A Two Dimensional Vertically Integrated Moving Boundary Hydrodynamic Model in Curvilinear Coordinates.

Vibhas Aravamuthan

Louisiana State University and Agricultural & Mechanical College

Follow this and additional works at: https://digitalcommons.lsu.edu/gradschool_disstheses

Recommended Citation

https://digitalcommons.lsu.edu/gradschool_disstheses/5933

This Dissertation is brought to you for free and open access by the Graduate School at LSU Digital Commons. It has been accepted for inclusion in LSU Historical Dissertations and Theses by an authorized administrator of LSU Digital Commons. For more information, please contact gradetd@lsu.edu.
INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI
A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
313/761-4700  800/521-0600
A TWO DIMENSIONAL VERTICALLY INTEGRATED MOVING BOUNDARY HYDRODYNAMIC MODEL IN CURVILINEAR COORDINATES

A Dissertation

Submitted to the Graduate Faculty of the Louisiana State University and Agricultural and Mechanical College in partial fulfillment of the requirements for the degree of Doctor of Philosophy

in

The Department of Civil and Environmental Engineering

by

Vibhas Aravamuthan
B.Tech., Indian Institute of Technology, Madras, 1985
May 1995
Acknowledgements

If at all this dissertation was possible, the credit goes to my advisor, Dr. J.N. Suhayda, Associate Professor of Civil Engineering, who guided me, supported me and encouraged me during my doctoral program. The invaluable discussions I had with him have thrown a major success factor into this study. I would also like to thank him for providing financial support throughout my stay as a doctoral student.

I would like to thank Professor V.P. Singh for the many invaluable discussions I had with him on both academic and non academic matters. I would also like to thank the other members of my doctoral advisory committee Dr. D.D. Adroit, Dr W.J. Wiseman, Dr. J.F. Cruise, Dr. M. Alawady and Dr. S.S. Iyengar for reviewing this dissertation and offering their helpful suggestions.

Financial support provided by the Department of Civil Engineering, LSU, and the U.S. Geological Survey, during the course of this study is gratefully acknowledged.

I wish to thank my friends Ganesh, Siva, Larry, Sankar, and Krishnan for the many useful discussions I have had with them during the course of this study. A special word of thanks goes to my best friend, and the greatest UNIX - C wizard I know of, Mr. Ramana Rao for the many enlightening discussions I had with him.

My friends Hegde, Jana, Shibu, Anwar, Rajendran, Srikanth, Loga, and many others made my life away from school a wonderful experience which I gratefully acknowledge.
My brother Sarang, sister Manjari, brother-in-law Raghu, nephew Nishadh, cousin Hemanth and aunt Padmaja deserve a special word of thanks for their constant encouragement throughout the course of my studies.

My parents Aravamuthan and Vanaja spent most of their life pouring love, encouragement and moral support throughout my life. Words cannot express my thanks to them. I dedicate this dissertation to them.
Contents

Acknowledgements .................................................................................................... ii

List of Tables ........................................................................................................ vi

List of Figures ....................................................................................................... vii

Abstract ................................................................................................................ xi

1 Introduction ........................................................................................................... 1
  1.1 Objectives and scope of the study ............................................................. 4
  1.2 Arrangement of this report ........................................................................ 5

2 Literature Survey .................................................................................................. 6
  2.1 Review of 2-D models ................................................................................ 8
     2.1.1 Finite element models ........................................................................ 8
     2.1.2 Finite difference methods ............................................................... 9
     2.1.3 Moving boundary models .............................................................. 13
  2.2 Summary ....................................................................................................... 16

3 Methodology ....................................................................................................... 18
  3.1 Introduction .................................................................................................. 18
  3.2 Boundary fitted coordinate method ......................................................... 19
     3.2.1 Algebraic grid generators ....... ......................................................... 22
        3.2.1.1 Transfinite interpolation ........................................................... 23
     3.2.2 Partial differential equation based grid generators ......................... 24
        3.2.2.1 Elliptic grid generator ................................................................ 25
     3.2.3 Adaptive grid generator .................................................................. 31
        3.2.3.1 Generation of patched grids ...................................................... 33
  3.3 Hydrodynamic model ................................................................................... 36
     3.3.1 Hydrodynamic equation in Cartesian Coordinates .......................... 37
     3.3.2 Transformation of governing equations to curvilinear Coordinate-
         nates ...................................................................................................... 38
     3.3.3 Characteristic analysis of the governing equations in curvilinear
         coordinate system. .............................................................................. 39
     3.3.4 Transformation of hydrodynamic equations to a moving curvi-
         linear coordinate system ................................................................... 42
     3.3.5 Boundary condition formulation ...................................................... 43
     3.3.6 Moving boundary formulation ......................................................... 45
3.4 Finite difference formulation ............................................................. 47
  3.4.1 Discretization of metric components ........................................ 53
  3.4.2 Discretization of boundary condition ........................................ 55
    3.4.2.1 Closed boundary represented by a solid wall ........ 55
    3.4.2.2 Open inflow boundary ............................................. 56
    3.4.2.3 Open outflow boundary .......................................... 56
    3.4.2.4 Closed moving boundary ....................................... 56
  3.5 Stability analysis ........................................................................... 58
  3.6 Solution algorithm ....................................................................... 64

4 Model Testing ....................................................................................... 65
  4.1 Comparison with an analytical solution with friction effect for a re-
    ctangular basin .............................................................................. 65
  4.2 Comparison with an analytical solution in a parabolic basin .... 71

5 Applications .......................................................................................... 79
  5.1 Tide induced circulation in Lake Pontchartrain ....................... 79
    5.1.1 Data description .................................................................... 79
    5.1.2 Implementation details ......................................................... 84
    5.1.3 Analysis of results .............................................................. 99
  5.2 Tide induced circulation in Bayou Chitigue .............................. 101
    5.2.1 Implementation details ......................................................... 102
    5.2.1.1 Calibration .................................................................... 109
    5.2.1.2 Verification ................................................................. 115
    5.2.2 Analysis of results .............................................................. 118

6 Conclusions ........................................................................................... 119
  6.1 Scope for future work ..................................................................... 120

References ........................................................................................... 121

Vita ......................................................................................................... 127
List of Tables

5.1 Tidal Constituents ................................................. 80
5.2 Average constituent amplitude of tidal elevations (feet) .......... 80
5.3 Local epoch (degrees) of each tidal constituent ..................... 81
List of Figures

3.1 (a) An irregular physical domain (b) The corresponding transformation into a computational domain .............................................................. 20

3.2 (a) Grid generated in the physical domain for converging section (b) The transformed rectangular computational domain ............................. 28

3.3 (a) Grid generated in a circular region connected to a rectangle (b) The transformed rectangular computational domain ........................................ 29

3.4 (a) Grid generated in the physical domain for a natural watershed located in Hastings, Nebraska (b) The transformed rectangular computational domain .......................................................... 30

3.5 Adaptive grid for the diagonal problem using the elliptic grid generator, 41 X 41 mesh, A=3.0, B=3.0 ........................................................... 34

3.6 Adaptive grid for a pill box problem using the elliptic grid generator, 41 X 41 mesh, A=10.0, B=10.0 .................................................................. 34

3.7 (a) Splitting of the physical domain into different zones for Lake Pontchartrain (b) The transformed rectangular computational domain 35

3.8 Characteristic cone .......................................................................................... 41

3.9 Finite difference grid showing the location of variables ......................... 48

4.1 Rectangular basin closed at three ends ......................................................... 68

4.2 Comparison of analytical and numerical solutions for water level elevations at a point 2500 m from the open boundary ............................ 69

4.3 Comparison of analytical and numerical solutions for water level elevations at a point 5000 m from the open boundary ............................ 69

4.4 Comparison of analytical and numerical solutions for water level elevations at a point 7500 m from the open boundary ............................ 69

4.5 Comparison of analytical and numerical solutions for u - velocity at a point 2500 m from the open boundary ................................. 70

4.6 Comparison of analytical and numerical solutions for u - velocity at a point 5000 m from the open boundary ................................. 70
4.7 Comparison of analytical and numerical solutions for u-velocity at a point 7500 m from the open boundary ................................................. 70
4.8 Grid in physical domain for the circular basin ................................. 74
4.9 The computational domain for the circular basin ............................... 74
4.10 Comparison of analytical and numerical solutions for water level elevations at grid point (10,10) .................................................. 75
4.11 Comparison of analytical and numerical solutions for water level elevations at grid point (15,15) ......................................................... 75
4.12 Comparison of analytical and numerical solutions for water level elevations at grid point (18,18) ......................................................... 75
4.13 Comparison of movement of the X-coordinate of grid point (1,10) located on the western boundary ...................................................... 76
4.14 Comparison of movement of the Y-coordinate of grid point (1,10) located on the western boundary ...................................................... 76
4.15 Comparison of movement of the X-coordinate of grid point (21,10) located on the western boundary ...................................................... 77
4.16 Comparison of movement of the Y-coordinate of grid point (21,10) located on the western boundary ...................................................... 77
5.1 Map of Lake Pontchartrain showing tide gage locations ................. 81
5.2 Grid in physical domain for Lake Pontchartrain ............................... 82
5.3 The computational domain for Lake Pontchartrain .......................... 82
5.4 Input tidal elevations at the mouth of The Rigolets ....................... 83
5.5 Input tidal elevations at the mouth of Chef Menteur Pass .............. 83
5.6 Observed and predicted tidal elevations at location P2 ................. 85
5.7 Observed and predicted tidal elevations at location P3 ................. 86
5.8 Observed and predicted tidal elevations at location P4 ................. 87
5.9 Observed and predicted tidal elevations at location P5 ................. 88
5.10 Observed and predicted tidal elevations at location P7 ............... 89
5.11 Influence of time step on water level elevation at location P4 ................................. 90
5.12 Grid at the end of 22.5 Hrs of simulation time .......................................................... 91
5.13 Grid at the end of 656 Hrs of simulation time .......................................................... 91
5.14 Boundary locations at the end of 0 Hrs, 22.5 Hrs and 656 Hrs of simulation time ................................................................................................................................. 92
5.15 Velocity vectors at the end of 22.5 Hrs of simulation time ........................................... 92
5.16 Velocity vectors at the end of 180 Hrs of simulation time ........................................... 93
5.17 Velocity vectors at the end of 656 Hrs of simulation time ........................................... 93
5.18 Tidal ellipses at four points traversing from north to south at mid bay........................... 94
5.19 Movement of grid point (1,10) located on the western boundary as a function of time ................................................................................................................................. 95
5.20 Movement of grid point (7,17) located on the northern boundary as a function of time ................................................................................................................................. 95
5.21 Movement of grid point (7,1) located on the southern boundary as a function of time ................................................................................................................................. 96
5.22 Effect of boundary slope on the movement of grid point (1,10) located on the western boundary as a function of time ................................................................. 96
5.23 Effect of boundary slope on the movement of grid point (7,17) located on the northern boundary as a function of time ................................................................. 97
5.24 Effect of boundary slope on the movement of grid point (7,1) located on the southern boundary as a function of time ................................................................. 97
5.25 Water depth as a function of time at a point located 100 m from grid point (1,10) on the western boundary ................................................................. 98
5.26 Water depth as a function of time at a point located 200 m from grid point (1,10) on the western boundary ................................................................. 98
5.27 Water depth as a function of time at a point located 400 m from grid point (1,10) on the western boundary ................................................................. 98
5.28 Map of Bayou Chitigue showing tide gage and current meter locations............................. 102
5.29 Grid in physical domain for Bayou Chitigue ................................................................ 103
5.30 The computational domain for Bayou Chitigue .................................. 103
5.31 Input tidal elevations in the bayou (calibration) .............................. 104
5.32 Comparison of predicted and observed u-velocity at the mouth of the pond at location G1 (calibration) .............................................................. 105
5.33 Comparison of predicted and observed v-velocity at the mouth of the pond at location G1 (calibration) .............................................................. 106
5.34 Comparison of predicted and observed water level elevations at the mouth of the pond at location G1 (calibration) ......................................................... 107
5.35 Comparison of predicted and observed water level elevations on the surface of the marsh at location G2 (calibration) ........................................... 108
5.36 Input tidal elevations in the bayou (verification) .............................. 110
5.37 Comparison of predicted and observed u-velocity at the mouth of the pond at location G1 (verification) .............................................................. 111
5.38 Comparison of predicted and observed v-velocity at the mouth of the pond at location G1 (verification) .............................................................. 112
5.39 Comparison of predicted and observed water level elevations at the mouth of the pond at location G1 (verification) ......................................................... 113
5.40 Comparison of predicted and observed water level elevations on the surface of the marsh at location G2 (verification) ........................................... 114
5.41 Grid at the end of 74.0 hours of simulation time .............................. 115
5.42 Grid at the end of 96.0 hours of simulation time .............................. 116
5.43 Marsh pond boundary locations at the end of 74 and 96 hours of simulation time .............................................................. 116
5.44 Velocity vectors at the end of 74 hours of simulation time ................ 117
5.45 Velocity vectors at the end of 96 hours of simulation time ................ 117
Abstract

A two dimensional, vertically averaged, hydrodynamic mathematical model on an adaptive boundary fitted grid is developed. This model addresses numerous drawbacks in the traditional finite difference models and incorporates such features as moving boundaries due to tidal incursion, and the ability to resolve irregular coastline geometries. A semi-implicit finite difference scheme is developed in order to overcome time step restrictions due to the gravity wave stability criterion. An eulerian-lagrangian formulation is used to solve the hydrodynamic equations on unsteady grid systems. The model is applied to study circulation features in Lake Pontchartrain (a medium sized physical system) and to the Bayou Chitigue channel pond system (a very small physical system), to demonstrate its ability to model systems of varying sizes and shapes.
Chapter 1

Introduction

The coastal areas of the world are thickly populated, and rely a great deal on the resources available there. The coastal wetlands of Louisiana, in particular are an extremely valuable natural resource in terms of fisheries and mineral deposits. Exploitation of these resources without regard to environmental consequences has resulted in the deterioration of these wetlands. It is estimated that nearly 50 square miles of wetlands are lost every year, resulting in the loss of habitat for plants and animals.

Marshes can be broadly classified into three categories based on the type of vegetation present. These are saltwater marshes, brackish marshes and freshwater marshes. This classification is based on the ambient salinity the vegetation can tolerate, with saltwater marshes tolerating large values of salinity and freshwater marshes tolerating very low values of salinity. Wetlands, in general, and Louisiana wetlands in particular, are characterized by the presence of small ponds dotted amongst the marsh vegetation, and it is often difficult to demarcate land water interfaces. These small ponds are extremely shallow and are connected by a myriad of small and large channels; some natural and others man made. A combination of natural and human activity has resulted in the deterioration of these marshes. One of the many reasons is the change in salinity caused by the construction of canals for oil and gas operations. These fluctuations in salinity result in the demise of surrounding vegetation, which in turn results in increased soil erosion. One of
the important factors contributing to vegetation loss is the duration of exposure of these plants to salinity changes. These fluctuations in salinity can be attributed to several physical phenomena such as tides, winds and increased discharge due to floods. The long term effects are generally due to tidal fluctuations. Although, these plants can endure sharp fluctuations in salinity for a short period of time, prolonged exposure affects them in an adverse fashion, and ultimately results in their death. Thus the duration of inundation is an important factor affecting these plants. With an increase in environmental awareness in the past few years, efforts are underway to preserve wetlands. This has prompted a deep interest in the study of coastal processes such as coastal circulation, sediment transport, pollutant transport and hurricane storm surges.

Natural processes occurring in the coastal zone are governed by certain physical laws. Based on these laws it is possible to develop models which can forecast the impact of various natural phenomena on the coastal zone. In general, such models can be broadly classified as 1) physical models and 2) mathematical models. The physical models are scaled down versions of the actual areas modeled. These models are extremely complicated to fabricate and are very expensive in terms of both cost and manpower. Such modeling efforts are undertaken by only a few research centers around the world. With the advent of digital computers, the mathematical model has gained much popularity as a predictive tool. The primary reason being that these models are extremely flexible and cost effective, and for most studies, a physical model is not required.

Over the past two decades, a number of mathematical models have been developed to model various water bodies around the world. The essential components of a mathematical model are the governing equations of mass, momentum, sediment
transport, pollutant transport etc. Depending on the physical phenomena modeled, some or all of the above components are used in the formulation of the model. The governing equations along with appropriate boundary conditions, are then solved by any of a number of available numerical techniques such as finite difference, finite element, boundary element, and boundary fitted coordinate method. Two of the more popular techniques used in the past are the finite difference and the finite element methods.

The finite difference techniques have enjoyed greater popularity due to their simplicity and rigorous mathematical basis regarding stability, consistency and convergence properties. The major drawback of this method is the inadequate representation of the irregular physical boundary. Conventional finite difference schemes approximate the irregular boundary in a 'stair case' fashion. This results in errors in the prescription of the boundary conditions, and thus induces errors in model predictions.

Investigators recognizing the above deficiencies of the finite difference method started using the finite element method. This method is capable of treating irregular physical domains. The main disadvantage of this method lies in its inherent complexity in problem formulation, and computer implementation. The other disadvantage is a poor understanding of such important factors as stability, convergence and the consistency of the solution algorithms.

Most of the numerical models developed, have concentrated primarily on large open water bodies. Mathematical modelling of wetlands has largely been ignored due to their inherent complexity in terms of very small area, very shallow depths, and very complicated physical shapes. The traditional finite difference and finite element models are unable to treat these complexities in an efficient way.
A recent technique, which is gaining popularity especially in the field of computational aerodynamics, is the boundary fitted coordinate method. In this method, the irregular physical domain is mapped on to a rectangular computational domain. The governing equations of the model are also transformed to the computational domain, and traditional finite difference techniques can then be used to solve model equations. Using an adaptive version of this method, it is possible to concentrate grid lines in areas displaying high gradients of velocity, pressure etc., and thus eliminate the use of nested computational grids in the model. It is also possible to incorporate moving boundaries into this method. The boundary fitted coordinate method thus combines the advantages of both finite difference and finite element methods and is thus ideally suited for modelling flows in regions of arbitrary shape.

1.1 Objectives and scope of the study

The objective and scope of the present study are:

1. Develop a two dimensional vertically integrated hydrodynamic model to study flow properties in physical regions of arbitrary shape.

2. Incorporate the ability to handle moving boundaries, by dynamically coupling a grid generator with the hydrodynamic model.

3. Incorporate the ability to handle very small and shallow physical regions.

4. Develop a semi implicit discretization scheme to remove gravity wave stability criteria, and to mathematically analyze its stability properties.

5. Compare the model with existing analytical solutions.
6. Implement the model to study flow patterns in a medium sized water body (Lake Pontchartrain) and a very small scale water body (Bayou Chitigue).

1.2 Arrangement of this report

In chapter 2, a brief review of literature on two dimensional models is presented. The drawbacks of the current state of the art in terms of applicability to wetland hydrodynamics problems is also discussed.

In chapter 3 the mathematical model is described in detail. Discussions include the grid generation method, the hydrodynamic model, and the coupling between the hydrodynamic model and the grid generator to obtain a versatile model capable of handling moving boundaries in a continuous manner.

In chapter 4, the model is tested by comparing it with analytical solutions.

In chapter 5, the model is implemented to study flow patterns in Lake Pontchartrain and Bayou Chitigue.
Chapter 2

Literature Survey

The use of numerical models in hydrodynamics has become increasingly popular in recent years. This is partly attributed to increasing environmental concerns and the rapid advancement and availability of computer hardware and software. This has resulted in easy to use numerical models and a class of model users has emerged distinct from model makers. We presently enjoy an array of packaged models intended for various hydrodynamic uses with sophisticated user interfaces and anticipate the development and distribution of increasingly complex models.

The fundamental equations governing the flow of fluids have been known for some time. Basically, we must deal with the conservation equations of mass and momentum, modified to reflect the hydrostatic and Boussinesq assumptions; in more difficult cases, these are coupled via an equation of state to conservation equations for heat, salt and possibly water quality components. Our confidence in the validity of these equations, rests on their basis in first principles, with only the turbulent exchanges treated in an empirical fashion. These equations are highly nonlinear and in general, detailed analytical solutions for complex systems with arbitrary initial and boundary conditions are not possible and one must resort to some approximate solution which can, in principle be refined to an arbitrary precision.

Based on the number of spatial dimensions used, numerical models can be broadly classified as one dimensional (1-D), two dimensional (2-D) or three dimensional (3-D) models. These models can be either unsteady or steady state models.
depending on whether time variations of the hydrodynamic variables are included or not. In general, unsteady models are used except for very simple engineering designs such as drainage ditches, and culverts.

One dimensional models are generally used to study long narrow water bodies such as rivers, channels, and straits where lateral and vertical variations of flow properties are not important. The importance of 1-D models cannot be overlooked, as for a number situations such as channel design, river stage forecasting etc., these models perform reasonably well. Several U.S. agencies like the Waterways Experiment Station (WES), United States Geological Survey (USGS), Coastal Engineering Research Center (CERC) etc. offer models for public use. Reviews of the capabilities of these models have been contributed by Thomas (1979), Feldman (1981) and Fread (1982).

Three dimensional models are used to study detailed flow properties in all three spatial directions. In general, 3-D models are computationally very expensive to run, and typically require large amounts of field data for calibration. In general 3-D models require about 10 times the computational effort of 2-D models and powerful super computers are required to run these models. The enhanced cost associated with these models precludes their use for medium scale engineering projects.

Based on cost effectiveness, ease of use and good detailed flow descriptions, 2-D models have become very popular with engineers and scientists. A brief review of existing literature on 2-D models is given below.
2.1 Review of 2-D models

Two dimensional models are generally used to study water bodies with large surface area or very deep narrow channels where vertical variation of flow variables is important. Depth integrated models are the most popular 2-D models for studying flow patterns in the X-Y plane. Based on the type of solution method used 2-D models can be classified into finite element (FE) models, finite difference (FD) models and models using the boundary fitted coordinate method (BFC).

2.1.1 Finite element models

Finite element models use the integral form of the governing equations solved on triangular or quadrilateral grids. The procedure consists of choosing a trial solution, and minimizing the residuals over the entire domain. The FE elements are generally linear and this results in a set of algebraic equations which are then solved by either direct or iterative methods. A number of FE methods have been reported in the literature. Grotkop (1973) calculated the oscillation of the North Sea due to the semi-diurnal tide. This analysis employed the finite element Galerkin technique in both space and time. This method has been found to be extremely time consuming (Nihoul 1976). Implicit finite element schemes for nonlinear models have been applied by Wang and Connor (1975) to study the Great Bay Estuary in New Hampshire. The main advantage of the finite element technique over the finite difference method is the ability to describe the bathymetry and lateral topography more accurately, but the technique has been found to be inefficient for transient problems (Pinder and Gray (1977)). For a steady state problem, the efficiency of the finite element method is comparable to the finite difference method. Hence, some
investigators have used the Fourier transform method to transform the transient equations into time independent forms and then apply the finite element technique (Pinder and Gray (1977)). The disadvantages of the FE method are that they are difficult to formulate and are computationally very expensive to solve. This is because the resulting algebraic equations are not sparse and efficient sparse matrix algorithms cannot be used to solve them. The mathematical tools for analysis of stability, consistency and convergence of the FE method for transient problems are yet to be developed. In spite of these drawbacks FE methods are used because of their ability to represent arbitrary geometries accurately.

2.1.2 Finite difference methods

The FD method is very popular with modelers because of its ease of use, rigorous mathematical basis, and lower computational overhead. The FD method typically consists of discretization of the governing equations on rectangular grids. The difference equations are then solved to obtain the spatial and temporal variations of the flow variables.

The finite difference method can be broadly classified into two categories: (1) direct methods and (2) characteristics method (Amein and Fang 1969). These categories are further classified into explicit and implicit methods. Each method has many variations depending on exactly how the partial differential equations are transformed into the finite difference analogues. Explicit direct and implicit direct methods are most common in tidal and storm surge models. For the sake of convenience, they are referred to as explicit and implicit methods in this study.
A characteristic method is suitable for tracing the disturbance of waves or the movement of fluid constituents. It requires some form of intermediate transformation when the solutions are required at fixed locations.

The explicit method has been used quite often for shallow water simulations (Fischer (1965), Gates (1966), Reid and Bodine (1968), Matthews and Mungall (1972)). Murty (1984) presents a review of various versions of the explicit method. The computations involved in this method are straightforward in that the unknown quantities at every grid point are solved for based on known values at the previous time step. Matrix inversion for solving a system of simultaneous equations is thus avoided. However, the time step used in this method is restricted by the gravity wave stability criterion also known as the Courant-Friedrichs-Lewy (CFL) criterion. This criterion states that no combination of signals can travel more than one grid point during a time step (Crowley (1970)). The stability analyses are generally performed on linearized versions of the equations as it is very difficult to perform such analyses on nonlinear equations. The stability criterion is given by $\Delta t = \Delta s / \sqrt{2gh}$ where $\Delta t$ is the allowable time step, $\Delta s$ is the minimum grid-spacing, $g$ is the acceleration due to gravity, and $H$ is the maximum water depth. The ratio $(\Delta t)/2gh/\Delta s$ is termed the Courant number.

By using an implicit method, unknowns in the finite difference equations are expressed in terms of other unknown values. Together with boundary conditions, a set of algebraic equations is solved to evaluate those related unknowns at one time. The time step is limited not by the stringent Courant stability criterion, but by accuracy considerations. An implicit method requires more computer storage than an explicit method. The computation time per time step is also greater for the implicit method if the same time step is used for both methods. However, the
implicit method permits a larger time step in the computations. Consequently the total computation time can be considerably reduced if an implicit method is used with a larger time step. In this case, the phase error associated with the implicit method should be taken into consideration (Leendertse (1967)).

According to the experience of some investigators (Diprima and Rogers (1969), Weare (1976)), the implicit method when applied to nonlinear systems, is, in general, not free from stability problems. The choice of an implicit or explicit method depends on the nature of the problem solved, and also on the availability of computer resources.

A two-dimensional model can be either vertically integrated or transversely integrated depending on the physics of the problem. A vertically averaged model calculates water movements on the horizontal X- and Y-directions. The currents in oceans or coastal areas are usually calculated through this type of a model. A transversely averaged model calculates flows in the X- and Z-directions. It can be used for calculating flows in estuaries where no significant variations in the current occur in the lateral direction. This is typically the case in long narrow estuaries. The velocities in both X- and Y-directions have to be considered if the estuarine geometry is broad.

When the implicit method is applied to a one-dimensional flow with N grid points, a set of N simultaneous arithmetic equations have to be solved at every time step. For a two-dimensional problem with M by N grid points, a set of (M X N) equations have to be solved at every time step. In a real problem, a significant amount of computer memory is required to store several (M X N) matrices where M and N are of the order of a hundred. Clearly, this is an enormous task requiring a lot of computation time. In order to overcome this problem, Peaceman and Rachford
(1955) proposed the alternating-direction-implicit (ADI) method. This method is related to that developed by Douglas (1955) for solving the two dimensional heat equation.

In the ADI method, a computational interval is divided into two half time steps. In the first half time step, computations are carried out row by row. A set of $N$ simultaneous equations are solved to evaluate the $N$ unknowns at each row. All unknowns are limited to the same row. Unknown terms in the other rows are substituted by their corresponding known values obtained at a previous time step. In the next half time step, similar operations are carried out column by column.

Gustafsson (1971) applied the ADI method to solve the equations of a shallow water problem. Weare (1976) pointed out the disadvantage of using an ADI method due to increased truncation errors resulting from the time splitting of the difference equations, especially when modelling irregular geometries.

Many vertically integrated, two dimensional, numerical models have been developed. Some of them have simplified the problem by ignoring one or more terms in the governing equations. Some of the models have used linearized equations (Heaps (1969)). If the advective terms are included, the equations become nonlinear and the computations become more troublesome. The role of nonlinear terms in the numerical instability has been of great interest to many investigators (Moe, Mathison and Hodgins (1978)). Nonlinear terms cannot be neglected in studying the gyre structures induced by circulations and hence these terms are retained in the present study.

The work of Leendertse (1967) constituted an important landmark for the modeling of two dimensional flows due to the fact that the model allowed a relatively large time step for simulations by using the implicit method. The author's claim
that the model is unconditionally stable has attracted much attention. Leendertse extended this model to perform water quality simulations. Hess and White (1974) applied it to Narragansett Bay. Tee (1976) used a similar model including the effect of eddy viscosity. Blumberg (1977a) applied a similar model to Chesapeake Bay. Stelling (1984) improved the stability of Leendertse’s method by using an angled derivative approach to discretize the advection term.

In some cases, a high density network is required at local areas either to supply detailed results or to include important information, while a coarse network may suffice in other areas. Such a model is used by the Federal Emergency Management Agency (FEMA) for calculating coastal flooding due to storm surges. A model with varied grid spacings (Bryan (1966), Abbott et. al. (1973)) may save some computation time without losing important features at local areas. Butler (1978b, 1978c) used a coordinate transformation in the form of a piecewise exponential stretch to obtain a smoothly varying grid system. The grid spacing varied from 150 ft to 900 ft for the Coos Bay Inlets model and from 604 m to 2,583 m for the Galveston Bay model. Pinder and Gray (1977) pointed out that the finite difference approximation is correct to the first order for a model with varied grid spacing and to the second order for an equally spaced grid.

2.1.3 Moving boundary models

Several numerical models for simulating moving boundaries have been proposed in the past. Most of them have used the FD technique and FE technique with fixed or deforming grids. The fixed grid technique usually treats the moving boundary by turning cells on or off at the boundary. The cells are considered flooded, and turned on, if the neighboring water depths are positive, and dry, and turned off,
if the neighboring water depths are negative. The cells are either fully flooded or fully dry. If large errors are to be avoided, a very fine mesh spacing must be used near the boundary. Flather and Heaps (1975) used the above technique to model tidal inundations in Morecombe Bay. The impulsive filling of grid cells results in the violation of the continuity equation and may result in possible instabilities.

Reid and Bodine (1968) used a weir type formulation near the closed boundary to model transient storm surges in Galveston Bay, Texas. They omitted both the nonlinear advection terms and coriolis terms in the governing equations and developed a FD model based on the linearized equations of motion. The bathymetry was represented in a stair-step fashion and a grid cell was allowed to flood if the neighboring grid cells were flooded. The actual mechanism of the flooding of a grid cell was achieved by using empirical weir formulas. A number of empirical coefficients are used which were determined by comparing model results with field data. A disadvantage of such a method is that continuous boundary movements cannot be modeled.

Yeh and Chou (1979) developed a nonlinear explicit FD model to simulate storm surges in a fixed grid system. The boundary between land and water was simulated as a discrete moving boundary. The boundary moved in discrete jumps, advancing or retreating according to the rise or recession of the surge level. During the rising surge, a grid point is added if the surge elevation of neighboring points is above the base elevation. During the recession of the surge grid cells are removed if the total water depth falls below a preset level. They compared the model with field data and also with a similar numerical model which used a fixed vertical wall at the closed boundary. The fixed boundary over predicted the surge levels by as much as 30 percent.
Hirt and Nichols (1981) proposed a particle in cell method FD scheme to simulate moving boundaries. They called this method the volume of fluid (VOF) technique. According to this technique, a step function is used to denote the volume of fluid in a given cell. The value of the step function is zero if the cell is dry and one if the cell is flooded. Cells with a value of the step function between zero and one must now contain the free boundary. The location of the free boundary is then determined using the value of the step function and the normal to the boundary. The resulting equations are solved using an iterative technique. The model was applied to the dam break problem, the undular bore problem and the breaking bore problem. They reported good agreement with experimental results.

Benque et al., (1982) developed a FD numerical model with the inclusion of the tidal flat using the method of fractional steps. They split the vertically integrated shallow water equations into three steps, which are advection, diffusion and propagation. A different numerical scheme was used for each step. The dry land was assumed to be covered with a thin layer of water and the flow was assumed to be governed by bottom friction. The actual movement of the boundary was assumed to take place during the propagation step, which was represented by a resistance equation. They reported that the treatment of the moving boundary in this manner violated the continuity equation slightly. They applied the model to the River Canche estuary system in France and reported good agreement with field data.

Lynch and Gray (1980) outlined a general technique whereby a moving boundary can be treated by a FE eulerian method. They chose the FE basis functions as functions of time so that FE boundaries track the moving boundary. They allowed only the boundary elements to move and kept the rest of the computational domain fixed. This then induces elements with larger area near the boundary thus
increasing the truncation error of the numerical scheme. They also pointed out that deforming elements might result in highly skewed meshes which might cause stability problems. They applied the model to simulate run up of waves on beaches. They treated only rectangular geometries and did not report any simulations on arbitrarily shaped physical regions.

Gopalakrishnan and Tung (1983) described a one dimensional FE long wave run up model. The moving shoreline was handled by allowing the shoreline element to deform so that the beach node always tracked the shoreline. A mechanism was provided by which if the element became too stretched it split into two elements. The element containing the shoreline node was allowed to deform but the other new element created remained fixed. They showed graphically the details of the run up process but did not show the run down process. It should be noted that this technique cannot be easily extended to two dimensions.

2.2 Summary

All the numerical models developed so far have been used mostly to model medium to large scale water bodies. The modelling of wetland hydrodynamics has largely been ignored, because of the complicated nature of the wetlands terrain. The unique characteristics of wetlands require mathematical models capable of treating some very specific flow conditions generally not encountered in modelling of large water bodies. These are:

1. Ability to handle very shallow flow regimes.

2. Accurate treatment of moving boundaries is essential to study the periodic inundation of vegetation surrounding small ponds.
3. Ability to use large time steps so that computational times can be minimized.

In general existing finite element and finite difference models do not possess all the above mentioned capabilities.
Chapter 3

Methodology

3.1 Introduction

A literature survey in the preceding chapter indicates the following deficiencies in the finite element and finite difference methods: 1) The finite element method, although capable of handling irregular boundaries, is very complicated to formulate and the theoretical aspects regarding stability, convergence, etc., for transient problems have not been investigated fully. 2) The traditional finite difference method, which uses a rectangular grid, is not capable of handling irregular shorelines resulting in possible errors in the simulation. The method also handles moving boundaries due to coastal flooding rather poorly. 3) A nested finite difference must be used in areas where large flow gradients are encountered. This requires an intimate knowledge of the flow structure in the region being modeled. Nesting may also cause wave reflection at the coarse-fine grid interface. A nested model also requires additional computer overhead in terms of processor time and memory.

The above disadvantages of the traditional methods have prompted an investigation of alternative methods free from these problems. A relatively new method which is gaining popularity in the area of Computational Aerodynamics is the Boundary Fitted Coordinate (BFC) method developed by Thompson (1977). In this method, an irregular region is mapped on to a rectangular computational domain. This is accomplished by solving two elliptic partial differential equations.
with the cartesian coordinates representing the irregular boundary, as Dirichlet boundary conditions. The governing flow equations comprising the continuity and the two momentum equations, and the appropriate boundary conditions, are also transformed into the rectangular computational domain. The boundary conditions are now prescribed on rectangular boundaries of the computational domain. The transformed flow equations in the computational domain are solved using finite difference techniques.

Using this method it is possible to incorporate moving boundaries, by keeping track of the leading edge of the wave at the boundary and moving the boundary such that the boundary always conforms to the leading edge of the wave. It is also possible to incorporate islands and other obstacles in the flow region.

Another feature of the BFC method is the capability of incorporating adaptive gridding. In adaptive gridding, it is possible to concentrate grid points in regions of high flow gradient. The adaptive grid generator is coupled with the flow equations, and moves grid points where flow gradients exist. This feature makes the use of nested grids unnecessary, and thus saves computer time. The BFC method and the hydrodynamic model are described in detail in the following sections.

3.2 Boundary fitted coordinate method

The boundary fitted coordinate (BFC) method consists of generating computational grids in arbitrary domains. The methods employed to achieve this are either analytical, like conformal mapping, or more commonly, numerical. Analytical methods such as conformal mapping work only when the physical regions can be described by regular euclidean shapes. In general, it is very difficult to obtain
Figure 3.1: (a) An irregular physical domain (b) The corresponding transformation into a computational domain
conformal modules for arbitrary regions. Thus, one is forced to use some numerical grid generation technique to generate grids in arbitrary regions. Although there are a number of ways to generate grids, we restrict our discussion to the methods used in this report.

In general a transformation from a physical space \((x,y)\) to a computational space \((\xi,\eta)\) can be expressed by the vector-valued function:

\[
\begin{pmatrix}
\xi \\
\eta
\end{pmatrix} = \begin{pmatrix}
\xi(x,y) \\
\eta(x,y)
\end{pmatrix}
\]  

(3.1)

The Jacobian of the transformation is

\[
J_1 = \begin{pmatrix}
\xi_x & \xi_y \\
\eta_x & \eta_y
\end{pmatrix}
\]  

(3.2)

The inverse transformation of equation (3.1) can be written as

\[
\begin{pmatrix}
x \\
y
\end{pmatrix} = \begin{pmatrix}
x(\xi,\eta) \\
y(\xi,\eta)
\end{pmatrix}
\]  

(3.3)

The Jacobian matrix for the above inverse transformation is

\[
J_1 = \begin{pmatrix}
x_\xi & x_\eta \\
y_\xi & y_\eta
\end{pmatrix}
\]  

(3.4)

The determinant of the above Jacobian is

\[
J_1 = x_\xi y_\eta - x_\eta y_\xi
\]  

(3.5)

The matrices of the Jacobians in equations (3.2) and (3.4) are related by

\[
(J_1) = (I_1)^{-1}
\]  

(3.6)

which gives the relationship

\[
\xi_x = \frac{y_\eta}{J}
\]  

(3.7)
\[
\xi_v = -\frac{x_\eta}{J} \\
\eta_z = -\frac{y_\xi}{J} \\
\eta_y = \frac{x_\xi}{J}
\]

The partial derivatives for a sufficiently differentiable function \( g \) are then obtained by use of the chain rule of differentiation:

\[
g_x = \frac{(y_\eta g_\xi - y_\xi g_\eta)}{J} \\
g_y = \frac{(x_\xi g_\eta - x_\eta g_\xi)}{J}
\]  

(3.8)

The transformation from an arbitrary physical space to a rectangular computational space is one to one if the Jacobian of the transformation is non zero. The proof is described by Mastin and Thompson (1978) and hence will not be repeated here.

Numerical grid generation methods can be broadly classified into two categories: 1) Algebraic grid generators and 2) Partial differential equations based grid generators.

### 3.2.1 Algebraic grid generators

Algebraic grid generation is basically an interpolation between boundaries or between intermediate surfaces. The interior grid points are generated by interpolating from known distribution of coordinate points on the boundaries. The interpolation functions are generally polynomials of varying degree or splines.

The main advantages of algebraic grid generators is their inherent speed in generating grids. Algebraic procedures also allow explicit control of grid point distribution. The main disadvantages of algebraic methods is their inability to generate
smooth grids. Smooth grids are essential as the truncation error of finite difference expressions decrease with increase in grid smoothness. Another disadvantage of algebraic methods is that they do not guarantee a one to one mapping for arbitrary regions. Algebraic methods have been known to fail when applied to concave physical regions. The most popular algebraic grid generator is the transfinite interpolation. Transfinite interpolation involves interpolation among functions defined along curves or surfaces, rather than among point values, and thus matches the function at a nondenumerable number of points. The nondenumerable aspect of transfinite interpolation comes from the possible infinity of points defining general boundaries as compared to a tensor product structure (i.e., product of projectors) that uses only corner information and, is therefore finite. In actual applications, however, these functions can be defined by piecewise linear functions. In higher dimensions, this interpolation can be expressed as a boolean sum of univariate projections. These functions can specify the values of the variables on the curve surface. It is also possible to specify derivatives on the curve surfaces, which can then be used to control the orthogonality of the grid lines near the curve surfaces.

The transfinite interpolation is used in this study to generate preliminary grids which act as initial conditions for the elliptic grid generator to be described in a later section. The transfinite method is described below. For other ways of generating algebraic grids the reader is referred to Thompson, et al. (1985).

3.2.1.1 Transfinite interpolation

The interior grid coordinates in two dimensions are obtained by linear interpolation of boundary coordinates. The interior grid points \( x_{i,j} \) and \( y_{i,j} \) are given
\[ 2x_{i,j} = \frac{JMAX - j}{JMAX - 1} x_{i,1} + \frac{j - 1}{JMAX - 1} x_{i,JMAX} + \]
\[ \frac{IMAX - i}{IMAX - 1} x_{1,j} + \frac{i - 1}{IMAX - 1} x_{1MAX,j} \]
\[ 2y_{i,j} = \frac{JMAX - j}{JMAX - 1} y_{i,1} + \frac{j - 1}{JMAX - 1} y_{i,JMAX} + \]
\[ \frac{IMAX - i}{IMAX - 1} y_{1,j} + \frac{i - 1}{IMAX - 1} y_{1MAX,j} \]

(3.9)

Fig. (3.1) shows a typical irregular region and the corresponding computational region.

3.2.2 Partial differential equation based grid generators

The algebraic grid generators described in the previous section, although computationally fast, have certain drawbacks. These include, a lack of guarantee of a one to one mapping of the physical domain to the computational domain (negative jacobians are possible), and a lack of smoothness. In order to overcome the drawbacks of algebraic grids, grid generators using partial differential equations (pde) were developed. The idea behind such methods is the use of pde's as grid generation equations with the grid coordinates prescribed at the boundaries acting as boundary conditions. The differential equations are generally nonlinear and hence an iterative method must be used to solve them. Partial differential equation grid generators based on hyperbolic equations (Nicolet et al. (1982)), parabolic equations (Noack and Anderson(1990)) and elliptic equations (Thomson et al. (1985)) have been developed. Of all the pde based grid generators, the elliptic generator of Thompson et al. (1985) is the most popular. This is because elliptic generators allow one to prescribe Dirichlet conditions all along the boundaries of the physical domain. Due to the strong ellipticity of these equations, a one to one mapping is
guaranteed. In this study the elliptic grid generator is used to generate grids and a brief description of this procedure is given below.

3.2.2.1 Elliptic grid generator

The basic idea behind this method is the transformation of an irregular domain in physical space to a rectangular domain in computational space. This is accomplished by solving two nonlinear elliptic partial differential equations (pde's) with the Dirichlet boundary condition. The coordinate system so generated has one coordinate specified as a constant on the boundaries and the distribution of the other specified along the boundaries. Regardless of the shape of the physical space all computations are performed on a rectangular computational space with boundary conditions specified along the boundaries. A brief discussion of Thompson's method is presented here for the sake of completeness; for more details, the reader is referred to Thompson et al. (1985). The elliptic generating system adopted by Thompson consists of two Poisson's equations given by:

\[ \xi_{xx} + \xi_{yy} = P(x, y) \]  \hspace{1cm} (3.10)

\[ \eta_{xx} + \eta_{yy} = Q(x, y) \]  \hspace{1cm} (3.11)

with the boundary conditions as shown in Fig. (3.1),

\[ \eta = \eta(x, y), \xi = \text{constant on } \Gamma_1 \]  \hspace{1cm} (3.12)

\[ \eta = \eta(x, y), \xi = \text{constant on } \Gamma_3 \]

\[ \xi = \xi(x, y), \eta = \text{constant on } \Gamma_2 \]

\[ \xi = \xi(x, y), \eta = \text{constant on } \Gamma_4 \]
where $\Gamma_1, \Gamma_2, \Gamma_3$, and $\Gamma_4$ are arcs which describe the outer boundary of the physical domain. Since all computations are to be performed on a rectangular transformed plane, it is necessary to interchange the dependent and independent variables. Thus, equations (3.10) and (3.11) are given as

\begin{align*}
\alpha_1 x_{\xi \xi} - 2\beta_1 x_{\xi \eta} - \gamma_1 x_{\eta \eta} + J^2 (P x_{\xi} + Q x_{\eta}) &= 0 \quad (3.13) \\
\alpha_1 y_{\xi \xi} - 2\beta_1 y_{\xi \eta} - \gamma_1 y_{\eta \eta} + J^2 (P y_{\xi} + Q y_{\eta}) &= 0 \quad (3.14)
\end{align*}

where

\begin{align*}
J &= \text{Jacobian of the transformation} = x_{\xi \eta} y_{\eta \xi} \\
\alpha_1 &= x_{\xi}^2 + y_{\eta}^2, \beta_1 = x_{\xi} x_{\eta} + y_{\xi} y_{\eta}, \gamma_1 = x_{\xi}^2 + y_{\eta}^2 \quad (3.15) \\
\alpha_1 &= x_{\eta}^2 + y_{\eta}^2, \beta_1 = x_{\xi} x_{\eta} + y_{\xi} y_{\eta}, \gamma_1 = x_{\xi}^2 + y_{\eta}^2 \quad (3.16)
\end{align*}

The functions $P$ and $Q$ are known as control functions and are used as coordinate controls in order to attract, repel or move coordinate lines to another coordinate line or a coordinate point. The following functional forms for $P$ and $Q$ as given by Thompson et al. (1985) were used.

\begin{align*}
P(\xi, \eta) &= -\sum_{i=1}^{N} a_i \text{sign}(1, (\xi - \xi_i)) \exp(-c_i |\xi - \xi_i|) \\
&\quad - \sum_{i=1}^{N} b_i \text{sign}(1, (\xi - \xi_i)) \exp(-d_i \sqrt{(\xi - \xi_i)^2 + (\eta - \eta_i)^2}) \quad (3.17) \\
Q(\xi, \eta) &= -\sum_{i=1}^{N} a_i \text{sign}(1, (\eta - \eta_i)) \exp(-c_i |\eta - \eta_i|) \\
&\quad - \sum_{i=1}^{N} b_i \text{sign}(1, (\eta - \eta_i)) \exp(-d_i \sqrt{(\xi - \xi_i)^2 + (\eta - \eta_i)^2}) \quad (3.18)
\end{align*}

Where, the function $\text{sign}(1, (\xi - \xi_i))$ returns the value 1 if the value of $(\xi - \xi_i)$ is greater than or equal to zero and -1 if $(\xi - \xi_i)$ is less than zero; $a_i$ is the amplitude
factor which controls the attraction of $\xi$ or $\eta$ line towards the corresponding $\xi_i$ or $\eta_i$ line; $b_i$ is the amplitude factor which controls the attraction of a $\xi$ or $\eta$ line towards a point $(\xi_i, \eta_i)$; $c_i$ is the decay factor which controls the decrease in attraction of a $\xi$ or $\eta$ line towards the corresponding $\xi_i$ or $\eta_i$ line; $d_i$ is the decay factor which controls the decrease in attraction of a $\xi$ or a $\eta$ line towards a point $(\xi_i, \eta_i)$.

Although the new system of equations is more complex, the boundary conditions are specified on straight boundaries and the coordinate spacings are uniform in the transformed plane. These advantages outweigh any disadvantages arising from the complexity of the equations to be solved. The generating system of equations (3.13) and (3.14) and the appropriate boundary conditions were discretized using central difference approximations and the resulting equations were solved using a successive over-relaxation method. All computations are made in a transformed rectangular computational domain. For details of finite difference representation of these grid generation equations the reader is referred to Thompson et al. (1985). The partial derivatives of the governing flow equations are appropriately transformed to the computational domain using transformation relations given by Thompson et al. (1977). The boundary-fitted coordinate method was applied to represent three example geometries: (1) a converging geometry, (2) a circular region, and (3) a natural watershed located near Hastings, Nebraska. Fig. (3.2a) shows the grid in a converging section, generated by the grid-generating system, and Fig. (3.2b) shows the transformed rectangular computational grid. In order to test the validity of the method, the inverse transformation of the transformed computational grid was taken which exactly reproduced Fig. (3.2a). Following the same method, grids were generated in the circular region as shown in Fig. (3.3a) and the actual watershed, as shown in Fig. (3.4a). The corresponding transformed rectangular
Figure 3.2: (a) Grid generated in the physical domain for converging section (b) The transformed rectangular computational domain
Figure 3.3: (a) Grid generated in a circular region connected to a rectangle (b) The transformed rectangular computational domain
Figure 3.4: (a) Grid generated in the physical domain for a natural watershed located in Hastings, Nebraska (b) The transformed rectangular computational domain
grids are displayed in Fig. (3.3b) and (3.4b). Again, the method was able to exactly reproduce the original geometries.

3.2.3 Adaptive grid generator

Adaptive grid generators can be either solution adaptive or boundary adaptive. In solution adaptive grid generators, the grid points are moved to locations where there are large gradients of the solution. The philosophy behind such an approach is the fact that truncation errors of finite difference representations decrease with a decrease in grid spacing. Thus, if there are large gradients in the solution in some regions of a grid, a smaller grid spacing in such regions would more accurately represent the solution gradient. This enables the use of fewer grid points resulting in faster simulation times without compromising accuracy. In boundary adaptive grid generators, the grid is regenerated at every time step as the boundary moves. A combination of the above is also possible where the grid is boundary adaptive as well as solution adaptive. Most of the popular adaptive grid schemes are based on some form of arc equidistribution. The concept of arc equidistribution is most easily explained by considering a one-dimensional example. Let \((x,t)\) represent the physical coordinates and define coordinates in the computational domain to be \((\xi, \tau)\). Furthermore, let the mesh spacing in the computational domain be uniform with grid points unit distance apart. Let \(W(x,t)\) be a weight function and consider the grid law

\[ x_\xi W = \text{constant} \quad (3.19) \]
with unit spacing in the computational domain, this equidistribution law may be represented by the finite difference equation

$$\Delta x W = \text{constant} \quad (3.20)$$

If \( W \) is large, the mesh spacing in physical space is small and vice versa. The weight function is normally written as some function of the dependent variables and this provides the relationship between grid spacing and the calculated solution. Grid construction is normally accomplished by solving a second-order differential equation. In this setting, equation (3.19) is differentiated with respect to \( \xi \) to obtain

$$x_{\xi\xi} + x_{\xi} \frac{W_{\xi}}{W} = 0 \quad (3.21)$$

Solution of this equation yields mesh point positions in space. A disadvantage of the above method is that in two dimensions the grid generated is highly skewed and this induces errors in the finite difference solution. In order to overcome the above problem, Anderson (1986) converted the elliptic grid generator given by equations (3.13) and (3.14) to an adaptive grid generator. The grid control functions \( P(\xi,\eta) \), and \( Q(\xi,\eta) \) in equations (3.13) and (3.14) can be viewed as equidistribution functions. The form for \( P(\xi,\eta) \), and \( Q(\xi,\eta) \) is as given by Thomas and Middlecof (1979) and are shown below.

$$P(\xi,\eta) = \frac{\alpha}{J^2} \phi \quad (3.22)$$

$$Q(\xi,\eta) = \frac{\gamma}{J^2} \psi \quad (3.23)$$

The relationship between \( \phi, \psi \) and the weight function is given by

$$\phi = \frac{1}{W_1} \frac{\partial W_1}{\partial \xi}$$

$$\psi = \frac{1}{W_2} \frac{\partial W_2}{\partial \eta} \quad (3.24)$$
With the above representation of $\phi$ and $\psi$, the elliptic grid generator given by equations (3.13) and (3.14) becomes an adaptive grid scheme. Some examples demonstrating the capability of the above adaptive grid generating scheme are given below. The first example is the diagonal problem. In this problem, a function $u$ is set equal to 1 in the upper diagonal region, and equal to zero in the lower diagonal region. This creates a gradient along the diagonal, and the adaptive grid generator recognizes this, and clusters grid points along the gradient. The form of the weight functions $W_1$ and $W_2$ is shown below:

$$W_1 = 1 + A|\frac{\partial u}{\partial \xi}|^2$$  \hspace{1cm} (3.25)

$$W_2 = 1 + A|\frac{\partial u}{\partial \eta}|^2$$ \hspace{1cm} (3.26)

The converged grid in the physical plane is shown in Fig. (3.5). The computational grid is just a square and is not shown here. The second example is the pill box problem. In this problem a function $u$ on a unit square is set equal to 0 for radius values less than 0.15 and greater than 0.35, and was allowed to vary linearly from 0 to 1 in between. The weight functions shown in equations (3.25) and (3.26) were used for this example. The converged grid is shown in Fig. (3.6). The above examples show the capabilities of the adaptive grid generator to sense gradients in the solution, and cluster grid lines in such locations.

### 3.2.3.1 Generation of patched grids

For very complicated domains, a transformation to a single rectangular domain is impossible. To overcome this problem, the computational domain is split into different zones, each of which is transformed to a logical rectangle. The coordinates at the zonal interfaces are smoothed in order to avoid discontinuity in the grid
Figure 3.5: Adaptive grid for the diagonal problem using the elliptic grid generator, 41 x 41 mesh, A=3.0, B=3.0

Figure 3.6: Adaptive grid for a pill box problem using the elliptic grid generator, 41 x 41 mesh, A=10.0, B=10.0
Figure 3.7: (a) Splitting of the physical domain into different zones for Lake Pontchartrain (b) The transformed rectangular computational domain
lines. An example domain for Lake Pontchartrain is shown in Fig. (3.7). The Lake system is split into four subdomains, and each of these subdomains is mapped onto a logical rectangle in the computational domain.

3.3 Hydrodynamic model

The grid generator described in the previous section is purely an abstract mathematical technique to generate meshes in arbitrary regions. If a pure eulerian description of fluid motion is used, the grid generator and the hydrodynamic model are essentially decoupled. The grid is generated once at the beginning of the simulation and is not altered during the remainder of the simulation. In the present model, however, an eulerian-lagrangian concept is used to describe the fluid motion. In an eulerian-lagrangian description of fluid motion, the grid moves in time due to either local changes in field variables (solution adaptive), or due to movement of the physical boundary. In either case, a new grid must be generated at every time step. The coupling between the grid generator and the hydrodynamic model is explicit in the sense that the grid parameters in terms of scale factors used in the hydrodynamic model are taken at a previous time level. This is done in order to obtain a linear system of equations for which efficient solution methods exist. An implicit coupling, on the other hand, would entail the solution of the grid generator along with the hydrodynamic equations resulting in the solution of systems of nonlinear equations, which are very difficult and time consuming to solve.

The following sections describe the hydrodynamic model and its interaction with the grid generator. An efficient coupling between the model and the grid
generator results in a versatile model capable of handling large distortions of the boundary.

3.3.1 Hydrodynamic equation in Cartesian Coordinates

The mathematical equations describing flow in shallow water can be obtained by vertically integrating the three dimensional Navier Stokes equation of fluid motion. Generally, it is assumed that the density of water over the depth is constant and the vertical pressure variation is hydrostatic, thus leading to the following continuity and momentum equations in a right-handed cartesian coordinate system.

Continuity

\[
(z)_t + (uH)_x + (vH)_y = 0 \quad (3.27)
\]

X-momentum

\[
(u)_t + u(u)_x + v(u)_y + g(z)_x - f v + \frac{\tau_{bx} - \tau_{sx}}{\rho H} = 0 \quad (3.28)
\]

Y-momentum

\[
(v)_t + u(v)_x + v(v)_y + g(z)_y + f u + \frac{\tau_{by} - \tau_{sy}}{\rho H} = 0 \quad (3.29)
\]

Where \( t \) is time, \( x \) and \( y \) are the spatial coordinates, \( z(x, y, t) \) is the free surface elevation, \( u \) and \( v \) are the depth averaged velocities in the \( x \) and \( y \) direction, \( h \) is the water depth. \( H = h + z \) is the total water depth, \( g \) is the acceleration due to gravity, \( \rho \) is the density of water, \( f \) is the coriolis parameter, \( \tau_{sx} \) and \( \tau_{sy} \) are the wind shear stresses in the \( x \) and \( y \) direction respectively, and \( \tau_{bx} \) and \( \tau_{by} \) are the bottom shear stresses in the \( x \) and \( y \) direction. \( \tau_{bx} \) and \( \tau_{by} \) can be expressed as

\[
\tau_{bx} = \frac{\rho g}{C_s^2} u \sqrt{u^2 + v^2} \quad (3.30)
\]
and

\[ \tau_{by} = \frac{pg}{C^2} v \sqrt{u^2 + v^2} \]  \hspace{1cm} (3.31)

where \(C\) is the Chezy coefficient, which is related to the Manning’s coefficient \(n\) as

\[ C = \frac{H^\frac{1}{k}}{n} \]  \hspace{1cm} (3.32)

### 3.3.2 Transformation of governing equations to curvilinear Coordinates

The continuity and momentum equations in Cartesian Coordinates are given by equations (3.27),(3.28) and (3.29). They are now transformed to the curvilinear coordinate system by the following transformation.

Consider a continuous function \(g(x, y)\). The partial derivatives of \(g(x,y)\) \(g_x\) and \(g_y\) are given by the chain rule of differentiation.

\[ g_x = g_{\xi}\xi_x + g_{\eta}\eta_x \hspace{1cm} g_y = g_{\xi}\xi_y + g_{\eta}\eta_y \]  \hspace{1cm} (3.33)

where \(x = x(\xi, \eta), y = y(\xi, \eta)\)

Using the above relationships, equations (3.27)–(3.29) can be transformed to

**Continuity**

\[ z_t + \xi_x(uH)\xi + \eta_x(uH)\eta + \xi_y(vH)\xi + \eta_y(vH)\eta = 0 \]  \hspace{1cm} (3.34)

**\(\xi\)-momentum**

\[ u_t + U u_{\xi} + V u_{\eta} + g\xi_x z_{\xi} + g\eta_x z_{\eta} = f v + T^\xi \]  \hspace{1cm} (3.35)

**\(\eta\)-momentum**

\[ v_t + U v_{\xi} + V v_{\eta} + g\xi_y z_{\xi} + g\eta_y z_{\eta} = -f u + T^\eta \]  \hspace{1cm} (3.36)

where \(U\) and \(V\) are the contravariant velocity terms given by

\[ U = u\xi_x + v\xi_y \hspace{1cm} V = u\eta_x + v\eta_y \]  \hspace{1cm} (3.37)
$T^\xi$ and $T^\eta$ are the stress (surface and bottom) in the $\xi$ and $\eta$ direction respectively. It should be noted that the above equations are in terms of both covariant and contravariant velocity. The covariant velocity vector is tangential to a coordinate line, and the contravariant velocity is normal to a curvilinear coordinate surface. In an orthogonal coordinate system both the covariant and contravariant velocities are equal.

3.3.3 Characteristic analysis of the governing equations in curvilinear coordinate system.

Equations (3.34)–(3.36) form a quasilinear system of hyperbolic differential equations. In order to determine a suitable semi-implicit discretization, where stability is independent of the gravity wave stability criterion, the characteristic cone of the governing equation is analyzed. For this purpose, equations (3.34) to (3.36) are written as follows

\begin{align*}
u_t + Uu_t + Vv_t + g\xi_zz + g\eta_zn &= f v + T^\xi \\
v_t + Uv_t + Vv_t + g\xi_zz + g\eta_zn &= -fu + T^\eta \\
z_t + Uz_t + Vz_t + H(u\xi_zz + \eta\eta_zz + v\xi_zz + \eta\eta_z) &= -U\xi - V\eta
\end{align*}

or, in matrix notation,

\begin{equation}
\dot{w} + A(w)w + B(w)w = D(w)
\end{equation}

where $w = (U, V, Z)^T$ and

$$A = \begin{pmatrix}
U & 0 & g\xi_x \\
0 & U & g\xi_y \\
H\xi_x & H\xi_y & U
\end{pmatrix}$$
\[ B = \begin{pmatrix} V & 0 & g\eta_x \\ 0 & V & g\eta_y \\ H\eta_x & H\eta_y & V \end{pmatrix} \]

\[ D = \begin{pmatrix} f\nu + T^x \\ -fu + T^y \\ -Uh_\xi - Vh_\eta \end{pmatrix} \]

If \( I \) is the identity matrix, the characteristic equation for equation (3.41) is given by

\[ \det(qI + rA + sB) = 0 \] (3.42)

that is

\[ (q + rU + sV)[(q + rU + sV)^2 - gH[(r\xi_y + s\eta_y)^2 + (r\xi_x + s\eta_x)^2]] = 0 \] (3.43)

The triples \((q, r, s)\) satisfying equation (3.43) are the directions normal to the characteristic cone at its vector (Forsyth and Wasow (1960)). Equation (3.43) decomposes into two equations

\[ q + rU + sV = 0 \] (3.44)

\[ (q + rU + sV)^2 - gH[(r\xi_y + s\eta_y)^2 + (r\xi_x + s\eta_x)^2] = 0 \] (3.45)

Hence, as shown in Fig. (3.8), the local characteristic cone with vertex in \((\xi_0, \eta_0, t_0)\) consists of the line through \((\xi_0, \eta_0, t_0)\) parallel to the vector \((U, V, 1)\), and the cone whose equation is given by

\[ [(\xi - \xi_0) - U(t - t_0)]^2 + [(\eta - \eta_0) - V(t - t_0)]^2 - gH(t - t_0)^2 = 0 \] (3.46)

On the cone surface, the gradient of the left hand side of equation (3.46) satisfies equation (3.45). For more details on the derivation of the characteristic cone,
see Abbot(1979). It can be seen from equation (3.45) that the first half of the characteristic cone depends only on the contravariant velocity $U$ and $V$; the second half which is defined by equation (3.43), depends also on the celerity of the wave $\sqrt{gH}$. It should also be noted that the term $gH$ in equation (3.43) arises from the off-diagonal term $g\xi_x$, $g\xi_y$, $H\xi_x$ and $H\xi_y$ from the matrix $A$ and $g\eta_x$, $g\eta_y$, $H\eta_x$ and $H\eta_y$ from the matrix $B$. They are the coefficients of $z\xi$ and $z\eta$ in equations (3.39) and (3.40) and $u\xi$, $u\eta$, $v\xi$, $v\eta$ in equation (3.40). If the numerical method is to be independent of the gravity wave stability criterion, these terms should be discretized in an implicit manner.
3.3.4 Transformation of hydrodynamic equations to a moving curvilinear coordinate system

The governing equations of continuity and momentum in a cartesian coordinate system (3.27), (3.28) and (3.29) are transformed to a general curvilinear moving coordinate system. The details of these transformations can be found in Thompson et al. (1985).

Continuity

\[ z_t + \xi_t z_\xi + \eta_t z_\eta + \frac{1}{J} (U J H)_\xi + \frac{1}{J} (V J H)_\eta = 0 \]  (3.47)

\( \xi \)-momentum

\[ (U J)_t + y_\eta [(U + \xi_t)(U x_\xi + V x_\eta)_\xi + (V + \eta_t)(U x_\xi + V x_\eta)_\eta] \]
\[-x_\eta [(U + \xi_t)(U y_\xi + V y_\eta)_\xi + (V + \eta_t)(U y_\xi + V y_\eta)_\eta] + \frac{g \alpha}{J} z_\xi \]
\[-\frac{g \beta}{J} z_\eta - f(\beta U + \alpha V) - (T_{\text{w}}x_\xi - T_{\text{w}}y_\eta) + (y_t)_\eta (U x_\xi + V x_\eta) \]
\[-(x_t)_\eta (U y_\xi + V y_\eta) + \frac{gn^2 \sqrt{(\gamma u^2 + 2\beta UV + \alpha V^2)(J U)}}{H^{4/3}} = 0 \]  (3.48)

\( \eta \)-momentum

\[ (V J)_t + x_\xi [(U + \xi_t)(V y_\xi + V y_\eta)_\xi + (V + \eta_t)(U y_\xi + V y_\eta)_\eta] \]
\[-y_\xi [(U + \xi_t)(U x_\xi + V x_\eta)_\xi + (V + \eta_t)(U x_\xi + V x_\eta)_\eta] + \frac{g \beta}{J} z_\xi \]
\[-\frac{g \gamma}{J} z_\eta - f(\gamma U + \beta V) - (T_{\text{w}}x_\xi - T_{\text{w}}y_\eta) + (y_t)_\xi (U x_\xi + V x_\eta) \]
\[-(x_t)_\xi (U x_\xi + V x_\eta) + \frac{gn^2 \sqrt{(\gamma u^2 + 2\beta UV + \alpha V^2)(J U)}}{H^{4/3}} = 0 \]  (3.49)

where

\[ \xi_t = \frac{y_t x_\xi - x_t y_\xi}{J} \quad \eta_t = \frac{x_t y_\xi - y_t x_\xi}{J} \]  (3.50)
$x_t, y_t =$ Grid speed in the physical domain

$U =$ Contravariant velocity component normal to $\eta =$ constant line = $\frac{u_x - v_x}{J}$

$V =$ Contravariant velocity component normal to $\xi =$ constant line = $\frac{u_x - v_y}{J}$

$\alpha = x^2 + y^2$, $\beta = x_\xi x_\eta + y_\xi y_\eta$, $\gamma = x^2 + y^2$

$x_\xi, x_\eta, y_\xi, \text{ and } y_\eta$ are scale factors.

It should be noted that the above equations constitute the shallow water equation in a moving coordinate system. Although the grid moves in the physical $(x, y)$ domain, the movement in the computational $(\xi, \eta)$ domain is expressed through the scale factor $(\xi_t, \eta_t)$. If the grid speed is equal to zero then the motion is pure eulerian, else, if the grid speed is equal to the fluid velocity then the motion is pure lagrangian. In general, in order to avoid excessive grid movement, the grid speed is less than the fluid velocity resulting in an eulerian-lagrangian formulation. In the present study the eulerian-lagrangian formulation is used. The details of obtaining the grid speeds and scale factors will be discussed in subsequent sections.

### 3.3.5 Boundary condition formulation

The formulation of boundary conditions in curvilinear co-ordinate systems (for $\xi = constant$ boundary) for the hyperbolic case will be considered in this section. For this purpose, the shallow water equations written in the form of equation (3.41) will be used. In order to analyze the boundary problem, the matrix $A$ must be diagonalized. In order to accomplish this, the eigenvector matrix of $A$ is used.

The eigenvalues of the matrix $A$ are,

$$
\begin{bmatrix}
U, U - \sqrt{gH \left( \xi_x^2 + \xi_y^2 \right)}, U + \sqrt{gH \left( \xi_x^2 + \xi_y^2 \right)}
\end{bmatrix}
$$

(3.51)
The corresponding eigenvector matrix of $A$ is

$$S = \begin{pmatrix}
1 & 1 & 1 \\
-\xi_x & \xi_y & 0 \\
0 & \sqrt{\frac{H}{g}(1 + \frac{\xi_x}{\xi_y})} & -\sqrt{\frac{H}{g}(1 + \frac{\xi_y}{\xi_x})}
\end{pmatrix} \quad (3.52)$$

The matrix $A$ is then diagonalized by the transformation $\Lambda = (S^{-1}AS)$ giving

$$\Lambda = \begin{pmatrix}
U & 0 & 0 \\
0 & U + \sqrt{gH(\xi_x^2 + \xi_y^2)} & 0 \\
0 & 0 & U - \sqrt{gH(\xi_x^2 + \xi_y^2)}
\end{pmatrix} \quad (3.53)$$

Multiplying equation (3.41) by $S$ we obtain

$$\omega_t + \Lambda \omega_t = F, \quad \omega = S^{-1}u = (\omega_0, \omega_1^+, \omega_2^-)$$

where

$$\omega_0 = \frac{\xi_y(u\xi_y - v\xi_x)}{\left(\xi_x^2 + \xi_y^2\right)} \quad (3.54)$$

$$\omega_1^+ = \frac{\xi_x}{2} \left[ \frac{U}{\left(\xi_x^2 + \xi_y^2\right)} + \frac{z}{\left(\xi_x^2 + \xi_y^2\right)} \sqrt{\frac{g}{H}} \right]$$

$$\omega_2^- = \frac{\xi_x}{2} \left[ \frac{U}{\left(\xi_x^2 + \xi_y^2\right)} - \frac{z}{\left(\xi_x^2 + \xi_y^2\right)} \sqrt{\frac{g}{H}} \right]$$

where the right hand side $F$ does not influence the characteristic matrix. Then the transformation to the curvilinear co-ordinate system changes the tangential velocity component of the covariant one and normal component of the contravariant one. For $0 < U < \sqrt{gH(\xi_x^2 + \xi_y^2)}$, the matrix $A$ has two positive eigenvalues and hence two boundary conditions are needed at an inflow boundary. For $U < 0$ the matrix $A$ has only one positive eigenvalue and hence only one boundary condition should be supplied at an outflow boundary. Thus based upon the above analysis, the following
form of boundary conditions should be used.

\[ V = v_1, U + C_p z = f_1 \quad \text{on the inflow, } \xi = \text{constant boundary.} \]  
\[ U = u_1, V + C_q z = g_1 \quad \text{on the inflow, } \eta = \text{constant boundary.} \]

\[ U - C_p z = f_2 \quad \text{on outflow, } \xi = \text{constant boundary.} \]
\[ V - C_q z = g_2 \quad \text{on outflow, } \eta = \text{constant boundary.} \]

where

\[ C_p = \sqrt{\frac{g}{H}(\xi_x^2 + \xi_y^2)} \quad \text{and} \quad C_q = \sqrt{\frac{g}{H}(\eta_x^2 + \eta_y^2)} \]  
\[ (3.57) \]

The equation \((3.57)\) is the radiation boundary condition which allows long waves to leave the model domain without reflection. In curvilinear co-ordinates, these boundary conditions may be written as

\[ U = \pm \sqrt{\frac{g}{H}(\xi_x^2 + \xi_y^2)} \quad \text{for } \xi = \text{constant} \]  
\[ (3.58) \]

\[ V = \pm \sqrt{\frac{g}{H}(\eta_x^2 + \eta_y^2)} \quad \text{for } \eta = \text{constant} \]  
\[ (3.59) \]

On the solid walls of the boundary, the following boundary conditions may be used.

\[ U = 0 \quad \text{on } \xi = \text{constant} \]  
\[ (3.60) \]
\[ V = 0 \quad \text{on } \eta = \text{constant} \]

### 3.3.6 Moving boundary formulation

The moving boundary has the property that it can continuously move with time. Its movement is mainly controlled by the topography of the coastal region, the tidal amplitude, the water level associated with a storm surge, etc. Based on
the Lagrangian description of fluid motion, the movement of a boundary can be expressed as

\[ \vec{X}(t) = \vec{X}_0 + \int_0^t \vec{v}_b \, dt \]  

(3.61)

where \( \vec{X}(t) \) is the location of the boundary at the time \( t \), \( \vec{X}_0 \) is the initial location, and \( \vec{v}_b \) is the velocity of the boundary motion.

In equations (3.47) to (3.50) it is seen that, in a moving coordinate system, the grid speed terms affect the water level elevation term in the continuity equation and, in the momentum equations the advective term is affected and a source term is introduced. In order to obtain the boundary condition at a moving boundary let us assume that the boundary, \( \xi = \text{constant} \), is moving. At a moving boundary, the total water depth \( H \) is assumed to be equal to zero. A further condition which is necessarily true at the moving boundary is that the velocity of the fluid at the moving boundary is equal to the velocity of the grid. The above condition guarantees that the physical boundary is determined by the fluid envelope. The above boundary conditions then reduce to:

\[ H = z + h = 0 \]  

(3.62)

For the covariant component of the velocity,

\[ u = x_t, \quad v = y_t \]  

(3.63)

and for the contravariant component of the velocity,

\[ U = -\xi_t, \quad V = -\eta_t \]  

(3.64)

The boundary velocities are obtained as follows. Expanding equation 3.47 we get

\[ z_t + \xi_t z_\xi + \eta_t z_\eta + \frac{1}{J} (VJ) H_\xi + \frac{1}{J} (VJ) H_\eta + \frac{H}{J} ((UJ)_\xi + (VJ)_\eta) = 0 \]  

(3.65)
Using equation 3.62 and 3.64 equation 3.64 becomes

\[ z_t = \xi_t h_\xi + \eta_t h_\eta \] (3.66)

At a \( \xi = \) constant boundary \( \eta_t \) is assumed to be equal to zero. The above condition states that the \( \xi = \) constant boundary moves with a velocity equal to the normal velocity and there is no tangential movement of the boundary. The above assumption is made in order to obtain grids which are not highly skewed at the boundary. At a \( \eta = \) constant boundary, \( \xi_t \) is assumed to be equal to zero. The finite difference representation of the above boundary conditions will be dealt with in subsequent sections.

### 3.4 Finite difference formulation

The governing equations of continuity and momentum are given by the equations (3.47) to (3.50). These equations are nonlinear, and in general, analytical solutions are not available. Thus numerical techniques must be employed to solve these equations along with initial and boundary conditions. The governing equations are expressed in the computational domain which is rectangular and hence finite difference techniques can be used to solve them. In order to solve the governing equations in the computational domain, a grid system must be chosen and the governing equations must be discretized on this grid. Different types of grid systems are possible from nonstaggered to fully staggered grids. In nonstaggered grids all the dependent variables are located at all the grid points. The disadvantage of such a system is that the dependent variables may become spatially decoupled resulting in spurious oscillations Arakawa (1966). A fully staggered grid system eliminates these oscillations and such a system is adopted in this study. The fully staggered
grid with the location of variables is shown in Fig. (3.9). The velocities are located at cell faces, the water level elevation is located at the cell center and the grid coordinates are located at cell vertices. The governing equations are then discretized on this staggered grid.

The characteristic analysis performed in section (3.3.3) showed us that in order to remove the restrictive gravity wave stability criterion the water level elevation terms \( z \) and the friction term must be discretized in an implicit manner in the momentum equation and the velocity terms must discretized implicitly in the continuity equation. This then results in the following discretization.

Continuity

\[
\begin{align*}
\frac{z_{i,j}^{n+1}}{\Delta t} &= Z_{i,j}^n - \Delta t[\xi_t Z_\xi + \eta_t Z_\eta]^n \\
&\quad - \frac{\theta \Delta t}{J_{i,j}^n}[H_{i+\frac{1}{2},j}^n(UJ)_{i+\frac{1}{2},j}^{n+1} - (UJ)_{i-\frac{1}{2},j}^{n+1} H_{i-\frac{1}{2},j}^n]

&\quad - \frac{\theta \Delta t}{J_{i,j}^n}[H_{i,j+\frac{1}{2}}^n(UJ)_{i,j+\frac{1}{2}}^{n+1} - (UJ)_{i,j-\frac{1}{2}}^{n+1} H_{i,j-\frac{1}{2}}^n]

&\quad - (1 - \theta) \Delta t \frac{J_{i,j}^n}{J_{i,j}^n}[H_{i+\frac{1}{2},j}^n(UJ)_{i+\frac{1}{2},j}^n - (UJ)_{i-\frac{1}{2},j}^n H_{i-\frac{1}{2},j}^n]
\end{align*}
\]
\[
\frac{(1 - \theta)\Delta t}{J_{i,j}^n}[H_{i,j+\frac{1}{2}}^n(VJ)_{i,j+\frac{1}{2}}^n - (VJ)_{i,j-\frac{1}{2}}^nH_{i,j-\frac{1}{2}}^n]
\]

**\( \xi \) - Momentum

\[
(JU)_{i,j+\frac{1}{2}}^{n+1} = (JU)_{i,j+\frac{1}{2}}^n - (x_\eta[(U + \xi_\eta)(Uy_\xi + Vy_\eta)\xi + (V + \eta_\eta)(Uy_\xi + Vy_\eta)\eta] - (x_\eta[(U + \xi_\eta)(Ux_\xi + Vx_\eta)\eta] + (V + \eta_\eta)(Ux_\xi + Vx_\eta)\eta)_{i,j+\frac{1}{2}}^n
\]

\[
g\theta \alpha_{i,j+\frac{1}{2}}^n z_{i,j+\frac{1}{2}}^{n+1} + \frac{g\theta \beta_{i,j+\frac{1}{2}}^n}{4J_{i,j+\frac{1}{2}}^n}[z_{i,j+1}^{n+1} - z_{i,j}^{n+1} - z_{i,j-1}^{n+1}]
\]

\[
g(1 - \theta)\gamma_{i,j+\frac{1}{2}}^n [z_{i,j+1}^{n+1} - z_{i,j}^{n+1}]
\]

\[
g(1 - \theta)\gamma_{i,j+\frac{1}{2}}^n [z_{i,j+1}^{n+1} - z_{i,j}^{n+1}]
\]

**\( \eta \) - Momentum

\[
(JV)_{i,j+\frac{1}{2}}^{n+1} = (JV)_{i,j+\frac{1}{2}}^n - (x_\xi[(U + \xi_\xi)(Uy_\xi + Vy_\eta)\xi + (V + \eta_\eta)(Uy_\xi + Vy_\eta)\eta] - (x_\xi[(U + \xi_\xi)(Ux_\xi + Vx_\xi)\xi] + (V + \eta_\eta)(Ux_\xi + Vx_\xi)\eta)_{i,j+\frac{1}{2}}^n
\]

\[
g\theta \alpha_{i,j+\frac{1}{2}}^n z_{i,j+\frac{1}{2}}^{n+1} + \frac{g\theta \beta_{i,j+\frac{1}{2}}^n}{4J_{i,j+\frac{1}{2}}^n}[z_{i,j+1}^{n+1} - z_{i,j}^{n+1} - z_{i,j-1}^{n+1}]
\]

\[
g(1 - \theta)\gamma_{i,j+\frac{1}{2}}^n [z_{i,j+1}^{n+1} - z_{i,j}^{n+1}]
\]

\[
g(1 - \theta)\gamma_{i,j+\frac{1}{2}}^n [z_{i,j+1}^{n+1} - z_{i,j}^{n+1}]
\]
where,

\[ \xi_i = \frac{u_i x_n - x_i y_n}{J} \]
\[ \eta_i = \frac{x_i y_n - u_i x_n}{J} \]

\[ x_t, y_t \quad \text{Grid speed in the physical domain} \]
\[ U \quad \text{Contravariant velocity component normal to } \eta = \text{constant line} \]
\[ V \quad \text{Contravariant velocity component normal to } \xi = \text{constant line} \]
\[ U = \frac{u y_n - u x_n}{J} \quad V = \frac{u x_n - u y_n}{J} \]
\[ \alpha = x_n^2 + y_n^2, \quad \beta = x_x x_n + y_y y_n, \quad \gamma = x_x^2 + y_y^2, \quad J = x_x y_n - x_n y_x \]

\[ x_x, x_y, y_x, y_y \quad \text{are scale factors} \]

Collecting all the terms represented at the previous time level \( n \), results in the following:

**Continuity:**

\[
\begin{align*}
Z^{n+1}_{i,j} &= (CZ)_{i,j}^n + (CZ1)_{i,j}^n(UJ)_{i+\frac{1}{2},j}^n + (CZ2)_{i,j}^n(UJ)_{i-\frac{1}{2},j}^n + \\
&
(CZ3)_{i,j}^n(UJ)_{i,j+\frac{1}{2}}^n + (CZ4)_{i,j}^n(UJ)_{i,j-\frac{1}{2}}^n
\end{align*}
\]

(3.70)

where,

\[
(CZ)_{i,j}^n = Z_{i,j}^n - \frac{(1 - \theta) \Delta t}{J_{i,j}^n} [H_{i+\frac{1}{2},j}^n(UJ)_{i+\frac{1}{2},j}^n - (UJ)_{i-\frac{1}{2},j}^n H_{i-\frac{1}{2},j}^n]
\]

(3.71)

\[
(CZ1)_{i,j}^n = -\frac{\theta \Delta t}{J_{i,j}^n} H_{i+1/2,j}^n
\]

(3.72)

\[
(CZ2)_{i,j}^n = \frac{\theta \Delta t}{J_{i,j}^n} H_{i-1/2,j}^n
\]

(3.73)

\[
(CZ3)_{i,j}^n = -\frac{\theta \Delta t}{J_{i,j}^n} H_{i,j+1/2}^n
\]

(3.74)
\[
\begin{align*}
[f_{1+u}^z - 1+f_{1+u}^z - f_{1+u}^z + 1+f_{1+u}^z]^{\xi + f_{1+u}^z}(z\Lambda) \\
\end{align*}
\]
\begin{align*}
\frac{f_{1+u}^z - 1+f_{1+u}^z}{f_{1+u}^z} + \frac{f_{1+u}^z - 1+f_{1+u}^z}{f_{1+u}^z} = \frac{f_{1+u}^z - 1+f_{1+u}^z}{f_{1+u}^z} (\Lambda) \\
\end{align*}

Where:

\[
\begin{align*}
\frac{1-f_{1+u}^z - 1+f_{1+u}^z - f_{1+u}^z + 1+f_{1+u}^z}{f_{1+u}^z} (z\Lambda) \\
\end{align*}
\]

\[
\begin{align*}
\frac{1-f_{1+u}^z - 1+f_{1+u}^z - f_{1+u}^z + 1+f_{1+u}^z}{f_{1+u}^z} (\Lambda) \\
\end{align*}
\]

\[
\begin{align*}
\frac{1-f_{1+u}^z - 1+f_{1+u}^z - f_{1+u}^z + 1+f_{1+u}^z}{f_{1+u}^z} (\Lambda) \\
\end{align*}
\]
where,

\[(CV)_{i,j+\frac{1}{2}}^n = (JV)_{i,j+\frac{1}{2}}^n - (x_i[(U + \xi) (U y_\xi + V y_\eta) + (V + \eta)(U y_\xi + V y_\eta)])^n - y_i[(U + \xi)(U x_\xi + V x_\eta) + (V + \eta)(U x_\xi + V x_\eta)].\]

\[g(1 - \theta)\gamma_{i,j+\frac{1}{2}}^n [z_{i,j+1}^n - z_{i,j}^n] \]

\[\frac{g(1 - \theta)\beta_{i,j+\frac{1}{2}}^n}{4J_{i,j+\frac{1}{2}}^n} [z_{i+1,j+1}^n + z_{i+1,j}^n - z_{i-1,j+1}^n - z_{i-1,j}^n] \]

\[+ [T]^n x_i - T^n y_i [U x_\xi + V x_\eta] - (y_i)_{\gamma}(U x_\xi + V x_\eta) - (x_i)_{\gamma}(U y_\xi + V y_\eta)]_{i,j+\frac{1}{2}}^n \]

\[-f(\beta_{i,j+\frac{1}{2}}^n U_{i,j+\frac{1}{2}}^n + \alpha_{i,j+\frac{1}{2}}^n V_{i,j+\frac{1}{2}}^n) \]

\[(CV)_{i,j+\frac{1}{2}}^n = \frac{(CV)_{i,j+\frac{1}{2}}^n}{1 + \Gamma_v} \]

\[(CV1)_{i,j+\frac{1}{2}}^n = \frac{g \theta \beta_{i,j+\frac{1}{2}}^n}{4J_{i,j+\frac{1}{2}}^n (1 + \Gamma_v)} \]

\[(CV2)_{i,j+\frac{1}{2}}^n = -\frac{g \theta \gamma_{i,j+\frac{1}{2}}^n}{J_{i,j+\frac{1}{2}}^n (1 + \Gamma_v)} \]

\[\Gamma_v = \left[\frac{g(mann)^2}{H^4/3}\right] \sqrt{\gamma U^2 + 2\beta UV + \alpha V^2}]_{i,j+\frac{1}{2}}^n \]

Similar expressions for \((UJ)_{i-\frac{1}{2},j}^{n+1}\) and \((VJ)_{i,j+\frac{1}{2}}^{n+1}\) are written as

\[(JU)_{i-\frac{1}{2},j}^{n+1} = (CU)_{i-\frac{1}{2},j}^n + (CU1)_{i-\frac{1}{2},j}^n [z_{i,j+1}^{n+1} - z_{i-1,j}^{n+1}] + \]

\[(CU2)_{i-\frac{1}{2},j}^n [z_{i+1,j+1}^{n+1} + z_{i+1,j}^{n+1} - z_{i-1,j+1}^{n+1} - z_{i-1,j}^{n+1}] \]

\[(JV)_{i,j-\frac{1}{2}}^{n+1} = (CV)_{i,j-\frac{1}{2}}^n + (CV1)_{i,j-\frac{1}{2}}^n [z_{i,j+1}^{n+1} - z_{i,j-1}^{n+1}] + \]

\[(CV2)_{i,j-\frac{1}{2}}^n [z_{i+1,j+1}^{n+1} + z_{i+1,j}^{n+1} - z_{i-1,j+1}^{n+1} - z_{i-1,j}^{n+1}] \]

Substituting equations (3.77), (3.82), (3.87) and (3.88) into equation (3.71) results in a system of linear equations in \(z\). This linear system of equations is diagonally
dominant and is solved using a successive over relaxation method. Once the \( z \) values are obtained at all the grid points the velocities are obtained explicitly from equations (3.77) and (3.82).

### 3.4.1 Discretization of metric components

The metric coefficients \( \alpha, \beta, \gamma \) and \( J \) in equations (3.68) to (3.70) also need to be discretized. All the above coefficients are taken at time level \( n \) i.e., they are discretized explicitly. The proper discretization of metric coefficients is very important as it might otherwise lead to the introduction of spurious sources into the model. As shown in Fig. (3.9), the \( x \) and \( y \) coordinates are given at nodal points \( (i + \frac{1}{2}, j + \frac{1}{2}) \) etc. The values of \( x \) and \( y \) at other coordinates is then obtained by simple two point or four point averaging. For example

\[
x_{i,j} = \frac{1}{4}(x_{i-\frac{1}{2},j-\frac{1}{2}} + x_{i+\frac{1}{2},j-\frac{1}{2}} + x_{i+\frac{1}{2},j+\frac{1}{2}} + x_{i-\frac{1}{2},j+\frac{1}{2}})
\]

(3.88)

\[
x_{i-\frac{1}{2},j} = \frac{1}{2}(x_{i-\frac{1}{2},j-\frac{1}{2}} + x_{i-\frac{1}{2},j+\frac{1}{2}})
\]

(3.89)

\[
x_{i,j-\frac{1}{2}} = \frac{1}{2}(x_{i-\frac{1}{2},j-\frac{1}{2}} + x_{i+\frac{1}{2},j-\frac{1}{2}})
\]

(3.90)

Similar values of \( y \) at above locations can be obtained by replacing \( x \) with \( y \) in the above equation. Once the \( x \) and \( y \) coordinates are obtained at all the points using equations (3.88) to (3.90), the metric terms are obtained as follows.

\[
(x_\xi)_{i,j-\frac{1}{2}} = (x_{i+\frac{1}{2},j-\frac{1}{2}} - x_{i-\frac{1}{2},j-\frac{1}{2}})
\]

(3.91)

\[
(x_\eta)_{i,j-\frac{1}{2}} = (x_{i,j} - x_{i,j-1})
\]

(3.92)

\[
(x_\zeta)_{i,j} = (x_{i+\frac{1}{2},j} - x_{i-\frac{1}{2},j})
\]

(3.93)
\[(x_\eta)_{i,j} = (x_{i,j+\frac{1}{2}} - x_{i,j-\frac{1}{2}}) \quad (3.94)\]

\[(x_\xi)_{i-\frac{1}{2},j} = (x_{i,j} - x_{i-1,j}) \quad (3.95)\]

\[(x_\eta)_{i-\frac{1}{2},j} = (x_{i-\frac{1}{2},j+\frac{1}{2}} - x_{i-\frac{1}{2},j-\frac{1}{2}}) \quad (3.96)\]

The values of \(y_\xi\) and \(y_\eta\) are obtained by replacing \(x\) with \(y\) in the above equations. Once the values of \(x_\xi\), \(x_\eta\), \(y_\xi\) and \(y_\eta\) are obtained at all points, the other metric coefficients \(\alpha\), \(\beta\), \(\gamma\) and \(J\) are obtained as follows:

\[\alpha = x_\eta^2 + y_\eta^2 \quad (3.97)\]

\[\alpha_{i,j-\frac{1}{2}} = (x_\eta)_{i,j-\frac{1}{2}}^2 + (y_\eta)_{i,j-\frac{1}{2}}^2 \quad (3.98)\]

\[\alpha_{i,j} = (x_\eta)_{i,j}^2 + (y_\eta)_{i,j}^2 \quad (3.99)\]

\[\alpha_{i-\frac{1}{2},j} = (x_\eta)_{i-\frac{1}{2},j}^2 + (y_\eta)_{i-\frac{1}{2},j}^2 \quad (3.100)\]

\[\beta = x_\xi x_\eta + y_\xi y_\eta \quad (3.101)\]

\[\beta_{i,j-\frac{1}{2}} = (x_\xi)_{i,j-\frac{1}{2}}(x_\eta)_{i,j-\frac{1}{2}} + (y_\xi)_{i,j-\frac{1}{2}}(y_\eta)_{i,j-\frac{1}{2}} \quad (3.102)\]

\[\beta_{i,j} = (x_\xi)_{i,j}(x_\eta)_{i,j} + (y_\xi)_{i,j}(y_\eta)_{i,j} \quad (3.103)\]

\[\beta_{i-\frac{1}{2},j} = (x_\xi)_{i-\frac{1}{2},j}(x_\eta)_{i-\frac{1}{2},j} + (y_\xi)_{i-\frac{1}{2},j}(y_\eta)_{i-\frac{1}{2},j} \quad (3.104)\]

\[\gamma = x_\xi^2 + y_\xi^2 \quad (3.105)\]

\[\gamma_{i,j-\frac{1}{2}} = (x_\xi)_{i,j-\frac{1}{2}}^2 + (y_\xi)_{i,j-\frac{1}{2}}^2 \quad (3.106)\]
\[ \gamma_{i,j} = (x\xi)^2_{i,j} + (y\xi)^2_{i,j} \quad (3.107) \]

\[ \gamma_{i-\frac{1}{2},j} = (x\xi)^2_{i-\frac{1}{2},j} + (y\xi)^2_{i-\frac{1}{2},j} \quad (3.108) \]

The Jacobian J is given by

\[ J = x\xi y_n - x_n y\xi \quad (3.109) \]

\[ J_{i,j-\frac{1}{2}} = (x\xi)_{i,j-\frac{1}{2}}(y_n)_{i,j-\frac{1}{2}} - (x_n)_{i,j-\frac{1}{2}}(y\xi)_{i,j-\frac{1}{2}} \quad (3.110) \]

\[ J_{i,j} = (x\xi)_{i,j}(y_n)_{i,j} - (x_n)_{i,j}(y\xi)_{i,j} \quad (3.111) \]

\[ J_{i-\frac{1}{2},j} = (x\xi)_{i-\frac{1}{2},j}(y_n)_{i-\frac{1}{2},j} - (x_n)_{i-\frac{1}{2},j}(y\xi)_{i-\frac{1}{2},j} \quad (3.112) \]

It should be noted that all the above metric coefficients are calculated at the previous time level n.

### 3.4.2 Discretization of boundary condition

The grid is generated such that at closed boundaries velocities are prescribed and at open boundaries water level elevations are prescribed at inflow and a radiation condition is prescribed at the outflow boundary. This then results in four types of boundary condition.

#### 3.4.2.1 Closed boundary represented by a solid wall

If the closed boundary is a solid wall, the normal velocity to the boundary is set equal to zero. For an eastern and a western boundary, this leads to the condition \( UJ = 0 \), and for a northern and a southern boundary \( VJ = 0 \).
3.4.2.2 Open inflow boundary

At an inflow boundary, the water level elevation is prescribed as a function of time, \( z = f(t) \) at an inflow boundary.

3.4.2.3 Open outflow boundary

At an outflow boundary, a radiation boundary condition is used. This allows the long waves to pass through the boundary without reflections. For an eastern boundary this condition is given by

\[
UJ = -\sqrt{\frac{g}{H}(x^2 + y^2)}
\]

and for a western boundary

\[
UJ = \sqrt{\frac{g}{H}(x^2 + y^2)}
\]

For a northern boundary

\[
VJ = \frac{g}{H}(x\xi + y\eta)
\]

and for a southern boundary

\[
VJ = -\frac{g}{H}(x\xi + y\eta)
\]

3.4.2.4 Closed moving boundary

The moving boundary condition for a closed boundary is given by \( H = 0 \) (total water depth is zero), and the covariant velocities \( u \) and \( v \) are equal to the grid speeds \( x_t \) and \( y_t \). The contravariant velocities \( UJ \) and \( VJ \) are equal to \(-J\xi_t\) and \(-J\eta_t\) respectively. This results in a boundary condition of the form

\[
z_t = \frac{(J\xi_t)h\xi}{J} + \frac{(J\eta_t)h\eta}{J}
\]
At eastern and western boundaries the above equation is discretized as follows. 

As mentioned previously at closed eastern and western boundaries only the normal velocity \( U_i \) is present. The water level elevation at the boundary is then obtained by a linear extrapolation from the next two interior water level elevation points. In order to illustrate this let us assume that the eastern boundary in Fig. (3.9) is the moving boundary. The water level elevation at a \( i - \frac{3}{2}, j \) location is obtained by

\[
\eta_i^{n+1} = \frac{3}{2} \eta_{i-1,j}^{n+1} - \frac{1}{2} \eta_{i,j}^{n+1}
\]

(3.118)

It is also assumed that \( \xi_i^{n+1} \) at \( i - \frac{3}{2}, j \) is zero. This then results in the following formulation for \( \xi_i^{n+1} \) at the grid point \( i - \frac{3}{2}, j \).

\[
(\xi_i^{n+1})_{i-\frac{3}{2},j} = \frac{(Z_i^{n+1} - Z_{i-\frac{3}{2},j}^{n})}{\Delta t(h_i^{n})_{i-\frac{3}{2},j}}
\]

(3.119)

Once the values of \((\xi_i^{n+1})_i\) along all the moving boundaries are known, the physical grid speeds \((x_t, y_t)\) are obtained by

\[
(\xi_i^{n+1})_{i-\frac{3}{2},j} = (y_t^{n+1})_{i-\frac{3}{2},j}(\xi_i^{n+1})_{i-\frac{3}{2},j} - (x_t^{n+1})_{i-\frac{3}{2},j}(\eta_i^{n+1})_{i-\frac{3}{2},j}
\]

(3.120)

The value of \( y_t^{n+1} \) on an eastern boundary is obtained by a zeroth order extrapolation of the covariant velocity \( v \) from the interior of the domain. For example,

\[
(y_t^{n+1})_{i-\frac{3}{2},j} = \frac{1}{2}(v_{i-1,j+1/2}^{n+1} + v_{i-1,j-1/2}^{n+1})
\]

(3.121)

The values of \((x_t^{n+1})_{i-\frac{3}{2},j}\) are obtained from equation (3.120). An analogous procedure is used for the other moving boundaries. Once the grid speeds \( x_t^{n+1} \) and \( y_t^{n+1} \) are determined at all the moving boundaries, the values of \( x \) and \( y \) at the new time level \( n+1 \) are determined by

\[
x^{n+1} = x^n + (x_t)^{n+1} \Delta t
\]

(3.122)
\[ y^{n+1} = y^n + (y_t)^{n+1} \Delta t \]  

(3.123)

where \( \Delta t \) is the time step. Using the new boundary values of \( x \) and \( y \) the elliptic grid generator is used to regenerate a new grid. The interior values of \( x \) and \( y \) obtained from the new grid are used to calculate the interior grid speeds \( x_{t}^{n+1}, y_{t}^{n+1} \) by first order backward time differencing of the grid points.

\[
(x_t)^{n+1} = \frac{x^{n+1} - x^n}{\Delta t} \tag{3.124}
\]

\[
(y_t)^{n+1} = \frac{y^{n+1} - y^n}{\Delta t} \tag{3.125}
\]

Once \( (x_t)^{n+1} \) and \( (y_t)^{n+1} \) are known at all the interior points, the values of \( (J_{\xi_t})^{n+1} \) and \( (J_{\eta_t})^{n+1} \) can be calculated using

\[
(J_{\xi_t})^{n+1} = (y_t)^{n+1}(x_{\eta})^{n+1} - (x_t)^{n+1}(y_{\eta})^{n+1} \tag{3.126}
\]

\[
(J_{\eta_t})^{n+1} = (x_t)^{n+1}(y_{\xi})^{n+1} - (y_t)^{n+1}(x_{\xi})^{n+1} \tag{3.127}
\]

### 3.5 Stability analysis

The stability analysis of the finite difference method described in the previous sections is carried out using the Von Neumann method under the assumption that the differential equations are linear and are defined on an infinite spatial domain, or with periodic boundary conditions on a finite domain. The above assumption is necessary as stability theories for non linear equations with arbitrary initial and boundary conditions are yet to be developed. To this end, the difference equations (3.68) to (3.70) are linearized. The continuity equation is linearized by assuming the total water depth \( H \) is a constant. The momentum equations are linearized
by assuming the grid speed and advection terms are equal to zero. The linearized difference equations are written as

Continuity

\[
\begin{align*}
&z^{n+1}_{i,j} - z^n_{i,j} + \frac{H \Delta t}{J^n_{i,j}} \theta[(UJ)^{n+1}_{i+\frac{1}{2},j} - (UJ)^{n+1}_{i-\frac{1}{2},j}] \\
&+ \frac{H \Delta t}{J^n_{i,j}} \theta[(VJ)^{n+1}_{i,j+\frac{1}{2}} - (VJ)^{n+1}_{i,j-\frac{1}{2}}] \\
&- \frac{H \Delta t}{J^n_{i,j}} (1 - \theta)[(UJ)^n_{i+\frac{1}{2},j} - (UJ)^n_{i-\frac{1}{2},j}] \\
&- \frac{H \Delta t}{J^n_{i,j}} (1 - \theta)[(VJ)^n_{i,j+\frac{1}{2}} - (VJ)^n_{i,j-\frac{1}{2}}] = 0
\end{align*}
\]

\(\xi\) - Momentum

\[
\begin{align*}
&(JU)^{n+1}_{i+\frac{1}{2},j}(\sqrt{1 + \Gamma_u \Delta t}) - (JU)^n_{i+\frac{1}{2},j} + \frac{g \alpha^n_{i+\frac{1}{2},j}}{J^n_{i+\frac{1}{2},j}} \Delta t \theta[z^{n+1}_{i+1,j} - z^n_{i,j}] \\
&- \frac{g \beta^n_{i+\frac{1}{2},j}}{4J^n_{i+\frac{1}{2},j}} \Delta t \theta[z^{n+1}_{i,j+1} + z^{n+1}_{i+1,j+1} - z^{n+1}_{i,j-1} - z^{n+1}_{i+1,j-1}] \\
&+ \frac{g \alpha^n_{i+\frac{1}{2},j}}{J^n_{i+\frac{1}{2},j}} (1 - \theta)[z^n_{i+1,j} - z^n_{i,j}] \\
&- \frac{g \beta^n_{i+\frac{1}{2},j}}{4J^n_{i+\frac{1}{2},j}} (1 - \theta)[z^n_{i,j+1} + z^n_{i+1,j+1} - z^n_{i,j-1} - z^n_{i+1,j-1}] = 0
\end{align*}
\]

\(\eta\) - Momentum

\[
\begin{align*}
&(JV)^{n+1}_{i,j+\frac{1}{2}}(\sqrt{1 + \Gamma_v \Delta t}) - (JV)^n_{i,j+\frac{1}{2}} + \frac{g \gamma^n_{i,j+\frac{1}{2}}}{J^n_{i,j+\frac{1}{2}}} \Delta t \theta[z^{n+1}_{i,j+1} - z^n_{i,j}] \\
&- \frac{g \beta^n_{i,j+\frac{1}{2}}}{4J^n_{i,j+\frac{1}{2}}} \Delta t \theta[z^{n+1}_{i+1,j+1} + z^{n+1}_{i+1,j} - z^{n+1}_{i-1,j+1} - z^{n+1}_{i-1,j}] \\
&+ \frac{g \gamma^n_{i,j+\frac{1}{2}}}{J^n_{i,j+\frac{1}{2}}} (1 - \theta)[z^n_{i,j+1} - z^n_{i,j}] \\
&- \frac{g \beta^n_{i,j+\frac{1}{2}}}{4J^n_{i,j+\frac{1}{2}}} (1 - \theta)[z^n_{i+1,j+1} + z^n_{i+1,j} - z^n_{i-1,j+1} - z^n_{i-1,j}] = 0
\end{align*}
\]
The above equations are the linearized continuity and momentum equations. Now, by changing the variables $U_J$, $V_J$ and $z$ with $\tilde{U} = U J \sqrt{1 + \frac{\Gamma_u \Delta t}{u}}$ and $\tilde{V} = V J \sqrt{1 + \frac{\Gamma_v \Delta t}{u}}$ respectively, and variable $\tilde{Z} = z \sqrt{\frac{\rho}{H}}$, Equations (3.129) to (3.131) become

Continuity

\begin{align*}
\tilde{Z}^{n+1}_{i,j} - \tilde{Z}^n_{i,j} &= \frac{H \Delta t}{J^n_{i,j}} \frac{\sqrt{gH}}{\sqrt{1 + \Gamma_u \Delta t}} \theta[(\tilde{U})^{n+1}_{i+\frac{1}{2},j} - (\tilde{U})^{n+1}_{i-\frac{1}{2},j}] \\
&+ \frac{H \Delta t}{J^n_{i,j}} \frac{\sqrt{gH}}{\sqrt{1 + \Gamma_v \Delta t}} \theta[(\tilde{V})^{n+1}_{i,j+\frac{1}{2}} - (\tilde{V})^{n+1}_{i,j-\frac{1}{2}}] \\
&- \frac{H \Delta t}{J^n_{i,j}} \frac{\sqrt{gH}}{\sqrt{1 + \Gamma_v \Delta t}} (1 - \theta)[(\tilde{U})^n_{i+\frac{1}{2},j} - (\tilde{U})^n_{i-\frac{1}{2},j}] \\
&- \frac{H \Delta t}{J^n_{i,j}} \frac{\sqrt{gH}}{\sqrt{1 + \Gamma_v \Delta t}} (1 - \theta)[(\tilde{V})^n_{i,j+\frac{1}{2}} - (\tilde{V})^n_{i,j-\frac{1}{2}}] = 0
\end{align*}

$\xi$ - Momentum

\begin{align*}
(\tilde{U})^{n+1}_{i+\frac{1}{2},j} - (\tilde{U})^n_{i+\frac{1}{2},j} &= 0 \\
&+ \frac{g \alpha^n_{i+\frac{1}{2},j} \Delta t}{J^n_{i+\frac{1}{2},j}} \frac{\sqrt{gH}}{\sqrt{1 + \Gamma_u \Delta t}} \theta[\tilde{Z}^{n+1}_{i+1,j} - \tilde{Z}^n_{i,j}] \\
&- \frac{g \beta^n_{i+\frac{1}{2},j} \Delta t}{4J^n_{i+\frac{1}{2},j}} \frac{\sqrt{gH}}{\sqrt{1 + \Gamma_u \Delta t}} \theta[\tilde{Z}^{n+1}_{i+1,j+1} + \tilde{Z}^{n+1}_{i+1,j+1} - \tilde{Z}^{n+1}_{i,j} - \tilde{Z}^{n+1}_{i+1,j-1}] \\
&+ \frac{g \alpha^n_{i+\frac{1}{2},j} \Delta t}{J^n_{i+\frac{1}{2},j}} \frac{\sqrt{gH}}{\sqrt{1 + \Gamma_v \Delta t}} (1 - \theta)[\tilde{Z}^n_{i+1,j} - \tilde{Z}^n_{i,j}] \\
&- \frac{g \beta^n_{i+\frac{1}{2},j} \Delta t}{4J^n_{i+\frac{1}{2},j}} \frac{\sqrt{gH}}{\sqrt{1 + \Gamma_v \Delta t}} (1 - \theta)[\tilde{Z}^n_{i,j+1} + \tilde{Z}^n_{i+1,j+1} - \tilde{Z}^n_{i,j-1} - \tilde{Z}^n_{i+1,j-1}] = 0
\end{align*}

$\eta$ - Momentum

\begin{align*}
(\tilde{V})^{n+1}_{i,j+\frac{1}{2}} - (\tilde{V})^n_{i,j+\frac{1}{2}} &= 0 \\
&+ \frac{g \gamma^n_{i,j+\frac{1}{2}} \Delta t}{J^n_{i,j+\frac{1}{2}}} \frac{\sqrt{gH}}{\sqrt{1 + \Gamma_u \Delta t}} \theta[\tilde{Z}^{n+1}_{i,j+1} - \tilde{Z}^n_{i,j}] \\
&- \frac{g \beta^n_{i,j+\frac{1}{2}} \Delta t}{4J^n_{i,j+\frac{1}{2}}} \frac{\sqrt{gH}}{\sqrt{1 + \Gamma_v \Delta t}} \theta[\tilde{Z}^{n+1}_{i+1,j+1} + \tilde{Z}^{n+1}_{i+1,j+1} - \tilde{Z}^{n+1}_{i+1,j} - \tilde{Z}^{n+1}_{i-1,j}]
\end{align*}
In order to perform the stability analysis of the system of equations (3.132) to (3.134) with the Von Neumann method, a Fourier mode is introduced for each field variable $\hat{U}$, $\hat{V}$ and $\hat{Z}$ and stability analysis is carried out on the corresponding amplitude functions. Specifically $\hat{U}$, $\hat{V}$ and $\hat{Z}$ are written as

\[
(\hat{U})_{i,j}^{n} = \Lambda^{n}\hat{U}e^{i(i+\frac{1}{2})\phi+i\psi} \quad (3.134)
\]

\[
(\hat{V})_{i,j}^{n+\frac{1}{2}} = \Lambda^{n}\hat{V}e^{i\phi+i(j+\frac{1}{2})\psi} \quad (3.135)
\]

\[
(\hat{Z})_{i,j}^{n} = \Lambda^{n}\hat{Z}e^{i\phi+i\psi} \quad (3.136)
\]

where, $\Lambda^{n}\hat{U}$, $\Lambda^{n}\hat{V}$ and $\Lambda^{n}\hat{Z}$ are amplitude functions of $\hat{U}$, $\hat{V}$ and $\hat{Z}$ at time level $n$, $\iota = \sqrt{-1}$, and $\phi$ and $\psi$ are the $\xi$ and $\eta$ direction phase angles.

Substituting equations (3.134) to (3.136) into equations (3.132) to (3.134) we get

\[
\dot{Z}(\Lambda - 1) + \frac{\Delta t\sqrt{gH}2\iota\sin \frac{\phi}{2}}{J_{i,j}^{n}\sqrt{1 + \Gamma\Delta t}}(\Lambda\theta + (1 - \theta)) = 0 \quad (3.137)
\]

\[
\dot{U}(\Lambda - 1) + \frac{\Delta t\alpha_{i,j}^{n+\frac{1}{2}}\sqrt{gH}2\iota\sin \frac{\phi}{2}}{J_{i,j}^{n+\frac{1}{2}}\sqrt{1 + \Gamma\Delta t}}(\Lambda\theta + (1 - \theta)) = 0 \quad (3.138)
\]

\[
- \frac{\Delta t\beta_{i,j}^{n}}{J_{i,j}^{n+\frac{1}{2}}\sqrt{1 + \Gamma\Delta t}}(\Lambda\theta + (1 - \theta)) = 0
\]
\[ \tilde{V}(\Lambda - 1) + \frac{\Delta t \gamma_{i,j+1/2}^n \sqrt{gH} \tilde{Z}_{2t} \sin \frac{\phi}{2}}{J_{i,j+1/2}^n \sqrt{1 + \Gamma_u \Delta t}} (\Lambda \theta + (1 - \theta)) \]  
\[ \tilde{U}(\Lambda - 1) + \frac{\Delta t \beta_{i,j+1/2}^n \sqrt{gH} \tilde{Z}_{2t} \cos \frac{\phi}{2} \sin \phi}{J_{i,j+1/2}^n \sqrt{1 + \Gamma_u \Delta t}} = 0 \]  
(3.139)

Equations (3.138) to (3.140) can be written

\[ \tilde{Z}(\Lambda - 1) + r \tilde{U}(\Lambda \theta + (1 - \theta)) + s \tilde{V}(\Lambda \theta + (1 - \theta)) = 0 \]  
(3.140)

\[ \tilde{U}(\Lambda - 1) + \tilde{Z}(\Lambda \theta + (1 - \theta))(p_1 + p_2) = 0 \]  
(3.141)

\[ \tilde{V}(\Lambda - 1) + \tilde{Z}(\Lambda \theta + (1 - \theta))(q_1 + q_2) = 0 \]  
(3.142)

where,

\[ p_1 = \frac{\Delta t \gamma_{i,j+1/2}^n \sqrt{gH} \tilde{Z}_{2t} \sin \frac{\phi}{2}}{J_{i,j+1/2}^n \sqrt{1 + \Gamma_u \Delta t}} \]

\[ p_2 = -\frac{\Delta t \beta_{i,j+1/2}^n \sqrt{gH} \tilde{Z}_{2t} \cos \frac{\phi}{2} \sin \phi}{J_{i,j+1/2}^n \sqrt{1 + \Gamma_u \Delta t}} \]

\[ q_1 = \frac{\Delta t \gamma_{i,j+1/2}^n \sqrt{gH} \tilde{Z}_{2t} \sin \frac{\phi}{2}}{J_{i,j+1/2}^n \sqrt{1 + \Gamma_u \Delta t}} \]

\[ q_2 = -\frac{\Delta t \beta_{i,j+1/2}^n \sqrt{gH} \tilde{Z}_{2t} \cos \frac{\phi}{2} \sin \phi}{J_{i,j+1/2}^n \sqrt{1 + \Gamma_u \Delta t}} \]

\[ r = \frac{\Delta t \sqrt{gH} \tilde{U}_{2t} \sin \frac{\phi}{2}}{J_{i,j}^n \sqrt{1 + \Gamma_u \Delta t}} \]

\[ s = \frac{\Delta t \sqrt{gH} \tilde{V}_{2t} \sin \frac{\phi}{2}}{J_{i,j}^n \sqrt{1 + \Gamma_u \Delta t}} \]
For purposes of analysis, equations (3.140) to (3.142) are written in matrix form
\[ A\mathbf{X} = 0 \]
where,
\[ A = \begin{pmatrix}
\Lambda - 1 & 0 & \ell (\Lambda \theta + 1 - \theta) (p_1 + p_2) \\
0 & \Lambda - 1 & \ell (\Lambda \theta + 1 - \theta) (q_1 + q_2) \\
\ell \tau (\Lambda \theta + 1 - \theta) & \ell \tau (\Lambda \theta + 1 - \theta) & \Lambda - 1
\end{pmatrix} \]
and,
\[ \mathbf{X} = [\hat{U} \hat{V} \hat{Z}]^T \]

The Neumann necessary and sufficient condition for stability states that the magnification factor \( \Lambda \) satisfy the inequality
\[ \max |\Lambda_{1,2,3}| \leq 1 \quad (3.143) \]
where \( \Lambda_{1,2,3} \) are the roots of the equations
\[ \det A = 0 \quad (3.144) \]

Solution of equation (3.144) yields the roots
\[ \Lambda_1 = 1 \quad \Lambda_2 = \frac{\Upsilon \theta (\theta - 1) + \ell \sqrt{\Upsilon}}{1 + \theta^2 \Upsilon} \quad \Lambda_3 = \frac{\Upsilon \theta (\theta - 1) - \ell \sqrt{\Upsilon}}{1 + \theta^2 \Upsilon} \quad (3.145) \]
where,
\[ \Upsilon = r(p_1 + p_2) + s(q_1 + q_2) \]

Applying equation (3.143) to the three roots yields
\[ (\Upsilon \theta (\theta - 1))^2 + \Upsilon \leq (1 + \theta^2 \Upsilon)^2 \quad (3.146) \]
The inequality in equation (3.146) is satisfied if \( \theta \geq \frac{1}{2} \). The satisfaction of equation (3.146) thus depends only on the implicitization coefficient \( \theta \) and not on \( \Upsilon \), which represents the flow properties. Thus, the method is unconditionally stable for all values of \( \theta \) in the range \( \frac{1}{2} \leq \theta \leq 1 \).
3.6 Solution algorithm

A broad outline of the steps involved in the mathematical model is given below:

1) An initial grid of the physical is generated using the elliptic grid generator.
2) All flow variables are initialized to zero.
3) The boundary conditions are prescribed at open boundaries.
4) The finite difference equations are solved and the flow variables are updated at the new time level. This calculation is implicit.
5) The moving boundary locations are calculated explicitly using the updated flow variables.
6) With the new boundary locations the grid generator is used to generate a new grid, and hence obtain the new x and y coordinate values in the interior of the domain.
7) The grid speeds are calculated next using a backward time differencing using the recently calculated grid locations and the grid locations at the previous time level. These grid speeds are grid speeds in the physical domain.
8) Once the grid speeds are known, the contravariant grid speeds $\xi_i$ and $\eta_i$ in the computational domain are calculated. These contravariant grid speeds appear in the advective terms in the momentum equation and as a source term in the continuity equation.
9) The time step is increased by one and steps 3 to 7 are repeated until the total simulation time is reached.
Chapter 4

Model Testing

In this section some simple tests are performed on the numerical model developed in the previous chapter. The numerical model is also compared with some well known analytical solutions. The first test performed on the model is the ‘no flow test’. In this test all the initial and boundary conditions are set equal to zero, and the model is run for about a week of model time. If the finite difference equations are represented correctly then all the variables in the model should not change from the initial state i.e. all model independent variables should remain equal to zero. The test although very simple gives a lot of information about the model behavior and is a very useful debugging tool. If the model is properly formulated, there should be no spurious sources or sinks. The model performed well in the ‘no flow test’ and the independent variables remained at their initial state even after 14 days of simulation time.

4.1 Comparison with an analytical solution with friction effect for a rectangular basin

The model developed, though valid for a general curvilinear domain is equally applicable to rectangular domain. With this in mind, the model is compared with an analytical solution to the two dimensional shallow water equations in a rectangular domain. With coriolis and advective terms deleted, the shallow water equation in
a rectangular domain are written as

\[(u)_t + g(z)_x + \frac{T^{bx}}{\rho h} = 0 \quad (4.1)\]

\[(v)_t + g(z)_y + \frac{T^{by}}{\rho h} = 0 \quad (4.2)\]

\[(z)_t + (uh)_x + (vh)_y = 0 \quad (4.3)\]

where \(\rho\) is the density of water. Assume the water depth \(h\) is constant and the bottom friction can be calculated by linear friction formulae.

\[T^{bx} = \rho F u \quad (4.4)\]

\[T^{by} = \rho F v \quad (4.5)\]

\[F = \frac{g\sqrt{u_{\text{max}}^2 + v_{\text{max}}^2}}{C^2} \quad (4.6)\]

where \(u_{\text{max}}\) and \(v_{\text{max}}\) are the maximum \(u\) and \(v\) velocities, and \(C\) is the Chezy coefficient. Given these, equations (4.1), (4.2) and (4.3) can be written as

\[u_t + g z_x + F u = 0 \quad (4.7)\]

\[v_t + g z_y + F v = 0 \quad (4.8)\]

\[z_t + (uh)_x + (vh)_y = 0 \quad (4.9)\]

The boundary conditions prescribed are

\[z(0, y, t) = z(t) \quad (4.10)\]

\[u(l, y, t) = 0 \quad (4.11)\]
\begin{align*}
v(x, 0, t) &= 0 \quad (4.12) \\
v(x, b, t) &= 0 \quad (4.13)
\end{align*}

where \( z(t) \) is the water level elevation as a function of time at the mouth of the domain. Subject to the above boundary conditions Rahman (1982) proposed an analytical solution to equations (4.7), (4.8) and (4.9) as follows.

\begin{align*}
z &= Re\left( A_0 \cos(kx) + \tan(kl)\sin(kx) \exp(-\omega t) \right) \\
&+ \sum_{n=1}^{\infty} \left[ A_n \cos(K_{1n}(b - y)) \cos(K_{2n}x) \\
&+ \tan(K_{2n}l) \sin(K_{2n}x) \exp(-\omega t) \right] \\
u &= Re\left( \frac{-g}{F - \omega} A_0 k(-\sin(kx) + \tan(kl)\cos(kx) \exp(-\omega t) \right) \\
&+ \frac{-g}{F - \omega} \sum_{n=1}^{\infty} \left[ A_n K_{2n} \cos[K_{1n}(b - y)] - \\
&\sin(K_{2n}x) + \tan(K_{2n}l) \sin(K_{2n}x) \exp(-\omega t) \right] \\
v &= Re\left( \frac{-g}{F - \omega} \sum_{n=1}^{\infty} \left[ A_n K_{1n} \sin[K_{1n}(b - y)] \\
&\cos(K_{2n}x) + \tan(K_{2n}l) \sin(K_{2n}x) \exp(-\omega t) \right] \right)
\end{align*}

where \( b \) and \( l \) are the width and length of the channel respectively, \( g \) is the acceleration due to gravity, \( \nu = \sqrt{-1} \), and \( \omega = \frac{\pi}{T} \) where \( T \) is the time period of the wave and

\begin{align*}
K_{1n} &= \frac{n\pi}{b} \quad (4.17) \\
k^2 &= \frac{\omega^2}{gh} \left( 1 + \frac{lF}{\omega} \right) \quad (4.18) \\
K_{2n}^2 &= \frac{\omega^2}{gh} \left( 1 + \frac{lF}{\omega} \right) - \left( \frac{n\pi}{b} \right)^2 \quad (4.19)
\end{align*}
The coefficients $A_0, ..., A_n$ can be obtained by using the condition

$$z(0, y, t) = z(t)e^{x(-i\omega t)}$$  \hspace{1cm} (4.20)

at $x=0$. They are given as

$$A_0 = \int_0^b zyy$$  \hspace{1cm} (4.21)

$$A_n = \int_0^b zcosK_1n(b - y)dy$$  \hspace{1cm} (4.22)

![Figure 4.1: Rectangular basin closed at three ends](image-url)
Figure 4.2: Comparison of analytical and numerical solutions for water level elevations at a point 2500 m from the open boundary.

Figure 4.3: Comparison of analytical and numerical solutions for water level elevations at a point 5000 m from the open boundary.

Figure 4.4: Comparison of analytical and numerical solutions for water level elevations at a point 7500 m from the open boundary.
Figure 4.5: Comparison of analytical and numerical solutions for u - velocity at a point 2500 m from the open boundary

Figure 4.6: Comparison of analytical and numerical solutions for u - velocity at a point 5000 m from the open boundary

Figure 4.7: Comparison of analytical and numerical solutions for u - velocity at a point 7500 m from the open boundary
To compare the numerical solution with the analytical solution, a rectangular basin, closed at three ends (Fig. (4.1)) is used. The basin is 10,000 m long 5000m wide and is closed at y = 0, y = 5000m and x = 10,000m. A sinusoidal tidal elevation with a constant amplitude of 50 cm is prescribed at x =0, which results in no flow in the y-direction, i.e. v = 0, and zero values for coefficients $A_1, \ldots, A_n$. The period of the forcing tide is 12 hrs. The Manning coefficient is 0.02 and the maximum velocity is 50 cm/sec. Thus the analytical solution to the tidal responses inside the basin can be obtained from equations (4.15) and (4.16). With a time step of 300sec the numerical solution is obtained.

The comparison between numerical and analytical solutions for water level elevations at three locations is shown Figures (4.2), (4.3) and (4.4). Figures (4.5), (4.6) and (4.7) show the variation of u velocities at three points in the basin. The numerical results are almost identical to the analytical solutions indicating that the numerical solutions are valid.

4.2 Comparison with an analytical solution in a parabolic basin

The numerical model is compared with an analytical solution for the oscillation of a body of water in a parabolic basin. These analytical solutions were presented by Thacker(1981). The motion of water in shallow basins is governed by the shallow water wave equations,

\[ u_t + u(u)_x + v(u)_y - f v + g(z)_x = 0 \]  
\[ v_t + u(v)_x + v(v)_y + f u + g(z)_y = 0 \]
\[ z_t + (uH)_x + (vH)_y = 0 \]  
\hspace{1.5in} (4.25)

where \( u \) and \( v \) are velocities corresponding to the orthogonal directions \( x \) and \( y \), \( z \) is the water level elevation, \( H \) is the total water depth \( (H = h + z) \) and \( h \) is the local water depth, \( f \) is the coriolis parameter and \( g \) is the acceleration due to gravity. The instantaneous shoreline is determined by the condition \( H = 0 \). It follows from equation (4.25) that the volume of water within the region for which the total depth \( H \) is positive remains constant in time as the shoreline moves about.

The undisturbed water depth in the basin is assumed to be given by

\[ h = h_0(1 - \frac{x^2}{L^2} - \frac{y^2}{L^2}) \]  
\hspace{1.5in} (4.26)

The above equation represents a circular basin of radius \( L \) with an undisturbed water depth \( h_0 \) at the center. The assumption of uniform flow results in the condition \( u_x = u_y = v_x = v_y = 0 \), and the following dispersion relationship is obtained.

\[ (\omega^2 - \frac{2gh_0}{L^2})^2 - f^2\omega^2 = 0 \]  
\hspace{1.5in} (4.27)

This then results in two basic solutions:

\[ u = -d_1\omega_1 \sin \omega_1 t \]  
\hspace{1.5in} (4.28)

\[ v = -d_1\omega_1 \cos \omega_1 t \]  
\hspace{1.5in} (4.29)

\[ z = \frac{2d_1h_0}{L} \left( \frac{x}{L} \cos \omega_1 t - \frac{y}{L} \sin \omega_1 t - \frac{d_1}{2L} \right) \]  
\hspace{1.5in} (4.30)

\[ \omega_1 = \frac{f}{2} + \left[ \frac{f^2}{4} + \frac{2gh_0}{L^2} \right]^{1/2} \]  
\hspace{1.5in} (4.31)

and

\[ u = -d_1\omega_2 \sin \omega_2 t \]  
\hspace{1.5in} (4.32)
\[ v = -d_1 \omega_2 \cos \omega_2 t \quad (4.33) \]

\[ z = \frac{2d_1 h_0}{L} \left( \frac{x}{L} \cos(\omega_2 t) - \frac{y}{L} \sin(\omega_2 t) - \frac{d_1}{2L} \right) \quad (4.34) \]

\[ \omega_2 = -\frac{f}{2} + \left[ \frac{f^2}{4} + \frac{2gh_0}{L^2} \right]^{1/2} \quad (4.35) \]

where, the constant \( d_1 \) determines the amplitude of the motion. For equations \( 4.28 \) to \( 4.31 \), the shoreline consists of points \((x, y)\) satisfying

\[ (x - d_1 \cos \omega_1 t)^2 + (y + d_1 \sin \omega_1 t)^2 = L^2 \quad (4.36) \]

and for equations \((4.32)\) to \((4.35)\)

\[ (x - d_1 \cos \omega_2 t)^2 + (y + d_1 \sin \omega_2 t)^2 = L^2 \quad (4.37) \]

In both cases the moving shoreline is a circle in the \((x, y)\) plane, and the motion is such that the center of the circle orbits around the center of the basin. The above problem is an initial value problem.

The numerical implementation consists of providing initial conditions for \( u, v, z, x, \) and \( y \) at time \( t = 0 \). A basin of radius 100 Km is chosen and the value of the constant \( d_1 \) is taken as 5000 m. The grid in physical space is shown in Fig. \((4.8)\) and the corresponding computational domain is shown in Fig. \((4.9)\). Since there are no frictional effects the basin oscillates indefinitely. The initial conditions are provided by equations \((4.28)\) to \((4.31)\).
Figure 4.8: Grid in physical domain for the circular basin

Figure 4.9: The computational domain for the circular basin
Figure 4.10: Comparison of analytical and numerical solutions for water level elevations at grid point (10,10)

Figure 4.11: Comparison of analytical and numerical solutions for water level elevations at grid point (15,15)

Figure 4.12: Comparison of analytical and numerical solutions for water level elevations at grid point (18,18)
Figure 4.13: Comparison of movement of the X-coordinate of grid point (1,10) located on the western boundary

Figure 4.14: Comparison of movement of the Y-coordinate of grid point (1,10) located on the western boundary
Figure 4.15: Comparison of movement of the X-coordinate of grid point (21,10) located on the western boundary

Figure 4.16: Comparison of movement of the Y-coordinate of grid point (21,10) located on the western boundary
Once the initial conditions are provided, the numerical model is solved on a grid system shown in Fig. (4.8). Figures (4.10), (4.11) and (4.12) show the water level elevations at three grid points (10,10), (15,15) and (18,18). The numerically predicted values agree very closely with analytical solutions. Figures (4.13) to (4.16) show the comparison between numerically predicted and analytical solution for the X and Y coordinate locations at grid points (1,10) and (21,10) at the western and eastern boundaries respectively.
Chapter 5

Applications

The numerical model is applied to model tide induced circulation in Lake Pontchartrain and Bayou Chitigue. The two geometries are chosen so as to test the models capability to model medium and very small scale water bodies.

5.1 Tide induced circulation in Lake Pontchartrain

Lake Pontchartrain is located north of and adjacent to New Orleans, Louisiana. The two major natural outlets of the lake are The Rigolets, approximately 13 km long and Chef Menteur Pass, approximately 10 km long, near the east end of the lake. Lake Pontchartrain is about 40 km wide at its widest point and about 64 km long. The tides in Lake Pontchartrain and the surrounding areas are primarily diurnal with mean tide ranges of the order of 10 cm.

5.1.1 Data description

An intensive data gathering effort was undertaken by the Waterways Experiment Station of the U.S Army Corps of Engineers in 1979 for the purposes of devising a hurricane protection plan (Outlaw (1982)). A number of tide gauges were deployed in Lake Pontchartrain, the Rigolets and Chef Menteur Pass. The location of the gauges used in this study is shown in Fig. (5.1). A least squares fourier analysis was performed on the tide data in order to obtain the various tidal constituents. The different tidal constituents and their corresponding periods are
Table 5.1: Tidal Constituents

<table>
<thead>
<tr>
<th>Tidal Constituent</th>
<th>Symbol</th>
<th>Period (Hours)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Principal lunar diurnal</td>
<td>O1</td>
<td>25.82</td>
</tr>
<tr>
<td>Luni-solar diurnal</td>
<td>K1</td>
<td>23.94</td>
</tr>
<tr>
<td>Principal solar diurnal</td>
<td>P1</td>
<td>24.07</td>
</tr>
<tr>
<td>Larger lunar elliptic</td>
<td>Q1</td>
<td>26.87</td>
</tr>
<tr>
<td>Smaller lunar elliptic</td>
<td>M1</td>
<td>24.84</td>
</tr>
<tr>
<td>Small lunar elliptic</td>
<td>J1</td>
<td>23.10</td>
</tr>
<tr>
<td>Principal lunar</td>
<td>M2</td>
<td>12.42</td>
</tr>
<tr>
<td>Principal solar</td>
<td>S2</td>
<td>12.00</td>
</tr>
<tr>
<td>Larger lunar elliptic</td>
<td>N2</td>
<td>12.66</td>
</tr>
<tr>
<td>Lunar overtides</td>
<td>M4, M6, M8</td>
<td>6.21, 4.14, 3.11</td>
</tr>
</tbody>
</table>

Table 5.2: Average constituent amplitude of tidal elevations (feet)

<table>
<thead>
<tr>
<th>Tide Gage</th>
<th>O1</th>
<th>K1</th>
<th>P1</th>
<th>Q1</th>
<th>M1</th>
<th>J1</th>
<th>M2</th>
<th>S2</th>
<th>N2</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>0.27</td>
<td>0.27</td>
<td>0.09</td>
<td>0.05</td>
<td>0.02</td>
<td>0.02</td>
<td>0.05</td>
<td>0.03</td>
<td>0.01</td>
</tr>
<tr>
<td>B4</td>
<td>0.35</td>
<td>0.38</td>
<td>0.10</td>
<td>0.07</td>
<td>0.02</td>
<td>0.00</td>
<td>0.07</td>
<td>0.04</td>
<td>0.02</td>
</tr>
<tr>
<td>P3</td>
<td>0.13</td>
<td>0.15</td>
<td>0.05</td>
<td>0.03</td>
<td>0.01</td>
<td>0.01</td>
<td>0.02</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>P4</td>
<td>0.12</td>
<td>0.12</td>
<td>0.03</td>
<td>0.03</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>P5</td>
<td>0.12</td>
<td>0.12</td>
<td>0.03</td>
<td>0.03</td>
<td>0.01</td>
<td>0.00</td>
<td>0.00</td>
<td>0.01</td>
<td>0.00</td>
</tr>
<tr>
<td>P2</td>
<td>0.13</td>
<td>0.14</td>
<td>0.04</td>
<td>0.02</td>
<td>0.01</td>
<td>0.00</td>
<td>0.01</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>P7</td>
<td>0.11</td>
<td>0.12</td>
<td>0.02</td>
<td>0.03</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

shown in Table (5.1). The amplitude of the tidal constituents and their corresponding local epoch is shown in Tables (5.2) and (5.3). The water level elevations at the above locations are obtained by using equation (5.1)

\[ z(t) = \sum_{i=1}^{J} A_i \cos(\omega_i t + \phi_i) \]

(5.1)

where \( A_i \) is the amplitude of the \( i^{th} \) tidal constituent and \( \phi_i \) is the corresponding phase angle, \( \omega_i = \frac{2\pi}{T_i} \) and \( T_i \) is the corresponding time period.

The tide gages labeled R1 and B4 were used as boundary conditions at the mouths of The Rigolets and Chef Menteur Pass respectively. The gages P3, P4 and P5 were used for model verification.
Table 5.3: Local epoch (degrees) of each tidal constituent

<table>
<thead>
<tr>
<th>Tide Gage</th>
<th>O1</th>
<th>K1</th>
<th>P1</th>
<th>Q1</th>
<th>M1</th>
<th>J1</th>
<th>M2</th>
<th>S2</th>
<th>N2</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>48.4</td>
<td>18.2</td>
<td>4.2</td>
<td>344.2</td>
<td>162.7</td>
<td>23.1</td>
<td>126.3</td>
<td>86.7</td>
<td>135.5</td>
</tr>
<tr>
<td>B4</td>
<td>358.9</td>
<td>29.4</td>
<td>29.0</td>
<td>357.9</td>
<td>118.9</td>
<td>277.9</td>
<td>131.3</td>
<td>141.2</td>
<td>174.1</td>
</tr>
<tr>
<td>P3</td>
<td>63.0</td>
<td>91.3</td>
<td>61.3</td>
<td>23.8</td>
<td>90.7</td>
<td>13.6</td>
<td>202.9</td>
<td>231.5</td>
<td>203.7</td>
</tr>
<tr>
<td>P4</td>
<td>85.9</td>
<td>115.9</td>
<td>117.1</td>
<td>81.3</td>
<td>289.2</td>
<td>53.7</td>
<td>240.0</td>
<td>239.2</td>
<td>254.0</td>
</tr>
<tr>
<td>P5</td>
<td>90.9</td>
<td>112.7</td>
<td>93.5</td>
<td>82.2</td>
<td>278.3</td>
<td>189.7</td>
<td>262.7</td>
<td>265.1</td>
<td>290.4</td>
</tr>
<tr>
<td>P2</td>
<td>76.6</td>
<td>109.4</td>
<td>71.9</td>
<td>39.2</td>
<td>269.8</td>
<td>72.3</td>
<td>223.6</td>
<td>285.3</td>
<td>198.6</td>
</tr>
<tr>
<td>P7</td>
<td>86.8</td>
<td>124.5</td>
<td>152.0</td>
<td>80.6</td>
<td>275.3</td>
<td>309.3</td>
<td>248.4</td>
<td>230.6</td>
<td>233.2</td>
</tr>
</tbody>
</table>

Figure 5.1: Map of Lake Pontchartrain showing tide gage locations
Figure 5.2: Grid in physical domain for Lake Pontchartrain

Figure 5.3: The computational domain for Lake Pontchartrain
Figure 5.4: Input tidal elevations at the mouth of The Rigolets

Figure 5.5: Input tidal elevations at the mouth of Chef Menteur Pass
5.1.2 Implementation details

The numerical model described in chapter 3 is applied to model tidal circulations in Lake Pontchartrain. An initial grid was generated using the elliptic grid generator described in chapter 3. The initial grid in the physical domain is shown in Fig. (5.2a) and the corresponding computational space is shown in Fig. (5.3). Bathymetric information was linearly interpolated from navigation charts on to the model grid. An input water level elevation is prescribed at the mouths of The Rigolets and Chef Menteur Pass. Initial conditions for velocity and water level elevation were assumed to be equal to zero. The model was then run for a period of 685 hours in order to remove the influence of wrong initial conditions. The model was then run for a period of 28 days, with a time step of 300 seconds, corresponding to courant numbers ranging from 2 to 10. A manning's coefficient of 0.024 was used. The moving boundary condition was applied only to the lake and wall boundary conditions (zero flow) were applied everywhere else. This was done in order to reduce computation time. The model was run on a Workstation and took about 30 minutes of CPU time.
Figure 5.6: Observed and predicted tidal elevations at location P2
Figure 5.7: Observed and predicted tidal elevations at location P3
Figure 5.8: Observed and predicted tidal elevations at location P4
Figure 5.9: Observed and predicted tidal elevations at location P5
Figure 5.10: Observed and predicted tidal elevations at location P7
Figure 5.11: Influence of time step on water level elevation at location P4
Figure 5.12: Grid at the end of 22.5 Hrs of simulation time

Figure 5.13: Grid at the end of 656 Hrs of simulation time
Figure 5.14: Boundary locations at the end of 0 Hrs, 22.5 Hrs and 656 Hrs of simulation time

Figure 5.15: Velocity vectors at the end of 22.5 Hrs of simulation time
Figure 5.16: Velocity vectors at the end of 180 Hrs of simulation time

Figure 5.17: Velocity vectors at the end of 656 Hrs of simulation time
Figure 5.18: Tidal ellipses at four points traversing from north to south at mid bay
Figure 5.19: Movement of grid point (1,10) located on the western boundary as a function of time

Figure 5.20: Movement of grid point (7,17) located on the northern boundary as a function of time
Figure 5.21: Movement of grid point (7,1) located on the southern boundary as a function of time.

Figure 5.22: Effect of boundary slope on the movement of grid point (1,10) located on the western boundary as a function of time.
Figure 5.23: Effect of boundary slope on the movement of grid point (7,17) located on the northern boundary as a function of time.

Figure 5.24: Effect of boundary slope on the movement of grid point (7,1) located on the southern boundary as a function of time.
Figure 5.25: Water depth as a function of time at a point located 100 m from grid point (1,10) on the western boundary

Figure 5.26: Water depth as a function of time at a point located 200 m from grid point (1,10) on the western boundary

Figure 5.27: Water depth as a function of time at a point located 400 m from grid point (1,10) on the western boundary
5.1.3 Analysis of results

The input tidal elevations at the mouths of The Rigolets and Chef Menteur Pass are shown in figures (5.4) and (5.5). The comparison of observed and model calculated elevations for five locations inside the lake denoted by P2, P3, P4, P5 and P7 (see Fig. (5.17)), are shown in (5.6), (5.7), (5.8), (5.9) and (5.10), respectively. The model predictions were in close agreement with observed data.

In order to study the effect of time step on model predictions, the model was run with time steps of 150, 300 and 600 seconds. All other parameters were held constant. The effect of time step on water level elevation at location P4 is shown in Fig. (5.11). No noticeable error in amplitude or phase is observed, for time steps of 300 and 150 seconds, when compared with observed data. When the time step was increased to 600 seconds a small error in amplitude and a phase error of 3.3 hours is observed. This error is attributed to numerical diffusion introduced by the first order upwind scheme used to discretize the advection terms in the momentum equation.

The models capability of simulating continuously deforming boundaries is illustrated in figures (5.12), (5.13). Fig. (5.12) shows the grid at the end of 22.5 hours and Fig. (5.13) shows the grid at the end of 656 hours of simulation time. For clarity, the boundary locations at three time levels (0 hours, 22.5 hours and 656 hours) are shown in Fig. (5.14).

The depth averaged velocity vector plots at three time levels (22.5 hours, 180 hours and 656 hours) are shown in figures (5.15), (5.16) and (5.17). These indicate that the velocities inside Lake Pontchartrain are very small and are of the order of 5 cm/s. A similar observation was made by Outlaw (1982).
The tidal ellipses at mid-bay at four points, traversing from north to south are shown in Fig. (5.18). The tidal ellipses are oriented towards the northwest-southeast direction at the northern end of the lake. Traversing from north to south, the tidal ellipse moves in a counterclockwise direction and is oriented towards the southwest-northeast direction, at the southern boundary.

The movement of a grid point located on the western, northern and southern boundaries is shown in figures (5.19) to (5.21). The maximum tidal expanse calculated by the model is about 2000 m. Such large tidal expanses can be attributed to the presence of very mild boundary slopes. Due to the unavailability of accurate topographic information, a slope of about 0.001 was assumed. Since the boundary movement is dependent on the boundary slope, the extent of flooding can be controlled varying the boundary slope.

In order to study the effect of boundary slope on boundary movement two simulations were performed, by keeping all the input variables identical, except the boundary slope. The boundary slope was made steeper by raising the water depths at grid points adjacent to land. For the mild slope case, the water depth along the boundaries was assumed to be 0.3 m and for the steep slope case a depth of 0.6 m was chosen. The effect of boundary slope on boundary movement at a western, northern and southern boundary point is shown in figures (5.22), (5.23) and (5.24). As seen from the figures, the boundary movement decreases for steeper slopes and is almost half of the distances obtained for the case of a mild slope.

Using the model, it is possible to obtain the temporal history of flooding at a point, located at specified distances from the boundary. Figures (5.25), (5.26) and (5.27) show the time variation of water depth at points located 100 m, 200 m and 400 m from grid point (1, 10) on the western boundary. Using this information
it is possible to calculate the inundation duration, which is defined as the total
time a point located at a distance from the boundary experiences flooding. The
inundation duration for the above points was calculated to be 48.8%, 36.33% and
19.68% for a point located at a distance of 100 m, 200 m and 400 m of the western
boundary. Such information may be useful for evaluating vegetation degradation
due to prolonged exposure to flooding.

5.2 Tide induced circulation in Bayou Chitigue

In order to demonstrate the models capabilities of simulating flow patterns in
very small physical systems, the model was applied to the Bayou Chitigue marsh
pond system. Bayou Chitigue is located in the salt marshes of coastal Louisiana
in the Terrebonne basin. The vegetation is dominated by *Spartina alterniflora* The
area is sediment poor, due to its remote location to a sediment source and as a result
the marsh is deteriorating rapidly. The area is interlaced with numerous shallow
canals and the marsh is hummocky with considerable exposed mud within the marsh.
The substrate is dominated by mineral sediment with very low bulk densities. A
permanent channel connects the pond to the bayou.

Field data were gathered for a period of six days from 12/10/90 to 12/16/90
at three locations. Water level measurements were made in the bayou and on
the surface of the marsh. An instrument package consisting of an electromagnetic
current meter and a pressure sensor were placed at the mouth of the channel leading
to the pond. A map showing the location of the gages is shown in Fig. (5.28). The
field data was then converted into engineering units and used without resorting to
any filtering.
5.2.1 Implementation details

The numerical model described in chapter 3 is applied to model tidal circulations in Bayou Chitigue. An initial grid was generated using the elliptic grid generator described in chapter 3. The initial grid in the physical domain is shown in Fig. (5.29) and the corresponding computational space is shown in Fig. (5.30). The first three days of field data was used for model calibration and the next three days for verification. Accurate bathymetric information was not available so reasonable values of depth based on spot measurements were used. A depth of 1 m was assumed in the bayou and a depth of 0.3 m was assumed in the channel and the pond. A value of 0.001 was used for the marsh slope. Water level prescribed at the two ends of the Bayou is used as boundary condition. The model results are compared with the field data at locations G1 and G2. Initial conditions for velocity and water level elevation were assumed to be equal to zero.
Figure 5.29: Grid in physical domain for Bayou Chitigue

Figure 5.30: The computational domain for Bayou Chitigue
Figure 5.31: Input tidal elevations in the bayou (calibration)
Figure 5.32: Comparison of predicted and observed u-velocity at the mouth of the pond at location G1 (calibration)
Figure 5.33: Comparison of predicted and observed v-velocity at the mouth of the pond at location G1 (calibration)
Figure 5.34: Comparison of predicted and observed water level elevations at the mouth of the pond at location G1 (calibration)
Figure 5.35: Comparison of predicted and observed water level elevations on the surface of the marsh at location G2 (calibration)
5.2.1.1 Calibration

The tidal elevations used as boundary conditions at either end of the bayou are shown in Fig. (5.31). The model was calibrated by varying the value of manning’s roughness coefficient, and the results compared with field data. A manning’s roughness coefficient of 0.035 was found to be satisfactory for the pond and the channels and a value of 0.2 for the marsh surface.

The model was run for a period of 3 days, with a time step of 30 seconds, corresponding to courant numbers ranging from 20 to 40. The moving boundary condition was applied only to the marsh pond and wall boundary conditions (zero flow) were applied everywhere else. This was done in order to reduce computation time. The model was run on a Workstation and took about 60 minutes of CPU time.

The comparison of observed and predicted u and v velocities at location G1 is shown in figures (5.32) and (5.33). As can be seen from the figures the model predicted velocities are in reasonable agreement with the field data. A comparison of the water levels at location G1 shown in Fig. (5.34) are also in good agreement. The moving boundary part of the model was calibrated by comparing observed water levels on the marsh surface at location G2 with model predicted values as shown in Fig. (5.35).
Figure 5.36: Input tidal elevations in the bayou (verification)
Figure 5.37: Comparison of predicted and observed u-velocity at the mouth of the pond at location G1 (verification)
Figure 5.38: Comparison of predicted and observed v-velocity at the mouth of the pond at location G1 (verification)
Figure 5.39: Comparison of predicted and observed water level elevations at the mouth of the pond at location G1 (verification)
Figure 5.40: Comparison of predicted and observed water level elevations on the surface of the marsh at location G2 (verification)
5.2.1.2 Verification

The model was run for a period of three days and compared with the leftover field data, without changing the parameters obtained during calibration. The water levels used as boundary condition at either ends of the bayou is shown in Fig. (5.36). The comparison between observed and predicted u and v velocities at location G1 is shown in figures (5.37) and (5.38). As can be seen from the figures the model predicted velocities continue to be in reasonable agreement with the field data. A comparison of the water levels at location G1 shown in Fig. (5.39) are also in good agreement. The observed water levels on the marsh surface at location G2 compare well with model predicted values as shown in Fig. (5.40), thus verifying the validity of the moving boundary aspect of the model.
Figure 5.42: Grid at the end of 96.0 hours of simulation time

Figure 5.43: Marsh pond boundary locations at the end of 74 and 96 hours of simulation time
Figure 5.44: Velocity vectors at the end of 74 hours of simulation time

Figure 5.45: Velocity vectors at the end of 96 hours of simulation time
5.2.2 Analysis of results

The model predicted velocities were smooth compared to the observed data. It should be noted that no filtering was applied to the field data, which has the high frequency component in it. The numerical model was able to predict the broad structure of the signal and as expected the high frequency components were not reproduced.

Figures (5.41) and (5.42) show the grid at the end of 74 hours and 96 hours of simulation time. For clarity, the boundary locations at two time levels (74 hours and 96 hours) are shown in Fig. (5.43). The smaller boundary is at the end of 74 hours and the larger boundary is at the end of 96 hours of simulation time. The depth averaged velocity vectors at 74 and 96 hours is shown in figures (5.44) and (5.45). The inundation duration calculated at location G2 was 89.8% for the observed data and 87.1% for the predicted data, during the calibration run. Corresponding values were 91.3% and 80.6% for the verification run.
Chapter 6

Conclusions

The following conclusions can be drawn from the study.

1. It is possible to model physical systems of varying sizes from medium scale systems like Lake Pontchartrain to extremely small scale systems such as the Bayou Chitigue channel pond system with moderate expenditure of computational time.

2. The boundary fitted coordinate method is especially attractive from the viewpoint of modeling small systems as errors due to improper application of boundary conditions is minimized.

3. The model is able to tolerate large variations in boundary movement, without resulting in instabilities.

4. The numerical scheme is stable and good phase resolution was obtained when compared with analytical solutions. When applied to Bayou Chitigue there was a phase error of about 1 hour. This is attributed to lack of good bathymetric information. It is also possible to reduce phase error by choosing a very small time step. The model was able to predict water levels on the marsh quite well thus proving the validity of the moving boundary formulation.
5. Although the model is unconditionally stable, very large time steps result in lack of good phase resolution. This is due in partly to the first order upwind formulation used to discretize the advection terms in the momentum equation.

6. The model was able to predict the tidal signal inside Lake Pontchartrain quite accurately.

7. The boundary movement is governed by the boundary slope, with large movements associated with mild slopes.

8. A drawback of studying very small systems is that inaccuracies or unavailability of good data results in poor model performance.

6.1 Scope for future work

1. Explore the possibility of formulating higher order upwind schemes with low numerical diffusion.

2. Explore the possibility of using high resolution satellite imagery to obtain land water interfaces as a function of time, and hence use it to calibrate the model.

3. The solution algorithm for solving the linear systems of equations, arising from the semi implicit formulation are currently solved using a successive over relaxation method. The model can be made more efficient by using multigrid methods to solve the linear system.

4. Explore the possibility of extending the model to three dimensions.
References


BUTLER, H.L. (1978c) Coastal flood simulation in stretched coordinates. 16 th Intl. Conf. on Coastal Engg.


Vita

Vibhas Aravamuthan was born in Nagpur, India on May 23, 1962. He received his B.Tech. degree in civil engineering from the Indian Institute of Technology, Madras, India in May 1985. He joined the department of Civil Engineering, for a Doctoral program in August 1988.
Candidate: Vibhas Aravamuthan

Major Field: Civil Engineering

Title of Dissertation: A Two Dimensional Vertically Integrated Moving Boundary Hydrodynamic Model in Curvilinear Coordinates

EXAMINING COMMITTEE:

Date of Examination: December 12, 1994