce the variables $\mathscr{A}$, $\mathscr{B}$ and $\mathscr{C}$ and we write the elliptic problem as:

$$\mathscr{A}_x + \mathscr{B}_y - \mathscr{C} = \mathscr{F}, \tag{D.10}$$

where:

$$\mathscr{A} = \frac{\wp_x}{H} - \frac{\left(\wp_x h_x + \wp_y h_y\right) h_x}{HY} = \frac{Y - h_x^2}{HY}\wp_x - \frac{h_x h_y}{HY}\wp_y, \tag{D.11}$$

$$\mathscr{B} = \frac{\wp_y}{H} - \frac{\left(\wp_x h_x + \wp_y h_y\right) h_y}{HY} = \frac{Y - h_y^2}{HY}\wp_y - \frac{h_x h_y}{HY}\wp_x, \tag{D.12}$$

$$\mathscr{C} = 6\left[\frac{2}{H^3}.\frac{Y-3}{Y} + \left[\frac{h_x}{H^2 Y}\right]_x + \left[\frac{h_y}{H^2 Y}\right]_y.\right]\wp \tag{D.13}$$

We integrate Eq. (D.10) over one grid cell $v_i = \left[x_{i-\frac{1}{2},j}, x_{i+\frac{1}{2},j}\right] \times \left[y_{i,j-\frac{1}{2}}, y_{i,j+\frac{1}{2}}\right]$:

$$\iint_{v_i} \mathscr{A}_x \, dx\, dy + \iint_{v_i} \mathscr{B}_y \, dx\, dy - \iint_{v_i} \mathscr{C} \, dx\, dy = \iint_{v_i} \mathscr{F} \, dx\, dy, \tag{D.14}$$

To simplify the notation we introduce the variables

$$\alpha = \frac{Y - h_x^2}{HY}, \qquad \beta = \frac{Y - h_y^2}{HY}, \qquad \gamma = -\frac{h_x h_y}{HY}. \tag{D.15}$$

We discretize the elliptic problem term by term:

$$\begin{aligned}\iint_{v_i} \mathscr{A}_x \, dx \, dy &= \Delta x \Delta y \left(\frac{\left[\alpha \wp_x\right]_{i+\frac{1}{2},j} - \left[\alpha \wp_x\right]_{i-\frac{1}{2},j}}{\Delta x} + \frac{\left[\gamma \wp_y\right]_{i+1,j} - \left[\gamma \wp_y\right]_{i-1,j}}{2\Delta x} \right) \\ &= \frac{\Delta y \left(\alpha_{i+1,j} + \alpha_{i,j}\right)}{2\Delta x} \wp_{i+1,j} - \frac{\Delta y \left(\alpha_{i+1,j} + 2\alpha_{i,j} + \alpha_{i-1,j}\right)}{2\Delta x} \wp_{i,j} + \frac{\Delta y \left(\alpha_{i,j} + \alpha_{i-1,j}\right)}{2\Delta x} \wp_{i-1,j} \\ &+ \frac{\gamma_{i+1,j}}{4} \wp_{i+1,j+1} - \frac{\gamma_{i+1,j}}{4} \wp_{i+1,j-1} - \frac{\gamma_{i-1,j}}{4} \wp_{i-1,j+1} + \frac{\gamma_{i-1,j}}{4} \wp_{i-1,j-1}.\end{aligned} \tag{D.16}$$

$$\begin{aligned}\iint_{v_i} \mathscr{B}_y \, dx \, dy &= \Delta x \Delta y \left(\frac{\left[\beta \wp_y\right]_{i,j+\frac{1}{2}} - \left[\beta \wp_y\right]_{i,j-\frac{1}{2}}}{\Delta y} + \frac{\left[\gamma \wp_x\right]_{i,j+1} - \left[\gamma \wp_x\right]_{i,j-1}}{2\Delta y} \right) \\ &= \frac{\Delta x \left(\beta_{i,j+1} + \beta_{i,j}\right)}{2\Delta y} \wp_{i,j+1} - \frac{\Delta x \left(\beta_{i,j+1} + 2\beta_{i,j} + \beta_{i,j-1}\right)}{2\Delta y} \wp_{i,j} + \frac{\Delta x \left(\beta_{i,j} + \beta_{i,j-1}\right)}{2\Delta y} \wp_{i,j-1} \\ &+ \frac{\gamma_{i,j+1}}{4} \wp_{i+1,j+1} - \frac{\gamma_{i,j+1}}{4} \wp_{i-1,j+1} - \frac{\gamma_{i,j-1}}{4} \wp_{i+1,j-1} + \frac{\gamma_{i,j-1}}{4} \wp_{i-1,j-1}.\end{aligned} \tag{D.17}$$

$$\begin{aligned}\iint_{v_i} \mathscr{C} \, dx \, dy = \Bigg\{ &\frac{12\Delta x \Delta y}{H_{i,j}^3} . \frac{Y_{i,j} - 3}{Y_{i,j}} + 3\Delta y \left(\left[\frac{h_x}{H^2 Y}\right]_{i+1,j} - \left[\frac{h_x}{H^2 Y}\right]_{i-1,j} \right) \\ &+ 3\Delta x \left(\left[\frac{h_y}{H^2 Y}\right]_{i,j+1} - \left[\frac{h_y}{H^2 Y}\right]_{i,j-1} \right) \Bigg\} \wp_{i,j}.\end{aligned} \tag{D.18}$$

We define the variable $\mathscr{K}$ such as:

$$\iint_{v_i} \mathscr{C} \, dx \, dy = \mathscr{K}_{i,j} \wp_{i,j}. \tag{D.19}$$

Let's consider N and M the numbers of grids in the x- and y- directions, respectively.
The discretization of the elliptic problem Eq. (D.3) can be expressed with a 9-point stencil in the inner cells of the domain (*i.e.*, $i = 1, ..., N-2$ and $j = 1, ..., M-2$):

$$\begin{aligned}a_{ij} \wp_{i+1,j+1} + b_{ij} \wp_{i+1,j} + c_{ij} \wp_{i+1,j-1} + d_{ij} \wp_{i,j+1} + e_{ij} \wp_{i,j} + f_{ij} \wp_{i,j-1} \\ + g_{ij} \wp_{i-1,j+1} + h_{ij} \wp_{i-1,j} + k_{ij} \wp_{i-1,j-1} = \mathscr{F}_{i,j},\end{aligned} \tag{D.20}$$

where:

$$a_{ij} = \frac{\gamma_{i+1,j} + \gamma_{i,j+1}}{4}, \tag{D.21}$$

$$b_{ij} = \frac{\Delta y \left(\alpha_{i+1,j} + \alpha_{i,j}\right)}{2\Delta x}, \tag{D.22}$$

$$c_{ij} = -\frac{\gamma_{i+1,j} + \gamma_{i,j-1}}{4}, \tag{D.23}$$

$$d_{ij} = \frac{\Delta x \left(\beta_{i,j+1} + \beta_{i,j}\right)}{2\Delta y}, \tag{D.24}$$

$$e_{ij} = -\frac{\Delta y \left(\alpha_{i+1,j} + 2\alpha_{i,j} + \alpha_{i-1,j}\right)}{2\Delta x} - \frac{\Delta x \left(\beta_{i,j+1} + 2\beta_{i,j} + \beta_{i,j-1}\right)}{2\Delta y} - \mathscr{K}_{i,j}, \tag{D.25}$$

$$f_{ij} = \frac{\Delta x \left(\beta_{i,j} + \beta_{i,j-1}\right)}{2\Delta y}, \tag{D.26}$$

$$g_{ij} = -\frac{\gamma_{i-1,j} + \gamma_{i,j+1}}{4}, \tag{D.27}$$

$$h_{ij} = \frac{\Delta y \left(\alpha_{i,j} + \alpha_{i-1,j}\right)}{2\Delta x}, \tag{D.28}$$

$$k_{ij} = \frac{\gamma_{i-1,j} + \gamma_{i,j-1}}{4}. \tag{D.29}$$

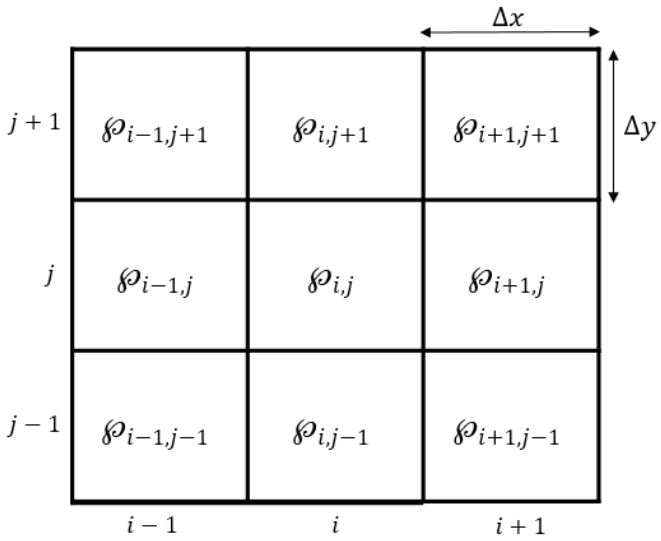


Figure D.2: Stencil dependency in the elliptic solver

System (D.20) may be written in matrix form as follows:

$$Ax = b, \tag{D.30}$$

The unknown vector x contains all depth-averaged non-hydrostatic pressures in the numerical cells $\wp_{i,j}$.

D.3 Reflective Boundary condition

We extend the 1D boundary conditions described in Khakimzyanov et al. [2017] to 2D. We consider a Cartesian computational domain $\mathcal{D} = [0, L_x] \times [0, L_y]$. The following mixed-type boundary conditions can be applied to the elliptic problem:

$$\left[\frac{\wp_x - \varrho h_x}{H} - g\eta_x\right]_{x=0}^{x=L_x} = 0, \tag{D.31}$$

$$\left[\frac{\wp_y - \varrho h_y}{H} - g\eta_y\right]_{y=0}^{y=L_y} = 0. \tag{D.32}$$

These boundary conditions are obtained from the momentum equations by setting both the velocity and the velocity gradient to zero at the boundaries. By replacing the value of ϱ (Eq. (D.9)) into the boundary conditions, we get:

$$\left[\frac{\wp_x}{H} - \frac{\left(\wp_x h_x + \wp_y h_y\right) h_x}{HY} - 6\frac{h_x}{H^2 Y}\wp\right]_{x=0}^{x=L_x} = \left[g\eta_x + \frac{\mathscr{R} h_x}{Y}\right]_{x=0}^{x=L_x}, \tag{D.33}$$

$$\left[\frac{\wp_y}{H} - \frac{\left(\wp_x h_x + \wp_y h_y\right) h_y}{HY} - 6\frac{h_y}{H^2 Y}\wp\right]_{y=0}^{y=L_y} = \left[g\eta_y + \frac{\mathscr{R} h_y}{Y}\right]_{y=0}^{y=L_y}. \tag{D.34}$$

Appendix E

Empirical Wave Spectra

E.1 Pierson-Moskowitz Spectrum

Various empirically-derived spectra are used to represent ocean waves. One of the simple approaches is the one proposed by Pierson Jr and Moskowitz [1964]. They assumed that for a steady wind field over a long time and a large area, the waves would come into equilibrium with the wind. Under this assumption, they derived the wave spectrum for the fully-developed sea as:

$$S_{\mathrm{PM}}(\omega) = \frac{\alpha g^2}{\omega^5} \exp\left(-\beta \left(\frac{\omega_0}{\omega}\right)^4\right), \tag{E.1}$$

where $\alpha = 0.0081$ is the Phillips constant. $\beta = 0.74$, $\omega_0 = g/U_{19.5}$, and $U_{19.5}$ is the wind speed at a height of 19.5 m above the sea surface. For practical reasons, it has also become common to relate the variables to the peak frequency f_p instead of the wind speed:

$$S_{\mathrm{PM}}(f) = \alpha g^2 (2\pi)^{-4} f^{-5} \exp\left(-\frac{5}{4}\left(\frac{f_p}{f}\right)^4\right). \tag{E.2}$$

We can also express the significant wave height H_s in terms of peak frequency f_p:

$$H_s = \frac{0.162 g}{(2\pi f_p)^2}. \tag{E.3}$$

This leads to the following two-parameter Pierson-Moskowitz spectrum (*i.e.*, Bretschneider spectrum):

$$S_{\mathrm{B}}(f) = \frac{5}{16}\frac{f_p^4}{f^5} H_s^2 \exp\left(-\frac{5}{4}\left(\frac{f_p}{f}\right)^4\right). \tag{E.4}$$

E.2 JONSWAP Spectrum

Hasselmann et al. [1973], after analyzing data collected during the Joint North Sea Wave Observation Project (JONSWAP), found that the wave spectrum is never fully developed. It continues to develop through non-linear, wave-wave interactions over very long times and distances. Hence, an extra and somewhat artificial factor was added to the Pierson-Moskowitz spectrum in order to improve the fit of their measurements. The JONSWAP spectrum is thus a Pierson-Moskowitz spectrum multiplied by an extra peak enhancement factor γ^r :

$$S_{\mathrm{J}}(f) = S_{\mathrm{PM}}(f)\,\gamma^r, \tag{E.5}$$

$$S_{\mathrm{J}}(f) = \alpha g^2 (2\pi)^{-4} f^{-5} \exp\left(-\frac{5}{4}\left(\frac{f_p}{f}\right)^4\right)\gamma^r, \tag{E.6}$$

where

$$r = \exp\left(-\frac{1}{2}\left(\frac{f - f_p}{\sigma f_p}\right)^2\right). \tag{E.7}$$

Wave data collected during the JONSWAP experiment was used to determine the values for the constants in the above equations:

$$\gamma = 3.3, \qquad \sigma = \begin{cases} 0.07, & f \leqslant f_p. \\ 0.09, & f > f_p. \end{cases} \tag{E.8}$$

E.3 TMA Spectrum

The JONSWAP spectrum was originally designed for developing seas in deep water. However, the generation of waves in shallow areas is affected by the limited water depth. Hughes [1984] proposed the following modified JONSWAP spectrum in shallow water (*i.e.*, TMA spectrum). It is based on the fact that low-frequency, or long-period, waves in shallow water have a restricted wave height. Therefore, the spectrum is multiplied by a function for limited depth:

$$S_{TMA}(f) = S_J(f)\,\Phi(2\pi f, h), \tag{E.9}$$

$$S_{TMA}(f) = \alpha g^2 (2\pi)^{-4} f^{-5} \Phi(2\pi f, h) \exp\left(-\frac{5}{4}\left(\frac{f_p}{f}\right)^4\right)\gamma^r, \tag{E.10}$$

where:

$$\Phi(2\pi f, h) = \begin{cases} 0.5\left(2\pi f\sqrt{h/g}\right)^2, & 2\pi f\sqrt{h/g} < 1. \\ 1 - 0.5\left(2 - 2\pi f\sqrt{h/g}\right)^2, & 1 \leqslant 2\pi f\sqrt{h/g} < 2. \\ 1, & 2\pi f\sqrt{h/g} \geqslant 2. \end{cases} \tag{E.11}$$

E.4 Illustration

We plot the three empirical spectra for the following wave parameters: $T_p = 20\,\mathrm{s}$ and $h = 20\,\mathrm{m}$:

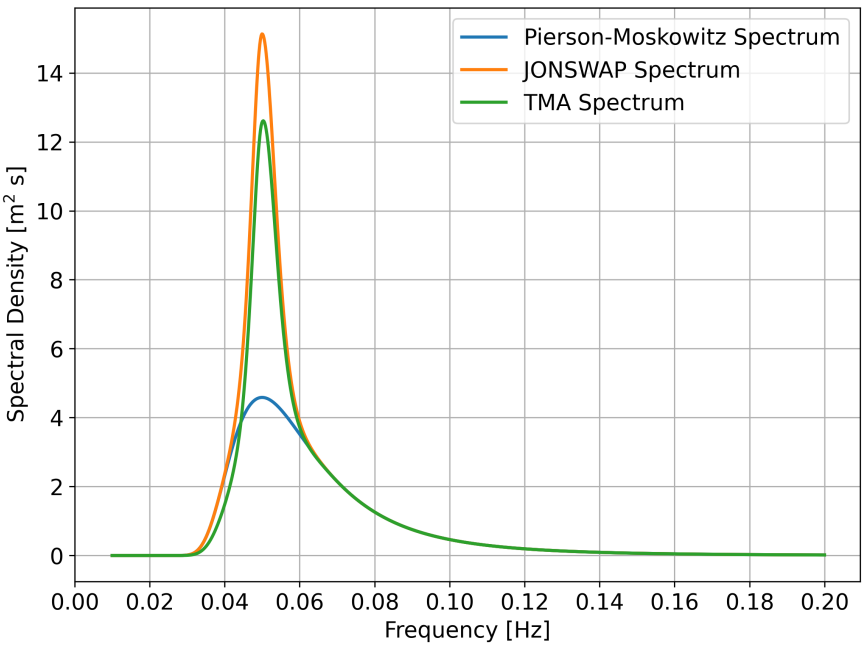


Figure E.1: Empirical spectra comparison