



HAL
open science

New nested grids technique for 2D shallow water equations

Huda Altaie

► **To cite this version:**

Huda Altaie. New nested grids technique for 2D shallow water equations. Numerical Analysis [math.NA]. Université Côte d'Azur, 2018. English. NNT : 2018AZUR4220 . tel-02271260

HAL Id: tel-02271260

<https://theses.hal.science/tel-02271260>

Submitted on 26 Aug 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Nouvelle Technique de Grilles Imbriquées pour les équations de Saint-Venant 2D

New Nested Grids Technique For 2D Shallow Water Equations

Huda Omran ALTAIE

Laboratoire Jean Alexandre Dieudonné LJAD

**Présentée en vue de l'obtention
du grade de docteur en Science**
Mathématiques de l'Université cote d'Azur

**Dirigée par : Fabrice Planchon / Pierre
Dreyfuss**

Soutenue le : 17/12 2018

Devant le jury, composé de :

Xavier Antoine, Professeur- Université de Lorraine,
Rapporteur
Stéphane Clain, Associate Professeur HDR- Université
de Minho (Portugal), Rapporteur
Isabelle Gallagher, Professeur- Ecole Normale
Supérieure de Paris, Examinatrice
Marjolaine Puel, Professeur- Université Nice Sophia-
Antipolis, Examinatrice
Fabrice Planchon, Professeur-Université Nice Sophia-
Antipolis, Directeur de thèse
Pierre Dreyfuss, Maître de conférences-Université Nice
Sophia-Antipolis, Co-Directeur de thèse

ABSTRACT

Most flows in the rivers, seas, and ocean are shallow water flow in which the horizontal length and velocity scales are much larger than the vertical ones. The mathematical formulation of these flows, so-called shallow water equations (SWEs). These equations are a system of hyperbolic partial differential equations and they are effective for many physical phenomena in the oceans, coastal regions, rivers and canals.

This thesis focuses on the design of a new two-way interaction technique for multiple nested grids 2DSWEs using the numerical methods. The first part of this thesis includes, proposing several ways to develop the derivation of shallow water model. The complete derivation of this system from Navier-Stokes equations is explained. Studying the development and evaluation of numerical methods by suggesting new spatial and temporal discretization techniques in a standard C-grid using an explicit finite difference method in space and leapfrog with Robert-Asselin filter in time which are effective for modeling in oceanic and atmospheric flows. Several numerical examples for this model using Gaussian level initial condition are implemented in order to validate the efficiency of the proposed method.

In the second part of our work, we are interested to propose a new two-way interaction technique for multiple nested grids to solve ocean models using four choices of higher restriction operators (update schemes) for the free surface elevation and velocities with high accuracy results. Our work focused on the numerical resolution of SWEs by nested grids. At each level of resolution, we used explicit finite differences methods on Arakawa C-grid. In order to be able to refine the calculations in troubled regions and move them into quiet areas, we have considered several levels of resolution using nested grids. This makes it possible to considerably increase the performance ratio of the method, provided that the interactions (spatial and temporal) between the grids are effectively controlled.

In the third part of this thesis, several numerical examples are tested to show and verify two-way interaction technique for multiple nested grids of shallow water models can works efficiently over different periods of time with nesting 3:1 and 5:1 at multiple levels. Some examples for multiple nested grids of the tsunami model with nesting 5:1 using moving boundary conditions are tested in the fourth part of this work.

Keywords: Nested grid, One-way nesting, Two-way nesting, Mesh refinement, Explicit finite difference method, Shallow water model, Derivation of 2D shallow water equations, Ocean model, Boundary conditions, Optimum feedback, Free surface flows, Hydrostatic pressure, Coastal regions, Incompressible fluid, Fluid dynamics, Tsunami deposit, Tsunami hazards, Dynamical interface.

Acknowledgment

First and foremost, I would like to express my sincere gratitude to my supervisors Prof. Fabrice Planchon and Mr. Pierre Dreyfuss for the continuous support of my Ph.D. study and research, for their patience, motivation, enthusiasm, and immense knowledge. Their guidance helped me in all the time of research and writing of this thesis. Also, for their confidence and constant encouragement, for their enthusiasm, for their many tips. I could not have imagined having excellent supervisors and mentors for my Ph.D. study.

I am aware of how lucky I was to have the opportunity to work with you. Thank you for your kindness towards me throughout these years and you have no doubt to make the path of research more passable and easier thank you Fabrice and Pierre.

Besides my supervisors, I would like to show my gratitude for the honorable jury members who accepted to examine my project: Prof. Xavier Antoine, Prof. Stéphane Clain, Prof. Isabelle Gallagher, and Prof. Marjolaine Puel.

Also, my special thanks goes to the Ministry of Higher Education in Iraq which provided to me the scholarship for my doctoral studies and for the financial support over the past four years.

Special thanks to Prof. Fadhel Subhi, Associate Dean, Al-Nahrain University and my friend in Iraq Dr. Butheynah Najad for their continued support me.

I would like to thank my friend Jiqiang Zheng and all my friends in laboratory J.A Dieudonné. I likewise thank the laboratory J.A Dieudonné for hosting me over the past four years. In a special way, I thank Chiara, Angélique, Victoria, Isabelle, Valérie, Jean-Marc, Roland, and Julia for their friendship and their valuable technical and administrative assistance. Special thanks to Prof. Michel Jambu.

Finally, my greatest love and gratitude goes to my family who gave me everything possible to enable me reach higher education levels. I only hope that they know how their love, support, and patience encouraged me to fulfill my and their dreams.

HUDA

List of Figures

Figure1-1	Shallow water equations in canals.	40
Figure1-2	Shallow water equations in dam break.	40
Figure1-3	Depth-averaged velocity distribution.	48
Figure2-1	Finite difference method.	70
Figure2-2	Arakawa staggered C-grid.	71
Figure2-3	Finite difference method in Arakawa C-grid.	72
Figure2-4	The standard Robert-Asselin filter.	73
Figure2-5	Show organization chart of the calculation program	91
Figure2-6	The numerical and physical domain of dependence.	93
Figure3-1	Comparison $l2-RE$ of free surface elevation, u -velocity and v -velocity for 2DNSWEs at $t = 10, 20, \dots, 100$ days.	105
Figure3-2	Simulation of free surface elevation at time $t = 100, 200, \dots, 1000$ hours.	108
Figure3-3	$l2-RE$ of free surface elevation for linear SWEs at $t = 10, 20, \dots, 1000$ hours.	109
Figure3-4	Comparison $l2-RE$ of free surface elevation, u -velocity and v -velocity for linear SWEs at $t = 10, 20, \dots, 100$ days.	109
Figure3-5	Comparison the approximate solution for free surface elevation, u -velocity and v -velocity for linear SWEs.	110
Figure3-6	Comparison $l2-RE$ of free surface elevation, u -velocity and v -velocity for linear SWEs.	111
Figure3-7	Comparison the approximate solution for free surface elevation, u -velocity and v -velocity for nonlinear SWEs.	111
Figure3-8	Comparison $l2-RE$ of free surface elevation, u -velocity and v -velocity for nonlinear SWEs.	112
Figure4-1	Shows Structured grid.	124
Figure4-2	Shows unstructured grid.	124
Figure4-3	Shows hybrid grid.	125
Figure4-4	Embedded nested domain.	126

Figure4-5	Overlapping grid configuration.	126
Figure4-6	Adaptive mesh refinement.	127
Figure4-7	Notations used in the definitions of the nested models.	131
Figure4-8	Copy interpolation scheme.	133
Figure4-9	Average interpolation scheme.	134
Figure4-10	Average interpolation scheme.	134
Figure4-11	Shapiro interpolation scheme.	135
Figure4-12	Full-weighting interpolation scheme.	136
Figure4-13	Full-weighting interpolation scheme.	136
Figure4-14	Detailed view of nested grid. Left panel: grid nesting at lower-left corner of sub-level grid region; Right panel: grid nesting at upper-right corner of sub-level grid region.	137
Figure4-15	Simulation involving two-way nesting.	140
Figure4-16	Simulation involving two-way nesting.	142
Figure4-17	Two-way nesting	145
Figure4-18	Embedding the fine grid within the coarse grid.	149
Figure4-19	Diagram for open boundary condition.	154
Figure4-20	Diagram for two way coupling.	155
Figure5-1	Simulation of free surface elevation in a coarse grid at time=1000 min.	162
Figure5-2	Simulation of free surface elevation in a coarse grid at time=2000 min.	163
Figure5-3	Simulation of free surface elevation in a coarse grid at time=3000 min.	163
Figure5-4	Simulation of free surface elevation in a finer grid at time=1000 min.	164
Figure5-5	Simulation of free surface elevation in a finer grid at time=2000 min.	164
Figure5-6	Simulation of free surface elevation in a finer grid at time=3000 min.	164
Figure5-7	Simulation of free surface elevation in a finer grid at time=2000 hour.	165
Figure5-8	Simulation of free surface elevation in two-way nesting grids at time=1000 min.	165
Figure5-9	Simulation of free surface elevation in two-way nesting grids at time=2000 min.	166
Figure5-10	Comparison the approximate solution for free surface elevation between the coarse and the fine grids at $t = 100$ min, 200 min, ..., 2000 min.	166
Figure5-11	Simulation of free surface elevation at $t = 10$ min.	167
Figure5-12	Simulation of free surface elevation at $t = 50$ min.	168
Figure5-13	Simulation of free surface elevation at $t = 100$ min.	168
Figure5-14	Simulation of free surface elevation at $t = 500$ hr.	168
Figure5-15	Simulation of free surface elevation at $t = 1000$ hr.	169

Figure5-16	Comparison $l2$ -error norm and $H1$ -error norm in a coarse grid.	169
Figure5-17	Comparison $H1$ -error norm and $l2$ -error norm in a fine grid.	169
Figure5-18	Comparison $H1$ -error norm in a coarse grid, and a fine grid for linear SWEs. . .	170
Figure5-19	Comparison $l2$ -error norm in a coarse grid, and a fine grid for linear SWEs. . .	170
Figure5-20	Show $l2$ -error norm and $H1$ -error norm in a coarse grid.	171
Figure5-21	Comparison $l2$ -error norm and $H1$ -error norm in a fine grid at $t=10, 20, \dots, 100$ days.	172
Figure5-22	Show $l2$ -error norm in a coarse grid, and a fine grid	172
Figure5-23	Show $H1$ -error norm in a coarse grid, and a fine grid	172
Figure5-24	$l2$ - RE of free surface elevation between a coarse grid and a fine grid in two-way nested grid.	174
Figure5-25	Comparison $l2$ - RE between a coarse grid and a fine grid when use four choices of restriction operator for the free surface elevation	174
Figure5-26	Comparison $l2$ - RE between a coarse grid and a fine grid when use four choices of restriction operator for free surface elevation	175
Figure5-27	Comparison $l2$ - RE between a coarse grid and a fine grid when use four choices of restriction operator for free surface elevation	175
Figure5-28	Comparison $l2$ - RE between a coarse grid and a fine grid when use four choices of restriction operator for free surface elevation	176
Figure5-29	Comparison between ABSE and $l2$ - RE of free surface elevation in one-way nesting grid when structured grid with a separate interface	177
Figure5-30	Comparison between ABSE and $l2$ - RE of free surface elevation in two-way nesting grid when structured grid with a separate interface	177
Figure5-31	Comparison $l2$ - RE of free surface elevation in two-way nesting for structured grid with a separate interface	178
Figure6-1	Multiple nested grids at multiple levels	188
Figure6-2	Shows $l2$ - RE of free surface elevation between a coarse grid and a in fine grid (grid 21)	189
Figure6-3	Show $l2$ - RE of free surface elevation between a coarse grid and a fine grid (grid 22)	190
Figure6-4	Comparison $l2$ - RE between a coarse grid and two fine grids in level 2	190
Figure6-5	$l2$ - RE between the grid 21 and the grid 31 in two-way nested grid	191
Figure6-6	Comparison $l2$ - RE between a coarse grid-grid 21 and grid 21-grid 31	192
Figure6-7	Simulation of free surface elevation in the coarse grid at time= 500 min	193
Figure6-8	Simulation of free surface elevation in the coarse grid at time= 1000 min	193

Figure6-9	Simulation of free surface elevation in the coarse grid at time= 2000 min	193
Figure6-10	Comparison <i>l2-RE</i> of free surface in two-way nested grid at t=500, 1000,...., 4000 hour	194
Figure6-11	Comparison <i>l2-RE</i> of free surface elevation between the coarse grid and two fine grids in level 2	194
Figure6-12	Shows <i>l2-RE</i> of free surface elevation between the grid 21 and the grid 31	195
Figure6-13	Compares <i>l2-RE</i> between a coarse grid-fine grid 22 (level 2) and fine grid 22-fine grid 32 in level 3	196
Figure6-14	Compares <i>l2-RE</i> between a coarse grid-fine grid 21 in level 2 and fine grid 21-fine grid 31 in level 3	196
Figure6-15	Comparison <i>l2-RE</i> between a coarse grid-grid 22 and grid 22-grid 32 for nonlinear SWEs	197
Figure6-16	Comparison <i>l2-RE</i> of free surface elevation between a coarse grid-grid 22 for nonlinear case and grid 22-grid 32 for nonlinear to linear	198
Figure6-17	Comparison <i>l2-RE</i> between a coarse grid-a fine grid in level 2 for nonlinear to linear and a fine grid in level 2-fine grid in level 3 for linear to nonlinear SWEs	198
Figure6-18	Comparison <i>l2-RE</i> in one-way nesting grid and two-way nesting grid using the average scheme	199
Figure6-19	Comparison <i>l2-RE</i> of the free surface elevation in one-way nesting and two-way nesting using Shapiro interpolation scheme	200
Figure6-20	Comparison <i>l2-RE</i> in one-way and two-way nesting using the full-weighting scheme	200
Figure6-21	Compares <i>l2-RE</i> in two-way nesting using the average scheme and the full- weighting scheme	201
Figure6-22	Compares <i>l2-RE</i> in two-way nesting using the average scheme and the full- weighting scheme	201
Figure6-23	ABSE and <i>l2-RE</i> of free surface elevation in one-way nesting	202
Figure6-24	ABSE and <i>l2-RE</i> of free surface elevation in two-way nesting	203
Figure6-25	ABSE between the coarse grid and the fine grid for linear 2DSWEs	203
Figure6-26	Comparison <i>l2-RE</i> and ABSE in two-way nesting grids	204
Figure6-27	<i>l2-RE</i> of free surface elevation in one-way nesting grid for linear SWEs	205
Figure6-28	<i>l2-RE</i> of free surface elevation in two-way nesting grid for linear SWEs	205
Figure6-29	<i>l2-RE</i> of free surface elevation in one-way nesting grid for nonlinear SWEs	205
Figure6-30	<i>l2-RE</i> of free surface elevation in two-way nesting grid for nonlinear SWEs	206

Figure6-31 <i>l2-RE</i> in two-way nesting using four different update schemes without separate dynamic interface and feedback interface	207
Figure6-32 <i>l2-RE</i> in two-way nesting using four different update schemes with separate dynamic interface and feedback interface	207
Figure6-33 Free surface elevation on the coarse grid domain using average method.	208
Figure6-34 Free surface elevation on the coarse grid domain using mix-low method	209
Figure6-35 Free surface elevation on the coarse grid domain using full-weighting method	210
Figure6-36 Free surface elevation on the coarse grid domain using mix-high method	211
Figure6-37 Comparison the ABSE between a coarse grid and a fine grid in level 2	212
Figure6-38 Comparison the ABSE between level 2 and level 3	212
Figure6-39 Comparison <i>l2-RE</i> between a coarse grid-a fine grid (level 2) and level 2-level 3	213
Figure6-40 Comparison <i>l2-RE</i> when use four choice for update schemes with a separate interface	213
Figure6-41 Comparison <i>l2-RE</i> when use four choice for update schemes without a separate interface	214
Figure7-1 Comparison <i>l2-RE</i> between a coarse grid- fine grid in level 2 and level 2-level 3 in case space refinement ratio is 1:5	221
Figure7-2 Comparison <i>l2-RE</i> between a coarse grid in level 1-fine grid in level 2 and fine grid in level 2-fine grid in level 3 in case the space refinement ratio is 1:3	221
Figure7-3 Comparison <i>l2-RE</i> between a coarse grid in level 1-fine grid in level 2 and fine grid in level 2-fine grid in level 3 in case 1:5	222
Figure7-4 Comparison <i>l2-RE</i> between a coarse grid in level 1-fine grid in level 2 and fine grid in level 2-fine grid in level 3 in case 1:3	222
Figure7-5 Comparison <i>l2-RE</i> and ABSE in one-way nesting in case 1:4	223
Figure7-6 Comparison <i>l2-RE</i> and ABSE in two-way nesting in case 1:5	223
Figure7-7 Comparison ABSE between a coarse grid in level 1-fine grid in level 2 and fine grid in level 2-fine grid in level 3	224
Figure7-8 Comparison <i>l2-RE</i> between a coarse grid in level 1-fine grid in level 2 and fine grid in level 2-fine grid in level 3 for 2DNSWEs	225
Figure7-9 Comparison <i>l2-RE</i> and ABSE for one-way nesting in case 1:3	225
Figure7-10 Comparison <i>l2-RE</i> and ABSE for two-way nesting in case 1:3	226
Figure7-11 Comparison <i>l2-RE</i> and ABSE for one-way nesting in case 1:4	226
Figure7-12 Comparison <i>l2-RE</i> and ABSE for two-way nesting in case 1:5	226
Figure7-13 Comparison <i>l2-RE</i> of free surface elevation , <i>u</i> -velocity and <i>v</i> -velocity	227
Figure8-1 Multiple nested grid at multiple levels for 2DNSWEs	238

Figure8-2 Comparison <i>l2-RE</i> of free surface between a coarse grid and level 2 (grid 21 and grid 24)	238
Figure8-3 Comparison <i>l2-RE</i> between a coarse grid and level 2 (grid 22 and grid 23)	238
Figure8-4 Comparison <i>l2-RE</i> between coarse grid-level 2 (grid 21), level 2-level 3 (grid 32) and level 3-level 4 (grid 43)	239
Figure8-5 Comparison <i>l2-RE</i> between a coarse grid-level 2 (grid 24), level 2-level 3 (grid 33), and level 3-level 4 (grid 46)	240
Figure8-6 Comparison <i>l2-RE</i> of free surface elevation between a coarse grid-level 2 (grid 23), level 2-level 3 (grid 34) and level 3-level 4 (grid 44)	241
Figure8-7 Comparison <i>l2-RE</i> between a coarse grid-level 2 (grid 23) and level 2-level 3 (grid 34) and level 3-level 4 (grid 42)	242
Figure8-8 Multiple nested grids at multiple levels for linear 2DSWEs	242
Figure8-9 Show <i>l2-RE</i> between a coarse grid and four fine grids (21, 22, 23 and 24)	243
Figure8-10 Comparison <i>l2-RE</i> of free surface elevation between a coarse grid-level 2 (grid 21), level 2-level 3 (grid 32) and level 3-level 4 (grid 43)	244
Figure8-11 Comparison <i>l2-RE</i> between a coarse grid-level 2 (grid 22), level 2-level 3 (grid 33) and level 3-level 4 (grid 44)	245
Figure8-12 Comparison <i>l2-RE</i> between a fine grid in level 2 and level 3 (grid 32) in cases separate and embedded grids.	246
Figure8-13 Comparison <i>l2-RE</i> for linear 2DSWEs when time step 0.010	247
Figure8-14 Comparison <i>l2-RE</i> for linear 2DSWEs when time step 0.030	247
Figure8-15 Comparison <i>l2-RE</i> between a coarse grid-level 2 (grid 24), level 2-level 3 (grid 31) and level 3-level 4 (grids 41)	248
Figure8-16 Comparison <i>l2-RE</i> of free surface elevation between a coarse grid-level 2 (grid 24), level 2-level 3 (grid 31) and level 3-level 4 (grids 45) embedded with itself	249
Figure8-17 Comparison <i>l2-RE</i> between level 1-level 2, level 2-level 3 and level 3-level 4	250
Figure8-18 Comparison <i>l2-RE</i> between level 1-level 2 in case nonlinear, level 2-level 3 in case nonlinear to linear and level 3-level 4 in case linear to nonlinear SWEs.	250
Figure8-19 Comparison <i>l2-RE</i> between a coarse grid-fine grid in level 2 nonlinear to linear case, fine grid in level 2-fine grid in level 3 linear to nonlinear case and level 3-fine grid in level 4 nonlinear to linear case.	251
FigureA-1 A sketch of moving boundary scheme	268

List of Tables

Table4.1	Shows the restriction operator for free surface elevation η and velocities (u, v) on Arakawa C-grid.	151
Table6.1	The information on the set up of the different grids for 2D non-linear SWEs . . .	189
Table6.2	The information about the coarse and fine grids for linear 2DSWEs	190
Table6.3	The information about the coarse and the fine grids at multiple levels for linear 2DSWEs	191
Table6.4	The information about the coarse and the fine grids at multiple levels for 2DNSWEs	195
Table6.5	The information about the coarse grid and the fine grids at multiple levels for the 2DNSWEs	197
Table7.1	The information about the coarse and fine grids at multiple regions (levels) for 2DNSWEs	220
Table8.1	The information on the set up of the different grids	237
Table8.2	The information on the set up of the different grids for 2DNSWEs	239
Table8.3	The information on the set up of the different grids at multiple level for 2DNSWEs	240
Table8.4	The information on the set up of the different grids at multiple levels	241
Table8.5	The information on the set up of the different grids at multiple levels for linear SWEs	243
Table8.6	The information on the set up of the different grids at multiple levels for linear 2DSWEs	244
Table8.7	The information on the set up of the different grids at multiple levels for linear SWEs	245
Table8.8	The information on the set up of different grids for linear SWEs	246
Table8.9	The information on the set up of the different grids at multiple levels for linear 2DSWEs	247
Table8.10	The information on the set up of the different grids at multiple levels	248
Table8.11	The information on the set up of the different grids at multiple levels for 2DNSWEs	249

Outline of This Work

The major purpose of our work is to design new technique for multiple nested grids of 2D non-linear shallow water equations (2DNSWEs) for structured grids using numerical schemes.

This thesis is organized into four parts with 9 chapters. The general framework of the various topics will be detailed in the parts and chapters that follow:

Part I: Mathematical and Numerical Model for 2D Depth-Integrated Shallow Water Equations

The principal objective of this part is to study the development of the efficiency of numerical methods for 2D depth-averaged non-linear shallow water models by proposing new spatial and temporal discretization techniques in a standard C-grid using an explicit finite difference method in space and leapfrog with Robert-Asselin filter in time. Firstly, a new technique to derive 2DSWEs is presented. Secondly, propose effective numerical methods, such as the explicit finite difference methods for solving ocean models. The different unknowns variables for the system are approximated on staggered grids and the numerical fluxes are computed with the proposed techniques.

This part consists of three chapters including:

Chapter 1: Derivation of 2D Shallow Water Equations

This chapter is devoted to the study of 2D non-linear shallow water equations. Firstly, presents a mathematical study of Navier-Stokes equations and 2DSWEs, which are obtained from a vertical integration of 3D Navier-Stokes equations by using a number of the assumptions. Secondly, we know that 2DSWEs can be derived in a number of ways with varying initial assumptions, we suggest two ways to expand the derivation of 2D depth-averaged SWEs from 3D Navier-Stokes equations using splitting of velocity and eddy dispersion coefficients. Another way is presented to derive 2DSWEs without using depth-averaged technique.

Chapter 2: An Explicit Staggered Finite Difference Scheme for 2D Shallow Water Equations (Numerical Techniques for 2DSWEs)

This chapter is a description of the development and evaluation of proposed numerical methods for 2D shallow water models, which depend on the choice of techniques and numerical methods used.

Chapter 3: Numerical Results for 2D Shallow Water Equations (Validations of the Model).

In order to validate the proposed method of 2D shallow water equations, some examples of the tsunami model are applied. Some examples for 2DSWEs are tested using Gaussian level initial condition. The application for rotating or (non-rotating) shallow water equations are given. Full model of 2DSWEs including Coriolis force as the source terms are considered. The numerical simulations are implemented by computer programming using Matlab and Fortran 90 under Dirichlet, reflexive boundary conditions and moving boundary conditions.

Part II: Coupling for Two-Way Nesting Grids: Mathematical Framework and Applications for Shallow Water Model (Nested Grid For 2D Shallow Water Equations)

In the major part of the thesis, we are interested to propose a new two-way interaction technique for multiple nesting grids to solve ocean models. This part consists of two chapters:

Chapter 4: The Configuration a Nested Grid for Shallow Water Models

A literature review of techniques used to try to increase the efficiency and accuracy of 2D shallow water models are presented. Some new algorithms are established to implement two-way interaction technique for this model. Two-way nesting systems depend on the type of interpolation, the location of the dynamical interface, conservation properties and type of update. Different cases of open boundary conditions for two-way nesting grids are studied.

In this chapter, **looks into the optimum feedback conditions** and interpolation techniques to maximize the feedback of the information and to ensure the conservation of properties. Four choices of the update scheme for the free surface elevation and velocities on Arakawa C-grid are applied.

Chapter 5: Performance of Two-Way Nesting Techniques for Shallow Water Equations

The principal objective of this chapter is to study the accuracy of two-way nesting performance techniques for structured grids between a coarse grid and a fine grid for 2DSWEs using an explicit finite difference methods with the refinement ratio 1:3.

We present and evaluate a set of options that made implementation of two-way nesting methods allowing simultaneous spatial and temporal refinement in shallow water model. Results showed two-way nested model can produce accurate high-resolution solutions in areas of interest and improve the realism of the solution in the low-resolution coarse domain for a much lower computational effort than the standard single grid high-resolution model.

Part III: Implementation and Validation of Solutions For 2D Shallow Water Models:
This part includes

Chapter 6: Applications of Two-way Nesting Technique for Multiple Nested Grids with Nesting 3:1 at Multiple Levels

Multiple nested grids can be employed together to save the time as well as obtain enough resolution in the goal region. In this chapter, the explicit finite difference methods which are used to construct a two-way nesting technique are applied successfully for a multiple-nested grid 2DNSWEs for structured grids when the refinement ratio is 1:3 using new algorithms and techniques provided under [Chapter 4](#).

In order to verify the performance of nesting technique, apply some examples of coupling 3 systems for shallow flow models. Comparison of l_2 -relative error norm (l_2 -RE) results for one-way and two-way nesting grids using four update interpolations. Finally, comparison of l_2 -relative error norm (l_2 -RE) results to the free surface elevation for some examples using four options of restriction operator with two cases of the refinement ratio.

Chapter 7: Multiple Nested Model for 2D Non-Linear Shallow water Equations with Nesting 5:1 at Multiple Levels

A two-way interaction technique for multiple nested grids of 2D non-linear shallow water models is constructed and it is applied when the refinement ratio 1:5.

Part IV : Numerical Results of The Tsunami Model

In this part, we discuss some numerical results of the tsunami model.

Chapter 8: Some Applications for Multiple Nested Grids of the Tsunami Model

Several examples for multiple nested grids of the tsunami model are applied when space refinement ratio is 1:5 and the temporal refinement ratio is 1:2 for 2D non-linear SWEs. The performance and accuracy of the model are tested and the results show are good.

Chapter 9: Some Recommendations (Future Works)

Some possible recommendations for future methods of research progress are presented.

All the numerical simulations are performed by computer programming using Matlab and Fortran 90.

Some Publications

This thesis contains several chapters including some articles which are published or submitted:

1. Some Sections in **Chapters 1 and 2:** It has been published in proceeding as: Huda Altaie, New Techniques of Derivations for 2D Shallow Water Equations, **International Journal of Advanced Scientific and Technical Research**, Issue 6 volume 3, May-June 2016.
2. **Chapter 3:** It has been published in proceeding as: Huda Altaie and Pierre Dreyfuss, Numerical Solutions For 2D Depth-Averaged Shallow Water Equations, **International Mathematical Forum**, Vol.13, 2018, no.2, 79-90.
3. Some Sections in **Chapters 4 and 5:** It has been published in proceeding as: Huda Altaie, A Two-Way Nesting for Shallow Water Model, **International Review of Physics (I.RE.PHY.)**, Vol. 11, N. 4, ISSN 1971-680X, August 2017.
4. Some Sections in **Chapter 6:** It has been submitted as: Huda Altaie and Fabrice Planchon, Results of higher accuracy for 2D shallow water equations with (separate/adjacent) for structured grids, 2018.
5. Some Sections in **Chapter 6:** It has been published in proceeding as: Huda Altaie, A Multiply Nested Model for Non-Linear Shallow Water Model, **Journal of Research and Reports on Mathematics**, Vol 1, no.2, 2018.
6. **Chapter 7:** It has been published in proceeding as: Huda Altaie, Numerical Model for Nested Shallow Water equations, **International Journal of Pure and Applied Mathematics- IJ-PAM**, Vol 118, no.4, 1033-1051, 2018.
7. **Chapter 8:** It has been published in proceeding as: Huda Altaie, **Application of a Two-Way Nested Model for Shallow Water**, **Journal of Research and Reports on Mathematics**, 2018.

Other Publications

1. Huda Altaie, **Numerical Methods for Solving the First order linear Fredholm-Voltterra integro Differential Equations**, Journal of Al Nahrain University Vol.12 (2), June, pp.123-127, 2009.
2. Huda Altaie, **Numerical Methods for Solving Linear Fredholm-Voltera integral equations**, Journal of Al Nahrain University Vol.11(3), December, pp. 131-134, 2009.
3. Huda Altaie, **Modified Third Order Iterative Method for Solving Nonlinear Equations**, Journal of Al Nahrain University Vol.16 (3), September, 2013, pp.239 -245.
4. Huda Altaie, **Modified Mid-Point method Method For Solving Linear Fredholm Integral Equations Of The Second Kind**, Journal of Al-Qadisiyah for Computer Science and Mathematics Vol.3 No.1, 2011.
5. Huda Altaie, **Solutions of Systems for the Linear Fredholm-Voltterra Integral Equations of the Second Kind**, Ibn Alhaitham J. Vol.23 (2) 2010.
6. Huda Altaie, **Solutions for linear Fredholm-Voltterra integral Equations of The second Kinds using the Repeated Corrected Trapezoidal and Simpson Methods**, Journal of Al Nahrain University Vol.13(2), june 2010, pp.194 -204.
7. Huda Altaie, **Easy Numerical Method to Solution a System of Linear Voltterra Integral Equations**, Journal of Al Nahrain University Vol.14, September pp.144-149, 2011.
8. Huda Altaie, **New Methods of Ordering 63 to Solve Nonlinear Equations**, 2011.

Mathematics Notations

C	Chezy bed roughness coefficient or Ekman coefficient 0.026.
C_D	The dimensionless coefficient of quadratic friction. Here, the quadratic drag coefficient is taken usually $C_D = 2.5 \cdot 10^{-3}$ but it may also take into account the bottom roughness.
c	Coarse grid/ parent grid.
C_a	Air drag coefficient, is taken usually $C_a = (0.8 + 0.0658\sqrt{u_{10} + v_{10}}) \times 10^{-3}$ where u_{10} and v_{10} shown below.
C_n	Courant number.
f	Coriolis parameter, $f = 2\omega \sin\phi$, where ω is the earth's angular velocity (s^{-1}) and ϕ is north latitude (positive northward), $f = 1.01 \times 10^{-4}$ Coriolis frequency at 42 latitude.
F	Volume force.
g	Acceleration due to gravity ($g = 9.81ms^{-2}$).
H	Total depth ($h + \eta$), where h is the mean sea depth (m) (or sometimes called water depth).
$\vec{\nabla}$	Gradient operator $\vec{\nabla} = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z})$.
Δ	Laplacien operator $\vec{\nabla}^2 = (\frac{\partial}{\partial x}(\frac{\partial}{\partial x}), \frac{\partial}{\partial y}(\frac{\partial}{\partial y}), \frac{\partial}{\partial z}(\frac{\partial}{\partial z}))$.
Ω	Domain of calculation
$t, \Delta t, \Delta x, \Delta y$	Time (s), time step, x -direction grid spacing, and y -direction grid spacing respectively.
(x, y)	Horizontal Cartesian spatial coordinates (m)
η	Elevation of the sea surface above mean sea level (or high surface water, free surface elevation).
(u, v)	Components of velocity in the x and y -directions respectively
u_{10}, v_{10}	The vertically averaged air velocities at a distance of 10m above the sea surface (ms^{-1}).
q	Vector velocity field (ms^{-1}).
ρ	Density of the fluid (sea water) assumed constant ($= 1027kgm^{-3}$).
ρ_a	Density of air assumed uniform (water density) ($= 1.225kgm^{-3}$).
ρ_0	Water mean density ($= 1.033kgm^{-3}$).
ν	Coefficient of viscosity (ms^{-2}).
p, p_a	The pressure, and pressure at the surface respectively.
$\tau_{i,j}$	Viscous shear stress in i -direction on a j -plane.
(\bar{u}, \bar{v})	Components of depth-averaged velocity (mean) in the x and y -directions respectively
\tilde{u}	The deviation of the mean velocity.
$\tilde{u}_i \tilde{u}_j$	Reynold stress.
$\omega = \frac{2\pi}{86164s}$	Angular velocity due to rotation of the earth.
ν_x, ν_y	Coefficient of viscosity in x -direction and y -direction, respectively (ms^{-2}).

Glossary

Shallow water equations	SWEs
2D Shallow water equations	2DSWEs
Nonlinear Shallow water equations	NSWEs
l_2 -relative error norm	l_2 -RE
3D Navier-Stokes equations	3DNSEs
Explicit finite difference method	EFDM
Absolute error	ABSE
Partial Differential Equations	PDEs
Courant Friedrichs Lewy	CFL

Development of 2DSWEs (Goals and Objectives)

The development of the efficiency and advantage of numerical methods for shallow water models are carried out in the following stages:

1. Propose a new technique for deriving 2DSWEs using splitting of velocity and horizontal eddy viscosity.
2. Suggest some ways to expand the derivation of 2D depth-averaged SWEs using 3D Navier-Stokes equations. Another way is presented to derive 2DSWEs without using depth-averaged technique.
3. New spatial and temporal discretization techniques in a standard C-grid using an explicit finite difference method and leapfrog with Robert-Asselin filter for 2DSWEs are applied.
4. Investigation of the proposed techniques which are able to solve 2DSWEs and some examples of the tsunami mode under moving boundary conditions are tested.
5. Dynamical coupling in a two-way nesting system is performed at a dynamical interface which is a separate/adjacent from a mesh interface.
6. Build some new algorithms to implement two-way interaction technique for ocean models.
7. Four choices of restriction operator for the free surface elevation and velocities on Arakawa C-grid are applied. The choice of the full-weighting and the average update operators are proposed which have the excellent properties regarding the filtering.
8. Suggested a new technique for multiple nested grids at multiple levels for shallow water models with nest 3:1 and 5:1.
9. Coupling multi systems for multiple nested grids are achieved for 2DSWEs with nest 3:1 and 5:1 at multiple levels.
10. Apply the proposed technique to multiple nested grids for some examples of the tsunami model.
11. All the simulation are made by using Dirichlet condition, reflexive conditions and moving boundary conditions. Boundary conditions for the nested domain are linearly interpolated from the coarse domain and feedback using the average scheme or (full-weighting scheme) with (separate/adjacent) dynamic interface and feedback interface from the high-resolution nested grids solution to the low-resolution coarse grids solution.

General Introduction

Background and Review

For oceanic phenomena problems such that complex geometry (i.e., with rivers and estuaries), in order to increase the horizontal resolution in a subregion without incurring the computational expense of high resolution over the entire domain. One effective way to overcome this difficulty is to build hierarchies of nested models with a focus on the area of interest. This technique has been widely used in meteorology and in oceanography, for which some examples of applications can be found in [90, 109].

The success or failure of these efforts will depend both on the nesting technique and on the characteristics of the basic ocean model. Conclusions on how nested models perform may depend on options of test problems. There are great works of literature describing nested models for the ocean. Major efforts include ([16, 36, 56, 90, 102, 109]). The drawback of this technique is the great number of parameters or the generation of new problems grid interaction, computational efficiency, and conservation properties (flux of mass and momentum) compared with a classical technique with a single expandable grid.

This system allows a local increase of the mesh resolution in areas where it seems to be necessary, by running the same model on a hierarchy of grid. Nesting (or embedding) techniques for structured mesh generally which indicate an economical way to improve the horizontal resolution, consists a local high-resolution grid embedded in a coarser-resolution one which covers the entire domain. In the case of one-way interaction, coarse grid solution provides boundary conditions for the high-resolution grid. In two-way nesting, the fine grid results are feedback to the coarse grid in addition to the use of the coarse grid results in specifying the boundary conditions of the fine grid. The interaction between the coarse grid and fine grid in two-way nesting can take place either at the dynamic interface between them [122] or over their overlapping region [90]. These methods have been applied successfully in atmosphere and ocean modelling ([36, 48, 52, 53, 99, 104]).

The nesting procedure should preferably conserve fluxes of mass and momentum across the interfaces. In meteorology, such a scheme was developed by Kurihara et al. [122]. Berger [13] have developed general adaptive mesh refinement algorithms for hyperbolic systems that also are conservative across interfaces. Ginis et al. [102] applying the technique proposed by Kurihara et al. [122] developed a nested primitive equation model that did not fictitiously increase or decrease the transports of mass, momentum, and heat through the dynamical interface. Angot [94] address the continuity and conservation properties across interfaces. Rowley and Ginis [27] included a mesh movement scheme in the

nested ocean model and stated that mass, heat, and momentum are conserved during the movement.

Most papers describing nested ocean modelling efforts discuss stability problems and unsmooth solutions across the interfaces. Spall [109] finding support in Zhang [52] state that it may be necessary to sacrifice exact conservation to obtain smooth, stable solutions. To stabilize, and smooth the solutions we find that combinations of horizontal and vertical diffusion, altering the solutions in time and relaxation techniques or nudging are often applied.

In the literature, both one-way and two-way interaction between the coarse and the fine grid have been considered. In the case of one-way interaction, coarse grid solution supply boundary conditions for the fine grid, but there is no feedback from the fine grid. Phillips and Shukla [98] argue that a two-way interaction gives correct solution on the fine grid and therefore is more favorable. The two-way nesting described in [90, 102, 109]. Also, a recent survey of two-way embedding algorithms can be found in ([16, 19, 47, 66]).

The various two-way interaction schemes mainly differ by type of interpolation, the location of dynamical interface, conservation properties (flux of mass, and momentum equations), and type of update.

However, a two-way interaction may introduce instabilities at the interface between the two grids, and such instabilities may lead to a severe degradation solution [52]. In some studies data from previously run coarse grid models are used to drive fine grid models ([65, 111]). Fox [56] compared one-way and two-way nesting and concluded that using the model in one-way nesting mode resulted in more noise at the fine grid mesh boundaries with a negligible decrease in computer time. In order to provide the boundary condition for the fine grid, the coarse grid variables must be interpolated to the fine grid. There are numerous techniques that are potentially interesting for performing two-way nesting (see [83, 102]). Based on studies with an idealized test case, they conclude that zeroth-order interpolation may create large phase errors, quadratic interpolation may create overshooting and they suggest the use of advection equivalent interpolation schemes.

Nested model grids may be adaptive and movable or static. In order to follow evolving oceanic features such as wavefronts and propagating eddies it may be beneficial to apply for instance adaptive mesh refinement methods for hyperbolic systems described by [13] or more recently by [16, 19, 47, 66] have recently applied this technique to study the propagation of the barotropic model and with a multi-layered quasi-geostrophic model.

Spall and Holland [109] apply the same time step both on the coarse and the fine grid arguing that the coarse grid contributes little to the overall expense and that it would add an additional level of computational complexity for very little gain to have different time steps on the two grid levels. With equal Courant numbers on all levels the quality of the wave propagation relative to the mesh size will be approximately the same, and most of the more papers on nesting also refine the time step with the

same factor as the spatial resolution, keeping the Courant numbers constant.

Warner et al [97] recommend using two-way nesting whenever possible since the solution is presumed to be more accurate when the coarse and nested grid solutions are allowed to interact with one another.

Phillips [98] studied the distortion of shallow-water Rossby and gravity waves in simulations using both a one-way and a two-way nest. The authors claimed that the two-way solution is more accurate than the one-way solution because coarse grid solution is nearer to that on the nested grid but they do not elaborate on this rather obvious point. In particular, there is no analysis of why reflections may be less when using two-way nesting.

Clark [25] and Chen [24] both performed simulations of two-dimensional linear vertically propagating mountain waves using nested grids to increase both the horizontal and vertical resolution near the mountain and they did not use the interpolation boundary condition but instead linearly interpolated fluxes to the boundary of the nested grid. This approach yields conservation of mass and momentum across the nested grid boundary which is a desirable property for some modelers.

Chen [24] used a similar nesting strategy in a fully compressible model to test several different boundary conditions including the interpolation boundary condition and a continuously stratified variant of the inflow-outflow boundary condition for shallow water flow [20].

In ocean model, the nesting is the degree of refinement from one level to the next and the grid ratio from 2:1 to 7:1 has been applied. Spall and Holland [109] conclude that 3:1 and 5:1 ratios perform quite well, and even ratio of 7:1 are able to reproduce the solution reasonably well while the features are mostly contained within the fine region. To apply small ratios like 2:1, which is used for instance in Rowley and Ginis [27], may force us to apply many grid levels before we achieve the resolution we would like to have in a given area. On the other hand, large ratios may cause instabilities and non-smooth solutions across the interfaces. There are numerous combinations of basic ocean models and nesting techniques that are potentially interesting and evidence on how these groups perform is gradually growing as they are applied both to idealized test cases and for more realistic oceanic problems [73, 90, 94].

Mathematical and Numerical Model For 2D
Depth-Averaged Shallow Water Equations

Chapter 1

Derivation of 2D Shallow Water Equations

Chapter 1

Derivation of 2D Shallow Water Equations

The results presented in this chapter (Sections 1.4 and 1.5) are the subject of the article [2].

As we know, there are several possible strategies to derive 2D non-linear shallow water equations. The first possibility is to integrate the 3D Navier-Stokes equations over the vertical direction and take averaged values for the velocities. The second alternative is to consider conservation laws for mass and momentum directly on an infinitesimal control volume.

This chapter **focuses** on the derivation of 2D depth-averaged SWEs in a manner that hopes to highlight the important assumptions required. Several approaches can be used to develop this derivation, we suggest two ways to expand the derivation of 2D shallow water flow from 3D Navier-Stokes equations using splitting of velocity and horizontal eddy viscosity. Another way is introduced to derive this model without using the depth-averaged technique.

Highlights

- Mathematical description of 3D Navier-Stokes equations is presented.
- The complete derivation of 2DSWEs is explained .
- Kinematic and dynamic boundary conditions are applied.

Contents

1.1	Introduction	39
1.2	Mathematical description for 3D Navier-Stokes equations	40
1.2.1	The conservation of mass law	40
1.2.2	The conservation of momentum	41
1.2.3	Stress components	41
1.3	The mathematical description of shallow water model	44
1.3.1	2D depth-averaged SWEs	44
1.3.2	General consideration:	44
1.3.3	The continuity equation	45
1.3.4	The momentum equations	45
1.3.5	Surface and bottom boundary conditions	46
1.3.6	Leibnitz Rule	47
1.4	Derivation of 2DSWEs from first formula of 3D Navier-Stokes equations	48
1.4.1	Principle	48
1.4.2	The depth-averaged continuity equation	49
1.4.3	The momentum depth-averaged equations	49
1.4.4	Remarks:	54
1.4.5	Bottom shear stress Formulas	56
1.5	Development of 2DSWEs derivation from the second formula of 3D Navier-Stokes equations	57
1.5.1	Results and discussions	57
1.6	Case 2: Derivation 2DSWEs without using the depth-averaged technique	60
1.7	The General formula of non-conservative form for the 2D shallow water equations	62
1.8	Summary and Conclusion	63

1.1 Introduction

Shallow water equations are hyperbolic partial differential equations (or parabolic if viscous shear is considered) which explain the flow behavior of rivers. These equations were established in (1775) by Laplace and were first used in (1871) by the physicist Adh emar Jean Claude Barr e de Saint-Venant. More generally, SWEs describe the evolution of unsteady flow (incompressible) of a fluid, not necessarily water with constant density. This system is essentially based on the assumption that the water depth is shallow.

This assumption implies that the model is hydrostatic and state that the velocity is constant with the depth, bounded from below by the bottom topography and from above by the water surface. The 3D incompressible Navier-Stokes equations are averaged over the depth to obtain SWEs.

These equations are applicable to mathematical concepts where the typical vertical scale is negligible compared to the typical horizontal scale, and they are effective for many physical phenomena in the oceans, coastal regions, rivers and canals, lake hydrodynamics, dam breaks,...etc (see [30, 40, 57, 73, 85, 87, 95, 101]).

The derivation of shallow water models effects has received an extensive coverage [63, 105] and numerical techniques for the approximation of these models have been recently proposed [33, 72]. This technique for deriving of non-linear SWEs is classical when the viscosity or the wind effects on the free surface are neglected [45, 54]. In the literature, we often find various non-linear SWEs which may include wind effects on the free surface, bottom topography and friction effects on the bottom, often defined using the Manning Chezy formula, or viscosity [43, 45, 68]. Several variations for shallow water models can be constructed from different assumptions regarding the nature of the fluid, such as viscosity, incompressibility, and properties of the domain in which the fluid is situated.

This chapter is devoted to highlighting some essential properties of 2DSWEs; it is organized as follows: Derivation of 2DSWEs is described so that they can be derived in a number of ways with varying initial assumptions and mathematical complexity. In Sections 1.2 and 1.3, we recall the mathematical description of 3D Navier-Stokes equations and 2DSWEs. A new technique for complete derivation for 2D non-linear SWEs is constructed in Sections 1.4 and 1.5. Finally, another way without using depth-averaged technique to derive this model is introduced in Section 1.6.

The following figures indicate to applications of SWEs in ocean modeling



Figure 1-1: Shallow water equations in canals.



Figure 1-2: Shallow water equations in dam break.

1.2 Mathematical Description for 3D Navier-Stokes Equations

The Navier-Stokes equations are based on Newton's second law and consist of a time-dependent continuity equation for conservation of mass, three time-dependent conservation of momentum equations which actually results from the fundamental relationship of fluid dynamics. These equations describe how the velocity, pressure, temperature, and density of a moving fluid are related. The equations were derived independently by G.G. Stokes, in England, and M. Navier, in France, in the early 1800's, which are obtained from the two basic principles of physical laws: ([71, 117]).

1.2.1 The conservation of mass law

The conservation of mass is the first principle used to develop the basic equations of fluid mechanics. Simply the conservation of mass implies that the total mass of a closed system in a region Ω is constant over time and mass can neither be created nor destroyed, where $\Omega \in R^3$ (see [51, 118]).

The general formula of the mass continuity in conservation form is:

$$\frac{\partial \rho}{\partial t} + \rho \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right) = 0$$

If the fluid density is constant, the physical interpretation of the equations implies that the change of density of a fluid particle is equal to the expansion of the fluid. Also, when assuming the fluid is incompressible. This means that ρ does not depend on p ($\rho = \rho_0$ is constant), which is not a necessary condition for incompressible flow, thus the derivative of the density over time is zero.

$$\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right) = 0$$

1.2.2 The conservation of momentum

The conservation of momentum rule is based on Newton's second law which states that: if the resulting force F acts on a body of mass m , then the rate of increase of linear momentum is equal to the force F (see [51, 118]). For an incompressible fluid and constant viscosity, the mathematical equation representing this law is:

$$\frac{Dq}{Dt} = -\frac{1}{\rho} \cdot \nabla p + \nu \nabla^2 (q) + F$$

Physical meaning of each term

- (a) $\frac{Dq}{Dt}$ Change in velocities over time, where D/Dt is the material or total derivative.
- (b) ∇p The internal pressure gradient of the fluid (the change in pressure)
- (c) $\nu \nabla^2 (q)$ The internal stress forces acting on the fluid.
- (d) F The external stress forces acting on the fluid.

1.2.3 Stress components

The stress state is represented as a symmetric tensor τ , whose components may be expanded in to various coordinate systems. The components of the velocity vector (u, v, w) align with the Cartesian-Coordinate directions (x, y, z) . For such fluids are called Newtonian fluids. If assume an incompressible fluid the components of stress tensors as follows: [91, 118].

$$\begin{aligned}\tau_{xx} &= \mu \cdot \left(\frac{\partial u}{\partial x} + \frac{\partial u}{\partial x} \right), & \tau_{yy} &= \mu \cdot \left(\frac{\partial v}{\partial y} + \frac{\partial v}{\partial y} \right), & \tau_{zz} &= \mu \cdot \left(\frac{\partial w}{\partial z} + \frac{\partial w}{\partial z} \right) \\ \tau_{xy} = \tau_{yx} &= \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right), & \tau_{xz} = \tau_{zx} &= \mu \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right), & \tau_{yz} = \tau_{zy} &= \mu \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right)\end{aligned}$$

where μ is called the coefficient of dynamic viscosity and the kinematic viscosity $\nu = \frac{\mu}{\rho}$.

In general, $\tau_{i,j}$ represent the viscous shear stresses in i -direction on a j -plane, which can be expressed in terms of fluid deformation rate:

$$\frac{\tau_{i,j}}{\rho} = \nu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)$$

The formula of the continuity equation and 3D incompressible Navier-Stokes equations of a non-conservation form are given by (see [9, 51, 118]).

$$\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right) = 0 \quad (1.1)$$

$$\rho \left(\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} \right) = F_x - \frac{\partial p}{\partial x} + \left(\frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} \right) \quad (1.2)$$

$$\rho \left(\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z} \right) = F_y - \frac{\partial p}{\partial y} + \left(\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} \right) \quad (1.3)$$

$$\rho \left(\frac{\partial w}{\partial t} + u \frac{\partial w}{\partial x} + v \frac{\partial w}{\partial y} + w \frac{\partial w}{\partial z} \right) = F_z - \frac{\partial p}{\partial z} + \left(\frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z} \right) \quad (1.4)$$

We will called the above formula, the first form

Where F_x , F_y and F_z represent the volume forces. If only the gravitational force and the Coriolis force are accounted for the vector of volume forces can be written as:

$$\vec{F} = \begin{pmatrix} 2\rho\omega v \sin\theta \\ -2\rho\omega u \sin\theta \\ -\rho g \end{pmatrix}$$

Remark:

If we substitute the component of the stress tensor τ_{xx} , τ_{yy} , τ_{zz} , τ_{xy} , τ_{xz} , and τ_{yz} in equations (1.2)-(1.4), we obtain:

For the case of the x -component

$$\begin{aligned}
 \rho \left(\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} \right) &= F_x - \frac{\partial p}{\partial x} + \mu \frac{\partial}{\partial x} \left(\frac{\partial u}{\partial x} + \frac{\partial u}{\partial x} \right) + \mu \frac{\partial}{\partial y} \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right) \\
 &\quad + \mu \frac{\partial}{\partial z} \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) \\
 &= F_x - \frac{\partial p}{\partial x} + \mu \left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 v}{\partial x \partial y} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 w}{\partial x \partial z} + \frac{\partial^2 u}{\partial z^2} \right] \\
 &= F_x - \frac{\partial p}{\partial x} + \mu \left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} + \frac{\partial}{\partial x} \underbrace{\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right)}_{\text{continuity equation} = 0} \right] \\
 &= F_x - \frac{\partial p}{\partial x} + \mu \left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right]
 \end{aligned}$$

Similarly, we can do for the y and z -components.

Then the formula of the continuity equation and 3D incompressible Navier-Stokes equations of a non-conservation form are given by (see [51, 77, 118]).

$$\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right) = 0 \tag{1.5}$$

$$\rho \left(\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} \right) = F_x - \frac{\partial p}{\partial x} + \mu \left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right] \tag{1.6}$$

$$\rho \left(\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z} \right) = F_y - \frac{\partial p}{\partial y} + \mu \left[\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2} \right] \tag{1.7}$$

$$\rho \left(\frac{\partial w}{\partial t} + u \frac{\partial w}{\partial x} + v \frac{\partial w}{\partial y} + w \frac{\partial w}{\partial z} \right) = F_z - \frac{\partial p}{\partial z} + \mu \left[\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2} \right] \tag{1.8}$$

We will called the above formula, the second form.

1.3 The Mathematical Description of Shallow Water Model

1.3.1 2D depth-averaged SWEs

In this section, we recall some basic concepts such as continuity equation and momentum equations. In order to SWEs to be applicable, there are some conditions must be met (see [15, 23, 38, 85, 96]).

1.3.2 General consideration:

1. In shallow water equations, the water depth h is much smaller than horizontal scale of motion l .

The major condition which will will be

$$\zeta = \frac{h}{l} \ll 1.$$

2. The vertical momentum exchange is negligible in comparison to the horizontal momentum exchange and the vertical velocity component w is much smaller than the horizontal component u , v (i.e $|w| \ll |u|$ and $|w| \ll |v|$).
3. A major assumption of depth averaging is that the flow in the vertical direction is small.
4. The main condition is that all terms in the z -direction of the equation (1.4) are small compared to the gravity and pressure terms (we assume here that the acceleration of the movement on the vertical is negligible except acceleration due to gravity). Thus, the z -direction of equation (1.4) reduces to

$$\frac{\partial p}{\partial z} = -\rho g$$

This implies that the pressure distribution over the vertical direction is **hydrostatic**.

5. Consider the free surface (water-air interface) at $z = \eta$ and the bottom (water-sediment interface) at $z = -h$ using the assumption of hydrostatic pressure. In addition to height of water already defined, There are two new variables will appear:

$$\bar{u} = \frac{1}{H} \int_{-h}^{\eta} u \, dz \qquad \bar{v} = \frac{1}{H} \int_{-h}^{\eta} v \, dz \qquad (1.9)$$

and $H = \eta + h$ is the total water depth. These averages are on the vertical of the horizontal components of the velocity vector will be called **vertically-averaged velocities**.

1.3.3 The continuity equation

The equation that results from applying mass conservation is called continuity equation. The depth-averaged form of this equation is (see [44, 69]):

$$\frac{\partial \eta}{\partial t} + \frac{\partial(Hu)}{\partial x} + \frac{\partial(Hv)}{\partial y} = 0 \quad (1.10)$$

Physical meaning of each term

The terms in the continuity equation have the following meanings:

- (a) $\frac{\partial \eta}{\partial t}$ Change the water surface elevation over time.
- (b) $\frac{\partial(Hu)}{\partial x}$ The gradient of x-component of the flow volume between surface and seafloor.
- (c) $\frac{\partial(Hv)}{\partial y}$ The gradient of y-component of above.

1.3.4 The momentum equations

For non-stratified well mixed coastal flows involving tides, winds and atmospheric. The depth-averaged momentum equation of conservative form along the x -direction and y -direction respectively are (see [44, 69]).

$$\begin{aligned} \frac{\partial(Hu)}{\partial t} + \frac{\partial(Hu^2)}{\partial x} + \frac{\partial(Huv)}{\partial y} - fHv = & -gH\frac{\partial \eta}{\partial x} + \nu_x H\left(\frac{\partial^2 u}{\partial x^2}\right) + \nu_y H\left(\frac{\partial^2 u}{\partial y^2}\right) + \frac{\rho_a}{\rho_0} C_a u_{10} \sqrt{u_{10}^2 + v_{10}^2} \\ & - \frac{\rho_0}{\rho_0} C_D u \sqrt{u^2 + v^2} \end{aligned} \quad (1.11)$$

$$\begin{aligned} \frac{\partial(Hv)}{\partial t} + \frac{\partial(Hvu)}{\partial x} + \frac{\partial(Hv^2)}{\partial y} + fHu = & -gH\frac{\partial \eta}{\partial y} + \nu_x H\left(\frac{\partial^2 v}{\partial x^2}\right) + \nu_y H\left(\frac{\partial^2 v}{\partial y^2}\right) + \frac{\rho_a}{\rho_0} C_a v_{10} \sqrt{u_{10}^2 + v_{10}^2} \\ & - \frac{\rho_0}{\rho_0} C_D v \sqrt{u^2 + v^2} \end{aligned} \quad (1.12)$$

where $\tau_x^s = \rho_a C_a u_{10} \sqrt{u_{10}^2 + v_{10}^2}$, $\tau_x^b = \rho_0 C_D u \sqrt{u^2 + v^2}$, $\tau_y^s = \rho_a C_a v_{10} \sqrt{u_{10}^2 + v_{10}^2}$, $\tau_y^b = \rho_0 C_D v \sqrt{u^2 + v^2}$, and the values C_a , C_D , u_{10} , v_{10} , ρ_0 , and ρ_a listed in mathematical notations Page 23.

Physical meaning of each term

The physical meaning of each term in the x -momentum equation is described below and the terms in the y -momentum equation are similar.

1. $\frac{\partial u}{\partial t}$ Change of the velocity over time (sometimes called the local variation of momentum over time).

2. $\frac{\partial u^2}{\partial x} + \frac{\partial uv}{\partial y}$ The advective terms. The terms are non-linear and are sometimes called spatial acceleration terms.

3. fv The Coriolis term.

4. $g\frac{\partial\eta}{\partial x}$ The gravity term, which means the force due to the gradient of the surface in the x -direction.

5. $\nu_x(\frac{\partial^2 u}{\partial x^2}) + \nu_y(\frac{\partial^2 v}{\partial y^2})$ Horizontal viscosity terms.

6. $\frac{\rho_a}{\rho_0} C_a u_{10} \sqrt{u_{10}^2 + v_{10}^2}$ The x - component of wind stress acting on the surface of the sea.

7. $C_D u \sqrt{u^2 + v^2}$ The x -component of friction acting on the bottom surface.

1.3.5 Surface and bottom boundary conditions

Shallow water equations have to be implemented with boundary conditions. In this section, boundary conditions will be discussed on the free surface and at the solid bottom.

1. Kinematic boundary conditions

The kinematic boundary conditions prescribe that the water particles can not cross the solid bottom nor the free surface. For the bottom, the normal velocity component must vanish. Since the free surface might be moving by itself, the normal velocity of the fluid should equal the normal velocity of the surface [29, 35].

a. At the solid bottom

We can explain this condition that the bottom is a material surface of the fluid, not crossed by the flow and stationary (no normal flow).

$$\left[u|_{z=-h} \frac{\partial(-h)}{\partial x} + v|_{z=-h} \frac{\partial(-h)}{\partial y} - w|_{z=-h} \right] = 0 \quad (1.13)$$

$$\Rightarrow w|_{z=-h} = u|_{z=-h} \frac{\partial(-h)}{\partial x} + v|_{z=-h} \frac{\partial(-h)}{\partial y}$$

b. At a free surface

Here, the situation is more complicated because of the fact that the boundary is moving with the fluid. (no relative normal flow)

$$\frac{\partial \eta}{\partial t} = - \left[u|_{z=\eta} \frac{\partial \eta}{\partial x} + v|_{z=\eta} \frac{\partial \eta}{\partial y} - w|_{z=\eta} \right] \quad (1.14)$$

2. Dynamic boundary conditions

We have dynamic boundary conditions for the forces that act at the bottom and surface boundaries. (see [29, 35]).

a. At the solid bottom:

For the bottom, we have the no-slip conditions $u = v = w = 0$.

The equation of the bottom stress can be represented as [39]:

$$\tau_x^b = - \left[\tau_{xx} \frac{\partial(-h)}{\partial x} + \tau_{yx} \frac{\partial(-h)}{\partial y} - \tau_{zx} \right]_{z=-h}$$

Similarly for y-direction.

b. At a free surface

The equation of wind stress at the water surface can be represented as [39]:

$$\tau_x^s = - \left[\tau_{xx} \frac{\partial \eta}{\partial x} + \tau_{yx} \frac{\partial \eta}{\partial y} - \tau_{zx} \right]_{z=\eta}$$

Similarly for y-direction.

1.3.6 Leibnitz rule

Leibniz's formula is applied to invert the differential operators and integration. This rule says that the derivative of the integral at the boundaries of the variables makes appear a derivative inside the integral and flow terms according to the formula [64].

$$\frac{\partial}{\partial x} \int_{A(x,y,t)}^{B(x,y,t)} F dz = \int_{A(x,y,t)}^{B(x,y,t)} \frac{\partial F}{\partial x} dz + F|_{(z=B)} \frac{\partial B}{\partial x} - F|_{z=A} \frac{\partial A}{\partial x}$$

Where $A(x,y,t)$, $B(x,y,t)$ be the bottom water depth and the water surface elevation respectively and $F(x,y,t)$ be a smooth function.

1.4 Derivation of 2DSWEs from the First Formula of 3D Navier-Stokes Equations

There will be some basic steps to derive 2DSWEs from the equations (1.1)-(1.4). Firstly, one needs to specify boundary conditions (BCs) for a water column. The second step is to carry out the depth-averaged integration. Finally, to apply the BCs within the integration operation. Beside the BCs, we will use Leibnitz rule to derive SWE [23].

1.4.1 Principle

Suppose we can split each momentary velocity in some types of mean and a random variation (volatile velocity) as follows:

$$u(z) = \bar{u}(z) + \tilde{u}(z) \quad v(z) = \bar{v}(z) + \tilde{v}(z) \quad \text{and} \quad w(z) = \bar{w}(z) + \tilde{w}(z)$$

For these distribution coefficient, the following relation are valid:

$$\int_{-h}^{\eta} \tilde{u}(z) dz = 0 \quad \int_{-h}^{\eta} \tilde{v}(z) dz = 0$$

From now, we will only consider the mean velocities and neglect the random variations, except the advection terms for 2DSWEs, which we will discuss later (see Section 1.4.4).

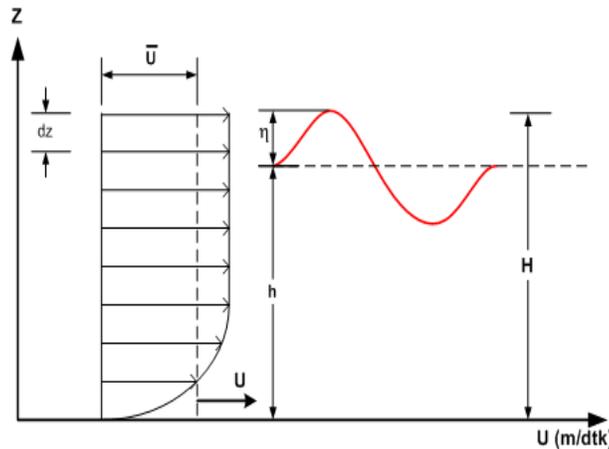


Figure 1-3: Depth-averaged velocity distribution.

1.4.2 The depth-averaged continuity equation

We can start to integrate the individual terms of the continuity equation (1.1) over the vertical as follows:

$$0 = \int_{-h}^{\eta} \left[\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right] dz = \underbrace{\int_{-h}^{\eta} \frac{\partial u}{\partial x} dz}_I + \underbrace{\int_{-h}^{\eta} \frac{\partial v}{\partial y} dz}_{II} + \underbrace{\int_{-h}^{\eta} \frac{\partial w}{\partial z} dz}_{III} \quad (1.15)$$

The terms I, II, and III can be simplified as follows:

$$\begin{aligned}
I &= \frac{\partial}{\partial x} \int_{-h}^{\eta} u dz + u|_{z=-h} \frac{\partial(-h)}{\partial x} - u|_{z=\eta} \frac{\partial \eta}{\partial x} \\
II &= \frac{\partial}{\partial x} \int_{-h}^{\eta} v dz + v|_{z=(-h)} \frac{\partial(-h)}{\partial x} - v|_{z=\eta} \frac{\partial \eta}{\partial y} \\
III &= w|_{z=\eta} - w|_{z=-h}
\end{aligned}$$

Hence, equation (1.15) leads to:

$$\begin{aligned}
0 &= \frac{\partial}{\partial x} \int_{-h}^{\eta} u dz + \frac{\partial}{\partial y} \int_{-h}^{\eta} v dz + \underbrace{\left[u \frac{\partial(-h)}{\partial x} + v \frac{\partial(-h)}{\partial y} - w \right]_{z=-h}}_{=0 \text{ by using equation (1.13)}} - \underbrace{\left[u \frac{\partial \eta}{\partial x} + v \frac{\partial \eta}{\partial y} - w \right]_{z=\eta}}_{=-\frac{\partial \eta}{\partial t} \text{ by using equation (1.14)}} \\
&= \frac{\partial \eta}{\partial t} + \frac{\partial}{\partial x} \int_{-h}^{\eta} u dz + \frac{\partial}{\partial y} \int_{-h}^{\eta} v dz
\end{aligned} \tag{1.16}$$

By using equation (1.9) in the terms 2 and 3 of equation (1.16). **Finally, we obtain the depth-averaged of the continuity equation** (some times called vertically-integrated continuity equation) as follows:

$$\frac{\partial \eta}{\partial t} + \frac{\partial(H\bar{u})}{\partial x} + \frac{\partial(H\bar{v})}{\partial y} = 0$$

1.4.3 The momentum depth-averaged equations

Consider the x-momentum equation given by equation (1.2)

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{xx}}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y} + \frac{1}{\rho} \frac{\partial \tau_{zx}}{\partial z} + \frac{1}{\rho} F_x$$

and the z-momentum given in equation (1.4)

$$\frac{\partial p}{\partial z} = -\rho g$$

By Integrating the above equation from the free surface at $z = \eta$ to some level z

$$\int_{p_s}^p dp = - \int_{\eta}^z \rho g dz$$

$$p - p_s = -\rho g(z - \eta)$$

Due to hydrostatic approach and constant density, the pressure depends on η and the vertical coordinate

$$p - p_s = \rho g\eta - \rho g z \quad \Rightarrow \quad p = p_s + \rho g\eta - \rho g z$$

$$\frac{-1}{\rho} \frac{\partial p}{\partial x} = \frac{-1}{\rho} \frac{\partial p_s}{\partial x} - g \frac{\partial \eta}{\partial x} + g \frac{\partial z}{\partial x}$$

where p_s means pressure at the free surface.

Thus,

$$\frac{-1}{\rho} \frac{\partial p}{\partial x} = -g \frac{\partial \eta}{\partial x} \quad \text{or} \quad \frac{-1}{\rho} \frac{\partial p}{\partial x} = -g \frac{\partial \eta}{\partial x} + g s_0$$

where s_0 the bottom slope.

The horizontal pressure gradients depend on the free surface η only

$$\frac{-1}{\rho} \frac{\partial p}{\partial x} = -g \frac{\partial \eta}{\partial x} \quad \text{and} \quad \frac{-1}{\rho} \frac{\partial p}{\partial y} = -g \frac{\partial \eta}{\partial y}$$

Then, the x -momentum equation becomes:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} = -g \frac{\partial \eta}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{xx}}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y} + \frac{1}{\rho} \frac{\partial \tau_{zx}}{\partial z} + \frac{1}{\rho} F_x \quad (1.17)$$

Then, by integrating the left hand side (LHS) of the equation (1.17), we obtain

$$\int_{-h}^{\eta} \left[\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} \right] dz = \underbrace{\int_{-h}^{\eta} \frac{\partial u}{\partial t} dz}_I + \underbrace{\int_{-h}^{\eta} \frac{\partial u^2}{\partial x} dz}_II + \underbrace{\int_{-h}^{\eta} \frac{\partial(uv)}{\partial y} dz}_III + \underbrace{\int_{-h}^{\eta} \frac{\partial(uw)}{\partial z} dz}_IV \quad (1.18)$$

To explain how to change the terms in the LHS of equation (1.18) from non-conservative to conservative form as follows:

By adding u times the continuity equation to x -momentum equation (1.17), we obtain:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} + u \frac{\partial u}{\partial x} + u \frac{\partial v}{\partial y} + u \frac{\partial w}{\partial z}$$

Re-arranging, we obtain right hand side (RHS) of the equation (1.18)

$$\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x} + \frac{\partial(uv)}{\partial y} + \frac{\partial(uw)}{\partial z}$$

1. Time derivative and advection terms

By using Leibnitz rule and integration theorem. The terms I , II , III , and IV can be simplified as follows:

$$\begin{aligned}
 I &= \int_{-h}^{\eta} \frac{\partial u}{\partial t} dz = \frac{\partial}{\partial t} \int_{-h}^{\eta} u dz + u|_{z=-h} \frac{\partial(-h)}{\partial t} - u|_{z=\eta} \frac{\partial \eta}{\partial t} \\
 II &= \int_{-h}^{\eta} \frac{\partial u^2}{\partial x} dz = \frac{\partial}{\partial x} \int_{-h}^{\eta} u^2 dz + u^2|_{z=-h} \frac{\partial(-h)}{\partial x} - u^2|_{z=\eta} \frac{\partial \eta}{\partial x} \\
 III &= \int_{-h}^{\eta} \frac{\partial uv}{\partial y} dz = \frac{\partial}{\partial y} \int_{-h}^{\eta} uv dz + uv|_{z=-h} \frac{\partial(-h)}{\partial y} - uv|_{z=\eta} \frac{\partial \eta}{\partial y} \\
 IV &= \int_{-h}^{\eta} \frac{\partial uw}{\partial z} dz = uw|_{z=\eta} - uw|_{z=-h}
 \end{aligned}$$

Hence, the equation (1.18) can be rewritten as:

$$\begin{aligned}
 &\int_{-h}^{\eta} \left[\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x} + \frac{\partial(uv)}{\partial y} + \frac{\partial(uw)}{\partial z} \right] dz = \\
 &\frac{\partial}{\partial t} \int_{-h}^{\eta} u dz + \frac{\partial}{\partial x} \int_{-h}^{\eta} u^2 dz + \frac{\partial}{\partial y} \int_{-h}^{\eta} uv dz - u|_{z=\eta} \left[\frac{\partial \eta}{\partial t} + \underbrace{u|_{z=\eta} \frac{\partial \eta}{\partial x} + v|_{z=\eta} \frac{\partial \eta}{\partial y} - w|_{z=\eta}}_{=-\frac{\partial \eta}{\partial t} \text{ by using equation (1.14)}} \right] \\
 &+ u|_{z=-h} \left[\underbrace{\frac{\partial(-h)}{\partial t}}_{=0} + \underbrace{u|_{z=-h} \frac{\partial(-h)}{\partial x} + v|_{z=-h} \frac{\partial(-h)}{\partial y} - w|_{z=-h}}_{=0 \text{ by using equation (1.13)}} \right]
 \end{aligned}$$

It follows that:

$$\frac{\partial}{\partial t} \int_{-h}^{\eta} u dz + \frac{\partial}{\partial x} \int_{-h}^{\eta} u^2 dz + \frac{\partial}{\partial y} \int_{-h}^{\eta} uv dz = \frac{\partial(H\bar{u})}{\partial t} + \frac{\partial(H\bar{u}^2)}{\partial x} + \frac{\partial(H\bar{u}\bar{v})}{\partial y}$$

By using equation (1.9) in the terms 1, 2, and 3 of the above equation.

Now, we will integrate the RHS of the equation (1.17)

$$\begin{aligned}
 \int_{-h}^{\eta} \left[-g \frac{\partial \eta}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{xx}}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y} + \frac{1}{\rho} \frac{\partial \tau_{zx}}{\partial z} + \frac{1}{\rho} F_x \right] dz &= \underbrace{\int_{-h}^{\eta} -g \frac{\partial \eta}{\partial x} dz}_I + \underbrace{\int_{-h}^{\eta} \left[\frac{1}{\rho} \frac{\partial \tau_{xx}}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y} + \frac{1}{\rho} \frac{\partial \tau_{zx}}{\partial z} \right] dz}_{II} \\
 &+ \underbrace{\int_{-h}^{\eta} \frac{1}{\rho} F_x dz}_{III} \tag{1.19}
 \end{aligned}$$

2. Pressure gradient, diffusion terms and Coriolis terms

The terms I , II , and III can be simplified as follows:

$$\begin{aligned}
 I &= - \int_{-h}^{\eta} g \frac{\partial \eta}{\partial x} dz = -g \frac{\partial}{\partial x} \int_{-h}^{\eta} \eta dz - g \eta_{z=\eta} \frac{\partial(\eta)}{\partial x} + g \eta_{z=-h} \frac{\partial(-h)}{\partial x} \\
 II &= \int_{-h}^{\eta} \left[\frac{1}{\rho} \frac{\partial \tau_{xx}}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y} + \frac{1}{\rho} \frac{\partial \tau_{zx}}{\partial z} \right] dz \\
 &= \frac{1}{\rho} \frac{\partial}{\partial x} \int_{-h}^{\eta} \tau_{xx} dz + \frac{1}{\rho} \frac{\partial}{\partial y} \int_{-h}^{\eta} \tau_{yx} dz - \frac{1}{\rho} \left[\tau_{xx} \frac{\partial \eta}{\partial x} + \tau_{yx} \frac{\partial \eta}{\partial y} - \tau_{zx} \right]_{z=\eta} \\
 &\quad + \frac{1}{\rho} \left[\tau_{xx} \frac{\partial(-h)}{\partial x} + \tau_{yx} \frac{\partial(-h)}{\partial y} - \tau_{zx} \right]_{z=-h} \\
 III &= \int_{-h}^{\eta} \frac{1}{\rho} F_x dz = \frac{1}{\rho} \int_{-h}^{\eta} (2\rho \omega v \sin \theta) dz = f H \bar{v}
 \end{aligned}$$

Now, applying the boundary condition by performing a stress balance at the surface, it can be show that:

$$\tau_x^s = - \left[\tau_{xx} \frac{\partial \eta}{\partial x} + \tau_{yx} \frac{\partial \eta}{\partial y} - \tau_{zx} \right]_{z=\eta}$$

Similarly at the bottom

$$\tau_x^b = - \left[\tau_{xx} \frac{\partial(-h)}{\partial x} + \tau_{yx} \frac{\partial(-h)}{\partial y} - \tau_{zx} \right]_{z=-h}$$

Then, the equation (1.19) leads to:

$$\begin{aligned}
 &\int_{-h}^{\eta} \left[-g \frac{\partial \eta}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{xx}}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y} + \frac{1}{\rho} \frac{\partial \tau_{zx}}{\partial z} + \frac{1}{\rho} \int_{-h}^{\eta} F_x \right] dz = \\
 &-g \frac{\partial}{\partial x} \int_{-h}^{\eta} \eta dz + g \eta_{z=\eta} \frac{\partial(\eta)}{\partial x} - g \eta_{z=-h} \frac{\partial(-h)}{\partial x} + \frac{1}{\rho} H \frac{\partial \bar{\tau}_{xx}}{\partial x} + \frac{1}{\rho} H \frac{\partial \bar{\tau}_{yx}}{\partial y} - \frac{\tau_x^b}{\rho} + \frac{\tau_x^s}{\rho} + f H \bar{v}
 \end{aligned}$$

By expanding terms that involve gravity, we get:

$$-g \frac{\partial}{\partial x} \int_{-h}^{\eta} \eta dz = -g \frac{\partial}{\partial x} \int_{-h}^{\eta} \eta dz = -g \eta \frac{\partial}{\partial x} \int_{-h}^{\eta} dz = -g \frac{\partial(\eta H)}{\partial x}$$

By using the chain rule for the following equation leads to:

$$\begin{aligned}
-g \frac{\partial(\eta H)}{\partial x} + g\eta \frac{\partial\eta}{\partial x} - g\eta \frac{\partial(-h)}{\partial x} &= -\eta g \frac{\partial(H)}{\partial x} - gH \frac{\partial\eta}{\partial x} + g\eta \frac{\partial\eta}{\partial x} - g\eta \frac{\partial(-h)}{\partial x} \\
&= g \left[-\eta \frac{\partial(H)}{\partial x} - H \frac{\partial\eta}{\partial x} + \eta \frac{\partial\eta}{\partial x} + \eta \frac{\partial(h)}{\partial x} \right] \\
&= -gH \frac{\partial\eta}{\partial x}
\end{aligned}$$

Hence, the RHS of equation (1.19) becomes:

$$-gH \frac{\partial\eta}{\partial x} + \frac{1}{\rho} H \frac{\partial\bar{\tau}_{xx}}{\partial x} + \frac{1}{\rho} H \frac{\partial\bar{\tau}_{yx}}{\partial y} - \frac{1}{\rho} \tau_x^b + \frac{1}{\rho} \tau_x^s + fH\bar{v}$$

Therefore, the depth-averaged of x -momentum equation (1.17) is:

$$\frac{\partial(H\bar{u})}{\partial t} + \frac{\partial(H\bar{u}^2)}{\partial x} + \frac{\partial(H\bar{u}\bar{v})}{\partial y} = -gH \frac{\partial\eta}{\partial x} + \frac{1}{\rho} H \frac{\partial\bar{\tau}_{xx}}{\partial x} + \frac{1}{\rho} H \frac{\partial\bar{\tau}_{yx}}{\partial y} - \frac{1}{\rho} \tau_x^b + \frac{1}{\rho} \tau_x^s + fH\bar{v}$$

By the same way, we obtain the depth-averaged of y -momentum equation (1.3) as follows:

$$\frac{\partial(H\bar{v})}{\partial t} + \frac{\partial(H\bar{v}^2)}{\partial y} + \frac{\partial(H\bar{u}\bar{v})}{\partial x} = -gH \frac{\partial\eta}{\partial y} + \frac{1}{\rho} H \frac{\partial\bar{\tau}_{yy}}{\partial y} + \frac{1}{\rho} H \frac{\partial\bar{\tau}_{xy}}{\partial x} - \frac{1}{\rho} \tau_y^b + \frac{1}{\rho} \tau_y^s - fH\bar{u}$$

Finally, we get 2D depth-averaged SWEs as the form:

$$\frac{\partial\eta}{\partial t} + \frac{\partial(H\bar{u})}{\partial x} + \frac{\partial(H\bar{v})}{\partial y} = 0 \quad (1.20)$$

$$\frac{\partial(H\bar{u})}{\partial t} + \frac{\partial(H\bar{u}^2)}{\partial x} + \frac{\partial(H\bar{u}\bar{v})}{\partial y} - fH\bar{v} = -gH \frac{\partial\eta}{\partial x} + \frac{1}{\rho} H \frac{\partial\bar{\tau}_{xx}}{\partial x} + \frac{1}{\rho} H \frac{\partial\bar{\tau}_{yx}}{\partial y} - \frac{1}{\rho} \tau_x^b + \frac{1}{\rho} \tau_x^s \quad (1.21)$$

$$\frac{\partial(H\bar{v})}{\partial t} + \frac{\partial(H\bar{v}^2)}{\partial y} + \frac{\partial(H\bar{u}\bar{v})}{\partial x} + fH\bar{u} = -gH \frac{\partial\eta}{\partial y} + \frac{1}{\rho} H \frac{\partial\bar{\tau}_{yy}}{\partial y} + \frac{1}{\rho} H \frac{\partial\bar{\tau}_{xy}}{\partial x} - \frac{1}{\rho} \tau_y^b + \frac{1}{\rho} \tau_y^s \quad (1.22)$$

1.4.4 Remarks:

1. Consider the equations (1.20)-(1.22), by using the following formula:

$$\begin{aligned} \frac{\partial(H\bar{u})}{\partial t} + \frac{\partial(H\bar{u}^2)}{\partial x} + \frac{\partial(H\bar{u}\bar{v})}{\partial y} &= H \frac{\partial(\bar{u})}{\partial t} + H\bar{u} \frac{\partial(\bar{u})}{\partial x} + H\bar{v} \frac{\partial(\bar{u})}{\partial y} + \bar{u} \underbrace{\left[\frac{\partial H}{\partial t} + \frac{\partial(H\bar{u})}{\partial x} + \frac{\partial(H\bar{v})}{\partial y} \right]}_{\text{depth integrated continuity equation} = 0} \\ &= H \frac{\partial(\bar{u})}{\partial t} + H\bar{u} \frac{\partial(\bar{u})}{\partial x} + H\bar{v} \frac{\partial(\bar{u})}{\partial y} \end{aligned}$$

We can get the system as follows:

$$\frac{\partial \eta}{\partial t} + \frac{\partial(H\bar{u})}{\partial x} + \frac{\partial(H\bar{v})}{\partial y} = 0 \quad (1.23)$$

$$\frac{H\partial(\bar{u})}{\partial t} + H\bar{u} \frac{\partial(\bar{u})}{\partial x} + H\bar{v} \frac{\partial(\bar{u})}{\partial y} - fH\bar{v} = -gH \frac{\partial \eta}{\partial x} + \frac{1}{\rho} H \frac{\partial \bar{\tau}_{xx}}{\partial x} + \frac{1}{\rho} H \frac{\partial \bar{\tau}_{yx}}{\partial y} - \frac{1}{\rho} \tau_x^b + \frac{1}{\rho} \tau_x^s \quad (1.24)$$

$$\frac{H\partial(\bar{v})}{\partial t} + H\bar{u} \frac{\partial(\bar{v})}{\partial x} + H\bar{v} \frac{\partial(\bar{v})}{\partial y} + fH\bar{u} = -gH \frac{\partial \eta}{\partial y} + \frac{1}{\rho} H \frac{\partial \bar{\tau}_{yy}}{\partial y} + \frac{1}{\rho} H \frac{\partial \bar{\tau}_{xy}}{\partial x} - \frac{1}{\rho} \tau_y^b + \frac{1}{\rho} \tau_y^s \quad (1.25)$$

2. Advection terms and Reynolds stresses

By substituting the split velocities given in Section (1.4.1) in advection terms for equation (1.21), we obtain (when calculating the random variations) :

$$\int_{-h}^{\eta} (\bar{u} + \tilde{u})(\bar{u} + \tilde{u}) dz = \int_{-h}^{\eta} \bar{u}^2 dz + \int_{-h}^{\eta} \tilde{u}\tilde{u} dz + \underbrace{2\bar{u} \int_{-h}^{\eta} \tilde{u} dz}_{=0} = H\bar{u}^2 + \int_{-h}^{\eta} \tilde{u}^2 dz$$

and

$$\int_{-h}^{\eta} (\bar{u} + \tilde{u})(\bar{v} + \tilde{v}) dz = \int_{-h}^{\eta} (\bar{u}\bar{v}) dz + \underbrace{\int_{-h}^{\eta} \tilde{u}\tilde{v} dz}_{=0} + \underbrace{\int_{-h}^{\eta} \bar{u}\tilde{v} dz}_{=0} + \int_{-h}^{\eta} \tilde{u}\tilde{v} dz = H\bar{u}\bar{v} + \int_{-h}^{\eta} \tilde{u}\tilde{v} dz$$

Thus

$$\begin{aligned} &\frac{\partial(H\bar{u})}{\partial t} + \frac{\partial(H\bar{u}^2)}{\partial x} + \frac{\partial(H\bar{u}\bar{v})}{\partial y} - fH\bar{v} \\ &= -gH \frac{\partial \eta}{\partial x} + \frac{1}{\rho} H \frac{\partial \bar{\tau}_{xx}}{\partial x} + \frac{1}{\rho} H \frac{\partial \bar{\tau}_{yx}}{\partial y} - \frac{\partial}{\partial x} \int_{-h}^{\eta} \tilde{u}^2 dz - \frac{\partial}{\partial x} \int_{-h}^{\eta} \tilde{u}\tilde{v} dz + \frac{1}{\rho} [\tau_x^b + \tau_x^s] \end{aligned}$$

where the terms $\tilde{u}_i\tilde{u}_j$ are called Reynolds stresses. Similarly the equation (1.22).

After substituting the split velocities and re-arrangements the above system, we obtain the 2D depth-integrated SWEs as follows:

$$\begin{aligned} \frac{\partial \eta}{\partial t} + \frac{\partial(H\bar{u})}{\partial x} + \frac{\partial(H\bar{v})}{\partial y} &= 0 \\ \frac{\partial(H\bar{u})}{\partial t} + \frac{\partial(H\bar{u}^2)}{\partial x} + \frac{\partial(H\bar{u}\bar{v})}{\partial y} - fH\bar{v} &= -gH\frac{\partial \eta}{\partial x} + \frac{\partial}{\partial x} \int_{-h}^{\eta} \left(\frac{\bar{\tau}_{xx}}{\rho} - \tilde{u}^2 \right) dz + \frac{\partial}{\partial y} \int_{-h}^{\eta} \left(\frac{\bar{\tau}_{yx}}{\rho} - \tilde{u}\tilde{v} \right) dz \\ &\quad - \frac{1}{\rho}\tau_x^b + \frac{1}{\rho}\tau_x^s \\ \frac{\partial(H\bar{v})}{\partial t} + \frac{\partial(H\bar{v}^2)}{\partial y} + \frac{\partial(H\bar{v}\bar{u})}{\partial x} + fH\bar{u} &= -gH\frac{\partial \eta}{\partial y} + \frac{\partial}{\partial x} \int_{-h}^{\eta} \left(\frac{\bar{\tau}_{xy}}{\rho} - \tilde{v}\tilde{u} \right) dz + \frac{\partial}{\partial y} \int_{-h}^{\eta} \left(\frac{\bar{\tau}_{yy}}{\rho} - \tilde{v}^2 \right) dz \\ &\quad - \frac{1}{\rho}\tau_y^b + \frac{1}{\rho}\tau_y^s \end{aligned}$$

Now, every simple model for the combined lateral momentum diffusion (due to turbulence) and dispersion (due to averaging out vertical velocity profile) is [79]:

$$\begin{aligned} \int_{-h}^{\eta} \left(\frac{\bar{\tau}_{xx}}{\rho} - \tilde{u}^2 \right) dz &= E_{xx} \left(\frac{\partial H\bar{u}}{\partial x} \right) \\ \int_{-h}^{\eta} \left(\frac{\bar{\tau}_{yx}}{\rho} - \tilde{v}\tilde{u} \right) dz &= E_{yx} \left(\frac{\partial H\bar{u}}{\partial y} \right) \\ \int_{-h}^{\eta} \left(\frac{\bar{\tau}_{xy}}{\rho} - \tilde{u}\tilde{v} \right) dz &= E_{xy} \left(\frac{\partial H\bar{v}}{\partial x} \right) \\ \int_{-h}^{\eta} \left(\frac{\bar{\tau}_{yy}}{\rho} - \tilde{v}^2 \right) dz &= E_{yy} \left(\frac{\partial H\bar{v}}{\partial y} \right) \end{aligned}$$

When E_{xx} , E_{xy} , E_{yx} , and E_{yy} are called the eddy dispersion coefficients (horizontal eddy viscosity), which are assumed to be constants in space (see [60, 79]).

Finally, The 2D depth-averaged of SWEs becomes:

$$\frac{\partial \eta}{\partial t} + \frac{\partial(H\bar{u})}{\partial x} + \frac{\partial(H\bar{v})}{\partial y} = 0 \quad (1.26)$$

$$\frac{\partial H\bar{u}}{\partial t} + \frac{\partial H\bar{u}^2}{\partial x} + \frac{\partial(H\bar{u}\bar{v})}{\partial y} - fH\bar{v} = -gH\frac{\partial \eta}{\partial x} + E_{xx} \left(\frac{\partial^2 H\bar{u}}{\partial^2 x^2} \right) + E_{yx} \left(\frac{\partial^2 H\bar{u}}{\partial^2 y^2} \right) - \frac{\tau_x^b}{\rho} + \frac{\tau_x^s}{\rho} \quad (1.27)$$

$$\frac{\partial(H\bar{v})}{\partial t} + \frac{\partial(H\bar{v}^2)}{\partial y} + \frac{\partial(H\bar{v}\bar{u})}{\partial x} + fH\bar{u} = -gH\frac{\partial \eta}{\partial y} + E_{xy} \left(\frac{\partial^2 H\bar{v}}{\partial^2 x^2} \right) + E_{yy} \left(\frac{\partial^2 H\bar{v}}{\partial^2 y^2} \right) - \frac{\tau_y^b}{\rho} + \frac{\tau_y^s}{\rho} \quad (1.28)$$

1.4.5 Bottom shear stress Formulas

$$\tau_x^b = \rho C_D u \sqrt{u^2 + v^2} \quad \text{and} \quad \tau_y^b = \rho C_D v \sqrt{u^2 + v^2}$$

where C_D is a coefficient often determined in one of the following formulas:

- | | |
|------------------------------|-------------------------------|
| 1. Formula of Darcy Weisbach | $C_D = \frac{1}{8} f_{DW}$ |
| 2. Formula of Chezy | $C_D = \frac{g}{C^2}$ |
| 3. Formula of Manning | $C_D = \frac{n^2 g}{H^{1/3}}$ |

Where n Manning's roughness. The value of n are given in [114].

1.5 Development of SWEs Derivation from the Second Formula of 3D Navier-Stokes Equations

Consider the system given by equations (1.5)-(1.8). By dividing ρ on all the terms of the equations and substitute $\mu = \nu\rho$, we obtain the system which is called Equations of the Governing Geophysical Flows as follows:

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

$$\frac{\partial u}{\partial t} + \left(u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} \right) - fv = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \frac{\partial}{\partial x} \left(\nu \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left(\nu \frac{\partial u}{\partial y} \right) + \frac{\partial}{\partial z} \left(\nu \frac{\partial u}{\partial z} \right) \quad (1.29)$$

$$\begin{aligned} \frac{\partial v}{\partial t} + \left(u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z} \right) + fu &= -\frac{1}{\rho} \frac{\partial p}{\partial y} + \frac{\partial}{\partial x} \left(\nu \frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial y} \left(\nu \frac{\partial v}{\partial y} \right) + \frac{\partial}{\partial z} \left(\nu \frac{\partial v}{\partial z} \right) \\ 0 &= -\frac{\partial p}{\partial z} - \rho g \end{aligned} \quad (1.30)$$

1.5.1 Results and discussions

Firstly, the depth-averaged integration of equation (1.5) is as follows:

$$\frac{\partial \eta}{\partial t} + \frac{\partial(H\bar{u})}{\partial x} + \frac{\partial(H\bar{v})}{\partial y} = 0$$

We can start integrating the equation (1.29) containing the viscous terms.

The result of integrating the LHS of equation (1.29) is: (see Section 1.4.3)

$$\frac{\partial(H\bar{u})}{\partial t} + \frac{\partial(H\bar{u}^2)}{\partial x} + \frac{\partial(H\bar{u}\bar{v})}{\partial y}$$

and integration of the Coriolis term in equation (1.29) is:

$$-\int_{-h}^{\eta} f v dz = -f \int_{-h}^{\eta} v dz = -f H \bar{v}$$

Now, we will start integrating the RHS of equation (1.29)

$$\begin{aligned} \int_{-h}^{\eta} \left(-\frac{1}{\rho} \frac{\partial p}{\partial x} + \frac{\partial}{\partial x} \left(\nu \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left(\nu \frac{\partial u}{\partial y} \right) + \frac{\partial}{\partial z} \left(\nu \frac{\partial u}{\partial z} \right) \right) dz &= \underbrace{\int_{-h}^{\eta} -\frac{1}{\rho} \frac{\partial p}{\partial x} dz}_I + \underbrace{\int_{-h}^{\eta} \frac{\partial}{\partial x} \left(\nu \frac{\partial u}{\partial x} \right) dz}_{II} \\ &+ \underbrace{\int_{-h}^{\eta} \frac{\partial}{\partial y} \left(\nu \frac{\partial u}{\partial y} \right) dz}_{III} + \underbrace{\int_{-h}^{\eta} \frac{\partial}{\partial z} \left(\nu \frac{\partial u}{\partial z} \right) dz}_{IV} \end{aligned} \quad (1.31)$$

The terms I , II , III , and IV can be simplified as follows:

$$\begin{aligned} I &= -\int_{-h}^{\eta} \frac{1}{\rho} \frac{\partial p}{\partial x} dz = -gH \frac{\partial \eta}{\partial x} \\ II &= \int_{-h}^{\eta} \frac{\partial}{\partial x} \left(\nu \frac{\partial u}{\partial x} \right) dz = \frac{\partial}{\partial x} \int_{-h}^{\eta} \left(\nu \frac{\partial u}{\partial x} \right) dz - \nu \frac{\partial u}{\partial x} \Big|_{z=\eta} \frac{\partial \eta}{\partial x} + \nu \frac{\partial u}{\partial x} \Big|_{z=-h} \frac{\partial(-h)}{\partial x} \\ &= \frac{\partial}{\partial x} \left(H \nu \frac{\partial \bar{u}}{\partial x} \right) - \nu \frac{\partial u}{\partial x} \Big|_{z=\eta} \frac{\partial \eta}{\partial x} + \nu \frac{\partial u}{\partial x} \Big|_{z=-h} \frac{\partial(-h)}{\partial x} \\ III &= \int_{-h}^{\eta} \frac{\partial}{\partial y} \left(\nu \frac{\partial u}{\partial y} \right) dz = \frac{\partial}{\partial y} \int_{-h}^{\eta} \left(\nu \frac{\partial u}{\partial y} \right) dz - \nu \frac{\partial u}{\partial y} \Big|_{z=\eta} \frac{\partial \eta}{\partial y} + \nu \frac{\partial u}{\partial y} \Big|_{z=-h} \frac{\partial(-h)}{\partial y} \\ &= \frac{\partial}{\partial y} \left(H \nu \frac{\partial \bar{u}}{\partial y} \right) - \nu \frac{\partial u}{\partial y} \Big|_{z=\eta} \frac{\partial \eta}{\partial y} + \nu \frac{\partial u}{\partial y} \Big|_{z=-h} \frac{\partial(-h)}{\partial y} \\ IV &= \int_{-h}^{\eta} \frac{\partial}{\partial z} \left(\nu \frac{\partial u}{\partial z} \right) dz = \nu \frac{\partial u}{\partial z} \Big|_{z=\eta} - \nu \frac{\partial u}{\partial z} \Big|_{z=-h} \end{aligned}$$

By arrange the terms *I*, *II*, *III*, and *IV* and combined to obtain:

$$\begin{aligned} & \int_{-h}^{\eta} \left(-\frac{1}{\rho} \frac{\partial p}{\partial x} + \frac{\partial}{\partial x} (\nu \frac{\partial u}{\partial x}) + \frac{\partial}{\partial y} (\nu \frac{\partial u}{\partial y}) + \frac{\partial}{\partial z} (\nu \frac{\partial u}{\partial z}) \right) dz = \\ & -gH \frac{\partial \eta}{\partial x} + \frac{\partial}{\partial x} (H\nu \frac{\partial \bar{u}}{\partial x}) + \frac{\partial}{\partial y} (H\nu \frac{\partial \bar{u}}{\partial y}) \\ & + \left(-\nu \frac{\partial u}{\partial x} \Big|_{z=\eta} \frac{\partial \eta}{\partial x} - \nu \frac{\partial u}{\partial y} \Big|_{z=\eta} \frac{\partial \eta}{\partial y} + \nu \frac{\partial u}{\partial z} \Big|_{z=\eta} \right) \\ & + \left(\nu \frac{\partial u}{\partial y} \Big|_{z=-h} \frac{\partial(-h)}{\partial y} + \nu \frac{\partial u}{\partial x} \Big|_{z=-h} \frac{\partial(-h)}{\partial x} - \nu \frac{\partial u}{\partial z} \Big|_{z=-h} \right) \end{aligned}$$

Now, applying the boundary condition by performed a stress balance at the surface, it can be show that:

$$\left(-\nu \frac{\partial u}{\partial x} \Big|_{z=\eta} \frac{\partial \eta}{\partial x} - \nu \frac{\partial u}{\partial y} \Big|_{z=\eta} \frac{\partial \eta}{\partial y} + \nu \frac{\partial u}{\partial z} \Big|_{z=\eta} \right) = \frac{\tau_x^w}{\rho}$$

Similarly, at the bottom

$$\left(\nu \frac{\partial u}{\partial y} \Big|_{z=-h} \frac{\partial(-h)}{\partial y} + \nu \frac{\partial u}{\partial x} \Big|_{z=-h} \frac{\partial(-h)}{\partial x} - \nu \frac{\partial u}{\partial z} \Big|_{z=-h} \right) = -\frac{\tau_x^b}{\rho}$$

Thus, the RHS of equation (1.29) can be rewrite as:

$$\begin{aligned} & \int_{-h}^{\eta} \left(-\frac{1}{\rho} \frac{\partial p}{\partial x} + \frac{\partial}{\partial x} (\nu \frac{\partial u}{\partial x}) + \frac{\partial}{\partial y} (\nu \frac{\partial u}{\partial y}) + \frac{\partial}{\partial z} (\nu \frac{\partial u}{\partial z}) \right) dz = -gH \frac{\partial \eta}{\partial y} + \frac{\partial}{\partial x} (H\nu \frac{\partial \bar{u}}{\partial x}) + \frac{\partial}{\partial y} (H\nu \frac{\partial \bar{u}}{\partial y}) \\ & + \frac{\tau_x^w}{\rho} - \frac{\tau_x^b}{\rho} \end{aligned}$$

Finally, we can be written the depth-averaged of the equation (1.29) as follows:

$$\frac{\partial(H\bar{u})}{\partial t} + \frac{\partial(H\bar{u}^2)}{\partial x} + \frac{\partial(H\bar{u}\bar{v})}{\partial y} - fH\bar{v} = -gH \frac{\partial \eta}{\partial x} + \frac{\partial}{\partial x} (H\nu \frac{\partial \bar{u}}{\partial x}) + \frac{\partial}{\partial y} (H\nu \frac{\partial \bar{u}}{\partial y}) + \frac{\tau_x^w}{\rho} - \frac{\tau_x^b}{\rho}$$

By the same way, we can obtain the depth-averaged of the equation (1.30) as follows:

$$\frac{\partial(H\bar{v})}{\partial t} + \frac{\partial(H\bar{u}\bar{v})}{\partial x} + \frac{\partial(H\bar{v}^2)}{\partial y} + fH\bar{u} = -gH \frac{\partial \eta}{\partial y} + \frac{\partial}{\partial x} (H\nu \frac{\partial \bar{v}}{\partial x}) + \frac{\partial}{\partial y} (H\nu \frac{\partial \bar{v}}{\partial y}) + \frac{\tau_y^w}{\rho} - \frac{\tau_y^b}{\rho}$$

The results from the depth-averaged of the development 2D depth-averaged SWEs as follows:

$$\frac{\partial \eta}{\partial t} + \frac{\partial(H\bar{u})}{\partial x} + \frac{\partial(H\bar{v})}{\partial y} = 0 \quad (1.32)$$

$$\frac{\partial(H\bar{u})}{\partial t} + \frac{\partial(H\bar{u}^2)}{\partial x} + \frac{\partial(H\bar{u}\bar{v})}{\partial y} - fH\bar{v} = -gH\frac{\partial \eta}{\partial x} + \nu H \left[\left(\frac{\partial^2 \bar{u}}{\partial x^2} \right) + \left(\frac{\partial^2 \bar{u}}{\partial y^2} \right) \right] + \frac{\tau_x^w}{\rho} - \frac{\tau_x^b}{\rho} \quad (1.33)$$

$$\frac{\partial(H\bar{v})}{\partial t} + \frac{\partial(H\bar{u}\bar{v})}{\partial x} + \frac{\partial(H\bar{v}^2)}{\partial y} + fH\bar{u} = -gH\frac{\partial \eta}{\partial y} + \nu H \left[\left(\frac{\partial^2 \bar{v}}{\partial x^2} \right) + \left(\frac{\partial^2 \bar{v}}{\partial y^2} \right) \right] + \frac{\tau_y^w}{\rho} - \frac{\tau_y^b}{\rho} \quad (1.34)$$

Remark

From now, we will only consider the mean velocities and neglect the random variations (i.e., we can put u, v instead of \bar{u} and \bar{v}).

1.6 Case 2: Derivation 2DSWEs Without Using the Depth Averaged Technique

Consider the x -momentum equation:

$$\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x} + \frac{\partial(uv)}{\partial y} + \frac{\partial(uw)}{\partial z} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{xx}}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y} + \frac{1}{\rho} \frac{\partial \tau_{zx}}{\partial z}$$

The depth-averaged equation for the above equation is:

$$\underbrace{\int_{-h}^{\eta} \frac{\partial u}{\partial t} dz}_I + \underbrace{\int_{-h}^{\eta} \frac{\partial u^2}{\partial x} dz}_II + \underbrace{\int_{-h}^{\eta} \frac{\partial(uv)}{\partial y} dz}_III + \underbrace{\int_{-h}^{\eta} \frac{\partial(uw)}{\partial z} dz}_IV = \underbrace{\int_{-h}^{\eta} -\frac{1}{\rho} \frac{\partial p}{\partial x} dz}_V + \underbrace{\int_{-h}^{\eta} \left[\frac{1}{\rho} \frac{\partial \tau_{xx}}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y} + \frac{1}{\rho} \frac{\partial \tau_{zx}}{\partial z} \right] dz}_VI \quad (1.35)$$

with $p = \rho g(\eta - z)$

$$\frac{\partial p}{\partial x} = \rho g \frac{\partial \eta}{\partial x} + g(\eta - z) \frac{\partial \rho}{\partial x} \quad (1.36)$$

Now, substitute equation (1.36) to equation (1.35), we obtain:

$$\underbrace{\int_{-h}^{\eta} \frac{\partial u}{\partial t} dz}_I + \underbrace{\int_{-h}^{\eta} \frac{\partial u^2}{\partial x} dz}_II + \underbrace{\int_{-h}^{\eta} \frac{\partial uv}{\partial y} dz}_III + \underbrace{\int_{-h}^{\eta} \frac{\partial uw}{\partial z} dz}_IV = \underbrace{\int_{-h}^{\eta} -\frac{1}{\rho} (\rho g) \frac{\partial \eta}{\partial x} dz - \frac{g}{\rho} (\eta - z) \frac{\partial \rho}{\partial x} dz}_V + \underbrace{\int_{-h}^{\eta} \left[\frac{1}{\rho} \frac{\partial \tau_{xx}}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y} + \frac{1}{\rho} \frac{\partial \tau_{zx}}{\partial z} \right] dz}_VI$$

That implies:

$$\begin{aligned}
& \underbrace{\int_{-h}^{\eta} \frac{\partial u}{\partial t} dz}_I + \underbrace{\int_{-h}^{\eta} \frac{\partial u^2}{\partial x} dz}_II + \underbrace{\int_{-h}^{\eta} \frac{\partial uv}{\partial y} dz}_III + \underbrace{\int_{-h}^{\eta} \frac{\partial uw}{\partial z} dz}_IV = \\
& \underbrace{\int_{-h}^{\eta} -g \frac{\partial \eta}{\partial x} dz - \int_{-h}^{\eta} -\frac{g}{\rho} (\eta - z) \frac{\partial \rho}{\partial x} dz}_V + \underbrace{\int_{-h}^{\eta} \left[\frac{1}{\rho} \frac{\partial \tau_{xx}}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y} + \frac{1}{\rho} \frac{\partial \tau_{zx}}{\partial z} \right] dz}_VI
\end{aligned}$$

The terms I, II, \dots, VI can be simplified using Leibniz rules as follows:

$$\begin{aligned}
I &= \int_{-h}^{\eta} \frac{\partial u}{\partial t} dz = \frac{\partial}{\partial t} \int_{-h}^{\eta} u dz + u(-h) \frac{\partial(-h)}{\partial t} - u(\eta) \frac{\partial(\eta)}{\partial t} = \frac{\partial}{\partial t} (u|_{-h}^{\eta}) - u(\eta) \frac{\partial \eta}{\partial t} \\
&= \frac{\partial}{\partial t} [u(\eta + h)] - u(\eta) \frac{\partial \eta}{\partial t} \\
II &= \int_{-h}^{\eta} \frac{\partial u^2}{\partial x} dz = \frac{\partial}{\partial x} \int_{-h}^{\eta} u^2 dz + u^2(-h) \frac{\partial(-h)}{\partial x} - u^2(\eta) \frac{\partial \eta}{\partial x} \\
&= \frac{\partial}{\partial x} [u^2 \sigma_{xx}(\eta + h)] - u^2(\eta) \frac{\partial \eta}{\partial x} \\
III &= \int_{-h}^{\eta} \frac{\partial uv}{\partial y} dz = \frac{\partial}{\partial y} \int_{-h}^{\eta} uv dz + uv(-h) \frac{\partial(-h)}{\partial y} - uv(\eta) \frac{\partial \eta}{\partial y} \\
&= \frac{\partial}{\partial y} [\sigma_{yx}(\eta + h)uv] - uv(\eta) \frac{\partial \eta}{\partial y} \\
IV &= \int_{-h}^{\eta} \frac{\partial uw}{\partial z} dz = uw(\eta) - uw(-h) = 0
\end{aligned}$$

$$V = - \int_{-h}^{\eta} g \frac{\partial \eta}{\partial x} dz = -g \frac{\partial \eta}{\partial x} \Big|_{-h}^{\eta} = -g \frac{\partial \eta}{\partial x} (\eta + h)$$

$$V = - \int_{-h}^{\eta} \frac{g}{\rho} (\eta - z) \frac{\partial \rho}{\partial x} dz = -\frac{1}{2} \frac{g}{\rho} (\eta + h)^2 \frac{\partial \rho}{\partial x}$$

$$\begin{aligned}
VI &= \int_{-h}^{\eta} \left[\frac{1}{\rho} \frac{\partial \tau_{xx}}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y} + \frac{1}{\rho} \frac{\partial \tau_{zx}}{\partial z} \right] dz = \frac{1}{\rho} \frac{\partial \tau_{xx}}{\partial x} \Big|_{-h}^{\eta} + \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y} \Big|_{-h}^{\eta} + \frac{1}{\rho} \tau_{zx} \Big|_{-h}^{\eta} \\
&= \frac{1}{\rho} \frac{\partial \tau_{xx}}{\partial x} (\eta + h) + \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y} (\eta + h) + \frac{1}{\rho} \tau_{zx}(\eta) - \frac{1}{\rho} \tau_{zx}(-h)
\end{aligned}$$

Re-arranging the above equations:

$$\begin{aligned} & \left[\frac{\partial}{\partial t} [u(\eta + h)] - u(\eta) \frac{\partial \eta}{\partial t} + \frac{\partial}{\partial x} (\sigma_{xx}(\eta + h)u^2) - u^2(\eta) \frac{\partial \eta}{\partial x} + \frac{\partial}{\partial y} (\sigma_{yx}(\eta + h)uv) - uv(\eta) \frac{\partial \eta}{\partial y} \right] \\ & = \left[-g \frac{\partial \eta}{\partial x} (\eta + h) - \frac{1}{2} \frac{g}{\rho} (\eta + h)^2 \frac{\partial \rho}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{xx}}{\partial x} (\eta + h) + \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y} (\eta + h) + \frac{1}{\rho} \tau_{zx}(\eta) - \frac{1}{\rho} \tau_{zx}(-h) \right] \end{aligned}$$

That implies:

$$\begin{aligned} & \left[\frac{\partial}{\partial t} u(\eta + h) - u(\eta) \frac{\partial \eta}{\partial t} + \frac{\partial}{\partial x} [\sigma_{xx}(h + \eta)u^2] - u \cdot u(\eta) \frac{\partial \eta}{\partial x} + \frac{\partial}{\partial y} [\sigma_{yx}(\eta + h)uv] - uv(\eta) \frac{\partial \eta}{\partial y} \right] \\ & = \left[-g \frac{\partial \eta}{\partial x} (\eta + h) - \frac{1}{2} \frac{g}{\rho} (\eta + h)^2 \frac{\partial \rho}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{xx}}{\partial x} (\eta + h) + \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y} (\eta + h) + \frac{1}{\rho} \tau_{zx}(\eta) - \frac{1}{\rho} \tau_{zx}(-h) \right] \end{aligned}$$

Thus

$$\begin{aligned} & \left[\frac{\partial}{\partial t} (u(\eta + h)) + \frac{\partial}{\partial x} [\sigma_{xx}(\eta + h)u^2] + \frac{\partial}{\partial y} [\sigma_{yx}(\eta + h)uv] - \underbrace{u \left[\eta \frac{\partial \eta}{\partial t} + u(\eta) \frac{\partial \eta}{\partial x} + v(\eta) \frac{\partial \eta}{\partial y} \right]}_{=0} \right] \\ & = \left[-g \frac{\partial \eta}{\partial x} (\eta + h) - \frac{1}{2} \frac{g}{\rho} (\eta + h)^2 \frac{\partial \rho}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{xx}}{\partial x} (\eta + h) + \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y} (\eta + h) + \frac{1}{\rho} \tau_{zx}(\eta) - \frac{1}{\rho} \tau_{zx}(-h) \right] \end{aligned}$$

when $\frac{\partial(-h)}{\partial x} = 0$, $\frac{\partial(-h)}{\partial y} = 0$ and $\sigma = 1$ therefore $\frac{\partial \eta}{\partial x} = \frac{\partial H}{\partial x}$.

Thus:

$$\frac{\partial}{\partial t} (Hu) + \frac{\partial}{\partial x} (Hu^2) + \frac{\partial}{\partial y} (Huv) = -gH \frac{\partial \eta}{\partial x} - \frac{1}{2} \frac{g}{\rho} (H)^2 \frac{\partial \rho}{\partial x} + \frac{1}{\rho} H \frac{\partial \tau_{xx}}{\partial x} + \frac{1}{\rho} H \frac{\partial \tau_{yx}}{\partial y} + \frac{1}{\rho} \tau_{zx}(\eta) - \frac{1}{\rho} \tau_{zx}(-h)$$

Then, we can obtain

$$\frac{\partial}{\partial t} (Hu) + \frac{\partial}{\partial x} (Hu^2) + \frac{\partial}{\partial y} (Huv) = -gH \frac{\partial \eta}{\partial x} - \frac{1}{2} \frac{g}{\rho} (H)^2 \frac{\partial \rho}{\partial x} + \frac{1}{\rho} H \frac{\partial \tau_{xx}}{\partial x} + \frac{1}{\rho} H \frac{\partial \tau_{yx}}{\partial y} + \frac{1}{\rho} \tau_x^s - \frac{1}{\rho} \tau_x^b$$

We define $\tau_{xz}(\eta) = \tau_x^s$ to the surface shear stress, $\tau_{xz}(-h) = \tau_x^b$ to the bottom shear stress.

Similarly, we can do the y -momentum equation

Finally, we get 2D shallow water equations as follows:

$$\frac{\partial \eta}{\partial t} + \frac{\partial (Hu)}{\partial x} + \frac{\partial (Hv)}{\partial y} = 0 \quad (1.37)$$

$$\frac{\partial}{\partial t} (Hu) + \frac{\partial}{\partial x} (Hu^2) + \frac{\partial}{\partial y} (Huv) = -gH \frac{\partial \eta}{\partial x} + \frac{1}{\rho} H \frac{\partial \tau_{xx}}{\partial x} + \frac{1}{\rho} H \frac{\partial \tau_{yx}}{\partial y} + \frac{1}{\rho} \tau_x^s - \frac{1}{\rho} \tau_x^b \quad (1.38)$$

$$\frac{\partial}{\partial t}(Hv) + \frac{\partial}{\partial y}(Hv^2) + \frac{\partial}{\partial x}(Huv) = -gH\frac{\partial\eta}{\partial y} + \frac{1}{\rho}H\frac{\partial\tau_{xy}}{\partial x} + \frac{1}{\rho}H\frac{\partial\tau_{yy}}{\partial y} + \frac{1}{\rho}\tau_y^s - \frac{1}{\rho}\tau_y^b \quad (1.39)$$

1.7 The General Formula of Non-Conservative Form for 2DSWEs

The general formula of 2DSWEs for non-conservative form that appears as follows:

$$\frac{\partial U}{\partial t} + \frac{\partial(E.U)}{\partial x} + \frac{\partial(G.U)}{\partial y} = S$$

where

$$U = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix} = \begin{pmatrix} \eta \\ u \\ v \end{pmatrix}, \quad E = \begin{bmatrix} E_1 \\ E_2 \\ E_3 \end{bmatrix} = \begin{pmatrix} u & H & 0 \\ g & u & 0 \\ 0 & 0 & u \end{pmatrix}$$

$$G = \begin{bmatrix} G_1 \\ G_2 \\ G_3 \end{bmatrix} = \begin{pmatrix} v & H & 0 \\ 0 & v & 0 \\ g & 0 & v \end{pmatrix}, \quad S = \begin{bmatrix} -u\frac{\partial h}{\partial x} - v\frac{\partial h}{\partial y} \\ \nu \left[\frac{\partial}{\partial x} \left(\frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left(\frac{\partial u}{\partial y} \right) \right] + fv + \frac{\tau_u^w}{H\rho} - \frac{\tau_u^b}{H\rho} \\ \nu \left[\frac{\partial}{\partial x} \left(\frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial y} \left(\frac{\partial v}{\partial y} \right) \right] - fu + \frac{\tau_v^w}{H\rho} - \frac{\tau_v^b}{H\rho} \end{bmatrix}$$

Where U means the vector of conserved variables, E is the vector of flux in x -direction, G is the vector of flux in y -direction and the term S may include various source terms such as bed friction, bed topography, wind stress, viscosity and the Coriolis parameter.

1.8 Summary and Conclusion

This chapter has highlighted a study of 2D depth-averaged non-linear shallow water equations. Two ways to derive this model were suggested using 3D Navier-Stokes equations. Firstly, we got system of 2DSWEs given in equations (1.26)-(1.28) using splitting of velocity and horizontal eddy viscosity from the first formula of 3DNSEs.

Secondly, another form of 2DSWEs was introduced in equations (1.32)-(1.34) using the second formula of 3DNSEs . Another way to derive 2DSWEs without using depth-averaged technique was presented in equations (1.37)-(1.39).

The new derivation of 2DSWEs model, including a particular viscous term, turbulent friction term, bed slope source term, Coriolis effects and capillary effects on the free surface is stressed out. For all applications seeking for steady-state solutions, the water surface elevation is usually constant.

Chapter 2

An Explicit Staggered Finite Difference Method for 2DSWEs (Numerical Techniques for SWEs)

Chapter 2

An Explicit Staggered Finite Difference Method for 2DSWEs

Some of the results presented in this chapter (Sections 2.4, 2.5 and 2.8) are the subject of an article [2].

The increased interest recently in the ocean modeling has led to exciting developments of mathematical modeling including the dispersion of pollutants from the coastal area, chemical, and ecological dynamics. One way to study the dynamic behavior of water is to develop the numerical methods for the oceans models.

Many finite difference methods have been first developed to solve these models in 2DSWEs framework. The time integration methods for these techniques used in this work are explicit. The development of these methods depends on the choice of techniques and the type of methods used, so most of the numerical schemes used to simulate ocean models are based on shallow water equations.

This chapter is organized as follows: Firstly, A review of an explicit finite difference method is presented in Section 2.1. Shallow flow models are introduced and propose numerous techniques using new spatial and temporal discretization in a standard C-grid using EFDM in space with a time staggered grid using leapfrog combined with simple Robert-Asselin filter are presented in Sections 2.3-2.6. New algorithms are established to implement the proposed technique. Finally, Open boundary conditions are studied in Section 2.9.

Highlights

- Study the development of the numerical methods for the ocean models.
- Some new algorithms are implemented for 2DSWEs.
- The Courant Friedrichs Lewy (CFL) condition for 2DSWEs is applied.

Contents

2.1	Overview of some numerical methods	69
2.1.1	Finite difference method	69
2.1.2	Arakawa C-grid	71
2.1.3	Finite difference method for staggered grid (The temporal grid)	71
2.2	Time-differencing schemes	72
2.2.1	Leapfrog scheme with Robert-Asselin Filter	72
2.3	Numerical discretization of 2D shallow water models when the time step $(n+1)$	73
2.3.1	Water level scheme	73
2.3.2	uv momentums scheme	74
2.3.3	Computational algorithm	77
2.4	Another way to numerical discretization of 2DSWEs when the time step $(n+1/2)$	77
2.4.1	Computational Algorithm in case the time step $(n+1/2)$	78
2.5	Numerical discretization for 2DSWEs using EFDMs in several cases	85
2.6	Numerical discretization for 2D linear SWEs using EFDMs	88
2.7	Program logic and data flow	90
2.8	Stability criteria	91
2.8.1	Consistency, convergence and stability	91
2.8.2	The Courant Friedrichs Lewy (CFL) Condition	92
2.8.3	The Courant Friedrichs Lewy (CFL) condition for 2DSWEs	93
2.8.4	Derivation the Courant Friedrichs Lewy (CFL) condition in 2D shallow water equations	93
2.9	Studying Boundary conditions	96
2.9.1	Radiative open boundary condition	95
2.10	Summary and Conclusion	100

2.1 Overview of some numerical methods

There are many techniques to bring an approximation to the analytical solution of partial differential equations in fluid dynamics such that finite difference method, finite volume method and finite element method which are able to give excellent results. In this work, an explicit center finite difference method and leapfrog scheme with Robert-Asselin filter are only concerned.

2.1.1 Finite difference method

Finite difference method is the oldest technique for calculating of dynamic fluid and has been seen in publications as early as (1928) with the basic theoretical paper by Courant, Friedrichs, and Lewy. Computational experiments were generated by using a finite difference approach [22, 87, 112].

The finite difference technique is the most widely used in 2D shallow water models. The major idea of the finite difference methods, the continuous differential operators of PDEs is replaced by the discrete difference operators. This requires discretization of the geometry, such that the difference operators can be written in terms of the grid points (see [17, 28, 88, 89]).

In this section, the most commonly used forward, centered and backward difference approximations are described. These types of approximations are used in the numerical discretization for 2D shallow water models.

The most commonly used the **Forward difference approximation with the first order is found namely**

$$\left(\frac{\partial u}{\partial x}\right)_i \simeq \frac{u_{i+1} - u_i}{\Delta x} + O(\Delta x)$$

• Centered difference

The centered difference approximation for the first derivative with is second-order convergent as the form:

$$\left(\frac{\partial u}{\partial x}\right)_i \simeq \frac{u_{i+1} - u_{i-1}}{2\Delta x} + O(\Delta x)^2$$

• Backward difference

The backward difference approximation as the form:

$$\left(\frac{\partial u}{\partial x}\right)_i \simeq \frac{u_{i-1} - u_i}{\Delta x} + O(\Delta x)$$

- **Centered difference**

The centered difference approximation for the second derivative as the form:

$$\left(\frac{\partial^2 u}{\partial x^2}\right)_i \simeq \frac{u_{i+1} - 2u_i + u_{i-1}}{(\Delta x)^2} + O(\Delta x)^2$$

In order to make a choice for an explicit finite difference method, we adopt the following criteria (see [18, 26, 75]).

1. Numerical solutions must be sufficiently accurate. Thus, the method should be consistent and stable. In according to practical experiments, second-order accuracy is satisfactory. It is also necessary that the numerical solution is not greatly affected by false solutions and rounding errors.
2. The proposed method should be strong. In our case, this means that the method should be applicable to a wide range of 2D flow problems in ocean modeling such as tidal problems in coastal seas, model problems in tidal tracts, or steady-state problems in rivers.
3. Suggested method must be computationally effective. The efficiency should not be achieved at the expense of durability, so durability has a high priority.
4. The numerical treatment of boundary conditions should be such that the overall accuracy and efficiency are not greatly reduced.

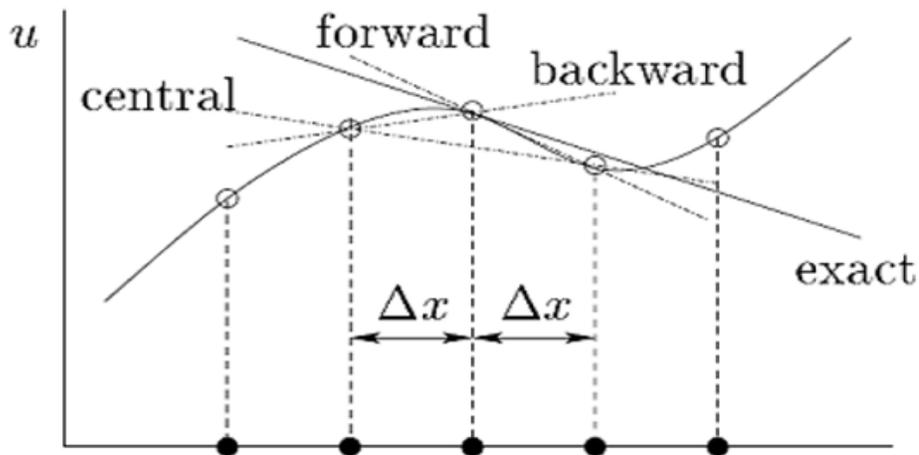


Figure 2-1: Finite difference method.

2.1.2 Arakawa C-grid

The performance of EFDM of horizontal space derivatives depends on the distribution of dependent variables on the grid [10, 11, 118]. Various types of grids have been developed and used by different authors for different purposes. Arakawa and Lamb investigate four different spatially staggered grids which are used for shallow water equations.

Figure 2-2 shows Arakawa staggered C-grid, u -components located at the centers of the left and right grid faces, v -components located at the centers of the upper and lower grid faces and η located at the center grid.

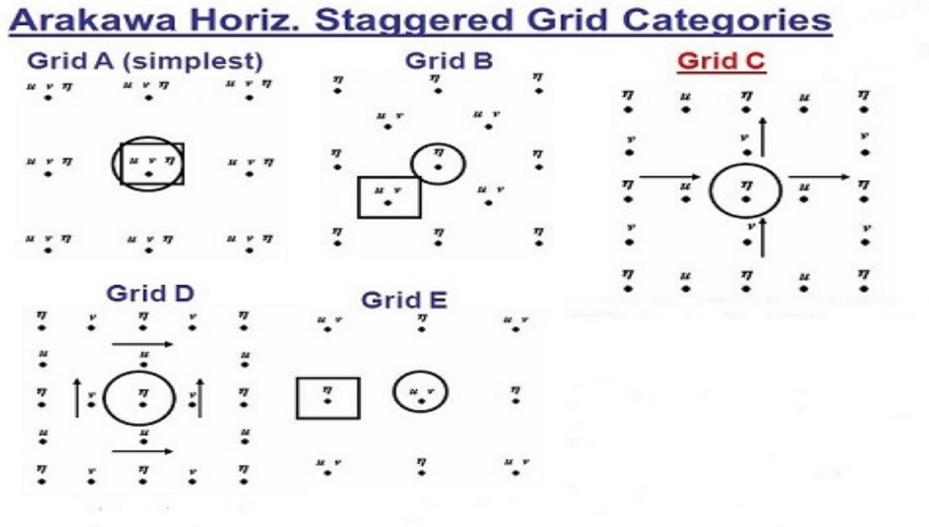


Figure 2-2: Arakawa staggered C-grid.

2.1.3 Finite difference method for staggered grid (the temporal grid)

The general finite difference method used is a three-level in time that is each time step includes three-time levels: the entry level, the half-time step level, and the full-time step level. As this case shows, in each time step one calculation of η is carried out for each set of calculations of u and v . This calculation uses η^{n-1} , η^n , u^n and v^n to provide a time η^{n+1} . To obtain η^{n+2} the values η^{n+1} , u^{n+1} , v^{n+1} and η^n are used.

Consider 2DSWEs on the rectangular space domain $\Omega := [0, L_x] \times [0, L_y]$ and the time interval $(0, T)$. Periodic boundary conditions are prescribed. The domain is meshed with a rectilinear grid of $N_x \times N_y$ cells. We set $M := \{1, 2, \dots, N_x\} \times \{1, 2, \dots, N_y\}$. The cell of coordinates (i, j) is denoted by $K_{i,j}$. The cell $K_{i,j}$ is denoted by $x_{i,j}$. The bottom edge, top edge, left edge and right edge are denoted by $x_{i,j-1}$, $x_{i,j+1}$, $x_{i-1,j}$ and $x_{i+1,j}$ respectively. The water height η is discretized at the center cells. The velocity in the x -direction is discretized at the edges normal to the x -direction, while the

velocity in the y -direction is discretized at the edges normal to the y -direction. The approximation of η at point $x_{i,j}$ and at time t_n is denoted by $\eta_{i,j}^n$. The approximation of u at point $x_{i+1,j}$ and time t_n is denoted by $u_{i+1,j}^n$, while the approximation of v at point $x_{i,j+1}$ and time t_n is denoted by $v_{i,j+1}^n$.

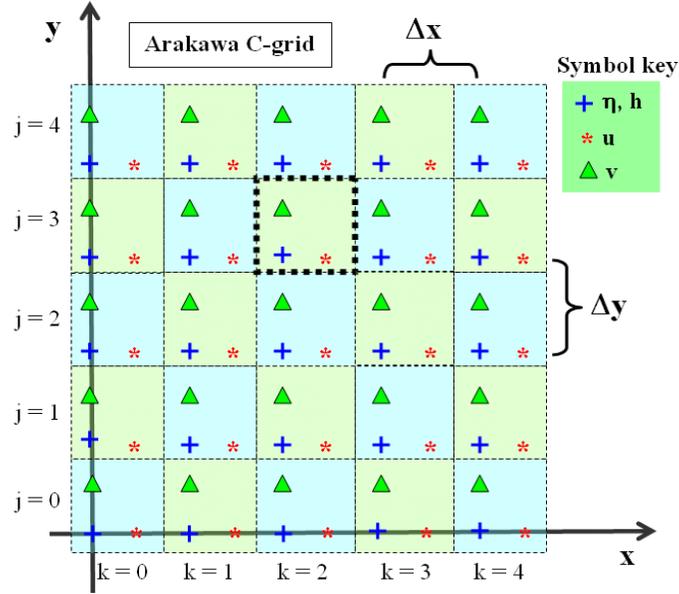


Figure 2-3: Finite difference method in Arakawa C-grid.

2.2 Time-Differencing Schemes

An explicit leapfrog method is most applied in combination with Robert-Asselin filter to smash the computational model and prevent time splitting. In this section, we discuss leapfrog method, Robert-Asselin filter and apply it to the leapfrog method.

2.2.1 Leapfrog scheme with Robert-Asselin filter

The leapfrog method can be expressed using the model

$$x_{n+1} = x_{n-1} + 2\Delta t F(x_n)$$

where x_{n+1} is the approximate solution that will be determined due to the other variables.

The leapfrog scheme uses information at three different time levels t_{n-1} , t_n and t_{n+1} with a total difference of $2\Delta t$. The numerical approximation to the derivative is known as a centered difference scheme. One problem with this scheme is the t_{n-1} time level as we only know our initial variable at time t_n and not also t_{n-1} . This is overcome by using a simple forward step first. Then it is possible to implement the integration using the leapfrog scheme. The leapfrog scheme yields second-order accuracy. The simple leapfrog scheme is unstable but can be stabilized using the Robert-Asselin filter [119].

Robert-Asselin filter was designed specifically for the leapfrog method in (1966) by Robert and in (1972) Asselin showed that it relieves the computational mode but leaves the physical style comparatively undamped [12, 34, 119, 120]. Since then it has become known as Robert-Asselin filter. After each leapfrog step, the filter mixes solutions from three successive time points at t_{n-1} , t_n and t_{n+1} which can be seen from Figure 2-4. The solution at the inner point at time t_n is displaced by:

$$d = \frac{\gamma}{2} [x_{n-1} - 2x_n + x_{n+1}]$$

where γ is the filter parameter and the values x_{n-1} , x_n and x_{n+1} correspond to the time points t_{n-1} , t_n and t_{n+1} respectively. Typically the filter parameter γ is taken to be **0.01**.

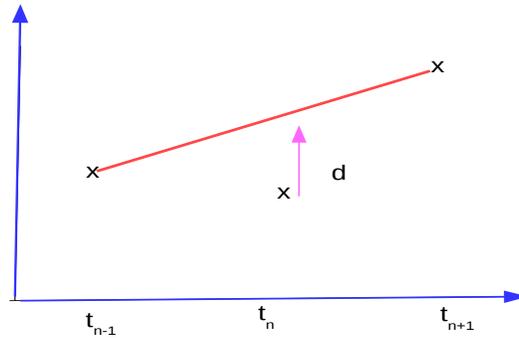


Figure 2-4: The standard Robert-Asselin filter.

2.3 Numerical Discretization of 2D Shallow Water Models when the Time Step ($n+1$)

In this section, the numerical discretization of 2DNSWEs is applied using EFDMs in a standard C-grid spacing with a time staggered grid using leapfrog combined with simple Robert-Asselin filtering.

2.3.1 Water level scheme

The finite difference first-order numerical scheme for the water level in the continuity equation (1.10) writes out:

$$\frac{\partial \eta}{\partial t} = - [(Hu)_{i+1/2,j} - (Hu)_{i-1/2,j}] / \Delta x - [(Hv)_{i,j+1/2} - (Hv)_{i,j-1/2}] / \Delta y$$

and centered finite difference scheme in space term writes down:

$$(Hu)_{i+1/2,j} = (H_{i,j} + H_{i+1,j})/2u_{i+1,j}$$

$$(Hu)_{i-1/2,j} = (H_{i-1,j} + H_{i,j})/2u_{i,j}$$

$$(Hv)_{i,j+1/2} = (H_{i,j} + H_{i,j+1})/2v_{i,j+1}$$

$$(Hv)_{i,j-1/2} = (H_{i,j-1} + H_{i,j})/2v_{i,j}$$

Thus, the full water level centered finite difference in space numerical scheme is:

$$\begin{aligned} \frac{\partial \eta}{\partial t} = & - [(H_{i,j} + H_{i+1,j})/2u_{i+1,j} - (H_{i-1,j} + H_{i,j})/2u_{i,j}] / \Delta x \\ & - [(H_{i,j} + H_{i,j+1})/2v_{i,j+1} - (H_{i,j-1} + H_{i,j})/2v_{i,j}] / \Delta y \end{aligned}$$

2.3.2 u and v momentums schemes

For the x -momentum given by equation (1.11), the first-order spatial discretization writes:

$$\begin{aligned} \frac{\partial(Hu)}{\partial t} = & - [(Huu)_{i+1/2,j} - (Huu)_{i-1/2,j}] / \Delta x - [(Huv)_{i,j+1/2} - (Huv)_{i,j-1/2}] / \Delta y + f(Hv) \\ & + \nu \left[(H \frac{\partial u}{\partial x})_{i+1/2,j} - (H \frac{\partial u}{\partial x})_{i-1/2,j} \right] / \Delta x + \nu \left[(H \frac{\partial u}{\partial y})_{i,j+1/2} - (H \frac{\partial u}{\partial y})_{i,j-1/2} \right] / \Delta y \\ & - gH(\eta_{i+1/2,j} - \eta_{i-1/2,j}) / \Delta x + \frac{\rho_a}{\rho_0} C_a u_{10} \sqrt{u_{10}^2 + v_{10}^2} - C_D u \sqrt{u^2 + v^2} \end{aligned}$$

Thus

$$(Huu)_{i+1/2,j} = H_{i,j}(u_{i+1,j} + u_{i,j})^2/4$$

$$(Huv)_{i-1/2,j} = H_{i-1,j}(u_{i-1,j} + u_{i,j})^2/4$$

$$(Huv)_{i,j+1/2} = (H_{i-1,j} + H_{i,j} + H_{i-1,j+1} + H_{i,j+1}) \times (u_{i,j+1} + u_{i,j})(v_{i-1,j+1} + v_{i,j+1})/16$$

$$(Huv)_{i,j-1/2} = (H_{i-1,j} + H_{i,j} + H_{i-1,j-1} + H_{i,j-1}) \times (u_{i,j} + u_{i,j-1})(v_{i-1,j} + v_{i,j})/16$$

$$f(Hv) = f(H_{i,j} + H_{i-1,j})/2 \times (v_{i-1,j} + v_{i,j} + v_{i,j+1} + v_{i-1,j+1})/4$$

and

$$\begin{aligned}
\nu \left[\left(H \frac{\partial u}{\partial x} \right)_{i+1/2,j} - \left(H \frac{\partial u}{\partial x} \right)_{i-1/2,j} \right] &= \nu \left[H \frac{u_{i+1,j} - u_{i,j}}{\Delta x} - H \frac{u_{i-1,j} - u_{i,j}}{\Delta x} \right] \\
\nu \left[\left(H \frac{\partial u}{\partial y} \right)_{i,j+1/2} - \left(H \frac{\partial u}{\partial y} \right)_{i,j-1/2} \right] &= \nu \left[H \frac{u_{i,j+1} - u_{i,j}}{\Delta y} - H \frac{u_{i,j} - u_{i,j-1}}{\Delta y} \right] \\
gH(\eta_{i+1/2,j} - \eta_{i-1/2,j}) &= g(H + H_{i-1,j})/2(\eta_{i,j} - \eta_{i-1,j}) \\
C_D u \sqrt{u^2 + v^2} &= C_D u_{i,j} \sqrt{(u_{i,j})^2 + (v_{i-1,j} + v_{i,j} + v_{i,j+1} + v_{i-1,j+1})^2}
\end{aligned}$$

Hence, rewriting the full x -momentum, we get:

$$\begin{aligned}
\frac{\partial(Hu)}{\partial t} &= - \left[H(u_{i+1,j} + u_{i,j})^2/4 - H_{i-1,j}(u_{i-1,j} + u_{i,j})^2/4 \right] / \Delta x \\
&\quad - \left[(H_{i-1,j} + H_{i,j} + H_{i-1,j+1} + H_{i,j+1}) \times (u_{i,j+1} + u_{i,j})(v_{i-1,j+1} + v_{i,j+1})/16 \right] / \Delta x \\
&\quad + \left[(H_{i-1,j} + H_{i,j} + H_{i-1,j-1} + H_{i,j-1})(u_{i,j} + u_{i,j-1})(v_{i-1,j} + v_{i,j})/16 \right] / \Delta y \\
&\quad + f(H_{i,j} + H_{i-1,j})/2 \times (v_{i-1,j} + v_{i,j} + v_{i,j+1} + v_{i-1,j+1})/4 \\
&\quad + \nu \left[H \frac{u_{i+1,j} - u_{i,j}}{\Delta x} - H \frac{u_{i,j} - u_{i-1,j}}{\Delta x} \right] / \Delta x \\
&\quad + \nu \left[H \frac{u_{i,j+1} - u_{i,j}}{\Delta y} - H \frac{u_{i,j} - u_{i,j-1}}{\Delta y} \right] / \Delta y \\
&\quad - g(H_{i,j} + H_{i-1,j})/2(\eta_{i,j} - \eta_{i-1,j}) / \Delta x + \rho_a / \rho_0 C_a u_{10} \sqrt{u_{10}^2 + v_{10}^2} \\
&\quad - C_D u_{i,j} \sqrt{(u_{i,j})^2 + (v_{i-1,j} + v_{i,j} + v_{i,j+1} + v_{i-1,j+1})^2}
\end{aligned}$$

Similarly, we can get the full y -momentum scheme by using clever symmetry one-to-one relations with x -momentum scheme which are as the following.

$$\begin{aligned}
\frac{\partial(Hv)}{\partial t} &= - \left[H(v_{i^*,j^*+1} + v_{i^*,j^*})^2/4 - H_{i^*,j^*-1}(v_{i^*,j^*-1} + v_{i^*,j^*})^2/4 \right] / \Delta y \\
&\quad - \left[(H_{i^*,j^*-1} + H_{i^*,j^*} + H_{i^*+1,j^*-1} + H_{i^*+1,j^*}) \times (v_{i^*+1,j^*} + v_{i^*,j^*})(u_{i^*+1,j^*-1} + u_{i^*+1,j^*})/16 \right] / \Delta y \\
&\quad + \left[(H_{i^*,j^*-1} + H_{i^*,j^*} + H_{i^*-1,j^*-1} + H_{i^*-1,j^*})(v_{i^*,j^*} + v_{i^*-1,j^*})(u_{i^*,j^*-1} + u_{i^*,j^*})/16 \right] / \Delta x \\
&\quad - f(H_{i^*,j^*} + H_{i^*,j^*-1})/2 \times (u_{i^*,j^*-1} + u_{i^*,j^*} + u_{i^*+1,j^*-1} + u_{i^*+1,j^*})/4 \\
&\quad + \nu \left[H \frac{v_{i^*,j^*+1} - v_{i^*,j^*}}{\Delta y} - H \frac{v_{i^*,j^*} - v_{i^*,j^*-1}}{\Delta y} \right] / \Delta y \\
&\quad + \nu \left[H \frac{v_{i^*+1,j^*} - v_{i^*,j^*}}{\Delta x} - H \frac{v_{i^*,j^*} - v_{i^*-1,j^*}}{\Delta x} \right] / \Delta x \\
&\quad - g(H + H_{i^*,j^*-1})/2(\eta_{i^*,j^*} - \eta_{i^*,j^*-1}) / \Delta y + \rho_a / \rho_0 C_a v_{10} \sqrt{v_{10}^2 + u_{10}^2} \\
&\quad - C_D v_{i,j} \sqrt{(v_{i^*,j^*})^2 + (u_{i^*,j^*-1} + u_{i^*,j^*} + u_{i^*+1,j^*} + u_{i^*+1,j^*-1})^2}
\end{aligned}$$

When (switch Δx and $\Delta y:\Delta x \leftrightarrow \Delta y$, switch i and $j:i \leftrightarrow j$, switch u and $v :u \leftrightarrow v$, switch signal of the Coriolis term $+ \leftrightarrow -$) and we assume that $i^*=j, j^*=i$

Finally, after applying an explicit centered finite difference and leapfrog schemes, the 2DSWEs become as:(**Which are approximately first order in space and second order in time**)

$$\begin{aligned} \eta_{i,j}^{n+1} = & \eta_{i,j}^{n-1} - \frac{\Delta t}{\Delta x} [(H_{i,j}^n + H_{i+1,j}^n)u_{i+1,j}^n - (H_{i-1,j}^n + H_{i,j}^n)u_{i,j}^n] \\ & - \frac{\Delta t}{\Delta y} [(H_{i,j}^n + H_{i,j+1}^n)v_{i,j+1}^n - (H_{i,j}^n + H_{i,j-1}^n)v_{i,j}^n] \end{aligned} \quad (2.1)$$

$$\begin{aligned} u_{i,j}^{n+1} = & \left[u_{i,j}^{n-1}(H_{i,j}^{n-1} + H_{i-1,j}^{n-1}) - \frac{\Delta t}{\Delta x} [H_{i,j}^n(u_{i+1,j}^n + u_{i,j}^n)^2 - H_{i-1,j}^n(u_{i-1,j}^n + u_{i,j}^n)^2] \right. \\ & - \frac{\Delta t}{4\Delta x} [(H_{i-1,j}^n + H_{i,j}^n + H_{i-1,j+1}^n + H_{i,j+1}^n) \times (u_{i,j+1}^n + u_{i,j}^n)(v_{i-1,j+1}^n + v_{i,j+1}^n)] \\ & + \frac{\Delta t}{4\Delta y} [(H_{i-1,j}^n + H_{i,j}^n + H_{i-1,j-1}^n + H_{i,j-1}^n)(u_{i,j}^n + u_{i,j-1}^n)(v_{i-1,j}^n + v_{i,j}^n)] \\ & + \frac{\Delta t}{2} f(H_{i,j}^n + H_{i-1,j}^n) \times (v_{i-1,j}^n + v_{i,j}^n + v_{i,j+1}^n + v_{i-1,j+1}^n) \\ & + \frac{4\Delta t}{\Delta x^2} \nu [H_{i-1,j}^n + H_{i,j}^n] (u_{i+1,j}^n - 2u_{i,j}^n + u_{i-1,j}^n) \\ & + \frac{4\Delta t}{\Delta y^2} \nu [H_{i-1,j}^n + H_{i,j}^n] (u_{i,j-1}^n - 2u_{i,j}^n + u_{i,j+1}^n) \\ & - g \frac{2\Delta t}{\Delta x} (H_{i,j}^n + H_{i-1,j}^n)(\eta_{i,j}^n - \eta_{i-1,j}^n) + 4\Delta t \rho_a / \rho_0 C_a u_{10} \sqrt{u_{10}^2 + v_{10}^2} \\ & \left. - 4\Delta t C_D u_{i,j}^{n+1} \sqrt{(u_{i,j}^n)^2 + (v_{i-1,j}^n + v_{i,j}^n + v_{i,j+1}^n + v_{i-1,j+1}^n)^2} \right] / (H_{i,j}^{n+1} + H_{i-1,j}^{n+1}) \end{aligned} \quad (2.2)$$

$$\begin{aligned} v_{i^*,j^*}^{n+1} = & \left[v_{i^*,j^*}^{n-1}(H_{i^*,j^*}^{n-1} + H_{i^*,j^*-1}^{n-1}) - \frac{\Delta t}{\Delta y} [H_{i^*,j^*}^n(v_{i^*,j^*+1}^n + v_{i^*,j^*}^n)^2 - H_{i^*,j^*-1}^n(v_{i^*,j^*-1}^n + v_{i^*,j^*}^n)^2] \right. \\ & - \frac{\Delta t}{4\Delta y} [(H_{i^*,j^*-1}^n + H_{i^*,j^*}^n + H_{i^*+1,j^*-1}^n + H_{i^*+1,j^*}^n) \times (v_{i^*+1,j^*}^n + v_{i^*,j^*}^n)(u_{i^*+1,j^*-1}^n + u_{i^*+1,j^*}^n)] \\ & + \frac{\Delta t}{4\Delta x} [(H_{i^*,j^*-1}^n + H_{i^*,j^*}^n + H_{i^*-1,j^*-1}^n + H_{i^*-1,j^*}^n)(v_{i^*,j^*}^n + v_{i^*-1,j^*}^n)(u_{i^*,j^*-1}^n + u_{i^*,j^*}^n)] \\ & - \frac{\Delta t}{2} f(H_{i^*,j^*}^n + H_{i^*,j^*-1}^n) \times (u_{i^*,j^*-1}^n + u_{i^*,j^*}^n + u_{i^*+1,j^*}^n + u_{i^*+1,j^*-1}^n) \\ & + \frac{4\Delta t}{\Delta y^2} \nu [H_{i^*,j^*-1}^n + H_{i^*,j^*}^n] (v_{i^*,j^*+1}^n - 2v_{i^*,j^*}^n + v_{i^*,j^*-1}^n) \\ & + \frac{4\Delta t}{\Delta x^2} \nu [H_{i^*,j^*-1}^n + H_{i^*,j^*}^n] (v_{i^*-1,j^*}^n - 2v_{i^*,j^*}^n + v_{i^*+1,j^*}^n) \\ & - g \frac{2\Delta t}{\Delta y} (H_{i^*,j^*}^n + H_{i^*,j^*-1}^n)(\eta_{i^*,j^*}^n - \eta_{i^*,j^*-1}^n) + 4\Delta t \rho_a / \rho_0 C_a v_{10} \sqrt{v_{10}^2 + u_{10}^2} \\ & \left. - 4\Delta t C_D v_{i^*,j^*}^{n+1} \sqrt{(v_{i^*,j^*}^n)^2 + (u_{i^*,j^*-1}^n + u_{i^*,j^*}^n + u_{i^*+1,j^*}^n + u_{i^*+1,j^*-1}^n)^2} \right] / (H_{i^*,j^*}^{n+1} + H_{i^*,j^*-1}^{n+1}) \end{aligned} \quad (2.3)$$

Remark:

In order for the leapfrog scheme to be stable, the dissipation (horizontal viscosity and any frictional) terms in the momentum equations should be lagged (skipped) that is evaluated at time level $n - 1$. The rest of the terms in above equations are evaluated at time level n . Therefore η equation is solved first, so that H^{n+1} is known, before solving for the velocity components u and v . Robert-Asselin filter is applied for u , v and η after integration at each time step [69]:

$$\begin{aligned}(\eta_{i,j}^n)_f &= (\eta_{i,j}^n) + \gamma(\eta_{i,j}^{n+1} - 2\eta_{i,j}^n + \eta_{i,j}^{n-1}) \\(u_{i,j}^n)_f &= (u_{i,j}^n) + \gamma(u_{i,j}^{n+1} - 2u_{i,j}^n + u_{i,j}^{n-1}) \\(v_{i^*,j^*}^n)_f &= (v_{i^*,j^*}^n) + \gamma(v_{i^*,j^*}^{n+1} - 2v_{i^*,j^*}^n + v_{i^*,j^*}^{n-1})\end{aligned}$$

2.3.3 Computational algorithm

The 2D depth-averaged shallow water equations are solved on the basis of the following algorithm:

1. Input model data and set initial data. At time $t = n\Delta t = 0$ (that is $n = 0$, and $t = n\Delta t$ also $u_{i,j}^0 = v_{i,j}^0 = 0, H_{i,j}^0 = h_{i,j}$) on the open boundary $H_{i,j}^0 = h_{i,j} + \eta_{i,j}^0$ are known. Get the unknown values $u_{i,j}^{n-1}, u_{i-1,j}^n, u_{i+1,j}^n, v_{i^*,j^*}^{n-1}, v_{i^*,j^*-1}^n, v_{i^*,j^*+1}^n$ and $\eta_{i,j}^{n-1}$ by a forward step.
2. Update model time to level $(n + 1)$, so $t = (n + 1)\Delta t$. Solve the continuity equation to find η^{n+1} and H^{n+1} using $u_{i,j}^n, v_{i^*,j^*}^n$.
3. Update model time to level $(n + 1)$. Solve the momentum equations for u^{n+1} and v^{n+1} using H^{n+1} .
4. Apply Robert-Asselin filter for u, v and η for each time step.
5. Return to step 2 and continue until the period of the simulation is completed.

2.4 Another Way to Numerical Discretization of 2DSWEs when the Time Step (n+1/2)

In this section, the numerical discretization for 2DSWEs is proposed by using an explicit finite difference in space and leapfrog combined with simple Robert-Asselin filtering in time.

The grid consists of three grid points which are the free surface elevation η and horizontal velocity components u and v . The spacing between different grid points in the x -direction and y -direction are Δx and Δy respectively and between similar grid points it is $2\Delta x$ and $2\Delta y$. Therefore, the area of the grid is $4\Delta x\Delta y$.

1. Discretization of continuity equation

Consider the finite difference expression of the continuity equation (1.10) centering it about the time level n , in discrete time and the η grid point in discrete space gives:

$$\left(\frac{\partial\eta}{\partial t}\right)_{\eta_{i,j}}^n = -\left(\frac{\partial(Hu)}{\partial x}\right)_{\eta_{i,j}}^n - \left(\frac{\partial(Hv)}{\partial y}\right)_{\eta_{i,j}}^n \quad (2.4)$$

The difference approximations of the terms in this equation are:

1.

$$\left(\frac{\partial\eta}{\partial t}\right)_{\eta_{i,j}}^n \simeq \frac{1}{\Delta t} \left(\eta_{i,j}^{n+1/2} - \eta_{i,j}^{n-1/2}\right) \quad (2.5)$$

2.

$$\left(\frac{\partial(Hu)}{\partial x}\right)_{\eta_{i,j}}^n \simeq \frac{1}{4\Delta x} [(H_{i,j}^n + H_{i+1,j}^n)u_{i,j}^n - (H_{i,j}^n + H_{i-1,j}^n)u_{i-1,j}^n] \quad (2.6)$$

3.

$$\left(\frac{\partial(Hv)}{\partial y}\right)_{\eta_{i,j}}^n \simeq \frac{1}{4\Delta y} [(H_{i,j}^n + H_{i,j+1}^n)v_{i,j+1}^n - (H_{i,j}^n + H_{i,j-1}^n)v_{i,j}^n] \quad (2.7)$$

Expression equation (2.5) is second order convergent because it is a centered difference in time and equations (2.6) and (2.7) are second order convergent in space because the difference $u_{i,j}^n - u_{i-1,j}^n$ is centered about η .

By substituting equations (2.5)-(2.6) and (2.7) in equation (2.4), we obtain the explicit expression for the water surface elevation at time level $(n + 1/2)$, namely

$$\eta_{i,j}^{n+1/2} = \eta_{i,j}^{n-1/2} - \frac{\Delta t}{4\Delta x} [(H_{i,j}^n + H_{i+1,j}^n)u_{i,j}^n - (H_{i,j}^n + H_{i-1,j}^n)u_{i-1,j}^n] - \frac{\Delta t}{4\Delta y} [(H_{i,j}^n + H_{i,j+1}^n)v_{i,j+1}^n - (H_{i,j}^n + H_{i,j-1}^n)v_{i,j}^n] \quad (2.8)$$

The total depth $H = \eta + h$ at time $(n + 1/2) \Delta t$ is approximated as

$$H_{i,j}^{n+1/2} = \eta_{i,j}^{n+1/2} + h_{i,j}$$

where $H_{i,j}$ is computed at the same position in the horizontal plane as $\eta_{i,j}$.

2. Discretization of the momentum equations

Case 1: The finite difference expression for the x-momentum equation (1.11) using centered differences about $u_{i,j}$ in space and about the $(n + 1/2)$ level in time is based on the following equation

$$\begin{aligned}
& \left(\frac{\partial u}{\partial t} \right)_{u_{i,j}}^{n+1/2} + \left(u \frac{\partial u}{\partial x} \right)_{u_{i,j}}^{n+1/2} + \left(u \frac{\partial v}{\partial y} \right)_{u_{i,j}}^{n+1/2} - (fv)_{u_{i,j}}^n \\
& = -g \left(\frac{\partial \eta}{\partial x} \right)_{u_{i,j}}^{n+1/2} + \left(\nu \frac{\partial^2 u}{\partial x^2} \right)_{u_{i,j}}^{n+1/2} + \left(\nu \frac{\partial^2 u}{\partial y^2} \right)_{u_{i,j}}^{n+1/2} \\
& + \left(\frac{\rho_a}{\rho_0} \frac{1}{H} C_a u_{10} \sqrt{u_{10}^2 + v_{10}^2} \right)_{u_{i,j}}^{n+1/2} - \frac{1}{H} C_D \left(u \sqrt{u^2 + v^2} \right)_{u_{i,j}}^{n+1/2}
\end{aligned} \tag{2.9}$$

The terms of this are differences as follows:

1.

$$\left(\frac{\partial u}{\partial t} \right)_{u_{i,j}}^{n+1/2} \simeq \frac{1}{\Delta t} \left(u_{i,j}^{n+1} - u_{i,j}^n \right) \tag{2.10}$$

Which is second order in time because this is a centered-difference in time.

2. Discretization of advection term:

$$\left(u \frac{\partial u}{\partial x} \right)_{u_{i,j}}^{n+1/2} \simeq (u_{i,j})^{n+1} \left(\frac{\partial u}{\partial x} \right)_{u_{i,j}}^n \simeq \frac{1}{4\Delta x} u_{i,j}^{n+1} (u_{i+1,j}^n - u_{i-1,j}^n) \tag{2.11}$$

Which is approximately second order in space.

3.

$$\left(v \frac{\partial u}{\partial y} \right)_{u_{i,j}}^{n+1/2} \simeq (v)_{u_{i,j}}^n \left(\frac{\partial u}{\partial y} \right)_{u_{i,j}}^n \simeq \frac{1}{16\Delta y} (v_{i,j+1}^n + v_{i+1,j+1}^n + v_{i,j}^n + v_{i+1,j}^n) (u_{i,j+1}^n - u_{i,j-1}^n) \tag{2.12}$$

Which is quasi second order in space.

4. Discretization of Coriolis force term

$$(fv)_{u_{i,j}}^n \simeq \frac{1}{8} (f_{i,j} + f_{i+1,j}) (v_{i,j+1}^n + v_{i+1,j+1}^n + v_{i,j}^n + v_{i+1,j}^n) \tag{2.13}$$

5. Discretization of Barotropic term

$$\left(\frac{\partial \eta}{\partial x} \right)_{u_{i,j}}^{n+1/2} \simeq \frac{1}{2\Delta x} \left(\eta_{i+1,j}^{n+1/2} - \eta_{i,j}^{n+1/2} \right) \tag{2.14}$$

which has a second order spatial discretisation error.

6. Discretization of horizontal viscosity terms

They are approximated as follows; At the $u_{i,j}$ grid point and the $(n + 1/2)$ time level.

$$\begin{aligned} (\nu_x)_{u_{i,j}}^{n+1/2} &= \frac{a}{2}(2\Delta x) (H)_{u_{i,j}}^{n+1/2} \\ &\simeq a\Delta x \frac{(H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2})}{2} \end{aligned} \quad (2.15)$$

and

$$\begin{aligned} (\nu_y)_{u_{i,j}}^{n+1/2} &= \frac{a}{2}(2\Delta y) (H)_{u_{i,j}}^{n+1/2} \\ &\simeq a\Delta y \frac{(H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2})}{2} \end{aligned} \quad (2.16)$$

This implies

$$\begin{aligned} \left(\nu_x \frac{\partial^2 u}{\partial x^2} \right)_{u_{i,j}}^{n+1/2} &\simeq \nu_x \left(\frac{\partial^2 u}{\partial x^2} \right)_{u_{i,j}}^n \\ &\simeq \frac{a}{8\Delta x} (u_{i+1,j}^n - 2u_{i,j}^n + u_{i-1,j}^n) (H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2}) \end{aligned} \quad (2.17)$$

which is approximately second order in space.

where a is the reduced eddy coefficient, which equal to 1.64×10^{-3} (see [85]).

7.

$$\begin{aligned} \left(\nu_y \frac{\partial^2 u}{\partial y^2} \right)_{u_{i,j}}^{n+1/2} &\simeq \nu_y \left(\frac{\partial^2 u}{\partial y^2} \right)_{u_{i,j}}^n \\ &\simeq \frac{a}{8\Delta y} (u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n) (H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2}) \end{aligned} \quad (2.18)$$

which is approximately second order in space.

8.

$$C_D \left(\frac{1}{H} u \sqrt{u^2 + v^2} \right)_{u_{i,j}}^{n+1/2} \simeq \frac{2C_D}{(H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2})} u_{i,j}^{n+1} \sqrt{(u_{i,j}^n)^2 + \frac{1}{16}(v_{i,j+1}^n + v_{i+1,j+1}^n + v_{i,j}^n + v_{i+1,j}^n)^2} \quad (2.19)$$

9.

$$\left(\frac{\rho_a}{\rho_0} \frac{1}{H} C_a u_{10} \sqrt{u_{10}^2 + v_{10}^2} \right)_{u_{i,j}}^{n+1/2} \simeq \frac{\rho_a}{\rho_0} \frac{2C_a}{(H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2})} u_{10} \sqrt{u_{10}^2 + v_{10}^2} \quad (2.20)$$

Substituting all the approximations (2.10)-(2.20) into equation (2.9) gives the corresponding finite difference approximation

$$\begin{aligned}
& \frac{1}{\Delta t} \left(u_{i,j}^{n+1} - u_{i,j}^n \right) + \frac{1}{4\Delta x} u_{i,j}^{n+1} (u_{i+1,j}^n - u_{i-1,j}^n) \\
& + \frac{1}{16\Delta y} (v_{i,j+1}^n + v_{i+1,j+1}^n + v_{i,j}^n + v_{i+1,j}^n) (u_{i,j+1}^n - u_{i,j-1}^n) \\
& - \frac{1}{8} (f_{i,j} + f_{i+1,j}) (v_{i,j+1}^n + v_{i+1,j+1}^n + v_{i,j}^n + v_{i+1,j}^n) \\
& = -g \frac{1}{\Delta x} (\eta_{i+1,j}^{n+1/2} - \eta_{i,j}^{n+1/2}) + \frac{a}{8\Delta x} (H_i^{n+1/2} + H_{i+1}^{n+1/2}) (u_{i+1,j}^n - 2u_{i,j}^n + u_{i-1,j}^n) \\
& + \frac{a}{8\Delta y} (H_i^{n+1/2} + H_{i+1}^{n+1/2}) (u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n) \\
& - \frac{2C_D}{(H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2})} u_{i,j}^{n+1} \sqrt{(u_{i,j}^n)^2 + \frac{1}{16}(v_{i,j+1}^n + v_{i+1,j+1}^n + v_{i,j}^n + v_{i+1,j}^n)^2} \\
& + \frac{\rho_a}{\rho_0} \frac{2C_a}{(H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2})} u_{10} \sqrt{u_{10}^2 + v_{10}^2}
\end{aligned}$$

That implies:

$$\begin{aligned}
u_{i,j}^{n+1} &= \left[u_{i,j}^n - \frac{\Delta t}{4\Delta x} u_{i,j}^{n+1} (u_{i+1,j}^n - u_{i-1,j}^n) - \frac{\Delta t}{16\Delta y} (v_{i,j+1}^n + v_{i+1,j+1}^n + v_{i,j}^n + v_{i+1,j}^n) (u_{i,j+1}^n - u_{i,j-1}^n) \right. \\
& + \frac{\Delta t}{8} (f_{i,j} + f_{i+1,j}) (v_{i,j+1}^n + v_{i+1,j+1}^n + v_{i,j}^n + v_{i+1,j}^n) - \frac{g\Delta t}{2\Delta x} (\eta_{i+1,j}^{n+1/2} - \eta_{i,j}^{n+1/2}) \\
& + \frac{a\Delta t}{8\Delta x} (u_{i+1,j}^n - 2u_{i,j}^n + u_{i-1,j}^n) (H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2}) \\
& + \frac{a\Delta t}{8\Delta y} (H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2}) (u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n) \\
& - \frac{2C_D\Delta t}{(H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2})} u_{i,j}^{n+1} \sqrt{(u_{i,j}^n)^2 + \frac{1}{16}(v_{i,j+1}^n + v_{i+1,j+1}^n + v_{i,j}^n + v_{i+1,j}^n)^2} \\
& \left. + \frac{\rho_a}{\rho_0} \frac{2C_a\Delta t}{(H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2})} u_{10} \sqrt{u_{10}^2 + v_{10}^2} \right]
\end{aligned}$$

Finally, we obtain

$$\begin{aligned}
u_{i,j}^{n+1} &= K_x^{-1} \left[u_{i,j}^n + \frac{\Delta t}{8} (f_{i,j} + f_{i+1,j}) (v_{i,j+1}^n + v_{i+1,j+1}^n + v_{i,j}^n + v_{i+1,j}^n) - \frac{g\Delta t}{2\Delta x} (\eta_{i+1,j}^{n+1/2} - \eta_{i,j}^{n+1/2}) \right. \\
& + \frac{a\Delta t}{8\Delta x} (u_{i+1,j}^n - 2u_{i,j}^n + u_{i-1,j}^n) (H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2}) \\
& - \frac{\Delta t}{16\Delta y} (v_{i,j+1}^n + v_{i+1,j+1}^n + v_{i,j}^n + v_{i+1,j}^n) (u_{i,j+1}^n - u_{i,j-1}^n) \\
& + \frac{a\Delta t}{8\Delta y} (H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2}) (u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n) \\
& \left. + \frac{\rho_a}{\rho_0} \frac{2C_a\Delta t}{(H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2})} u_{10} \sqrt{u_{10}^2 + v_{10}^2} \right] \tag{2.21}
\end{aligned}$$

where

$K_x =$

$$\left(1 + \frac{\Delta t}{4\Delta x} (u_{i+1,j}^n - u_{i-1,j}^n) + \frac{2C_D\Delta t}{(H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2})} \sqrt{(u_{i,j}^n)^2 + \frac{1}{16}(v_{i,j+1}^n + v_{i+1,j+1}^n + v_{i,j}^n + v_{i+1,j}^n)^2} \right)$$

Case 2: At the v_i point and the $(n + 1/2)$ time level 2D depth-averaged y -momentum:

$$\begin{aligned} & \left(\frac{\partial v}{\partial t} \right)_{v_{i,j}}^{n+1/2} + v \left(\frac{\partial u}{\partial x} \right)_{v_{i,j}}^{n+1/2} + v \left(\frac{\partial v}{\partial y} \right)_{v_{i,j}}^{n+1/2} - (fu)_{v_{i,j}}^n \\ &= -g \left(\frac{\partial \eta}{\partial y} \right)_{v_{i,j}}^{n+1/2} + \left(\nu \frac{\partial^2 v}{\partial x^2} \right)_{v_{i,j}}^{n+1/2} + \left(\nu \frac{\partial^2 v}{\partial y^2} \right)_{v_{i,j}}^{n+1/2} \\ &+ \left(\frac{\rho_a}{\rho_0} \frac{1}{H} C_a v_{10} \sqrt{u_{10}^2 + v_{10}^2} \right)_{v_{i,j}}^{n+1/2} - C_D \frac{1}{H} \left(v \sqrt{u^2 + v^2} \right)_{v_{i,j}}^{n+1/2} \end{aligned} \quad (2.22)$$

The approximations for the terms of equation (2.22) as follows:

1.

$$\left(\frac{\partial v}{\partial t} \right)_{v_{i,j}}^{n+1/2} \simeq \frac{1}{\Delta t} (v_{i,j}^{n+1} - v_{i,j}^n) \quad (2.23)$$

which is second order in time because this is a centred-difference in time.

2. Discretization of advection term

$$\left(u \frac{\partial v}{\partial x} \right)_{v_{i,j}}^{n+1/2} \simeq (u_{i,j})^{n+1} \left(\frac{\partial v}{\partial x} \right)_{v_{i,j}}^n \simeq \frac{1}{16\Delta x} (u_{i,j}^{n+1} + u_{i-1,j}^{n+1} + u_{i,j-1}^{n+1} + u_{i-1,j-1}^{n+1}) (v_{i+1,j}^n - v_{i-1,j}^n) \quad (2.24)$$

which is quasi second order in space.

3.

$$\left(v \frac{\partial v}{\partial y} \right)_{v_{i,j}}^{n+1/2} \simeq (v_{i,j})^{n+1} \left(\frac{\partial v}{\partial y} \right)_{v_{i,j}}^n \simeq \frac{1}{4\Delta y} v_{i,j}^{n+1} (v_{i,j+1}^n - v_{i,j-1}^n) \quad (2.25)$$

which is approximately second order in space.

4. Discretization of Coriolis force term

$$(fu)_{v_{i,j}}^{n+1/2} \simeq \frac{1}{8} (f_{i,j} + f_{i,j-1}) (u_{i,j}^{n+1} + u_{i-1,j}^{n+1} + u_{i,j-1}^{n+1} + u_{i-1,j-1}^{n+1}) \quad (2.26)$$

5. Discretization of barotropic term

$$\left(\frac{\partial \eta}{\partial y} \right)_{v_{i,j}}^{n+1/2} \simeq \frac{1}{\Delta y} (\eta_{i,j}^{n+1/2} - \eta_{i,j-1}^{n+1/2}) \quad (2.27)$$

which has a second order spatial discretisation error.

6. Discretization of horizontal eddy viscosity coefficients

At the $v_{i,j}$ grid point and the $(n + 1/2)$ time level. They are approximated as follows:

$$\begin{aligned} \left(\nu_x \frac{\partial^2 v}{\partial x^2} \right)_{v_{i,j}}^{n+1/2} &\simeq (\nu_x)_{v_{i,j}}^{n+1/2} \left(\frac{\partial^2 v}{\partial x^2} \right)_{v_{i,j}}^n \\ &\simeq \frac{a}{8\Delta x} \left(H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2} \right) (v_{i+1,j}^n - 2v_{i,j}^n + v_{i-1,j}^n) \end{aligned} \quad (2.28)$$

which is approximately second order in space.

7.

$$\begin{aligned} \left(\nu_y \frac{\partial^2 v}{\partial y^2} \right)_{v_{i,j}}^{n+1/2} &\simeq (\nu_y)_{v_{i,j}}^n \left(\frac{\partial^2 v}{\partial y^2} \right)_{v_{i,j}}^n \\ &\simeq \frac{a}{8\Delta y} \left(H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2} \right) (v_{i,j+1}^n - 2v_{i,j}^n + v_{i,j-1}^n) \end{aligned} \quad (2.29)$$

which is approximately second order in space .

Substituting all the approximations (2.23)-(2.29) in to equation (2.22) yield the approximation

$$\begin{aligned} &\frac{1}{\Delta t} (v_{i,j}^{n+1} - v_{i,j}^n) + \frac{1}{4\Delta y} v_{i,j}^{n+1} (v_{i,j+1}^n - v_{i,j-1}^n) \\ &\quad + \frac{1}{16\Delta x} (u_{i,j+1}^{n+1} + u_{i-1,j}^{n+1} + u_{i,j-1}^{n+1} + u_{i-1,j-1}^{n+1}) (v_{i+1,j}^n - v_{i-1,j}^n) \\ &\quad - \frac{1}{8} (f_{i,j} + f_{i,j-1}) (u_{i,j}^{n+1} + u_{i-1,j}^{n+1} + u_{i,j-1}^{n+1} + u_{i-1,j-1}^{n+1}) \\ &= -g \frac{1}{2\Delta y} (\eta_{i,j}^{n+1/2} - \eta_{i,j-1}^{n+1/2}) \\ &\quad + \frac{a}{8\Delta x} (v_{i+1,j}^n - 2v_{i,j}^n + v_{i-1,j}^n) (H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2}) \\ &\quad + \frac{a}{8\Delta y} (H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2}) (v_{i,j+1}^n - 2v_{i,j}^n + v_{i,j-1}^n) \\ &\quad - \frac{2C_D}{(H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2})} v_{i,j}^{n+1} \sqrt{(v_{i,j}^n)^2 + \frac{1}{16} (u_{i,j}^{n+1} + u_{i-1,j}^{n+1} + u_{i,j-1}^{n+1} + u_{i-1,j-1}^{n+1})^2} \\ &\quad + \frac{\rho_a}{\rho_0} \frac{2C_a}{(H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2})} v_{10} \sqrt{u_{10}^2 + v_{10}^2} \end{aligned}$$

That implies:

$$\begin{aligned}
v_{i,j}^{n+1} = & \left[v_{i,j}^n - \frac{\Delta t}{4\Delta y} v_{i,j}^{n+1} (v_{i,j+1}^n - v_{i,j-1}^n) - \frac{\Delta t}{16\Delta x} (u_{i,j+1}^{n+1} + u_{i-1,j}^{n+1} + u_{i,j-1}^{n+1} + u_{i-1,j-1}^{n+1}) (v_{i+1,j}^n - v_{i-1,j}^n) \right. \\
& + \frac{\Delta t}{8} (f_{i,j} + f_{i,j-1}) (u_{i,j}^{n+1} + u_{i-1,j}^{n+1} + u_{i,j-1}^{n+1} + u_{i-1,j-1}^{n+1}) \\
& - \frac{g\Delta t}{2\Delta y} (\eta_{i,j}^{n+1/2} - \eta_{i,j-1}^{n+1/2}) + \frac{a\Delta t}{8\Delta x} (v_{i+1,j}^n - 2v_{i,j}^n + v_{i-1,j}^n) (H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2}) \\
& + \frac{a\Delta t}{8\Delta y} (H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2}) (v_{i,j+1}^n - 2v_{i,j}^n + v_{i,j-1}^n) \\
& - \frac{2C_D\Delta t}{(H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2})} v_{i,j}^{n+1} \sqrt{(v_{i,j}^n)^2 + \frac{1}{16}(u_{i,j}^{n+1} + u_{i-1,j}^{n+1} + u_{i,j-1}^{n+1} + u_{i-1,j-1}^{n+1})^2} \\
& \left. + \frac{\rho_a}{\rho_0} \frac{2C_a\Delta t}{(H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2})} v_{10} \sqrt{u_{10}^2 + v_{10}^2} \right]
\end{aligned}$$

Finally, we obtain:

$$\begin{aligned}
v_{i,j}^{n+1} = & K_y^{-1} \left[v_{i,j}^n - \frac{\Delta t}{16\Delta x} (u_{i,j}^{n+1} + u_{i-1,j}^{n+1} + u_{i,j-1}^n + u_{i-1,j-1}^{n+1}) (v_{i+1,j}^n - v_{i-1,j}^n) \right. \\
& - \frac{\Delta t}{8} (f_{i,j} + f_{i,j-1}) (u_{i,j}^{n+1} + u_{i-1,j}^{n+1} + u_{i,j-1}^{n+1} + u_{i-1,j-1}^{n+1}) \\
& - \frac{g\Delta t}{2\Delta y} (\eta_{i,j}^{n+1/2} - \eta_{i,j-1}^{n+1/2}) + \frac{a\Delta t}{8\Delta x} (v_{i+1,j}^n - 2v_{i,j}^n + v_{i-1,j}^n) (H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2}) \\
& + \frac{a\Delta t}{8\Delta y} (H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2}) (v_{i,j+1}^n - 2v_{i,j}^n + v_{i,j-1}^n) \\
& \left. + \frac{\rho_a}{\rho_0} \frac{2C_a\Delta t}{(H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2})} v_{10} \sqrt{u_{10}^2 + v_{10}^2} \right] \quad (2.30)
\end{aligned}$$

where

$$K_y = \left(1 + \frac{\Delta t}{4\Delta y} (v_{i,j+1}^n - v_{i,j-1}^n) + \frac{2C_D\Delta t}{(H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2})} \sqrt{(v_{i,j}^n)^2 + \frac{1}{16}(u_{i,j}^{n+1} + u_{i-1,j}^{n+1} + u_{i,j-1}^{n+1} + u_{i-1,j-1}^{n+1})^2} \right)$$

The values at the full time level are found using

$$\begin{aligned}
\eta_{i,j}^{n+1} = & \eta_{i,j}^n - \frac{\Delta t}{8\Delta x} \left[(H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2})(u_{i,j}^{n+1} + u_{i,j}^n) - (H_{i-1,j}^{n+1/2} + H_{i,j}^{n+1/2})(u_{i-1,j}^{n+1} + u_{i-1,j}^n) \right] \\
& - \frac{\Delta t}{8\Delta y} \left[(H_{i,j}^{n+1/2} + H_{i,j+1}^{n+1/2})(v_{i,j+1}^{n+1} + v_{i,j+1}^n) - (H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2})(v_{i,j}^{n+1} + v_{i,j}^n) \right] \quad (2.31)
\end{aligned}$$

which is second order convergent in space and time.

Note that, in the above $\eta_{i,j}^{n+1/2}$ are calculated for all i, j then the pair $u_{i,j}^{n+1}, v_{i,j}^{n+1}$ are calculated for all i, j and finally $\eta_{i,j}^{n+1}$ calculated for all i, j .

2.4.1 Computational algorithm in case the time step $(n+1/2)$

The 2D depth-averaged shallow water equations are solved on the basis of the following algorithm:

1. Input model data and set initial data. At time $t = n\Delta t = 0$ (that is $n = 0$, and $t = n \Delta t$ also $u_{i,j}^0 = v_{i,j}^0 = 0, H_{i,j}^0 = h_{i,j}$) on the open boundary $H_{i,j}^0 = h_{i,j} + \eta_{i,j}^0$ are known. Get the values $u_{i,j}^{n-1/2}, v_{i,j}^{n-1/2}$ and $\eta_{i,j}^{n-1/2}$ by a forward step.
2. Update model time to level $(n + 1/2)$, so $t = (n + 1/2) \Delta t$. Solve the continuity equation to find $\eta^{n+1/2}$ and $H^{n+1/2}$ using u^n, v^n .
3. Update model time to level $(n + 1)$. Solve the momentum equations for u^{n+1} and v^{n+1} using $\eta^{n+1/2}$.
4. Update model to time level $(n + 1)$. Solve continuity equation for η^{n+1} and H^{n+1} .
5. Apply Robert-Asselin filter for u, v and η for each time step.
6. Return to step 2 and continue until the period of the simulation is completed.

2.5 Numerical Discretization for 2DSWEs Using EFDMs in Several Cases

Consider that the system of 2DSWEs given in equations (1.32)-(1.34) when the wind stress, viscosity and the force f are ignored. Therefore, this model becomes:

$$\frac{\partial \eta}{\partial t} + \frac{\partial(Hu)}{\partial x} + \frac{\partial(Hv)}{\partial y} = 0 \quad (2.32)$$

$$\frac{\partial(Hu)}{\partial t} + \frac{\partial(Hu^2)}{\partial x} + \frac{\partial(Huv)}{\partial y} + gH \frac{\partial \eta}{\partial x} + \frac{\tau_x}{\rho} = 0 \quad (2.33)$$

$$\frac{\partial(Hv)}{\partial t} + \frac{\partial(Huv)}{\partial x} + \frac{\partial(Hv^2)}{\partial y} + gH \frac{\partial \eta}{\partial y} + \frac{\tau_y}{\rho} = 0 \quad (2.34)$$

By applying an explicit centered finite difference and leapfrog schemes with Robert-Asselin in above system. **There are two cases:**

Case 1. When the time step $(n + 1/2)$

1. **Continuity equation:** By using the equation (2.8) in Section 2.4, we get:

$$\begin{aligned}\eta_i^{n+1/2} = & \eta_{i,j}^{n-1/2} - \frac{1}{4\Delta x} [(H_{i,j}^n + H_{i+1,j}^n)u_i^n - (H_{i,j}^n + H_{i-1,j}^n)u_{i-1,j}^n] \\ & - \frac{1}{4\Delta y} [(H_{i,j}^n + H_{i,j+1}^n)v_{i,j+1}^n - (H_{i,j}^n + H_{i,j-1}^n)v_{i,j}^n]\end{aligned}$$

a. x -momentum equation

By using the equation (2.21) in Section 2.4 and assume the wind stress, viscosity, and the force f are ignored, we get:

$$\begin{aligned}u_i^{n+1} = & K_x^{-1} \left[u_{i,j}^n - \frac{\Delta t}{16\Delta y} (v_{i,j+1}^n + v_{i+1,j+1}^n + v_{i,j}^n + v_{i+1,j}^n) (u_{i,j+1}^n - u_{i,j-1}^n) \right. \\ & \left. + \frac{\rho_a}{\rho_0} \frac{2C_a\Delta t}{(H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2})} u_{10} \sqrt{u_{10}^2 + v_{10}^2} - \frac{g\Delta t}{2\Delta x} (\eta_{i+1,j}^{n+1/2} - \eta_{i,j}^{n+1/2}) \right]\end{aligned}$$

$K_x =$

$$\left(1 + \frac{\Delta t}{4\Delta x} (u_{i+1,j}^n - u_{i-1,j}^n) + \frac{2C_D\Delta t}{(H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2})} \sqrt{(u_{i,j}^n)^2 + \frac{1}{16}(v_{i,j+1}^n + v_{i+1,j+1}^n + v_{i,j}^n + v_{i+1,j}^n)^2} \right)$$

b. y -momentum equation:

By using the equation (2.30) in Section 2.4, we get:

$$\begin{aligned}v_{i,j}^{n+1} = & K_y^{-1} \left[v_{i,j}^n - \frac{\Delta t}{16\Delta x} (u_{i,j}^{n+1} + u_{i-1,j}^{n+1} + u_{i,j-1}^{n+1} + u_{i-1,j-1}^{n+1}) (v_{i+1,j}^n - v_{i-1,j}^n) \right. \\ & \left. - \frac{g\Delta t}{2\Delta y} (\eta_{i,j}^{n+1/2} - \eta_{i,j-1}^{n+1/2}) + \frac{\rho_a}{\rho_0} \frac{2C_a\Delta t}{(H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2})} v_{10} \sqrt{u_{10}^2 + v_{10}^2} \right]\end{aligned}$$

where

$$K_y = 1 + \frac{\Delta t}{4\Delta y} (v_{i,j+1}^n - v_{i,j-1}^n) - \frac{2C_D\Delta t}{(H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2})} \sqrt{(v_{i,j}^n)^2 + \frac{1}{16}(u_{i,j}^{n+1} + u_{i-1,j}^{n+1} + u_{i,j-1}^{n+1} + u_{i-1,j-1}^{n+1})^2}$$

Values of η at the full time level are found using

$$\begin{aligned}\eta_{i,j}^{n+1} = & \eta_{i,j}^n - \frac{\Delta t}{8\Delta x} [(H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2})(u_{i,j}^{n+1} + u_{i,j}^n) - (H_{i-1,j}^{n+1/2} + H_{i,j}^{n+1/2})(u_{i-1,j}^{n+1} + u_{i-1,j}^n)] \\ & - \frac{\Delta t}{8\Delta y} [(H_{i,j}^{n+1/2} + H_{i,j+1}^{n+1/2})(v_{i,j+1}^{n+1} + v_{i,j+1}^n) - (H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2})(v_{i,j}^{n+1} + v_{i,j}^n)]\end{aligned}$$

The total depth $H = \eta + h$ at time $(n + 1/2)\Delta t$ is approximated as $H_{i,j}^{n+1} = \eta_{i,j}^{n+1} + h_{i,j}$. Then, Robert-Asselin filter is applied for u , v and η after integration at each time step.

Case 2. When the time step ($n + 1$)

1. Continuity equation

As the same equation (2.1) in Section 2.3.

$$\begin{aligned}\eta_{i,j}^{n+1} = & \eta_{i,j}^{n-1} - \frac{\Delta t}{\Delta x} [(H_{i,j}^n + H_{i+1,j}^n)u_{i+1,j}^n - (H_{i-1,j}^n + H_{i,j}^n)u_{i,j}^n] \\ & - \frac{\Delta t}{\Delta y} [(H_{i,j}^n + H_{i,j+1}^n)v_{i,j+1}^n - (H_{i,j}^n + H_{i,j-1}^n)v_{i,j}^n]\end{aligned}$$

a. x -momentum equation

By using the equation (2.2) in Section 2.3 and assume the wind stress, viscosity, and the force f are ignored, we obtain:

$$\begin{aligned}\frac{\partial(Hu)}{\partial t} = & - [H(u_{i+1} + u_{i,j})^2/4 - H_{i-1}(u_{i-1} + u_{i,j})^2/4] / \Delta x \\ & - \{(H_{i-1,j} + H_{i,j} + H_{i-1,j+1} + H_{i,j+1}) \times (u_{i,j+1} + u_{i,j})(v_{i-1,j+1} + v_{i,j+1})/16\} / \Delta x \\ & + (H_{i-1,j} + H_{i,j} + H_{i-1,j-1} + H_{i,j-1})(u_{i,j} + u_{i,j-1})(v_{i-1,j} + v_{i,j})/16\} / \Delta y \\ & - g(H_{i,j} + H_{i-1,j})/2(\eta_{i,j} - \eta_{i-1,j})/\Delta x - C_D u_{i,j} \sqrt{(u_{i,j})^2 + (v_{i-1,j} + v_{i,j} + v_{i-1,j+1} + v_{i,j+1})^2} \\ & + \frac{\rho_a}{\rho_0} \Delta t C_a u_{10} \sqrt{u_{10}^2 + v_{10}^2}\end{aligned}$$

Finally, we obtain

$$\begin{aligned}u_{i,j}^{n+1} = & \left[u_{i,j}^{n-1} (H_{i,j}^{n-1} + H_{i-1,j}^{n-1}) - \frac{\Delta t}{\Delta x} [H_{i,j}^n (u_{i+1}^n + u_{i,j}^n)^2 - H_{i-1,j}^n (u_{i-1}^n + u_{i,j}^n)^2] \right. \\ & - \frac{\Delta t}{4\Delta x} \{ (H_{i-1,j}^n + H_{i,j}^n + H_{i-1,j+1}^n + H_{i,j+1}^n) \times (u_{j+1}^n + u_{i,j}^n)(v_{i-1,j+1}^n + v_{i,j+1}^n) \\ & + \frac{\Delta t}{4\Delta y} \{ (H_{i-1,j}^n + H_{i,j}^n + H_{i-1,j-1}^n + H_{i,j-1}^n)(u_{i,j}^n + u_{i,j-1}^n)(v_{i-1,j}^n + v_{i,j}^n) \\ & - g \frac{2\Delta t}{\Delta x} (H_{i,j}^n + H_{i-1,j}^n)(\eta_{i,j}^n - \eta_{i-1,j}^n) + 4\Delta t \rho_a / \rho_0 C_a u_{10} \sqrt{u_{10}^2 + v_{10}^2} \\ & \left. - 4\Delta t C_D u_{i,j}^{n+1} \sqrt{(u_{i,j}^n)^2 + (v_{i-1,j}^n + v_{i,j}^n + v_{i,j+1}^n + v_{i-1,j+1}^n)^2} \right] / (H_{i,j}^{n+1} + H_{i-1,j}^{n+1})\end{aligned}$$

b. y -momentum equation

By using the equation (2.3) in Section 2.3, we get:

$$\begin{aligned}\frac{\partial(Hv)}{\partial t} = & - [H(v_{i^*,j^*+1} + v_{i^*,j^*})^2/4 - H_{i^*,j^*-1}(v_{i^*,j^*-1} + v_{i^*,j^*})^2/4] / \Delta y \\ & - \{(H_{i^*,j^*-1} + H_{i^*,j^*} + H_{i^*+1,j^*-1} + H_{i^*+1,j^*}) \times (v_{i^*+1,j^*} + v_{i^*,j^*})(u_{i^*+1,j^*-1} + u_{i^*+1,j^*})/16\} / \Delta y \\ & - (H_{i^*,j^*-1} + H_{i^*,j^*} + H_{i^*-1,j^*-1} + H_{i^*-1,j^*})(v_{i^*,j^*} + v_{i^*-1,j^*})(u_{i^*,j^*-1} + u_{i^*,j^*})/16\} / \Delta x \\ & - g(H_{i^*,j^*} + H_{i^*,j^*-1})/2(\eta - \eta_{j^*-1})/\Delta y - C_D v_{i^*,j^*} \sqrt{(v_{i^*,j^*})^2 + (u_{i^*,j^*-1} + u_{i^*,j^*} + u_{i^*+1,j^*} + u_{i^*+1,j^*-1})^2} \\ & + \frac{\rho_a}{\rho_0} \Delta t C_a u_{10} \sqrt{u_{10}^2 + v_{10}^2}\end{aligned}$$

Finally, we obtain

$$\begin{aligned}
v_{i^*,j^*}^{n+1} = & \left[v_{i^*,j^*}^{n-1} (H_{i^*,j^*}^{n-1} + H_{i^*,j^*-1}^{n-1}) - \frac{\Delta t}{\Delta y} [H_{i^*,j^*}^n (v_{j^*+1}^n + v_{i^*,j^*}^n)^2 - H_{i^*,j^*-1}^n (v_{j^*-1}^n + v_{i^*,j^*}^n)^2] \right. \\
& - \frac{\Delta t}{4\Delta y} \{ (H_{i^*,j^*-1}^n + H_{i^*,j^*}^n + H_{i^*+1,j^*-1}^n + H_{i^*+1,j^*}^n) \times (v_{i^*+1,j^*}^n + v_{i^*,j^*}^n) (u_{i^*+1,j^*-1}^n + u_{i^*+1,j^*}^n) \\
& + \frac{\Delta t}{4\Delta x} \{ (H_{i^*,j^*-1}^n + H_{i^*,j^*}^n + H_{i^*-1,j^*-1}^n + H_{i^*-1,j^*}^n) (v_{i^*,j^*}^n + v_{i^*-1,j^*}^n) (u_{i^*,j^*-1}^n + u_{i^*,j^*}^n) \\
& - g \frac{2\Delta t}{\Delta y} (H_{i^*,j^*}^n + H_{i^*,j^*-1}^n) (\eta_{i^*,j^*}^n - \eta_{i^*,j^*-1}^n) + 4\Delta t \rho_a / \rho_0 C_a v_{10} \sqrt{v_{10}^2 + u_{10}^2} \\
& \left. - 4\Delta t C_D v_{i^*,j^*}^{n+1} \sqrt{(v_{i^*,j^*}^n)^2 + (u_{i^*,j^*-1}^n + u_{i^*,j^*}^n + u_{i^*+1,j^*}^n + u_{i^*+1,j^*-1}^n)^2} \right] / (H_{i^*,j^*}^{n+1} + H_{i^*,j^*-1}^{n+1})
\end{aligned}$$

2.6 Numerical Discretization for 2D Linear SWEs Using EFDMs

Consider the system of 2DSWEs given in equations (1.32)-(1.34) when the terms $\frac{\partial u^2}{\partial x} = 0$, $\frac{\partial v^2}{\partial y} = 0$, $\frac{\partial uv}{\partial y} = 0$, $\frac{\partial uv}{\partial x} = 0$, $\frac{\partial u^2}{\partial y} = 0$, $\frac{\partial v^2}{\partial x} = 0$ and $\nu = 0$.

Case 1: When the time step $(n + 1/2)$

a. x -momentum equation

The finite difference expression for the x -momentum equation using centered differences about u_i in space and about the $(n + 1/2)$ level in time is based on the equation.

$$\left(\frac{\partial u}{\partial t} \right)_{u_{i,j}}^{n+1/2} - (fv)_{u_{i,j}}^n + g \left(\frac{\partial \eta}{\partial x} \right)_{u_{i,j}}^{n+1/2} = \left(\frac{\rho_a}{\rho_0} \frac{1}{H} C_a u_{10} \sqrt{u_{10}^2 + v_{10}^2} \right)_{u_{i,j}}^{n+1/2} - C_D \frac{1}{H} \left(u \sqrt{u^2 + v^2} \right)_{u_{i,j}}^{n+1/2} \quad (2.35)$$

The difference approximations of the terms as follows:

1.

$$\left(\frac{\partial u}{\partial t} \right)_{u_{i,j}}^{n+1/2} \simeq \frac{1}{\Delta t} \left(u_{i,j}^{n+1} - u_{i,j}^n \right) \quad (2.36)$$

2. Discretization of Coriolis force and Barotropic terms

$$(fv)_{u_{i,j}}^n \simeq \frac{1}{8} (f_{i,j} + f_{i+1,j}) \left(v_{i,j+1}^{n+1} + v_{i+1,j+1}^{n+1} + v_{i,j}^{n+1} + v_{i+1,j}^{n+1} \right) \quad (2.37)$$

3.

$$\left(\frac{\partial \eta}{\partial x} \right)_{u_{i,j}}^{n+1/2} \simeq \frac{1}{2\Delta x} \left(\eta_{i+1,j}^{n+1/2} - \eta_{i,j}^{n+1/2} \right) \quad (2.38)$$

Substituting all the approximations (2.36)-(2.38) in to equation (2.35) yield the approximation

$$\begin{aligned}
u_i^{n+1} = & u_i^n + \frac{\Delta t}{8}(f_i + f_{i+1}) \left(v_{i,j+1}^{n+1} + v_{i+1,j+1}^{n+1} + v_{i,j}^{n+1} + v_{i+1,j}^{n+1} \right) \\
& - \frac{g\Delta t}{2\Delta x} (\eta_{i+1}^{n+1/2} - \eta_i^{n+1/2}) \\
& - \frac{2C_D \Delta t}{\left(H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2} \right)} u_{i,j}^{n+1} \sqrt{(u_{i,j}^n)^2 + \frac{1}{16}(v_{i,j+1}^n + v_{i+1,j+1}^n + v_{i,j}^n + v_{i+1,j}^n)^2} \\
& + \frac{\rho_a}{\rho_0} \frac{2C_a \Delta t}{\left(H_{i,j}^{n+1/2} + H_{i+1,j}^{n+1/2} \right)} u_{10} \sqrt{u_{10}^2 + v_{10}^2}
\end{aligned} \tag{2.39}$$

b. y -momentum equation

At the v_i point and the $(n + 1/2)$ time level the depth-averaged y -momentum namely

$$\left(\frac{\partial v}{\partial t} \right)_{v_i}^{n+1/2} - (fu)_{v_i,j}^n + g \left(\frac{\partial \eta}{\partial x} \right)_{v_i,j}^{n+1/2} = \left(\frac{\rho_a}{\rho_0} \frac{1}{H} C_a v_{10} \sqrt{u_{10}^2 + v_{10}^2} \right)_{v_i,j}^{n+1/2} - C_D \frac{1}{H} \left(v \sqrt{u^2 + v^2} \right)_{v_i,j}^{n+1/2} \tag{2.40}$$

The difference approximations of the terms as follows:

1.

$$\left(\frac{\partial v}{\partial t} \right)_{v_i}^{n+1/2} \simeq \frac{1}{\Delta t} (v_i^{n+1} - v_i^n) \tag{2.41}$$

2.

$$(fu)_{v_i}^{n+1/2} \simeq \frac{1}{8} (f_{i,j} + f_{i,j-1}) \left(u_{i-1,j}^{n+1} + u_{i,j}^{n+1} + u_{i,j-1}^{n+1} + u_{i-1,j-1}^{n+1} \right) \tag{2.42}$$

3.

$$\left(\frac{\partial \eta}{\partial y} \right)_{v_i}^{n+1/2} \simeq \frac{1}{\Delta y} \left(\eta_j^{n+1/2} - \eta_{j-1}^{n+1/2} \right) \tag{2.43}$$

Substituting all the approximations (2.41)-(2.43) in to equation (2.40) yield the approximation

$$\begin{aligned}
v_i^{n+1} = & v_i^n + \frac{\Delta t}{8}(f_{i,j} + f_{i,j-1}) \left(u_{i-1,j}^{n+1} + u_{i,j-1}^{n+1} + u_{i,j}^{n+1} + u_{i-1,j-1}^{n+1} \right) \\
& - \frac{g\Delta t}{2\Delta y} (\eta_j^{n+1/2} - \eta_{j-1}^{n+1/2}) \\
& - \frac{2C_D \Delta t}{\left(H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2} \right)} v_{i,j}^{n+1} \sqrt{(v_{i,j}^n)^2 + \frac{1}{16}(u_{i,j}^{n+1} + u_{i-1,j}^{n+1} + u_{i,j-1}^{n+1} + u_{i-1,j-1}^{n+1})^2} \\
& + \frac{\rho_a}{\rho_0} \frac{2C_a \Delta t}{\left(H_{i,j}^{n+1/2} + H_{i,j-1}^{n+1/2} \right)} v_{10} \sqrt{u_{10}^2 + v_{10}^2}
\end{aligned} \tag{2.44}$$

Case 2: When the time step $(n + 1)$

We can written the 2D linear SWEs as following:

Continuity equation

As the same equation (2.1) in Section 2.3

$$\begin{aligned} \eta_{i,j}^{n+1} = & \eta_{i,j}^{n-1} - \frac{\Delta t}{\Delta x} [(H_{i,j}^n + H_{i+1,j}^n)u_{i+1,j}^n - (H_{i-1,j}^n + H_{i,j}^n)u_{i,j}^n] \\ & - \frac{\Delta t}{\Delta y} [(H_{i,j}^n + H_{i,j+1}^n)v_{i,j+1}^n - (H_{i,j}^n + H_{i,j-1}^n)v_{i,j}^n] \end{aligned} \quad (2.45)$$

a. x -momentum equation

$$\begin{aligned} u_{i,j}^{n+1} = & \left[u_{i,j}^{n-1}(H_{i,j}^{n-1} + H_{i-1,j}^{n-1}) + \frac{\Delta t}{2} f(H_{i,j}^n + H_{i-1,j}^n) \times (v_{i-1,j}^n + v_{i,j}^n + v_{i,j+1}^n + v_{i-1,j+1}^n) \right. \\ & - g \frac{2\Delta t}{\Delta x} (H_{i,j}^n + H_{i-1,j}^n)(\eta_{i,j}^n - \eta_{i-1,j}^n) + 4\Delta t \rho_a / \rho_0 C_a u_{10} \sqrt{u_{10}^2 + v_{10}^2} \\ & \left. - 4\Delta t C_D u_{i,j}^{n+1} \sqrt{(u_{i,j}^n)^2 + (v_{i-1,j}^n + v_{i,j}^n + v_{i,j+1}^n + v_{i-1,j+1}^n)^2} \right] / (H_{i,j}^{n+1} + H_{i-1,j}^{n+1}) \end{aligned} \quad (2.46)$$

b. y -momentum equation

$$\begin{aligned} v_{i^*,j^*}^{n+1} = & \left[v_{i^*,j^*}^{n-1}(H_{i^*,j^*}^{n-1} + H_{i^*,j^*-1}^{n-1}) - \frac{\Delta t}{2} f(H_{i^*,j^*}^n + H_{i^*,j^*-1}^n) \times (u_{i^*,j^*-1}^n + u_{i^*,j^*}^n + u_{i^*+1,j^*}^n + u_{i^*+1,j^*-1}^n) \right. \\ & - g \frac{2\Delta t}{\Delta y} (H_{i^*,j^*}^n + H_{i^*,j^*-1}^n)(\eta_{i^*,j^*}^n - \eta_{i^*,j^*-1}^n) + 4\Delta t \rho_a / \rho_0 C_a v_{10} \sqrt{v_{10}^2 + u_{10}^2} \\ & \left. - 4\Delta t C_D v_{i^*,j^*}^{n+1} \sqrt{(v_{i^*,j^*}^n)^2 + (u_{i^*,j^*-1}^n + u_{i^*,j^*}^n + u_{i^*+1,j^*}^n + u_{i^*+1,j^*-1}^n)^2} \right] / (H_{i^*,j^*}^{n+1} + H_{i^*,j^*-1}^{n+1}) \end{aligned} \quad (2.47)$$

2.7 Program Logic and Data Flow

In this section, the structure of the computational program and some major subroutines are briefly described.

Flow charts: (Organizational chart)

The major structures and functionalities of the program are demonstrated. It can be seen that the program has three major parts: Initialization or configuration, shallow water model, and transport. Main components of the first part, initialization are mesh or grid systems and finite difference operators initialization after data files are read. All the physical and numerical parameters used in the model are provided for users and can be changed according to the condition of the simulation. New runs will

review the parameters used for the last time as well, they will be used if no change has been made. It is clear the sequence of the flow chart includes solving momentum equations and the continuity equation. Boundary conditions have symbolic meaning that there are applied to all processes involved. Figure 2-5 shows the flowchart of the model.

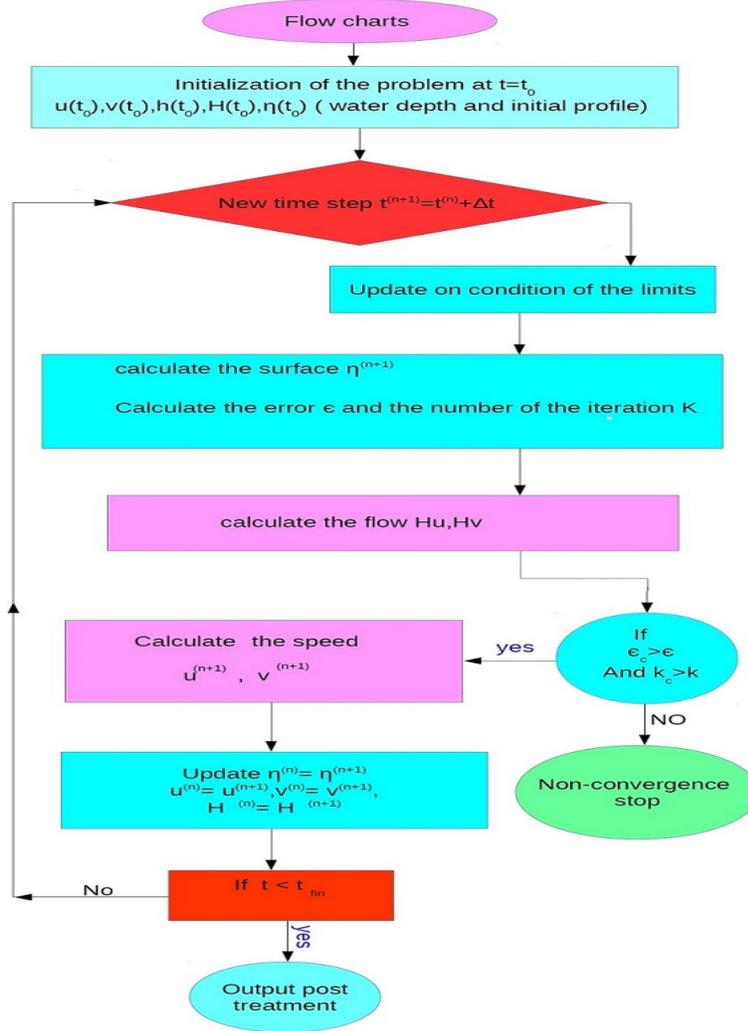


Figure 2-5: Show organization chart of the calculation program

2.8 Stability Criteria

A successful numerical scheme must approach the solution without staining acute regions or introducing false oscillations. It should approximate the solution increasingly accurately as the number of grid cells is increased with the property that the numerical solution tends to the actual solution as the grid size tends to zero.

Since the solution to a hyperbolic or parabolic problem, we must ensure that the computational grid is such that the time step Δt is small enough that the wave does not have time to exit the other side of the element of width Δx . This leads us to the following important condition. (i.e., the accuracy of the solution depends on the chosen values of Δt and Δx .)

2.8.1 Consistency, convergence and stability ([31, 113])

Definition of consistency

Given a partial differential equations $Lu = f$ and a finite difference scheme $L_{\Delta t, \Delta x}v = f$, we say that the finite difference scheme is **consistent** with the PDE if for any smooth function $\varphi(t, x)$

$$(L\varphi - L_{\Delta t, \Delta x}\varphi) \rightarrow 0 \quad \text{as} \quad \Delta t, \Delta x \rightarrow 0$$

Definitions of convergency

A one-step finite difference scheme approximating a PDE is a **convergent scheme** if for any solution to the PDE $u(t, x)$ and the solution of the finite difference scheme v_m^n , such that v_m^0 converges to $u_0(x)$ as $(m\Delta x)$ converges to x then v_m^n converges to $u(t, x)$ as $(n\Delta t, m\Delta x)$ convergence to (t, x) as $\Delta t, \Delta x \rightarrow 0$. The notation v_m^n means the value of v at the grid point (t_n, x_m) .

Definitions of stability

A finite difference scheme $P_{\Delta t, \Delta x}v_i^n = 0$ for a first-order equation is **stable** in a stability region Λ if there is an integer J such that for any positive time T , there is a constant C_T , such that

$$\|v^n\|_{\Delta x} \leq C_T \sum_{j=0}^J \|v^j\|_{\Delta x}$$

for $0 \leq \Delta t \leq T$, with $(\Delta t, \Delta x) \in \Lambda$

Lax-Richtmyer equivalence theorem

In numerical analysis, the Lax equivalence theorem is the fundamental theorem in the analysis of finite difference methods for the numerical solution of partial differential equations. It states that for (see [92, 113]):

A consistent finite difference scheme for a partial differential equation for a well-posed linear initial value problem, the method is convergent if and only if it is stable.

$$\text{Consistency} + \text{Stability} \Leftrightarrow \text{Convergence}$$

Notice that

- The Lax theorem does not apply to non-linear PDE because consistency and stability are often insufficient for convergence and convergence need not imply stability in general. Also, the numerical method can convergence very plausibly to incorrect results despite being consistent and stable and sometimes in non-linear PDE difficult to find the exact solution.

- Any numerical method that violates the CFL condition misses information affecting the exact solution and may blow up to infinity. For this reason, the CFL condition is necessary but not sufficient for numerical stability.

2.8.2 The Courant Friedrichs Lewy (CFL) Condition

In (1928) Richard Courant, Kurt Friedrichs and Hans Lewy of the university of Gottingen in Germany published a famous paper in mathematics, the Courant Friedrichs Lewy (CFL) condition is a necessary condition for convergence while solving certain partial differential equations (usually hyperbolic PDEs) numerically by the method of finite differences ([42, 92, 113]). It states:

The numerical domain of dependence must contain the physical domain of dependence.

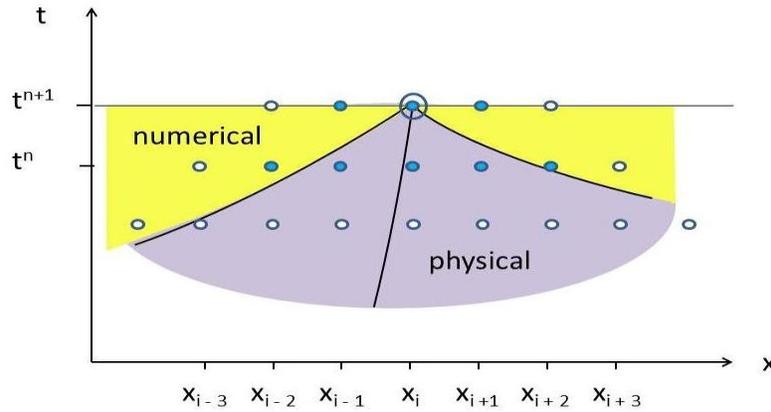


Figure 2-6: The numerical and physical domain of dependence.

The Courant condition, is a condition for convergence while numerically solving PDEs. It places a constraint on the maximum time step that can be used for a chosen grid spacing in order to ensure the stability of the numerical solution. The condition is often expressed in terms of the Courant number C_n defined as: ([16, 53, 107])

$$C_n = v \frac{\Delta t}{\Delta x}$$

where v is the celerity of the gravity wave. Specifying wave celerity, above equation becomes:

$$C_n = \sqrt{gH} \frac{\Delta t}{\Delta x}$$

2.8.3 The Courant Friedrichs Lewy (CFL) Condition for 2DSWEs

The Courant number CFL condition for 2D shallow water equations is defined as follows: [49, 69, 110]

$$C = \Delta t \left(\sqrt{gH_{max}} \right) \left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} \right)^{\frac{1}{2}}$$

or

$$\max \left[\left(2\Delta t \sqrt{gH} \right) \left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} \right) \right] < 1$$

Using the stability condition $CFL < 1$ in above equation. The following stability criterion is obtained for the optimal time step [75]

$$\Delta t \leq \frac{\Delta x \Delta y}{\left(\sqrt{gH_{max}} \right) \sqrt{(\Delta x^2 + \Delta y^2)}}$$

where H_{max} is the maximum depth water.

2.8.4 Derivation the Courant Friedrichs Lewy (CFL) condition in 2D shallow water equations

The Courant number CFL condition for 2DSWEs is defined as follows [69]:

$$C = \Delta t \left(\sqrt{gH} + V_{max} \right) \left(\frac{1}{\Delta x} + \frac{1}{\Delta y} \right)$$

Using the stability condition $CFL < 1$ in above equation. The following stability criterion is obtained for the optimal time step

$$\Delta t \leq \frac{1}{\left(\sqrt{gH} + V_{max} \right) (\Delta x^{-1} + \Delta y^{-1})}$$

To prove this CFL condition for the 2DSWEs

Consider the 2DNSWEs given in equations (1.29)-(1.31), when $f=0$, $\nu=0$, $\tau_u^w = 0$, $\tau_u^b = 0$, $\tau_v^w = 0$ and $\tau_v^b = 0$ and dividing on H , we obtain:

$$\frac{\partial \eta}{\partial t} + \frac{\partial(Hu)}{\partial x} + \frac{\partial(Hv)}{\partial y} = 0$$

$$\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x} + \frac{\partial(uv)}{\partial y} + g \frac{\partial \eta}{\partial x} = 0$$

$$\frac{\partial v}{\partial t} + \frac{\partial(vu)}{\partial y} + \frac{\partial v^2}{\partial y} + g \frac{\partial \eta}{\partial y} = 0$$

The 2D shallow-water equations can be written in the matrix form

$$\frac{\partial}{\partial t} \begin{pmatrix} \eta \\ u \\ v \end{pmatrix} + \begin{pmatrix} u & H & 0 \\ g & u & 0 \\ 0 & 0 & u \end{pmatrix} \frac{\partial}{\partial x} \begin{pmatrix} \eta \\ u \\ v \end{pmatrix} + \begin{pmatrix} v & H & 0 \\ 0 & v & 0 \\ g & 0 & v \end{pmatrix} \frac{\partial}{\partial y} \begin{pmatrix} \eta \\ u \\ v \end{pmatrix} = - \begin{pmatrix} u \frac{\partial h}{\partial x} + v \frac{\partial h}{\partial y} \\ 0 \\ 0 \end{pmatrix}$$

We can easily to check the eigenvalues of the first coefficient matrix are $\{u, u \pm \sqrt{gH}\}$ and the eigenvalues of the second coefficient matrix are $\{v, v \pm \sqrt{gH}\}$. The CFL in two-dimension case as follows:

$$C_r = \left(\frac{u_x \Delta t}{\Delta x} + \frac{v_y \Delta t}{\Delta y} \right) \leq 1$$

where u_x, u_y are the velocity in x, y -directions respectively.

where $u_x = \{u, u \pm \sqrt{gH}\}$ and $u_y = \{v, v \pm \sqrt{gH}\}$

$$\Rightarrow CFL = \left(\max \{u, u \pm \sqrt{gH}\} \frac{\Delta t}{\Delta x} + \max \{v, v \pm \sqrt{gH}\} \frac{\Delta t}{\Delta y} \right) < 1$$

Let

$$\begin{aligned} V_{max} + \sqrt{gH} &= \max \left\{ (u, u \pm \sqrt{gH}), (v, v \pm \sqrt{gH}) \right\} \\ &\Rightarrow (V_{max} + \sqrt{gH}) \left(\frac{\Delta t}{\Delta x} + \frac{\Delta t}{\Delta y} \right) < 1 \\ &\Rightarrow \Delta t \left(\frac{1}{\Delta x} + \frac{1}{\Delta y} \right) < \frac{1}{(V_{max} + \sqrt{gH})} \\ &\Rightarrow \Delta t < \frac{1}{(V_{max} + \sqrt{gH}) (\Delta x^{-1} + \Delta y^{-1})} \end{aligned}$$

Similarly, we can prove the another formula

Since

$$\begin{aligned} &\Rightarrow (V_{max} + \sqrt{gH}) \left(\frac{\Delta t}{\Delta x} + \frac{\Delta t}{\Delta y} \right) < 1 \\ &\Rightarrow (V_{max} + \sqrt{gH}) \left(\frac{\Delta t}{\Delta x^2} + \frac{\Delta t}{\Delta y^2} \right) < 1 \\ &\Rightarrow \Delta t \left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} \right)^{\frac{1}{2}} < \frac{1}{(V_{max} + \sqrt{gH})} \\ &\Rightarrow \Delta t < \frac{1}{(V_{max} + \sqrt{gH}) (\sqrt{\Delta x^2 + \Delta y^2})} \end{aligned}$$

2.9 Studying Boundary Conditions

2.9.1 Radiative open boundary condition

In this section, we derive the radiative open boundary conditions on C-grid for the 2DSWEs provided by [61, 100]:

Case 1: Western boundary radiative condition

The western boundary radiative condition is defined, for the elevation and the component of velocity perpendicular to the boundary.

1. For the elevation and the component of velocity perpendicular to the boundary is defined:

$$\begin{aligned}\eta_{1,j}^{l+1} &= \eta_{1,j}^l - 2\frac{\Delta t}{\Delta x}\sqrt{gH_{1,j}^l}\left(\eta_{1,j}^l - \eta_{2,j}^l\right) \\ u_{1,j}^{l+1} &= -\sqrt{\frac{g}{H_{1,j}^{l+1}}}\eta_{1,j}^{l+1} \\ H_{1,j}^{l+1} &= \eta_{1,j}^{l+1} + h_{1,j}\end{aligned}$$

for $j = 1, 2, \dots, N$ and for the velocity component tangent to the boundary is defined by:

$$v_{1,j}^{l+1} = \left(v_{1,j}^l (H_{1,j}^l + H_{1,j-1}^l) - 2\frac{\Delta t}{\Delta x}\sqrt{g\frac{(H_{1,j}^l + H_{1,j-1}^l)}{2}}(v_{1,j}^l - v_{2,j}^l) \right) / (H_{1,j}^{l+1} + H_{1,j-1}^{l+1})$$

for $j = 2, 3, \dots, N$.

Notes:

1. If $j = 1$ then the boundary radiative condition becomes:

1. For the elevation and the component of velocity perpendicular to the boundary is defined

$$\begin{aligned}\eta_{1,1}^{l+1} &= \eta_{1,1}^l - 2\frac{\Delta t}{\Delta x}\sqrt{gH_{1,1}^l}\left(\eta_{1,1}^l - \eta_{2,1}^l\right) \\ u_{1,1}^{l+1} &= -\sqrt{\frac{g}{H_{1,1}^{l+1}}}\eta_{1,1}^{l+1}\end{aligned}$$

and for the velocity component tangent to the boundary is defined by:

$$v_{1,2}^{l+1} = \left(v_{1,2}^l (H_{1,2}^l + H_{1,1}^l) - 2\frac{\Delta t}{\Delta x}\sqrt{g\frac{(H_{1,2}^l + H_{1,1}^l)}{2}}(v_{1,2}^l - v_{2,2}^l) \right) / (H_{1,2}^{l+1} + H_{1,1}^{l+1})$$

if $j = 2$.

2. If $i = 1$ and $j = N$ then the boundary radiative condition becomes:

1. For the elevation and the component of velocity perpendicular to the boundary is defined by:

$$\begin{aligned}\eta_{1,N}^{l+1} &= \eta_{1,N}^l - 2\frac{\Delta t}{\Delta x}\sqrt{gH_{1,N}^l}\left(\eta_{1,N}^l - \eta_{2,N}^l\right) \\ u_{1,N}^{l+1} &= -\sqrt{\frac{g}{H_{1,N}^{l+1}}}\eta_{1,N}^{l+1} \\ H_{1,N}^{l+1} &= \eta_{1,N}^{l+1} + h_{1,N}\end{aligned}$$

For the velocity component tangent to the boundary is defined by:

$$v_{1,N}^{l+1} = \left(v_{1,N}^l (H_{1,N}^l + H_{1,N-1}^l) - 2\frac{\Delta t}{\Delta x}\sqrt{g\frac{(H_{1,N}^l + H_{1,N-1}^l)}{2}}\left(v_{1,N}^l - v_{2,N}^l\right) \right) / \left(H_{1,N}^{l+1} + H_{1,N-1}^{l+1} \right)$$

Case2: Eastern boundary radiative condition

For the elevation and the component of velocity perpendicular to the boundary is defined

$$\begin{aligned}\eta_{M,j}^{l+1} &= \eta_{M,j}^l - 2\frac{\Delta t}{\Delta x}\sqrt{gH_{M,j}^l}\left(\eta_{M,j}^l - \eta_{M-1,j}^l\right) \\ u_{M+1,j}^{l+1} &= -\sqrt{\frac{g}{H_{M,j}^{l+1}}}\eta_{M,j}^{l+1} \\ H_{M,j}^{l+1} &= \eta_{M,j}^{l+1} + h_{M,j}\end{aligned}$$

for $j = 1, 2, \dots, N$ and for the velocity component tangent to the boundary is defined by:

$$v_{M,j}^{l+1} = \left(v_{M,j}^l (H_{M,j}^l + H_{M,j-1}^l) - 2\frac{\Delta t}{\Delta x}\sqrt{g\frac{(H_{M,j}^l + H_{M,j-1}^l)}{2}}\left(v_{M,j}^l - v_{M-1,j}^l\right) \right) / \left(H_{M,j}^{l+1} + H_{M,j-1}^{l+1} \right)$$

for $j = 2, 3, \dots, N$.

Notes:

1. If $i = M$ and $j = N$ then the boundary radiative condition becomes:

1. For the elevation and the component of velocity perpendicular to the boundary is defined

$$\begin{aligned}\eta_{M,N}^{l+1} &= \eta_{M,N}^l - 2\frac{\Delta t}{\Delta x}\sqrt{gH_{M,N}^l}\left(\eta_{M,N}^l - \eta_{M-1,N}^l\right) \\ u_{M+1,N}^{l+1} &= -\sqrt{\frac{g}{H_{M,N}^{l+1}}}\eta_{M,N}^{l+1} \\ H_{M,N}^{l+1} &= \eta_{M,N}^{l+1} + h_{M,N}\end{aligned}$$

For the velocity component tangent to the boundary is defined by:

$$v_{M,N}^{l+1} = \left(v_{M,N}^l (H_{M,N}^l + H_{M,N-1}^l) - 2\frac{\Delta t}{\Delta x}\sqrt{g\frac{(H_{M,N}^l + H_{M,N-1}^l)}{2}}\left(v_{M,N}^l - v_{M-1,N}^l\right) \right) / \left(H_{M,N}^{l+1} + H_{M,N-1}^{l+1} \right)$$

2. If $i = M$ and $j = 2$ then the boundary radiative condition becomes:

1. For the elevation and the component of velocity perpendicular to the boundary is defined

$$\begin{aligned}\eta_{M,2}^{l+1} &= \eta_{M,2}^l - 2\frac{\Delta t}{\Delta x}\sqrt{gH_{M,2}^l}\left(\eta_{M,2}^l - \eta_{M-1,2}^l\right) \\ u_{M+1,2}^{l+1} &= -\sqrt{\frac{g}{H_{M,2}^{l+1}}}\eta_{M,2}^{l+1} \\ H_{M,2}^{l+1} &= \eta_{M,2}^{l+1} + h_{M,2}\end{aligned}$$

for the velocity component tangent to the boundary is defined by:

$$v_{M,2}^{l+1} = \left(v_{M,2}^l (H_{M,2}^l + H_{M,1}^l) - 2\frac{\Delta t}{\Delta x}\sqrt{g\frac{(H_{M,2}^l + H_{M-1}^l)}{2}}(v_{M,2}^l - v_{M-1,2}^l) \right) / (H_{M,2}^{l+1} + H_{M,1}^{l+1})$$

Case 3: The Southern boundary radiative condition is defined

For the elevation and the component of velocity perpendicular to the boundary

$$\begin{aligned}\eta_{i,1}^{l+1} &= \eta_{i,1}^l - 2\frac{\Delta t}{\Delta y}\sqrt{gH_{i,1}^l}\left(\eta_{i,1}^l - \eta_{i,2}^l\right) \\ v_{i,1}^{l+1} &= -\sqrt{\frac{g}{H_{i,1}^{l+1}}}\eta_{i,1}^{l+1} \\ H_{i,1}^{l+1} &= \eta_{i,1}^{l+1} + h_{i,1}\end{aligned}$$

for $i = 1, 2, \dots, M$ and for the velocity component tangent to the boundary is defined by:

$$u_{i,1}^{l+1} = \left(u_{i,1}^l (H_{i,1}^l + H_{i-1,1}^l) - 2\frac{\Delta t}{\Delta y}\sqrt{g\frac{(H_{i,1}^l + H_{i-1,1}^l)}{2}}(u_{i,1}^l - u_{i,2}^l) \right) / (H_{i,1}^{l+1} + H_{i-1,1}^{l+1})$$

for $i = 2, 3, \dots, M$.

1. If $i = 1$ for the elevation and the component of velocity perpendicular to the boundary

$$\begin{aligned}\eta_{1,1}^{l+1} &= \eta_{1,1}^l - 2\frac{\Delta t}{\Delta y}\sqrt{gH_{1,1}^l}\left(\eta_{1,1}^l - \eta_{1,2}^l\right) \\ v_{1,1}^{l+1} &= -\sqrt{\frac{g}{H_{1,1}^{l+1}}}\eta_{1,1}^{l+1} \\ H_{1,1}^{l+1} &= \eta_{1,1}^{l+1} + h_{1,1}\end{aligned}$$

and for the velocity component tangent to the boundary is defined by:

$$u_{2,1}^{l+1} = \left(u_{2,1}^l (H_{2,1}^l + H_{1,1}^l) - 2\frac{\Delta t}{\Delta y}\sqrt{g\frac{(H_{2,1}^l + H_{1,1}^l)}{2}}(u_{2,1}^l - u_{2,2}^l) \right) / (H_{2,1}^{l+1} + H_{1,1}^{l+1})$$

for $i = 2$.

2. if $i = M$, for the elevation and the component of velocity perpendicular to the boundary

$$\begin{aligned}\eta_{M,1}^{l+1} &= \eta_{M,1}^l - 2 \frac{\Delta t}{\Delta y} \sqrt{g H_{M,1}^l} \left(\eta_{M,1}^l - \eta_{M,2}^l \right) \\ v_{M,1}^{l+1} &= - \sqrt{\frac{g}{H_{M,1}^{l+1}}} \eta_{M,1}^{l+1} \\ H_{M,1}^{l+1} &= \eta_{M,1}^{l+1} + h_{M,1}\end{aligned}$$

For the velocity component tangent to the boundary is defined by:

$$u_{M,1}^{l+1} = \left(u_{M,1}^l (H_{M,1}^l + H_{M-1,1}^l) - 2 \frac{\Delta t}{\Delta y} \sqrt{g \frac{(H_{M,1}^l + H_{M-1,1}^l)}{2}} \left(u_{M,1}^l - u_{M,2}^l \right) \right) / \left(H_{M,1}^{l+1} + H_{M-1,1}^{l+1} \right)$$

Case 4: Northern boundary radiative condition

For the elevation and the component of velocity perpendicular to the boundary is defined:

$$\begin{aligned}\eta_{i,N}^{l+1} &= \eta_{i,N}^l - 2 \frac{\Delta t}{\Delta y} \sqrt{g H_{i,N}^l} \left(\eta_{i,N}^l - \eta_{i,N-1}^l \right) \\ v_{i,N+1}^{l+1} &= - \sqrt{\frac{g}{H_{i,N}^{l+1}}} \eta_{i,N}^{l+1} \\ H_{i,N}^{l+1} &= \eta_{i,N}^{l+1} + h_{i,N}\end{aligned}$$

for $i = 1, 2, \dots, M$ and for the velocity component tangent to the boundary is defined by:

$$u_{i,N}^{l+1} = \left(u_{i,N}^l (H_{i,N}^l + H_{i-1,N}^l) - 2 \frac{\Delta t}{\Delta y} \sqrt{g \frac{(H_{i,N}^l + H_{i-1,N}^l)}{2}} \left(u_{i,N}^l - u_{i,N-1}^l \right) \right) / \left(H_{i,N}^{l+1} + H_{i-1,N}^{l+1} \right)$$

for $i = 2, 3, \dots, M$.

1. if $i = 1$ then for the elevation and the component of velocity perpendicular to the boundary is defined

$$\begin{aligned}\eta_{1,N}^{l+1} &= \eta_{1,N}^l - 2 \frac{\Delta t}{\Delta y} \sqrt{g H_{1,N}^l} \left(\eta_{1,N}^l - \eta_{1,N-1}^l \right) \\ v_{1,N+1}^{l+1} &= - \sqrt{\frac{g}{H_{1,N}^{l+1}}} \eta_{1,N}^{l+1} \\ H_{1,N}^{l+1} &= \eta_{1,N}^{l+1} + h_{1,N}\end{aligned}$$

for the velocity component tangent to the boundary is defined by:

$$u_{2,N}^{l+1} = \left(u_{2,N}^l (H_{2,N}^l + H_{1,N}^l) - 2 \frac{\Delta t}{\Delta y} \sqrt{g \frac{(H_{2,N}^l + H_{1,N}^l)}{2}} \left(u_{2,N}^l - u_{2,N-1}^l \right) \right) / \left(H_{2,N}^{l+1} + H_{1,N}^{l+1} \right)$$

for $i = 2$.

2. If $i = M$ Northern boundary radiative condition

For the elevation and the component of velocity perpendicular to the boundary is defined

$$\eta_{M,N}^{l+1} = \eta_{M,N}^l - 2 \frac{\Delta t}{\Delta y} \sqrt{g H_{M,N}^l} (\eta_{M,N}^l - \eta_{M,N-1}^l)$$

$$v_{M,N+1}^{l+1} = - \sqrt{\frac{g}{H_{M,N}^{l+1}}} \eta_{M,N}^{l+1}$$

$$H_{M,N}^{l+1} = \eta_{M,N}^{l+1} + h_{M,N}$$

For the velocity component tangent to the boundary is defined by:

$$u_{M,N}^{l+1} = \left(u_{M,N}^l (H_{M,N}^l + H_{M-1,N}^l) - 2 \frac{\Delta t}{\Delta y} \sqrt{g \frac{(H_{M,N}^l + H_{M-1,N}^l)}{2}} (u_{M,N}^l - u_{M,N-1}^l) \right) / (H_{M,N}^{l+1} + H_{M-1,N}^{l+1})$$

2.10 Summary and Conclusion

In this chapter, development of numerical methods has been suggested for 2D shallow water model using explicit methods. Several cases of numerical discretization for 2D depth-averaged shallow water equations has been applied using center finite difference method in space and leapfrog with Asselin-Roberts filtering in time. Some kinds of boundary conditions have been studied. Several algorithms were proposed for 2DSWEs. Accordingly, the new techniques and algorithms presented above presented will be used in the **later chapters**.

Chapter 3

Numerical Results for Shallow Water Equations

Chapter 3

Numerical Results for 2D Shallow Water Equations

Some of the results presented in this chapter (Sections 3.1 and 3.2) are the subject of an article [8].

Shallow water equations in two-dimensional are used for some ocean models applications using the new discretization techniques for numerical methods suggested in Chapter 2.

Dirichlet, reflexive boundary conditions and moving boundary conditions are applied at the boundaries and implementation numerical simulations are conducted by computer programming using Matlab and Fortran 90. The performance of the proposed technique is tested on several examples of the tsunami model. Some examples of 2DSWEs are applied using Gaussian level initial condition. The numerical results indicate that the model has high accuracy and efficiency by using these techniques.

Highlights

- Some applications for 2DNSWEs are implemented using EFDM in space and leapfrog method with Robert-Asselin filter in time.
- Some examples of 2DSWEs are applied using Gaussian level initial condition.
- L_2 -relative error norm is used to show high accuracy of the results.

Contents

3.1 Numerical solutions For 2DSWEs	104
3.2 Some applications of 2D depth-averaged non-linear SWEs	109
3.3 Some examples of 2DSWEs are applied using Gaussian level initial condition	112
3.4 Summary and Conclusion For This Part	115

3.1 Numerical Solutions for 2DSWEs

Note: For all examples, we apply EFDM in space and leapfrog method with Robert-Asselin filter in time to approximate the 2D depth-averaged non-linear SWEs. For the stability using CFL condition which is given in Section 2.8.3.

Example 1

In this example, we use 2DNSWEs given in equations (2.1)-(2.3) which are approximately first order in space and second order in time, when the bottom stress, wind stress, $f=0$, and viscosity horizontal in x -axis and y -axis are equal zero in a rectangular domain $\Omega = [0, L_x] \times [0, L_y]$

1. Initial condition:

Initially, the water is at rest with a water drop of 10m and a zero flow everywhere.i.e., no-slip boundary conditions $u = v = 0$ on $\partial\Omega$ and

$$\eta(x, y, t = 0) = 10 * \exp((-5((x)^2 + (y)^2)))$$

$$u(x, y, t = 0) = 0, v(x, y, t = 0) = 0$$

2. Boundary conditions:

Here, reflexive boundary conditions were implemented at the boundaries with CFL condition 0.13.

$$\begin{aligned} \eta(x, 1) &= \eta(x, 2); & u(x, 1) &= u(x, 2); & v(x, 1) &= -v(x, 2); \\ \eta(x, n) &= \eta(x, n - 1); & u(x, n) &= u(x, n - 1); & v(x, n) &= -v(x, n - 1); \\ \eta(1, y) &= \eta(2, y); & u(1, y) &= -u(2, y); & v(1, y) &= v(2, y); \\ \eta(n, y) &= \eta(n - 1, y); & u(n, y) &= -u(n - 1, y); & v(n, y) &= v(n - 1, y); \end{aligned}$$

3. Numerical parameters: (configurations)

The computational domain is discretized by a grid whose size is regular. Numerical values of the parameters are chosen as follows: $L_x=L_y=200$, $\Delta x=\Delta y=0.10$, the time step $\Delta t=0.001$ sec, and the simulation duration $t = 100$ days.

4. Results and discussion:

First of all, we tested the computational stability of this model by using CFL condition which is 0.13 less than 1 (see Section 2.8.3). The time integrations were performed for 100 days and the

calculations were stable.

A global l_2 -relative error norm in space (l_2 - RE) which means l_2 -relative error at all grids point in the model domain is formulated as (see [76]):

$$RE(Q_{i,j}) = \sqrt{\sum_j^{M_l} \sum_i^{N_c} \left[\left(\frac{Q_{i,j}^n - Q_{i,j}^{n-1}}{Q_{i,j}^n} \right)^2 \right]}$$

where N_c , M_l represents the number of cells, $Q_{i,j}$ is the computed values of the variables for the i -th and j -th cells, when n and $n-1$ denote the current and previous time levels. A smaller RE represents less variation in time and thus manifests the results closer to the steady state.

The following figure shows the comparison l_2 - RE for free surface elevation, u -velocity and v -velocity for 2DNSWEs.

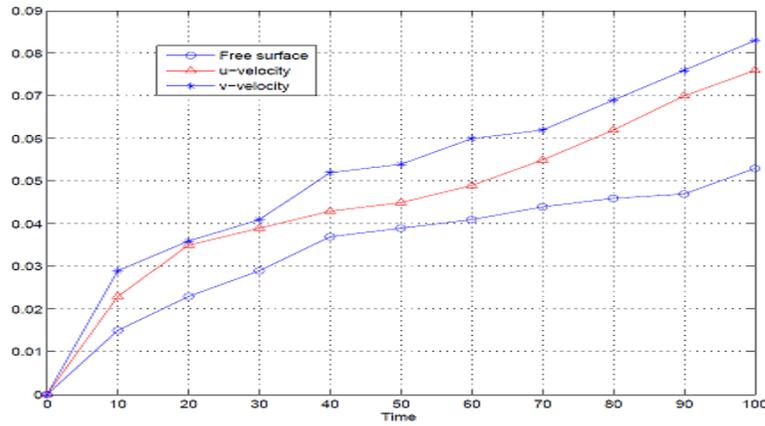


Figure 3-1: Comparison l_2 - RE of free surface elevation, u -velocity and v -velocity for 2DNSWEs at $t = 10, 20, \dots, 100$ days.

Example 2:

In this example, we use 2D linear SWEs given in equations (2.45)-(2.47) which are approximately first order in space and second order in time.

1. Initial condition:

Initially, the water is at rest with a water drop of 1.6m and a zero flow everywhere.

$$\eta(i, j) = 10 * \exp(-((i - i_0)^2 + (j - j_0)^2)/(k^2))$$

where $i_0 = 15$, $j_0 = 15$ and $k = 6$.

2. Boundary conditions:

Here, Dirichlet boundary conditions were implemented at the boundaries with CFL condition 0.22.

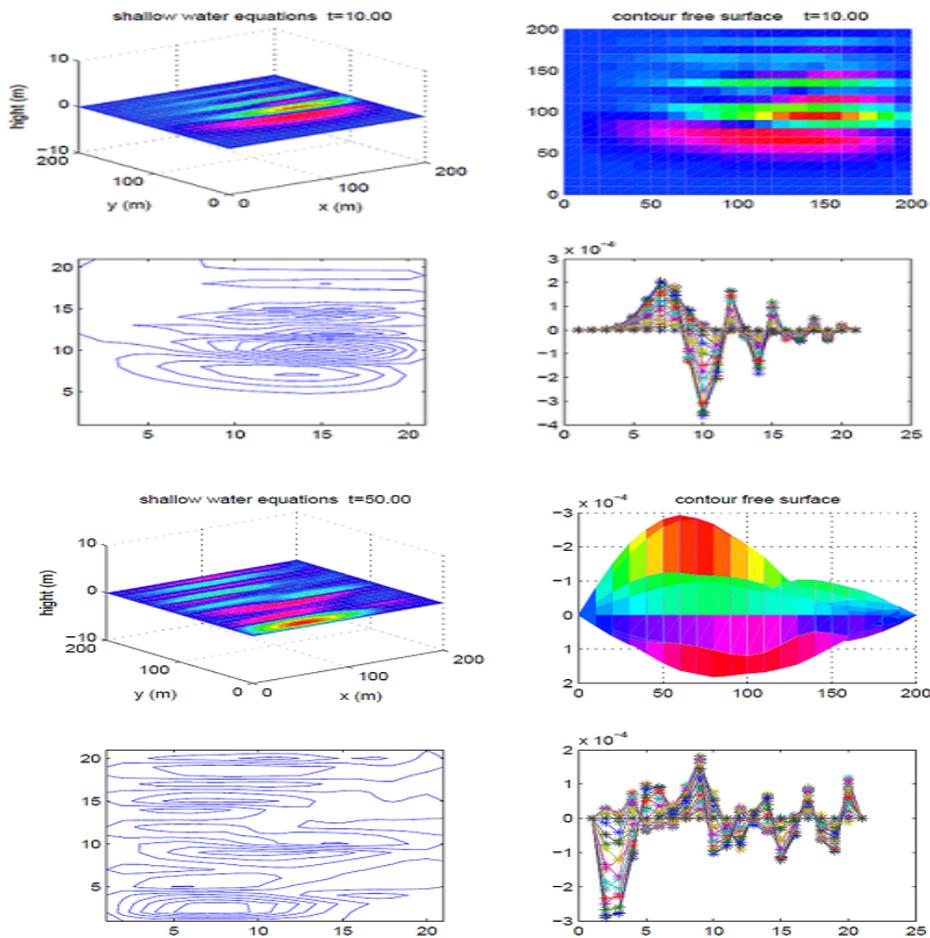
3. Numerical parameters: (configurations)

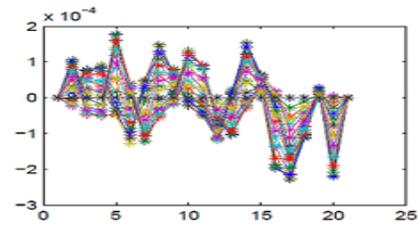
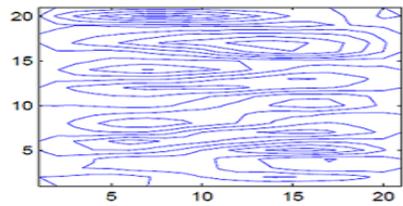
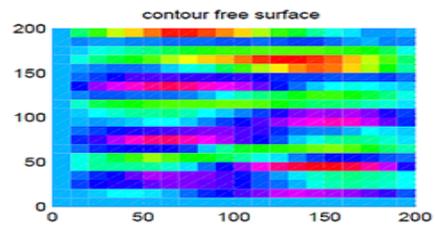
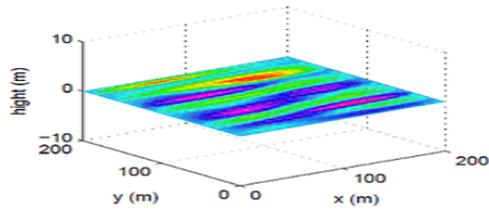
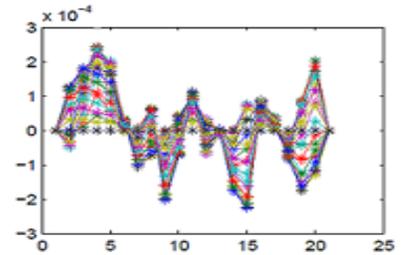
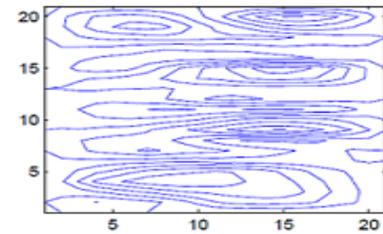
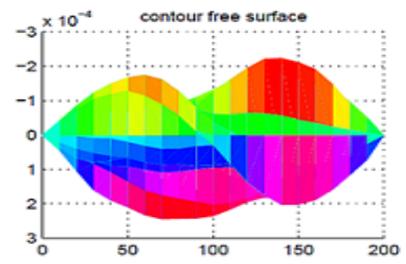
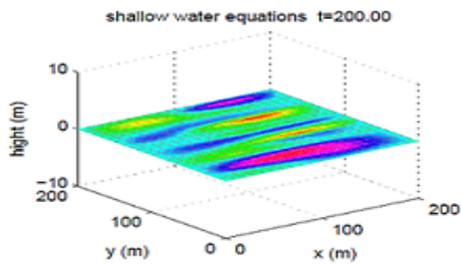
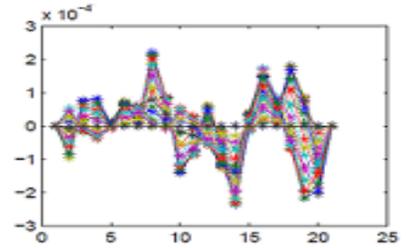
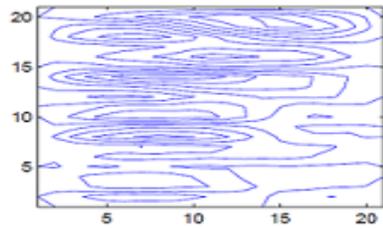
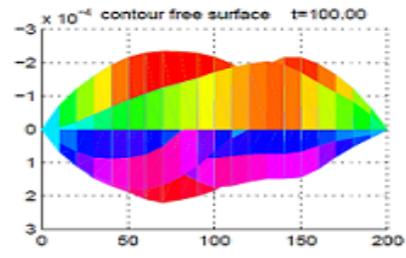
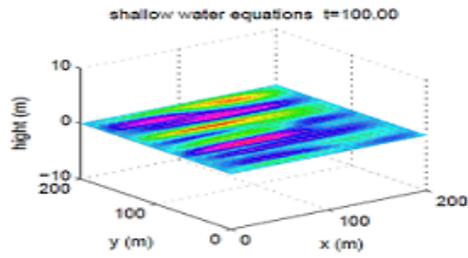
The computational domain is discretized by a grid whose size is regular. Numerical values of the parameters are chosen as follows: $L_x=L_y = 200$, $\Delta x =0.10$, $\Delta y=0.10$, $h=10\text{m}$, the time step $\Delta t = 0.001$ sec at time $t = 100, 200, \dots, 1000$ hours and total steps 3600.

4. Results and discussion:

First of all, we tested the computational stability of this model by using CFL condition which is 0.22 less than 1 (see Section 2.8.3). The time integrations were performed for 1000 hours.

The following figures show the simulation of free surface elevation at time $t = 100, 200, \dots, 1000$ hours.





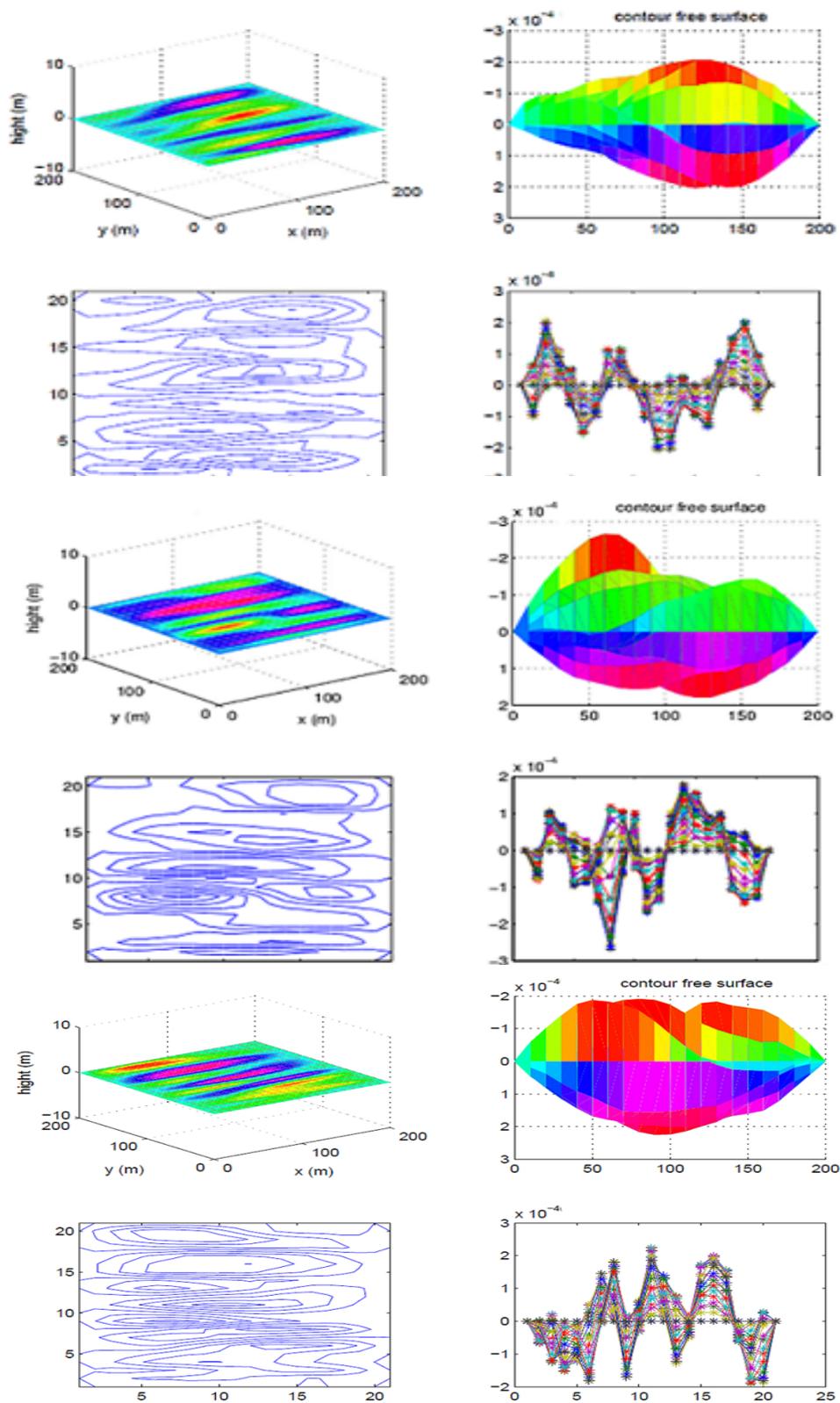


Figure 3-2: Simulation of free surface elevation at time $t = 100, 200, \dots, 1000$ hours.

The following figures show l_2 -RE of free surface elevation, comparison l_2 -RE of free surface elevation, u -velocity and v -velocity and approximate solution for free surface elevation for linear SWEs.

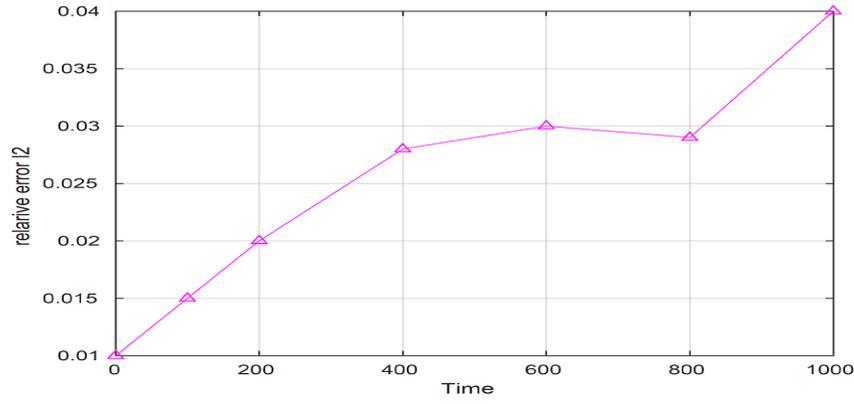


Figure 3-3: l_2 -RE of free surface elevation for linear SWEs at $t = 10, 20, \dots, 1000$ hours.

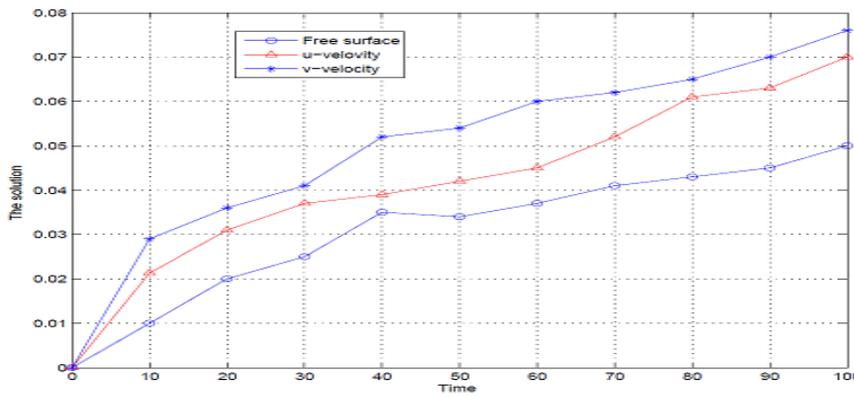


Figure 3-4: Comparison l_2 -RE of free surface elevation, u -velocity and v -velocity for linear SWEs at $t = 10, 20, \dots, 100$ days.

3.2 Some Applications of 2D Depth-Averaged Non-Linear SWEs

Consider the system given by equations (2.32)-(2.34) in Section 2.5 with neglect of viscosity, wind stresses, and the Coriolis force f terms, by dividing H into all terms of the x -momentum and y -momentum equations, we get the system as follows [69]:

$$\frac{\partial \eta}{\partial t} + \frac{\partial(Hu)}{\partial x} + \frac{\partial(Hv)}{\partial y} = 0 \quad (3.1)$$

$$\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x} + \frac{\partial(uv)}{\partial y} + g \frac{\partial \eta}{\partial x} + \frac{\tau_x}{H\rho} = 0 \quad (3.2)$$

$$\frac{\partial v}{\partial t} + \frac{\partial(uv)}{\partial x} + \frac{\partial v^2}{\partial y} + g \frac{\partial \eta}{\partial y} + \frac{\tau_y}{H\rho} = 0 \quad (3.3)$$

The bottom friction comes from Manning's formula, where n is roughness coefficient. In this simulation, n takes 0.013. (see Section 1.4.6).

Initial condition:

Initially, it is assumed that the motion in the domain is observed from an initial state of rest, so $u(x, y, 0)=v(x, y, 0)=0$ and at the beginning of a simulation start, this initial water surface obtained from a given data file, as well as the value fluxes, are zero on all grids. A full description of initial condition, boundary condition and model configuration can be found in ([114], [121]) (For more detailed see Appendix).

Example 1: Case 1: 2D linear shallow water equations

In this example, we use a system of equations (3.1)-(3.3) for linear case in a rectangular domain $\Omega = [0, L_x] \times [0, L_y]$. The computational domain is discretized by a grid whose size is regular.

1. Boundary conditions:

Here, radiation open boundary conditions (moving boundary conditions) were implemented at the boundaries with CFL condition 0.6 (see Appendix).

2. Numerical simulations, results and discussion:

The numerical values of the parameters are chosen as follows: $nx=ny =120$, $\Delta x=\Delta y =9$, time step $t = 2.5e^{-2}sec$, water depth 10 m and the total steps 125.

The first figure compares the approximate solution for free surface elevation, u -velocity and v -velocity and the second figure shows $l2-RE$ of free surface elevation, u -velocity and v -velocity for 2D linear SWEs at different time $t = 1000, 2000, \dots, 5000$ sec.

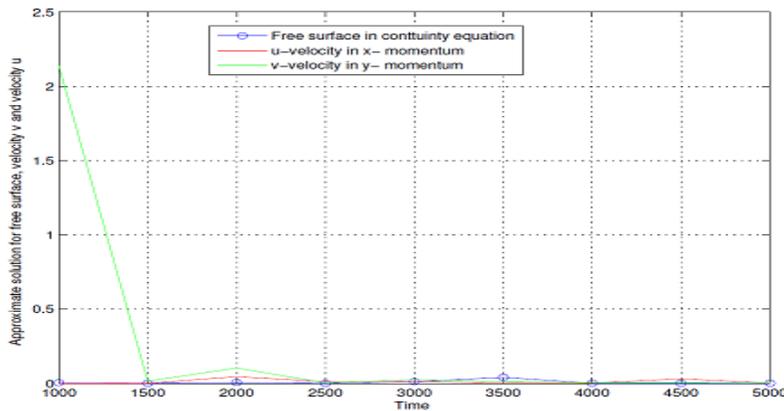


Figure 3-5: Comparison the approximate solution for free surface elevation, u -velocity and v -velocity for linear SWEs.

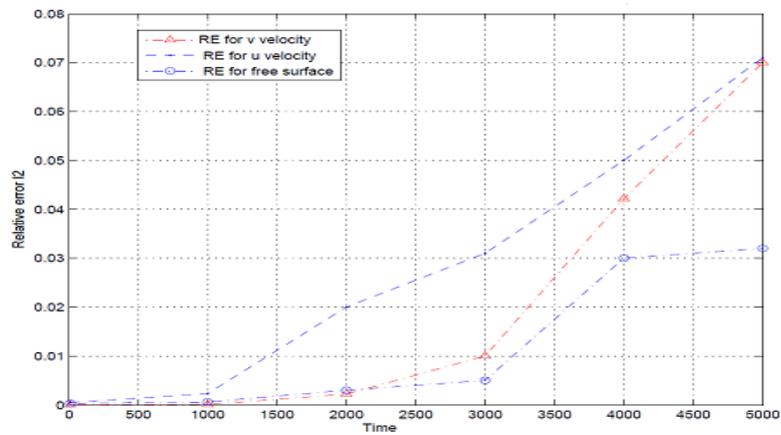


Figure 3-6: Comparison l_2 -RE of free surface elevation, u -velocity and v -velocity for linear SWEs.

Example 2: Case 2: 2D nonlinear shallow water equations

In this example, we introduce some applications for 2D depth-averaged nonlinear SWEs given in equations (3.1)-(3.3) which are approximately second order in space and time.

1. Boundary conditions:

Here, radiation open boundary conditions (moving boundary conditions) were implemented at the boundaries with CFL condition 0.7 (see Appendix).

2. Numerical simulations, results and discussion:

The numerical values of the parameters are chosen as follows: $nx = ny = 120$, $\Delta x = \Delta y = 9$, time step $t = 2.5e^{-2}sec$ and the water depth 10m.

The first figure compares the approximate solution for free surface elevation, u -velocity and v -velocity and the second figure shows l_2 -RE of free surface elevation, u -velocity and v -velocity at different time $t = 1000, 2000, \dots, 5000$ sec.

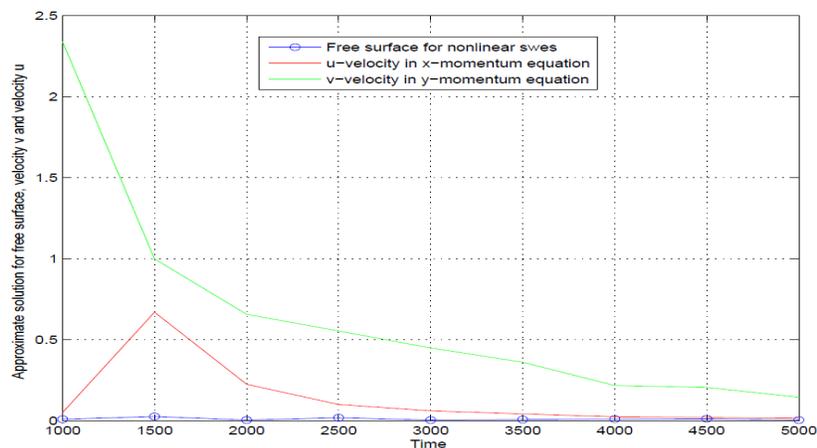


Figure 3-7: Comparison the approximate solution for free surface elevation, u -velocity and v -velocity for nonlinear SWEs.

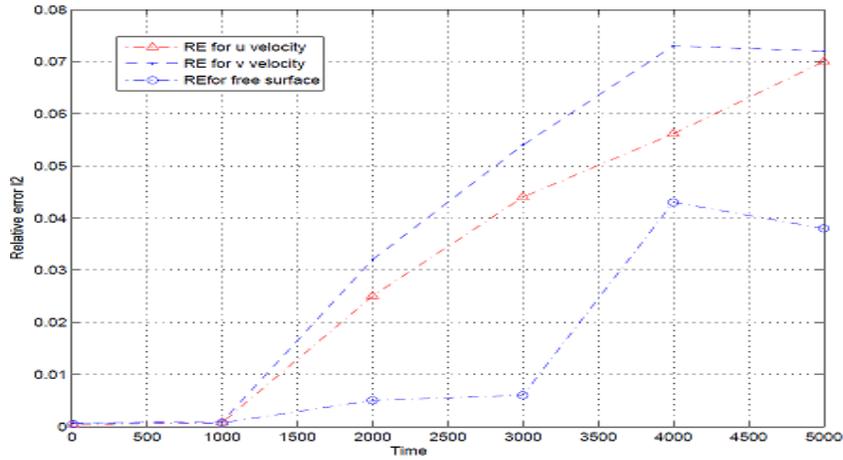


Figure 3-8: Comparison l_2 -RE of free surface elevation, u -velocity and v -velocity for nonlinear SWEs.

3.3 Some Applications of 2DSWEs Using Gaussian Level Initial Condition

In this section, we use 2DNSWEs given in equations (2.1)-(2.3), when the bottom stress, wind stress, Coriolis force, and viscosity horizontal in x -axis and y -axis are equals zero in the domain $\Omega = [0, L_x] \times [0, L_y]$ (see [1, 103]).

Initial condition:

Initially the Gaussian initial elevation is given by:

$$\eta(x, y) = h_0 * \exp(-((x - x_0)^2 + (y - y_0)^2)/2\sigma^2)$$

Where h_0 is the Gaussian bell height at the center, $\sigma = \sigma_x = \sigma_y$ is the Gaussian bell width along both x and y -axis and (x_0, y_0) are the coordinate of the Gaussian bell center.

Gaussian bump with its center at the domain origin (center). It has a height of 1cm and a width of 60m along both x and y -axis.

Example 1: Gaussian Bump Description

Numerical parameters:

The computational domain is discretized by a grid whose size is regular. Numerical values of the parameters are chosen as follows: number of grid 30×30 , $\Delta x = 20m$, $\Delta y = 20m$ (grid length) and the time step $\Delta t = 0.1s$, the simulation duration $t = 100s$, water depth $h = 10m$, the initial characteristic speed of the flow $U_0 = 4.9e-03$, energy = $2.77e+03$, horizontal length = 600m, and vertical length = 10m.

Boundary conditions:

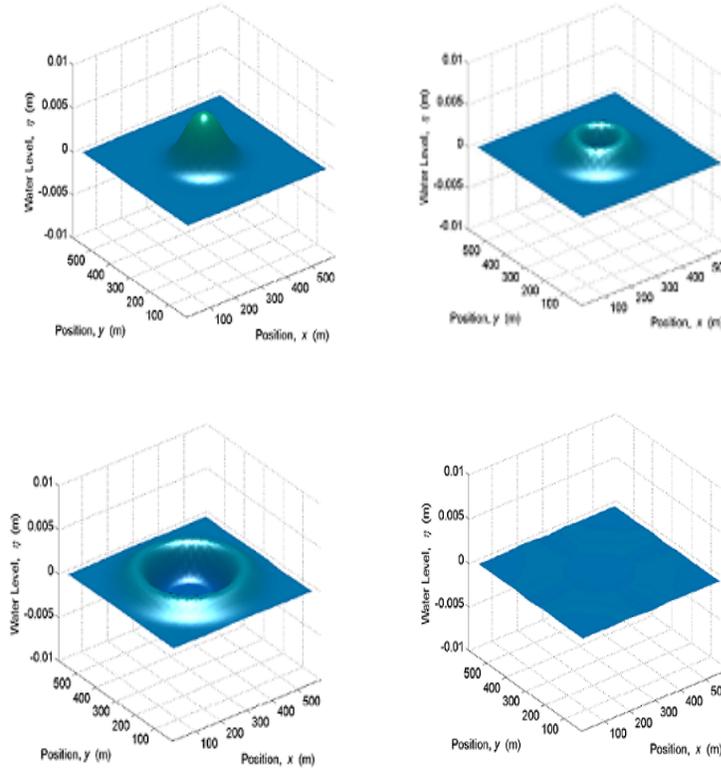
Here, Boundaries: level is set to Sommerfeld, normal is set to fletcher boundary conditions and

tangent is set to Dirichlet boundary conditions with CFL condition 0.198.

Results and discussion:

Gaussian bump is released, the wave front is radiated at the boundaries with radiative scheme for the water elevation, the tangential velocity and a flather radiative scheme for the normal velocity.

The following figures represent sequence of snapshots of water level at different times.



Example 2: Closed step Bump Description

Numerical parameters:

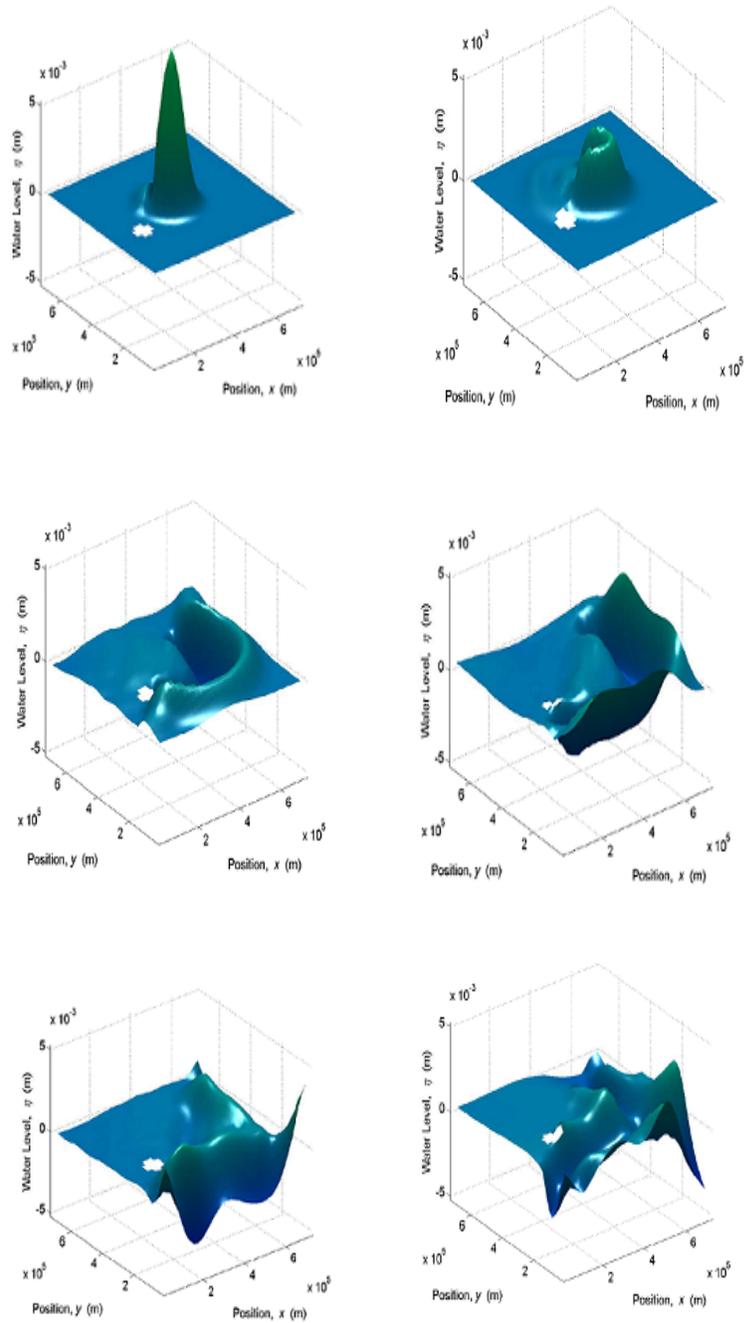
The computational domain is discretized by a grid whose size is regular. Numerical values of the parameters are chosen as follows: number of grids 37×37 , $\Delta x = 20000m$, $\Delta y = 20000m$, the time step $\Delta t = 500s$, the simulation duration $t = 100000s$, water depth $h=10m$, island $40000m \times 80000m$, radius $25000m$, $U_0=4.9e-03$, energy= $286e+04$, horizontal length $7.4e+05$, and the vertical length $10m$ and total steps 200.

Boundary conditions:

Closed boundary conditions with CFL condition 0.99. Here, Boundaries: normal is set to Dirichlet boundary conditions and tangent is set to Dirichlet boundary conditions.

Results and discussion:

The following figures represent sequence of snapshots of water level at different times.



3.4 Summary and Conclusion for this Part

Different concepts were introduced for 2D shallow water equations. We identified the mathematical problem and we mentioned the special cases of the 2DSWEs.

Three ways to get the full derivation of 2D shallow water models were presented in the simplest way to satisfy the curiosity of fresh physical oceanographers.

There are different types of numerical methods. The advantage of an explicit center finite difference method and leapfrog with Robert-Asselin filter in naturally conserving mass and the structured mesh is the ability in dealing with the regular boundaries of a two-dimensional.

An explicit staggered scheme to simulate 2DSWEs has been introduced. This scheme is simple, accurate and straight forward both for this model. Good results were obtained through comparison the approximate solutions and l_2 -relative error norm of free surface elevation, u -velocity, and v -velocity.

The performance of a new technique for 2DSWEs have been evaluated under Dirichlet, reflexive boundary conditions and moving boundary conditions. Some examples for 2DSWEs were applied using Gaussian level initial condition.

Several examples have been tested of the tsunami model and the results were obtained with high accuracy by using l_2 - RE . The results demonstrated the applicability and benefits of this technique.

Coupling for Two-Way Nesting Grids
Mathematical Framework and Applications
for Shallow Water Models

Chapter 4

The Configuration of a Nested Grid for Shallow Water Models

Chapter 4

The Configuration of a Nested Grid For Shallow Water Models

The results presented in this chapter (Section 4.7) are the subject of an article [3]

A new two-way interaction technique for coupling nested grids of 2D depth-averaged non-linear SWEs for structured grids is designed. Studying the accuracy of two-way nesting performance techniques between a coarse grid and a fine grid are suggested using an explicit finite difference method on Arakawa C-grid in space and leapfrog method with Robert-Asselin filter in time. This model consists of a higher-resolution (fine grid) nested 3:1 or 5:1 in a low-resolution (coarse grid) on which covers the entire domain. The formulation of the nesting grids algorithms allows flexibility for any ratio of grid sizes between two sub-regions. Assuming the water depth is constant along the boundary condition for the fine grid model to ensure the conservation of transfer.

To verify the ability and benefits of nested grid models, several numerical examples are applied to show and check the proposed technique can works efficiently over different periods of time and the results indicate good nesting performance technique. The two-way interaction systems depend on the type of interpolation, the location of dynamical interface, conservation properties (flux of mass and momentum equations), and type of update.

In this chapter, we will **focus on several major aspects** including configuration of nested grid for 2DSWEs, and design of interpolation/restriction operators. We begin with a literature review of techniques used to try to increase efficiency and accuracy of 2DSWEs. Some new algorithms are established to implement two-way interaction technique for this model which are given in Section 4.4. Intergrid transfer operators (interpolations and updates) are given. Then, looks into the optimum feedback conditions and interpolation techniques to maximize the feedback of the information. Four choices of the update schemes for free surface elevation and velocities on Arakawa C-grid are applied which are listed in Section 4.5.

Highlights

- Two grids are nested; one is a coarse grid and the other is a fine grid.
- The interaction between the two grids is two-directional.
- Description of a new two-way interactive nesting technique for 2DSWEs.
- Build some new algorithms to implement two-way interactive technique for ocean models.
- Some types of interpolations and updates are given.
- Four choices of the update scheme for the free surface and velocities are applied.

Contents

4.1	Literature Review	123
4.1.1	Classification of grids	123
4.1.2	Mesh structure	125
4.1.3	Separation of dynamic and feedback interface	127
4.1.4	One-way nested models	127
4.1.5	Two-way nested models	128
4.1.6	General formulation of the nested models	130
4.2	Other nesting considerations	131
4.2.1	Multiple nesting	132
4.2.2	Some important notes	132
4.2.3	Updating interpolation/ Feedback	132
4.3	Description of the numerical model	136
4.3.1	Spatial and temporal refinements	137
4.4	Methodology	138
4.4.1	Two-way nesting grids Algorithm (Horizontal embedding procedure)	138
4.4.2	General case of computational algorithms when the space refinement factor is 1:3 and temporal refinement factor is 1:2	146
4.4.3	Transferring (Interpolation) algorithm	147
4.5	Free surface and velocity updates	150
4.6	Algorithm for coupling the coarse and fine grids using open BC	151
4.6.1	Coupling Procedures	155
4.7	Summary and Conclusion	156

4.1 Literature Review

The problem of computational cost is a common problem in coastal modeling. The major determinant of computational cost is the spatial resolution or temporal resolution, whether water modeling deep in the ocean or shallow coastal seas. In most numerical models, the temporal resolution is directly linked to the spatial resolution and a higher spatial resolution thus impose a higher temporal resolution which adds more to the computational cost with spatial domain for great ranges and times level of weeks to years, the computational cost of numerical modeling can become very high.

For coastal models, there is an additional problem related to spatial resolution, the position of open water boundaries. Such boundaries must be located so that their conditions will not adversely affect in the area of interest.

The most popular numerical manners used to solve spatial resolution or temporal resolution problems are nested grid techniques. Nested grids allow one to increase spatial or temporal resolutions or both in a sub-region of the model domain without incurring the computational expenses for high-resolution over the entire domain. They can significantly reduce the number of computational grid points and thus the computational cost.

A feature of traditional nested grid techniques, the nested grids are static, that means locations on nested grids are steady for a simulation model. Further optimization of the computational cost can be achieved if the nested grids are able to move with the advantage of interest thus reduces the area over which high resolution is required. This technique is known as adaptive meshing. Although not as common as the traditional nesting techniques, but it has been successfully implemented in a number of cases [13, 55].

In this section, recall some basic concepts about one-way and two-way interaction techniques.

4.1.1 Classification of Grids

1. Structured grid

Structured grids are fixed by regular communication. The possible element choices are quadrilateral in 2D and hexahedra in 3D. This model is highly space efficiently, i.e., Since the neighborhood relationships are defined through the storage arrangement. Some other advantages of a structured grid over unstructured grid are better convergence and higher accuracy [21, 58].

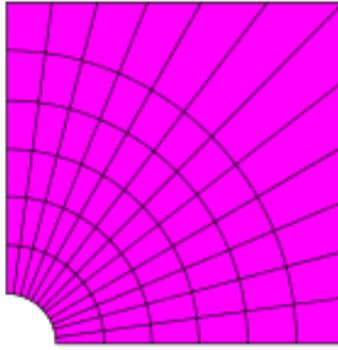


Figure 4-1: Shows Structured grid.

2. Unstructured grids

An unstructured grid is identified by irregular communication. It cannot easily be expressed as a two-dimensional or three-dimensional array in computer memory. This allows for any possible element that a solver might be able to use. Compared to structured meshes, this model can be highly space inefficient since it calls for the explicit storage of neighborhood relationships. These grids typically employ triangles in 2D and tetrahedral in 3D [86].

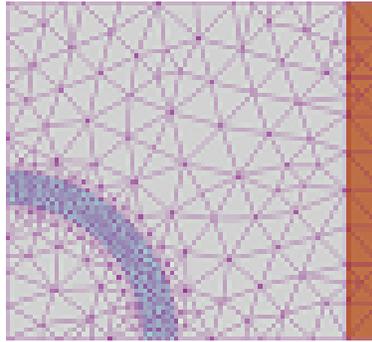


Figure 4-2: Shows unstructured grid.

3. Hybrid grids

A hybrid grid contains a mixture of structured parts and unstructured parts. It integrates the structured meshes and the unstructured meshes in an efficient manner. Those portions of the geometry that are regular can have structured grids and those that are complex can have unstructured grids. These grids can be non-conformal, which means that grid lines don't need to match at block boundaries.

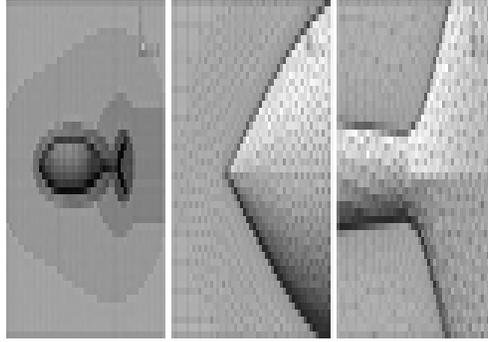


Figure 4-3: Shows hybrid grid.

4.1.2 Mesh structure

The structure of a two-way nested mesh is composed of two main types, the first is an overlapping method and the second is a seamlessly embedded method that consists of two types: adjacent and separate. The adjacent seamlessly embedded meshes were applied in the early generation of two-way nested domains [14, 37, 62, 84, 93, 100].

The procedure involves the time integral for the coarse and nested domain to progress simultaneously. Boundary data from the coarse domain is interpolated on to the interface between the nested and coarse domains to allow the forcing of the nested domain. The feedback/updating data from the nested domain is transmitted through the same interface which the boundary data is interpolated on [48].

The separated embedded model involves separating the nested and coarse domain by a mesh. This mesh also known as a window frame is shown in Figure 4-4 as domain 2. The frame consists of an overlapping of the two different resolution domains. The interior boundary of the coarse domain is the mesh interface (feedback) and the fine domains foreign boundary where the boundary conditions for the domain are generated is the dynamic interface (input). The separation of the interfaces allows for only internally generated nested domain values being used in the feedback calculation. The first step involves the simulation of the coarse domain and the mesh/window frame domain (domains 1 and 2) in Figure 4-4. The second step involves the use of the data generated at the dynamic interface as boundary conditions for the simulation over domain 2 and the nested grid in domain 3. Time interpolation is performed at the dynamic interface and spatial interpolation is performed at the mesh interface. This method allows two-way interaction due to domain 1 being influenced by domain 2 and the coarse domain influences the nested domain by providing the boundary conditions for the nested domain simulation. This method is a very common method in nested modeling and has been used in a large number of nesting schemes in both meteorology and hydrology [52, 98, 122].

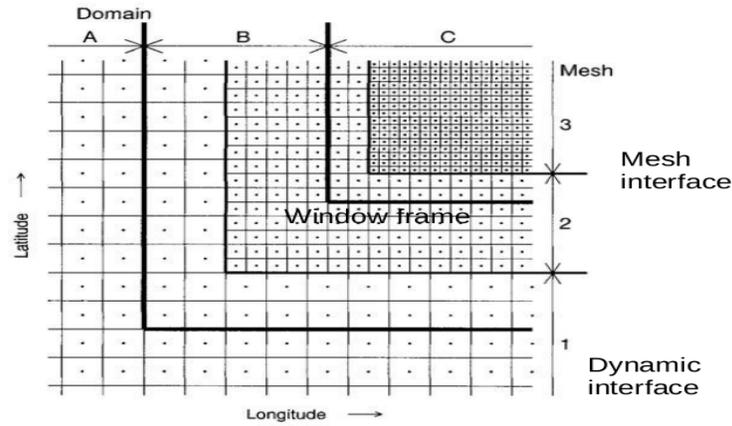


Figure 4-4: Embedded nested domain.

The second type of two-way nested modelling procedure is an overlapping method that involves the extension of the coarse domain over the full nested domain used by [67]. The overlapping method has been used in a number of applications [41, 56, 109].

The style includes the coarse domain integrating for one-time step and boundary data from the coarse domain are interpolated onto the dynamic interface (see Figure 4-4) between the nested and coarse domains. The nested domain is then integrated using the boundary data until the time step is equal to the time step of the coarse domain. Conversion of the high-resolution nested domain data occurs at points where the coarse domain grid points coincide with the fine domain grid points and are updated using some interpolation method of the enclosed nested grid data. Figure 4-5 shows an illustration of the grid configuration for a nested ratio of 3:1.

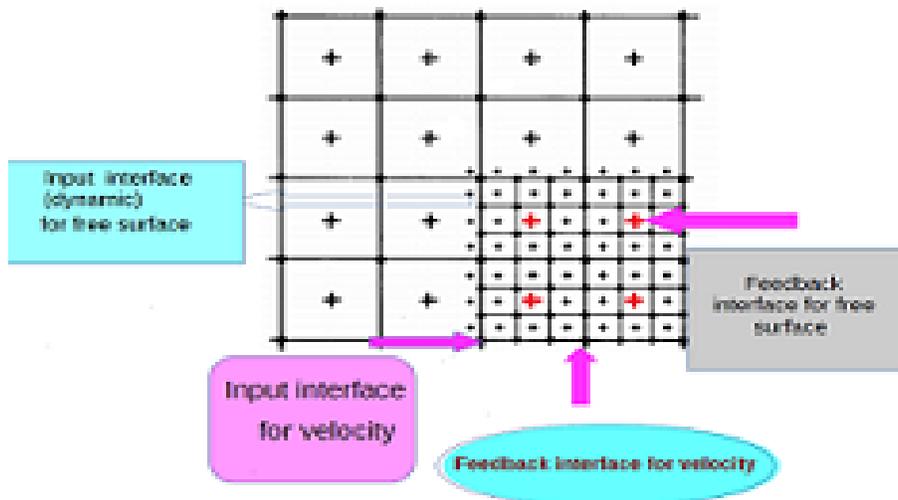


Figure 4-5: Overlapping grid configuration.

4.1.3 Separation of dynamic and feedback interface

The dynamic interface shows the fine mesh boundary, where the solution of a fine grid is forcible by the coarse solution; the feedback interface is the outer limit of the area where the coarse solution is updated by the fine solution. There are several reasons for separating dynamic and feedback interfaces. The separation can be composed of two coarse grid cells. In these cases, the coarse grid points used in the interpolations are not updated. Another reason for using a mesh separation between dynamic and feedback interfaces is that if noise is produced, it will be larger near the dynamic interface so that it is safer not to use the fine grid values near the dynamic interfaces. Several authors have proposed to separate the feedback interface from the dynamic interface [48, 52, 98].

4.1.4 One-way nested models

One-way nesting is applied to models to generate a higher level of accuracy in the field of interest. The nested domain is located so it is embedding in the coarse domain one, which involves the entire domain. Figure 4-6 shows a sample grid configuration for a one-way nested model with 3:1 spatial nesting ratio.

This model runs almost completely separately to calculate the domain of interest in both levels of resolution. The interaction between the domains occurs in the area of the nested domain boundaries. The boundary conditions for the nested domain are generated from the coarse domains data that are interpolated in space and time. Initially, the coarse domain is integrated in time and the data required for the nested domains boundary conditions are modified to collaborate with the fine domains resolution. The fine domain solution is then performed using the boundary data with the fine domain time step.

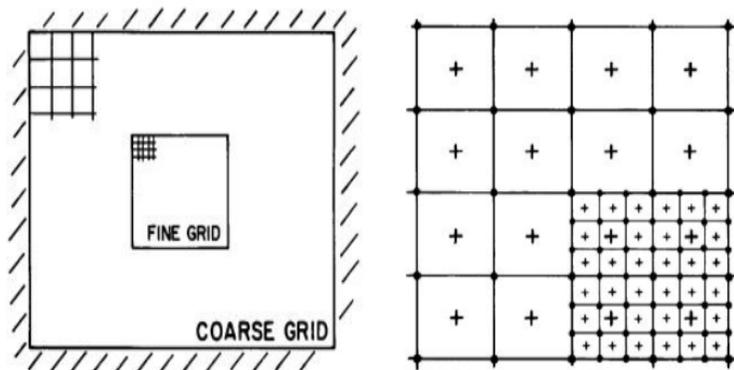


Figure 4-6: Adaptive mesh refinement.

Ginis [102] was the first to apply one-way nesting technique in hydrodynamics models and it has since been applied to many studies of oceans and coastal waters. Debreu [50] developed and applied a one-way nested model of the central California upwelling system. The method involved the integration

of a nested grid into the Regional Oceanic Modelling System (ROMS). The model was applied to a domain that spanned the continental Pacific coast of the United States and nested an area that covers the central upwelling region of California around Monterey Bay. One-way nested grids have also been used by [70] to represent eddy fields in the Aegean and Levantine basins located in the Mediterranean Sea.

There are two main schemes to run the one-way nesting technique. The initial scheme involves a complete solution for the coarse and fine models with respect to the simulation time. In this style, the coarse model is fully run for the model simulation time. The data required for the boundary conditions are stored in a data file to be used for the fine domain model that is then fully run for the simulation time. This method is known as an uncoupled modeling procedure [13].

The coupled modelling procedure is more attractive than the uncoupled because it does not require big quantities of the store. In this method, the coarse model is run for one time step and data required for the nested domains boundary conditions are assigned, allowing the nested model to proceed to a time step equal to the coarse domains. A coarse domain continues to the next time step when the fine domain has been integrated.

A major importance in nesting techniques is the conservation of properties between the coarse and fine grid models and the treatment of fine grid interior noise generated as a result of incompatibilities between the two models [70]. The implementation of boundary conditions is therefore crucial and can add complexity and computational cost. The underlying assumption in one-way nested models is that the larger-scale motion determines the small-scale motion without feedback from the processes occurring within the nested region; thus properties need only be conserved in one direction. For this reason and the fact that the models may be run independently. Also, one-way nesting is both easier to implement and usually less computationally expensive than two-way nesting.

4.1.5 Two-way nested models

The coarse mesh and fine mesh in a two-way nested model, are necessarily dynamically connected each effect the other and neither can be run independently. The interaction in the direction of a coarse to fine is similar in manner to one-way nested systems. The coarse model is integrated in time and boundary for the fine model are interpolated in time and space from adjacent coarse mesh, thus the fine model is integrated in time using the interpolated boundary data. However, unlike one-way nested systems, once the fine model has been integrated a coarse grid is then updated using a fine grid value via some numerical procedure.

Spall and Holland [109] were among the first to use a two-way nesting to a model for oceanographic

applications. This applies to two kinds of test cases in artificial domains that are relevant to oceanic phenomena. Fox [56] adapted the Spall and Holland method to investigate problems that may appear when fronts and other oceanic features intersect the boundaries between the domains and also the performance of the model when topographical features are present.

Oey [90] used a two-way nested model in the Norwegian coast to simulate meanders and eddies in the coastal currents. The two-way nested model used by [102] to model the tropical Pacific ocean employed a two-way interactive method for the application in hurricane forecasting modeling. The interaction between the two domains occurs in an area called the dynamic interface located near the nested domain boundary. Results presented showed that the interaction at the dynamic interface improved the conservation of properties between the two domains.

Dynamical interaction between coarse and fine domains can be achieved in various ways. The most common technique is to transfer information from the fine to the coarse mesh and vice versa in the zone where the two meshes overlap.

Conservation of properties is one of the most important aspects of nesting techniques between grids. A two-way nesting technique must ensure conservation of properties when passing variables from the coarse grid to the fine through the boundary conditions, also when the coarse grid variables are updated from the fine. Ginis [102] state that the overlapping grid method used by Spall and Holland [109] and Oey and Chen [90] do not necessarily conserve fluxes of mass, heat and momentum at the interfaces between coarse and fine meshes. In the scheme developed by [102] the interaction at the dynamical interface is specified as a flux condition; this enables improved conservation of mass, momentum and heat.

The treatment of noise generated in the fine grid was also mentioned as a problem in relation to one-way nesting. The same problem applies to two-way nesting techniques but the solution is more complex. Not only must the technique minimize the disturbances in fine grid values which can occur near the mesh interface as a result of grid incompatibility but it must also prevent those disturbances from passing out of the fine grid and into the coarse grid. A disturbance propagating from a fine grid to a coarse grid may undergo false reflection back to the fine grid or aliasing as it enters the coarse grid [102]. These interface-generated problems may lead to numerical instabilities that can seriously affect the results over the entire domain.

Two-way nesting should give more accurate simulations than one-way nesting. Barth [41] compare results from a two-way nested/one-way nested and coarse model of the Ligurian Sea. The two-way nested model was found to better represent currents within the Sea than both the coarse and one-way nested models.

4.1.6 General formulation of the nested models

We consider the general case of a high-resolution model covering the local domain ω embedded in a coarser resolution model covering the larger domain Ω with clear notations the local high-resolution grid and the global coarse resolution grid are denoted respectively as ω_h and Ω_H . The corresponding state vectors are denoted respectively as x_h and x_H . We also denote as ω_H , the part of the grid Ω_H corresponding to the local domain ω [37].

In the case of one-way interaction, the coarse grid model provides boundary conditions to the high-resolution model using an interpolation operator I_H^h . Semi-discretized equations of the nested system can be written as follows:

Domain Ω_H

$$\frac{\partial x_H}{\partial t} = F(x_H) \quad \text{on} \quad \Omega_H \times [0, T]$$

$$x_H(t = 0) = x_H^0$$

Domain ω_h

$$\frac{\partial x_h}{\partial t} = F(x_h, x_{\partial\omega}) \quad \text{on} \quad \omega_h \times [0, T]$$

$$x_h(t = 0) = x_h^0$$

$$x_{\partial\omega} = I_H^h(x_H) \quad \text{on} \quad \partial\omega_h \times [0, T]$$

where $x_{\partial\omega}$ represents the information coming from the coarse grid onto $\partial\omega_h$, the boundary of the fine grid. The one-way interaction is said to be passive since there is no retraction from the local model onto the global model. From a practical point of view this also means that both models do not have to be run simultaneously (the global model can be run first and its solution can then be used offline by the local model).

In the case of two-way interactions a feedback term from the fine grid onto the coarse grid is added. The coarse solution is updated locally (in ω_H° , the interior of ω_H) by the high-resolution solution using a restriction factor G_h^H . Semi-discretized equations of the nested system can be written as follows:

Domain Ω_H

$$\frac{\partial x_H}{\partial t} = F(x_H, x_\omega) \quad \text{on} \quad \Omega_H \times [0, T]$$

$$x_H(t = 0) = x_H^0$$

$$x_\omega = G_h^H(x_h) \quad \text{on} \quad \omega_H^\circ \times [0, T]$$

Domain Ω_H Domain ω_H

$$\frac{\partial x_h}{\partial t} = F(x_h, x_{\partial\omega}) \quad \text{on} \quad \omega_h \times [0, T]$$

$$x_h(t = 0) = x_h^0$$

$$x_{\partial\omega} = I_H^h(x_H) \quad \text{on} \quad \partial\omega_h \times [0, T]$$

where x_ω represents the information coming from the fine grid onto the coarse grid in ω_H° . The two-way interactions are said to be active. In that case both models should be run at one time since they permanently exchange information.

After discretization, the problems have to be integrated in time in a specific order. The model is first integrated on the coarse grid Ω_H and then on the high-resolution ω_h grid with boundary conditions given by a spatial and temporal interpolation of the coarse values. Finally, a feedback can be applied.

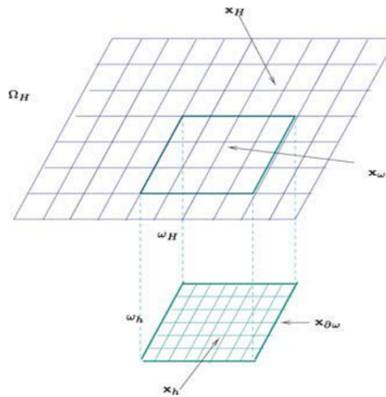


Figure 4-7: Notations used in the definitions of the nested models.

4.2 Other Nesting Considerations

4.2.1 Multiple nesting

Most of models which used the space refinement ratio for one-way $\frac{\Delta x_c}{\Delta x_f} \leq 4$ and two-way nesting grids have used the space refinement ratio $\frac{\Delta x_c}{\Delta x_f} \leq 7$. Many nesting experiments have shown that acceptable results are obtained for spatial nesting refinement ratio for 3:1 and 5:1 [109]. The vast majority of both one-way and two-way nested models employ 3:1 nesting refinement ratio. Both [41, 109] report that the use of higher nesting ratios results leads to significant degradation of model performance. For example, Spall and Holland [109] found that model accuracy began to deteriorate at 7:1 nesting ratio. Koch [106] give two reasons for this. Firstly, that higher grid refinement ratios require too many fine grid points to adequately resolve the coarse grid waves. Secondly, that the incompatibilities between the grids are so large that wave reflection and noise generation become excessive. In order to achieve a

high resolution at nesting ratios in excess of 5:1, multiple nested models have been developed. These models allow to minimize nested domains to multiple levels of nesting [99].

4.2.2 Some important notes

1. In fact, the most investigators have found the conservation condition unnecessary in hydrostatic models in order to obtain a smooth solution at the interface [106]. Others, have found do not necessarily conserve fluxes of mass, heat and momentum at the interfaces between coarse and fine meshes [90].
2. Choosing odd mesh refinement ratio simplifies grids interactions. In the case, a coarse grid point always has one underlying high-resolution point while using an even mesh refinement ratio that means a coarse grid point does not have a corresponding point on the fine grid.
3. In the separated embedded model, time interpolation is performed at the dynamic interface and spatial interpolation is performed at the mesh interface.
4. There are five methods of noise control have been used in two-way nested grids models: smoothing operators, enhanced explicit diffusion, interface conditions modified to remove over specification, damping time-integration scheme and mesh separation scheme.
5. There are many methods of interpolation for the fine grid interface conditions (linear, cubic Lagrange, bilinear, cubic-spline, equivalently) and several different filters for the coarse grid interface conditions (full-weight, average, Shuman filtering, Shapiro filtering,...etc).
6. There are two methods to achieve the conservation, the flux correction method, which applied when the system with no time refinement and Kurihara (box) methods. In Kurihara method which applied in a grid with the variables located in cell's center.

4.2.3 Updating interpolation/feedback

Interpolation techniques are wanted in order to efficient data transfer with data is transferred between various domains of spatial and temporal resolution. There are two prime goals for an interpolation scheme to be optimum: (1) to maximize the information that is transferred. (2) to reduce the generation of noise.

Techniques of interpolation techniques used in the transfer of information from the coarse domain to the nested domain are usually of a polynomial form or a linear/bilinear form. Problems can arise with the use of polynomial techniques in areas of sharp gradients due to the formation of excess oscillation of the interpolation variables. Therefore, linear interpolation is more widely used for both spatial and temporal interpolation [70, 99]. Based on studies which conclude that zeroth-order interpolation may

create large phase errors, quadratic interpolation may create overshooting and they suggest the use of advection equivalent interpolation schemes.

There are four major updating interpolation steps for the transfer of information from the fine domain into the coarse domain (1) Direct copy (2) Basic averaging procedure (3) Shapiro and (4) Fully weighted averaging procedure [52].

1. Direct copy is the most sharp interpolation technique with only the nested grid point that lies directly in the domain of the coarse grid point being used in the procedure.

$$\phi_{I,J}^c = \phi_{i,j}^n$$

where $\phi_{I,J}^c$ represents the coarse grid point and $\phi_{i,j}^n$ represents the nested grid point that overlays the center of the coarse grid cell, i,j representing the grid point locations in relation to the x -direction and y -direction in the nested domain and I,J representing the coarse domain grid point locations in relation to x -direction and y -direction.

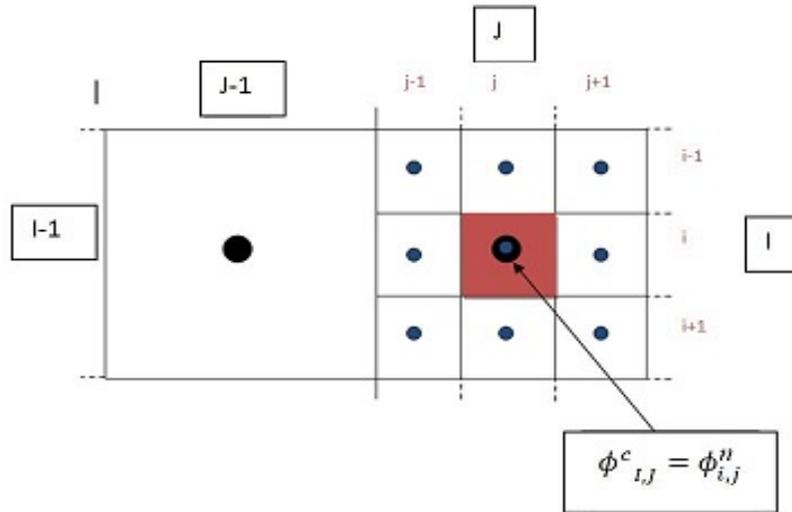


Figure 4-8: Copy interpolation scheme.

2. The average procedure. All fine grid points that are enclosed in the coarse cell [25].

The following equation shows the formation of the average scheme for a mesh refinement factor of 3:1.

$$\phi_{I,J}^c = \frac{1}{9} (\phi_{i-1,j-1}^n + \phi_{i-1,j}^n + \phi_{i-1,j+1}^n + \phi_{i,j-1}^n + \phi_{i,j}^n + \phi_{i,j+1}^n + \phi_{i+1,j-1}^n + \phi_{i+1,j}^n + \phi_{i+1,j+1}^n)$$

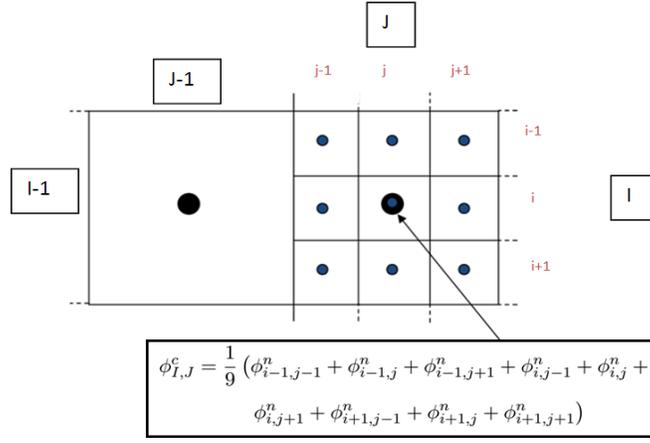


Figure 4-9: Average interpolation scheme.

And the following equation shows the formation of the average scheme for a mesh refinement factor 5:1.

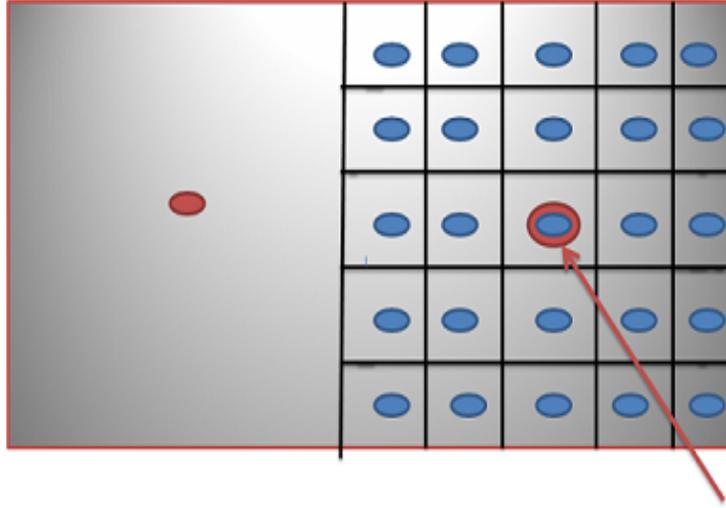


Figure 4-10: Average interpolation scheme.

$$\phi_{I,J}^c = \frac{1}{25} (\phi_{i-2,j-2}^n + \phi_{i-2,j-1}^n + \phi_{i-2,j}^n + \phi_{i-2,j+1}^n + \phi_{i-2,j+2}^n + \phi_{i-1,j-2}^n + \phi_{i-1,j-1}^n + \phi_{i-1,j}^n + \phi_{i-1,j+1}^n + \phi_{i-1,j+2}^n + \phi_{i,j-2}^n + \phi_{i,j-1}^n + \phi_{i,j}^n + \phi_{i,j+1}^n + \phi_{i,j+2}^n + \phi_{i+1,j-2}^n + \phi_{i+1,j-1}^n + \phi_{i+1,j}^n + \phi_{i+1,j+1}^n + \phi_{i+1,j+2}^n + \phi_{i+2,j-2}^n + \phi_{i+2,j-1}^n + \phi_{i+2,j}^n + \phi_{i+2,j+1}^n + \phi_{i+2,j+2}^n)$$

where $\phi_{I,J}^c$ representing the coarse point (black circle) that is being updated and $\phi_{i,j}^n$ being the fine grid values in the same cell (blue circle).

This scheme is based on the assumption that the fine grid variables over laying the one coarse grid cell have a uniform distribution of value.

3. The Shapiro interpolation scheme. Is based on the assumption that the nested grid point that lies in the central region of the coarse grid (blue square) is of equal importance to the sum of the other nested grid points enclosed in the coarse grid cell (red square) [52].

$$\phi_{I,J}^c = \frac{1}{16} (\phi_{i-1,j-1}^n + \phi_{i-1,j}^n + \phi_{i-1,j+1}^n + \phi_{i,j-1}^n + 8\phi_{i,j}^n + \phi_{i,j+1}^n + \phi_{i+1,j-1}^n + \phi_{i+1,j}^n + \phi_{i+1,j+1}^n)$$

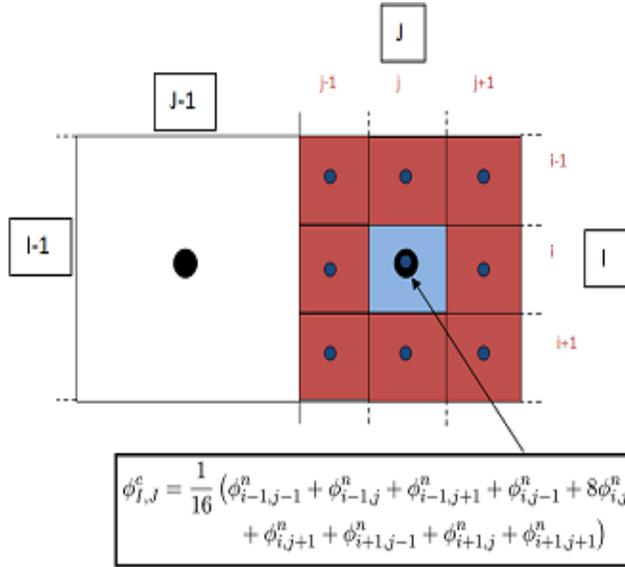


Figure 4-11: Shapiro interpolation scheme.

4. The final interpolation scheme. Is the full weighted averaging method and assumes that the interpolated value used for the updating procedure should be influenced mainly by nested grid points close to the center of the coarse grid point being updated (green+blue squares) and less by the more distant points (red square) [48]. The following equation presents the fully weighted scheme for a nesting ratio 3.

$$\phi_{I,J}^c = \frac{1}{20} (\phi_{i-1,j-1}^n + 2\phi_{i-1,j}^n + \phi_{i-1,j+1}^n + 2\phi_{i,j-1}^n + 8\phi_{i,j}^n + 2\phi_{i,j+1}^n + \phi_{i+1,j-1}^n + 2\phi_{i+1,j}^n + \phi_{i+1,j+1}^n)$$

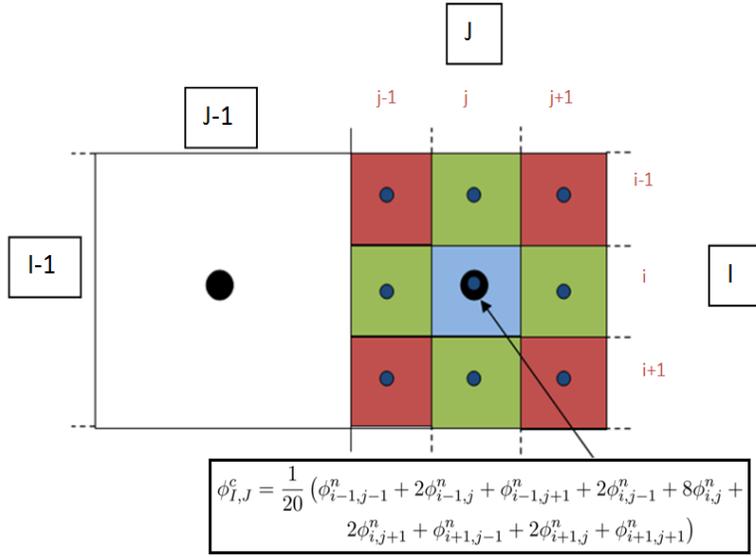


Figure 4-12: Full-weighting interpolation scheme.

And the following equation presents the fully weighted scheme for a nesting ratio 5.

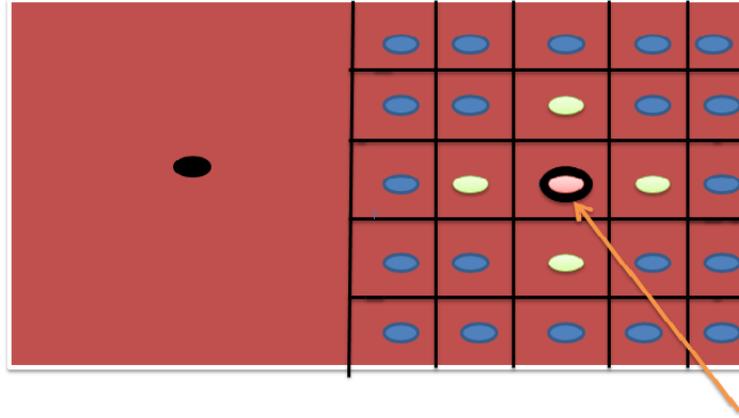


Figure 4-13: Full-weighting interpolation scheme.

$$\begin{aligned} \phi_{I,J}^c = \frac{1}{52} & (\phi_{i-2,j-2}^n + \phi_{i-2,j-1}^n + \phi_{i-2,j}^n + \phi_{i-2,j+1}^n + \phi_{i-2,j+2}^n + \phi_{i-1,j-2}^n + \phi_{i-1,j-1}^n + 4\phi_{i-1,j}^n + \\ & \phi_{i-1,j+1}^n + \phi_{i-1,j+2}^n + 2\phi_{i,j-1}^n + \phi_{i,j}^n + 4\phi_{i,j+1}^n + 16\phi_{i,j}^n + 4\phi_{i,j+1}^n + \phi_{i,j+2}^n + \phi_{i+1,j-2}^n + \\ & \phi_{i+1,j-1}^n + 4\phi_{i+1,j}^n + \phi_{i+1,j+1}^n + \phi_{i+1,j+2}^n + \phi_{i+2,j-2}^n + \phi_{i+2,j-1}^n + \phi_{i+2,j}^n + \phi_{i+2,j+1}^n + \phi_{i+2,j+2}^n) \end{aligned}$$

4.3 Description of the Numerical Model

Configuration of nested grid for 2D shallow water equations

As we know, through the waves crash, their length will become much smaller than in the deep ocean. Finer grids will be necessary to resolve wave features in shallow water regions. There are several reasons for applying multiple nested grid model, when the water depth varies within the computational domain,

it might be desirable that different grid size and time step size be employed in various subregions so that the frequency dispersion is adequately represented. The nested grid configuration allows to obtain detailed information in the coastal region. Finer grids should be used only in specific regions of interests.

In this model, linear or nonlinear for 2D shallow water equations can be assigned to a specific sub-level region and any ratio of grid size between two sub-regions can be used, however, it should be an integer.

Here, we describe the manner for exchanging information between two nested grid of different grid sizes. As shown in Figure 4-12 below a smaller grid system is nested in a larger grid system with the ratio of 1:3. The arrows represent the velocity flows across each grid cell, while squares and points indicate the locations where the water surface is evaluated.

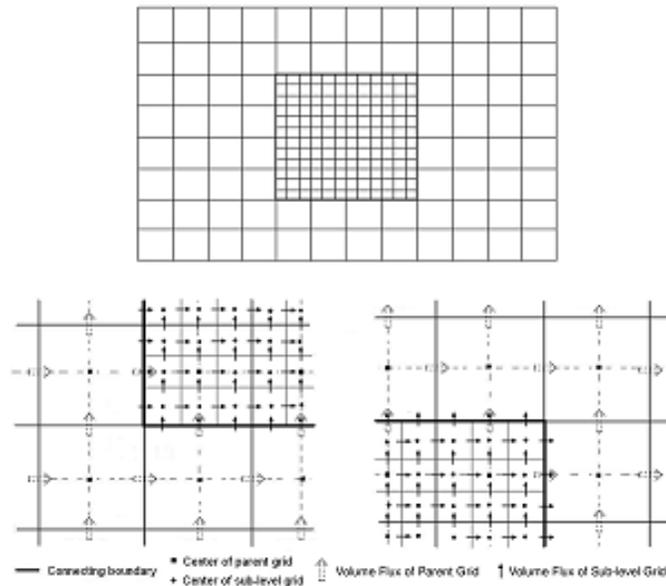


Figure 4-14: Detailed view of nested grid. Left panel: grid nesting at lower-left corner of sub-level grid region; Right panel: grid nesting at upper-right corner of sub-level grid region.

4.3.1 Spatial and temporal refinements

The basic configuration of the grid consists of three grid points which are the free surface elevation η and the horizontal velocity components u and v with the spatial refinement factor being an odd integer m , which is defined by:

$$m = \frac{\Delta x_c}{\Delta x_f}$$

where Δx_c and Δx_f are the coarse and fine grid lengths respectively. The temporal refinement factor is defined by the integer p given by:

$$p = \frac{\Delta t_c}{\Delta t_f}$$

where Δt_c and Δt_f are time step for the coarse and fine grids respectively

4.4 Methodology

Consider a model that consists of a higher-resolution (fine grid) nested 3:1 or 5:1 in a low-resolution (coarse grid) on which covers the entire domain. The information on the flux values (velocities) and the free surface elevation is exchanged on the boundaries between two nested grid regions. At each new time level, the flux values (velocities) or/and the free surface elevation on the boundary of a **finer grid** are obtained by linearly interpolating both spatially and temporally. At each next time level for a outer **coarse grid**, the free surface elevation or/and the flux values (velocities) on a coarser grid are updated by the average scheme (or fully-weighted scheme) both spatially and temporally.

In this section, some new algorithms for two-way interaction techniques for 2DNSWEs are established. Four choices of restriction operator for the free surface elevation η and velocities (u, v) on Arakawa C-grid are introduced.

4.4.1 Two-way nesting grids algorithm (Horizontal embedding procedure)

Let describe these procedure step by step. Suppose all values in the inner region with finer resolutions and the outer region with coarsest grid resolution, are known at time level $t = n\Delta t$ and we need to obtain the inner and the outer region values at the next time step $t = (n + 1)\Delta t$. Since the outer grid region and the inner grid region adopt different grid sizes, the time step sizes for each region are different due to the requirement of stability. At a certain level of time, velocity flows in both large and small grid models are determined from the momentum equations, with except of velocity flows for the smaller grid system along the boundaries between two grid regions. These data are determined by interpolating the adjoining velocity flows from the large grid model.

Assume that all model variables at time $t = n\Delta t$ are known and the time step of the inner region is one half the time for the outer region.

Case 1: The time step for the fine region is one half the time for the coarse region and the space refinement ratio is 1:3

-Step 1. Get the water surface elevation η at $t = (n + 1/2)\Delta t$ in the coarse grid by solving continuity equation.

-Step 2. To solve the continuity equation in the fine grid, we need to have the flux information along the connected boundary at $t = n\Delta t$. So the flux values (velocities) in the coarse grids at the connected boundary are linearly interpolated and then those interpolated the values are set to the fluxes in the inner grids at the boundary.

-Step 3. Solve the water surface elevation η at $t = (n + 1/4)\Delta t$ in a fine grid using continuity equation.

-Step 4. Solve the flux values at $t = (n + 1/2)\Delta t$ in a fine grid by using momentum equations.

-Step 5. Solve the water surface elevation η at $t = (n + 3/4)\Delta t$ in the fine grid using continuity equation. Here, we should have the flux information along the connected boundary at $t = (n + 1/2)\Delta t$. To get these information we can do the in the following way. First, since we already know the free surface elevation at $t = (n + 1/2)\Delta t$ and flux values at $t = n\Delta t$ in the outer region, we can get the flux velocities in the outer region along the connected boundary at $t = (n + 1)\Delta t$ by solving momentum equation. Second, these flux values at $t = (n + 1)\Delta t$ are linearly interpolated along the connected boundary. To get the value at $t = (n + 1/2)\Delta t$, outer flux values at $t = n\Delta t$ and $t = (n + 1)\Delta t$ are also linearly interpolated. Those spatially and timely interpolated flux values are assigned to the flux in the inner grid at the boundary.

-Step 6. To transfer the information from the inner grid region to the outer region, the free surface elevation in the inner grid region is spatially averaged over the grid size of the outer region. These average elevation values at $t = (n + 3/4)\Delta t$ are then time averaged with those at $t = (n + 1/4)\Delta t$ in the inner region and update the information at $t = (n + 1/2)\Delta t$ from a fine grid to a coarse grid using the full-weighting operator (or average operator) in both spatially and timely values in the inner grid region, update the elevation value at $t = (n + 1/2)\Delta t$ in the coarse grid.

-Step 7. Get the flux values at $t = (n + 1)\Delta t$ in the fine grid by solving momentum equations.

-Step 8. Get the flux values at $t = (n + 1)\Delta t$ in the coarse grid by solving momentum equations.

-Step 9. Transfer all the information at $t = (n + 1)\Delta t$ from a fine grid to a coarse grid using the full-weighting operator (or average operator) in both spatially and timely values in the inner grid region, update the information at $t = (n + 1)\Delta t$ in the coarse grid.

The following figure shows the detailed time marching scheme of nested grid setup.

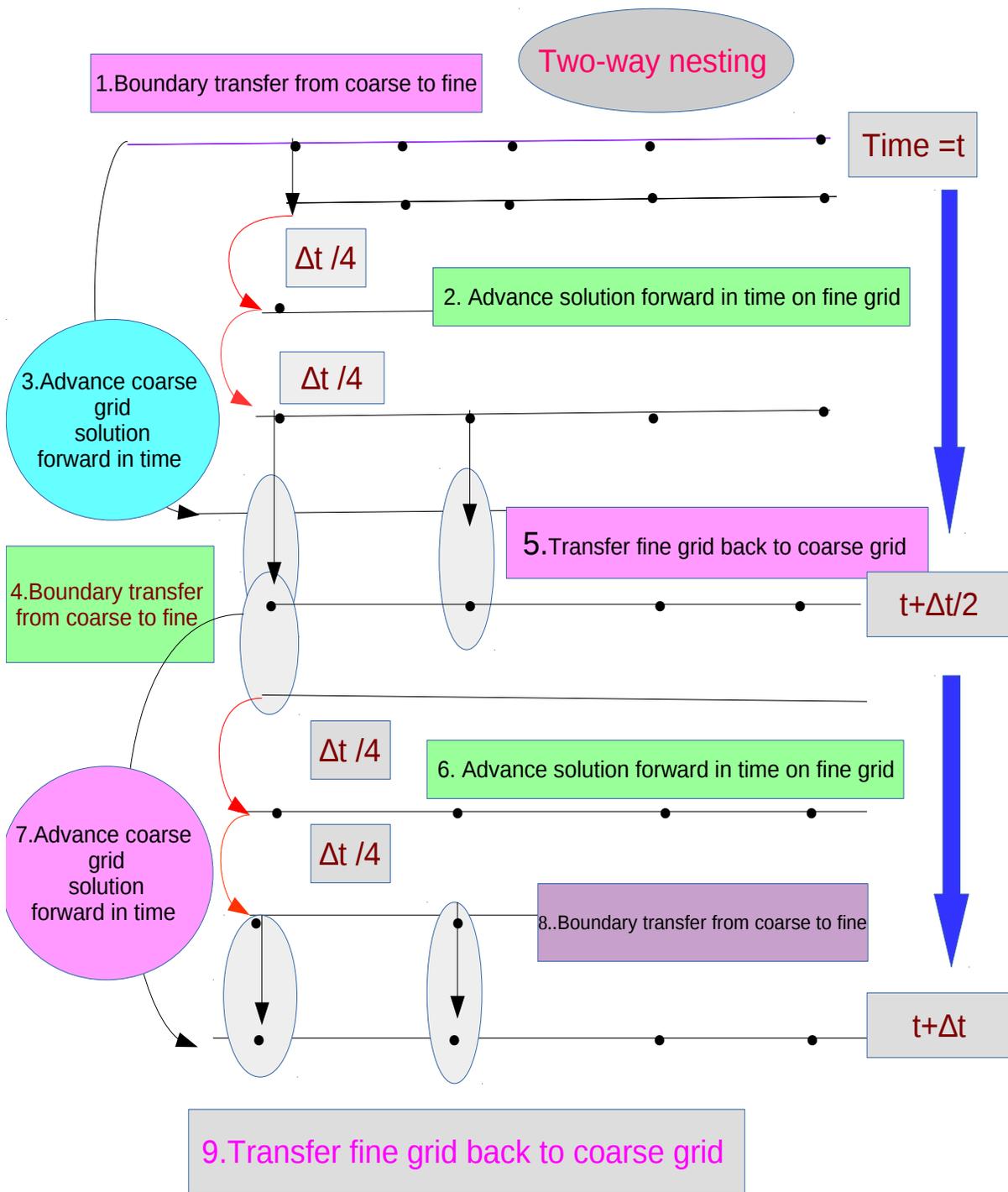


Figure 4-15: Simulation involving two-way nesting.

Note:

When the mesh refinement factor is 1:3 in 2-dimension, for each one coarse grid point there are 9 fine grid points corresponding to it. Also, When the mesh refinement factor is 1:5 in 2-dimension, for each one coarse grid point there are 25 fine grid points corresponding to it.

Case 2: Both space and time refinement ratio are 1:3

-Step 1. Get the water surface elevation at $t = (n + 1/2)\Delta t$ in the outer region (coarse grid) by solving continuity equation.

-Step 2. To solve the water surface elevation in the inner region, we need to have the flux information along the connected boundary at $t = n\Delta t$. So the flux values in the outer grids at the connected boundary are linearly interpolated and then those interpolated values are set to the fluxes in the inner at the boundary.

-Step 3. Solve the water surface elevation at $t = (n + 1/6)\Delta t$ in the inner grid region using continuity equation.

-Step 4. Solve the flux values at $t = (n + 2/6)\Delta t$ in the inner grid region using momentum equations.

-Step 5. Solve the water surface elevation at $t = (n + 3/6)\Delta t$ in the inner grid region. To transfer (update) the information from the inner grid region to the outer region, if the free surface elevation in the inner grid region lies at the same position for the outer region. Therefore, the values transferred directly to the coarse grid in the outer region. Otherwise, use the full-weighting or average operators in both spatially and timely values for updating.

-Step 6. Solve the flux values at $t = (n + 4/6)\Delta t$ in the inner grid region using continuity equation.

-Step 7. Solve the water surface elevation at $t = (n + 5/6)\Delta t$ in the inner grid region by solving continuity equation.

-Step 8. Solve the flux values at $t = (n + 1)\Delta t$ in the inner region by solving momentum equations.

-Step 9. Solve the flux values at $t = (n + 1)\Delta t$ in the outer region by solving momentum equations.

-Step 10. Transfer all the information at $t = (n + 1)\Delta t$ from the inner to outer region and update the values .

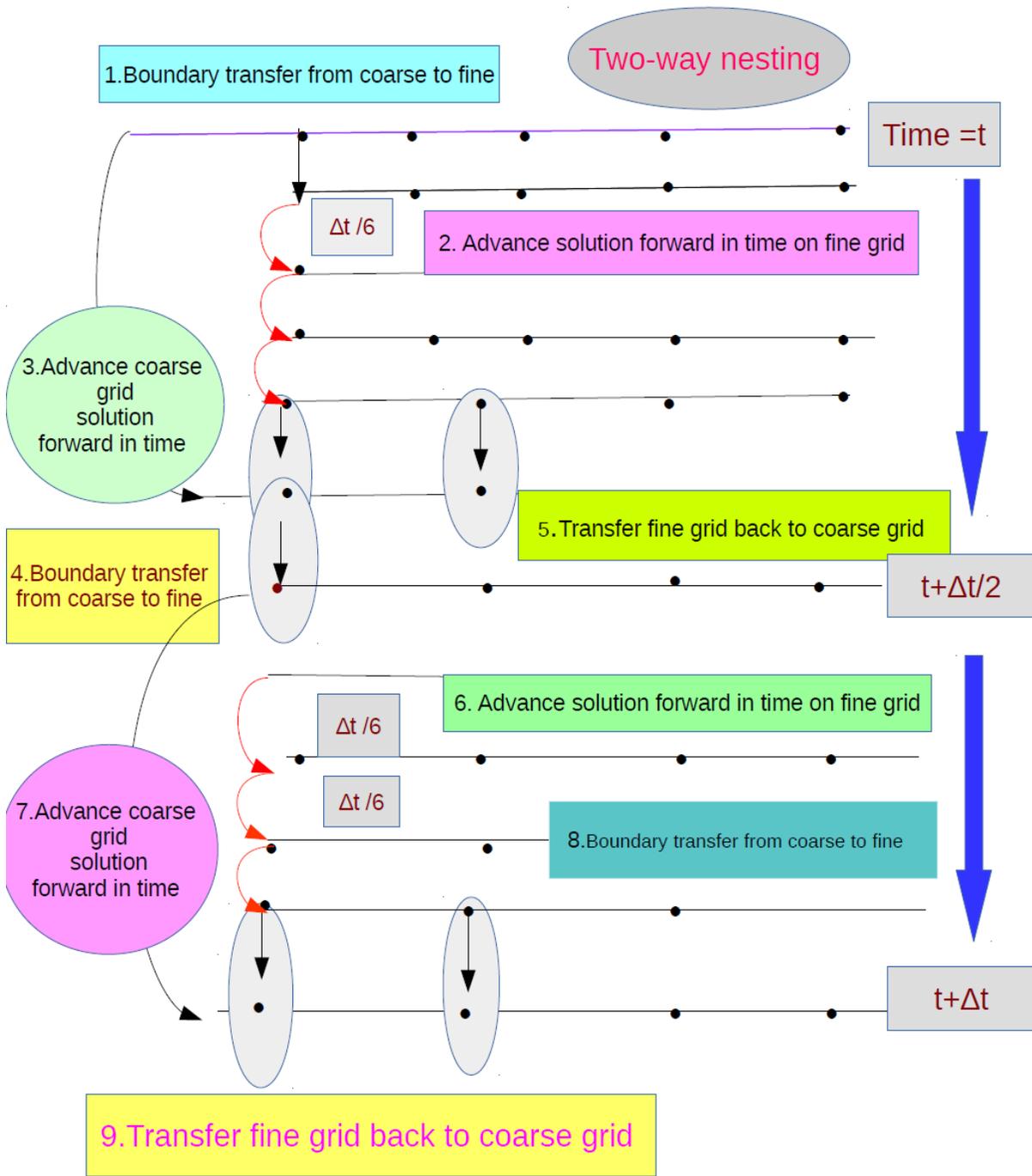


Figure 4-16: Simulation involving two-way nesting.

Case 3: The time step for the fine region is the same time step for the coarse region and the space refinement ratio 1:3 (no time refinement)

- Step 1.** Get the water surface elevation at $t = (n + 1/2)\Delta t$ in the outer region by solving continuity equation.
- Step 2.** To solve the continuity equation in the inner region but, we need to have the flux information along the connected boundary at $t = n\Delta t$. So the flux values in the outer grids at the connected boundary are interpolated and then those interpolated values are set to the fluxes in the inner at the boundary.
- Step 3.** Get the free surface elevation at $x = (n + 1/3)\Delta x$, $y = (n + 1/3)\Delta y$ and $x = (n + 2/3)\Delta x$, $y = (n + 2/3)\Delta y$ in the inner grid region by solving continuity equation and use the average in space between two points to obtain the free surface elevation at $x = (n + 1/2)\Delta x$, $y = (n + 1/2)\Delta y$.
- Step 4.** Get the free surface elevation at $t = (n + 1/2)\Delta t$ in the inner grid region. Transfer the information (update both spatially and temporally) from a fine grid on to a coarse grid at each time step of a coarse grid. If the feedback is a copy grid. Therefore, the values transferred directly over the grid size for the outer region (when the free surface in the inner grid located at the same position for the outer region). Otherwise, use the average operator or full-weighting operator for updating.
- Step 5.** Get the flux values at $t = (n + 1)\Delta t$ in the inner region by solving momentum equations.
- Step 6.** Get the flux values at $t = (n + 1)\Delta t$ in the outer region by solving momentum equations.
- Step 7.** Transfer all the information at $t = (n + 1)\Delta t$ from the inner to outer regions.

Remarks:

1. In all previous algorithms (cases 1,2 and 3), use the discrete formulation of 2D shallow water equations given by Section 2.4 when time step $(n + \frac{1}{2})$.
2. In case 4 and case 5, use the discrete formulation of 2D shallow water equations given by Section 2.3 when time step $(n + 1)$.

Case 4: Both space and the time refinement ratio are 1:3

Suppose all the information about the velocities and the free surface elevation in the inner region (with finer resolution) and the outer region (The parent grid with the coarsest grid resolution), are known at time level $t = n\Delta t$ and we need to obtain the inner and the outer region values at the next time steps $t = (n + 1)\Delta t$ and $t = (n + 2)\Delta t$.

- Step 1:** Get all information for the water surface elevation at $t = (n + 1)\Delta t$ in the outer region by solving continuity equation.
- Step 2:** Get all information for the flux values at $t = (n + 1)\Delta t$ in the outer region by solving momentum equations.
- Step 3:** To solve the continuity equation in the inner region, we need to have all the information along the connected boundary at $t = n\Delta t$. So the information in the outer grids at the connected boundary are linearly interpolated and then those interpolated values are set to the information in the inner region at the boundary.
- Step 4:** Get the the water surface elevation at $t = (n+1/3)\Delta t$ in the inner region by solving continuity equation.
- Step 5:** Get all information for the flux values at $t = (n + 1/3)\Delta t$ in the inner region by solving momentum equations.
- Step 6:** Get the the water surface elevation at $t = (n+2/3)\Delta t$ in the inner region by solving continuity equation.
- Step 7:** Solve the flux values at $t = (n+2/3)\Delta t$ in the inner grid region by using momentum equations.
- Step 8:** Get the water surface elevation at $t = (n + 1)\Delta t$ in the inner grid region by using continuity equation.
- Step 9:** Solve the flux values at $t = (n + 1)\Delta t$ in the inner grid region by using momentum equations.
- Step 10:** Get the water surface elevation at $t = (n+4/3)\Delta t$ in the inner grid region by using continuity equation.
- Step 11:** Get the flux values at $t = (n+4/3)\Delta t$ in the inner grid region by using momentum equations.
- Step 12:** Get the water surface elevation at $t = (n + 5/3)\Delta t$ in the inner grid region by solving continuity equation.
- Step 13:** Get the flux values at $t = (n + 5/3)\Delta t$ in the inner grid region by momentum equations.
- Step 14:** To transfer the information from the inner grid region to the outer region, if the water surface elevation and the flux values at $t = (n + 1)\Delta t$ in the inner grid region is located at the same position for the coarse grid region then use copy grid. Or, use the average method or full-weighting method in both spatially and timely values for updating.
- Step 15:** Get the water surface elevation at $t = (n+2)\Delta t$ in the inner grid region by solving continuity equation.
- Step 16:** Get the flux values at $t = (n + 2)\Delta t$ in the inner region by using momentum equations.

- Step 17:** Solve the free surface elevation and velocities at $t = (n + 2)\Delta t$ in the outer region using continuity equation and momentum equations by using updated information.
- Step 18:** Transfer the information from the inner grid region to the outer region at $t = (n + 2)\Delta t$.

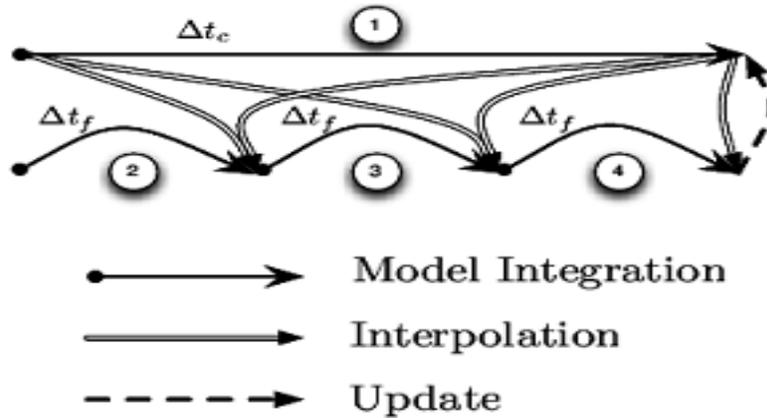


Figure 4-17: Two-way nesting

Case 5: The time step for the fine region is one half of the coarse region and the space ratio 1:3

Suppose all the information about velocities and the free surface elevation in the inner region (with finer resolutions) and the outer region (the parent grid with the coarsest grid resolution), are known at time level $t = n\Delta t$, we need to obtain the inner and the outer region values at the next time steps $t = (n + 1)\Delta t$ and $t = (n + 2)\Delta t$.

- Step 1:** Get all information for the water surface elevation at $t = (n + 1)\Delta t$ in the outer region by solving continuity equation.
- Step 2:** Get all information for the flux values at $t = (n + 1)\Delta t$ in the outer region by solving momentum equations.
- Step 3:** To solve the continuity equation in the inner region, we need to have all the information along the connected boundary at $t = n\Delta t$. So the information in the outer grids at the connected boundary are linearly interpolated and then those interpolated values are set to all the information in the inner at the boundary.
- Step 4:** Get the the water surface elevation at $t = (n + 1/2)\Delta t$ in the inner region by solving continuity equation.
- Step 5:** Get all information for the flux values at $t = (n + 1/2)\Delta t$ in the inner region by solving momentum equations.

- Step 6:** Get the water surface elevation at $t = (n + 1)\Delta t$ in the inner grid region by using continuity equation.
- Step 7:** Solve the flux values at $t = (n + 1)\Delta t$ in the inner grid region by using momentum equations.
- Step 8:** Get the water surface elevation at $t = (n + 3/2)\Delta t$ in the inner grid region by using continuity equation.
- Step 9:** Get the flux values at $t = (n + 3/2)\Delta t$ in the inner grid region by using momentum equations.
- Step 10:** To transfer the information from the inner grid region to the outer region, if the free surface elevation and the flux values at $t = (n + 1)\Delta t$ in the inner grid region is located at the same position for the coarse grid region then use copy grid. Otherwise, use average or full-weighting operators.
- Step 11:** Get the water surface elevation at $t = (n + 2)\Delta t$ in the inner grid region by solving continuity equation.
- Step 12:** Get the flux values at $t = (n + 2)\Delta t$ in the inner region by using momentum equations.
- Step 13:** Solve the free surface elevation and velocities at $t = (n + 2)\Delta t$ in the outer region using continuity equation and momentum equations by using updated the information.
- Step 14:** Transfer the information from the inner grid region to the outer region at $t = (n + 2)\Delta t$.

4.4.2 General case of computational algorithms and automatic systems when the space refinement factor is 1:3 and temporal refinement factor is 1:2

A nest is a finer-resolution model. It may be embedded together within a coarser-resolution (parent) model or independently as a separate model. The nest covers a portion of the parent domain and is driven along its lateral boundaries by the parent domain.

Here, we use the case when finer-resolution model embedded together within a coarser-resolution (parent) model.

There are three computational algorithms, an algorithm for solving the governing equations, the transferring algorithm and the back transferring algorithm. The dynamical nesting algorithm follows these steps:

(a)- For the coarse model (outer domain):

Input model data and set initial data at time $t = 0$ (that is $n = 0$ and $t = n\Delta t$ also $u_i^0 = v_i^0 = 0, H_i^0 = h_i$) on the boundary $H_i^0 = h_i + \eta_i^0$, where h is the still water depth.

1. Solve continuity equation to find $\eta_c^{n+1/2}$ using horizontal velocities components u_c^n and v_c^n at

time level n . Elevations at time level $(n - 1/2)$ and the total depth H at time level $(n - 1/2)$. Here, u_c^n , v_c^n and $\eta_c^{n-1/2}$ are known.

2. Solve the momentum equations for u_c^{n+1} and v_c^{n+1} using $\eta_c^{n+1/2}$ and u_c^n and v_c^n .
3. Solve the continuity equation for η_c^{n+1} .

(b)-For the fine model (inner domain):

The algorithm for the fine model requires more steps than the coarse model depending on the temporal refinement factor. The following illustrates the situation where the temporal refinement factor $p = 2$. Input model data and set initial data at time $t = 0$ (that is $n = 0$ and $t = n\Delta t$ also $u_i^0 = v_i^0 = 0, H_i^0 = h_i$) on the boundary $H_i^0 = h_i + \eta_i^0$.

1. Solve continuity equation for $\eta_f^{n+1/4}$ using $u_f^n, v_f^n, \eta_f^{n-1/4}$ and certain values of $\eta_c^{n+1/2}$ and $\eta_c^{n-1/2}$, which coincide with the fine grid open boundary points.
2. Solve momentum equations for $u_f^{n+1/2}$ and $v_f^{n+1/2}$ using $\eta_f^{n+1/4}$, we can get the value of $\eta_c^{n+1/2}$ using continuity equation.
3. Solve continuity equation for $\eta_f^{n+3/4}$ using $u_f^{n+1/2}, v_f^{n+1/2}, \eta_f^{n+1/4}$ and certain values of $\eta_c^{n+1/2}$ and $\eta_c^{n-1/2}$ which coincide with the fine grid open boundary points.
4. Solve momentum equations for u_f^{n+1} and v_f^{n+1} using $\eta_f^{n+3/4}$ and we can get the value of η_c^{n+1} using continuity equation.
5. Feedback all the information from the inner grid region to the outer region.

4.4.3 Transferring (interpolation) algorithm

Case 1: The transferring algorithm is described as follows when the temporal refinement factor is 1:2 :

1. When the coarse model that is at the end of time level $n + 1$ and having the values of the coarse grid points namely $\eta_A^{n+1/2}, \eta_D^{n+1/2}$ and $\eta_H^{n+1/2}$ calculate $\eta_a^{n+1/4}, \eta_d^{n+1/4}$ and $\eta_h^{n+1/4}$ by using the following interpolation in time

$$\eta_a^{n+1/4} = (3\eta_A^{n+1/2} + \eta_A^{n-1/2})/4$$

and value of $\eta_d^{n+1/4}$ and $\eta_h^{n+1/4}$ can be calculate similarly

$$\eta_d^{n+1/4} = (3\eta_D^{n+1/2} + \eta_D^{n-1/2})/4$$

and

$$\eta_h^{n+1/4} = (3\eta_H^{n+1/2} + \eta_H^{n-1/2})/4$$

2. Compute the values of elevations at the same overlapping fine grid points for the next half time level using

$$\eta_a^{n+3/4} = (5\eta_A^{n+1/2} - \eta_A^{n-1/2})/4$$

$$\eta_d^{n+3/4} = (5\eta_D^{n+1/2} - \eta_D^{n-1/2})/4$$

$$\eta_h^{n+3/4} = (5\eta_H^{n+1/2} - \eta_H^{n-1/2})/4$$

3. Use linear interpolation in space to calculate the elevations at the other grid elements on the interface between the fine and coarse grids, namely $\eta_b^{n+1/4}$, $\eta_b^{n+3/4}$, $\eta_c^{n+1/4}$, $\eta_c^{n+3/4}$ and so on. The formula for the grid point b at the time level $n + 1/4$ is

$$\eta_b^{n+1/4} = (2\eta_a^{n+1/4} + \eta_d^{n+1/4})/3$$

and a similar formula can be used for the grid point f using $\eta_a^{n+1/4}$ and $\eta_h^{n+1/4}$. For grid point c the equation is

$$\eta_c^{n+1/4} = (\eta_a^{n+1/4} + 2\eta_d^{n+1/4})/3$$

A similar formula applies for grid point g using the data from a and h .

$$\eta_g^{n+1/4} = (\eta_a^{n+1/4} + 2\eta_h^{n+1/4})/3$$

If the elevation at e is just inside the grid, that is, the coastal boundary passes through the velocity of this grid then $\eta_e^{n+1/4}$ must be extrapolated from a and d that is

$$\eta_e^{n+1/4} = (4\eta_d^{n+1/4} - \eta_a^{n+1/4})/3$$

Similar formula at each grid point can be used to obtain the values of η at the level $n + 3/4$.

$$\eta_b^{n+3/4} = (2\eta_a^{n+3/4} + \eta_d^{n+3/4})/3$$

$$\eta_c^{n+3/4} = (\eta_a^{n+3/4} + 2\eta_d^{n+3/4})/3$$

$$\eta_e^{n+3/4} = (4\eta_d^{n+3/4} - \eta_a^{n+3/4})/3$$

The following figure shows embedding the fine grid within the coarse grid. The larger bars and crosses are the overlapping grid points. The coordinates of the coarse grid are shown by capital letters and for the fine grid by lower case letters.

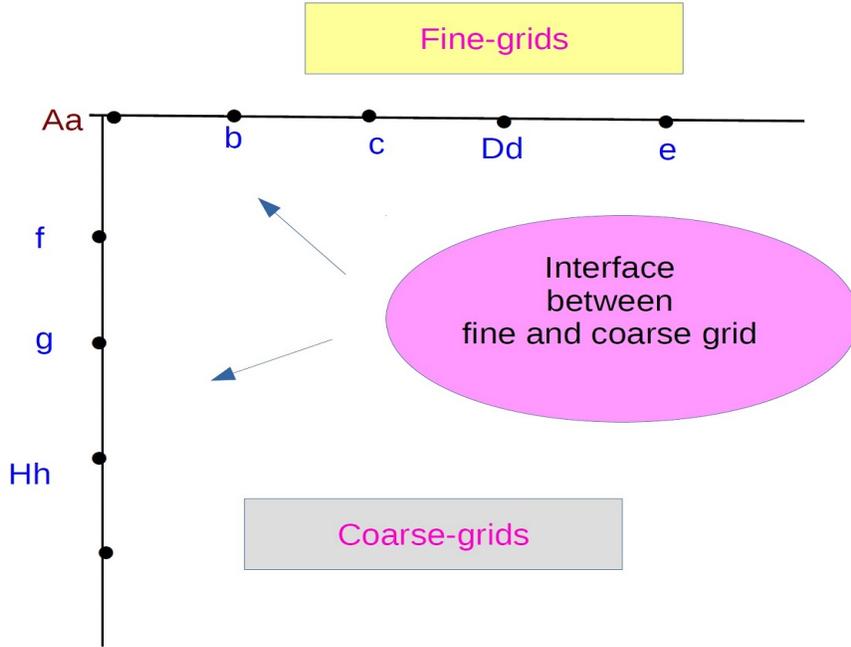


Figure 4-18: Embedding the fine grid within the coarse grid.

Case 2: The transferring algorithm is described as follows when the temporal refinement factor is 1:3

1. When the coarse model that is at the end of time level $n + 2$ and having the values of the coarse grid points namely η_A^{n+1} , η_D^{n+1} and η_H^{n+1} calculate $\eta_a^{n+1/3}$, $\eta_d^{n+1/3}$ and $\eta_h^{n+1/3}$ by using the following interpolation in time

$$\eta_a^{n+1/3} = (2\eta_A^{n+1} + \eta_A^{n-1})/3$$

and value of $\eta_d^{n+1/3}$ and $\eta_h^{n+1/3}$ can be calculate similarly

$$\eta_d^{n+1/3} = (2\eta_D^{n+1} + \eta_D^{n-1})/3$$

and

$$\eta_h^{n+1/3} = (2\eta_H^{n+1} + \eta_H^{n-1})/3$$

2. Compute the values of elevations at the same overlapping fine grid points for the next half time level using

$$\eta_a^{n+5/3} = (4\eta_A^{n+1} - \eta_A^{n-1})/3$$

$$\eta_d^{n+5/3} = (4\eta_D^{n+1} - \eta_D^{n-1})/3$$

$$\eta_h^{n+5/3} = (4\eta_H^{n+1} - \eta_H^{n-1})/3$$

3. Use linear interpolation in space to calculate the elevations at the other grid elements on the interface between the fine and coarse grids, namely $\eta_b^{n+1/3}$, $\eta_b^{n+5/3}$, $\eta_c^{n+1/3}$, $\eta_c^{n+5/3}$ and so on. The formula for the grid point b at the time level $n + 1/3$ is

$$\eta_b^{n+1/3} = (2\eta_a^{n+1/3} + 4\eta_d^{n+1/3})/6$$

and a similar formula can be used for the grid point f using $\eta_a^{n+1/3}$ and $\eta_h^{n+1/3}$.

$$\eta_f^{n+1/3} = (2\eta_a^{n+1/3} + 4\eta_h^{n+1/3})/6$$

For grid point c the equation is

$$\eta_c^{n+1/3} = (4\eta_a^{n+1/3} + 2\eta_d^{n+1/3})/6$$

A similar formula applies for grid point g using the data from a and h .

$$\eta_g^{n+1/3} = (4\eta_a^{n+1/3} + 2\eta_h^{n+1/3})/6$$

Otherwise, if the elevation at e is just inside the grid, that is the coastal boundary passes through the velocity of this grid then $\eta_e^{n+1/3}$ must be extrapolated from a and d , that is:

$$\eta_e^{n+1/3} = (7\eta_d^{n+1/3} - \eta_a^{n+1/3})/6$$

Similar formula at each grid point can be used to obtain the values of η at the level $n + 5/3$.

$$\eta_e^{n+5/3} = (7\eta_a^{n+5/3} - \eta_d^{n+5/3})/6$$

and the formula for the grid point b at the next time level $n + 5/3$ is

$$\eta_b^{n+5/3} = (2\eta_a^{n+5/3} + 4\eta_d^{n+5/3})/6$$

and

$$\eta_c^{n+5/3} = (4\eta_a^{n+5/3} + 2\eta_d^{n+5/3})/6$$

4.5 Free surface and Velocity Updates

In this section, we introduce four choices of restriction operator for the free surface elevation η and velocities (u, v) on Arakawa C-grid suggested by [48] and we will apply these options to several examples in various ways in later chapters.

Operator	Direction x, y	η	u - velocity	v -velocity
Average	x	Average	Average	Average
Average	y	Average	Average	Average
Full-weight	x	Full-weight	Full-weight	Full-weight
Full-weight	y	Full-weight	Full-weight	Full-weight
Update-mix-low	x	Average	Copy	Average
Update-mix-low	y	Average	Average	Copy
Update-mix-high	x	Full-weight	Average	Full-weight
Update-mix-high	y	Full-weight	Full-weight	Average

Table 4.1: Shows the restriction operator for free surface elevation η and velocities (u, v) on Arakawa C-grid.

4.6 Algorithm for Coupling the Coarse and Fine Grids Using Open Boundary Conditions

Open boundary conditions can be obtained for the fine grid model from the coarse grid model. The water elevations and velocities at the open boundaries in the coarse grid model are stored at certain time intervals. The time step used in the fine-grid model is usually smaller than the one used in the coarse-grid model. Therefore, a temporal interpolation scheme is required to provide the corresponding values in the open boundary for the fine-grid model at each step. The velocities and elevations along the open boundaries for the fine-grid model are prescribed using spatial interpolation on the coarse grid model.

A typical interpolation for open boundary using a nested-grid system. The spatial interpolation schemes are:

1. Western open boundary

(Here, if i -start, then $i = 1$ and $j = 2$ to end-1 also $U_{i,j}, V_{i,j}$, and $\eta_{i,j} \neq 0$)

$$u_1 = u_2 = u_3 = u_4 = U_{i,j}$$

$$u(1, 1) = u(1, 2) = u(1, 3) = u(1, 4) = U_{i,j}$$

$$v_1 = V_{i,j}$$

$$v_2 = v_3 = v_4 = SI(V_{i,j}, V_{i,j+1})$$

$$\eta_1 = \eta_2 = \eta_3 = \eta_4 = SI(\eta_{i-1,j}, \eta_{i,j})$$

where u_1, u_2, u_3, u_4 represent the value of velocity u in fine grid and $v_1, v_2, v_3, v_4, \eta_1, \eta_2, \eta_3,$ and η_4 represent the value of velocity v and elevation η in fine grid and $U_{i,j}, V_{i,j}$ represent the value of velocities in coarse grid. Also, $SI(V_1, V_2)$ refers to a linear interpolation between two variables V_1 and V_2 more accurate scheme for above equations may include values from additional neighboring grids.

2. Eastern open boundary

(Here, if $i = \text{end}$, then $i = \text{end}$ and $j = 2$ to $\text{end}-1$ also $U_{i,j}, V_{i,j}, \eta_{i,j} \neq 0$). The spatial interpolation schemes are:

$$u_1 = u_2 = u_3 = u_4 = U_{i+1,j}$$

$$v_1 = V_{i,j}$$

$$v_2 = v_3 = v_4 = SI(V_{i,j}, V_{i,j+1})$$

$$\eta_1 = \eta_2 = \eta_3 = SI(\eta_{i+1,j}, \eta_{i,j})$$

3. Southern open boundary

(Here, if $j = \text{start}$, then $j = 1$ and $i = 2$ to $\text{end}-1$). The spatial interpolation schemes are:

$$v_1 = v_2 = v_3 = v_4 = V_{i,j}$$

$$u_1 = U_{i,j}$$

$$u_2 = u_3 = u_4 = SI(U_{i,j}, U_{i+1,j})$$

$$\eta_1 = \eta_2 = \eta_3 = SI(\eta_{i,j-1}, \eta_{i,j})$$

4. Northern open boundary

(Here, if $j = \text{end}$, then $j = \text{end}$ and $i = 1$ to $\text{end}-1$). The spatial interpolation schemes are:

$$v_1 = v_2 = v_3 = v_4 = V_{i,j+1}$$

$$u_1 = U_{i,j}$$

$$u_2 = u_3 = u_4 = SI(U_{i+1,j}, U_{i,j})$$

$$\eta_1 = \eta_2 = \eta_3 = SI(\eta_{i,j+1}, \eta_{i,j})$$

5. West-North open boundary

The spatial interpolation schemes are:

The corner point is $(1, j\text{-end})$, to find $u_{i,j}$ use linear interpolation. The spatial interpolation schemes are:

$$u_{1,j} = U_{1,j\text{-end}}$$

$$v_{1,j} = SI(V_{1,j\text{-end}-1}, V_{1,j\text{-end}})$$

$$\eta_{1,j} = SI(\eta_{1,j\text{-end}}, \eta_{2,j\text{-end}})$$

6. North-East open boundary

Here, the corner point $(i\text{-end}, j\text{-end})$. The spatial interpolation schemes are:

$$u_{i\text{-end},j\text{-end}} = SI(U_{i\text{-end}-1,j\text{-end}}, U_{i\text{-end},j\text{-end}})$$

$$v_{i\text{-end},j\text{-end}} = V_{i\text{-end},j\text{-end}}$$

$$\eta_{i\text{-end},j\text{-end}} = SI(\eta_{i\text{-end},j\text{-end}}, \eta_{i\text{-end},j\text{-end}-1})$$

7. East-South open boundary

Here, the corner point $(i\text{-end}, 1)$. The spatial interpolation schemes are:

$$v_{i\text{-end},1} = SI(V_{i\text{-end},2}, V_{i\text{-end}-1,1})$$

$$u_{i\text{-end},1} = U_{i\text{-end},1}$$

$$\eta_{i\text{-end},1} = SI(\eta_{i\text{-end},1}, \eta_{i\text{-end}-1,1})$$

8. South-West open boundary

The spatial interpolation schemes are:

Here, the corner point $(1,1)$

$$u_{1,1} = SI(U_{1,1}, U_{2,1})$$

$$v_{1,1} = V_{1,1}$$

$$\eta_{1,1} = SI(\eta_{1,1}, \eta_{1,2})$$

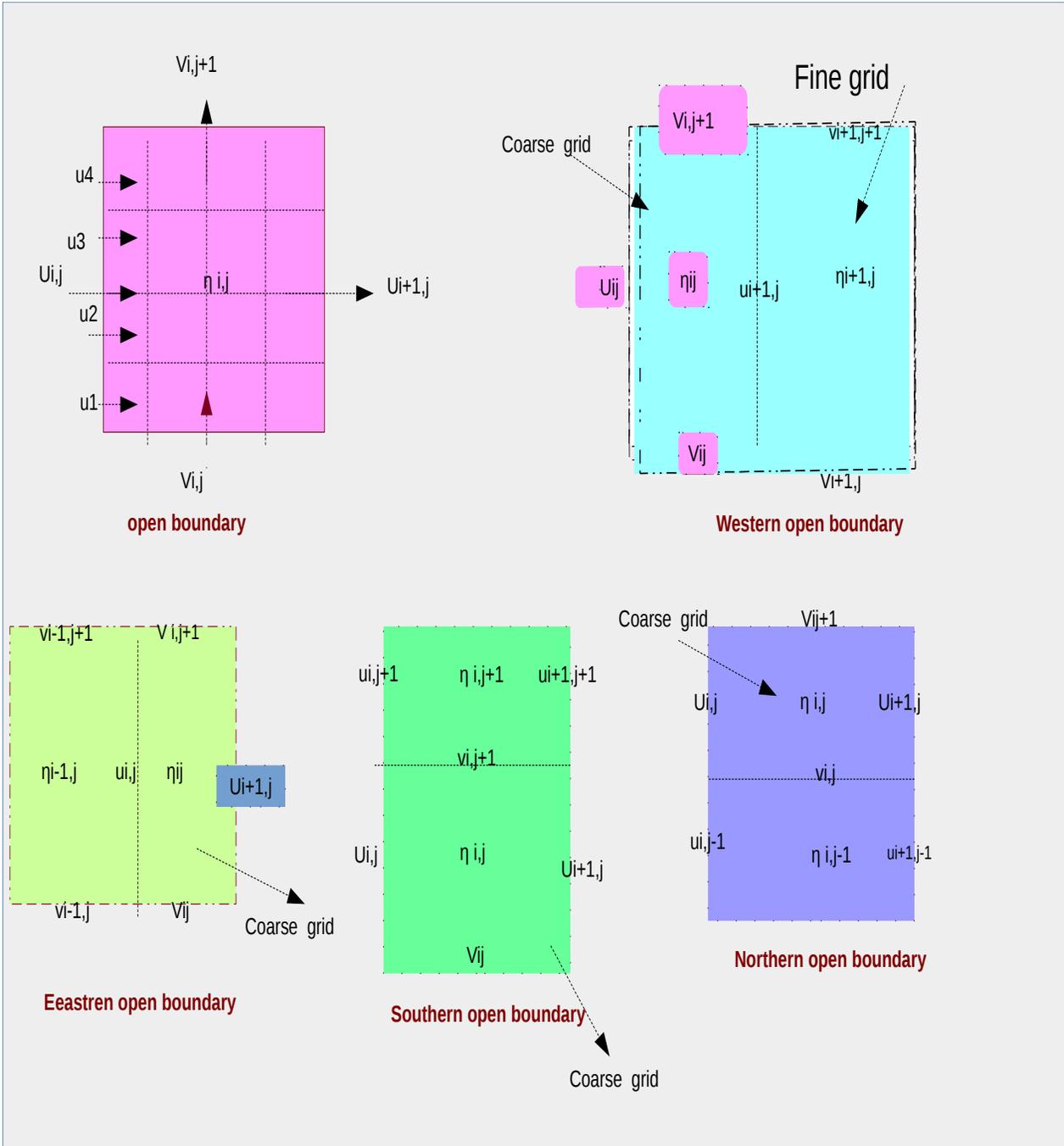


Figure 4-19: Diagram for open boundary condition.

4.7 Summary and Conclusion

A two-way nested model was developed and it is a modified form of the one-way nesting procedure for 2DSWEs. This model is capable of simulating the processes at a high-resolution and allowing the data to be feedback into the low-resolution domain. The grid structure employed was an embedded method and identified an effective data transmission method was introduced in maximizing the transmission of data while preventing the generation of an error. Some of different update interpolation scheme was also used such that direct copy, average, Shapiro, full-weighting, Mix-low, and Mix-high were applied.

This chapter discussed several important issues, including: Different structured grids were outlined in relation to their spatial resolution and temporal resolution. The advantages and disadvantages of nesting modeling techniques were discussed. The two-way nested modeling technique was explained in detail with regard to its mathematical formulation. Different cases of open boundary conditions for the two-way nesting grids were studied. Some new algorithms and diagrams were built for the two-way nesting technique to solve 2DNSWEs with the refinement ratio used is 1:3 or 1:5 or both. Finally, four choices of restriction operator for the free surface and the velocities on Arakawa C-grid have been suggested. All new algorithms and techniques presented in this chapter will be used in the subsequent chapters.

Chapter 5

Performance of Two-Way Nesting Techniques for SWEs

Chapter 5

Performance of Two-Way Nesting Techniques for SWEs

The results presented in this chapter are the subject of an article [\[3\]](#).

The principal objective of this chapter is to study the efficiency of two-way nesting performance technique for structured grids between the coarse and the fine grids for linear 2DSWEs using an explicit finite difference method in space and leapfrog with Robert-Asselin filter in time. Multiple nested grid technique is described and successfully applied to 2DSWEs. This model consists of a higher-resolution (fine grid) model with nesting 3:1 embedded in a low-resolution (coarse grid) model on which covers the entire domain.

A numerical scheme to construct a two-way nested grid model is proposed. Dynamical coupling in a two-way nesting system is performed at a dynamical interface which is a separate/adjacent from a mesh interface. Dynamical interaction is achieved by a method which conserves mass, and the momentum of the system.

The main role for the update schemes is to use four choices of restriction operator for the free surface elevation η and velocities u, v on Arakawa C-grid. Comparison nested model results with a fine grid and a coarse grid results, show that two-way nesting technique works efficiently over different periods of time. The results indicate good performance of the nesting technique.

Highlights

- Propose numerical methods which used to construct a new two-way interaction techniques for 2DSWEs.
- Full-weighting and the average update operator are applied for some examples 2DSWEs.
- The reliability and the accuracy of 2DSWEs are tested through several examples.

Contents

5.1	Configuration of the numerical experiments	160
5.2	Methodology (Development)	161
5.3	Experiments with linear shallow water models	162
5.3.1	Case 1: When space refinement ratio is 1:3 and the temporal refinement ratio is 1:2 for adjacent structured grids	162
5.3.2	Case 2: When the space and temporal refinement ratio is 1:3 for the structured grid with a separate interface using algorithm 4, Chapter 4	170
5.4	Comparison <i>l2-RE</i> when use four choices of restriction operator with the refinement ratio in both space and temporal equals 1:3	173
5.5	Comparison between one-way and two-way nesting grid when the refinement ratio in both space and temporal equals 1:3	176
5.6	Summary and Conclusion	179

5.1 Configuration of the Numerical Experiments

A nested grid model means in a region of a single grid size, there are one or more regions with smaller grid sizes situated in which eventually forms a hierarchy of grids or grid levels. The parent grid in the nested model is a copy of the coarse grid model with the same resolutions and the parameters while the child grid is a copy of the fine grid model. The coarse and fine models were used to investigate the relationship between the model accuracy and resolution. The results are presented and demonstrate the applicability and benefits of nested models.

The performance and accuracy of the nested model are determined in the following manner. The coarse model results are first compared with the fine model results to determine the accuracy of the fine model. Two-way nested model results are then compared with the fine model results to determine the accuracy of the nested model. Both errors are then compared in order to determine the level of improvement in model accuracy achieved by using the nested model instead of the coarse model.

Investigate some tests to determine a suitable feedback operator, which contains an appropriate feedback condition and interpolation technique. The optimum feedback operator is identified by using averaging interpolation procedure or (full- weighting). The validity of the nesting method is shown for some examples and the numerical results show that the model more accurately by using these methods.

5.2 Methodology (Development)

Consider the nested grids modeling in which a coarse grid contains one or more than one fine grid in one level or multiple levels. we **suggest** a new technique for multiple nested grids of linear 2DSWEs by using some types of interpolations and restriction operators given in Chapter 4.

We present and evaluate a set of choices made an implementation of two-way nesting methods allowing simultaneous spatial and temporal refinement in shallow water model. The choice a full-weighting operator is applied for update the water surface elevation η , and velocities when a type of structured grid **without a separate** (adjacent) interface. Otherwise, average update operator is used, when a structured grid is **a separate dynamic interface** and feedback interface (see Section 4.5).

Some Notes

1. For all examples, we apply an explicit center finite difference scheme in space and leapfrog scheme with Robert-Asselin filter in time to approximate 2D depth-averaged linear SWEs for two-way nested model which contain coarse and fine grids.
2. For all examples, we find $l2$ -error norm, $H1$ -error norm in a coarse grid and a fine grid. Also, find $l2-RE$ of free surface elevation for two-way nesting grids.
3. All the simulation are made by using Dirichlet boundary conditions.
4. Boundary conditions for the nested domain are linearly interpolated (both spatially and temporally) from the coarse domain and feedback (both spatially and temporally) using the average method or full-weighting method with (separable/adjacent) interface from the high-resolution nested grids solution to the low-resolution coarse grids solution.
5. The bottom friction comes from Manning's formula, which is uniform throughout the grids, where n is roughness coefficient. In this simulation, n takes 0.013.
6. When a specific domain rotation is required. Here, we need to determine the location of the new rotated nested in relation to the coarse domain. This is done by identifying the localized coordinates for each grid point in the rotated domain with respect to the coarse domain.

5.3 Experiments with Linear Shallow Water Models

5.3.1 Case 1: When space refinement ratio is 1:3 and the temporal refinement ratio is 1:2 for structured grids

Here, by using algorithm 1, Chapter 4, the number of grids for the coarse and fine grids are equals when consider only the common points to both grids.

Example 1:

In this example, we use EFDMs to approximate 2DSWEs given in equations (2.32)-(2.34) for linear case in a rectangular domain $\Omega = [0, L_x] \times [0, L_y]$ which are approximately second order in space and time using the formula in Section 2.5, case 1. For the stability using CFL condition which is given in Section 2.8.3, Chapter 3.

1. Numerical parameters: (Simulation)

The computational domain is discretized by a grid whose size is regular. Numerical values of the parameters in a coarse grid are chosen as follows: $L_x=450$, $L_y=360$, $nx=150$, $ny=120$, $\Delta x=\Delta y=3$, and time step $\Delta t=2.5e^{-2}s$, the water depth $h=10m$, and CFL condition 0.6 with the initial condition $u=v=0$ and η given by data file.

2. Results and discussion

First of all, we tested the computational stability and accuracy of this system of equations. The time integration were performed at different times. The calculations were stable.

The following figures show the simulation of free surface elevation at time $t = 1000, 2000, 3000$ min in the coarse grid domain.

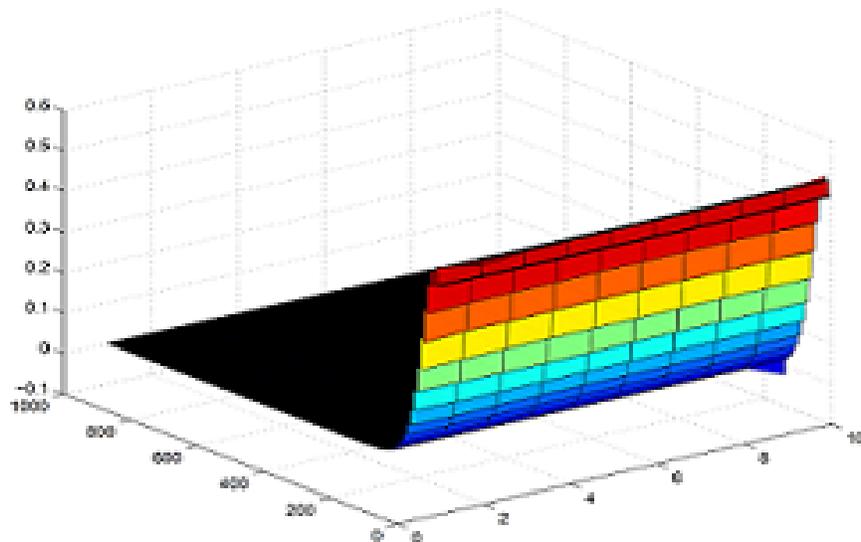


Figure 5-1: Simulation of free surface elevation in a coarse grid at time=1000 min.

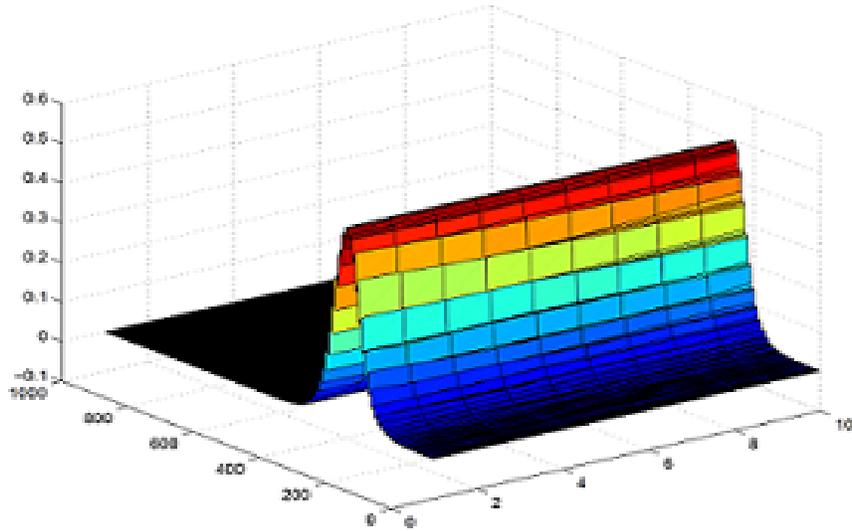


Figure 5-2: Simulation of free surface elevation in a coarse grid at time=2000 min.

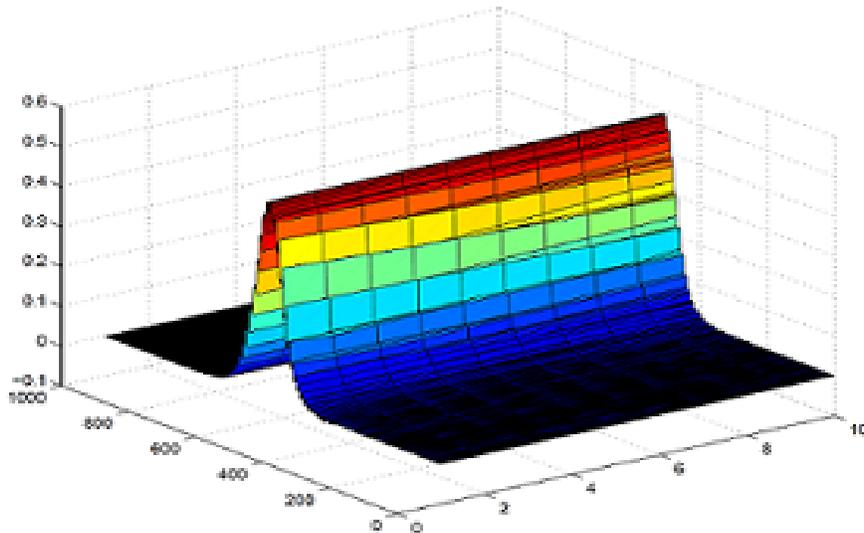


Figure 5-3: Simulation of free surface elevation in a coarse grid at time=3000 min.

Example 2:

For the same previous example, we find the simulation of free surface elevation in a fine grid.

1. Numerical parameters:

The computational domain is discretized by a grid whose size is regular. Numerical values of the parameters in a fine grid are chosen as follows: number of grids 450×360 , by consider only the common points between coarse and fine grids, the number of fine grids becomes $nx=150$, $ny=120$, $\Delta x=1$, $\Delta y=1$, time step $\Delta t=1.25e^{-2}s$, the water depth $h=10m$, and CFL condition is 0.6.

2. Results and Discussion

The following figures show the simulation of free surface elevation at different time in the fine grid domain.

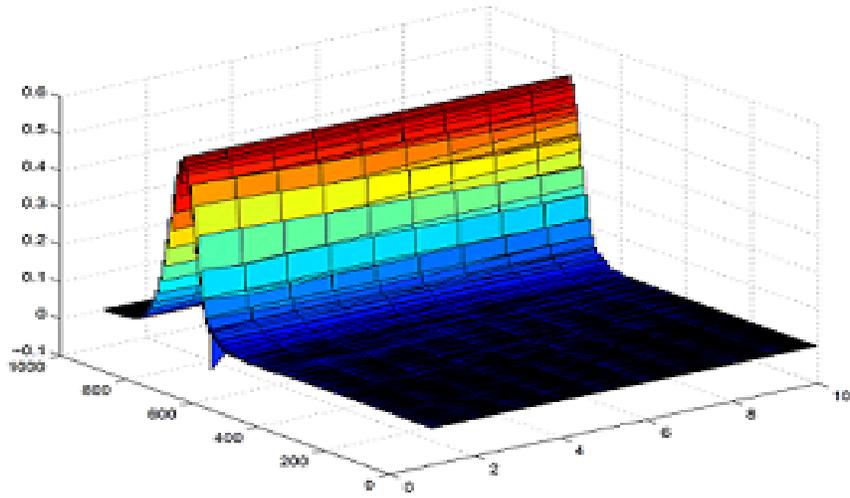


Figure 5-4: Simulation of free surface elevation in a finer grid at time=1000 min.

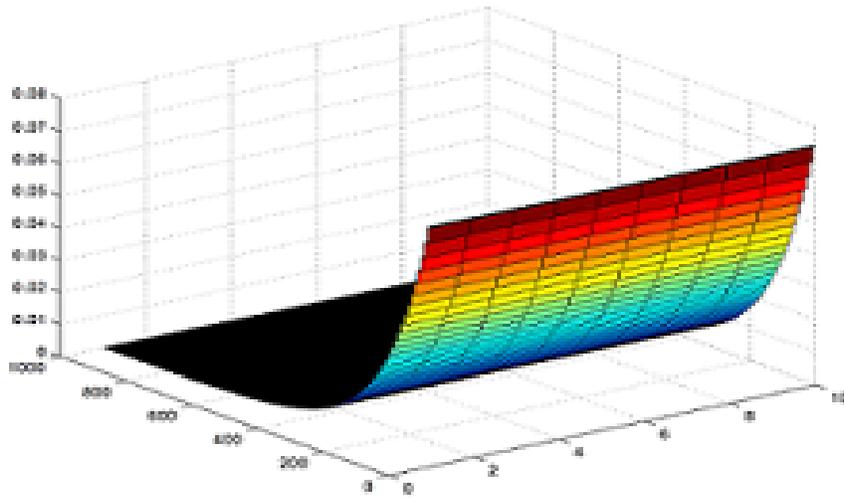


Figure 5-5: Simulation of free surface elevation in a finer grid at time=2000 min.

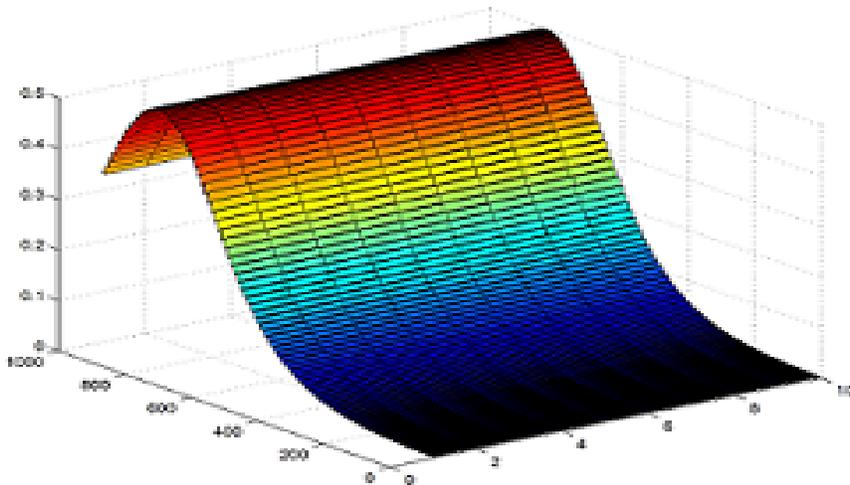


Figure 5-6: Simulation of free surface elevation in a finer grid at time=3000 min.

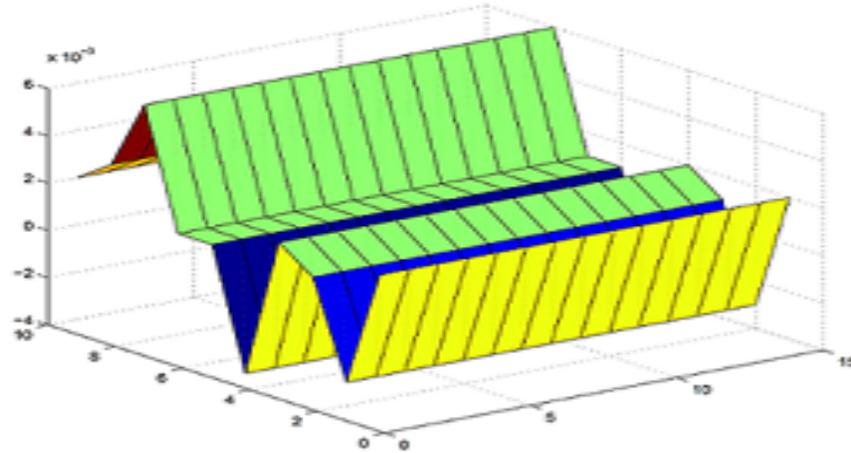


Figure 5-7: Simulation of free surface elevation in a finer grid at time=2000 hour.

Example 3: Comparison the solution between the coarse and the fine grids for adjacent structured grids.

For the same previous example, we find the approximate solution for the water surface elevation in two-way nesting technique and comparison the solution for the water surface elevation between the coarse and the fine grids.

1. Numerical parameters:

The Numerical values of the parameters are chosen as follows: $nx = 150, ny = 120, \Delta x = \Delta y = 3$, time step $\Delta t = 2.5e^{-2}$ sec in a coarse grid, $\Delta x = \Delta y = 1$, the number of fine grids $nx = 150, ny = 120$ by consider only the common points between coarse and fine grids with the space refinement ratio 1:3, the water depth $h = 10$ m, and CFL condition is 0.6.

2. Results and discussion

The time integrations were performed at 2000 min. The calculations were stable.

The following figures show the simulation of free surface elevation in two-way nesting grid at different time.

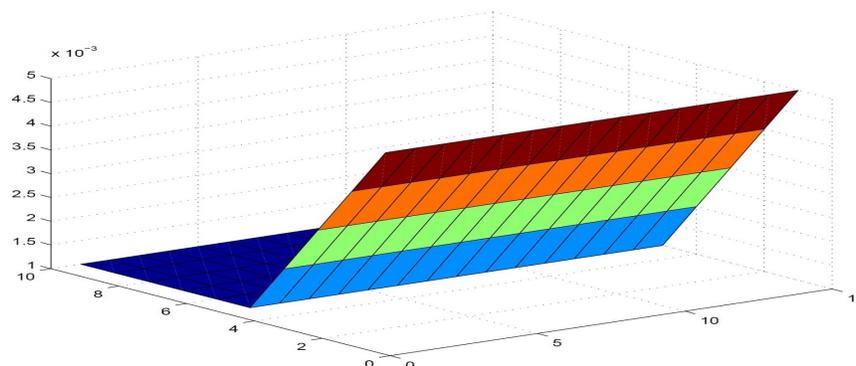


Figure 5-8: Simulation of free surface elevation in two-way nesting grids at time=1000 min.

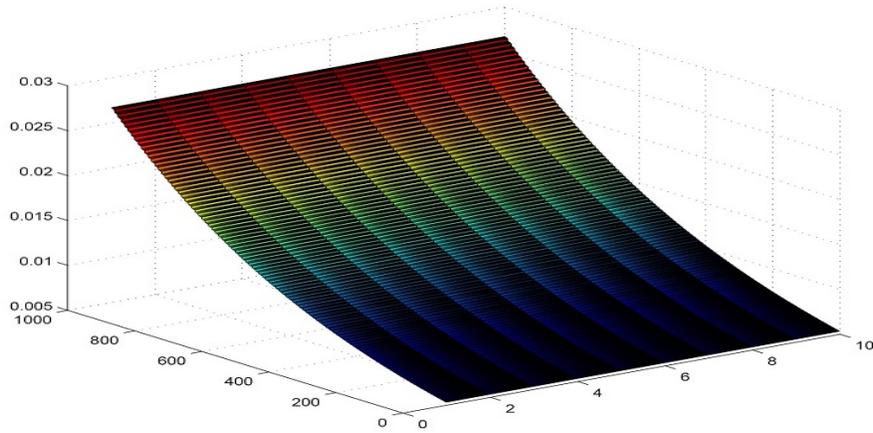


Figure 5-9: Simulation of free surface elevation in two-way nesting grids at time=2000 min.

The following figure compares the approximate solution for free surface elevation between a coarse grid and a fine grid

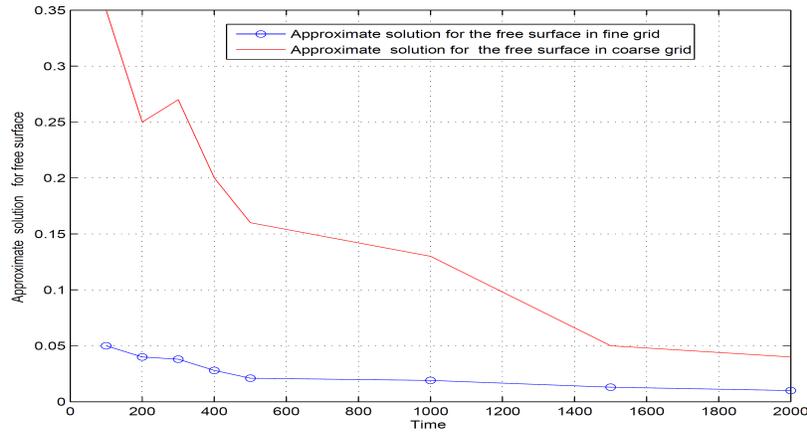


Figure 5-10: Comparison the approximate solution for free surface elevation between the coarse and the fine grids at $t = 100$ min, 200 min, ..., 2000 min.

Notes:

1. Let $e_c = \| u_c - u_f \|_{L^2}$ and $e_f = \| u_f - u_t \|_{L^2}$, where e_c and e_f are L^2 -error norm in the coarse grids and the fine grids respectively, u_c , u_f , and u_t are computed values of coarse grids variables, computed values of fine grids variables and computed values of two-way nesting grids variables.
2. Let $e_c = \| u_c - u_f \|_{H^1}$ and $e_f = \| u_f - u_t \|_{H^1}$, where e_c and e_f are H^1 -error norm in the coarse grids and the fine grids respectively.

Example 4:

In this example, we find l_2 -error norm and H_1 -error norm in a coarse grid, and a fine grid by using 2DSWEs given in equations (2.45)-(2.47) for a linear case in a domain $\Omega = [0, L_x] \times [0, L_y]$.

Numerical parameters, results and discussion

The computational domain is discretized by a grid whose size is regular. Numerical values of the parameters are chosen as follows: If we take different values of time $t = 10, 20, \dots, 1000$ hours to find l_2 -error norm and H_1 -error norm of the water surface elevation, when $L_x=L_y=300$, $n_x=100$, $n_y=100$, $\Delta x=\Delta y=3$, time step=0.01s, and total steps 36000 in a coarse grid, $h=10$, $\Delta x=\Delta y=1$, time step=0.005s in a fine grid, when consider only the common points to both grids, the number of grids becomes $n_x=n_y= 100$ in a fine grids, CFL condition is 0.04 and the initial condition for the same Example 2, Section 3.1, Chapter 3.

In the following figures show the sequence of snapshots for the approximate solution of free surface elevation at different values of time $t = 10, 20, \dots, 1000$ hours and the other figures show l_2 -error norm and H_1 -error norm in a coarse grid, and in a fine grid.

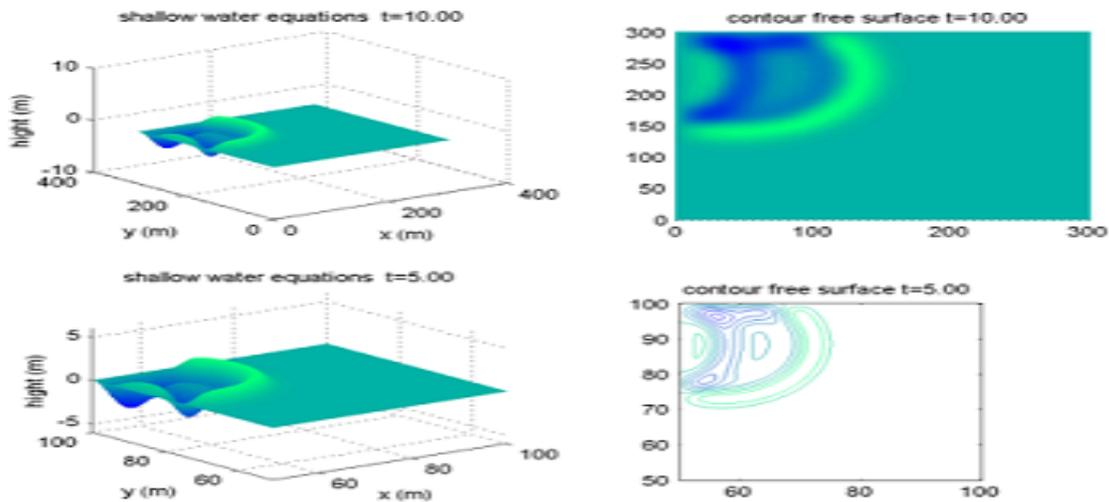


Figure 5-11: Simulation of free surface elevation at $t = 10$ min.

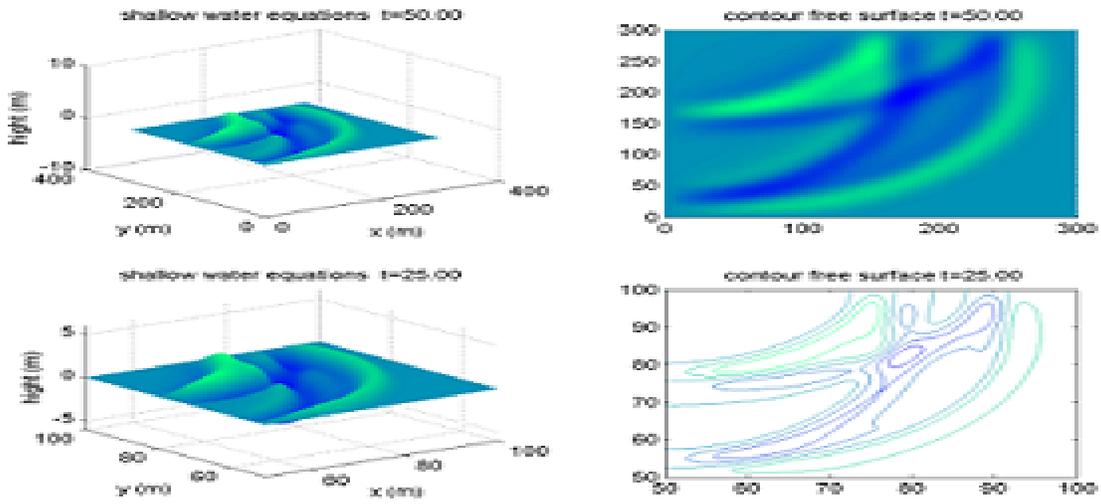


Figure 5-12: Simulation of free surface elevation at $t = 50$ min.

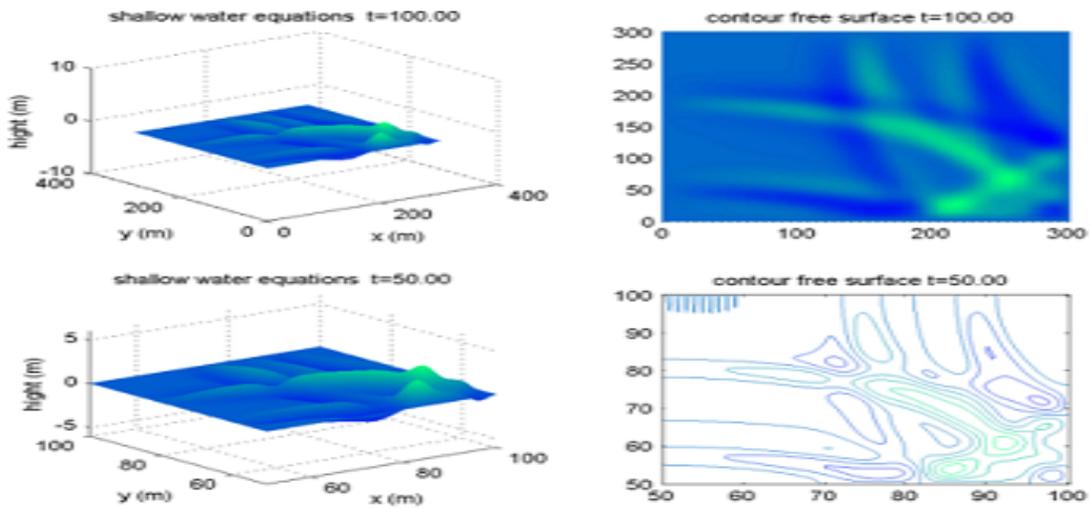


Figure 5-13: Simulation of free surface elevation at $t = 100$ min.

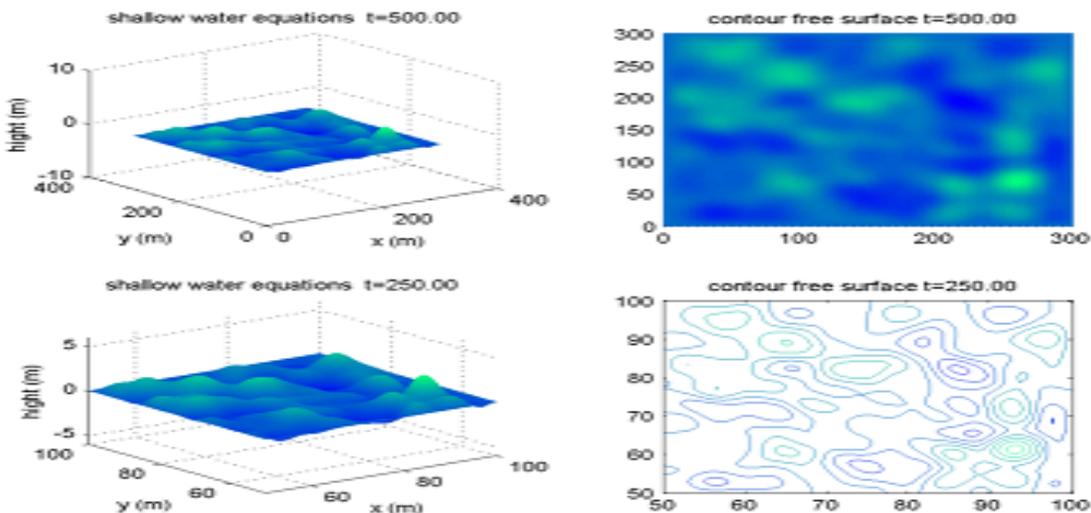


Figure 5-14: Simulation of free surface elevation at $t = 500$ hr.

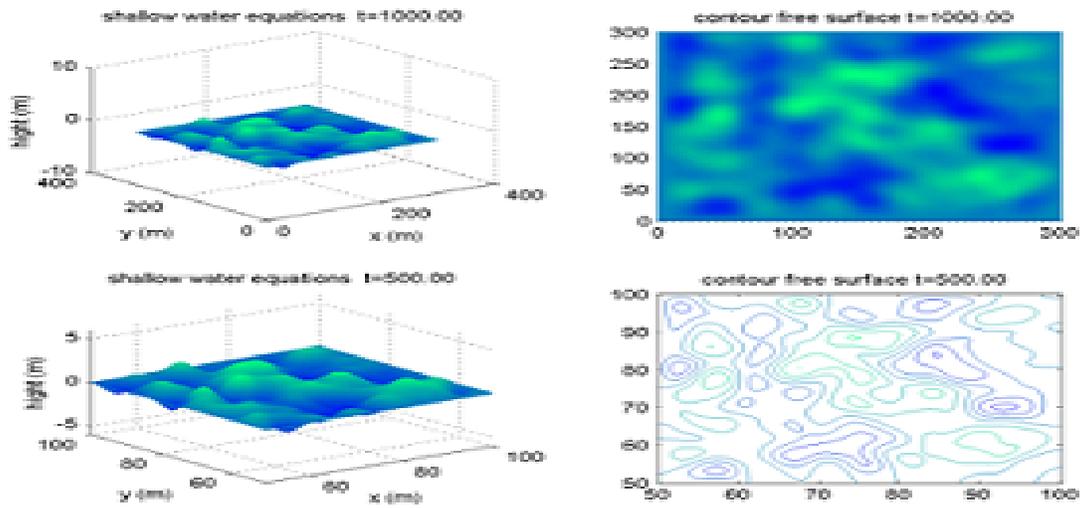


Figure 5-15: Simulation of free surface elevation at $t = 1000$ hr.

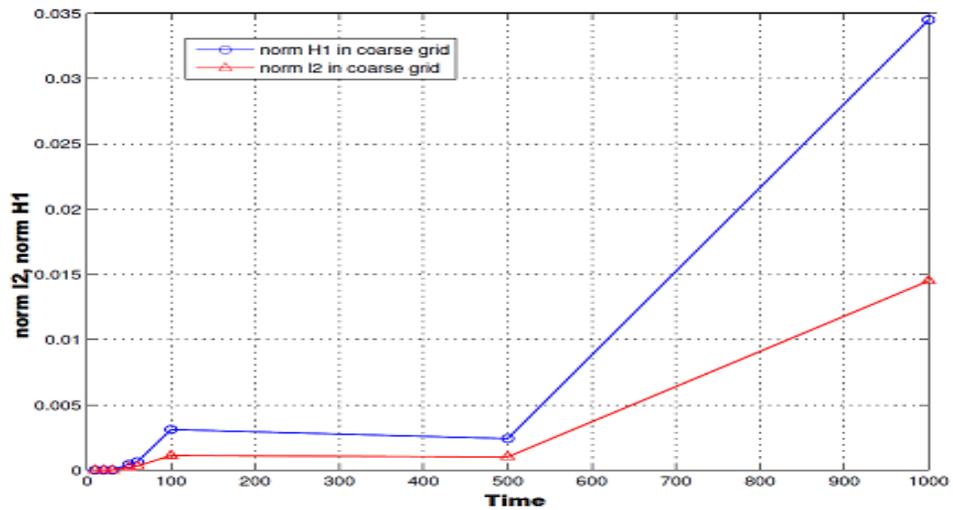


Figure 5-16: Comparison l_2 -error norm and H_1 -error norm in a coarse grid.

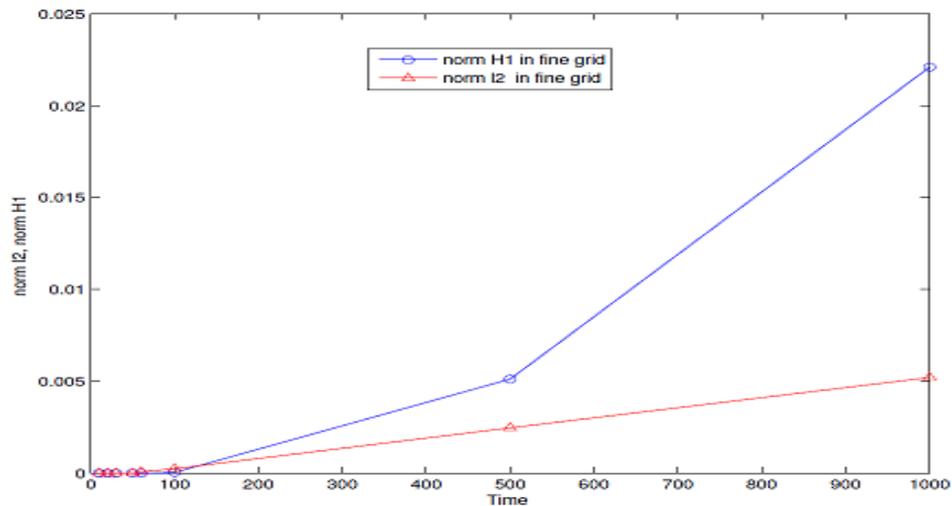


Figure 5-17: Comparison H_1 -error norm and l_2 -error norm in a fine grid.

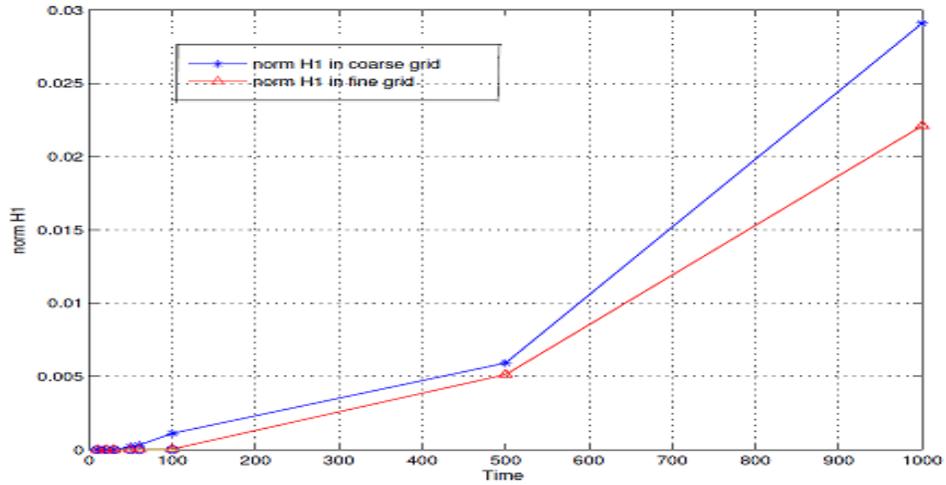


Figure 5-18: Comparison H_1 -error norm in a coarse grid, and a fine grid for linear SWEs.

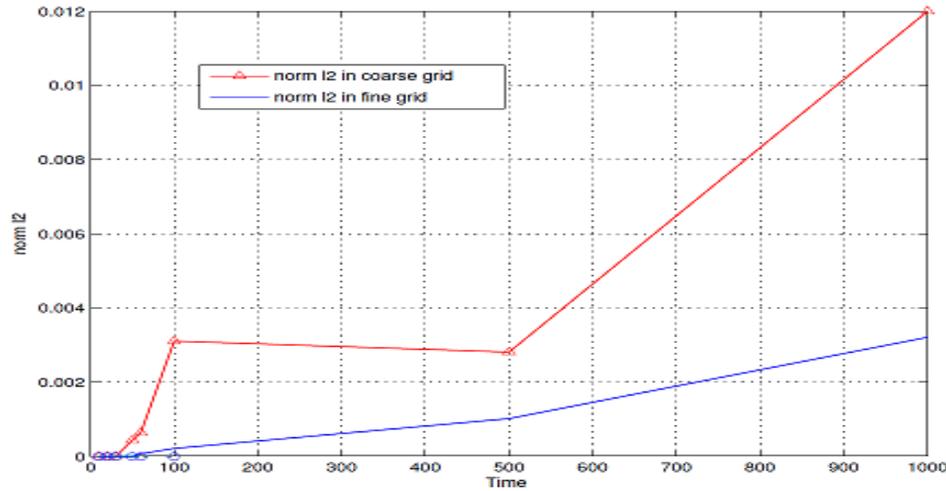


Figure 5-19: Comparison l_2 -error norm in a coarse grid, and a fine grid for linear SWEs.

5.3.2 Case 2: When both space and temporal refinement ratio are 1:3 for the structured grid with a separate interface using algorithm 4, Chapter 4

Note:

We assume that the time step in a fine grid is equal one third of the time step in a coarse grid. Therefore, we need to provide the interpolation between the coarse and the fine grids, always use Dirichlet feedback in velocities and the water surface elevation because if the two grids lies with the same positions then for the update scheme, we use copy grids or a direct-injection update scheme (copy of child values at corresponding locations of the parent grid without spatial filter). Otherwise, we use the average method to update.

Example 1:

In this example, we find l_2 -error norm and $H1$ -error norm in a coarse grid by using 2DSWEs given in equations (2.45)-(2.47) for linear case with initial condition for the same Example 2, Section 3.1, Chapter 3.

1. Numerical parameters:

The numerical values of the parameters are chosen as follows: If we take different values of time $t = 20, 30, \dots, 100$ days to find l_2 -error norm and $H1$ -error norm in a coarse grid, when $L_x=L_y=300$, $\Delta x=\Delta y=3$, time step=0.01s, and the number of grids 100×100 in a coarse grid, $\Delta x=\Delta y=1$, time step =0.0033s, 100×100 in a fine grid when consider only the common points to both grids, and the water depth=10.

2. Results and discussion

First of all, we tested the computational stability of this system of equations using CFL condition which is 0.02. The time integrations were performed for 100 days. The following figure shows l_2 -error norm and $H1$ -error norm in a coarse grid at $t = 10, 20, \dots, 100$ days.

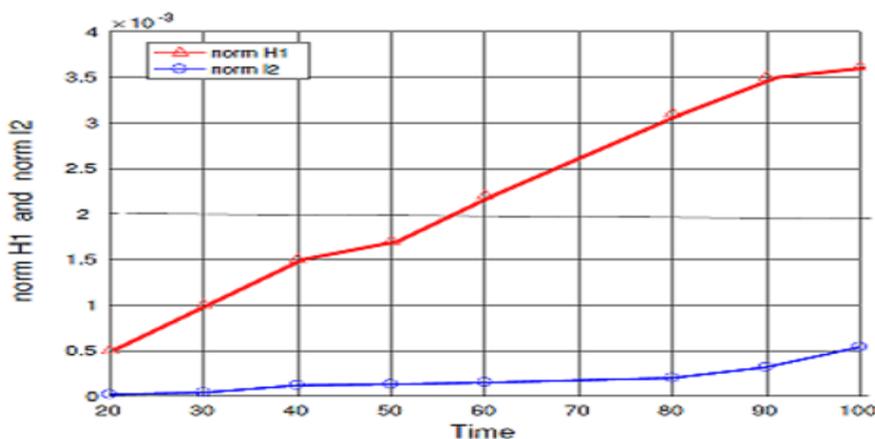


Figure 5-20: Show l_2 -error norm and $H1$ -error norm in a coarse grid.

Example 2:

By the same previous example, we find l_2 -error norm and $H1$ -error norm of free surface elevation in a fine grid using the 2DSWEs given in equations (2.45)-(2.47) for linear case.

1. Numerical parameters:

The computational domain is discretized by a grid whose size is regular. Numerical values of the parameters are chosen as follows: If we take different values of time $t = 20, 30, \dots, 100$ days to find l_2 -error norm and $H1$ -error norm of free surface elevation between a fine grid and a two-way nesting grid, when $L_x=L_y=300$, $\Delta x=\Delta y=3$, time step=0.01s, and the number of grids 100×100 in a coarse grid, $\Delta x=\Delta y=1$, time step =0.033s in a fine grid and when consider only the common points to both grids, the number of fine grids becomes 100×100 ,

2. Results and discussion

The following figures show l_2 -error norm and H_1 -error norm of free surface elevation in a coarse grid and a fine grid.

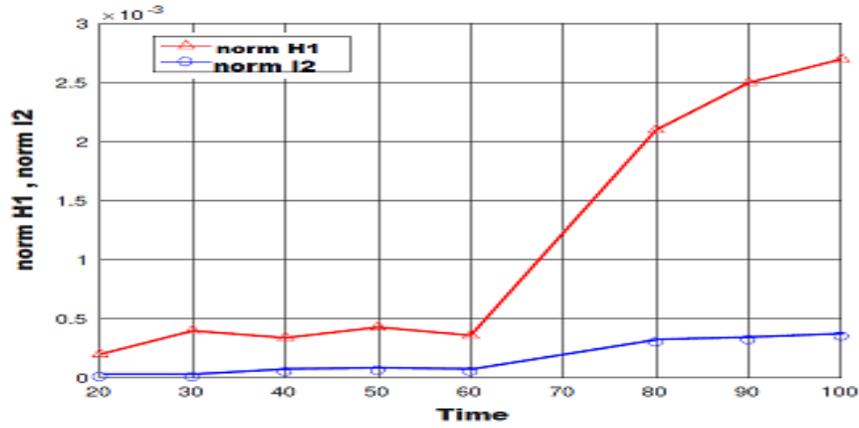


Figure 5-21: Comparison l_2 -error norm and H_1 -error norm in a fine grid at $t=10, 20, \dots, 100$ days.

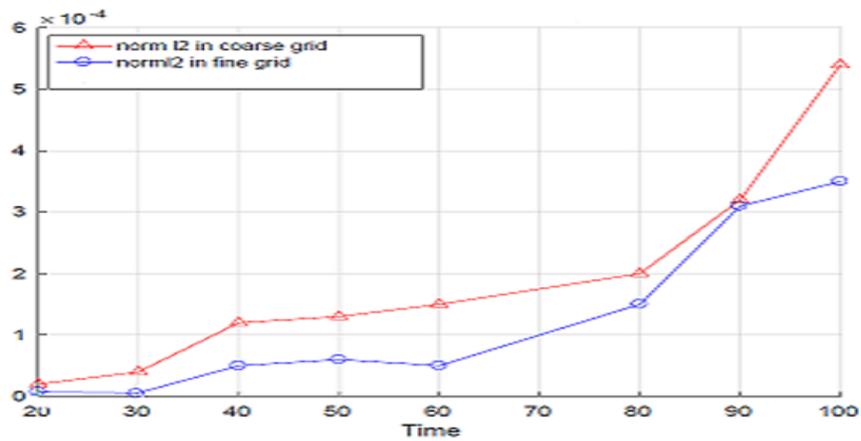


Figure 5-22: Show l_2 -error norm in a coarse grid, and a fine grid

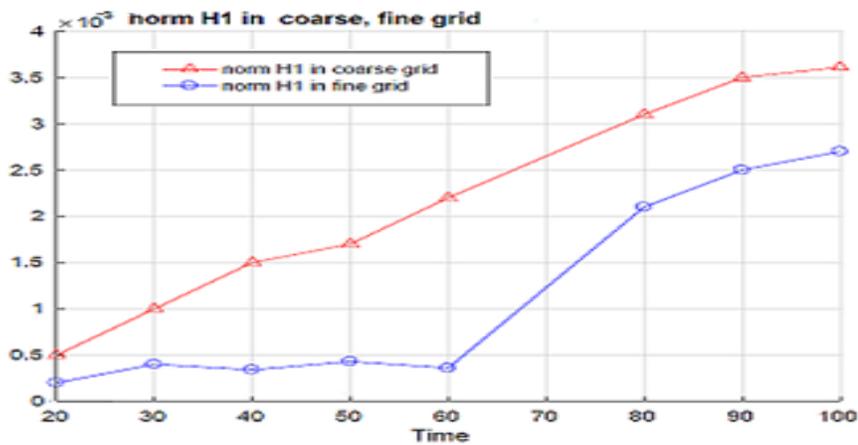


Figure 5-23: Show H_1 -error norm in a coarse grid, and a fine grid

Notes that $L2$ -error norm and $H1$ -error norm in a coarse grid are larger than $l2$ -error norm and $H1$ -error norm in a fine grid. Also, $l2$ -error norm in a fine grid is closed to zero when length of step or time step is very small, and the $l2$ -error norm results are the best results.

5.4 Comparison $l2$ -relative error norm using four choices of restriction operator for several cases

For all examples, three models are found: a coarse model, a fine model and nested model (contain coarse and fine grids). **The evaluation of the model is performed by calculating the averaged relative error at all grid points in the model domain with the use of the following equation:**

$$RE_{i,j}^t = \frac{\sum_{i=1}^{i=N} \sum_{j=1}^{j=M} |\Phi_{i,j}^t - \phi_{i,j}^t|}{\sum_{i=1}^{i=N} \sum_{j=1}^{j=M} |\phi_{i,j}^t|} \quad \text{for } i = 1, 2, 3, \dots, N \quad \text{for } j = 1, 2, 3, \dots, M$$

The absolute error (ABSE), and ($l2$ -RE) between the coarse grid and the fine grid are calculated as:

$$ABSE_{i,j}^t = \sum_{i=1}^{i=N} \sum_{j=1}^{j=M} |\Phi_{i,j}^t - \phi_{i,j}^t| \quad \text{for } i = 1, 2, 3, \dots, N \quad \text{for } j = 1, 2, 3, \dots, M$$

and

$$l2 - RE_{i,j}^n = \sqrt{\sum_j^M \sum_i^N \left[\left(\frac{\Phi_{i,j}^n - \phi_{i,j}^n}{\phi_{i,j}^n} \right)^2 \right]}$$

Where $l2$ -RE means global $l2$ -relative error norm in space, N and M means the total of cells, i, j corresponds the index of the coarse grids, $n = \sum_{t=1}^{t=r}$, where r corresponds the number of total time, Φ is the coarse grid variable and ϕ is the fine grid variable.

Example 1: When both space and temporal refinement ratio are 1:3

In this example, we use EFDm in space and leapfrog with Robert-Asselin filter in time to approximate 2DSWEs given in equations (2.32)-(2.34) for linear case with initial condition $u=v=\eta=0$.

1. Numerical parameters:

The computational domain is discretized by a grid whose size is regular. Numerical values of the parameters are chosen as follows: If we take different values of time $t = 20, 30, \dots, 100$ days to find $l2$ -RE of free surface elevation between the coarse grid and the fine grid, when $L_x = L_y = 300$, $\Delta x = \Delta y = 3$, time step = 0.01s, $h = 10$, and the number of grids 100×100 in a coarse grid, $\Delta x = \Delta y = 1$, time step = 0.0033s

when consider only the common points to both grids, the number of fine grids becomes 100×100 with CFL condition 0.02.

2. Results and Discussion

The following figure shows l_2 -RE of free surface elevation between a coarse grid and a fine grid in two-way nested grid.

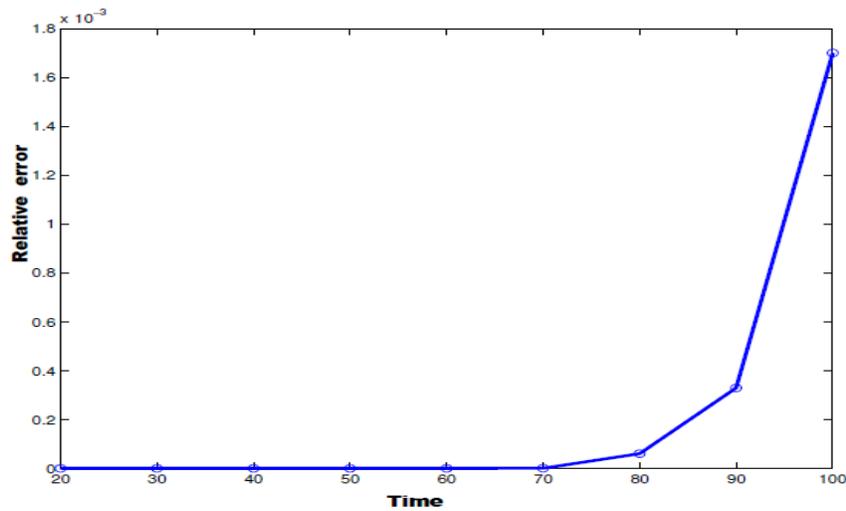


Figure 5-24: l_2 -RE of free surface elevation between a coarse grid and a fine grid in two-way nested grid.

The following figure compares l_2 -RE of free surface elevation between a coarse grid and a fine grid in two-way nested grid when using four choices of restriction operator for the free surface elevation η with a separate interface.

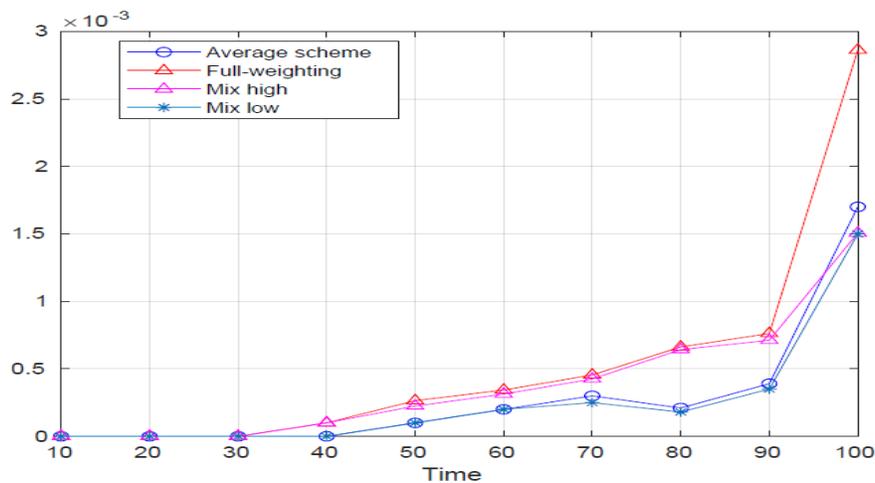


Figure 5-25: Comparison l_2 -RE between a coarse grid and a fine grid when use four choices of restriction operator for the free surface elevation

The following figure compares l_2 -RE of free surface elevation between a coarse grid and a fine grid

in two-way nested grid when use four choices of restriction operator for free surface elevation η without a separate interface.

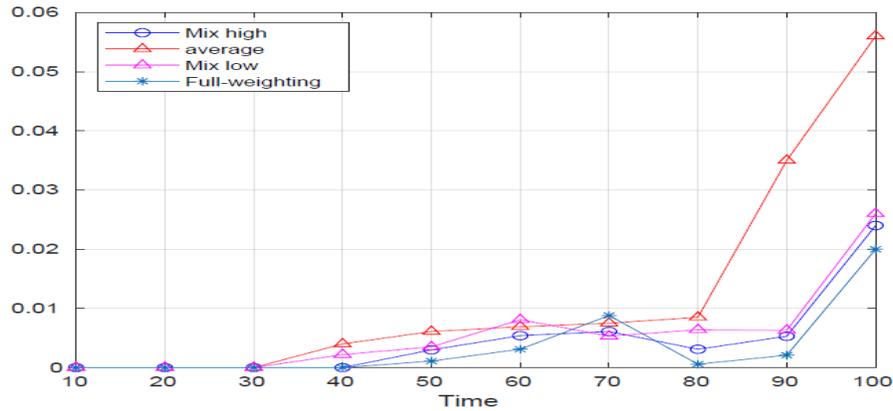


Figure 5-26: Comparison l_2 - RE between a coarse grid and a fine grid when use four choices of restriction operator for free surface elevation

Example 2: When the space refinement ratio is 1:3 and temporal refinement ratio is 1:2

By the same previous example, we use EFDM in space and leapfrog with Robert-Asselin filter in time to approximate 2DSWEs given in equations (2.32)-(2.34) for linear case.

2. Results and Discussion

The following figure compares l_2 - RE of free surface elevation between a coarse grid and a fine grid in two-way nested grid when using four choices of restriction operator for free surface elevation η with a separate interface.

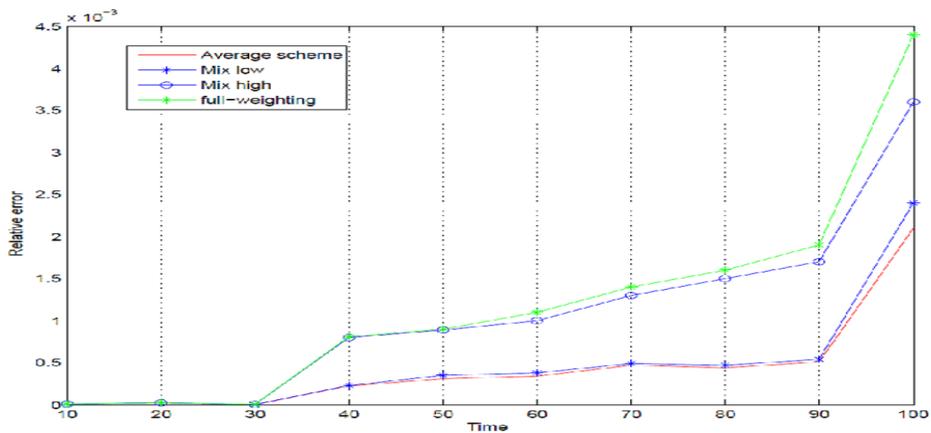


Figure 5-27: Comparison l_2 - RE between a coarse grid and a fine grid when use four choices of restriction operator for free surface elevation

Example 3: When the space refinement ratio is 1:3 (no temporal refinement)

By the same previous example, we use EFDM in space and leapfrog with Robert-Asselin filter in time to approximate 2DSWEs given in equations (2.32)-(2.34) for linear case.

2. Results and Discussion

The following figure compares l_2 -RE of free surface elevation between a coarse grid and a fine grid in two-way nested grid when using four choices of restriction operator for free surface elevation η with a separate interface.

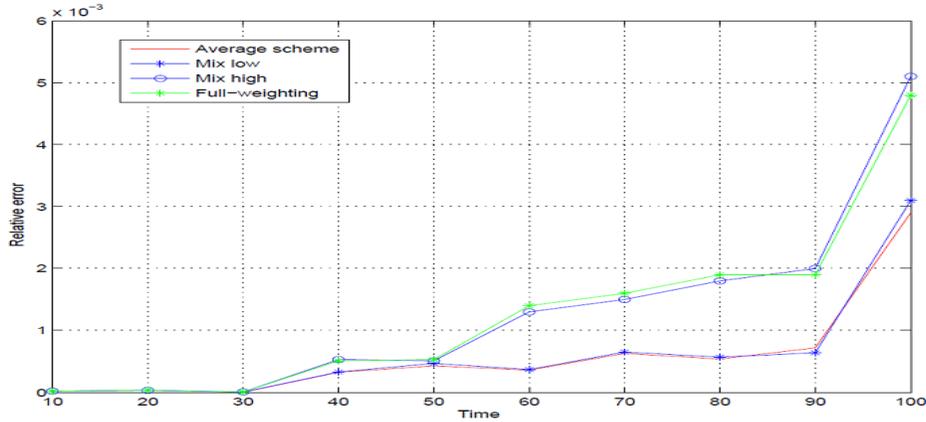


Figure 5-28: Comparison l_2 -RE between a coarse grid and a fine grid when use four choices of restriction operator for free surface elevation

5.5 Comparison between one-way and two-way nesting grids when both space and temporal refinement ratio are 1:3

Example 1: When the type of structured grid is separable

In this example, we use EFDM in space and leapfrog scheme with Robert-Asselin filter in time to approximate 2DSWEs given in equations (2.32)-(2.34) for linear case with the initial condition $u=v=\eta=0$.

Numerical parameters, results and discussion

The computational domain is discretized by a grid whose size is regular. Numerical values of the parameters are chosen as follows: If we take different values of time $t=20, 30, \dots, 100$ days to find ABSE and l_2 -RE of free surface elevation in one-way nesting and two-way nesting grid for linear 2DSWEs, when $n_x=n_y=100$, $\Delta x=\Delta y=3$ in a coarse grid and the time step in coarse grid is 0.005s, and $\Delta x=\Delta y=1$ in a fine grids when consider only the common points to both grids, the number of fine grids becomes 100×100 with CFL 0.02.

The following figures show the ABSE and l_2 -RE of free surface elevation in one-way and two-way nesting grid.

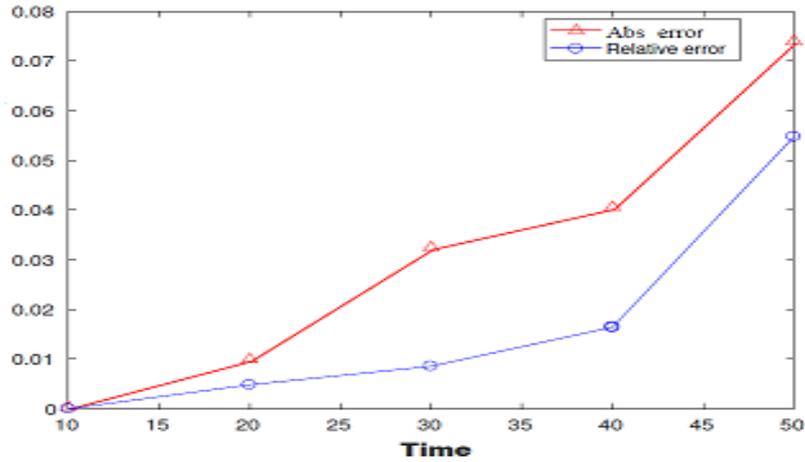


Figure 5-29: Comparison between ABSE and l_2 -RE of free surface elevation in one-way nesting grid when structured grid with a separate interface

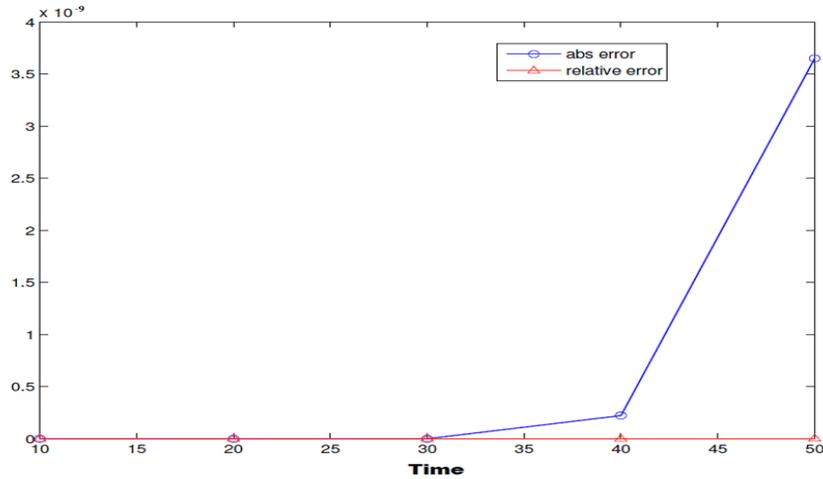


Figure 5-30: Comparison between ABSE and l_2 -RE of free surface elevation in two-way nesting grid when structured grid with a separate interface

The following figures compare l_2 -RE of free surface elevation in two-way nesting when use **four update choice** for structured grid with a separate interface.

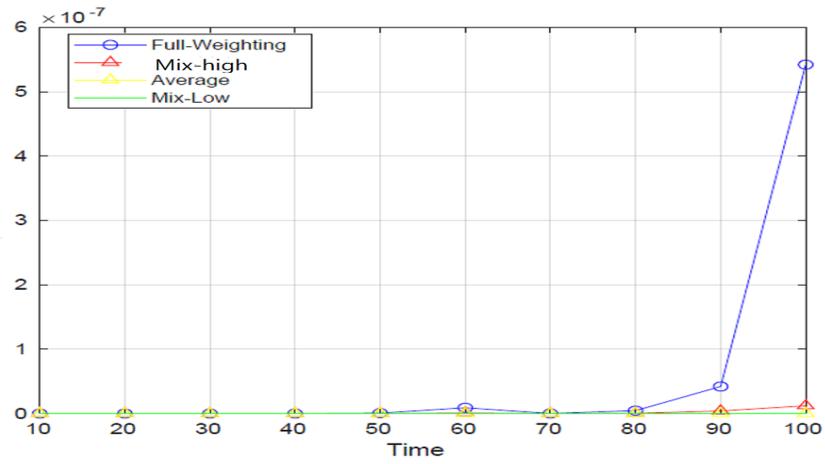


Figure 5-31: Comparison l_2 -RE of free surface elevation in two-way nesting for structured grid with a separate interface

5.6 Summary and Conclusion

This chapter dealt with the two-way interaction technique between a coarse grid and a fine grid for 2D depth-averaged linear SWEs and a new approach was presented to treat this problem. To verify the efficiency of the nested grid model, some numerical examples were introduced with nesting 3:1.

It was showed that two-way nesting techniques perform better than one-way nesting techniques. In particular, a two-way nesting ensures dynamical consistency between the coarse grid and the fine grid occurs frequently. In general, good results were observed.

In this chapter, we discussed several examples of different cases for space and temporal refinement ratio, some major points can be deduced:

1. When the space and temporal refinement ratio is odd always use for feedback Dirichlet feedback (copy grid) because a coarse grid always has one underlying high-resolution point. Otherwise, use the full-weighting or the average scheme to update the velocities and free surface elevation (both spatially and temporally) for structured grids with (adjacent/separable) interface.
2. For other cases, when the space refinement ratio is odd and the time refinement ratio is even, or the time step in the fine grid is an equal time step in the coarse grid when the refinement ratio is odd, we use linear interpolation and for feedback use the full-weighting method or the average method.
3. When comparing the results of *l2-RE* for some examples in Section 5.4, we found: Firstly, when the space and the temporal refinement ratio is odd, the results here are the best. Secondly, the results in case the temporal refinement is even and space refinement is odd. Finally, the results in case the time step in the fine and coarse grid is fixed (no time refinement, refinement in space only).

Implementation and Validation of 2D Shallow Water Models

Chapter 6

Applications of Two-Way Interaction Technique For Multiple Nested Grids

Chapter 6

Applications of Two-Way Interaction Technique For Multiple Nested Grids

The results presented in this chapter (Sections 6.3-6.5) are the subject of an article [5] and the results presented in (Sections 6.9) are the subject of an article [7]

Multiple nested grids can be employed at one time to save the time as well as get enough resolution in the goal region. In this chapter, an explicit finite difference method which used to construct a two-way interaction technique is applied successfully for multiple nested grids of 2DSWEs. The nesting implementation allows several nesting level and several grids at any particular level using Dirichlet boundary conditions. Generally, high accuracy is given for SWEs.

Discusses some examples when the space refinement ratio is 1:3 for multiple nested grids at multiple levels for 2DSWEs using the algorithms and techniques provided under Chapter 4, Section 4.5. In order to verify the performance of nesting techniques, apply some examples of coupling 3 systems for shallow flow models.

Thus, the results of l_2 -RE using a new technique for four options to update restriction operators are compared. The results are best when using the full-weighting method which it has excellent properties regarding the filtering for a type of structured grid without a separable interface. Otherwise, the average method is used to update the system.

Comparison of l_2 -RE results of free surface elevation in one-way and two-way nesting grids using four update interpolations. Several tests of numerical examples are presented to get the approximate solutions for free surface elevation using four different update schemes. Finally, comparison the results of l_2 -RE when the space refinement ratio is 1:3 and the temporal refinement ratio is 1:2 with $\Delta x \neq \Delta y$.

Highlights

- An explicit methods which used to construct a two-way interaction technique are applied successfully for multiple nested grids of 2DSWEs.
- Discusses some examples when the space refinement ratio is 1:3 for multiple nested grids at multiple levels for 2DSWEs.
- Two-way nested coupling of 3 models for multiple grids at multiple levels are achieved for 2DSWEs.
- High accuracy results for 2DSWEs are obtained using some kinds of higher-restriction operators which are average method and full-weighting, update mix-low, and update mix-high.
- The validity of the nesting method is shown in some examples.

Contents

6.1	Development	187
6.2	Configuration of the numerical testing	187
6.2.1	Basic design of nested model	187
6.2.2	Grid comparison	187
6.3	Case 1: Multiple nested grids for linear 2DSWEs for structured grids	189
6.4	Case 2: Multiple nested grids for 2DNSWEs for structured grids	192
6.5	Coupling systems for 2DNSWEs for structured grid without a separate interface	196
6.6	Comparison of one-way and two-way nesting by using different schemes	199
6.7	Comparison between one-way and two-way nesting When the time step in a coarse grid is the same time step in a fine grid and the space refinement ratio is 1:3	202
6.7.1	Case 1: For 2D depth-averaged linear shallow water equation	202
6.7.2	Case 2: For 2D depth-averaged nonlinear shallow water equation	203
6.8	The space refinement ratio is 1:3 and temporal refinement ratio is 1:2 when $\Delta x \neq \Delta y$	204
6.9	High accuracy results for 2DSWEs With (separate/ adjacent) structured grids	206
6.9.1	$L2-RE$ for two-way nested grid by using four update interpolations when both space and temporal refinement ratio are 1:3	206
6.10	Multiply nested techniques for 2DSWEs when the type of structured grids (adjacent/separate)	211
6.11	Summary and Conclusions	214

6.1 Development

In this chapter, we will explain that using the higher-order restriction operators (feedback) given in Section 4.5 can lead to strong improvements to the results, which is required to prevent aliasing and noise on the coarse grids with Dirichlet boundary conditions to approximate 2DSWEs for structured grids with (separate/adjacent) interface.

6.2 Configuration of the numerical testing

In this section: Firstly, we offer design for 2D depth-averaged NSWs using EFDM and leapfrog method with Robert-Asselin filter. Secondly, applying some examples of two-way interaction technique and show that this technique works efficiently under different conditions.

6.2.1 Basic design of nested model

The main design elements of the model are specified as follows:

- It should consist of two similar models a parent grid (coarse grid) model and a child grid (fine grid) model.
- Both models should calculate the same variables by solving the same formulations of the governing equations.
- The parent grid should provide boundary data for the child grid.
- The models should be dynamically linked to allow automatic transfer of boundary data at each child model time step.
- The child model should be able to operate on a number of nested domains not just one level of nesting. i.e. all nested domains should not use the same nesting ratio (sometimes use different nesting ratio in the same level or multiple-levels).

6.2.2 Grid comparison

Key features of the two model grids are listed below for comparison:

1. Coarse grid
2. Fine grid
3. Grid generation: The fine grid coordinates are interpolated from the coarse grid with one third (or other) of the coarse grid cell size.

4. Water depth Setup: The fine grid water depths are linearly interpolated from the coarse grid.
5. Nested grid model initialization: The fine grid is initialized with parameters (velocity flux and free surface) interpolated from the coarse grid when the fine grid model starts.
6. Interaction between coarse grid and fine grid:
7. Time stepping: The coarse grid model's time step is N times the fine grid model's time step.
8. Boundary forcing: The boundary conditions in the fine grid are interpolated from coarse grid cells at each time step.
9. The Model: The coarse grid model and the fine grid model starts at the same time.

Some Notes:

1. In all examples, we apply an explicit center finite difference method in space and leapfrog method with Robert-Asselin filter in time to approximate the 2D depth-averaged NSWEs.
2. In all examples, we find *l2-RE* of free surface elevation for two-way interaction techniques 2DSWEs.
3. All simulations are made using open boundary conditions, we use a nested grid with interface condition for the fine grids linear interpolation both spatially and temporally and update interface condition for the coarse grid using the average method or full-weighting method (both spatially and temporally) using initial condition $u=v=\eta=0$.
4. The bottom friction comes from Manning's formula which is uniform throughout the grids, where n is roughness coefficient. In this simulation, n takes 0.013.

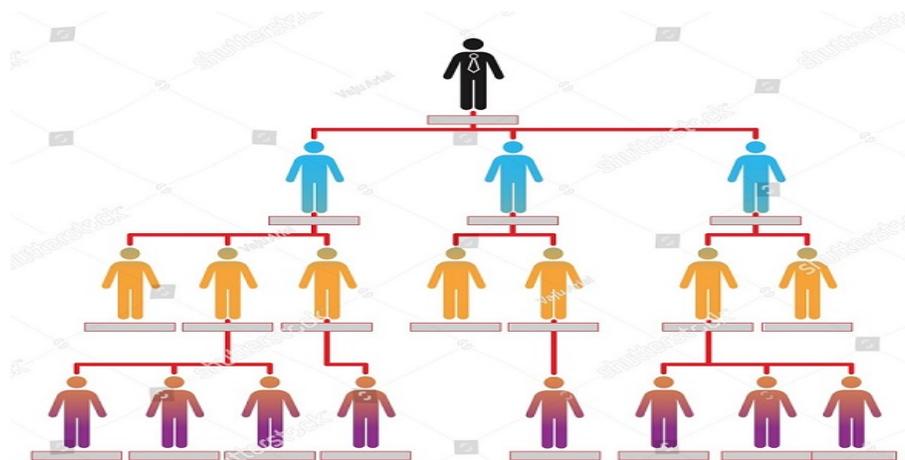


Figure 6-1: Multiple nested grids at multiple levels

6.3 Case 1: Multiple nested grids for linear 2DSWEs for structured grids (without a separate interface)

Here, the space refinement ratio is 1:3 and the temporal refinement ratio is 1:2 and using the average method to update interface condition for the coarse grid. also, the number of grids for the coarse and fine grids are equal by consider only the common points to both grids using algorithm 1, Chapter 4. For the stability using CFL condition given by Section 2.8.3, Chapter 3.

Example 1:

In this example, we use the system for 2DSWEs given in equations (2.32)-(2.34) for a linear case to find *l2-RE* of free surface elevation when a coarse grid contains only one fine grid (child embedded in parent) at different times $t=500, 1000, \dots, 4000$ hours with CFL condition 0.6. The information about a coarse grid (level 1) and a fine grid (level 2) are given in Table 6.1 using Dirichlet open boundary conditions.

The following figure shows *l2-RE* of free surface elevation between a coarse grid and a fine grid.

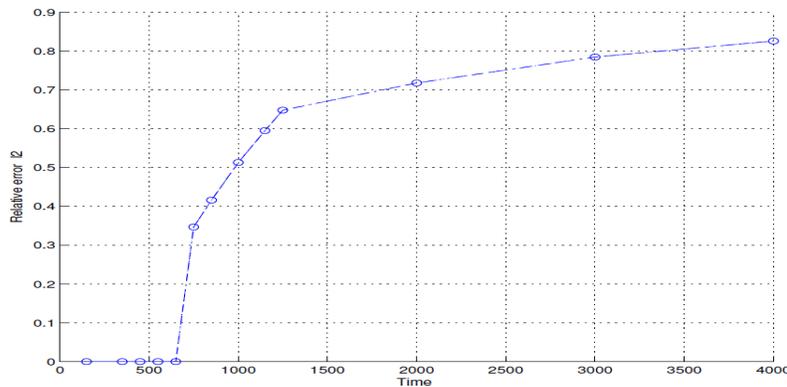


Figure 6-2: Shows *l2-RE* of free surface elevation between a coarse grid and a in fine grid (grid 21)

Information	Grid 01 (coarse grid)	Grid 21 (fine grid)
Number of grids	150×150	150×150
Length grid size	3	1
Coarse grid	non	Grid 01
Grid size ratio	non	3
Time step in sec	0.025	0.0125
(East-West)	1-150	51-100
(South-North)	1-150	1-50
Bottom stress	0.013	0.013

Table 6.1: The information on the set up of the different grids for 2D non-linear SWEs

Example 2:

In this example, we use the system of 2DSWEs given in equations (2.32)-(2.34) for linear case to find l_2 -RE in case a coarse grid contains more than one fine grid at different times $t=500, 1000, \dots, 4000$ hour with CFL condition 0.6. The information about a coarse grid and a fine grid are given in Table 6.2 using Dirichlet open boundary conditions. The first figure shows l_2 -RE of free surface evaluation between a coarse grid in level 1 and a fine grid in level 2 (grid 22) and the second figure compares l_2 -RE of free surface elevation between a coarse grid and two fine grids (grids 21 and 22).

Information	Grid 01 (coarse grid)	Grid 21 (fine grid)	Grid 22 (fine grid)
Number of grids	150×150	150×150	150×150
Length grid size	3	1	1
Coarse grid	non	Grid 01	Grid 01
Grid size ratio	non	3	3
Time step in sec	0.025	0.0125	0.0125
(East-West)	1-150	51-100	101-150
(South-North)	1-150	1-50	101-150

Table 6.2: The information about the coarse and fine grids for linear 2DSWEs

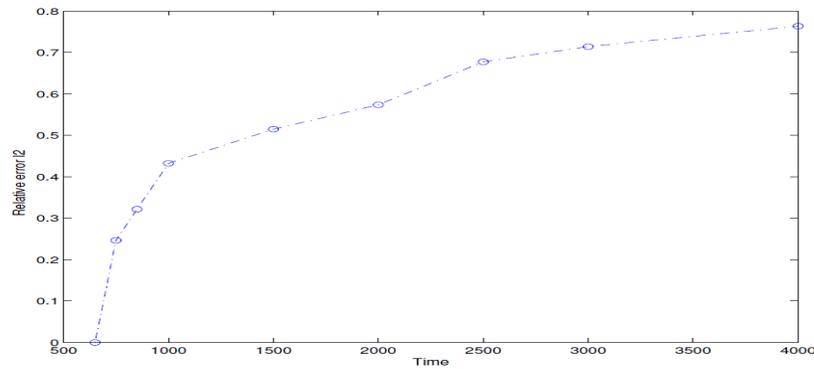


Figure 6-3: Show l_2 -RE of free surface elevation between a coarse grid and a fine grid (grid 22)

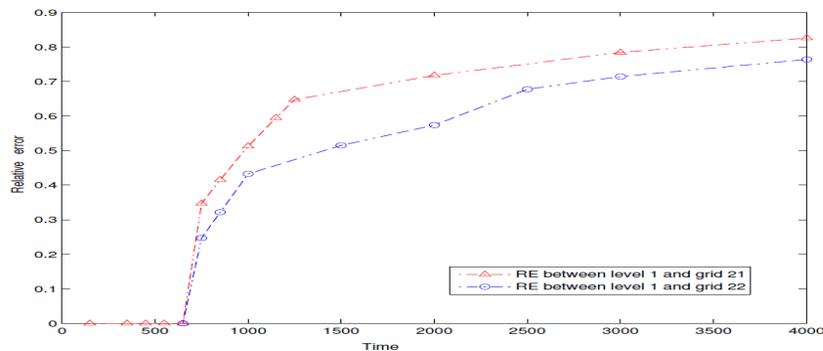


Figure 6-4: Comparison l_2 -RE between a coarse grid and two fine grids in level 2

Notes that: Two fine grids are separate with each other and embedding in a coarse grid.

Example 3:

In this example, we use the system of 2DSWEs given in equations (2.32)-(2.34) for a linear case to find $l2-RE$ in case a fine grid contains again one fine grid in another level at different times $t=500, 1000, \dots, 4000$ hour. The information about a coarse grid and a fine grid are given in Table 6.3 using Dirichlet open boundary conditions.

Information	Grid 01 (coarse grid)	Grid 21 (fine grid)	Grid 31 (level 3)
Number of grids	150×150	150×150	150×150
Length grid size	3	1	0.33
Coarse grid	non	Grid 01	Grid 21
Time step in sec	0.025	0.0125	0.00625
East-West	1-150	51-100	1-50
South-North	1-150	1-50	1-50

Table 6.3: The information about the coarse and the fine grids at multiple levels for linear 2DSWEs

The first figure shows $l2-RE$ of free surface elevation between a fine grid in level 2 and a fine grid in level 3 in two-way nested grid and the second figure comparison $l2-RE$ of free surface elevation between a coarse grid-fine grid (grid 21) and grid 21-fine grid (grid 31) in two-way nested grid.

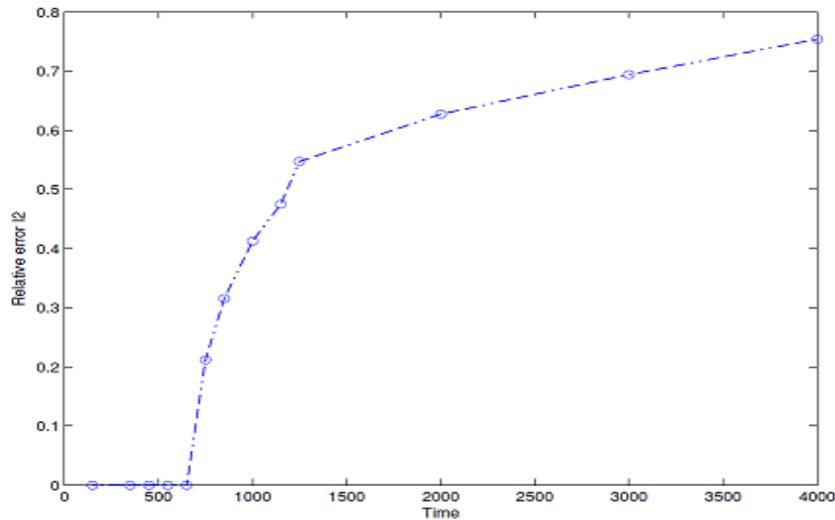


Figure 6-5: $l2-RE$ between the grid 21 and the grid 31 in two-way nested grid

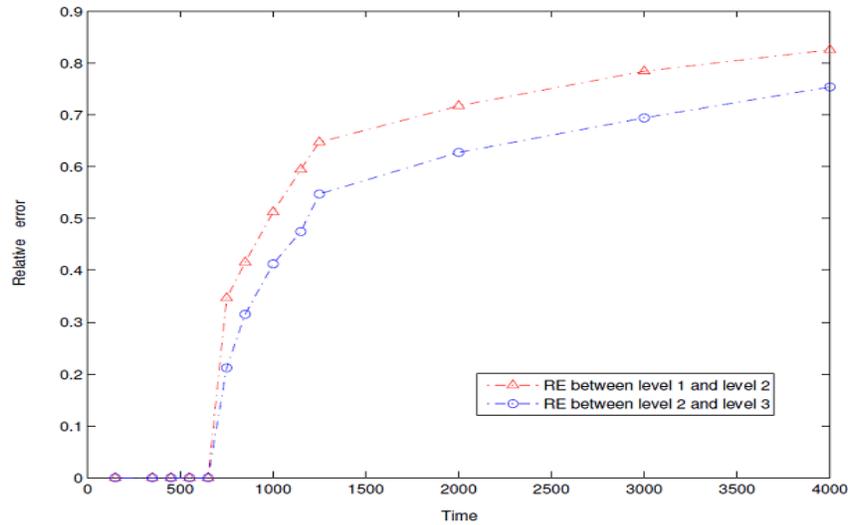


Figure 6-6: Comparison l_2 - RE between a coarse grid-grid 21 and grid 21-grid 31

Notes that:

The grid 31 is separate from the grid 21, we can choose another grid embedded in the grid 21 and comparing results in two cases. The best results were obtained when the grids are separate from each other (see Section 8.4, Example 3).

6.4 Case 2 : Multiple nested grids for NSWEs for structured grids (without a separate interface)

Here, the space refinement ratio is 1:3 and the temporal refinement ratio is 1:2 and using the average method to update interface condition for the coarse grid. also, the number of grids for the coarse and fine grids are equal by consider only the common points to both grids using algorithm 1, Chapter 4.

Example 1:

In this example, we use EFDM to approximate 2DNSWEs given in equations (2.32)-(2.34) (with $\nu = 0$, wind stress =0, non-rotated $f=0$). A sequence of snapshots of free surface elevation η in the coarse grids at different times $t=500, 1000, \dots, 4000$ hour. The information about the coarse and the fine grids are given by Table 6.1.

The following figure represent the simulation of free surface elevation in a coarse grid at different time $t=500, 1000, \dots, 2000$ hour.

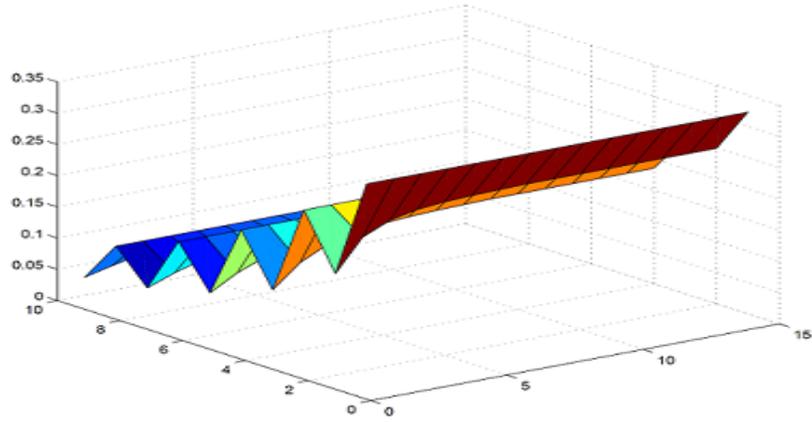


Figure 6-7: Simulation of free surface elevation in the coarse grid at time= 500 min

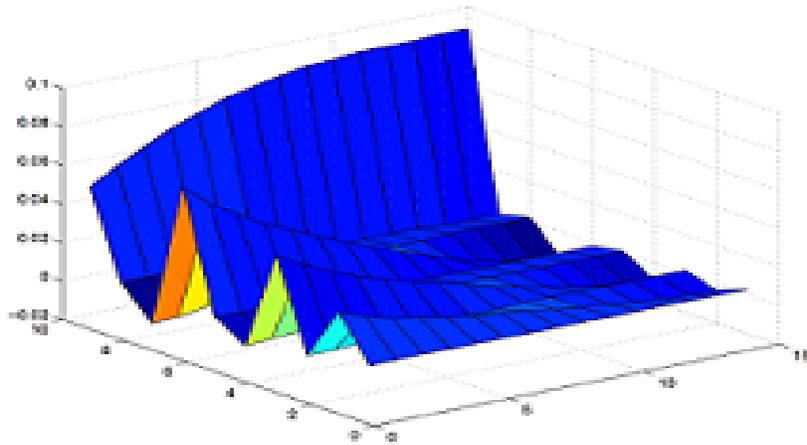


Figure 6-8: Simulation of free surface elevation in the coarse grid at time= 1000 min

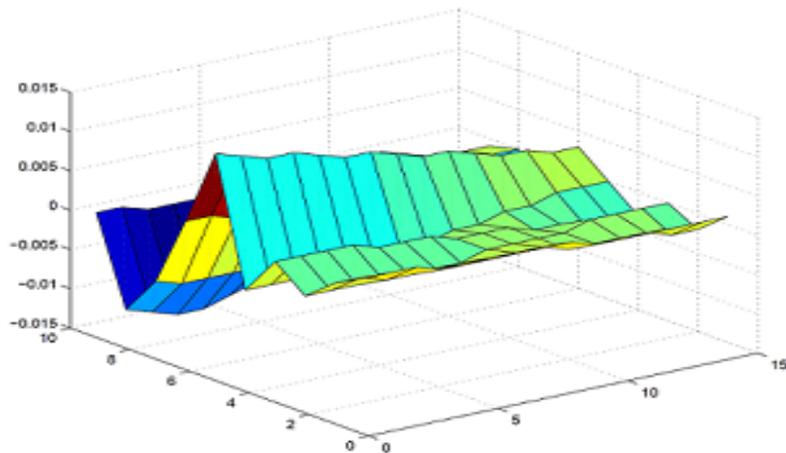


Figure 6-9: Simulation of free surface elevation in the coarse grid at time= 2000 min

Example 2:

By the same previous example, we use 2DNSWEs given in equations (2.32)-(2.34) to find l_2 -RE of free surface elevation at difference times $t=500, 1000, \dots, 4000$ hour when a coarse grid contains only one fine grid. The information about the coarse grid and the fine grid are given by Table 6.1. The following figure represent l_2 -RE of free surface elevation in two-way nested grid at $t=500$ min, 1000, ..., 4000 hour.

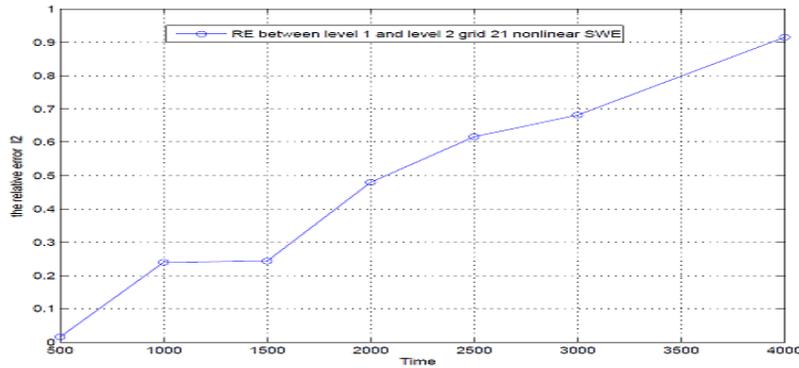


Figure 6-10: Comparison l_2 -RE of free surface in two-way nested grid at $t=500, 1000, \dots, 4000$ hour

Example 3: Comparison RE_{l_2} between the coarse grid and two fine grids

In this example, we use 2DNSWEs given in equations (2.32)-(2.34) to find l_2 -RE when a coarse grid contains more than one fine grid at the different times $t=500, 1000, \dots, 4000$ hour using Dirichlet open boundary conditions. The information about the coarse and the fine grids are given by Table 6.2.

The following figure compares l_2 -RE of free surface elevation between the coarse grid and two fine grids in two-way nested grid at the same level (level 2) at $t=500, 1000, \dots, 4000$ hour.

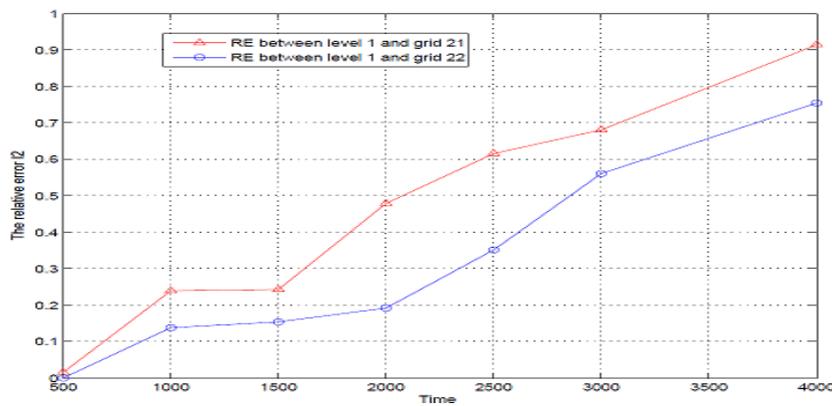


Figure 6-11: Comparison l_2 -RE of free surface elevation between the coarse grid and two fine grids in level 2

Example 4 :

In this example, we find *l2-RE* for 2DNSWEs given in equations (2.32)-(2.34) when a fine grid in level 2 contains again one fine grid in level 3 at difference times $t= 500, 1000, \dots, 4000$ hour. The information about the coarse and the fine grids are given by Table 6.3 .

The following figure shows *l2-RE* of free surface elevation between level 2 (grid 21) and level 3 (grid 31) in two-way nested grid at $t= 500, 1000, \dots, 4000$ hour.

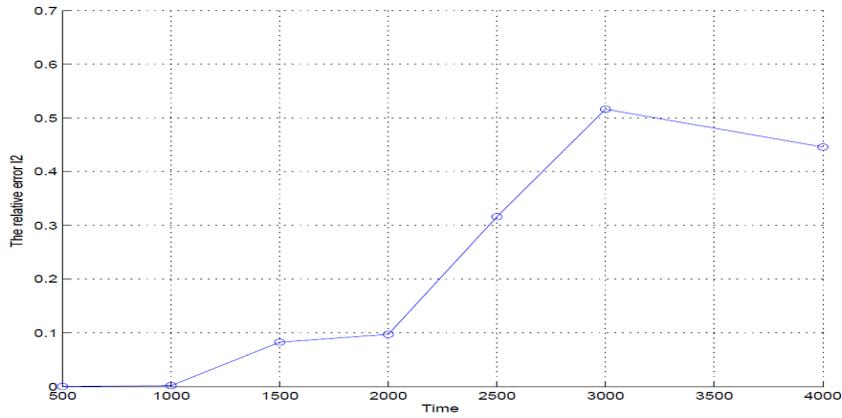


Figure 6-12: Shows *l2-RE* of free surface elevation between the grid 21 and the grid 31

Example 5:

In this example, we find *l2-RE* for 2DNSWEs given in equations (2.32)-(2.34) when the second fine grid in level 2 contains again one fine grid in level 3 at the different times $t= 500, 1000, \dots, 4000$ hour. The following figures compare *l2-RE* of free surface elevation between a coarse grid-level 2 (grid 21), fine grid 21-level 3 (grid 31) and a coarse grid-level 2 (grid 22), fine grid 22-level 3 (grid 32) in two-way nested grid.

Information	Grid 01 (coarse grid)	Grid 22 (fine grid)	Grid 32 (level 3)
Number of grids	150×150	150×150	150×150
Length grid size	3	1	0.33
Coarse grid	non	Grid 01	Grid 22
Grid size ratio	non	3	3
Time step in sec	0.025	0.0125	0.00625
East-West	1-150	101-150	101-150
South- North	1-150	101-150	1-50

Table 6.4: The information about the coarse and the fine grids at multiple levels for 2DNSWEs

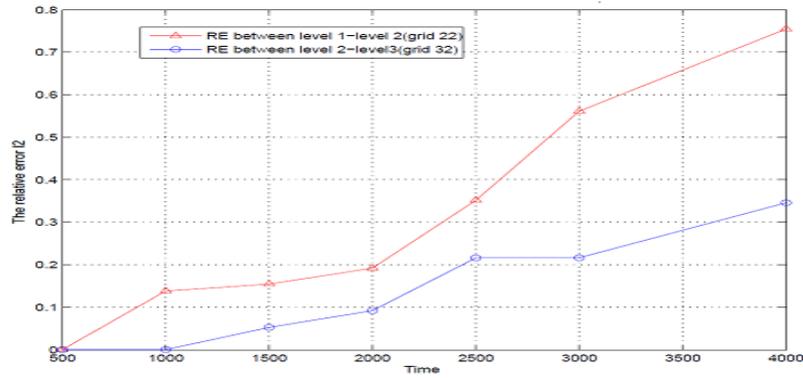


Figure 6-13: Compares l_2 -RE between a coarse grid-fine grid 22 (level 2) and fine grid 22-fine grid 32 in level 3

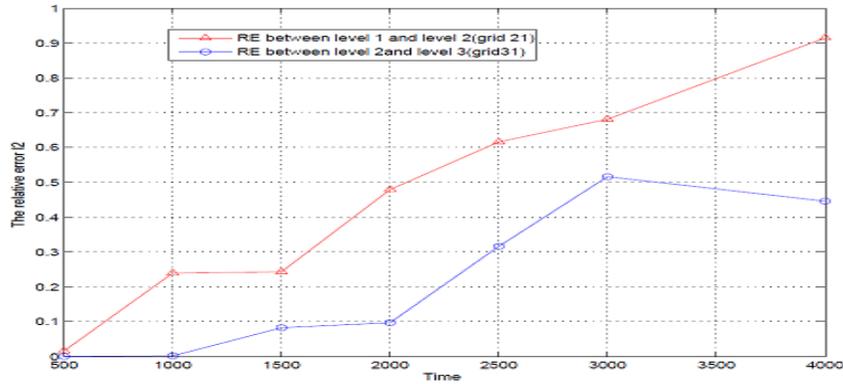


Figure 6-14: Compares l_2 -RE between a coarse grid-fine grid 21 in level 2 and fine grid 21-fine grid 31 in level 3

6.5 Coupling systems for 2DNSWEs for a structured grid (multiple levels) without a separate interface

Here, the space refinement ratio is 1:3 and the temporal refinement ratio is 1:2 and using the average method to update interface condition for the coarse grid. also, the number of grids for the coarse and fine grids are equal by consider only the common points to both grids using algorithm 1, Chapter 4.

Example 1: Case 1: Coupling 3 systems for 2DNSWEs

In this example, we find l_2 -RE of free surface elevation for 2DNSWEs given in equations (2.32)-(2.34), when a coarse grid contains more than one level (multiple levels) at different times $t= 500, 1000, \dots, 4000$ hour. The information about a coarse grid and the fine grids are given by Table 6.5 using Dirichlet boundary condition.

Firstly, when a coarse grid located in 1st-level, applying EFDM to approximate 2DNSWEs. Sec-

only, applying EFDM to approximate 2DNSWEs in level 2. Then, coupling a coarse grid with a fine grid in the 2nd-level. Thirdly, applying EFDM to approximate 2DNSWEs in level 3. Then, coupling between a fine grid in 2nd-level and a fine grid in the 3rd-level.

Information	Grid 01	Grid 21	Grid 22	Grid 31	Grid 32
Number of grids (Domain size)	150×150	150×150	150×150	150×150	150×150
Length grid size	3	1	1	0.33	0.33
Coarse grid	non	Grid 01	Grid 01	Grid 21	Grid 22
Grid size ratio	non	3	3	3	3
Time step in sec	0.025	0.0125	0.0125	0.00625	0.00625
SWEs	non-linear	non-linear	non-linear	non-linear	non-linear
East-West	1-150	51-100	101-151	1-50	101-150
South-North	1-150	1-50	101-151	1-50	1-50

Table 6.5: The information about the coarse grid and the fine grids at multiple levels for the 2DNSWEs

The following figure compares l_2 -RE of free surface elevation between a coarse grid-level 2 (grid 22) and grid 22-level 3 (grid 32) in two-way nested grid for 2DNSWEs

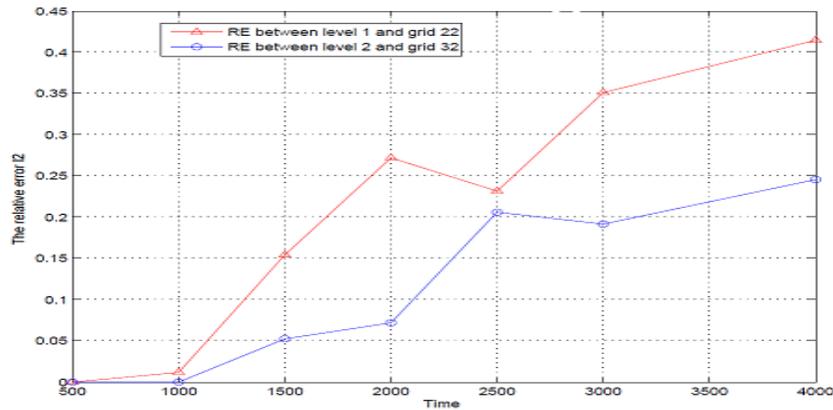


Figure 6-15: Comparison l_2 -RE between a coarse grid-grid 22 and grid 22-grid 32 for nonlinear SWEs

Example 2: Case 2: Coupling 3 systems for 2D nonlinear / linear SWEs

In this example, we find l_2 -RE for 2DNSWEs given in equations (2.32)-(2.34), when a coarse grid contains more than one level at different times $t = 500, 1000, \dots, 4000$ hour. The information about the coarse and fine grids are given by Table 6.5.

Firstly, when a coarse grid located in 1st-level, apply EFDM to approximate 2DNSWEs. Secondly, applying EFDM to approximate 2DNSWEs in level 2. Thus, coupling a coarse grid with the fine grid in 2nd-level. Thirdly, applying EFDM to approximate linear 2DSWEs in level 3. Thus, coupling between a fine grid in level 2 and the fine grids in the 3rd-level.

The following figure compares l_2 -RE between a coarse grid-level 2 (grid 22 nonlinear case) and grid 22-level 3 (grid 32 in linear case) in two-way nested grid for 2DSWEs.

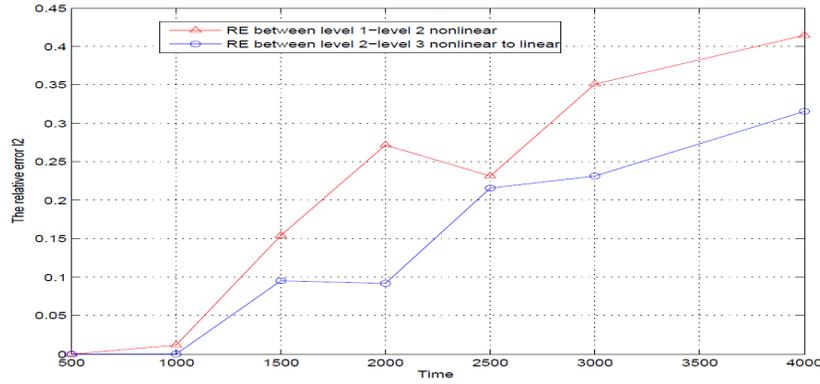


Figure 6-16: Comparison l_2 -RE of free surface elevation between a coarse grid-grid 22 for nonlinear case and grid 22-grid 32 for nonlinear to linear

Example 3: Case 3: Coupling 3 systems for 2DNSWEs

In this example, we find l_2 -RE for 2DNSWEs given in equations (2.32)-(2.34) when a coarse grid contains more than one level at different times $t=500, 1000, \dots, 4000$ hour. The information about the coarse grid and the fine grids are given by Table 6.5.

Firstly, when 1st-level contains a coarse grid, applying EFDM to approximate 2DNSWEs. Secondly, applying EFDM to approximate linear 2DSWEs in level 2. Thus, coupling a coarse grid with a fine grid in 2nd-level grid. Thirdly, applying EFDM to approximate linear 2DSWEs in level 3. Thus, a coupling between fine grid in level 2 and fine grid in level 3.

The following figure compares l_2 -RE between a coarse grid-level 2 (grid 22 in linear case) and grid 22-level 3 (grid 32 in nonlinear case) in two-way nested grid for 2DSWEs.

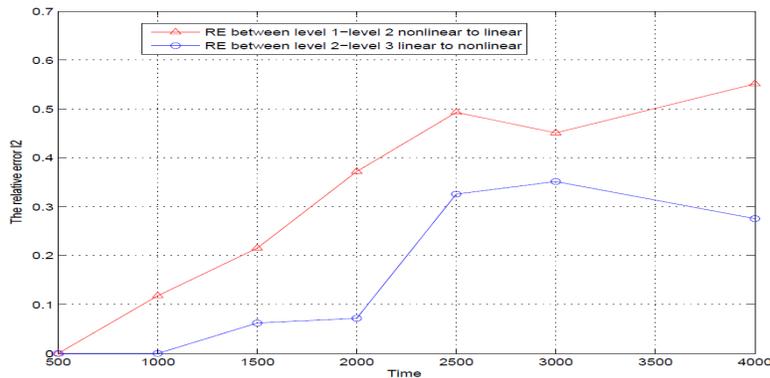


Figure 6-17: Comparison l_2 -RE between a coarse grid-a fine grid in level 2 for nonlinear to linear and a fine grid in level 2-fine grid in level 3 for linear to nonlinear SWEs

Notice that: The results in Example 1 are better than the other results in Examples 2 and 3.

6.6 Comparison of one-way and two-way nesting by using different schemes

Here, the space refinement ratio is 1:3 and time refinement ratio is 1:2 and using different scheme to update interface condition for the coarse grid. also, the number of grids for the coarse and fine grids are equal by consider only the common points to both grids using algorithm 5, Chapter 4.

Example 1 :

In this example, we use system of 2DNSWEs given in equations (2.1)-(2.3) with (non-rotated $f=0$, viscosity=0, and wind stress=0), if we take different values of time $t= 10, 20, \dots, 200$ hour, when $nx = ny = 200$, $\Delta x = \Delta y = 1$, and $\Delta t = 0.01$ s in a coarse grid with total steps 7200 and $nx = ny = 200$, $\Delta x = \Delta y = 0.33$, and $\Delta t = 0.005$ s in fine grid to find ABSE and $l2-RE$ of free surface elevation in one-way nesting and two-way nesting grids. The following figures show $l2-RE$ of free surface elevation in one-way nesting grids and two-way nesting grids using three different update schemes. All the simulation are made by using Dirichlet boundary conditions **without a separate dynamic and feedback interface** with CFL condition 0.138.

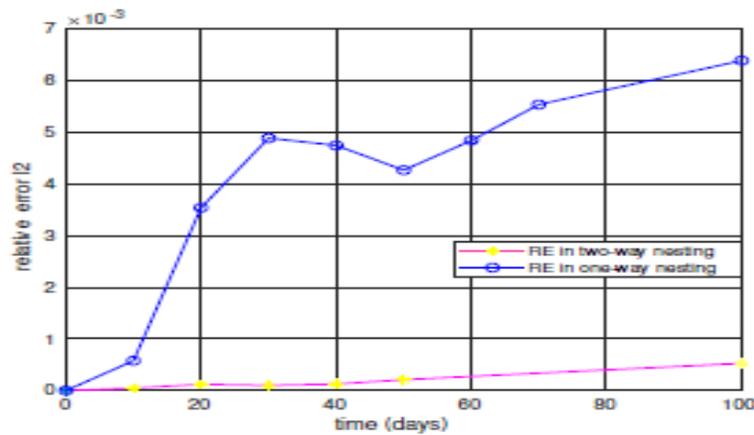


Figure 6-18: Comparison $l2-RE$ in one-way nesting grid and two-way nesting grid using the average scheme

When use Shapiro interpolation scheme, $l2-RE$ of free surface elevation in one-way nesting grid and two-way nesting grid as shown in the following figure.

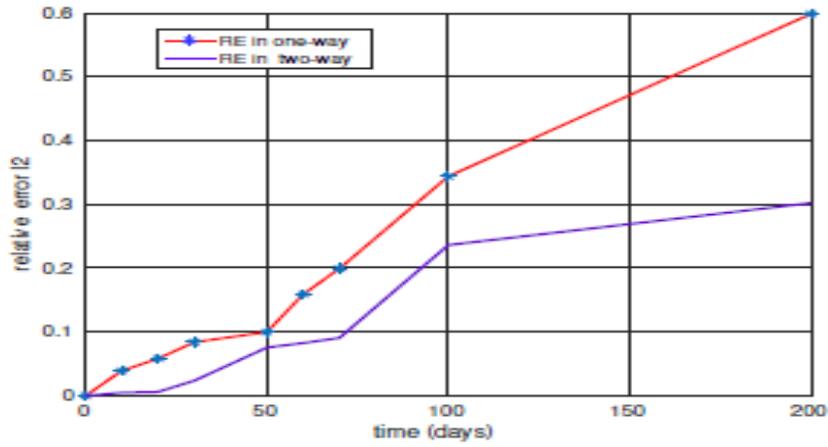


Figure 6-19: Comparison l_2 -RE of the free surface elevation in one-way nesting and two-way nesting using Shapiro interpolation scheme

When using the full-weighting interpolation scheme, l_2 -RE in one-way nesting and two-way nesting grids as shown in the following figure.

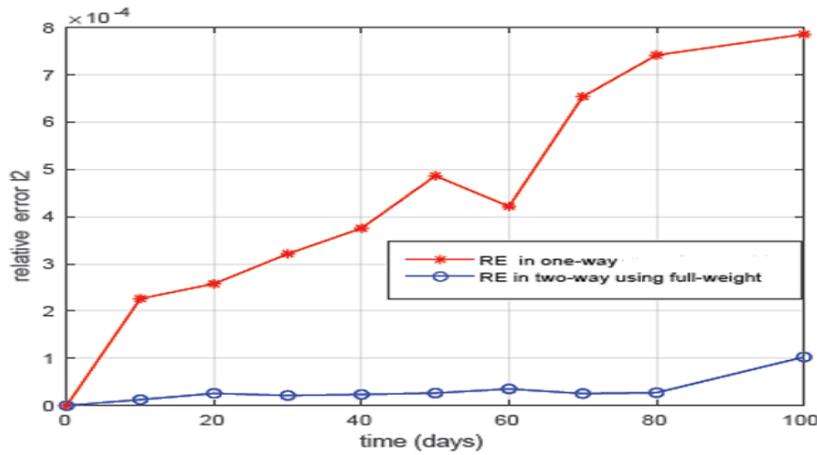


Figure 6-20: Comparison l_2 -RE in one-way and two-way nesting using the full-weighting scheme

The following figure compares l_2 -RE of free surface elevation in two-way nesting by using the average method and the full-weighting method.

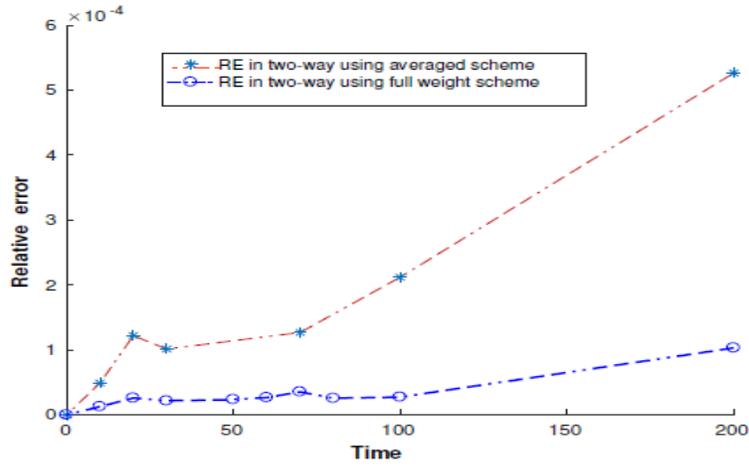


Figure 6-21: Compares l_2 -RE in two-way nesting using the average scheme and the full-weighting scheme

Example 2:

For the same previous example, we compare l_2 -RE of free surface elevation in two-way nesting grids by using the full-weighting and the average update schemes with a **separate** dynamic interface and feedback interface. All the simulation are made by using Dirichlet boundary conditions. Good results were obtained.

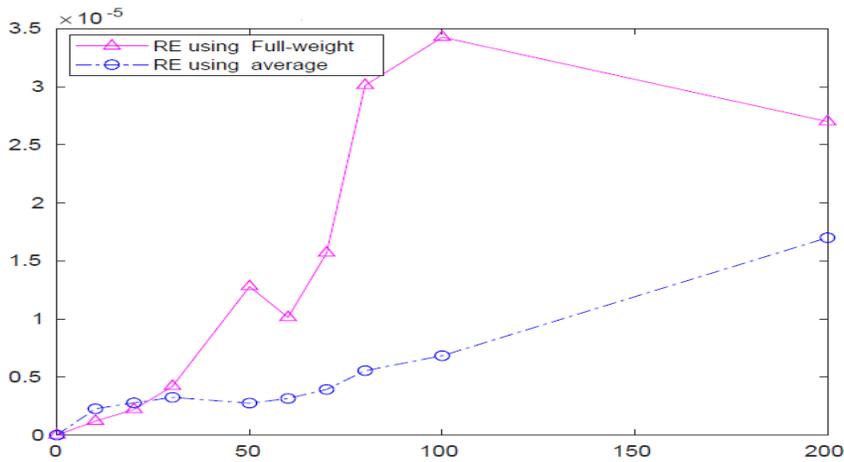


Figure 6-22: Compares l_2 -RE in two-way nesting using the average scheme and the full-weighting scheme

6.7 Comparison between one-way nesting and two-way nesting grids when space refinement ratio is 1:3 (no time refinement)

Here, the space refinement ratio is 1:3 and the number of grids for the coarse and fine grids are equal by consider only the common points to both grids.

6.7.1 Case 1: For 2D depth-averaged linear SWEs with separate dynamic interface and feedback interface

Example 1 :

In this example, we find ABSE and $l2-RE$ of free surface elevation in one-way nesting and two-way nesting grids for 2DSWEs given in equations (2.32)-(2.34) for linear case with bottom stress=0, when use reflexive boundary condition at different values of time $t= 10, 20, \dots, 50$ days, $nx=ny=300$, $\Delta x=\Delta y=3$, and $\Delta t=0.01s$ in coarse grid with total steps 43200 and $nx=ny=300$, $\Delta x=\Delta y=1$, and $\Delta t=0.01s$ in fine grid with CFL 0.04.

Where $\eta(i, j) = 10 * exp((-5((x)^2 + (y)^2))$ and $u(x, y, t = 0)=0, v(x, y, t = 0)=0$

The following figures show the ABSE and $l2-RE$ of free surface elevation in one-way nesting and two-way nesting grid.

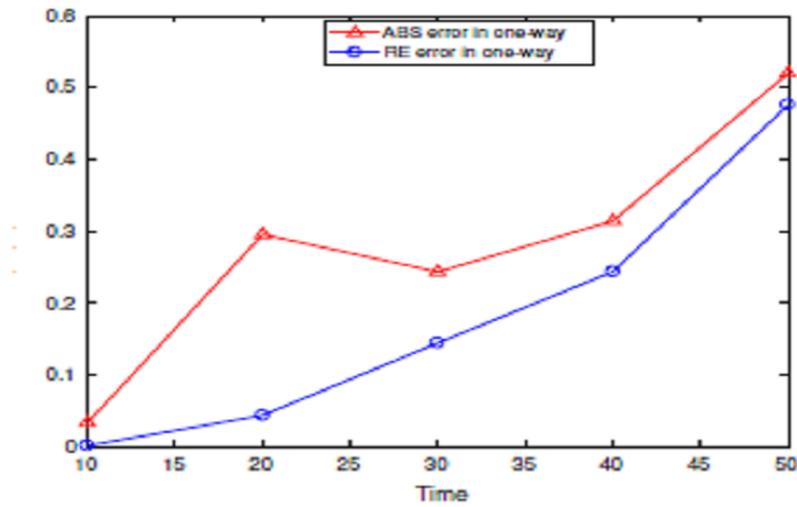


Figure 6-23: ABSE and $l2-RE$ of free surface elevation in one-way nesting

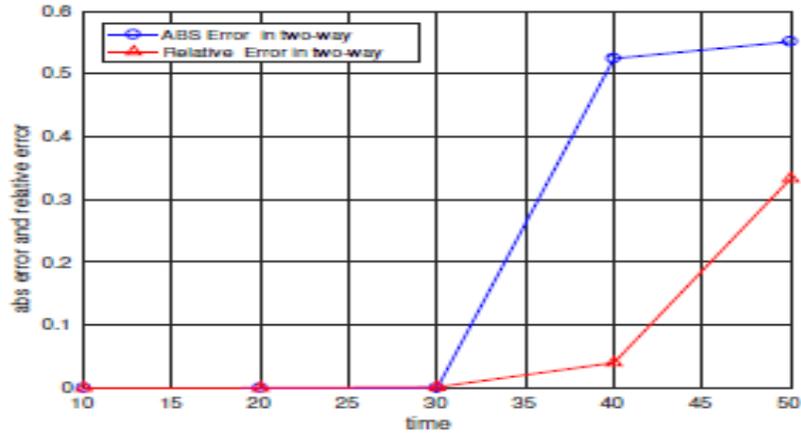


Figure 6-24: ABSE and $l2-RE$ of free surface elevation in two-way nesting

Example 2:

In this example, we find ABSE of free surface elevation in two-way nesting grid for 2DSWEs given in equations (2.32)-(2.34) for linear case with bottom stress =0, when a fine grid contains again one fine grid in another level at different values of time $t= 10, 20, \dots, 200$ days by using reflexive boundary condition when $nx=ny=100, \Delta x=\Delta y=3$, and $\Delta t=0.01s$ in a coarse grid and $nx = ny = 100, \Delta x=\Delta y=1$, and $\Delta t=0.01$ in a fine grid. The following figure compares ABSE of free surface elevation between a coarse grid and a fine grid in level 2 in two-way nested grid with CFL condition 0.02.

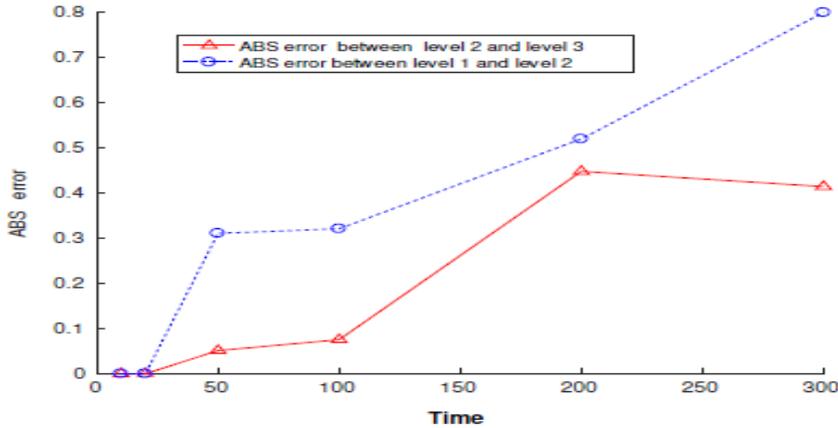


Figure 6-25: ABSE between the coarse grid and the fine grid for linear 2DSWEs

6.7.2 Case 2: For 2D depth-averaged NSWs with separate dynamic and feedback interfaces

Example 1:

In this example, we use the system of 2DNSWEs given by equations (2.1)-(2.3) with non-rotated $f=0$, wind stress, and bottom stress =0, if we take different values of time $t= 10, 20, \dots, 1000$ hours when

$nx = ny = 300$, $\Delta x = \Delta y = 1$, $\Delta t = 0.01s$ in a coarse grid and $nx = ny = 300$, $\Delta x = \Delta y = 0.33$, $\Delta t = 0.01s$ in fine grid to find ABSE and $l2-RE$ in two-way nesting. In this model, we use a multiple-nested grid with interface condition for the fine grids linear interpolation and update interface condition for the coarse grid using the average method with CFL condition 0.02.

The following figure shows ABSE and $l2-RE$ of free surface elevation in two-way nesting grids using Dirichlet boundary conditions.

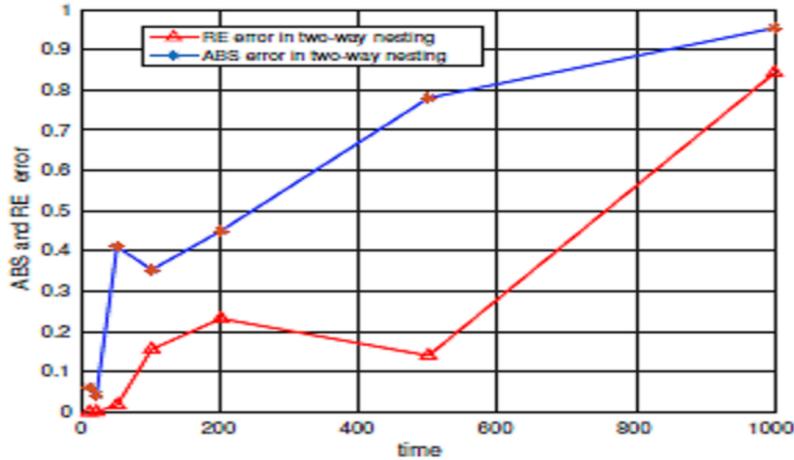


Figure 6-26: Comparison $l2-RE$ and ABSE in two-way nesting grids

6.8 The space refinement ratio is 1:3 and temporal refinement ratio is 1:2 when $\Delta x \neq \Delta y$

Example 1:

In this example, we use the system of 2D non-linear SWEs given by equations (2.1)-(2.3) with (wind stress=0, bottom stress=0, and $f=0$ in nonlinear case) and equations (2.32)-(2.34) with (bottom stress= 0) for linear case when $\Delta x \neq \Delta y$, $nx=ny=300$, $\Delta x=3$ and $\Delta y=6$, $\Delta t=0.01s$ in coarse grid with total steps 18000 and when a coarse grid has the space refinement ratio in x -direction is different in y -direction, when $\Delta x=1$ and $\Delta y=2$, $nx=ny=300$, the number of grids 300×150 , and $\Delta t=0.005s$ in the fine grid at different time $t=10, 20, \dots, 500$ hours with CFL condition 0.02. using Dirichlet boundary conditions for nonlinear case and reflexive boundary conditions for linear case **with a separate dynamic interface and feedback interface.**

The following figures represent $l2-RE$ of free surface elevation for 2D non-linear SWEs in one-way nesting and two-way nesting grids.

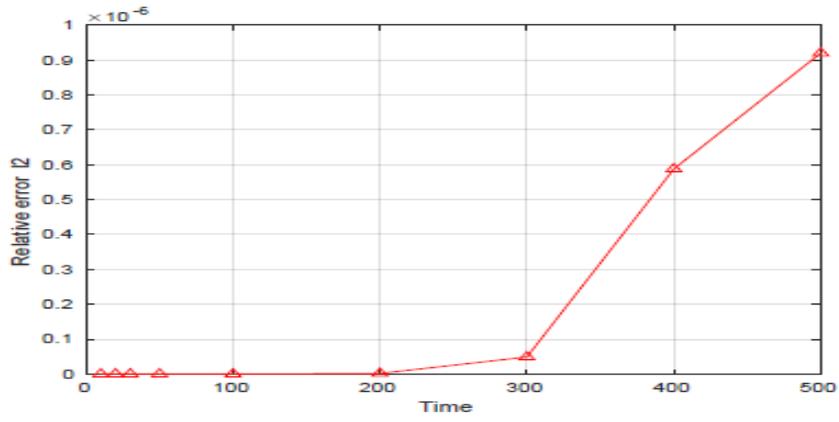


Figure 6-27: l_2 -RE of free surface elevation in one-way nesting grid for linear SWEs

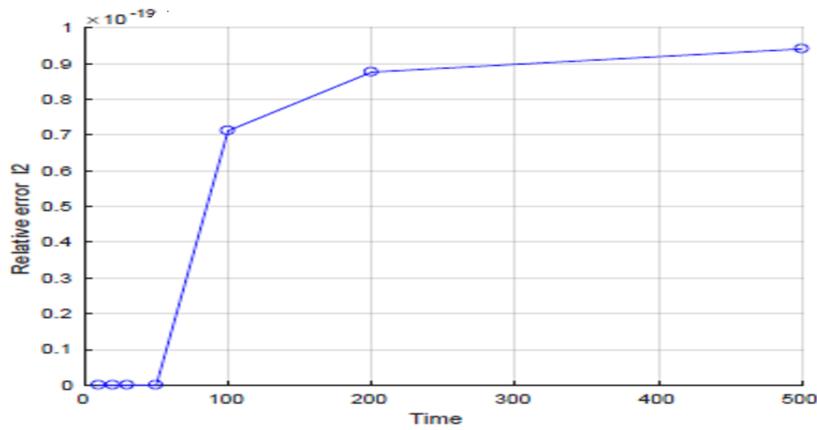


Figure 6-28: l_2 -RE of free surface elevation in two-way nesting grid for linear SWEs

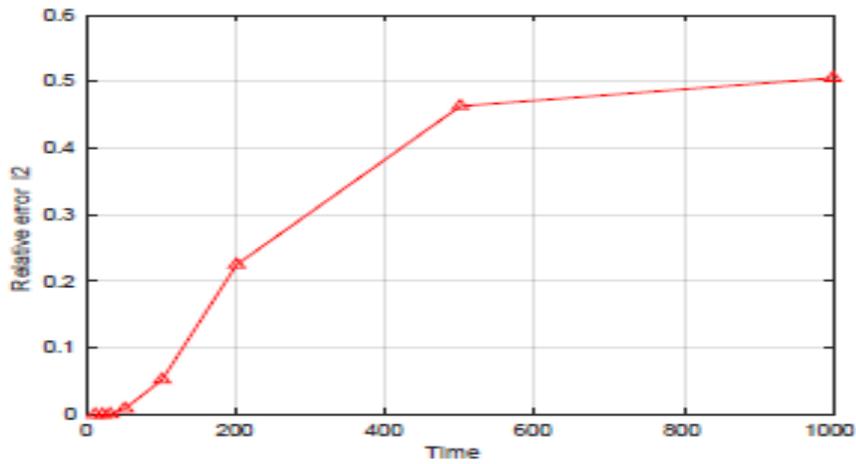


Figure 6-29: l_2 -RE of free surface elevation in one-way nesting grid for nonlinear SWEs

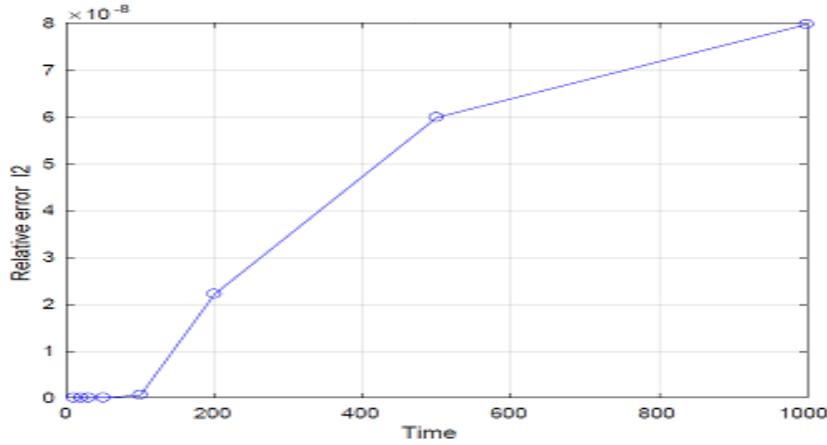


Figure 6-30: l_2 -RE of free surface elevation in two-way nesting grid for nonlinear SWEs

Notes: When comparing these results with the results given by Sections 6.7.1 and 6.7.2, the results indicate in Section 6.8 are the best.

6.9 High accuracy results for 2DSWEs with separate (or without) dynamic and feedback interface

In this section, several tests of numerical examples are presented to get the approximate solutions for free surface elevation using four different update schemes: Average, full-weighting, update mix-low, and update mix-high (with separate or without separate) dynamic and feedback interface using Dirichlet boundary conditions. Comparison l_2 -RE of free surface elevation for some examples by using four choices of restriction operator with two cases of the refinement factor.

6.9.1 Comparison l_2 -RE of free surface elevation in two-way nested grid by using four update operators when both space and temporal refinement ratio are 1:3 using Algorithm 4, Chapter 4

Example 1:

In this example, we use the system for 2DNSWEs given in equations (2.1)-(2.3) with non-rotated $f=0$, wind stress and viscosity= 0, if we take different values of time $t= 10, 20, \dots, 100$ days when $n_x=n_y=200$, $\Delta x=\Delta y=1$, $\Delta t=0.01s$ in coarse grid and $n_x=n_y=200$, $\Delta x=\Delta y=0.33$, $\Delta t=0.0033s$ in fine grid (space and temporal refinement ratio 1:3) to find l_2 -RE of free surface elevation with initial condition $u=v=\eta=0$.

The following figures compare l_2 -RE of free surface elevation using different update schemes. All simulations in the first figure is made by using Dirichlet boundary conditions **without separate**

dynamic interface and feedback interface and the second figure with a **separate** interface.

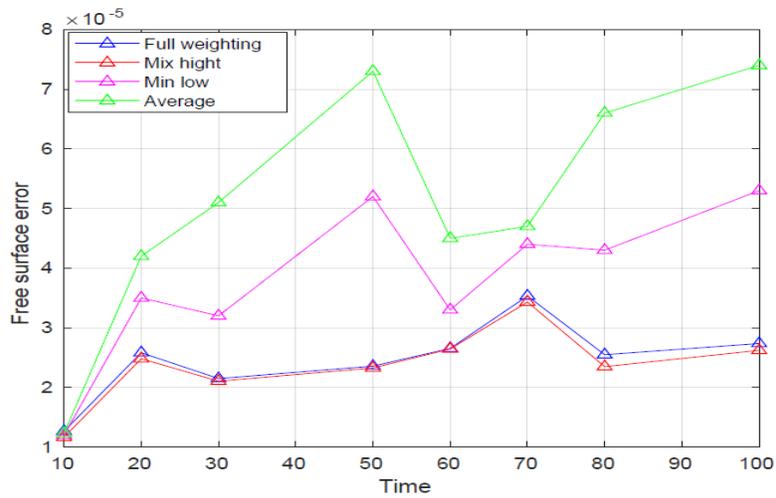


Figure 6-31: l_2 -RE in two-way nesting using four different update schemes without separate dynamic interface and feedback interface

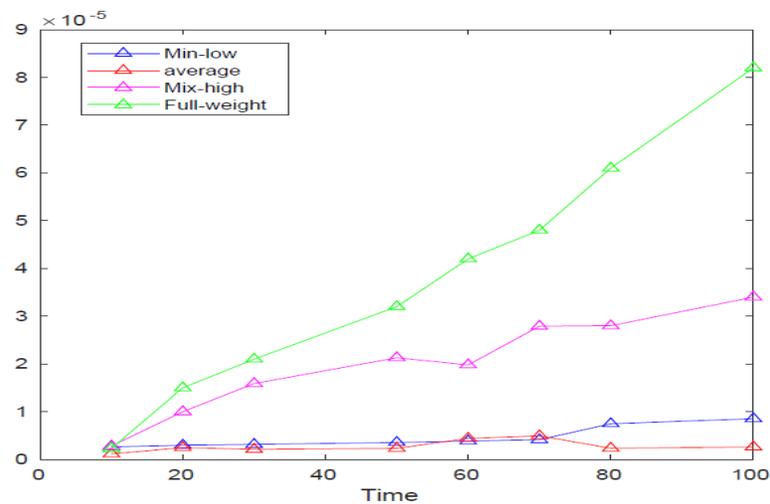


Figure 6-32: l_2 -RE in two-way nesting using four different update schemes with separate dynamic interface and feedback interface

The following figures show the approximate solutions on the coarse grid are obtained using four different update schemes: Average, full-weighting, update mix-low and update mix-high. The simulations based on the average and update mix-low operators lead to good solutions when the interface is a separate.

The following figures show the free surface elevation on the coarse grid domain after 10 days, 20 days, 30 days for using the average method.

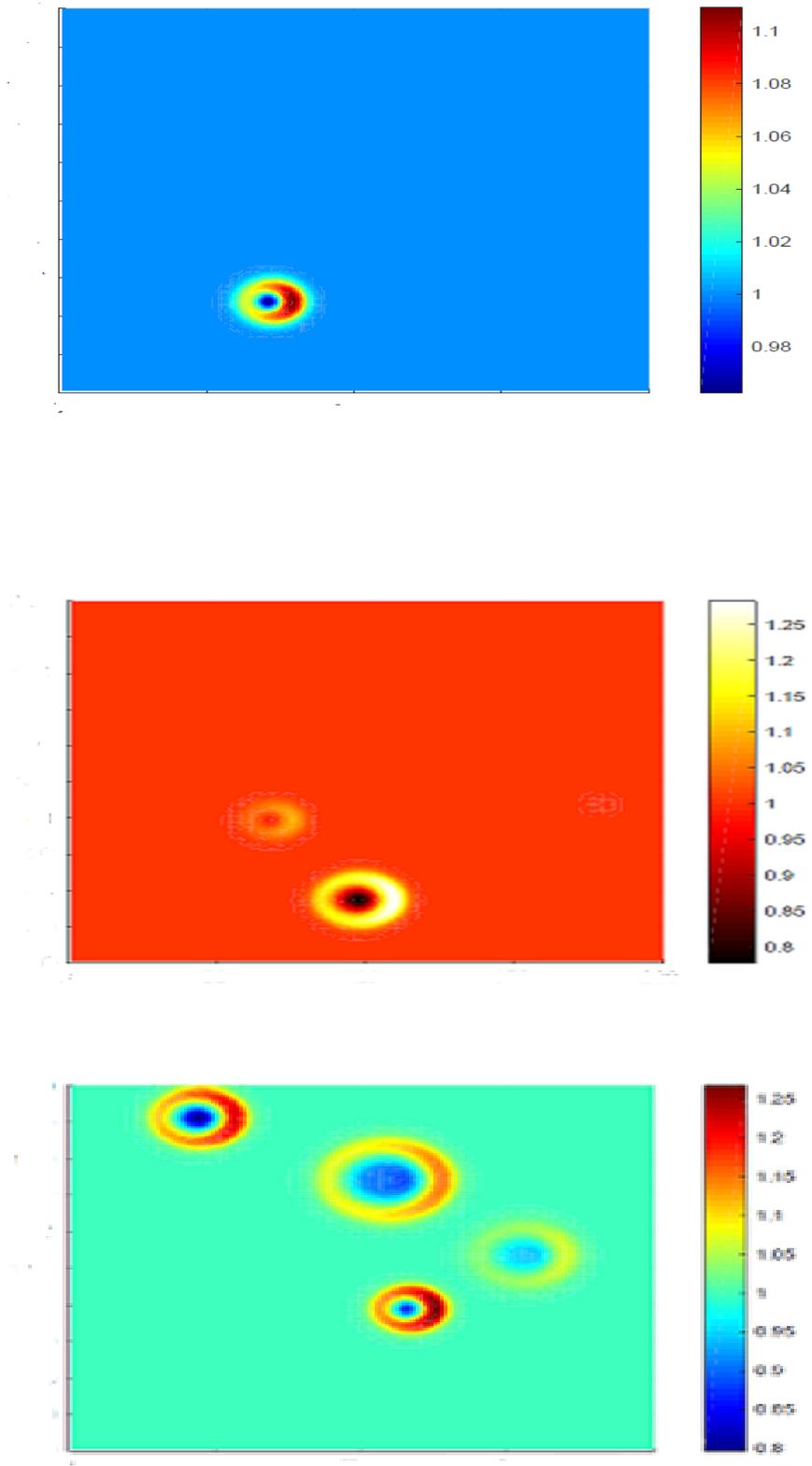


Figure 6-33: Free surface elevation on the coarse grid domain using average method.

The following figures show the free surface elevation on the coarse grid domain after 10 days, 20 days, 30 days for using **mix-low update method**.

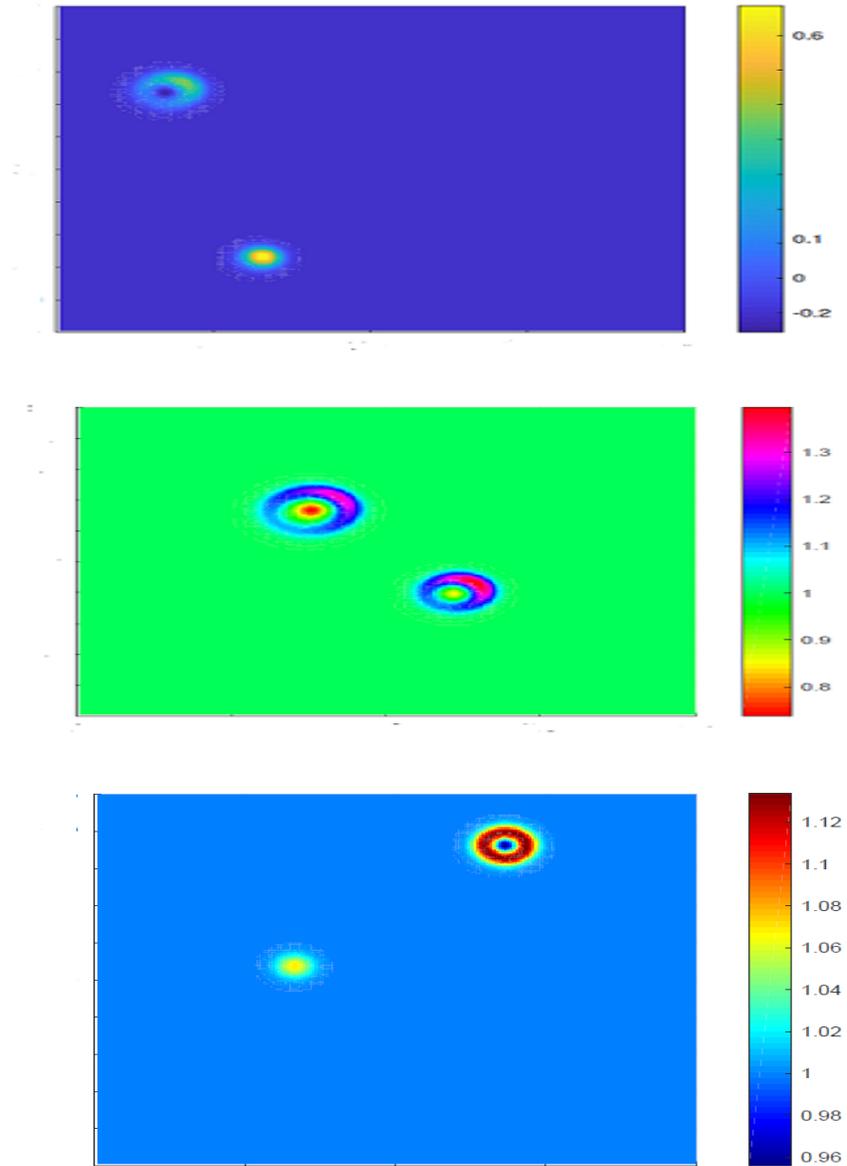
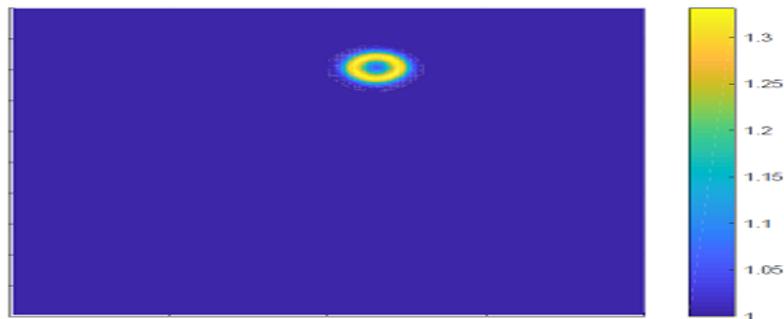


Figure 6-34: Free surface elevation on the coarse grid domain using mix-low method

The following figures show the free surface elevation on the coarse grid domain after 10 days, 20 days, 30 days by using **full-weighting method**.



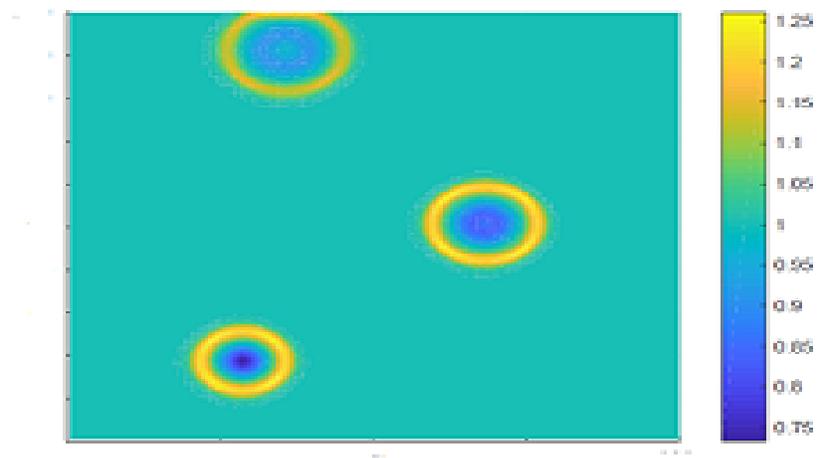
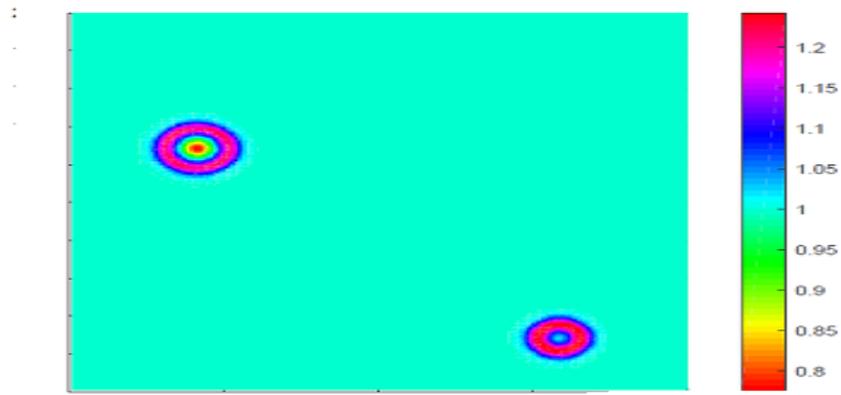
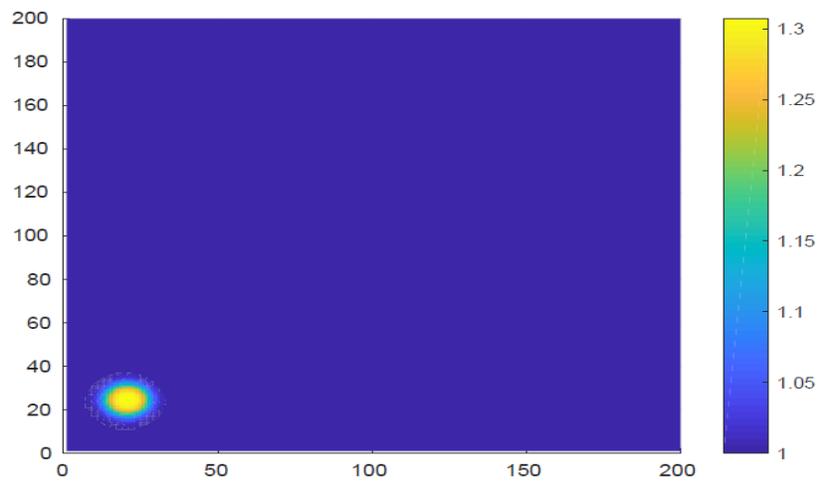


Figure 6-35: Free surface elevation on the coarse grid domain using full-weighting method

The following figures show the free surface elevation on the coarse grid domain after 10 days, 20 days, 30 days for using **mix-high method**.



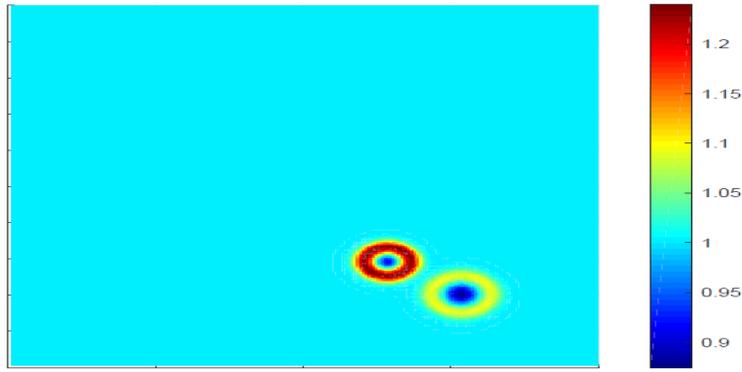


Figure 6-36: Free surface elevation on the coarse grid domain using mix-high method

6.10 Multiply nested techniques for 2DSWEs when the type of structured grids separate (or adjacent) interface

Here, the space refinement ratio 1:3 and time refinement ratio 1:2 using algorithm 5, Chapter 4.

Example 1:

In this example, we find ABSE of free surface elevation in two-way nesting grid for 2DSWEs given in equations (2.32)-(2.34) in a linear case when a fine grid contains again one fine grid in another level (child embedded or separable to parent).

Numerical parameters and results

The computational domain is discretized by a grid whose size is regular. Numerical values of the parameters are chosen as follows: If we take different values of time at difference times $t = 10, 20, \dots, 100$ days to find ABSE of free surface elevation in two-way nesting grid when $nx=ny=200$, $\Delta x=\Delta y=3$, $\Delta t=0.01s$ in a coarse grid and $nx=ny=200$, $\Delta x=\Delta y=1$, $\Delta t=0.005s$, and the time step

in fine grid is one half time in a coarse grids at each level and the space refinement ratio is 1:3 by using Dirichlet boundary condition. In this model, we use linear interpolation and for updating use the average method with a separate dynamic and feedback interface.

The following figures compares the ABSE of free surface elevation between (coarse grid-fine grid in level 2) and (fine grid in level 2-fine grid in level 3) for linear 2DSWEs.

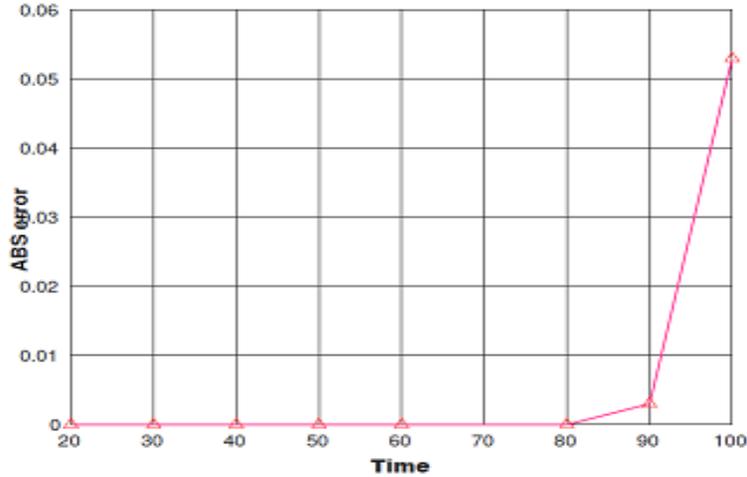


Figure 6-37: Comparison the ABSE between a coarse grid and a fine grid in level 2

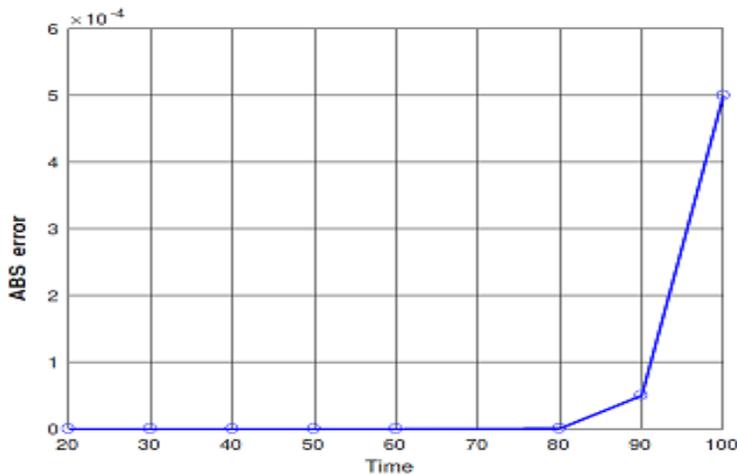


Figure 6-38: Comparison the ABSE between level 2 and level 3

Example 2:

In this example, we find *l2-RE* of free surface elevation in two-way nesting grid for 2DNSWEs given in equations (2.1)-(2.3) with wind stress=0, bottom stress =0, and $f=0$ when a fine grid contains again one fine grid in another level.

Numerical parameters and results

The computational domain is discretized by a grid whose size is regular. Numerical values of the parameters are chosen as follows: If we take different values of time $t= 10, 20, \dots, 1000$ hours when

$nx=ny=100$, $\Delta x=\Delta y=3$, $\Delta t=0.01s$ in a coarse grid (level 1), $nx=ny=100$, $\Delta x=\Delta y=1$, $\Delta t=0.005$ in a fine grid (level 2) and $nx=ny=100$, $\Delta x=\Delta y=0.33$, $\Delta t=0.0025s$ in (level 3) by using Dirichlet boundary conditions when consider only the common points for all grids.

The following figure compares $l2-RE$ between a coarse grid-fine grid (level 2) and fine grid (level 3) with a separate interface.

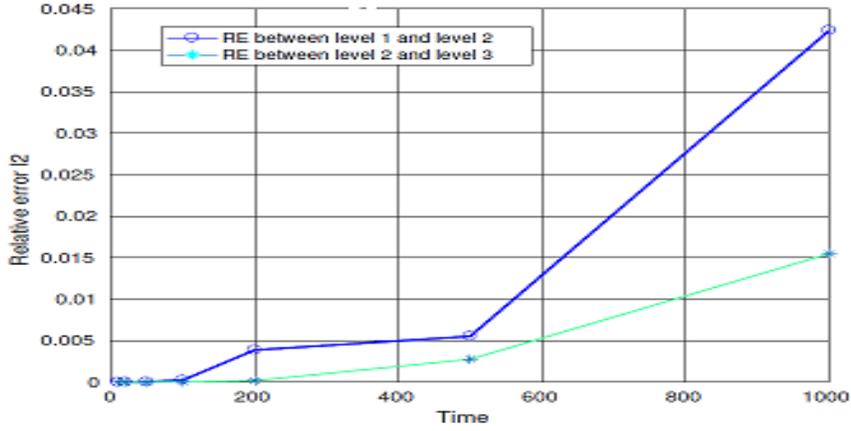


Figure 6-39: Comparison $l2-RE$ between a coarse grid-a fine grid (level 2) and level 2-level 3

The following figures compare $l2-RE$ between a coarse grid and a fine grid in level 2 when use four choice for update schemes with a separate interface and without a separate interface.

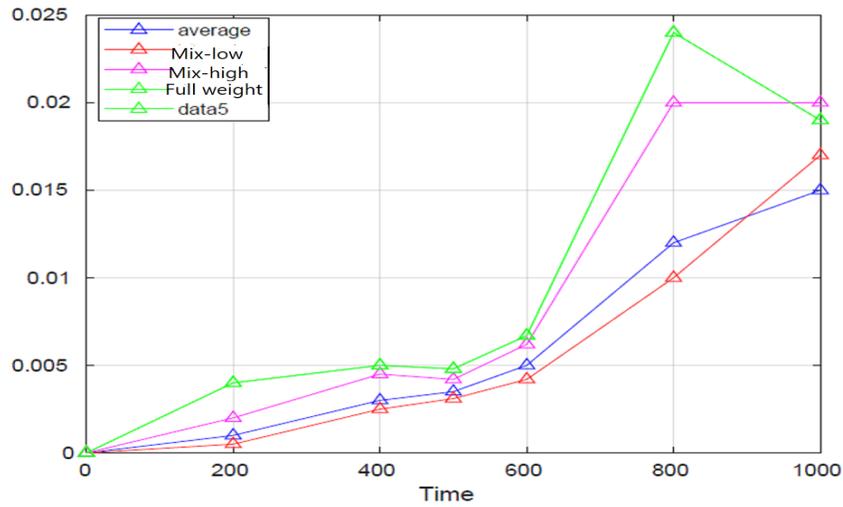


Figure 6-40: Comparison $l2-RE$ when use four choice for update schemes with a separate interface

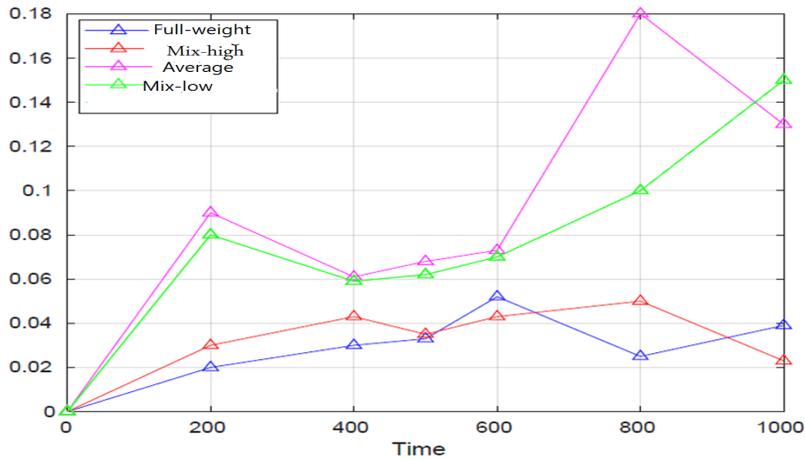


Figure 6-41: Comparison l_2 -RE when use four choice for update schemes without a separate interface

6.11 Summary and Conclusions

Effectively, a two-way nesting technique for 2D depth-averaged SWEs was developed. Results generated showed the two-way nested model can provide an accurate high-resolution solution for an area of interest and allow high-resolution data to update the lower-resolution coarse domain without incurring the computational cost for high resolution over the full domain.

Accuracy and efficiency are two main indicators in assessing the implementation of the numerical models. The approach used in this chapter appears the possibility of increasing accuracy and efficiency of the modeling results within a two-way nesting grid model.

To verify the nested multiply grid model, several numerical examples were presented and it was shown, in particular, a two-way nesting technique ensures dynamical consistency between a coarse grid and a fine grid occurs frequently. Some examples of coupling 3 systems for non-linear SWEs with nest 3:1 were applied. In general, good results were observed.

Compared the results of l_2 -RE of the free surface elevation using four choices of update restriction and the experiences indicated that the results of full-weighting and mix-high methods are the best results and very close to each other in the case without a separate interface. Finally, the results of l_2 -RE when $\Delta x \neq \Delta y$ were compared and the best results were shown compared to the case $\Delta x = \Delta y$. Several examples were tested and the results showed that the best resolution when used mix-low method or average method and the results between them are very close to each other when used a separate interface.

Chapter 7

Multiple Nested Grids For 2D Shallow Water Models

Chapter 7

Multiple Nested Grids For 2D Shallow Water Models

The results presented in this chapter (Section 7.2) are the subject of an article [6]

In this chapter, a two-way interaction technique for multiple nested grids at multiple levels (multiple regions) of 2DNSWEs are constructed. This model consists of a fine grid model nested 5:1 within a coarse grid large area model with different non-linear components in each region.

Although different grids size are employed in each subregion, physical variables in all subregions are solved simultaneously and it allows any ratio of grid sizes between two subregions. This model is highlighted by using an explicit center finite difference scheme in space and leapfrog with Robert-Asselin filter in time for the constant depth water with Dirichlet boundary conditions. The numerical solutions induced by the explicit methods are optimally controlled by choosing an appropriate time step size and grid size.

Comparison of l_2 -relative error norm results for several examples of 2DSWEs with nesting 3:1 and 5:1, show the ability and accuracy of this technique over different periods of time at multiple levels.

Highlights

- Suggested a new technique for a multiply nested grid of 2D shallow water models.
- Comparison of l_2 -RE results when the model has the spatial refinement ratio 1:3 and 1:5.
- Two-way nested coupling of 3 models for multiple grids at multiple levels (regions) is achieved for 2DSWEs with nesting 5:1.
- Demonstrate the accuracy and efficiency of the modeling results when using performance of the multiple nested grid techniques.

Contents

7.1	Two-Way nesting grid Algorithm	219
7.2	Numerical results for multiple nested grids	220
7.2.1	Example 1: When the space refinement ratio is 1:5 and the temporal refinement ratio is 1:2 for 2DNSWEs	220
7.2.2	Example 2: Comparison the numerical solutions of the nested grid at multiple levels when the space refinement ratio is 1:3 and 1:5 and temporal refinement ratio is 1:2	221
7.2.3	Example 3: When the space refinement ratio is 1:5 and temporal refinement ratio is 1:2 for linear 2DSWEs	222
7.2.4	Example 4: When the refinement ratio in both space and time equals 1:5 for linear 2DSWEs	223
7.2.5	Example 5: When the space refinement ratio is 1:5 and temporal refinement ratio is 1:2 for linear 2DSWEs	224
7.2.6	Example 6: When the space refinement ratio is 1:5 and temporal refinement ratio is 1:2 for 2DNSWEs	224
7.2.7	Example 7: Comparison of results when both space and temporal refinement ratio are 1:3 and 1:5 in cases one-way and two-way nesting	225
7.3	Example 8: Comparison <i>l2-RE</i> between the free surface, u-velocity and v-velocity when the space refinement is 1:5 and temporal refinement is 1:2	226
7.4	Summary and Conclusions	228

7.1 Two-Way nesting grid algorithm

Case 1: Space and temporal refinement factor equals 1:5
--

Suppose all information about the flux values and the water surface elevation in the inner region (with finer resolutions) and the outer region (the parent grid with the coarsest grid resolution), are known at time level $t = n\Delta t$ and we need to solve the inner and the outer region values at the next time steps $t = (n + 1)\Delta t$ and $t = (n + 2)\Delta t$.

Step 1: Get all information for the free surface elevation at $t = (n + 1)\Delta t$ in the outer region by solving continuity equation.

Step 2: Get all information for the flux values at $t = (n + 1)\Delta t$ in the outer region by solving momentum equations.

Step 3: To solve the continuity equation in the inner region, we need to have the flux information along the connected boundary at $t = n\Delta t$. So the information in the outer grids at the connected boundary are linearly interpolated and then those interpolated values are set to the fluxes in the inner at the boundary.

Step 4: Get the free surface elevation at $t = (n + 1/5)\Delta t$, $t = (n + 2/5)\Delta t$, $t = (n + 3/5)\Delta t$ and $t = (n + 4/5)\Delta t$ in the inner region by solving continuity equation.

Step 5: Get all information for the flux values at $t = (n + 1/5)\Delta t$, $t = (n + 2/5)\Delta t$, $t = (n + 3/5)\Delta t$ and $t = (n + 4/5)\Delta t$ in the inner region by solving momentum equations.

Step 6: Get the free surface elevation at $t = (n + 1)\Delta t$ in the inner grid region by using continuity equation.

Step 7: Solve the flux values at $t = (n + 1)\Delta t$ in the inner grid region by using momentum equations.

Step 8: Get the free surface elevation at $t = (n + 6/5)\Delta t$, $t = (n + 7/5)\Delta t$, $t = (n + 8/5)\Delta t$ and $t = (n + 9/5)\Delta t$ in the inner region in the inner grid region by using continuity equation.

Step 9: Get the flux values at $t = (n + 6/5)\Delta t$, $t = (n + 7/5)\Delta t$, $t = (n + 8/5)\Delta t$ and $t = (n + 9/5)\Delta t$ in the inner region by using momentum equations.

Step 10: To transfer the information from the inner grid region to the outer region, if the free surface elevation and the flux values at $t = (n + 1)\Delta t$ in the inner grid region is located at the same position for the coarse grid region then use copy grid. Otherwise, use the average or full-weighting operators.

Step 11: Get the free surface elevation at $t = (n + 2)\Delta t$ in the inner grid region by solving continuity equation.

Step 12: Get the flux values at $t = (n + 2)\Delta t$ in the inner region by using momentum equations.

Step 13: Solve the free surface elevation and the velocities at $t = (n + 2)\Delta t$ in the outer region using continuity equation and momentum equations.

Step 14: Transfer the information from the inner region to the outer region at $t = (n + 2)\Delta t$.

7.2 Numerical results for multiple nested grids

Some notes:

1. In all examples, Dirichlet boundary condition are applied, we use a nested grid with interface condition for the fine grids linear interpolation both spatially and temporally and update interface condition for the coarse grid by using the average method (both spatially and temporally).
2. In all examples, we find $l2-RE$ of free surface elevation in two-way nested grids using Algorithm 5 ,Chapter 4 and the number of coarse grids and fine grids are equals when consider only the common points to both grids with initial condition $u=v=\eta=0$.

7.2.1 Example 1: The space refinement ratio is 1:5 and the temporal refinement ratio is 1:2 for 2DNSWEs (with adjacent grids)

In this example, we find $l2-RE$ for 2DNSWEs given in equations (2.1)-(2.3) with $f=0$, viscosity=0 and wind stress=0 in level 1 (coarse grid) which contains one fine grid located in level 2 which contain another fine grid that located in level 3 at different times $t= 1000, 2000, \dots, 4000$ hour. The following table shows the information about the coarse grid and fine grids at multiple levels (regions).

Information	level 1 (coarse grid)	level 2 (fine grid)	level 3 (fine grid)
Number of grids	100×100	100×100	100×100
Length grid size	5	1	0.2
Coarse grid	non	Grid 01	Grid 22
Grid size ratio	non	5	5
Time step in sec	0.010	0.005	0.0025
SWEs	non-linear	non-linear	non-linear
(East-West)	1-100	61-80	61-80
(North-South)	1-100	61-80	11-30
CFL condition	0.7	0.7	0.7

Table 7.1: The information about the coarse and fine grids at multiple regions (levels) for 2DNSWEs

The following figure shows l_2 -RE between a coarse grid in level 1 and a fine grid in level 2 and l_2 -RE between fine grid in level 2 and fine grid in level 3.

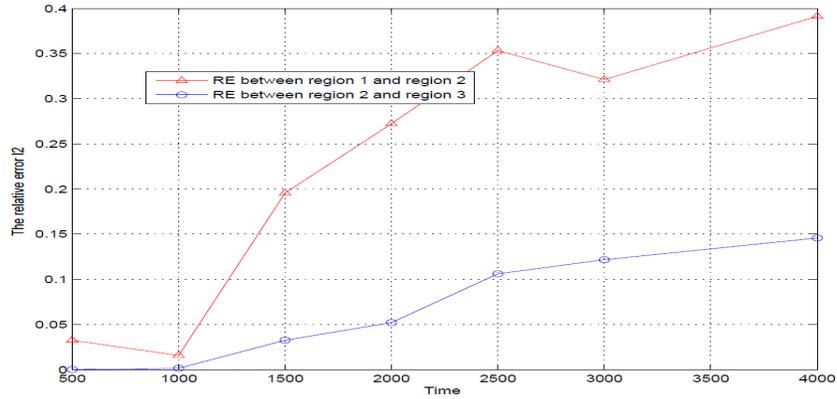


Figure 7-1: Comparison l_2 -RE between a coarse grid- fine grid in level 2 and level 2-level 3 in case space refinement ratio is 1:5

7.2.2 Example 2: Comparison l_2 -RE of the nested grid at multiple levels (regions) when the space refinement ratio is 1:3 and 1:5 and temporal refinement ratio is 1:2

As the same of the previous example, we find l_2 -RE for 2DNSWEs given by equations (2.1)-(2.3) with ($f=0$, viscosity=0 and wind stress=0) in level 1 (coarse grid) which contains one fine grid located in level 2 which contains another fine grid located in level 3 at difference times $t=1000, 2000, \dots, 4000$ hour.

The following figure represents l_2 -RE between a coarse grid-fine grid in level 2 and fine grid in level 2-fine grid in level 3 in case 1:3. The results when space refinement ratio 1:5 are the best.

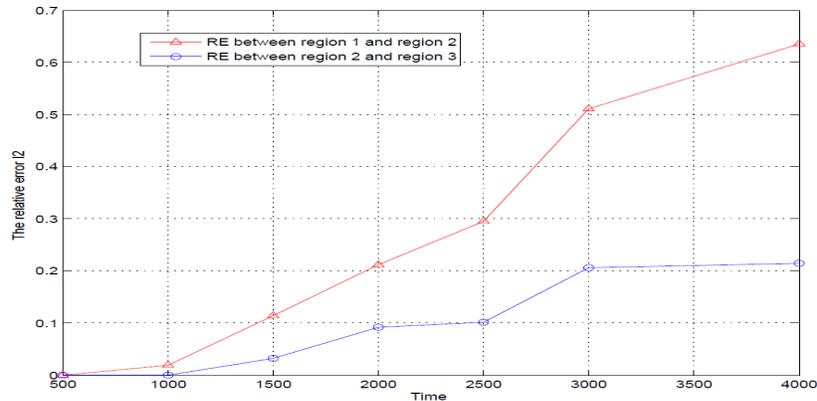


Figure 7-2: Comparison l_2 -RE between a coarse grid in level 1-fine grid in level 2 and fine grid in level 2-fine grid in level 3 in case the space refinement ratio is 1:3

7.2.3 Example 3: When the space refinement ratio is 1:5 and temporal refinement ratio is 1:2 for linear 2DSWEs

In this example, we find l_2 -RE for 2DSWEs given in equations (2.33)-(2.35) for linear case in level 1 (coarse grid) which contains one fine grid located in level 2 which contains another one fine grid located in level 3 for difference times $t = 1000, 2000, \dots, 5000$ hour. The information about the coarse and fine grids are given by Table 7.1. The first figure compares l_2 -RE between a coarse grid in level 1-fine grid in level 2 and fine grid in level 2-fine grid in level 3 in case 1:5 and the second figure compares l_2 -RE between a coarse grid in level 1-fine grid in level 2 and fine grid in level 2-fine grid in level 3 in case 1:3.

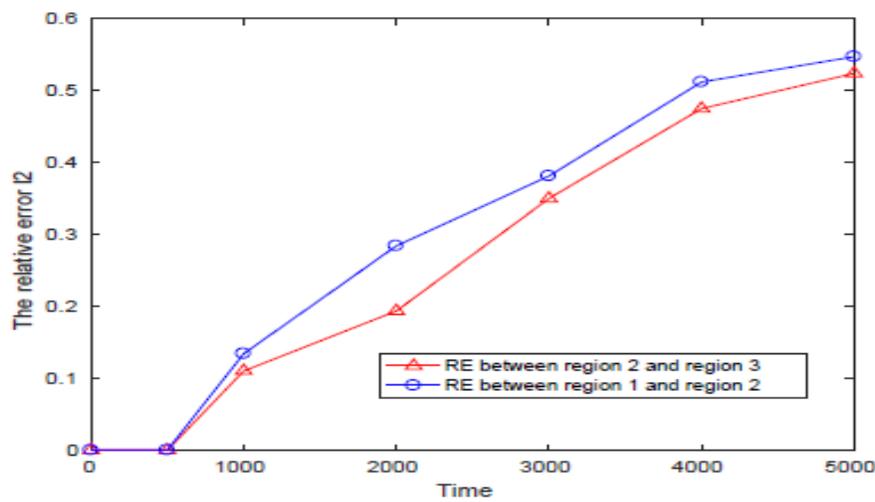


Figure 7-3: Comparison l_2 -RE between a coarse grid in level 1-fine grid in level 2 and fine grid in level 2-fine grid in level 3 in case 1:5

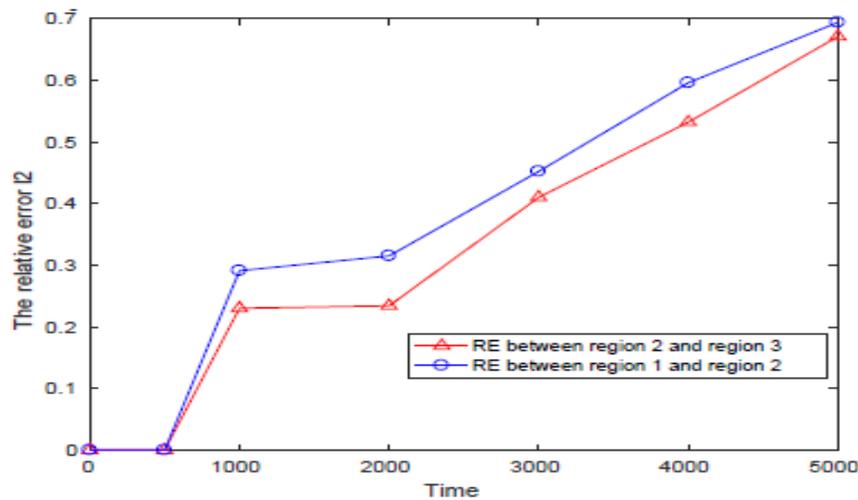


Figure 7-4: Comparison l_2 -RE between a coarse grid in level 1-fine grid in level 2 and fine grid in level 2-fine grid in level 3 in case 1:3

7.2.4 Example 4: When the refinement ratio in both space and time equals 1:5 for linear 2DSWEs with a separate interface

In this example, we use 2DSWEs given in equations (2.32)-(2.34) for linear case. If we take different values of time $t = 20, 30, \dots, 100$ days to find l_2 -RE between a coarse grid and a fine grid when $n_x=100$, $n_y=100$, $\Delta x=3$, $\Delta y=3$ in a coarse grid, $n_x=100$, $n_y=100$, $\Delta x=0.6$, $\Delta y=0.6$ in fine grid and the time step in a coarse grid is 0.005s.

The following figures compare ABSE and l_2 -RE in one-way nesting and two-way nesting for the linear 2DSWEs. When comparing this results with the results in Example 1, Section 5.5, we obtained good results by using space refinement ratio 1:5 because the large ratio gives very well connected boundary conditions.

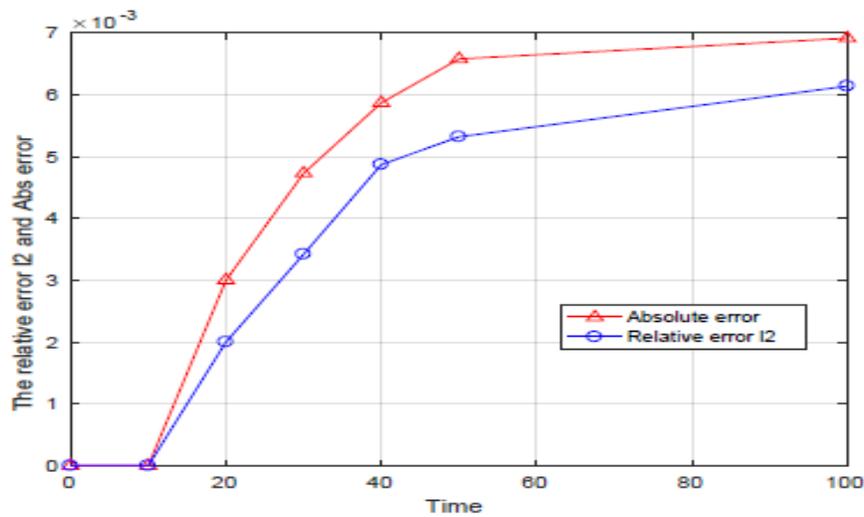


Figure 7-5: Comparison l_2 -RE and ABSE in one-way nesting in case 1:4

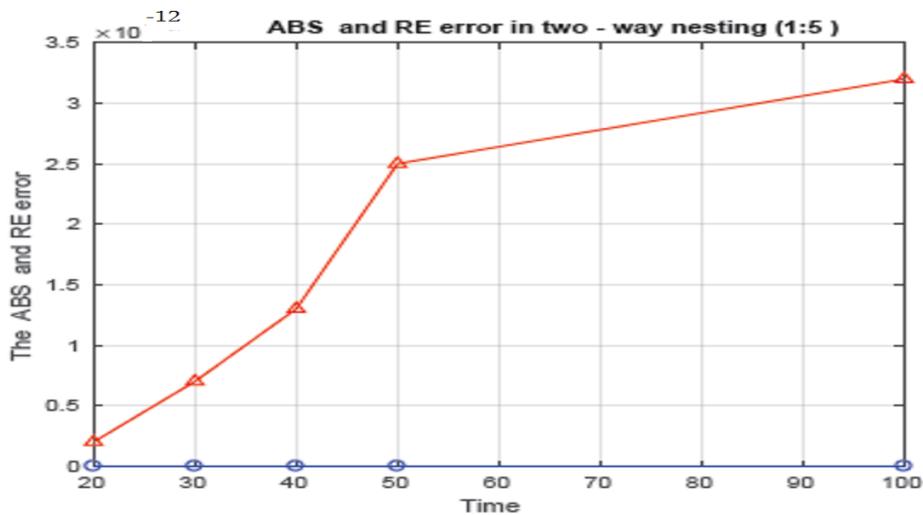


Figure 7-6: Comparison l_2 -RE and ABSE in two-way nesting in case 1:5

7.2.5 Example 5: When the space refinement ratio is 1:5 and temporal refinement ratio is 1:2 for linear 2DSWEs with a separate interface

In this example, we find ABSE for 2DSWEs given in equations (2.32)-(2.34) for a linear case when a fine grid contains again one fine grid in another level at difference times $t= 10,20, \dots,300$ hour by using the Dirichlet open boundary condition when $nx=ny=200$, $\Delta x=\Delta y=3$, $\Delta t=0.01s$ in a coarse grid and $nx=ny=200$, $\Delta x=\Delta y=0.6$, $\Delta t=0.005s$ in a fine grid.

The following figure compares ABSE of free surface elevation between a coarse grid in level 1 and fine grid in level 2. When comparing these results with the results in Example 1, Section 6.10, we obtained very well results when space refinement ratio is 1:5.

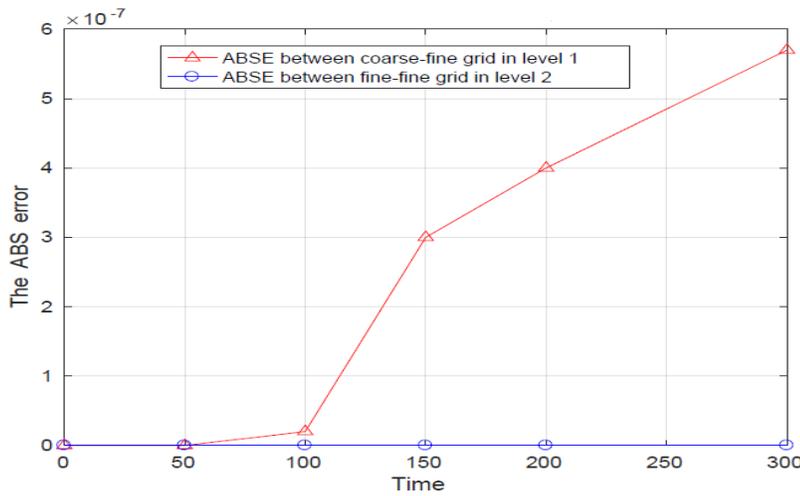


Figure 7-7: Comparison ABSE between a coarse grid in level 1-fine grid in level 2 and fine grid in level 2-fine grid in level 3

7.2.6 Example 6: When space refinement ratio is 1:5 and temporal refinement ratio is 1:2 for 2DNSWEs with a separate interface

In this example, we find $l2-RE$ of free surface elevation for 2DNSWEs given in equations (2.1)-(2.3) with wind stress=0, bottom stress =0, and $f=0$, when a fine grid contains again one fine grid in another level (region). If we take different values of time $t= 100,200,\dots, 1000$ hours, $nx=ny=100$, $\Delta x=\Delta y=3$, $\Delta t=0.01s$ in a coarse grid, and $nx=ny=100$, $\Delta x=\Delta y=0.6$, $\Delta t=0.005s$ in a fine grid (level 2) and $nx=ny=100$, $\Delta x=\Delta y=0.12$, $\Delta t=0.00025s$ in fine grid (level 3) with CFL condition 0.02.

The following figures show $l2-RE$ of free surface elevation between a coarse grid in level 1 and fine grid in level 2. The results are very good compared to the results in Example 2, Section 6.10.

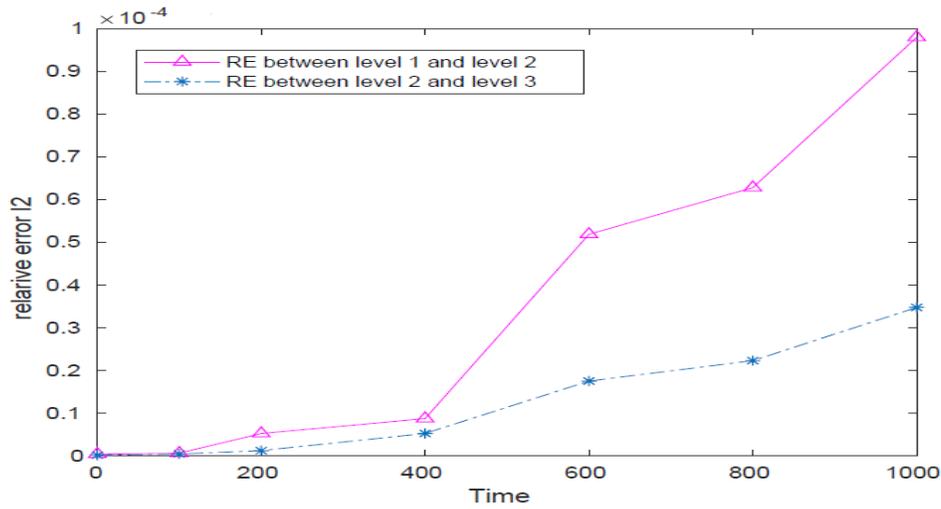


Figure 7-8: Comparison l_2 -RE between a coarse grid in level 1-fine grid in level 2 and fine grid in level 2-fine grid in level 3 for 2DNSWEs

7.2.7 Example 7 : Comparison of results when both space and temporal refinement ratio are 1:3 and 1:5 in cases one-way nesting and two-way nesting

In this example, we find l_2 -RE in one-way nesting and two-way nesting for 2DNSWEs given in equations (2.32)-(2.34) at different values of time $t = 100, 200, \dots, 1000$ hours, when $n_x = n_y = 300$, $\Delta x = \Delta y = 1$, and $\Delta t = 0.01$ s in a coarse grid using initial condition $u = v = \eta = 0$ with CFL condition 0.13.

The following figures show l_2 -RE and ABSE in one-way nesting and two-way nesting grids in two cases 1:3 and 1:5.

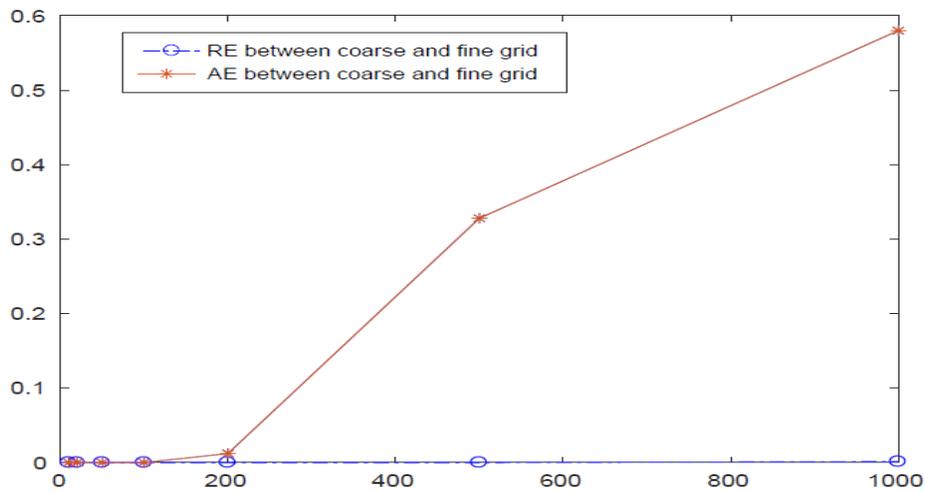


Figure 7-9: Comparison l_2 -RE and ABSE for one-way nesting in case 1:3

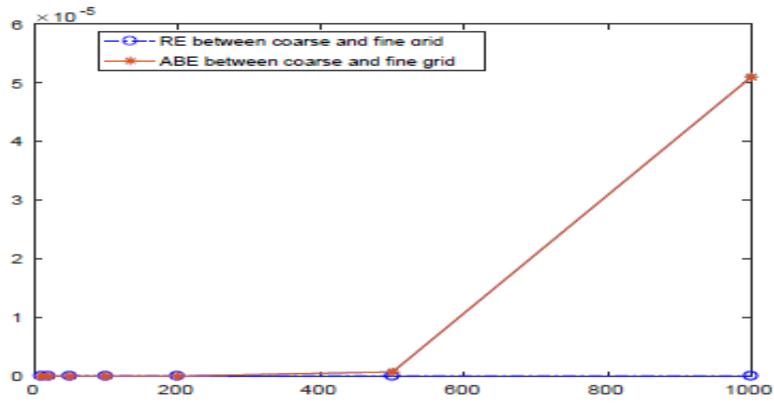


Figure 7-10: Comparison l_2 - RE and ABSE for two-way nesting in case 1:3

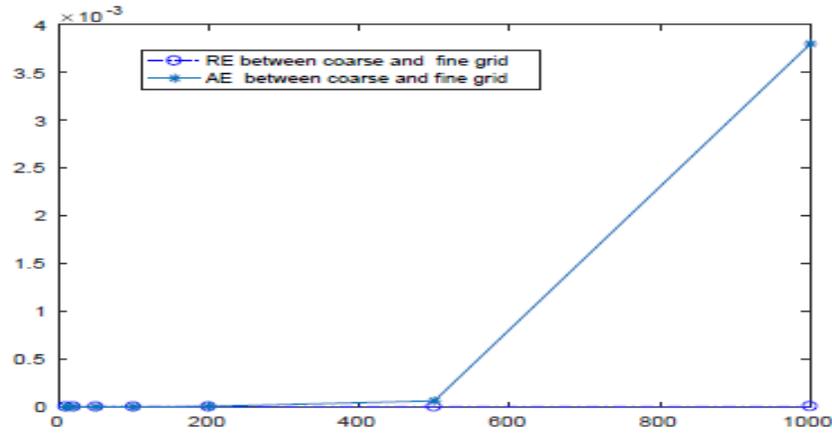


Figure 7-11: Comparison l_2 - RE and ABSE for one-way nesting in case 1:4

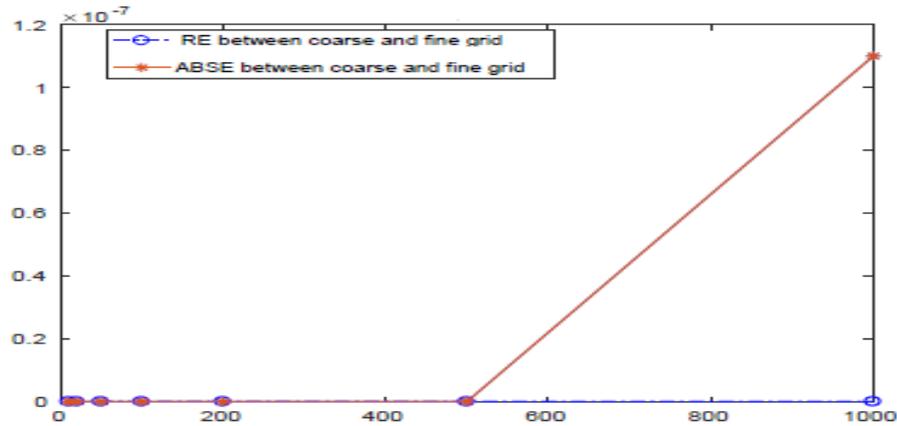


Figure 7-12: Comparison l_2 - RE and ABSE for two-way nesting in case 1:5

7.3 Example 8: Comparison l_2 - RE between the free surface elevation, u-velocity and v-velocity

In this example, we use system of 2DNSWEs given in equations (2.1)-(2.3) with ($\nu=0$, wind stress=0 and $f=0$) to find l_2 - RE for free surface, u -velocity and v -velocity at the different times $t=500$, 1000 ,

...,4000 hours. Numerical values of the parameters are chosen as follows: $nx=ny=150$, $\Delta x=\Delta y=5$, $\Delta t=0.0025$ s in a coarse grid and $nx=ny=150$, $\Delta x=\Delta y=1$, $\Delta t=0.00015$ s in a fine grid when the space refinement is 1:5 and temporal refinement is 1:2 with CFL condition 0.02.

The following figure shows $l2-RE$ of free surface elevation, u -velocity and v -velocity between a coarse grid and a fine grid.

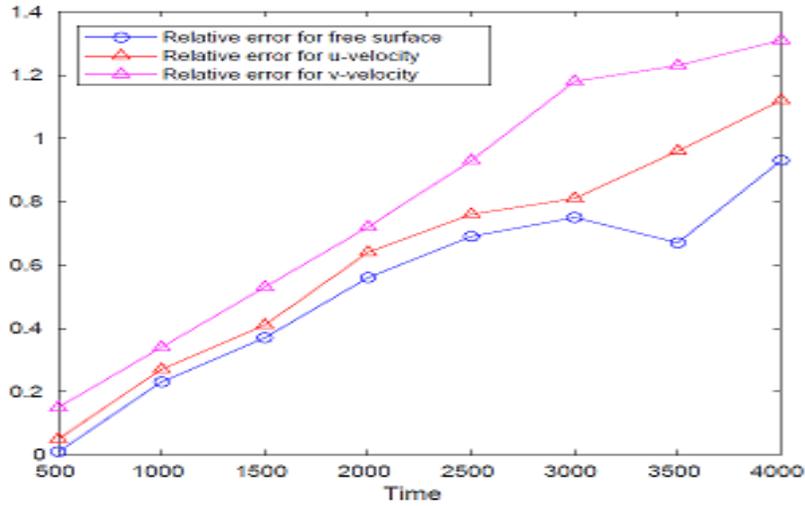


Figure 7-13: Comparison $l2-RE$ of free surface elevation, u -velocity and v -velocity

7.4 Summary and Conclusions

This chapter focused on a new technique for multiple nested grids at multiple levels (regions) of 2DSWEs. Nesting procedure was tested with data under different conditions. The approach used in this chapter was showed the possibility of increasing accuracy and efficiency of the modeling results within a two-way nesting technique with nest 5:1.

To verify multiple nested grid models, several numerical examples were applied and it was shown that two-way nesting techniques perform very well when the space refinement ratio 1:3 and 1:5. Comparison of l_2 - RE results when the space refinement ratio is 1:3 and 1:5. The results were compared between one-way and two-way nesting grids for cases 1:3 and 1:5 and very well results were obtained when the refinement factor is 1:5. Finally, l_2 - RE results were compared between the free surface elevation, u -velocity, and v -velocity and the results showed that l_2 - RE of free surface elevation are the best results.

Numerical Results Of The Tsunami Model

Chapter 8

Some Applications For Multiple Nested Grids Of The Tsunami Model

Chapter 8

Some Applications For Multiple Nested Grids Of The Tsunami Model

Some of the results presented in this chapter (**Sections 8.3-8.5**) are the subject of an article [\[4\]](#)

A two-way interaction technique for multiple nested grids of the tsunami model is proposed. This model adopts staggered an explicit center finite difference method and leapfrog with Robert-Asselin filter which contains linear and nonlinear components in each subregion with Cartesian coordinates and consists of a fine grid model nested 5:1 within a coarse grid large area model. A nested grid model, dynamically coupled up to multiple levels with various grid resolution, can be implemented in the model to fulfill the need to simulate this model in different levels. Nested grid system means in a region of one grid size, there are one or more regions with smaller grid sizes situated in which finally form a hierarchy of grids and grid levels.

In a two-way nested grid model, the information for velocity components and free surface elevation from the coarse mesh can enter and effect to the fine mesh in each time step of the solution process using linear interpolation and the information feedback from the fine mesh to the coarse mesh using the average scheme. To verify multiple nested grids model, several of numerical examples are applied and the results demonstrate the applicability and benefits of nesting.

Highlights

- Suggest a new technique for multiple nested grids of the tsunami model.
- Discuss some numerical examples of the tsunami model when the model has the spatial refinement ratio 1:5.
- Coupling four systems for multiple grids at multiple levels is achieved of the tsunami model with nesting 5:1.
- The capacity and benefits of nesting show by some examples.

Contents

8.1	Introduction	235
8.2	Development	236
8.3	Numerical Results	237
8.3.1	Case 1: When the space refinement ratio is 1:5 and temporal refinement ratio is 1:2 for 2D nonlinear SWEs	237
8.4	Case 2: When the space refinement ratio is 1:5 and temporal refinement ratio is 1:2 for linear 2DSWEs	242
8.4.1	Comparison l_2 -RE for the free surface in cases separate grids and embedded grids	245
8.4.2	Comparison the results of l_2 -RE when use the time step are 0.010 and 0.030 For linear SWEs	246
8.4.3	Comparison the results of l_2 -RE when the space refinement ratio are 1:3 and 1:5 in multiple levels for nonlinear SWEs	248
8.5	Case 3: Coupling three systems when the space refinement ratio 1:5 with no refinement of time	249
8.5.1	Example 1 : Case 1 : Coupling (embedding) systems for 2D nonlinear SWEs	249
8.5.2	Example 2: Case 2: Coupling systems for 2D non-linear shallow water equations	250
8.5.3	Example 3: Case 3: Coupling systems for 2D nonlinear / linear shallow water equations	251
8.6	Summary and Conclusions	252

8.1 Introduction

Tsunami is a Japanese word that is a combination of two-word roots (tsu) means the port and nami means a wave which meaning harbor wave. It is a typical long wave (massive wave) in the ocean, are mostly caused by large earthquakes on the seafloor or water surface disturbances over a sufficiently large area and it can cause huge destruction when they hit coastlines.

Tsunamis are able to move long distances (across the ocean) without too much energy dissipation, which is due to lower friction slightly in the deep ocean. It can efficiently transfer energy released by earthquakes which is sometimes huge enough to severely damage facilities on the coastline. In the case of deep oceans, the wavelengths of tsunamis are of the order of 10km or 100km and the wave period ranges from around ten minutes to two hours.

The speed of tsunamis can reach as fast as 970 kilometers per hour (600 miles per hour) in the deepest oceans although the wave height is often merely tens of centimeters. As a result, in the deep sea, it is difficult to visually recognize a traveling tsunami. Regardless of its harmless performance in the deep ocean, when a tsunami approaches shallow water, its wave height increases significantly up to 30 m above sea level [46, 78, 116].

The tsunamis has been observed and recorded since ancient times, especially in Japan and the Mediterranean areas. The earliest recorded tsunami occurred in (2,000) B.C. off the coast of Syria. The oldest reference tsunami record is dated back to the 16th century in the United States [108].

The first historical reference to a tsunami is conjectured to be the wave created after the volcanic eruption of Thera in ancient Greece, around (1500-1450) B.C. Others, support that tsunami history dates even further back to around 6100 B.C, when a tsunami was supposedly triggered in the Norwegian Sea by the Storegga Slides ([32, 46].

In recent years, the most devastating tsunami was triggered by (2004) Sumatra earthquake off the coast of Indonesia and caused tremendous property loss and over 225,000 casual ties in the surrounding countries of Indian Ocean, especially in Indonesia, Sri Lanka, Thailand and India (see [115, 116]).

To mitigate the tsunami, it is very important to construct inundation maps along with those coastlines vulnerable to tsunami flooding. These maps should be developed based on the historical tsunami events and hypothetical scenarios. To produce realistic and reliable inundation estimates, it is essential to use a numerical model that calculates accurately the tsunami propagation from a source region to the coastal areas of concern and the subsequent tsunami run-up and inundation [114].

At the beginning of a simulation start, the initial water surface elevation (obtained from a given data file) is interpolated into all sub-regions grids, as well as fluxes values of velocities, are zero on all grids.

The following figures show the brutal tsunami



This chapter is arranged as follows: Discuss some numerical examples of 2D non-linear shallow water models given in Sections 2.4, 2.6 and 2.7 when the time step $(n+1/2)$. Compare the results when the model has the spacial refinement ratio 1:5.

8.2 Development

In most previous studies, with an adaptive mesh refinement case, the update is still made with this average formula because the grid can move from one time step to another, so that an average restriction is necessary to globally ensure the conservation before and after the regridding (renewal) step and that this was also the main reason for using the average restriction operator in ocean models when using moving mesh methods.

In this chapter, we suggest a new technique for multiple nested grids of the tsunami model that given in [108, 114, 121] using an explicit finite difference method and leapfrog with Robert-Asselin filter with moving boundary condition, linear interpolation and to update use the average operator. For more details (see Appendix).

Notes

1. For all examples, we use the formulas of numerical discretization 2DSWEs given in Sections 2.4, 2.5, and 2.6 and we find $l2-RE$ of free surface elevation in two-way nested grids.
2. The bottom friction comes from Manning's formula which is uniform throughout the grids, where n is roughness coefficient. In this simulation, n takes 0.013.
3. All simulations are made using moving boundary conditions, we use a nested grid with interface

condition for the fine grids linear interpolation and update interface condition for the coarse grid by using the average method in case **adjacent structured grids**.

8.3 Numerical Results

A full description of initial condition, boundary condition and model configuration can be found in ([114, 121]). The information for the velocities and the free surface elevation are exchanged on the boundaries between two nested grid regions. At each new time level, the information on the boundary of a finer grid are obtained by linearly interpolating in (both spatially and temporally). At each next time level for the outer parent grid, the free surface elevation and velocity on a coarser grid are updated by averaging scheme (both spatially and temporally).

8.3.1 Case 1: When space refinement ratio is 1:5 and temporal refinement ratio is 1:2 for 2DNSWEs

Example 1:

In this example, we use system of 2DNSWEs given in equations (2.8), (2.21), and (2.30) in Section 2.4 with ($\nu=0$, non-rotate $f=0$, wind stress=0) to find *l2-RE* of free surface elevation in case a coarse grid contains four fine grids in level 2 at difference times $t= 500, 1000, \dots, 5000$ sec when the water depth is constant.

The information on the set up of different grids for 2DNSWEs are given below.

Information	Grid 01	Grid 21	Grid 22	Grid 23	Grid 24
Number of grids	100×100	100×100	100×100	100×100	100×100
Length grid size	10	2	2	2	2
Coarse grid	non	Grid 01	Grid 01	Grid 01	Grid 01
Grid size ratio	non	5	5	5	5
Time step in sec	0.020	0.01	0.01	0.01	0.01
SWEs	non-linear	non-linear	non-linear	non-linear	non-linear
CFL condition	0.7	0.7	0.7	0.7	0.7
Bottom stress	0.013	0.013	0.013	0.013	0.013

Table 8.1: The information on the set up of the different grids

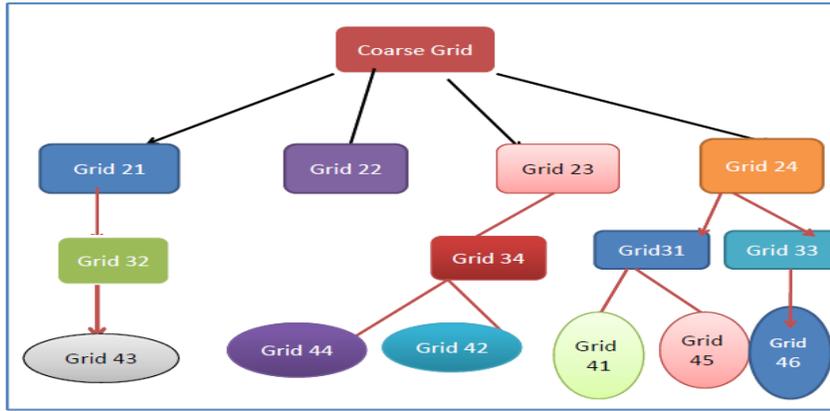


Figure 8-1: Multiple nested grid at multiple levels for 2DNSWEs

The following figures show l_2 -RE of free surface elevation in two-way nesting between a coarse grid and level 2 (grid 21, grid 24, grid 22, and grid 23).

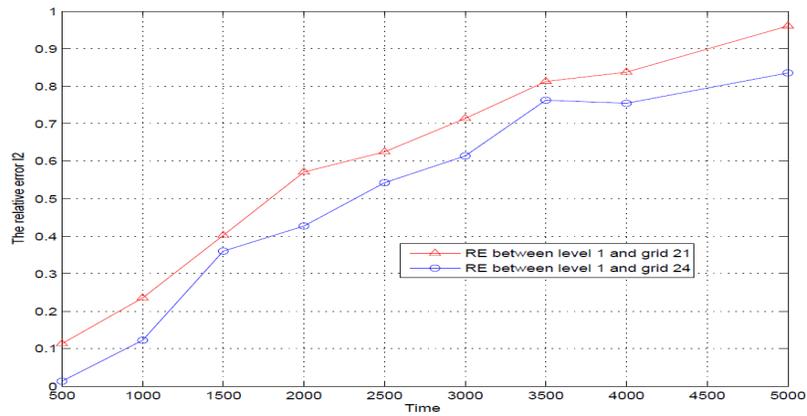


Figure 8-2: Comparison l_2 -RE of free surface between a coarse grid and level 2 (grid 21 and grid 24)

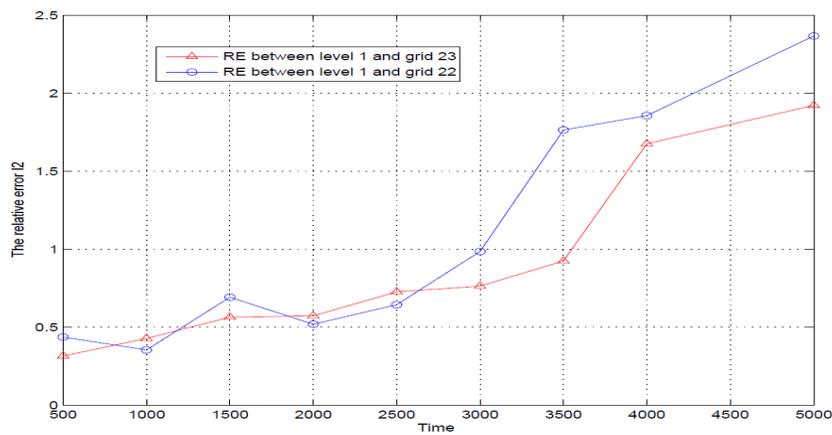


Figure 8-3: Comparison l_2 -RE between a coarse grid and level 2 (grid 22 and grid 23)

Notes that: All the fine grids here are separate from each other.

Example 2 :

By the same previous example, we use system of 2DNSWEs given by equations in Section (2.4) with ($\nu=0$, non-rotate $f=0$, wind stress=0) to find $l2-RE$ of free surface elevation in case a coarse grid contains one fine grid in level 2 which contains again one fine grid in level 3. Also, a fine grid in level 3 contains again one fine grid in level 4 at difference times $t=500, 1000, \dots, 5000$ sec.

The information on the set up of the different grids for 2DNSWEs are given below:

Information	Grid 01	Grid 21	Grid 32	Grid 43
Number of grids	100 × 100	100 × 100	100 × 100	100 × 100
Length grid size	10	2	0.40	0.08
Coarse grid	non	Grid 01	Grid 21	Grid 32
Grid size ratio	non	5	5	5
Time step in sec	0.020	0.01	0.005	0.0025
SWEs	non-linear	non-linear	non-linear	non-linear
Bottom stress	0.013	0.013	0.013	0.013
CFL condition	0.7	0.7	0.7	0.6

Table 8.2: The information on the set up of the different grids for 2DNSWEs

The following figures show $l2-RE$ of free surface elevation between a coarse grid-level 2 (grid 21), grid 21-level 3 (grid 32) and grid 32-level 4 (grid 43).

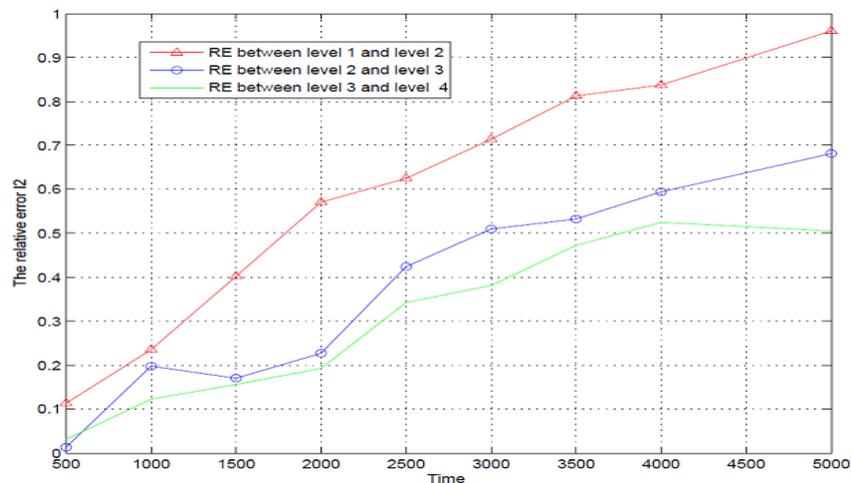


Figure 8-4: Comparison $l2-RE$ between coarse grid-level 2 (grid 21), level 2-level 3 (grid 32) and level 3-level 4 (grid 43)

Example 3 :

By the same previous example, we use system of 2DNSWEs to find l_2 -RE of free surface elevation in case a coarse grid contains another one fine grid (grid 24) in level 2 which contains again another two fine grids (grid 31 and grid 33) in level 3. Also, the fine grids in level 3 which contain again another one fine grid (grid 41) in level 4 at difference times $t= 500, 1000, \dots, 5000$ sec.

The information on the set up of the different grids for 2DNSWEs are given below:

Information	Grid 01	Grid 24	Grid 31	Grid 33	Grid 46
Number of grids	100×100	100×100	100×100	100×100	100×100
Length grid size	10	2	0.40	0.40	0.08
Coarse grid	non	Grid 01	Grid 24	Grid 24	Grid 33
Grid size ratio	non	5	5	5	5
Time step in sec	0.020	0.01	0.005	0.005	0.0025
SWEs	non-linear	non-linear	non-linear	non-linear	non-linear
CFL condition	0.7	0.7	0.7	0.6	0.6

Table 8.3: The information on the set up of the different grids at multiple level for 2DNSWEs

The following figures show l_2 -RE of free surface elevation between a coarse grid-level 2 (grid 24), grid 24-level 3 (grid 33), and grid 33-level 4 (grid 46).

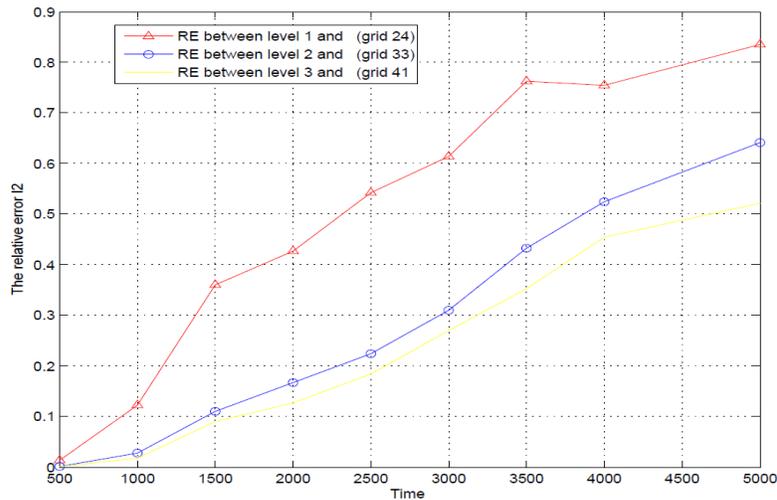


Figure 8-5: Comparison l_2 -RE between a coarse grid-level 2 (grid 24), level 2-level 3 (grid 33), and level 3-level 4 (grid 46)

Example 4:

By the same previous example, we use system of 2DNSWEs with ($\nu=0$, non-rotate $f=0$, wind stress=0) to find $l2-RE$ of free surface elevation in case a coarse grid contains one fine grid in level 2 which contains again one fine grid in level 3. Also, a fine grid in level 3 that contains again two fine grids in level 4 at difference times $t= 500, 1000, \dots, 5000$ sec.

Information	Grid 01	Grid 23	Grid34	Grid 44	Grid 42
Number of grids	100×100	100×100	100×100	100×100	100×100
Length grid size	10	2	0.40	0.08	0.133
Coarse grid	non	Grid 01	Grid 23	Grid34	Grid34
Grid size ratio	non	5	5	5	3
Time step in sec	0.020	0.01	0.005	0.0025	0.0025
SWEs	non-linear	non-linear	non-linear	non-linear	non-linear

Table 8.4: The information on the set up of the different grids at multiple levels

Note

The space refinement ratio at level 2, level 3 is 1:5 and the space refinement ratio at level 4 with grid 42 is 1:3. The following figures show $l2-RE$ of free surface elevation between a coarse grid-level 2 (grid 23), level 2-level 3 (grid 34), and level 3-level 4 (grid 44 and grid 42).

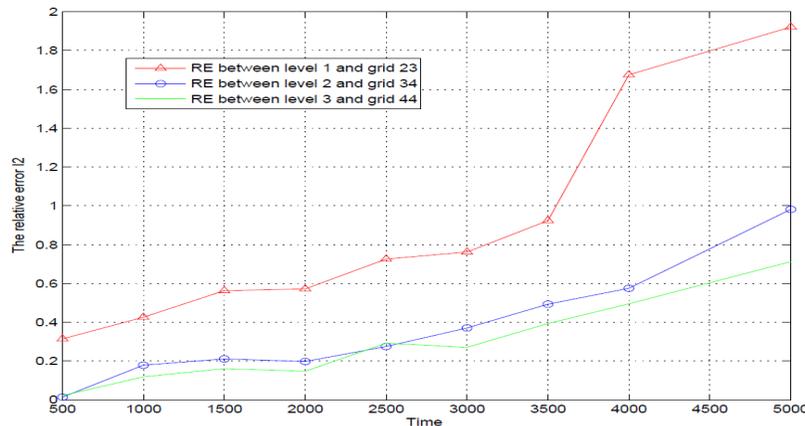


Figure 8-6: Comparison $l2-RE$ of free surface elevation between a coarse grid-level 2 (grid 23), level 2-level 3 (grid 34) and level 3-level 4 (grid 44)

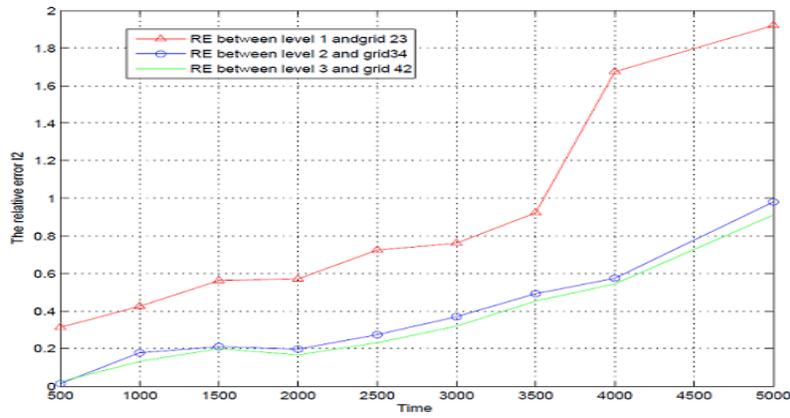


Figure 8-7: Comparison l_2 -RE between a coarse grid-level 2 (grid 23) and level 2-level 3 (grid 34) and level 3-level 4 (grid 42)

Notes that: All the results in a grid 24 are the best and the results in a grid 44 are better than the results in a grid 42.

8.4 Case 2: When the space refinement ratio is 1:5 and temporal refinement ratio is 1:2 for linear 2DSWEs

Example 1:

In this example, we use system of linear 2DSWEs given in equations (2.8), (2.39), and (2.44) in Section 2.6 with non-rotate $f=0$, wind stress=0 to find l_2 -RE of free surface elevation in case a coarse grid which contains four fine grids at difference times $t=500, 1000, \dots, 5000$ sec.

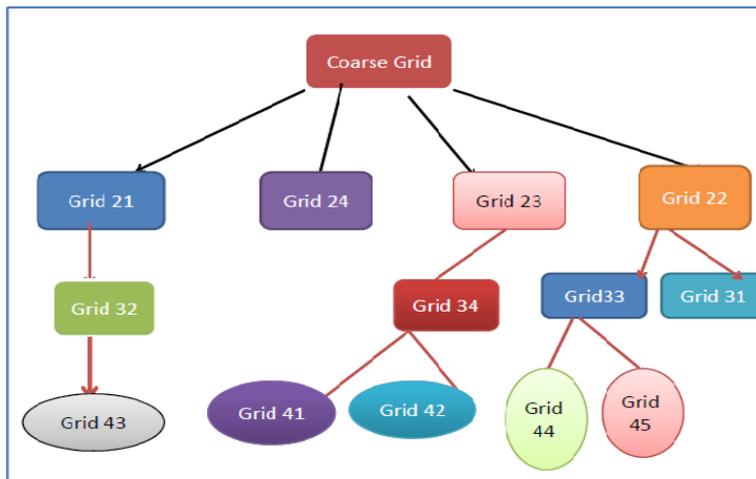


Figure 8-8: Multiple nested grids at multiple levels for linear 2DSWEs

The information on the set up of the different grids for linear 2DSWEs are given below:

Information	Grid 01	Grid 21	Grid 22	Grid 23	Grid 24
Number of grids	100×100	100×100	100×100	100×100	100×100
Length grid size	10	2	2	2	2
Coarse grid	non	Grid 01	Grid 01	Grid 01	Grid 01
Grid size ratio	non	5	5	5	5
Time step in sec	0.020	0.01	0.01	0.01	0.01
SWEs	linear	linear	linear	linear	linear
CFL condition	0.7	0.7	0.7	0.7	0.7

Table 8.5: The information on the set up of the different grids at multiple levels for linear SWEs

The following figures show l_2 - RE of free surface elevation between a coarse grid and four fine grids (21, 22, 23 and 24).

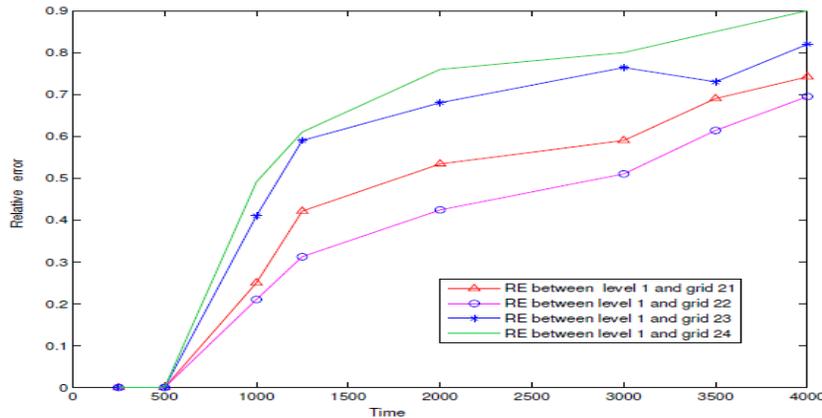


Figure 8-9: Show l_2 - RE between a coarse grid and four fine grids (21, 22, 23 and 24)

Results in grid 22 \implies Results in grid 21 \implies Results in grid 23 \implies Results in grid 24

Example 2:

By the same previous example, we use system of linear 2DSWEs given in Section 2.6 with non-rotate $f=0$, wind stress=0 to find l_2 - RE of free surface elevation in case a coarse grid which contains one fine grid in level 2 that contains again one fine grid in level 3. Also, a fine grid in level 3 contains again one fine grid in level 4 at difference times $t=500, 1000, \dots, 5000$ sec.

The information on the set up of the different grids for linear 2DSWEs are given below:

Information	Grid 01	Grid 21	Grid 32	Grid 43
Number of grids	100×100	100×100	100×100	100×100
Length grid size	10	2	0.40	0.08
Coarse grid	non	Grid 01	Grid 21	Grid32
Grid size ratio	non	5	5	5
Time step in sec	0.020	0.01	0.005	0.0025
SWEs	non-linear	non-linear	non-linear	non-linear
CFL condition	0.7	0.7	0.7	0.6

Table 8.6: The information on the set up of the different grids at multiple levels for linear 2DSWEs

The following figures show l_2 -RE of free surface elevation between a coarse grid-level 2 (grid 21), level 2-level 3 (grid 32) and level 3-level 4 (grid 43).

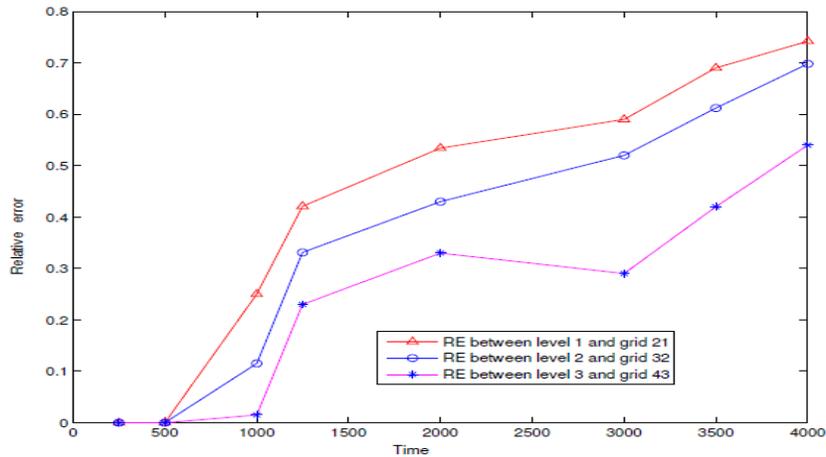


Figure 8-10: Comparison l_2 -RE of free surface elevation between a coarse grid-level 2 (grid 21), level 2-level 3 (grid 32) and level 3-level 4 (grid 43)

Example 3:

By the same previous example, we use system of linear 2DSWEs with non-rotate $f=0$, wind stress=0 to find l_2 -RE of free surface elevation in case a coarse grid which contains one fine grid (child embedded or separate in parent) in level 2 that contains again two fine grids in level 3. Also, the fine grids in level 3 contain again one fine grid in level 4 at difference times $t= 500, 1000, \dots, 5000$ sec.

The information on the set up of the different grids for linear 2DSWEs are given below:

Information	Grid 01	Grid 22	Grid 31	Grid 33	Grid 44
Number of grids	100×100	100×100	100×100	100×100	100×100
Length grid size	10	2	0.40	0.40	0.08
Coarse grid	non	Grid 01	Grid 22	Grid22	Grid33
Grid size ratio	non	5	5	5	5
Time step in sec	0.020	0.01	0.005	0.005	0.0025
CFL condition	0.7	0.7	0.7	0.6	0.6

Table 8.7: The information on the set up of the different grids at multiple levels for linear SWEs

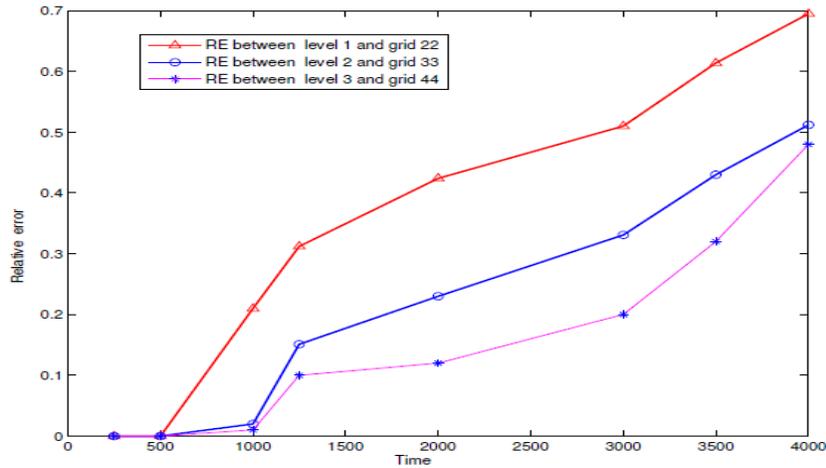


Figure 8-11: Comparison $l2-RE$ between a coarse grid-level 2 (grid 22), level 2-level 3 (grid 33) and level 3-level 4 (grid 44)

8.4.1 Comparison $l2-RE$ of free surface elevation for cases separate grids and embedded grids

Example 1:

By the same previous example, we use system of linear 2DSWEs with non-rotate $f=0$, wind stress=0 to find $l2-RE$ of free surface elevation in case a coarse grid contains one fine grid (grid 21) in level 2 which contains again one fine grid in level 3 (child separable or embedded in parent) at difference times $t= 500, 1000, \dots, 5000$ sec.

Information	Grid 01	Grid 21	Grid 32
Number of grids	100×100	100×100	100×100
Length grid size	10	2	.40
Coarse grid	non	Grid 01	Grid 21
Grid size ratio	non	5	5
Time step in sec	0.025	0.0125	0.00625
SWEs	non-linear	non-linear	non-linear

Table 8.8: The information on the set up of different grids for linear SWEs

The following figures show l_2 -RE of free surface elevation between a fine grid in level 2 and level 3 (grid 32) in cases separate and embedded grids.

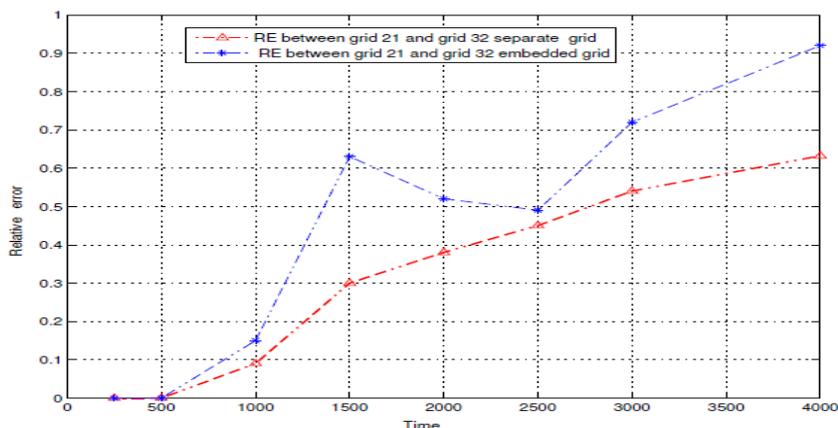


Figure 8-12: Comparison l_2 -RE between a fine grid in level 2 and level 3 (grid 32) in cases separate and embedded grids.

8.4.2 Comparison the results of l_2 -RE when use the time step are 0.010 and 0.030 For linear SWEs

Example 1:

By the same previous example, we use system of linear 2DSWEs with non-rotate $f=0$, wind stress=0 to find l_2 -RE of free surface elevation when the time steps are (0.010 and 0.030) in case a coarse grid contains one fine grid (grid 22) in level 2 which contains again two fine grids in level 3. Also, a fine grid in level 3 contains again one fine grid in level 4 at difference times $t=500, 1000, \dots, 5000$ sec.

The information on the set up of the different grids for linear 2DSWEs are given below:

Information	Grid 01	Grid 22	Grid 31	Grid 33	Grid 44
Number of grids	100×100	100×100	100×100	100×100	100×100
Length grid size	10	2	0.40	0.40	0.08
Coarse grid	non	Grid 01	Grid 22	Grid22	Grid33
Grid size ratio	non	5	5	5	5
Time step in sec	0.010/0.030	0.005/ 0.015	0.0025 /0.0075	0.0025/0.0075	0.00125 /0.00375
SWEs	non-linear	non-linear	non-linear	non-linear	non-linear
CFL condition	0.7	0.7	0.7	0.6	0.6

Table 8.9: The information on the set up of the different grids at multiple levels for linear 2DSWEs

The following figures show l_2 -RE of free surface elevation between a coarse grid-level 2 (grid 22), grid 22-level 3 (grid 33) and level 3 (grid 33)-level 4 (grid 44) in case the time steps are 0.010 and 0.030.

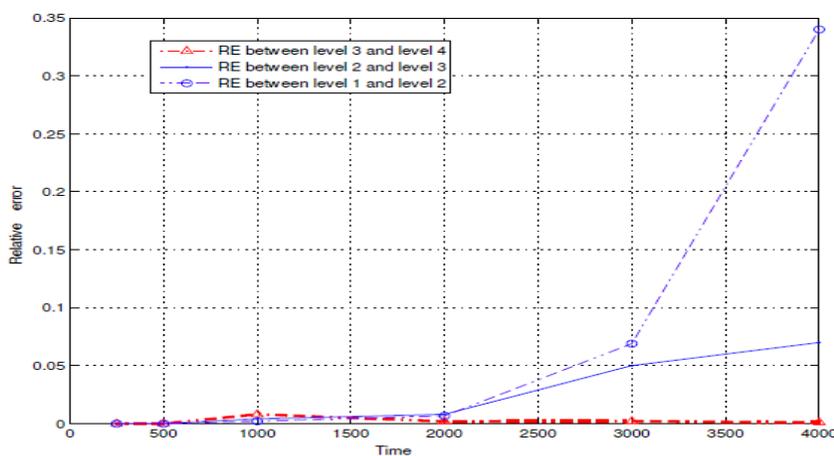


Figure 8-13: Comparison l_2 -RE for linear 2DSWEs when time step 0.010

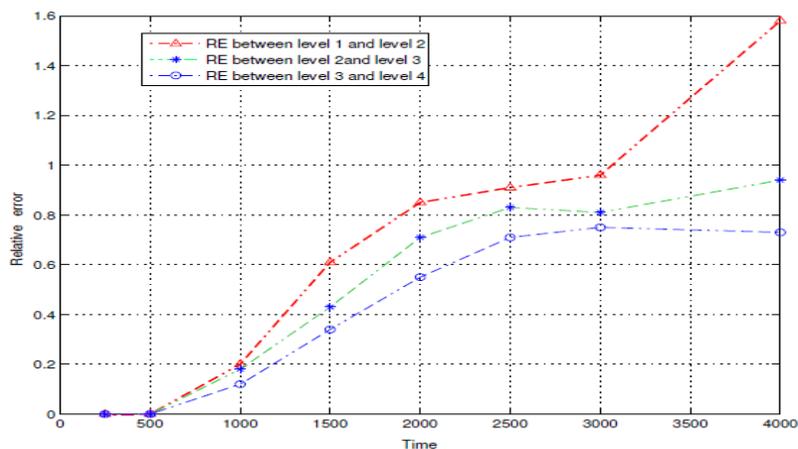


Figure 8-14: Comparison l_2 -RE for linear 2DSWEs when time step 0.030

8.4.3 Comparison the results of $l2-RE$ when the space refinement ratio are 1:3 and 1:5 in multiple levels for NSWEs

Example 1:

Notes that: The space refinement ratio in levels 2 and 3 is 1:5 and in level 4 is 1:3

In this example, we use system of 2DNSWEs given in Section 2.4 with (ν , non-rotate $f=0$, wind stress=0) to find $l2-RE$ of free surface elevation in case a coarse grid contains one fine grid in level 2 which contains again one fine grid in level 3. Also, a fine grid in level 3 contains again two fine grids in level 4 at different times $t=500, 1000, \dots, 5000$ sec.

The information on the set up of the different grids for 2DNSWEs are given below:

Information	Grid 01	Grid 24	Grid31	Grid 41
Number of grids	100×100	100×100	100×100	100×100
Length grid size	10	2	0.40	0.133
Coarse grid	non	Grid 01	Grid 24	Grid 31
Grid size ratio	non	5	5	3
Time step in sec	0.025	0.0125	0.00625	0.003125
SWEs	non-linear	non-linear	non-linear	non-linear
CFL condition	0.7	0.7	0.7	0.6

Table 8.10: The information on the set up of the different grids at multiple levels

The following figures show $l2-RE$ of free surface between a coarse grid-level 2 (grid 24), level 2-level 3 and level 3-level 4 (grids 41).

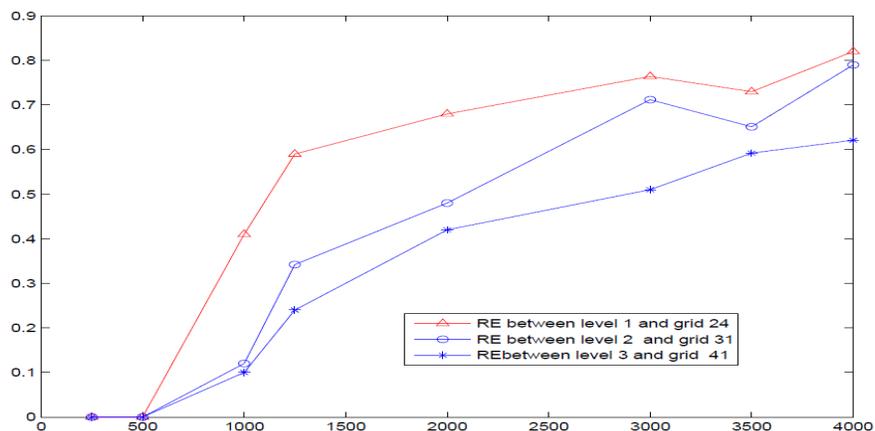


Figure 8-15: Comparison $l2-RE$ between a coarse grid-level 2 (grid 24), level 2-level 3 (grid 31) and level 3-level 4 (grids 41)

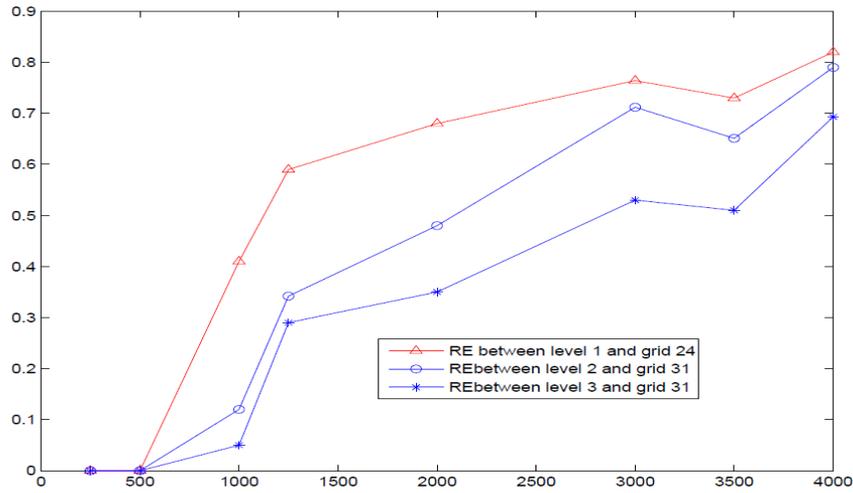


Figure 8-16: Comparison l_2 -RE of free surface elevation between a coarse grid-level 2 (grid 24), level 2-level 3 (grid 31) and level 3-level 4 (grids 45) embedded with itself

8.5 Case 3: Coupling multiple systems when the space refinement ratio 1:5 with no time refinement

8.5.1 Example 1 : Case 1 : Coupling four systems for 2DNSWEs

In this example, we find l_2 -RE of free surface elevation for 2DNSWEs ($\nu=0$, wind stress and $f=0$), when a coarse grid contains only one fine grid at difference times $t=500, 1000, \dots, 4000$ sec. The information about the coarse and fine grids are given in table below :

Information	Grid 01	Grid 24	Grid 33	Grid 46
Number of grids	100×100	100×100	100×100	100×100
Length grid size	10	2	0.40	0.08
Coarse grid	non	Grid 01	Grid 24	Grid 33
Grid size ratio	non	5	3	3
Time step in sec	0.020	0.01	0.005	0.0025
SWEs	non-linear	non-linear	non-linear	non-linear
CFL condition	0.7	0.7	0.6	0.6

Table 8.11: The information on the set up of the different grids at multiple levels for 2DNSWEs

In this case. Firstly, when a coarse grid located in 1st-level, applying EFDM to approximate 2DNSWEs. Secondly, applying EFDM to approximate 2DNSWEs in a 2nd-level. Then, coupling a coarse grid with a fine grid in a 2nd-level grid. Thirdly, applying EFDM to approximate 2DNSWEs.

Then, coupling a fine grid in a 2nd-level with the fine grid in the 3rd-level. Fourthly, applying EFDM to approximate 2DNSWEs. Then, coupling a fine grid in the 3rd-level with the fine grid in the 4th-level.

The following figure Compares $l2-RE$ of free surface elevation between a coarse grid-level 2 with refinement ratio 1:5, level 2-level 3 and level 3-level 4 with space refinement ratio 1:3.

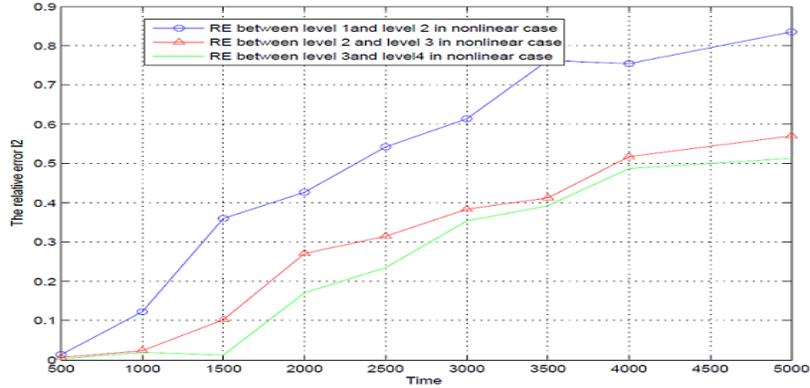


Figure 8-17: Comparison $l2-RE$ between level 1-level 2, level 2-level 3 and level 3-level 4

8.5.2 Example 2: Case 2: Coupling four systems for 2D non-linear SWEs

In this example, we find $l2-RE$ of free surface elevation for 2DNSWEs ($\nu = 0$, the wind stress =0, $f=0$) when a coarse grid contains more than one level at different times $t=500, 1000, \dots, 4000$ sec. The information about the coarse and fine grids are the same previous example:

In this case, Firstly, when a coarse grid in 1st-level, applying EFDM to approximate 2DNSWEs. Secondly, applying EFDM to approximate 2DNSWEs. Then, coupling a coarse grid with the fine grid in a 2nd-level grid. Thirdly, applying EFDM to approximate linear 2DSWEs. Then, coupling a fine grid in 2nd-level with the fine grid in a 3rd-level. Fourthly, applying EFDM to approximate 2DNSWEs. Then, coupling a fine grid in the 3rd-level with the fine grid in the 4th-level.

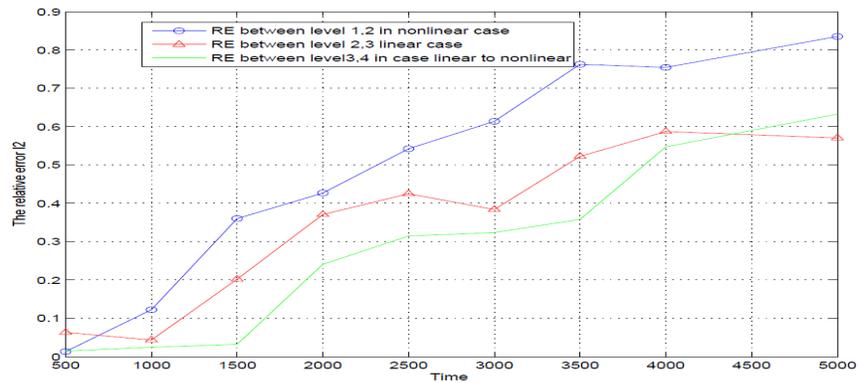


Figure 8-18: Comparison $l2-RE$ between level 1-level 2 in case nonlinear, level 2-level 3 in case nonlinear to linear and level 3-level 4 in case linear to nonlinear SWEs.

8.5.3 Example 3: Case 3: Coupling four systems for 2D nonlinear / linear SWEs

In this example, we find l_2 -RE of free surface elevation for 2DNSWEs ($\nu = 0$, wind stress =0, $f=0$) when a coarse grid contains more than one level at different times $t=500, 1000, \dots, 4000$ sec. The information about the coarse and fine grids are the same previous example.

In this case, Firstly, when a coarse grid in 1st-level, applying EFDM to approximate 2DNSWEs. Secondly, applying EFDM to approximate linear 2DSWEs. Then, coupling a coarse grid with the fine grid in a 2nd-level. Thirdly, applying EFDM to approximate 2DNSWEs. Then, coupling the fine grid in 2nd-level with the fine grid in the 3rd-level. Fourthly, applying EFDM to approximate linear 2DSWEs. Then, coupling the fine grid in the 3rd-level with the fine grid in the 4th-level.

The following example shows l_2 -RE of free surface elevation between a coarse grid-fine grid in level 2 nonlinear to linear case, fine grid in level 2-fine grid in level 3 linear to nonlinear case and level 3-level 4 nonlinear to linear case.

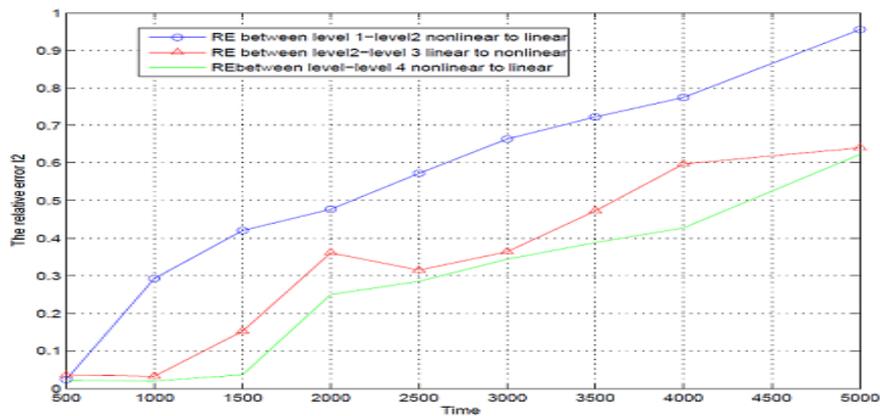


Figure 8-19: Comparison l_2 -RE between a coarse grid-fine grid in level 2 nonlinear to linear case, fine grid in level 2-fine grid in level 3 linear to nonlinear case and level 3-fine grid in level 4 nonlinear to linear case.

8.6 Summary and Conclusions

This chapter dealt with new techniques for multiple nested grids that were applied to the tsunami model. To verify the efficiency of the nested grid model, several of numerical examples were proposed with the nesting 5:1. The accuracy of the nested grid depends on the location of the open boundaries of the nested domains. Boundaries should be located in areas of the coarse grid domain where high accuracy to minimize the error being passed from coarse to fine grids. It was showed the performance techniques using two-way nesting with the ratio 5:1, when applied some examples for non-linear 2DSWEs at multiple levels.

Some examples for coupling four systems at multiple levels of 2DSWEs have been tested for three cases when space refinement ratio equals 1:5 and 1:3 with no refinement of time. L_2 -relative error results using different time steps were compared. In general, good results were observed.

Chapter 9

Recommendations (Future Works)

1. Apply two-way interaction for multiple nested grids in some applications of life such that weather and climate models in Coordinates spherical using an explicit finite difference methods or finite volume methods, implicit finite difference methods.
2. Apply a two-way nesting technique using an explicit finite difference method and leapfrog with Robert-Asselin filter with radiative open boundary condition.
3. Apply a new technique of two-way nesting grids in some application life for example hydrodynamic model for the port of new york for 3DSWEs with moving boundary condition.
4. Study an existence of solutions for shallow water models with non-homogenous boundary conditions, global weak solution and smoothness.
5. Apply two-way interaction technique for multiple nested grids of 2DSWEs when a specific domain rotation is required using the explicit methods.
6. Apply two-way interaction technique for SWEs using irregular geometry boundary conditions.
7. Study two-way nesting grid for ocean models when the water depth discontinuous with moving boundary conditions.

Bibliography

- [1] [https://code.google.com/p/shel/downloads / list](https://code.google.com/p/shel/downloads/list). 2010.
- [2] Altaie.Huda. *New Techniques Of Derivations for Shallow Water Equations*. International Journal of Advanced Scientific and Technical Research , 6, 3, 131-151, 2016.
- [3] Altaie.Huda. *A Two-Way Nesting for Shallow Water Model*. International Review of Physics (I.RE.PHY.) , 2 (4), 105-111., 2017.
- [4] Altaie.Huda. *Application for the 2D shallow water equations of the tsunami model*. Research and Reports on Mathematics, 2018.
- [5] Altaie.Huda. *A Multiply Nested Model for Non-Linear Shallow Water Model*. Research and Reports on Mathematics, 1:2, 2018.
- [6] Altaie.Huda. *Numerical Model for Nested Shallow Water equations*. International Journal of Pure and Applied Mathematics, 118,n.4, 2018.
- [7] Altaie.Huda and Fabrice Planchon. *Results of higher accuracy for 2D shallow water equations with (separate/adjacent)for structured grids, submit*. 2018.
- [8] Altaie.Huda and Pierre Dreyfuss. *Numerical Solutions for 2D Depth-averaged Shallow Water Equations*. International Mathematical Forum, 13 (2), 79-90, 2018.
- [9] John D. Anderson. *Fundamentals of Aerodynamics*. 2nd Edition McGraw-Hill, New York, 1991.
- [10] A. Arakawa and V.R. Lamb. *Computational design of the basic dynamical processes of the UCLA general circulation model*. Methods in Computational Physics, 17, 173-265, 1977.
- [11] Arakawa.A. *Computational design for long-term numerical integration of the equations of fluid motion: Two-dimensional incompressible flow*. Journal of Computational Physics, 1, 119-143, 1997.
- [12] R. Asselin. *Frequency filter for time integrations*. Mon. Wea. Rev, 100, 487-490, 1972.

- [13] M.J Berger and J. Olinger. *Adaptive mesh refinement for hyperbolic partial differential equations*. Journal of Computational Physics, 53, 484-512, 1984.
- [14] G.E. Birchfield. *Numerical prediction of hurricane movement with the use of fine grid*. Journal of Meteorology, 17, 406-414, 1960.
- [15] Culbert B.Laney. *Computational Gasdynamics*. Cambridge University Press, Chapter 2, 5-21, 1989.
- [16] Debreu.L; Blayo.E and Barnier. B. *A general adaptive multi-resolution approach to ocean modelling: experiments in a primitive equation model of the North Atlantic*. Ocean modelling, 303-313, 2005.
- [17] A. F. Blumberg and G. L. Mellor. *A coastal ocean numerical model*. Mathematical Modelling of Estuarine Physics. Lecture Notes on Coastal and Estuarine Studies, 1, Springer-Verlag, 203-219, 1980.
- [18] A. F. Blumberg and G. L. Mellor. *Diagnostic and prognostic numerical circulation studies of the South Atlantic Bight*. Journal of Geophysical Research, 88, C8, 4579-4592, 1983.
- [19] A.Biastoch; CW. Boning and JR. Lutjeharms. *Agulhas leakage dynamics affects decadal variability in atlantic overturning circulation*. Nature international journal of science, 456, 489-492, 2008.
- [20] Stephen R. Carpenter and David M Lodge. *Effect of submersed mocrphttes on ecosystem processes* Aquatic Botony, 26, 341-370, 1986.
- [21] J.E. Castillo. *Mathematical aspects of grid Generation*. Society for Industrial and applied Mathematics, Chapters 1 and 7, 1991.
- [22] V. Casulli and R. T. Cheng. *Semi-implicit finite difference methods for three-dimensional shallow water flow*,. International Journal for Numerical Methods in Fluids, 15 (6), 629-648, 1992.
- [23] C.B.Verugdenhil. *Numerical Method of Shallow Water Flow*. Springer-Kluwer Academic publishers, Chapter 2 ; 15-35, Institute for marine and Atmospheric , Utrecht University , Netherlands, 2009,.
- [24] C. Chen. *A nested grid, Nonhydrostatic, Elastic model Using a terrain following coordinate transformation: the radiative-nesting boundary conditions*. Monthly Weather Review, 119, 2852-2869, 1991.

- [25] Terry L. Clark and R. D. Farley. *Severe downslope windstorm calculations in two and three spatial dimensions using anelastic interactive grid nesting*, *Journal of the Atmospheric Sciences*, 41(3), 329-350. 1984.
- [26] Peter Crowhurst. *Numerical Solutions of One-Dimensional Shallow Water Equations*. Australian mathematical Science institute, 1-24, 2013.
- [27] C.Rowley and I.Ginis. *Implementation of a mesh movement scheme in a multiply nested ocean model and its application to air-sea interaction studies*. *Monthly Weather Review*; 127: 1879-1896., 1999.
- [28] John C.Strikwerda. *Finite Difference Scheme and PDEs*. Society for industrial and applied mathematics, 2004.
- [29] Benoit Cushman-Roisin and Jean-Marie Beckers. *Introduction To Geophysical Fluid Dynamics*. Academic press, Thayer School of Engineering Dartmouth College, USA, 2007.
- [30] C.Zoppou and S. Roberts. *Catastrophic collapse of water supply reservoirs in urban areas*. *Journal of Hydraulic Engineering*, vol 125(7): 686-695, July, 1999.
- [31] Andreas Dedner. *Computational PDEs, Chapter 2*. April , 2011.
- [32] A.G.Dawson; D.Long and D.E.Smith. *The storegga slides: Evidence from eastern scotland for a possible tsunami*. *Marine Geology*, 82(3-4), 271-276, 1988.
- [33] A. Duran and F. Marche. *Discontinuous-Galerkin discretization of a new class of Green-Naghdi equations*. *Communications in Computational Physics*, p.130, 2014.
- [34] Dale R. Durran. *The third-order Adams Bashforth method:An Attractive Alternative to Leapfrog Time Differencing*,. *Mon. Wea. Rev*, 119, 702-720, 1991.
- [35] Dale R. Durran. *Numerical Methods For Fluid Dynamics With Applications To Geophysics*. Springer Science, Business Media, University of Washington, Department of Atmospheric Sciences, U.S.A, Chapter 3, 89-141, 2010.
- [36] Blayo .E and Debreu .L. *Revisiting open boundary conditions from the point of view of characteristic variables*. *Journal of Ocean Modeling*, 9 (3), 231-252, 2005.
- [37] Edward.J and Harrison.Jr. *Three-dimensional numerical simulations of tropical systems utilizing nested finite grids*. *J. Atmos. Sci.*, 30, 1528-1543, 1973.
- [38] Juri Elken. *Lecture notes ,Dynamical oceanography,Chapter 5 ,30-50*. 2007.

- [39] Julie Pietrzak et al. *A three-dimensional hydrostatic model for coastal and ocean modelling using a generalised topography following co-ordinate system*. Ocean Modelling, 4(2), 173-205, 2002.
- [40] A.B. Kennedy et al. *Boussinesq modeling of wave transformation breaking and runup II: 2D*. Journal of Water way Port Coastal and Ocean Engineering, 126(1), 39-47, 2000.
- [41] A.Barth et al. *Two-way nested model of mesoscale circulation features in the Ligurian Sea*. Progress in Oceanography, 66, 171-189, 2005.
- [42] Adcroft. A et al. *Implementation of an Atmosphere Ocean General Circulation Model on the Expanded Spherical Cube*. Monthly Weather Review, 132, 2845-2863, 2004.
- [43] Agoshkov.V.I et al. *Mathematical and numerical modelling of shallow water flow*. Comput. Mech., 11:280-299., 1993.
- [44] Bengt Andersson et al. *Computational Fluid Dynamics for Engineers*. United Kingdom at the University Press, Cambridg, 2012.
- [45] Benque.JP et al. *Mathematical and numerical modelling of shallow water flow*. J. Waterw. Port Coast. Ocean Division Proc ASCE., 108: 396-417., 1982.
- [46] Bondevik Stein et al. *Record-breaking height for 8000-year-old tsunami in the North Atlantic*. Eos. Transactions American Geophysical Union, 84(31), 289-293, 2003.
- [47] G.Sannino et al. *An eddy-permitting model of the mediterranean sea with a two-way grid refinement at the strait of gibraltar*. Ocean Model. 30 (1), 56-72., 2009.
- [48] Laurent Debreu et al. *Two-Way embedding algorithms for a split-explicit free surface ocean model. In preparation for Ocean Model*. Journal of Ocean Modeling, 49-50, 1-21, (2012).
- [49] P. Brufau et al. *The shallow water equations: An example of hyperbolic system*. Real Academia de Ciencias de Zaragoza, 31, 89-119, 2008.
- [50] Pierrick Penven et al. *Evaluation and application of the ROMS 1-way embedding procedure to the central California upwelling system*. Ocean Modelling, 12, 157-187, 2006.
- [51] Schlichting et al. *Boundary layer theory*,. Springer publishing house Berlin Heidelberg, chapter 3, 53-88, 1997.
- [52] Zhang et al. *A Two-Way Interactive Nesting Procedure with Variable Terrain Resolution*, Monthly weather review , 14, 1330-1339. 1986.

- [53] Jinyu Shenget et.al. *A new two-way nested technique for ocean modelling based on the smoothed semi-prognostic method*. Ocean dynamics, 55, 162-177, 2005.
- [54] Ferrari.S and Saleri.F. *A new two dimensional shallow water model including pressure effects and slow varying bottom topography*. M2AN, 38: 211-234., 2004.
- [55] Flather.R.A. and N.S.Heaps. *Tidal computations for Morecambe Bay*. Geophysical Journal of the Royal Astronomical Society, 42, 489-517., 1975.
- [56] Alan D. Fox and Stephen J. Maskell. *Two way interactive nesting of primitive equation ocean model with topography*. Journal of Physical Oceanography, 25, 2977-2996., 1995.
- [57] David L. George. *Numerical Approximation of the Nonlinear Shallow Water Equations with Topography and Dry Beds*. Master of Science, University of Washington, 2004.
- [58] P.L. George. *Automatic Mesh Generation, Wiley*. 1991.
- [59] David A. Greenberg. *Modeling the mean barotropic circulation in the Bay of Fundy and Gulf of Maine*. Journal of Physical Oceanography, vol 13, 886-904, 1983.
- [60] Jean-Michel Hervouet. *Hydrodynamique des écoulements à surface libre: Modélisation numérique avec la méthode des éléments finis*. Paris: Presses de l'Ecole Nationale des Ponts et Chaussées, 2003.
- [61] Herzfeld.M. *The role of numerical implementation on open bound- ary behaviour in limited area ocean models*. Ocean Modelling , 27, 1-2, 18-32, 2009.
- [62] W.R. Holland. and G.Vallis. *A high resolution model of the California Current embedded into a basin-scale North Pacific circulation model*,. 1990.
- [63] Roberto Camassa; Darryl D. Holm and James M. Hyman. *A new integrable shallow water equation*. Comparison with Saint-Venant and Boussinesq systems. Discrete Contin., Adv. Appl. Math., 31: 1-33, 1994.
- [64] https://en.wikipedia.org/wiki/Leibniz_integral_rule.
- [65] J.Berntsen and E.Svendsen. *Using the Skagex dataset for evaluation of ocean model skills*. Journal of Marine Systems ; 18 (4): 313-331. 1999.
- [66] Marchesiello. P; McWilliams. JC and Shchepetkin. A. *Equilibrium structure and dynamics of the California Current System*. J. Phys. Oceanogr. 33, 753-783, 2003.

- [67] Robert W. Jones. *A nested grid for a three-dimensional model of a tropical cyclone*. J. Atmos. Sci., 34, 1528-1553, 1977a.
- [68] J.W.Elder. *The dispersion of marked fluid in turbulent shear flow*. Journal of Fluid Mechanics , 5(4), 544-560, 1959.
- [69] Lakshmi H. Kantha and Carol Anne Clayson. *Numerical Models of Oceans and Oceanic Processes*. Academic Press ,Chapters 2 and 3, 2000.
- [70] G. Korres and A. Lascaratoe. *A one-way nested eddy resolving model of the Aegean and Levantine basins:implementation and climatologically runs*. Annales Geophysicae, 21, 205-220, 2003.
- [71] Z. Kowalik and T.S. Murty. *Numerical Modeling of Ocean Dynamics*,. Advanced Series on Ocean Engineering, vol.5, Chapter 3, 1993.
- [72] D. Lannes and F. Marche. *A new class of fully nonlinear and weakly dispersive Green- Naghdi models for efficient 2D simulations*. Journal of Computational Physics, 238-268, 2014.
- [73] H.M. Cekirge; R.W. Lardner and R.J.raga. *Adaptation of the solution of the two-dimensional tidal equations using the method of characteristics to wind induced currents and storm surges*. Computers and Mathematics with Applications, 12A, 10, 1081-1090, 1986.
- [74] L.Debreu and E.Blayo. *Two-way embedding algorithms: a review*. Ocean dynamics, 58, 415-428, 2008.
- [75] J. J. Leendertse. *Aspects of a Computational Model for Long-Period Water- Wave Propagation*,. The Rand Corporation, Report RM-5294-PR, 1967.
- [76] Jingming Hou; Qihua Liang and Xilin Xia. *Robust absorbing boundary conditions for shallow water flow models*. Environ Earth Sci , 74, 7407-7422, September 2015.
- [77] H.W. Liepmann and A. Roshko. *Elements Of Gasdynamics*. Courier Corporation, New York, 1957.
- [78] Zhonglu Lin. *Modelling Generation and Propagation of 2009 Samoa-Tonga Tsunami*. Ph.D Thesis,University of Cambridge, 2015.
- [79] L.Simon. *Contribution to the modelling of surface water flow and transport*. Ph.D.Thesis, Paris, France, Ecole Nationale des Ponts et Chaussees, 1994.
- [80] L. Mansinha and D. E. Smylie. *The displacement fields of inclined faults*. Bull. Seism.Soc. Am., 61, 1433-1440, 1971.

- [81] James L. Martin and Steven C. McCutcheon. *Hydrodynamics and transport for water quality modelling*. Lewis Publishers, Chapter 1, 1999.
- [82] M.A.Spall and A.R.Robinson. *A hybrid coordinate open ocean primitive equation model, Mathematics and Computers in simulation*. 31, 241-269, 1989.
- [83] Kiran Alapaty; Rohit Mathur and Talat Odman. *Intercomparison of spatial interpolation schemes for use in nested grid models, Monthly Weather Review* , 126: 243-249. 1998.
- [84] M.B. Mathur. *A multiple grid primitive equation model to simulate the development of an asymmetric hurricane*. J. Atmos. Sci., 31, 371-393, 1964.
- [85] Karyn Matthews. *A Spherical Coordinates Tidal Model of the Great Australian Bight using a new Coastal Boundary Representation*. Ph. D. thesis, The Department of Applied Mathematics, The University of Adelaide, South Australia,p.35, 1996.
- [86] D.J. Mavriplis. *Mesh Generation and adaptivity for complex geometries and flows*. Handbook of Computational Fluid Mechanics, Chapter 7, 417-459, 1996.
- [87] N.Goutal and F.Maurel. *A finite volume solver for 1D shallow-water equations applied to an actual river*. International Journal for Numerical Methods in Fluids, 38: 1-19, 2002.
- [88] N.S.Heaps. *A two-dimensional numerical sea model*. Philosophical Transactions of the Royal Society of London. Series A. Mathematical and Physical Sciences, 265, 93-137, 1969.
- [89] N.S.Heaps. *Three-dimensional numerical model of the Irish Sea*. Geophysical Journal. Royal Astronomical Society, 35, 99-120, 1973.
- [90] LY. Oey and P. Chen. *A nested grid ocean model with application to the simulation of meanders and eddies in the Norwegian Coastal Current*. Journal of Geophysical Research, 97, 20063-20086, 1992.
- [91] M. Okada. *Surface deformation due to shear and tensile faults in a half-space*. Bull. Seism.Soc. Am., 75(4),1135-1154, 1985.
- [92] Peter J. Olver. *Numerical Analysis Lecture Notes ,page (188-209)*. 2008.
- [93] Y. Ookochi. *A computational scheme for the nesting fine mesh in the primitive equation model*. journal of meteorology Society of Japan, 50, 37-47, 1972.
- [94] P.Angot and M.Laugier. *The FIC method of conservation connection between nested subdomains for an ocean circulation model, Comptes Rendus de l' Academie des Sciences Paris ; 319(Serie II):993-1000*. 1994.

- [95] S.K.Dube; P.C.Sinha and G.D. Roy. *The numerical simulation of storm surges along the Bangla Desh coast*. Transactions of the American Mathematical Society, 9, 121-133, 1985.
- [96] J. Pedlosky. *Geophysical Fluid Dynamics*. Springer, Second edition, 22-56, 1987.
- [97] T. Warner; R. Peterson and R. Treadon. *A tutorial on lateral boundary conditions as a basic and potentially serious limitation to regional numerical weather prediction*. Bull.Amer. Meteor. Soc., 78, 2599-2617, 1997.
- [98] N. Phillips and J. Shukla. *On the Strategy of Combining Coarse and Fine Grid Meshes in Numerical Weather Prediction*, 12(5), 329-350. J. Appl. Meteor., 1973.
- [99] J. Pullen and J.S. Allen. *Modeling studies of the coastal circulation off northern California: Statistics and patterns of winter time flow*,. Journal of Geophysical Research, vol 106, NO.C11, 26, 959-26, 984, 2001.
- [100] Flather. R.A. *A tidal model of the northwest European continental shelf*. Mem. Soc. R. Sci. Liege, 10(6), 141-164, 1976.
- [101] Prasada Rao. *A moving domain formulation for modeling two-dimensional open channel transient flows*. Applied Mathematics and Computation, 154, 3, 769-781, 2004.
- [102] I. Ginis; R. A. Richardson and L. M. Rothstein . *Design of a multiply nested primitive equation ocean model*. Monthly Weather Review, 126, 1054-1079, 1998.
- [103] Guillaume Riflet and Ramiro Neves. *Developing a shallow-waters finite-differences numerical model to study convectively dominated flows near the boundaries*. International Conference on Engineering Optimization, September 6-9, Lisbon, Portugal, 2010.
- [104] G.S. Rodenhuis. *Two dimensional nearly horizontal flow models*. In:Abbot, N.B. and Price, W.A. (Eds), Coastal, estuarial, and harbour engineers' reference book, Routledge, UK., 129-144, 1994.
- [105] J. Sainte-Marie and M.O. Bristeau. *Derivation of a non-hydrostatic shallow water model. Comparison with Saint-Venant and Boussinesq systems*. Discrete Contin., Dyn. Syst. Ser. B, 10(4):733-759, 2008.
- [106] Koch S.E. and McQueen J.T. *A survey of nested grid techniques and their potential for use within the MASS weather prediction model*.NASA Technical Memorandum 87808, National Aeronautics and Space Administration, USA, N87-18944. 1987.
- [107] Alexander F. Shchepetkin. *An Adaptive, Courant-number-dependent Implicit Scheme for Vertical Advection in Oceanic Modeling*. Ocean Modeling , 91, 38-69, 2015.

- [108] F.Imamura; N. Shuto and C.Goto. *Propagation of tsunamis*. the Sixth Congress of the Asian and Pacific Regional Division, Int. Assoc. Hydraul. Res., Kyoto, Japan, 1988.
- [109] Spall.M.A and WR Holland. *A nested primitive equation model for ocean applications*,. Journal of physical oceanography, 21, 205-220, 1991.
- [110] G.S. Stelling. *On the construction of computational methods for shallow water flow problem*. Thesis Applied Sciences, 1984.
- [111] Einar Svendse et al. *Model simulation of the Skagerrak circulation and hydrography during SKAGEX* , *Journal of Marine Systems*; 8: 3-4, 219-236. 1996.
- [112] Vidar Thomee et al. *From finite differences to finite elements: A short history of numerical analysis of partial differential equations*. Journal of Computational and Applied Mathematics, 128(1-2), 1-54, 2001.
- [113] Lloyd N. Trefethen. *Finite Difference and Spectral Methods for Ordinary and Partial Differential Equations*. Chapter 4, 146, 1994.
- [114] X. Wang. *manual of comcot*. Cornell University, 2007.
- [115] X. Wang and Philip L.F.Liu. *Numerical simulations of the 2004 indian ocean tsunamis - coastal effects*. Journal of Earthquake and Tsunami, 1(3), 273-297, 2007.
- [116] X. Wang and P. L.F. Liu. *An analysis of 2004 sumatra earthquake fault plane mechanisms and indian ocean tsunami*. J. Hydraulic Res., 44(2), 147-154, 2006.
- [117] Weiyan. *Shallow water hydrodynamics*,. Amesterdam, water and power press, Elsevier, 1992.
- [118] John F. Wendt. *Computational Fluid Dynamics: An Introduction*,. Thired edition, Chapter 2, 26-35, 2009.
- [119] P. D. Williams. *Achieving seventh-order amplitude accuracy in leapfrog integrations* , *Monthly Weather Review*, 141 (9),3037-3051. 2013.
- [120] P.D. Williams. *A proposed modification to the Robert-Asselin time filter*,. Mon. Wea. Rev, 137, 2538-2546, 2009.
- [121] Philip L.F.Liu; Seung-Buhm Woo and Yong-Sik Cho. *Computer programs for tsunami propagation and inundation*. Technical report, Cornell University, 1998.
- [122] Gregory J.Tripoli Yoshio Kurihara and Morris A.Bender. *Design of a movable nested-mesh primitive equation model*. Monthly Weather Review, 107, 239-249, 1979.

Appendix A

A.1 Open boundary conditions

1. **Southern boundary condition $j=1$ and $i=2:ix-1$** . For the elevation and the component of velocity to the boundary is defined

$$\eta_{i,j} = \frac{\sqrt{\frac{(u_{i,j}+u_{i-1,j})^2}{2} + (v_{i,j})^2}}{\sqrt{gH_{i,j}}}$$

2. **Northern boundary condition $j=jy$ and $i=2:ix-1$** . For the elevation and the component of velocity to the boundary is defined

$$\eta_{i,j} = \frac{\sqrt{\frac{(u_{i,j}+u_{i-1,j})^2}{2} + (v_{i,j-1})^2}}{\sqrt{gH_{i,j}}}$$

3. **Western boundary condition $i=1$ and $j=2: jy-1$** . For the elevation and the component of velocity to the boundary is defined

$$\eta_{i,j} = \frac{\sqrt{\frac{(v_{i,j}+v_{i,j-1})^2}{2} + (u_{i,j})^2}}{\sqrt{gH_{i,j}}}$$

4. **Eastern boundary condition $i=1$ and $j=2: jy-1$** . For the elevation and the component of velocity to the boundary is defined

$$\eta_{i,j} = \frac{\sqrt{\frac{(v_{i,j}+v_{i,j-1})^2}{2} + (u_{i-1,j})^2}}{\sqrt{gH_{i,j}}}$$

5. Spacial cases:

1. If $i=1$ and $j=1$. For the elevation and the component of velocity to the boundary is defined:

$$\eta_{i,j} = \frac{\sqrt{(v_{i,j})^2 + (u_{i,j})^2}}{\sqrt{gH_{i,j}}}$$

2. If $i=ix$ and $j=1$. For the elevation and the component of velocity to the boundary is defined :

$$\eta_{ix,1} = \frac{\sqrt{(v_{ix,1})^2 + (u_{ix-1,1})^2}}{\sqrt{gH_{ix,1}}}$$

3. If $i=1$ and $j=j_y$. For the elevation and the component of velocity to the boundary is defined :

$$\eta_{1,j_y} = \frac{\sqrt{(v_{1,j_y-1})^2 + (u_{1,j_y})^2}}{\sqrt{gH_{1,j_y}}}$$

4. If $i=i_x$ and $j=j_y$. For the elevation and the component of velocity to the boundary is defined :

$$\eta_{i_x,j_y} = \frac{\sqrt{\frac{(\eta_{i_x-1,j_y}) + (\eta_{i_x,j_y-1})}{2}}}{\sqrt{gH_{i_x,j_y}}}$$

A.2 Boundary Scheme

Initially, The free surface displacement as well as the fluxes values, is zero in all grids at the beginning.

Figure A-1 explains the scheme for moving boundary given by ([115, 121]). The SWL denotes the mean water level and η represents the free water displacement. The grid point on the dry land is called water depth h , is negative and its absolute value is the elevation of the landscape, measured from MWL (Mean Water Level). Therefore, the total depth $H = h + \eta$ has a negative value in a dry cell and a positive value in a wet cell.

The free surface displacements, η at the next time step in the entire computational domain (i.e. both dry and wet cells) are determined by continuity equation together with boundary conditions along offshore boundaries. The volume fluxes, Hu are zero at the grid points along the shoreline. Consequently, the free surface displacements, η are also zero at a dry grid. However, a numerical algorithm is required to decide whether the total water depth is high enough to flood the dry cells next to the wet cells and then move the shoreline.

Here, the one-dimensional example is applied to demonstrate the moving boundary algorithm. The volume flux, Hu , is calculated at grid points $i - 1/2$ and $i + 1/2$ while the water depth, h , and free water displacement, η are computed at $i-1$, i , and $i + 1$. In time step $t = n\Delta t$, the i cell is a wet cell with positive total depth while the $i+1$ cell is dry with negative total depth. The shoreline is therefore between the i cell and the $i+1$ cell. Hence, the $i+1/2$ grid point has zero volume flux. As a result, the shoreline does not move onshore.

On the other hand, if the water surface surges as on in the right side of Figure A-1, the value of volume flux, Hu , is no longer zero. Then, the shoreline may move one grid onshore. The next step is to compute the total depth H_{i+1} from the continuity equation.

After that, the following algorithm can finally decide whether the shoreline should be moved.

1. If $H_i > 0$ possible cases can be summarized as follows:
2. If $H_{i+1} \leq 0$ and $h_{i+1} + \eta_i \leq 0$ then the shoreline remains between grid points i and $i + 1$ and the volume flux $(Hu)_{i+1/2}$ remains zero.

3. $H_{i+1} \leq 0$ and $h_{i+1} + \eta_i > 0$ then the shoreline moves to between grid points $i + 1$ and $i + 2$ and the volume flux $Hu_{i+1/2}$ may have a nonzero value while $(Hu)_{i+3/2}$ is assigned to be zero. The flooding depth is $H_f = h_{i+1} + \eta_i$.
4. If $H_{i+1} > 0$ then the shoreline moves to between grid points $i + 1$ and $i + 2$. The volume flux $(Hu)_{i+1/2}$ may have a nonzero value while $(Hu)_{i+3/2}$ has a zero value and the flooding depth is $H_f = \max(h_{i+1} + \eta_i, h_{i+1} + \eta_{i+1})$.

Let now derive the algorithm in two-dimension problem and The corresponding y-direction algorithm has the same procedure as that for the x direction.

1. If $H_{i,j} > 0$ and $H_{i+1,j} > 0$ then $H_f = \frac{1}{2}(H_{i,j} + H_{i+1,j})$ where

$$H_f = \frac{1}{4}(H_{i,j} + H_{i+1,j} + H_{i,j} + H_{i+1,j+1})$$

possible cases can be summarized as:

- If $H_{i,j} > 0$ and $H_{i+1,j} > 0$ and $h_{i+1,j} + \eta_{i,j} > 0$ then $HH = h_{i+1,j} + \eta_{i,j}$
- 2. $H_{i,j} \leq 0$ and $H_{i+1,j} > 0$ and $h_{i,j} + \eta_{i+1,j} > 0$ then $HH = h_{i,j} + \eta_{i+1,j}$.

Else

$$H_{u(i,j)} = 0$$

Now, algorithm in y- direction has the same procedure as that for the x direction.

1. If $H_{i,j} > 0$ and $H_{i,j+1} > 0$ then $H_{Q1} = \frac{1}{2}(H_{i,j} + H_{i,j+1})$

$$H_{Q2} = \frac{1}{4}(H_{i,j} + H_{i+1,j} + H_{i,j+1} + H_{i+1,j+1})$$

possible cases can be summarized as:

- If $H_{i,j} > 0$ and $H_{i,j+1} < 0$ and $h_{i,j+1} + \eta_{i,j} > 0$ then $HH = h_{i,j+1} + \eta_{i,j}$.
- 2. $H_{i,j} \leq 0$ and $H_{i,j+1} > 0$ and $h_{i,j} + \eta_{i,j+1} > 0$ then $HH = h_{i,j} + \eta_{i,j+1}$.

Else $H_{v(i,j)} = 0$

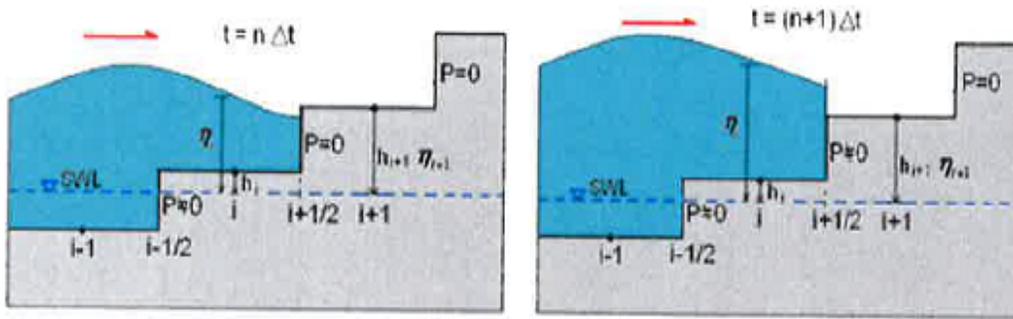


Figure A-1: A sketch of moving boundary scheme

Résumé

Les écoulements en eau peu profonde se rencontrent dans de nombreuses situations d'intérêts: écoulements de rivières et dans les lacs, mais aussi dans les mers et océans (courants de marée, tsunami, etc.). Ils sont modélisés par un système d'équations aux dérivées partielles, où les inconnues sont la vitesse de l'écoulement et la hauteur d'eau. On peut supposer que la composante verticale de la vitesse est petite devant les composantes horizontales et que ces dernières sont indépendantes de la profondeur. Le modèle est alors donné par les équations de shallow water (SWEs).

Cette thèse se concentre sur la conception d'une nouvelle technique d'interaction de plusieurs grilles imbriquées pour modèle en eau peu profonde en utilisant des méthodes numériques. La première partie de cette thèse comprend, La dérivation complète de ces équations à partir des équations de Navier-Stokes est expliquée. Etudier le développement et l'évaluation des méthodes numériques en utilisant des méthodes de différences finies et plusieurs exemples numériques sont appliqués utilisant la condition initiale du niveau gaussien pour 2DSWEs.

Dans la deuxième partie de la thèse, nous sommes intéressés à proposer une nouvelle technique d'interaction de plusieurs grilles imbriquées pour résoudre les modèles océaniques en utilisant quatre choix des opérateurs de restriction avec des résultats de haute précision.

Notre travail s'est concentré sur la résolution numérique de SWE par grilles imbriquées. A chaque niveau de résolution, nous avons utilisé une méthode classique de différences finies sur une grille C d'Arakawa, avec un schéma de leapfrog complété par un filtre d'Asselin. Afin de pouvoir affiner les calculs dans les régions perturbées et de les alléger dans les zones calmes, nous avons considéré plusieurs niveaux de résolution en utilisant des grilles imbriquées. Ceci permet d'augmenter considérablement le rapport performance de la méthode, à condition de régler efficacement les interactions (spatiales et temporelles) entre les grilles.

Dans la troisième partie de cette thèse, plusieurs exemples numériques sont testés pour 2DSWE avec imbriqués 3:1 et 5:1. Finalement, la quatrième partie de ce travail, certaines applications de grilles imbriquées pour le modèle tsunami sont présentées.

Mots clés : Grille imbriquée, méthode des différences finies explicite, modèle d'eau peu profonde, dérivation des équations 2D d'eau peu profonde, modèle océan, conditions aux limites, les flux de surface libres, hydrostatique pression, les régions côtières, fluide Incompressible, dynamique des fluide, Dépôt de tsunami, Risques de tsunami, Interface dynamique.

