COMPUTER CODE DEVELOPMENT FOR NUMERICAL SOLUTION OF DEPTH INTEGRATED SHALLOW WATER EQUATIONS TO STUDY FLOOD WAVES

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

BEHİYE NİLAY İŞCEN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
CIVIL ENGINEERING

FEBRUARY 2015

Approval of the thesis:

COMPUTER CODE DEVELOPMENT FOR NUMERICAL SOLUTION OF DEPTH INTEGRATED SHALLOW WATER EQUATIONS TO STUDY FLOOD WAVES

submitted by **BEHİYE NİLAY İŞCEN** in partial fulfillment of the requirements for the degree of **Master of Science in Civil Engineering Department, Middle East Technical University** by,

Prof. Dr. Gülbin Dural Ünver
Dean, Graduate School of **Natural and Applied Sciences** ____________________

Prof. Dr. Ahmet Cevdet Yalçıner
Head of Department, **Civil Engineering** ____________________

Prof. Dr. İsmail Aydın
Supervisor, **Civil Engineering Department, METU** ____________________

Examining Committee Members:

Prof. Dr. Nuray Tokyay
Civil Engineering Department, METU ____________________

Prof. Dr. İsmail Aydın
Civil Engineering Department, METU ____________________

Prof. Dr. A. Burcu Altan Sakarya
Civil Engineering Department, METU ____________________

Assoc. Prof. Dr. Mete Köken
Civil Engineering Department, METU ____________________

Assist. Prof. Dr. Nuray Öktem
Mathematics Department, ÇOMU ____________________

Date: **02.02.2015**

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: Behiye Nilay İŞCEN

Signature :

ABSTRACT

COMPUTER CODE DEVELOPMENT FOR NUMERICAL SOLUTION OF DEPTH INTEGRATED SHALLOW WATER EQUATIONS TO STUDY FLOOD WAVES

İşcen, Behiye Nilay

M.S., Department of Civil Engineering
Supervisor: Prof. Dr. İsmail Aydın

February 2015, 77 pages

Floods are the most common natural risks to human beings because the most populated areas in the world are vulnerable to flood disasters. Floods are likely to become increasingly severe and more frequent due to climate change, population growth, change of land use, irrigation, deforestation and urban development inside the flood plains. Inundation risk assessment primarily requires numerical solution to a mathematical model, which appropriately describes hydraulics of flood waves over terrains including natural river beds to highly populated urban areas. This thesis aims the development and the validation of a computer code to solve the depth integrated shallow water equations for flow around a rectangular obstacle in a prismatic channel. A high-resolution shock capturing solution algorithm is implemented to investigate all possible flow cases. Boundary conditions for various flow configurations are considered. Conservativeness, grid adaptivity and computational stability are investigated features of the code developed.

Keywords: Shallow Water Equations, Depth-Averaged Equations, Riemann Solver, Flood Waves

ÖZ

TAŞKIN DALGALARINI ÇALIŞMAK AMACIYLA DERİNLİK ENTEGRELİ SIĞ SU DENKLEMLERİNİN NÜMERİK ÇÖZÜMÜ İÇİN BİLGİSAYAR KODU GELİŞTİRİLMESİ

İşcen, Behiye Nilay

Yüksek Lisans, İnşaat Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. İsmail Aydın

Şubat 2015, 77 sayfa

Dünyada nüfusun en kalabalık olduğu bölglerin sel felaketlerine karşı savunmasız yerler olması nedeniyle sel baskınları en çok karşılaşılan doğal afetlerdir. İklim değişikliği, nüfus artışı, arazi kullanımında değişim, sulama, ormansızlaşma ve taşkın havzalarındaki kentsel yerleşim, sel felaketlerinin giderek daha şiddetli ve daha sık görülmesine sebep olmaktadır. Sel risk değerlendirmesi, öncelikli olarak, taşkın dalgalarının doğal nehir yataklarından, nüfus yoğunluğu fazla olan kentsel alanlara kadar olan arazi üzerinde hidrolik davranışını tanımlayan matematiksel bir modelin sayısal çözümüne ihtiyaç duyar. Bu tez, prizmatik kanal içerisindeki dikdörtgen biçimli bir engelin etrafında gelişen akım için derinlik entegreli sığ su denklemlerini çözmek amaçlı bir bilgisayar kodu geliştirilmesi ve bunun doğrulamasını amaçlamaktadır. Olası tüm akım durumlarını araştırmak için, yüksek çözünürlüklü şok yakalayan bir algoritma uygulanmaktadır. Çeşitli akım konfigürasyonları için sınır koşulları dikkate alınmaktadır. Geliştirilen programın korunumu sağlaması, çözüm ağı adaptasyonu ve hesaplama kararlılığı incelenmektedir.

Anahtar Kelimeler: Sığ Su Denklemleri, Derinlik Entegreli Denklemler, Riemann Çözücü, Taşkın Dalgası

"All that is gold does not glitter,
Not all those who wander are lost;
The old that is strong does not wither,
Deep roots are not reached by the frost."

— J.R.R. Tolkien

ACKNOWLEDGEMENTS

Foremost, I would like to express my sincere gratitude to *Prof. Dr. İsmail Aydın* for his endless guidance, tremendous support and encouragement. His continuous dedication to the academic life, teaching and research is admirable and inspirational for me. I feel lucky and honored to have a chance to work with him and learn from him. My appreciation for his confidence in me to make it possible to complete this work is immeasurable.

I also wish to thank *Asst. Prof. Dr. Nuray Öktem* for always being so kind, helpful and motivating. Without her excellent guidance, knowledge and endless energy, it would be impossible to complete this work. I deeply appreciate for her motivation and generosity.

I also would like to thank *Prof. Dr. Burcu Altan Sakarya*, my undergraduate professor who is an inspiration for me to follow an academic career in the field of hydromechanics.

The burden of writing this thesis was lessened significantly by the support of my roommate *Ezgi Köker*. I would like to express my gratitude to her for helping me get through the difficult times and for her unfailing emotional support.

I extend my heartfelt thanks to my friend and my colleague *Betül Aytöre* with whom I have shared moments of deep anxiety but also of big excitement. I am grateful for her presence and caring she provided.

I deeply thank my friends *Asuman Aybey*, *Gülay Oskay Ünay* and *Halil Ünay* for making me feel cared and loved. Their presence is very important. We share our times, our passion and our tea with all our hearts.

I am grateful for the encouragement of my childhood friend *Benal Süslü*. Thank you for taking such good care of me for all those years. I am grateful to have such a sister.

I thank with love to *Çağdaş Bilici*, my best friend and the most excellent boyfriend. None of this would have been possible without his love and encouragement. He has supported me with all his strength not only throughout this graduate experience, but also throughout my years I have spent with him.

I am forever indebted to my parents and my sister for their unconditional trust and endless support. It was their generously given love and encouragement that made me who I am now. To them I dedicate this thesis.

TABLE OF CONTENTS

ABSTRACT v

ÖZ vi

ACKNOWLEDGEMENTS viii

TABLE OF CONTENTS x

LIST OF FIGURES xii

LIST OF TABLES xiv

LIST OF SYMBOLS & ABBREVIATIONS xv

INTRODUCTION 1

1.1. Introduction and General Description of the Problem 1

1.2. Objectives of the Study 3

DERIVATION OF GOVERNING EQUATIONS 5

2.1. Depth-Averaged Shallow Water Equations 5

2.1.1. Derivation of the Governing Equations 7

2.1.2. Conservation of Mass 8

2.1.3. Conservation of Momentum 9

2.1.4. Further Assumptions and Boundary Conditions 12

2.2. Numerical Schemes for 2D Depth-Averaged Shallow Water Equations 19

2.2.1. Finite Volume Methods 19

2.2.2. Upwind Schemes 19

2.2.3. Riemann Problem 20

2.2.4. Godunov’s Method 22

2.2.5. Approximate Riemann Solvers 24

2.2.6. Roe’s Approximate Riemann Solver 24

2.2.7. Higher-Order and High-Resolution (Total Variation Diminishing) Schemes 26
2.2.8. MUSCL-type High-Order Schemes..... 26
2.2.9. Slope Limiter Approach..... 28
2.2.10. Boundary Conditions 30
2.2.11. Wall Boundary Condition 30
2.2.12. Symmetry Boundary Condition 30
2.2.13. Inflow/Outflow Boundary Condition..... 30
2.2.14. Periodic Boundary Condition 31
NUMERICAL SOLUTION METHOD ADOPTED TO PRESENT PROBLEM..... 33
3.1. Equations and their characteristics 33
3.2. 2D Computational Domain and Numerical Scheme 35
3.3. Stability Criteria and Boundary Conditions 44
RESULTS AND DISCUSSIONS 47
4.1. 1D Test cases..... 47
4.2. 2D Solutions 50
4.1.1. Supercritical Flow Test Case 51
4.1.2. Subcritical Flow Test Case 62
4.1.3. 2D Problem with Periodic Boundary Conditions 70
CONCLUSIONS AND RECOMMENDATIONS 73
REFERENCES..... 75

LIST OF FIGURES

Figure 2.1.Vertical plane of integration of RANS 6
Figure 2.2. Depth-averaging variables 12
Figure 2.3. Structure of the solution of the Riemann problem (Toro, 2009) 20
Figure 2.4. Definition sketch for dam-break problem (Guinot, 2010) 22
Figure 2.5. Godunov's scheme and the local Riemann problem 23
Figure 2.6. Piece-wise linear MUSCL reconstruction of data in a single cell (i), with the boundary extrapolated values u_L and u_R (Toro, 2009) 27
Figure 2.7. Piece-wise linear MUSCL representation (Hirsch, 1990) 28
Figure 2.8. Limiter functions in φ,r diagram (Versteeg & Malalasekera, 2007) 29
Figure 3.1. Computational domain for the flow past horizontally aligned and evenly spaced square blocks 35
Figure 3.2. Collocated grid arrangement of the velocity components and the water depth 36
Figure 3.3. Linear one-sided extrapolation of interface values for k= -1 (Hirsch, 1990).. 38
Figure 4.1. 1D test case for the analytical and numerical solutions of dam-break problem for subcritical state where HL=10 m and HR=5 m obtained at t=50sec using Roe scheme 48
Figure 4.2. 1D test case for the analytical and numerical solutions of dam-break problem for supercritical state where HL=10 m and HR=0.1 m obtained at t=50sec using Roe scheme.. 49
Figure 4.3. Channel dimensions for all test cases 50
Figure 4.4. Supercritical surface profile (IT=8000 & mesh size=0.02 m) 53
Figure 4.5. 2D Flow field (Supercritical, IT=8000 & mesh size=0.02 m)................. 54
Figure 4.6. 2D Vector field (Supercritical, IT=8000 & mesh size=0.02 m) 55
Figure 4.7. Supercritical surface profile (IT=12000 & mesh size=0.02 m) 57
Figure 4.8. Supercritical surface profile (IT=20000 & mesh size=0.02 m) 58

Figure 4.9. Error of supercritical flow case (IT=20000 & mesh size=0.02 m).......... 59
Figure 4.10. The change in discharge with the # of iterations (supercritical flow case & mesh size=0.02m) 61
Figure 4.11. Subcritical surface profile (IT=20000 & mesh size=0.02 m)................ 64
Figure 4.12. Error of subcritical flow case (IT=20000 & mesh size=0.02 m)........... 65
Figure 4.13. The change in discharge with the # of iterations (subcritical flow case & mesh size=0.02m) 66
Figure 4.14. Subcritical (choked) surface profile (IT=20000 & mesh size=0.02 m). 68
Figure 4.15. Error of subcritical choked flow case (IT=20000 & mesh size=0.02 m) 69
Figure 4.16. Channel with periodic arrangement of the blocks 70

LIST OF TABLES

Table 4.1. Supercritical flow data 51
Table 4.2. Subcritical flow data 62
Table 4.3. Subcritical choked flow data 67

LIST OF SYMBOLS & ABBREVIATIONS

A	Jacobian matrix of flux E
B	Jacobian matrix of flux G
E_C, G_C	convective fluxes in x and ydirections, respectively
E_D, G_D	diffusive fluxes in x and y directions, respectively
E	flux vector in x direction
F	total flux vector
G	flux vector in y direction
H_0	uniform flow depth
S	source vector
$S_{0,x}, S_{0,y}$	bottom slope in x and y directions
$S_{f,x}, S_{f,y}$	friction slope in x and y directions
U	vector of conserved variables
U_0	unifor flow velocity
$\tilde{a}^1, \tilde{a}^2, \tilde{a}^3$	eigenvalues of matrix A
$\tilde{b}^1, \tilde{b}^2, \tilde{b}^3$	eigenvalues of matrix B
b	channel bottom elevation
c	celerity
c_f	friction coefficient
$\tilde{e}^1, \tilde{e}^2, \tilde{e}^3$	eigenvectors of matrix A
$\tilde{f}^1, \tilde{f}^2, \tilde{f}^3$	eigenvectors matrix B
g	gravitational acceleration

h	water depth
n	Manning's coefficient
q_x	unit discharge in x direction
q_y	unit discharge in y direction
s	water surface elevation
t	time
u	velocity in x direction
$\tilde{u}, \tilde{v}, \tilde{c}$	Roe averages
$\bar{u}, \bar{v}, \bar{w}, \bar{p}$	time averaged values of flow variables
v	velocity in y direction
$\tilde{\alpha}^k$	wave strengths in x direction
$\tilde{\beta}^k$	wave strengths in y direction
$\Delta x, \Delta y$	step sizes in x and y directions, respectively
Δt	time step
ρ	density of fluid
$\tau_{x,b}, \tau_{y,b}$	bottom friction in x and y directions
$\tau_{x,s}, \tau_{y,s}$	surface friction in x and y directions
τ_{ij}	combined viscous and turbulent stresses

CFL	Courant-Friedrichs-Lewy condition
$MUSCL$	Monotonic Upstream-Centered Scheme for Conservation Laws
RP	Riemann Problem
TVD	Total Variation Diminishing

CHAPTER 1

INTRODUCTION

1.1. Introduction and General Description of the Problem

Most of the environmental problems and the related physical phenomena are the subjects of mathematical models based on depth-averaged equations. A variety of these mathematical models use the depth-averaged shallow water equations to understand and solve the gravity induced water flows. Sea currents, tides in oceans, flood waves in rivers, atmospheric flows, channel flows, wave breaking are a few examples of a wide variety of physical phenomenon modeled with shallow water equations.

Fundamental physical laws, which base the continuum hypothesis, have an undeniable importance in hydraulics. Using the continuum hypothesis, the governing equations are found by applying mass and momentum conservation principles to a control volume that continuum hypothesis agrees. Governing equations of free surface flows are defined with three dimensional Navier-Stokes equations with the assumption of Newtonian, viscous and incompressible fluid. They form a hyperbolic system of nonlinear conservation laws. Due to their complexity, they have no analytical solution. For the computational fluid flow studies, it is also difficult and expensive to implement Navier-Stokes equations numerically in free surface flow problems. Additionally, working in three dimensional approach requires complicated discretization methods and meshing. Therefore, three dimensional Navier-Stokes equations are simplified to two dimensional depth-averaged shallow water equations with the assumption that horizontal domain is noticeably larger than the vertical domain of the problem.

Starting from the 19th century, the shallow water equations started to gain importance and become an indispensable tool for modeling. Although these equations can basically describe a simplified version of complex environmental problems, they can still cover most of the essential characteristics, which can affect fluid motion in open channels. Despite its considerable simplicity compared to three dimensional Navier-Stokes equations, the two dimensional depth-averaged shallow water equations have no analytical solution either. They are solved by approximate methods. Analytical solutions are available for only a few one-dimensional cases.

The solution of shallow water equations are challenging because they include discontinuous solutions even though the initial data is smooth. These discontinuities like shock waves (surges and trans-critical flows in channels) result in undesired failings of numerical methods. Therefore, shock-capturing methods are utilized to solve two dimensional shallow water equations in flow fields involving discontinuities in the flow variables.

In this study finite volume method is utilized to discretize the governing equations since the finite volume approach is designed based on conservation of mass, momentum and energy, and higher order and shock capturing schemes are easily implemented to finite volume based numerical algorithms. For the numerical flux calculation, approximate Riemann solver of Roe (Roe, 1981) is utilized which is a Godunov type upwind method. Since the method is first order accurate in space and time, accuracy of the scheme is increased from first order to second order by applying Monotonic Upstream-Centered Scheme for Conservation Laws (MUSCL) approach (van Leer, 1979) and the monotonicity of the scheme is ensured by using slope limiting functions. Finally, the second order, high resolution Godunov type finite volume scheme is obtained to utilize for the solution of the two dimensional depth-averaged shallow water equations.

1.2. Objectives of the Study

The aim of this study is to develop a computer code that can solve the two dimensional depth-averaged shallow water equations governing the free surface fluid flow with discontinuous flow variables such as water depth. Flow around a rectangular block located in a prismatic channel is considered as a special test case to validate the code. Capabilities of the code are presented examining various flow configurations with different types of boundary conditions. Computational features such as stability, conservativeness, grid resolution requirements and efficiency of adopted algorithms are also illustrated.

First chapter introduces the thesis and its objectives. In Chapter 2, governing equations of the modeled fluid flow are presented, and the numerical schemes available in literature for the solution of these equations are reviewed. In Chapter 3, the numerical solution method adopted to the present problem is described. Additionally, the discretization procedure and the implementation of the boundary conditions are explained, and the numerical error definition is given. Chapter 3 mainly describes the numerical method adopted to the present problem. The discretization procedure is given in detail and the chapter is completed with a brief discussion of the stability criteria and the implementation of boundary conditions. In Chapter 4, results of the one-dimensional numerical model is compared with the analytical solution available in literature to validate the shock-capturing ability of the numerical scheme used. Then, the results of the two dimensional numerical model are presented comparatively for different flow states. Convergence and the numerical accuracy of the problem are discussed. Finally, the thesis is completed with the important remarks and the recommendations for further studies.

CHAPTER 2

DERIVATION OF GOVERNING EQUATIONS

2.1. Depth-Averaged Shallow Water Equations

Three dimensional Reynolds Averaged Navier-Stokes (RANS) equations (Versteeg & Malalasekera, 2007) govern the turbulent movement of the incompressible fluid, and are written as,

$$\frac{\partial \bar{u}}{\partial t}+\bar{u}\frac{\partial \bar{u}}{\partial x}+\bar{v}\frac{\partial \bar{u}}{\partial y}+\bar{w}\frac{\partial \bar{u}}{\partial z}=-\frac{1}{\rho}\frac{\partial \bar{p}}{\partial x}+\frac{1}{\rho}\left(\frac{\partial \tau_{xx}}{\partial x}+\frac{\partial \tau_{yx}}{\partial y}+\frac{\partial \tau_{zx}}{\partial z}\right) \tag{2.1}$$

$$\frac{\partial \bar{v}}{\partial t}+\bar{u}\frac{\partial \bar{v}}{\partial x}+\bar{v}\frac{\partial \bar{v}}{\partial y}+\bar{w}\frac{\partial \bar{v}}{\partial z}=-\frac{1}{\rho}\frac{\partial \bar{p}}{\partial y}+\frac{1}{\rho}\left(\frac{\partial \tau_{xy}}{\partial x}+\frac{\partial \tau_{yy}}{\partial y}+\frac{\partial \tau_{zy}}{\partial z}\right) \tag{2.2}$$

$$\frac{\partial \bar{w}}{\partial t}+\bar{u}\frac{\partial \bar{w}}{\partial x}+\bar{v}\frac{\partial \bar{w}}{\partial y}+\bar{w}\frac{\partial \bar{w}}{\partial z}=-g-\frac{1}{\rho}\frac{\partial \bar{p}}{\partial z}+\frac{1}{\rho}\left(\frac{\partial \tau_{xz}}{\partial x}+\frac{\partial \tau_{yz}}{\partial y}+\frac{\partial \tau_{zz}}{\partial z}\right) \tag{2.3}$$

$$\frac{\partial \bar{u}}{\partial x}+\frac{\partial \bar{v}}{\partial y}+\frac{\partial \bar{w}}{\partial z}=0 \tag{2.4}$$

In the above equations, u, v and w are the velocity components in x, y and z directions, respectively, p is the pressure and over bar indicates time averaged point values of the flow variables. The stress terms representing combined viscous and turbulent stresses are defined by τ_{ij}. g is the gravitational acceleration and ρ is the fluid density. Numerical solution of 3D RANS for large-scale free surface flow problems may become very expensive when advanced turbulence models are used. Another important issue is the determination of location of the free surface which is a difficult task introducing additional complexities in terms of turbulence modelling and therefore requiring extra computation time.

In many hydraulic engineering problems the flow depth is small compared to horizontal extent and variation of flow quantities over the vertical extent has less significance in the analysis and design processes. Thus, determination of water depth, bottom friction and horizontal components of depth-averaged velocity may be sufficient for many engineering purposes. Such a simplified solution may be obtained from depth-averaged equations valid for shallow flows. The two dimensional depth-averaged flow equations are obtained by averaging three dimensional RANS equations over the flow depth. A definition sketch showing integration domain in a vertical plane is shown in Figure 2.1 where b defines the bottom elevation, h is the water depth and s represents the surface elevation.

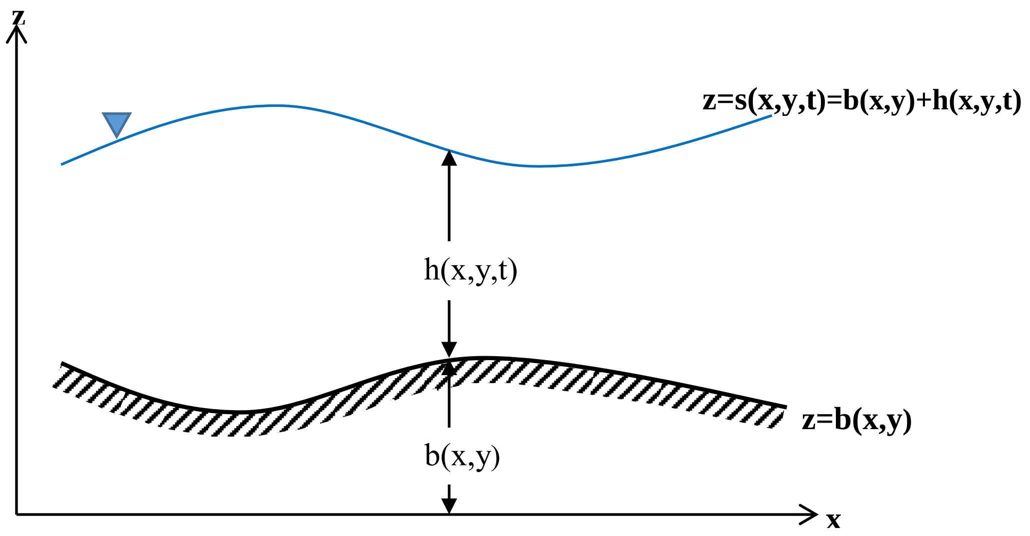

Figure 2.1.Vertical plane of integration of RANS

2.1.1. Derivation of the Governing Equations

Four basic steps are followed to complete derivation of the equations. Firstly, the hydrostatic balance relation is obtained. Since horizontal length scale is assumed much larger than the vertical length scale, many terms can be neglected in momentum equation in z-direction leading an ordinary differential equation for the pressure.

$$0 = -g - \frac{1}{\rho}\frac{\partial \bar{p}}{\partial z} \tag{2.5}$$

Integration over the vertical axis from the bed to the free surface gives the hydrostatic pressure distribution

$$\frac{\partial \bar{p}}{\partial z} = -\rho g \tag{2.6}$$

$$\partial \bar{p} = -\rho g \partial z \tag{2.7}$$

$$\int_{p_s}^{\bar{p}} \partial \bar{p} = -\int_{z}^{s} \rho g \partial z \tag{2.8}$$

$$\bar{p} - p_s = -(\rho g z - \rho g s) \tag{2.9}$$

$$\bar{p} = p_s + \rho g s - \rho g z \tag{2.10}$$

where p_s is the pressure on the free surface, s is the elevation of the free surface which can be a function of horizontal coordinates x and y and time, t.

For the integration of continuity equation (2.4) and horizontal components of the momentum equations (2.1) and (2.2) the Leibniz integral rule (Flanders, 1973) given below is utilized.

$$\int_{A(x,y)}^{B(x,y,t)} \frac{\partial f}{\partial x} dz = \frac{\partial}{\partial x}\int_{A}^{B} f\, dz - f_{z=B}\frac{\partial B}{\partial x} + f_{z=A}\frac{\partial A}{\partial x} \tag{2.11}$$

2.1.2. Conservation of Mass

Consider the Reynolds averaged continuity equation (2.4),

$$\frac{\partial \bar{u}}{\partial x}+\frac{\partial \bar{v}}{\partial y}+\frac{\partial \bar{w}}{\partial z}=0 \tag{2.12}$$

which can be vertically integrated from bottom to the free surface as

$$\int_b^s \frac{\partial \bar{u}}{\partial x} dz+\int_b^s \frac{\partial \bar{v}}{\partial y} dz+\int_b^s \frac{\partial \bar{w}}{\partial z} dz=0 \tag{2.13}$$

Now, by using Leibniz integral rule (Equation 2.11), Equation 2.13 can be rewritten as

$$\left(\frac{\partial}{\partial x}\int_b^s \bar{u}\, dz-\bar{u}_s\frac{\partial s}{\partial x}+\bar{u}_b\frac{\partial b}{\partial x}\right)+\left(\frac{\partial}{\partial y}\int_b^s \bar{v}\, dz-\bar{v}_s\frac{\partial s}{\partial y}+\bar{v}_b\frac{\partial b}{\partial y}\right)+(\bar{w}_s-\bar{w}_b) = 0 \tag{2.14}$$

Kinematic boundary conditions (Toro, 2001) are applied

$$\bar{w}_s=\frac{Ds}{Dt}=\frac{\partial s}{\partial t}+\bar{u}_s\frac{\partial s}{\partial x}+\bar{v}_s\frac{\partial s}{\partial y} \tag{2.15}$$

$$\bar{w}_b=\frac{Db}{Dt}=\frac{\partial b}{\partial t}+\bar{u}_b\frac{\partial b}{\partial x}+\bar{v}_b\frac{\partial b}{\partial y} \tag{2.16}$$

Then, Equation (2.14) is rearranged such that

$$\frac{\partial}{\partial x}\int_b^s \bar{u}\, dz+\frac{\partial}{\partial y}\int_b^s \bar{v}\, dz-\left(\bar{u}_s\frac{\partial s}{\partial x}+\bar{v}_s\frac{\partial s}{\partial y}-\bar{w}_s\right)+\left(\bar{u}_b\frac{\partial b}{\partial x}+\bar{v}_b\frac{\partial b}{\partial y}-\bar{w}_b\right) = 0 \tag{2.17}$$

In Equation (2.17), the first integral term $\int_b^s \bar{u}\, dz$ represents the flux, q_x. Similarly, the second integral term $\int_b^s \bar{v}\, dz$ is the flux, q_y, in y-direction. The third term expresses the temporal change of the water surface elevation which equals to $-\frac{\partial s}{\partial t}$. The last term can also be written as temporal change of bed elevation, $-\frac{\partial b}{\partial t}$ which is equal to zero since the bed boundary is assumed to be fixed.

Therefore, Equation (2.17) is reduced to

$$\frac{\partial s}{\partial t} + \frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} = 0 \tag{2.18}$$

Also, the change of water surface is decomposed such that

$$\frac{\partial s}{\partial t} = \frac{\partial (b+h)}{\partial t} = \frac{\partial b}{\partial t} + \frac{\partial h}{\partial t} = \frac{\partial h}{\partial t} \tag{2.19}$$

Finally, the following depth-averaged continuity equation is achieved

$$\frac{\partial h}{\partial t} + \frac{\partial (h\tilde{u})}{\partial x} + \frac{\partial (h\tilde{v})}{\partial y} = 0 \tag{2.20}$$

where $\tilde{u}$ and $\tilde{v}$ are vertically averaged horizontal velocity components.

2.1.3. Conservation of Momentum

In the momentum equation in x-direction (Equation 2.1) firstly, the time derivative and the advection terms are vertically integrated

$$I = \int_b^s \frac{\partial \bar{u}}{\partial t} dz + \int_b^s \frac{\partial \bar{u}^2}{\partial x} dz + \int_b^s \frac{\partial \overline{uv}}{\partial y} dz + \int_b^s \frac{\partial \overline{uw}}{\partial z} dz \tag{2.21}$$

Again, Leibniz integral rule is applied to each integral term in Equation (2.21)

$$\begin{aligned} I = &\left(\frac{\partial}{\partial t}\int_b^s \bar{u}\, dz - \bar{u}_s \frac{\partial s}{\partial t} + \bar{u}_b \frac{\partial b}{\partial t}\right) + \left(\frac{\partial}{\partial x}\int_b^s \bar{u}^2\, dz - \bar{u}_s^2 \frac{\partial s}{\partial x} + \bar{u}_b^2 \frac{\partial b}{\partial x}\right) \\ &+ \left(\frac{\partial}{\partial y}\int_b^s \overline{uv} dz - \overline{uv}_s \frac{\partial s}{\partial y} + \overline{uv}_b \frac{\partial b}{\partial y}\right) + (\overline{uw}_s) - (\overline{uw}_b) \end{aligned} \tag{2.22}$$

Then, I can be arranged as

$$\begin{aligned} I = &\frac{\partial}{\partial t}\int_b^s \bar{u}\, dz + \frac{\partial}{\partial x}\int_b^s \bar{u}^2\, dz + \frac{\partial}{\partial y}\int_b^s \overline{uv} dz - \left[\bar{u}_s\left(\frac{\partial s}{\partial t} + \bar{u}_s \frac{\partial s}{\partial x} + \bar{v}_s \frac{\partial s}{\partial y} - \bar{w}_s\right)\right] \\ &+ \left[\bar{u}_b\left(\frac{\partial b}{\partial t} + \bar{u}_b \frac{\partial b}{\partial x} + \bar{v}_b \frac{\partial b}{\partial y} - \bar{w}_b\right)\right] \end{aligned} \tag{2.23}$$

By substituting the kinematic boundary conditions given in Equations (2.15) and (2.16), the last two terms are eliminated in Equation (2.23) and finally the equation (2.21) is reduced to vertically integrated form below.

$$I = \frac{\partial}{\partial t}\int_b^s \bar{u}\, dz + \frac{\partial}{\partial x}\int_b^s \bar{u}^2\, dz + \frac{\partial}{\partial y}\int_b^s \overline{uv} dz \tag{2.24}$$

Secondly, the pressure term is integrated from bottom to the free surface

$$II = -\int_b^s \frac{1}{\rho}\frac{\partial \bar{p}}{\partial x}\, dz \tag{2.25}$$

where the hydrostatic pressure, $\bar{p}$ is previously stated in Equation (2.10).

After differentiating Equation (2.10) with respect to x and dividing by g as follows

$$-\frac{1}{\rho}\frac{\partial \bar{p}}{\partial x} = -\frac{1}{\rho}\frac{\partial p_s}{\partial x} - g\frac{\partial s}{\partial x} \tag{2.26}$$

Neglecting the free surface pressure, p_s leads

$$-\frac{1}{\rho}\frac{\partial \bar{p}}{\partial x} = -g\frac{\partial s}{\partial x} \tag{2.27}$$

The vertical averaging continues with the following steps.

$$II = -\int_b^s \frac{1}{\rho}\frac{\partial \bar{p}}{\partial x}\, dz = -\int_b^s g\frac{\partial s}{\partial x}\, dz \tag{2.28}$$

$$= -g\left[\frac{\partial}{\partial x}\int_b^s s\, dz - s\frac{\partial s}{\partial x} + s\frac{\partial b}{\partial x}\right] \tag{2.29}$$

$$= -g\left[\frac{\partial (sh)}{\partial x} - s\frac{\partial s}{\partial x} + s\frac{\partial b}{\partial x}\right] \tag{2.30}$$

$$= -g\left[s\frac{\partial h}{\partial x} + h\frac{\partial s}{\partial x} - s\frac{\partial (s-b)}{\partial x}\right] \tag{2.31}$$

$$= -g\left[s\frac{\partial h}{\partial x} + h\frac{\partial s}{\partial x} - s\frac{\partial h}{\partial x}\right] \tag{2.32}$$

and finally, Equation (2.25) becomes

$$II = -gh\frac{\partial s}{\partial x} \tag{2.33}$$

The third and the last terms of the momentum equation in x-direction are the viscous terms given in vertically integrated form below.

$$III = \frac{1}{\rho}\left[\int_b^s \frac{\partial \tau_{xx}}{\partial x}\,dz + \int_b^s \frac{\partial \tau_{yx}}{\partial y}\,dz + \int_b^s \frac{\partial \tau_{zx}}{\partial z}\,dz\right] \tag{2.34}$$

Integrating each term from bottom to the surface yields

$$\begin{aligned} III = \frac{1}{\rho}\Bigg[\left(\frac{\partial}{\partial x}\int_b^s \tau_{xx}\,dz - \tau_{xx,s}\frac{\partial s}{\partial x} + \tau_{xx,b}\frac{\partial b}{\partial x}\right) \\ + \left(\frac{\partial}{\partial y}\int_b^s \tau_{yx}\,dz - \tau_{yx,s}\frac{\partial s}{\partial y} + \tau_{yx,b}\frac{\partial b}{\partial y}\right) + \left(\tau_{zx,s} - \tau_{zx,b}\right)\Bigg] \end{aligned} \tag{2.35}$$

which can be rearranged such that

$$\begin{aligned} III = \frac{1}{\rho}\frac{\partial}{\partial x}\int_b^s \tau_{xx}\,dz + \frac{1}{\rho}\frac{\partial}{\partial y}\int_b^s \tau_{yx}\,dz - \frac{1}{\rho}\left(\tau_{xx,s}\frac{\partial s}{\partial x} + \tau_{yx,s}\frac{\partial s}{\partial y} - \tau_{zx,s}\right) \\ + \frac{1}{\rho}\left(\tau_{xx,b}\frac{\partial b}{\partial x} + \tau_{yx,b}\frac{\partial b}{\partial y} - \tau_{zx,b}\right) \end{aligned} \tag{2.36}$$

Dynamic boundary conditions on the bottom (Equation 2.37) and on the water surface (Equation 2.38) are implemented (Toro, 2001)

$$\tau_{x,s} = -\tau_{xx,s}\frac{\partial s}{\partial x} - \tau_{yx,s}\frac{\partial s}{\partial y} + \tau_{zx,s} \tag{2.37}$$

$$\tau_{x,b} = \tau_{xx,b}\frac{\partial b}{\partial x} + \tau_{yx,b}\frac{\partial b}{\partial y} - \tau_{zx,b} \tag{2.38}$$

and the final version of the vertically integrated viscous terms are obtained.

$$III = \frac{1}{\rho}\frac{\partial}{\partial x}\int_b^s \tau_{xx}\,dz + \frac{1}{\rho}\frac{\partial}{\partial y}\int_b^s \tau_{yx}\,dz - \frac{1}{\rho}\left(-\tau_{x,s}\right) + \frac{1}{\rho}\left(\tau_{x,b}\right) \tag{2.39}$$

$$III = \frac{1}{\rho}\frac{\partial}{\partial x}\int_b^s \tau_{xx}\,dz + \frac{1}{\rho}\frac{\partial}{\partial y}\int_b^s \tau_{yx}\,dz + \frac{1}{\rho}\left(\tau_{x,s} + \tau_{x,b}\right) \tag{2.40}$$

2.1.4. Further Assumptions and Boundary Conditions

In the depth averaging process which is illustrated in Figure 2.2, the following treatment is done to decompose the variables of the vertical field into a depth-averaged variable and a fluctuation part, in analogy to Reynolds-averaging such that

$$\bar{u} = \tilde{u} + \hat{u} \tag{2.41}$$

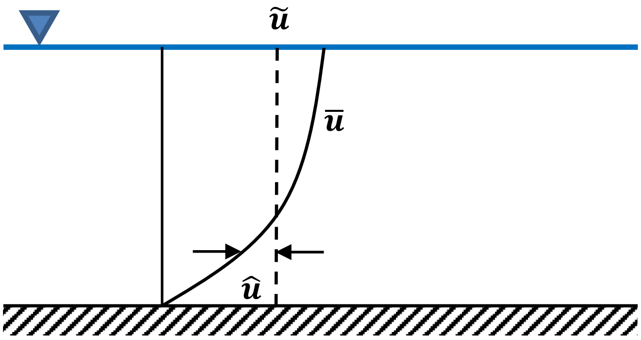

Figure 2.2. Depth-averaging variables

Depth-averaging is defined such that

$$\int_b^s \bar{u}\, dz = \tilde{u}h \tag{2.42}$$

$$\int_b^s \bar{v}\, dz = \tilde{v}h \tag{2.43}$$

where

$$\bar{u}(x, y, z, t) = \tilde{u}(x, y, t) + \hat{u}(x, y, z, t) \tag{2.44}$$

$$\bar{v}(x, y, z, t) = \tilde{v}(x, y, t) + \hat{v}(x, y, z, t) \tag{2.45}$$

and the definition implies that

$$\int_b^s \hat{u}\, dz = 0 \quad and \quad \int_b^s \hat{v}\, dz = 0 \tag{2.46}$$

The vertical averaging of the derivative and the advection terms were previously obtained such that

$$I = \frac{\partial}{\partial t}\int_b^s \bar{u}\, dz + \frac{\partial}{\partial x}\int_b^s \bar{u}^2\, dz + \frac{\partial}{\partial y}\int_b^s \overline{uv} dz \tag{2.47}$$

Second and the third terms are integrated using the variable decomposition and the depth averaging procedure stated above

$$\int_b^s \bar{u}^2\, dz = \int_b^s (\tilde{u} + \hat{u})^2\, dz = \int_b^s (\tilde{u}^2 + 2\tilde{u}\hat{u} + \hat{u}^2)\, dz$$

$$= \tilde{u}^2 \int_b^s dz + 2\tilde{u}\int_b^s \hat{u} dz + \int_b^s \hat{u}^2 dz \tag{2.48}$$

$$\int_b^s \bar{u}^2\, dz = \tilde{u}^2 h + \int_b^s \hat{u}^2 dz \tag{2.49}$$

$$\frac{\partial}{\partial y}\int_b^s \overline{uv} dz = \int_b^s (\tilde{u}\tilde{v} + \tilde{u}\hat{v} + \hat{u}\tilde{v} + \hat{u}\hat{v})\, dz \tag{2.50}$$

$$\frac{\partial}{\partial y}\int_b^s \overline{uv} dz = \tilde{u}\tilde{v}h + \int_b^s (\hat{u}\hat{v})\, dz \tag{2.51}$$

Equation (2.47) becomes

$$I = \frac{\partial}{\partial t}(\tilde{u}h) + \left[\frac{\partial}{\partial x}(\tilde{u}^2 h) + \frac{\partial}{\partial x}\int_b^s \hat{u}^2 dz\right] + \left[\frac{\partial}{\partial y}(\tilde{u}\tilde{v}h) + \frac{\partial}{\partial y}\int_b^s (\hat{u}\hat{v})\, dz\right] \tag{2.52}$$

where,

$$\frac{\partial}{\partial t}(\tilde{u}h) = \tilde{u}\frac{\partial h}{\partial t} + h\frac{\partial \tilde{u}}{\partial t} = \tilde{u}\frac{\partial (s-b)}{\partial t} + h\frac{\partial \tilde{u}}{\partial t} = \tilde{u}\frac{\partial s}{\partial t} + h\frac{\partial \tilde{u}}{\partial t} \tag{2.53}$$

$$\frac{\partial}{\partial x}(\tilde{u}^2 h) = \tilde{u}\frac{\partial h\tilde{u}}{\partial x} + h\tilde{u}\frac{\partial \tilde{u}}{\partial x} \tag{2.54}$$

$$\frac{\partial}{\partial y}(\tilde{u}\tilde{v}h) = \tilde{u}\frac{\partial h\tilde{v}}{\partial y} + h\tilde{v}\frac{\partial \tilde{u}}{\partial y} \tag{2.55}$$

All integrated (averaged) expressions are substituted, and the depth-averaged momentum equation in x-direction is obtained

$$h\frac{\partial \tilde{u}}{\partial t}+ h\tilde{u}\frac{\partial \tilde{u}}{\partial x}+h\tilde{v}\frac{\partial \tilde{u}}{\partial y}+\tilde{u}\left[\frac{\partial s}{\partial t}+\frac{\partial (h\tilde{u})}{\partial x}+\frac{\partial (h\tilde{v})}{\partial y}\right]$$

$$=-gh\frac{\partial s}{\partial x}+\frac{\partial}{\partial x}\int_b^s\left(\frac{\tau_{xx}}{\rho}-\hat{u}^2\right)dz+\frac{\partial}{\partial y}\int_b^s\left(\frac{\tau_{yx}}{\rho}-\hat{u}\hat{v}\right)dz+\left(\frac{\tau_{x,s}}{\rho}-\frac{\tau_{x,b}}{\rho}\right) \quad (2.56)$$

The equation is rearranged by neglecting the higher order terms $\hat{u}^2$ and $\hat{u}\hat{v}$, and it is rewritten by omitting the over bar on the dependent variables considering that all dependent variables in the equations are now depth-averaged.

$$h\frac{\partial u}{\partial t}+ hu\frac{\partial u}{\partial x}+hv\frac{\partial u}{\partial y}$$

$$=-gh\frac{\partial s}{\partial x}+\frac{\partial}{\partial x}\left(\frac{h\tau_{xx}}{\rho}\right)+\frac{\partial}{\partial y}\left(\frac{h\tau_{yx}}{\rho}\right)+\left(\frac{\tau_{x,s}}{\rho}-\frac{\tau_{x,b}}{\rho}\right) \quad (2.57)$$

Similar procedures are applied to derive depth-averaged momentum equation in y-direction. In conclusion, two dimensional, depth-averaged shallow flow equations in conservative form are

$$\frac{\partial h}{\partial t}+\frac{\partial hu}{\partial x}+\frac{\partial hv}{\partial y}=0 \quad (2.58)$$

$$\frac{\partial hu}{\partial t}+\frac{\partial huu}{\partial x}+\frac{\partial huv}{\partial y}=-gh\frac{\partial s}{\partial x}+\frac{\partial}{\partial x}\left(\frac{h\tau_{xx}}{\rho}\right)+\frac{\partial}{\partial y}\left(\frac{h\tau_{yx}}{\rho}\right)+\left(\frac{\tau_{x,s}}{\rho}-\frac{\tau_{x,b}}{\rho}\right) \quad (2.59)$$

$$\frac{\partial hv}{\partial t}+\frac{\partial huv}{\partial x}+\frac{\partial hvv}{\partial y}=-gh\frac{\partial s}{\partial y}+\frac{\partial}{\partial x}\left(\frac{h\tau_{xy}}{\rho}\right)+\frac{\partial}{\partial y}\left(\frac{h\tau_{yy}}{\rho}\right)+\left(\frac{\tau_{y,s}}{\rho}-\frac{\tau_{y,b}}{\rho}\right) \quad (2.60)$$

The pressure term $-gh\frac{\partial s}{\partial x}$ is treated such that

$$-gh\frac{\partial s}{\partial x} = -gh\frac{\partial(b+h)}{\partial x} = -gh\frac{\partial b}{\partial x} - gh\frac{\partial h}{\partial x} \tag{2.61}$$

where

$$\frac{\partial b}{\partial x} = S_{0,x} \tag{2.62}$$

Then,

$$-gh\frac{\partial s}{\partial x} = -ghS_{0,x} - g\frac{\partial}{\partial x}\left(\frac{1}{2}h^2\right) \tag{2.63}$$

Similarly,

$$-gh\frac{\partial s}{\partial y} = -ghS_{0,y} - g\frac{\partial}{\partial y}\left(\frac{1}{2}h^2\right) \tag{2.64}$$

$S_{0,x}$ and $S_{0,y}$ are the bed slopes in x and y directions, respectively.

Bottom shear stresses in x and y directions, $\tau_{x,b}$ and $\tau_{y,b}$ are written in terms of a friction coefficient, c_f

$$\tau_{x,b} = \rho c_f u\sqrt{u^2+v^2} \tag{2.65}$$

$$\tau_{y,b} = \rho c_f v\sqrt{u^2+v^2} \tag{2.66}$$

where $c_f = \frac{gn^2}{h^{1/3}}$

Kinematic bottom shear stress, $-\frac{\tau_{x,b}}{\rho}$ is rewritten in terms of friction slope

$$-\frac{\tau_{x,b}}{\rho} = -\frac{\rho c_f u\sqrt{u^2+v^2}}{\rho} = -\left(\frac{gn^2}{h^{1/3}}\right)u\sqrt{u^2+v^2} * \frac{h}{h} = -ghS_{f,x} \tag{2.67}$$

and similarly,

$$-\frac{\tau_{y,b}}{\rho} = -ghS_{f,y} \tag{2.68}$$

where $S_{f,x}$ and $S_{f,y}$ are the friction slopes in x and y directions, respectively.

Then, the governing momentum equations become,

$$\frac{\partial hu}{\partial t} + \frac{\partial huu}{\partial x} + \frac{\partial huv}{\partial y}$$

$$= -ghS_{0,x} + \frac{\partial}{\partial x}\left(\frac{h}{\rho}\tau_{xx}\right) + \frac{\partial}{\partial y}\left(\frac{h}{\rho}\tau_{yx}\right) + \left(\frac{\tau_{x,s}}{\rho} - ghS_{f,x} - g\frac{\partial}{\partial x}\left(\frac{1}{2}h^2\right)\right) \qquad (2.69)$$

$$\frac{\partial hu}{\partial t} + \frac{\partial huv}{\partial x} + \frac{\partial hvv}{\partial y}$$

$$= -ghS_{0,y} + \frac{\partial}{\partial x}\left(\frac{h}{\rho}\tau_{xy}\right) + \frac{\partial}{\partial y}\left(\frac{h}{\rho}\tau_{yy}\right) + \left(\frac{\tau_{y,s}}{\rho} - ghS_{f,y} - g\frac{\partial}{\partial y}\left(\frac{1}{2}h^2\right)\right) \qquad (2.70)$$

If the surface friction is neglected and only bottom friction is taken into account, the equations are written as:

$$\frac{\partial h}{\partial t} + \frac{\partial hu}{\partial x} + \frac{\partial hv}{\partial y} = 0 \qquad (2.71)$$

$$\frac{\partial hu}{\partial t} + \frac{\partial\left(hu^2 + \frac{1}{2}gh^2\right)}{\partial x} + \frac{\partial huv}{\partial y}$$

$$= -gh\left(S_{0,x} + S_{f,x}\right) + \frac{\partial}{\partial x}\left(\frac{h}{\rho}\tau_{xx}\right) + \frac{\partial}{\partial y}\left(\frac{h}{\rho}\tau_{yx}\right) \qquad (2.72)$$

$$\frac{\partial hv}{\partial t} + \frac{\partial huv}{\partial x} + \frac{\partial\left(hv^2 + \frac{1}{2}gh^2\right)}{\partial y}$$

$$= -gh\left(S_{0,y} + S_{f,y}\right) + \frac{\partial}{\partial x}\left(\frac{h}{\rho}\tau_{xy}\right) + \frac{\partial}{\partial y}\left(\frac{h}{\rho}\tau_{yy}\right) \qquad (2.73)$$

To allow compact presentations, the depth-averaged flow equations are written in vector form.

$$\frac{\partial U}{\partial t}+\frac{\partial E}{\partial x}+\frac{\partial G}{\partial y}=S \tag{2.74}$$

E and *G* represent total fluxes in x and y directions, respectively and they are given by

$$E=E_C-E_D \quad \& \quad G=G_C-G_D \tag{2.75}$$

where E_C and G_C are the convective flux vectors in x and y directions, respectively, E_D and G_D are the diffusive flux vectors in x and y directions, respectively. The variables U, E_C, E_D, G_C, G_D, and S are defined in matrix forms as follows

$$U=\begin{bmatrix} h \\ hu \\ hv \end{bmatrix}$$

$$E_C=\begin{bmatrix} hu \\ hu^2+\frac{1}{2}gh^2 \\ huv \end{bmatrix} \quad E_D=\begin{bmatrix} 0 \\ \frac{h}{\rho}\tau_{xx} \\ \frac{h}{\rho}\tau_{xy} \end{bmatrix}$$

$$G_C=\begin{bmatrix} hv \\ huv \\ hv^2+\frac{1}{2}gh^2 \end{bmatrix} \quad G_D=\begin{bmatrix} 0 \\ \frac{h}{\rho}\tau_{yx} \\ \frac{h}{\rho}\tau_{yy} \end{bmatrix}$$

$$S=\begin{bmatrix} 0 \\ -gh\left(S_{0,x}+S_{f,x}\right) \\ -gh\left(S_{0,y}+S_{f,y}\right) \end{bmatrix} \tag{2.76}$$

Assuming that the flow is homogeneous, incompressible, inviscid, two dimensional with hydrostatic pressure distribution and absence of wind forces, the governing flow equations become

$$\frac{\partial h}{\partial t} + \frac{\partial hu}{\partial x} + \frac{\partial hv}{\partial y} = 0 \tag{2.77}$$

$$\frac{\partial hu}{\partial t} + \frac{\partial\left(hu^2 + \frac{1}{2}gh^2\right)}{\partial x} + \frac{\partial huv}{\partial y} = -gh\left(S_{0,x} + S_{f,x}\right) \tag{2.78}$$

$$\frac{\partial hv}{\partial t} + \frac{\partial huv}{\partial x} + \frac{\partial\left(hv^2 + \frac{1}{2}gh^2\right)}{\partial y} = -gh\left(S_{0,y} + S_{f,y}\right) \tag{2.79}$$

Under these assumptions, Equation (2.74) is rewritten such that *E* and *G* represent only convective fluxes E_C and G_C, respectively, and the variables *U*, *E*, *G* and *S* become

$$U = \begin{bmatrix} h \\ hu \\ hv \end{bmatrix} \quad E = \begin{bmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \\ huv \end{bmatrix} \quad G = \begin{bmatrix} hv \\ huv \\ hv^2 + \frac{1}{2}gh^2 \end{bmatrix} \quad S = \begin{bmatrix} 0 \\ -gh\left(S_{0,x} + S_{f,x}\right) \\ -gh\left(S_{0,y} + S_{f,y}\right) \end{bmatrix} \tag{2.80}$$

2.2. Numerical Schemes for 2D Depth-Averaged Shallow Water Equations

2.2.1. Finite Volume Methods

For the standard finite difference approaches, finite differences are utilized to approximate derivatives. However, especially around discontinuities, the discretized approximations to differential equations do not work as expected. Accordingly, these approaches or methods cannot properly succeed near discontinuities.

Moreover, hyperbolic systems are more vulnerable to various discontinuities. In the cases where hyperbolic system of equations are needed to be solved, finite volume methods can be accepted as more appropriate and convenient approaches compared to other solutions like utilizing the point wise approximations at grid points.

Formulation for finite volume methods focuses on subdividing the spatial domains to relatively small “finite volumes”, meaning the grid cells. By doing this, this method approximates the average values for the solution of every discrete cell. By means of the fluxes through the sides of each grid cell, these averaged values are recalculated for each time step. However, defining appropriate numerical flux functions, which can represent the physical fluxes acceptably, is quite essential and sometimes can be problematic.

2.2.2. Upwind Schemes

Hyperbolic partial differential equations represent propagation problems. In nature, many flow problems have hyperbolic characteristics. In a hyperbolic problem, former property of a point considerably affects the flow condition by affecting instant property of another point. It means that the information about the solution or a disturbance propagates. Upwind schemes include the direction where the information is transmitted in numerical flow field by incorporating a finite difference approach. Accordingly, upwind methods are utilized to define more accurate numerical flux functions.

2.2.3. Riemann Problem

The Riemann problem is a one-dimensional initial value problem for the nonlinear system of hyperbolic conservation laws. It has a piecewise initial data which is discontinuous across distance x.

$$u_t + f_x(u) = 0 \tag{2.81}$$

$$u_0 = u(x,0) = \begin{cases} u_L & if \;\; x < 0 \\ u_R & if \;\; x > 0 \end{cases}$$

The solution structure of the general Riemann problem is defined with a set of waves, namely: shock waves, contact discontinuities (shear waves) and rarefaction waves, and the constant states u_L and u_R separated by the waves present in the solution. L and R stands for left and right, respectively.

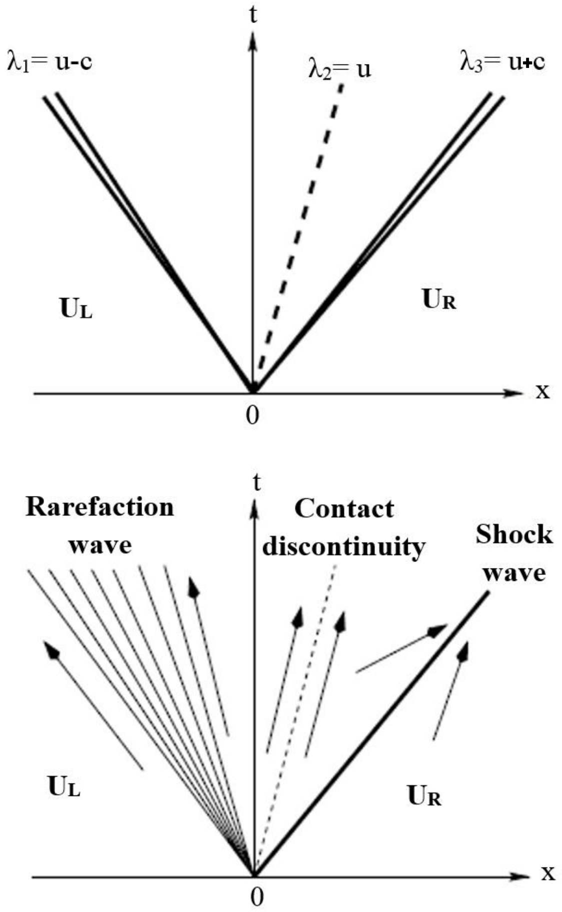

Figure 2.3. Structure of the solution of the Riemann problem (Toro, 2009)

As it is illustrated in Figure 2.3, these three waves are associated with three eigenvectors. The wave associated with the λ_2 field is a contact discontinuity and those associated with λ_1 and λ_3 is either rarefaction or shock waves. The middle wave associated with the λ_2 is a contact discontinuity, and it moves at a speed of u. The left and right waves associated with λ_1 and λ_3 are either rarefaction waves or shock waves having speeds $u - c$ or $u + c$, respectively.

Rarefaction waves are the regions of the solution where the variable u and the wave speed smoothly vary between two states u_L and u_R (George, 2004). They are both continuous across the rarefaction wave. Shock waves, on the other hand, are characterized by a discontinuity in both the variable u and the wave speed. Neither the variable u nor the wave speed is continuous across a shock wave. Since u_L and u_R have different values on both sides of the shock, the solution is considered as discontinuous. Both states u_L and u_R are identical in the case of a contact discontinuity. The wave is a contact discontinuity when the conserved variable u jumps discontinuously across the wave while the wave speed is continuous across the wave (Guinot, 2010). When the dam-break problem is considered as an example, the wave pattern which is a combination of rarefaction and shock waves can be illustrated by Figure 2.4. Figure shows that the solution of dam-break problem consist of a rarefaction wave propagating upstream and a shock wave moving downstream.

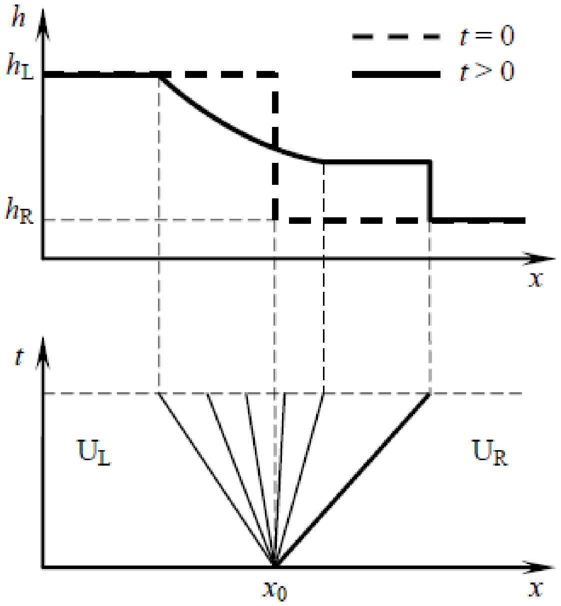

Figure 2.4. Definition sketch for dam-break problem (Guinot, 2010)

By definition, the Riemann problem is one-dimensional. However, it is also very commonly utilized when solving the multidimensional problems. Riemann problem can be solved in order to compute the numerical flux through cell interface in a finite volume grid and to update the cell quantities. Particularly, Godunov type methods based on Riemann problem solution capture and control the shock waves and the contact discontinuities that may emerge in the solution.

2.2.4. Godunov's Method

Godunov's method changes the course of computational fluid dynamics by eliminating the many deficiencies, which struggles earlier numerical methods. The Godunov's scheme that is based on a first order upwind method solves nonlinear system of conservation laws, and aims to give the most accurate solution around discontinuities.

Exact or the approximate solution of the Riemann problem is the common element of Godunov type methods. The method solves Riemann problem for each time step and at each cell interface.

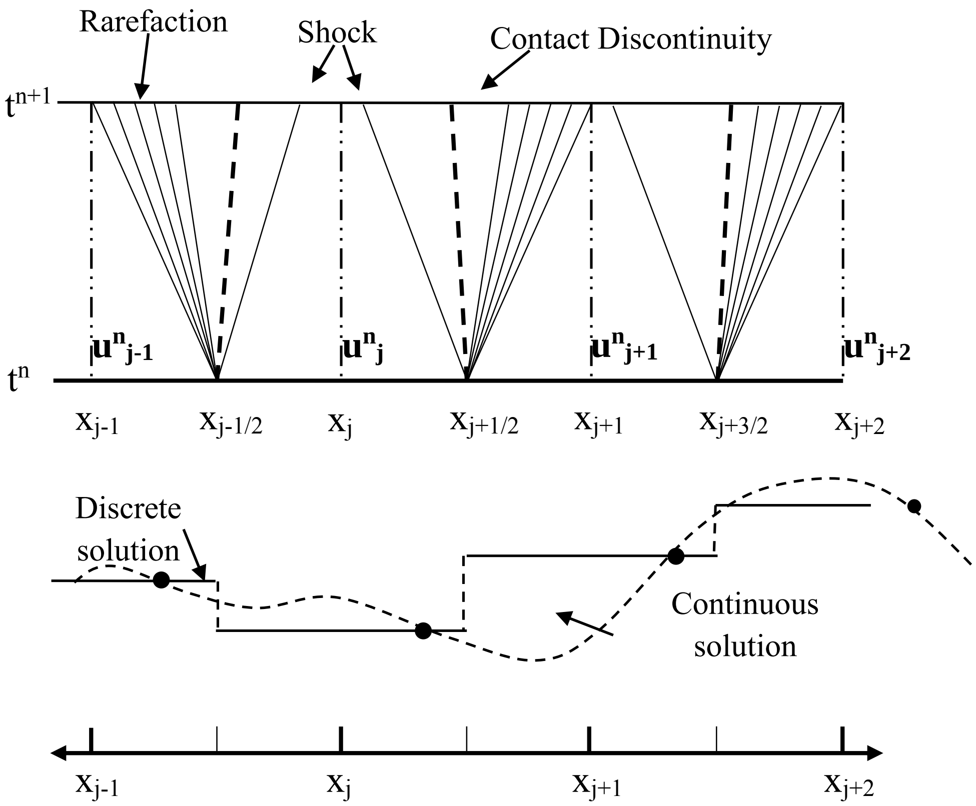

Figure 2.5. Godunov's scheme and the local Riemann problem

The Godunov method can be written in conservative form (Toro, 2009)

$$U_i^{n+1} = U_i^n + \frac{\Delta t}{\Delta x}\left[F_{i+1/2} - F_{i-1/2}\right] \tag{2.82}$$

where the numerical fluxes are given by

$$F_{i+1/2} = F\left(U_{i+1/2}(0)\right) \tag{2.83}$$

$$F_{i-1/2} = F\left(U_{i-1/2}(0)\right) \tag{2.84}$$

where $U_{i+1/2}(0)$ is the solution of the Riemann problem $RP(U_i^n, U_{i+1}^n)$ along the t-axis, and similarly $U_{i-1/2}(0)$ is the solution of the Riemann problem $RP(U_{i-1}^n, U_i^n)$ along the t-axis and both evaluated at x/t=0.

2.2.5. Approximate Riemann Solvers

From computational point of view, it is relatively expensive to use exact Riemann solution for non-linear problems (Toro, 2009). Therefore, approximate Riemann solutions became preferable considering their less computational cost, easier application and accuracy. Approximate solutions of the Riemann problem are derived from solution to linear Riemann problem. The object of all kind of approximate solvers is to estimate the Godunov flux at cell interfaces. The most widely used approximate solvers are HLL, HLLC, Roe's and Osher's approximate Riemann solvers (Toro, 2009).

2.2.6. Roe's Approximate Riemann Solver

Roe's method is an accurate and accordingly widely applied method in the field of computational fluid dynamics. The method successfully captures stationary shocks and contact discontinuities. The idea behind Roe's solution is to convert the nonlinear hyperbolic system of equations to an equivalent linear system.

For two dimensional case, numerical flux for the interface (i+1/2, j) has the form (Alcrudo & Garcia-Navarro, 1993)

$$F^{*}_{i+1/2,j} = \frac{1}{2}\left[F^{n}_{R} + F^{n}_{L} - \left|A_{i+1/2,j}\right|(U_R - U_L)\right] \qquad (2.85)$$

where $A_{i+1/2,j}$ is the Jacobian matrix of the flux F, and U_R, U_L are the left and right states of the conservative variables.

$$A_{i+1/2,j} = \frac{\partial(F \cdot n)}{\partial U} = \begin{bmatrix} 0 & n_x & n_y \\ (c^2 - u^2)n_x - uvn_y & 2un_x - vn_y & un_y \\ -uvn_x + (c^2 - v^2)n_y & vn_x & un_x + 2vn_y \end{bmatrix} \qquad (2.86)$$

where n_x, n_y are the unit normal vectors.

In order to obtain an equivalent linear system of equations, Roe matrix $\widetilde{\mathrm{A}}$ corresponding to the interface between two cells is constructed. Variables u, v, c of the Jacobian matrix $A_{i+1/2,j}$ are replaced by so-called Roe averages $\tilde{u}, \tilde{v}, \tilde{c}$.

Roe-averaged quantities of the velocities and the celerity are calculated as follows (Roe, 1981)

$$\tilde{u} = \frac{u_R\sqrt{h_R} + u_L\sqrt{h_L}}{\sqrt{h_R} + \sqrt{h_L}} \qquad \tilde{v} = \frac{v_R\sqrt{h_R} + v_L\sqrt{h_L}}{\sqrt{h_R} + \sqrt{h_L}} \qquad \tilde{c} = \sqrt{\frac{g(h_R + h_L)}{2}} \tag{2.87}$$

The eigenvalues of Roe matrix $\tilde{A}$ are,

$$\tilde{a}^1 = \tilde{u}n_x + \tilde{v}n_y + \tilde{c}$$

$$\tilde{a}^2 = \tilde{u}n_x + \tilde{v}n_y$$

$$\tilde{a}^3 = \tilde{u}n_x + \tilde{v}n_y - \tilde{c} \tag{2.88}$$

and the corresponding eigenvectors are

$$\tilde{e}^1 = \begin{pmatrix} 1 \\ \tilde{u} + \tilde{c}n_x \\ \tilde{v} + \tilde{c}n_y \end{pmatrix} \qquad \tilde{e}^2 = \begin{pmatrix} 0 \\ -\tilde{c}n_y \\ \tilde{c}n_x \end{pmatrix} \qquad \tilde{e}^3 = \begin{pmatrix} 1 \\ \tilde{u} - \tilde{c}n_x \\ \tilde{v} - \tilde{c}n_y \end{pmatrix} \tag{2.89}$$

In two dimensional space, numerical intercell flux of Roe's scheme on a linear system is

$$F(U_R, U_L) = \tfrac{1}{2}\left[F(U_R) + F(U_L) - \left|\tilde{A}\right|(U_R - U_L)\right] \tag{2.90}$$

and at cell interface (i+1/2,j), it is defined as

$$F^*_{i+1/2,j} = \tfrac{1}{2}\left[F^n_R + F^n_L - \sum_{k=1}^{3} \tilde{\alpha}^k|\tilde{a}^k|\tilde{e}^k\right] \tag{2.91}$$

where $\tilde{\alpha}^k$ are the wave strengths defined as

$$\tilde{\alpha}^{1,3} = \frac{\Delta h}{2} \pm \frac{1}{2\tilde{c}}\left[\Delta(hu)n_x + \Delta(hv)n_y - \left(\tilde{u}n_x + \tilde{v}n_y\right)\Delta h\right] \tag{2.92}$$

$$\tilde{\alpha}^2 = \frac{1}{\tilde{c}}\left\{[\Delta(hv) - \tilde{v}\Delta h]n_x - [\Delta(hu) - \tilde{u}\Delta h]n_y\right\} \tag{2.93}$$

The wave strengths are dependent to the jumps in the intermediate states of the conserved variables

$$\Delta U = U_R - U_L = \begin{pmatrix} \Delta h = h_R - h_L \\ \Delta hu = hu_R - hu_L \\ \Delta hv = hv_R - hv_L \end{pmatrix} \tag{2.94}$$

More detailed information about Roe's scheme is given in Chapter 3.

2.2.7. Higher-Order and High-Resolution (Total Variation Diminishing) Schemes

As it is stated before, Godunov's scheme has first-order accuracy and similar to other first-order methods, it suffers from inaccuracies due to numerical diffusion. High order accuracy and absence of unphysical oscillations are conflicting necessities in numerical approaches. High order schemes may come up with oscillatory solutions. On the other hand, although unphysical oscillations do not show up in monotone methods, they lead at most to first order accurate solutions.

There are nonlinear methods like TVD, namely, total variation diminishing methods that brings both higher-order accuracy and non-oscillatory behavior requirements together. TVD methods ensure that the total variation of the numerical solution is not increasing with time. Flux limiter and the slope limiter approaches are the basis for TVD methods (Toro, 2009).

2.2.8. MUSCL-type High-Order Schemes

The development of high order methods can be achieved via MUSCL approach. By means of MUSCL approach which is introduced by van Leer, first order upwind method of Godunov can be enhanced to second or even third order accuracy. As stated earlier, false oscillations surely will occur as a result of these higher order extensions. Therefore, by utilizing some TVD constraints into MUSCL approach, non-linear high order schemes are developed. Thus, the difficulty in satisfying two conflicting necessities is overcome.

Piecewise linear reconstruction is defined first for simple one-dimensional case.

$$u_i(x) = u_i^n + \frac{(x - x_i)}{\Delta x}\Delta u_i \quad , \qquad x \in [0, \Delta x] \tag{2.95}$$

where $\Delta u_i/\Delta x$ is the slope of $u_i(x)$, x_i is the location of the cell center equal to $\Delta x/2$. The value of $u_i(x)$ at the left boundary is extrapolated such that (Toro, 2009)

$$u_i^L = u_i(0) = u_i^n - \frac{1}{2}\Delta u_i \tag{2.96}$$

Similarly, the right boundary value which is still within the cell *i*

$$u_i^R = u_i(\Delta x) = u_i^n + \frac{1}{2}\Delta u_i \tag{2.97}$$

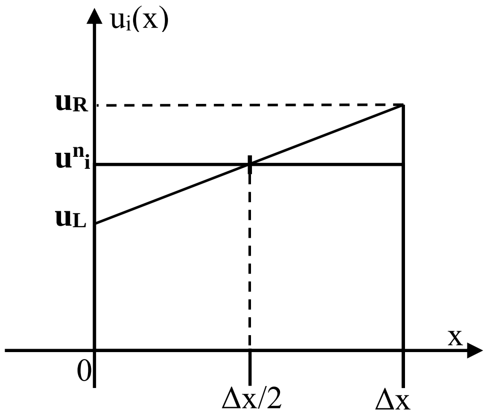

Figure 2.6. Piece-wise linear MUSCL reconstruction of data in a single cell (i), with the boundary extrapolated values u_L and u_R (Toro, 2009)

Δu_i may be expressed with a more general equation which also accounts for the different variable extrapolations at the cell interface utilizing a parameter *w*

$$\Delta u_i = \frac{1}{2}\left[(1 + w)\Delta u_{i-1/2} + (1 - w)\Delta u_{i+1/2}\right] \tag{2.98}$$

$$\Delta u_{i+1/2} = u_{i+1}^n - u_i^n \tag{2.99}$$

$$\Delta u_{i-1/2} = u_i^n - u_{i-1}^n \tag{2.100}$$

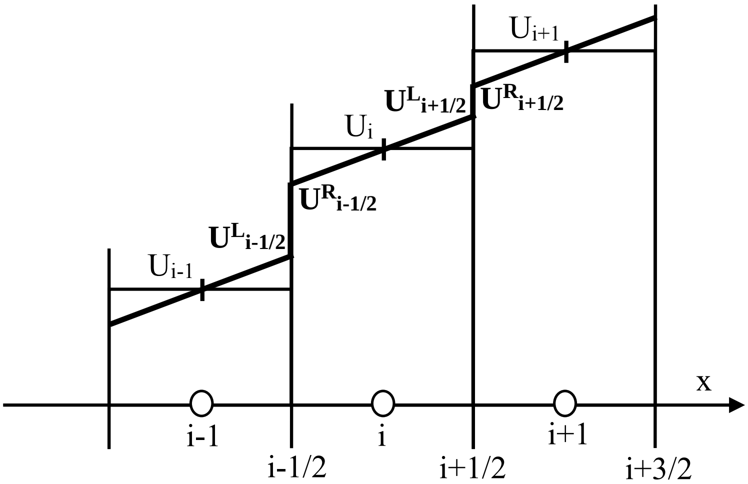

Figure 2.7. Piece-wise linear MUSCL representation (Hirsch, 1990)

The two dimensional methodology based on MUSCL data reconstruction and more detailed information about variable extrapolation will be given in the Chapter 3.

In order to avoid oscillations, the reconstructed slopes in MUSCL approach are limited with a slope limiter function that guarantees new extreme cannot be produced in the solution (Anastasiou and Chan, 1997). Moreover, the slope limiter function works correctly for both shock and smooth regions while not allowing oscillations near discontinuities and staying inactive in smooth sections.

2.2.9. Slope Limiter Approach

As previously stated, the idea behind slope limiting is to limit physically unrealistic values due to shocks, discontinuities or the sharp changes in the solution domain. Applying a limiter function, intermediate states of variables at the interface (i+1/2) become

$$u_R = u_{i+1}^n - \frac{1}{2}\varphi(r_{i+1/2})\Delta u_{i+1} \tag{2.101}$$

$$u_L = u_i^n + \frac{1}{2}\varphi(r_{i+1/2})\Delta u_i \tag{2.102}$$

The limiter φ is a function of r which is the ratio of upwind difference to local difference of variables. There are numerous limiter functions defined in literature. Some of the most commonly used limiter functions are Superbee, van Leer, van Albada and Minmod limiters (Versteeg & Malalasekera, 2007).

$$\varphi_{superbee}(r) = max[0, min(2r, 1), min(r, 2)] \tag{2.103}$$

$$\varphi_{vanLeer}(r) = \frac{r + |r|}{1 + |r|} \tag{2.104}$$

$$\varphi_{vanAlbada}(r) = \frac{r^2 + r}{r^2 + 1} \tag{2.105}$$

$$\varphi_{minmod}(r) = max[0, min(1, r)] \tag{2.106}$$

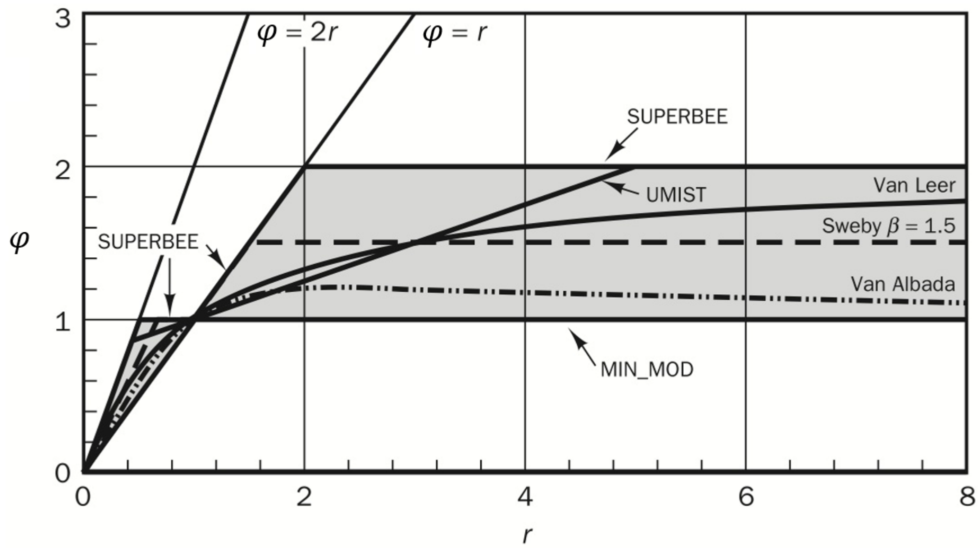

Figure 2.8. Limiter functions in φ,r diagram (Versteeg & Malalasekera, 2007)

2.2.10. Boundary Conditions

Numerical modelling of fluid flow is highly dependent on the correct implementation of boundary conditions. False or weak implementation of boundary conditions leads to both convergence and accuracy problems in the solution. The more physical conditions of flow field are applied, the more precise solutions are obtained.

In the following section, wall boundary, symmetry boundary, inflow/outflow boundary and periodic boundary conditions will be discussed.

2.2.11. Wall Boundary Condition

Steady-state inviscid flow solutions have slip boundary or slip velocity conditions applied on solid walls. Slip boundary condition implies nonzero tangential velocity while velocity normal to the wall is zero, and consequently, at the boundary velocity is parallel to the surface. In other words, Neumann type boundary condition is applied on the wall boundaries that imposes zero derivative for the velocity normal to the surface. Having zero normal velocity results in zero flux normal to the wall boundary meaning that there is no flow across the wall. While mass flux equals zero, hydrostatic pressure term is left as nonzero in the calculation of momentum fluxes.

2.2.12. Symmetry Boundary Condition

Similar to the implementation of wall boundary conditions, at a symmetry boundary no flow and no mass flux across the boundary conditions are implemented by setting normal velocity to zero.

2.2.13. Inflow/Outflow Boundary Condition

As mentioned previously, boundary conditions should also reflect the physical characteristics of the flow. In the view of such consideration, inflow/outflow boundary treatment is carried out according to the state of flow that is whether the flow is subcritical or supercritical. As Brufau and Garcia-Navarro (2000) stated, for two dimensional subcritical flows, boundary conditions for two variables must be imposed for inflow boundaries, while one is required at the outflow. On the other hand, for supercritical flows, imposition of three variables is required at inflow and none of the variables at outflow.

2.2.14. Periodic Boundary Condition

Periodic boundary condition is applied when the flow domain repeats itself with identical geometry. Periodic boundary condition can be easily imposed by copying inflow boundary interior computational cells to outside of the outflow boundary (in the ghost cells) and by copying outflow boundary interior computational cells to the upstream of inflow boundary (in the ghost cells).

CHAPTER 3

NUMERICAL SOLUTION METHOD ADOPTED TO PRESENT PROBLEM

3.1. Equations and their characteristics

The governing equations of depth-averaged flow are previously derived from Navier-Stokes equations. For the two dimensional inviscid flow, differential equations are given by

$$\frac{\partial U}{\partial t} + \nabla F = S \tag{3.1}$$

$$\mathrm{F} \cdot \mathrm{n} = \mathrm{E} n_x + \mathrm{G} n_y \tag{3.2}$$

$$\frac{\partial U}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial G}{\partial y} = S \tag{3.3}$$

where U is the vector of conservation variables, E and G are the vectors of the convection fluxes in x and y directions, respectively. S is the vector of sources of momentum in which the bed slope and the bed friction along x and y directions are taken into account. As it is stated previously in this work, viscous terms are not included in the equations. Diffusion of momentum by viscous and turbulence effects is neglected.

The vector of conserved variables is

$$U = \begin{pmatrix} h \\ hu \\ hv \end{pmatrix} \tag{3.4}$$

The flux components are

$$E = \begin{pmatrix} hu \\ hu^2 + gh^2/2 \\ huv \end{pmatrix} \; G = \begin{pmatrix} hv \\ hvu \\ hv^2 + gh^2/2 \end{pmatrix} \tag{3.5}$$

and the source vector is

$$S = \begin{pmatrix} 0 \\ -gh\left(S_{0,x} + S_{f,x}\right) \\ -gh\left(S_{0,y} + S_{f,y}\right) \end{pmatrix} \tag{3.6}$$

The system of nonlinear partial differential equations can also be expressed in the equivalent non-conservative form. (Alcrudo & Garcia-Navarro ,1993)

$$\frac{\partial U}{\partial t} + A\frac{\partial U}{\partial x} + B\frac{\partial U}{\partial y} = S \tag{3.7}$$

Then, the Jacobian matrices of the fluxes are

$$A = \frac{\partial E}{\partial U} = \begin{bmatrix} 0 & 1 & 0 \\ c^2 - u^2 & 2u & 0 \\ -uv & v & u \end{bmatrix} \qquad B = \frac{\partial G}{\partial U} = \begin{bmatrix} 0 & 0 & 1 \\ -uv & -v & u \\ c^2 - v^2 & 0 & 2v \end{bmatrix} \tag{3.8}$$

The respective eigenvalues, which are a representation of the characteristic speeds, can be obtained as

$$A\begin{cases} \tilde{a}^1 = u + c \\ \tilde{a}^2 = u \\ \tilde{a}^3 = u - c \end{cases} \qquad B\begin{cases} \tilde{b}^1 = v + c \\ \tilde{b}^2 = v \\ \tilde{b}^3 = v - c \end{cases} \tag{3.9}$$

and the corresponding eigenvectors are

$$A\begin{cases} e^1 = (1, u + c, v) \\ e^2 = (0,0,c) \\ e^3 = (1, u - c, v) \end{cases} \qquad B\begin{cases} f^1 = (1, u, v + c) \\ f^2 = (0, -c, 0\,) \\ f^3 = (1, u, v - c) \end{cases} \tag{3.10}$$

This form of the equations is particularly useful in upwind schemes since Jacobian matrices and corresponding eigenvalues and eigenvectors are linked to the physics of problem by containing any wave propagation information.

3.2. 2D Computational Domain and Numerical Scheme

In the present work, a higher-order Godunov type upwind method based on MUSCL variable extrapolation is used to solve two dimensional depth-averaged flow equations. A numerical model has been developed to describe the free surface flow around a square block fixed on the bed in a rectangular channel. It is also aimed to model the fluid flow around periodically spaced square blocks. Considering the symmetry and the periodicity, the computational domain has been chosen as shown in the Figure 3.1.

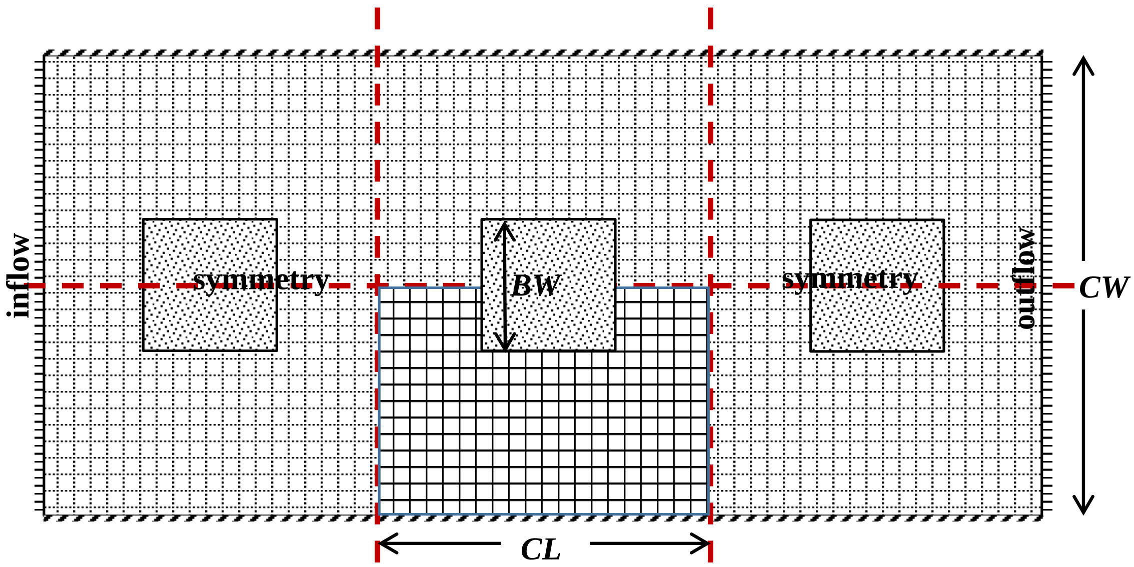

Figure 3.1. Computational domain for the flow past horizontally aligned and evenly spaced square blocks

where *BW* is the block width also equals to the block length, *CW* is the channel width, *CL* is the length of the channel within the computational domain. For both *BW* and *CW*, half of their lengths is considered within the computational domain.

Computational domain has a rectangular geometry which has been divided into uniform rectangular grids. Collocated grid configuration shown in the Figure 3.2 is proposed to deal with this regular geometry. All variables are located at the center of the computational cells and the conservation laws are applied to the same control volume.

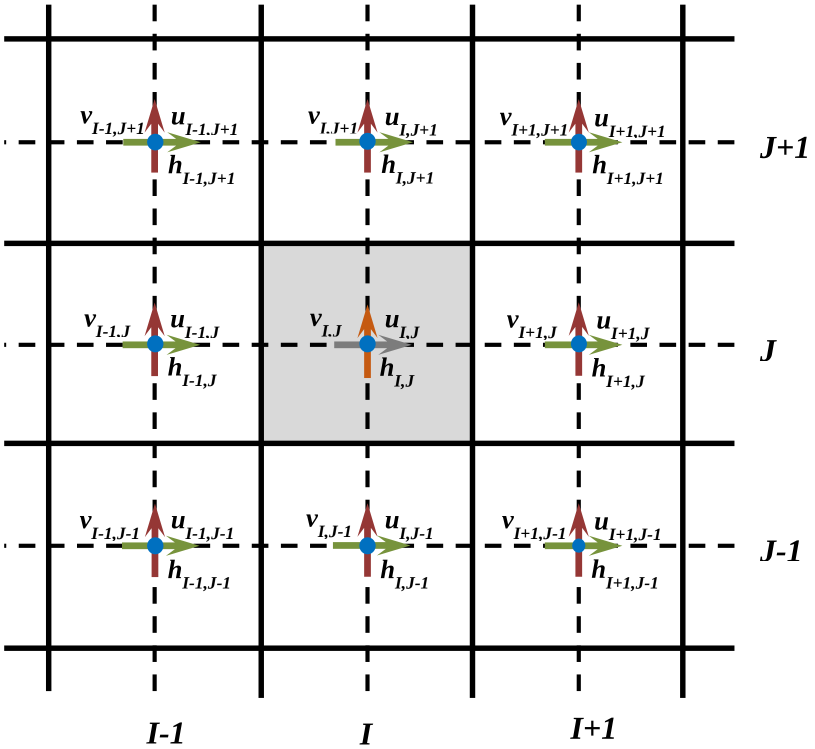

Figure 3.2. Collocated grid arrangement of the velocity components and the water depth

In general, the two dimensional finite volume formulation with a source term is given by the equation

$$U_{i,j}^{n+1} = U_{i,j}^{n} - \frac{\Delta t}{\Delta x}\left(E_{i+1/2,j} - E_{i-1/2,j}\right) - \frac{\Delta t}{\Delta y}\left(G_{i,j+1/2} - G_{i,j-1/2}\right) + \Delta t\, S_{i,j}^{n} \qquad (3.11)$$

where $E_{i+1/2,j}$ is the intercell flux between cells (i,j) and (i+1,j) and $G_{i,j+1/2}$ is the intercell flux between cells (i,j) and (i,j+1).

In this work, solution at (n+1)th time level for the computational point (i,j), $U_{i,j}^{n+1}$ is reached by using a conservative predictor-corrector algorithm as suggested by Alcrudo & Garcia-Navarro (1993). Predictor step calculations are performed at the half-time step for each cell. In the first half time step, intermediate values of variables, $U_{i,j}^{P}$ are obtained and the numerical fluxes are calculated for each cell.

Approximate solution to the conserved variables in the predictor step are obtained with the equation

$$U_{i,j}^{P} = U_{i,j}^{n} - \frac{\Delta t}{2\Delta x}\left(E_{i+1/2,j}^{*} - E_{i-1/2,j}^{*}\right) - \frac{\Delta t}{2\Delta y}\left(G_{i,j+1/2}^{*} - G_{i,j-1/2}^{*}\right) + \frac{\Delta t}{2}\, S_{i,j}^{n} \quad (3.12)$$

where $E_{i+1/2,j}^{*}$ and $G_{i,j+1/2}^{*}$ are the numerical fluxes evaluated at the half-time step.

In corrector step, next time step variables, $U_{i,j}^{n+1}$ are calculated using predictor values of numerical fluxes. It should be noted that the time step size is one full time step in the corrector step. Solution to the conserved variables at (n+1)th time level for the computational cell (i,j) is

$$U_{i,j}^{n+1} = U_{i,j}^{n} - \frac{\Delta t}{\Delta x}\left(E_{i+1/2,j}^{*P} - E_{i-1/2,j}^{*P}\right) - \frac{\Delta t}{\Delta y}\left(G_{i,j+1/2}^{*P} - G_{i,j-1/2}^{*P}\right) + \Delta t\, S_{i,j}^{P} \quad (3.13)$$

where $E_{i+1/2,j}^{*P}$ and $G_{i,j+1/2}^{*P}$ are the predicted numerical fluxes in x and y directions, respectively.

As stated previously in Chapter 2, approximation to the numerical fluxes plays a vital role in application of a finite volume method. In this study, discretization of the numerical fluxes is performed based on Roe's approximate Riemann solver with Monotonic Upstream Schemes for Conservation Laws (MUSCL) (Van Leer, 1979) extrapolation. Roe's scheme is first order accurate in space; however, it is extended to second order spatial accuracy through MUSCL scheme. In order to maintain stability and to ensure monotonicity of this high-order scheme, slope limiter technique is applied which is discussed later. In order to define Roe numerical flux function through cell interface, interface values of conserved variables must be defined.

Intermediate values are obtained by a combination of backward and forward extrapolations at the left and the right of the cell interface. The reconstruction can be formulated as (Hirsch, 1990):

$$U^{L}_{i+1/2,j} = U_{i,j} + \frac{\varepsilon}{4}\left[(1-k)\left(U_{i,j} - U_{i-1,j}\right) + (1+k)\left(U_{i+1,j} - U_{i,j}\right)\right] \tag{3.14}$$

$$U^{R}_{i+1/2,j} = U_{i+1,j} - \frac{\varepsilon}{4}\left[(1-k)\left(U_{i+2,j} - U_{i+1,j}\right) + (1+k)\left(U_{i+1,j} - U_{i,j}\right)\right] \tag{3.15}$$

where $U^{R}_{i+1/2}$ is the extrapolated value of the conserved variable at the right of (i+1/2) within cell (i+1) and $U^{L}_{i+1/2}$ is the extrapolated value of the conserved variable at the left of (i+1/2) within cell (i). In Equations (3.14) and (3.15), the parameter ε defines the order of the scheme and the parameter k considers the combination of backward and forward extrapolations and it can take values between -1 and 1.

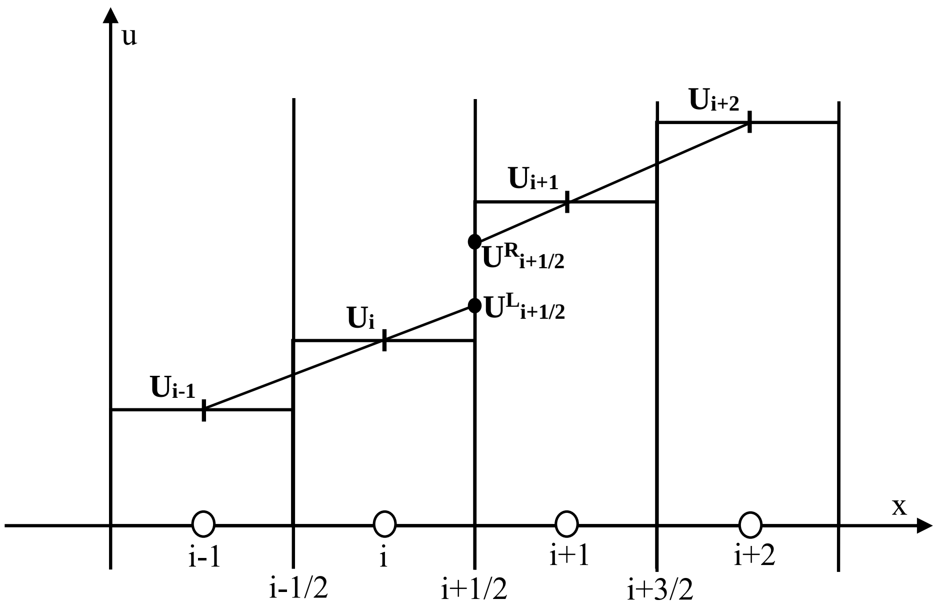

Figure 3.3. Linear one-sided extrapolation of interface values for k= -1 (Hirsch, 1990)

In Equations (3.14) and (3.15), ε=0 corresponds to a first-order scheme while ε=1 leads to a higher-order scheme. Moreover, when k=1 the interface values are the arithmetic mean of the adjacent cell values and the upwind character is totally lost. With k=0 a linear interpolation between one upstream and one downstream cell is obtained. In this study, k is set to -1 which corresponds to a second order accurate, linear one-side extrapolation at the interface between the averaged values at the two upstream cells (i) and (i-1) such that,

$$U^L_{i+1/2,j} = U_{i,j} + \frac{1}{2}\left(U_{i,j} - U_{i-1,j}\right) \tag{3.16}$$

$$U^R_{i+1/2,j} = U_{i+1,j} - \frac{1}{2}\left(U_{i+2,j} - U_{i+1,j}\right) \tag{3.17}$$

While modelling hyperbolic conservation laws by using high-order finite volume methods, incorporating a slope-limiter method is very essential to deal with stability. Limiters are defined with a parameter, *r* which is the ratio of upwind difference to local difference,

$$r_{i+1/2} = \frac{\Delta U_{upwind}}{\Delta U_{local}} \tag{3.18}$$

and the limiting function is represented by $\varphi(r)$. Upwind direction is determined based on the characteristic wave speed (u-c) at the relevant cell interface (Toro, 2000).

Intermediate boundary extrapolated values are written such that,

$$U^L_{i+1/2,j} = U_{i,j} + \frac{1}{2}\varphi\left(r_{i+1/2,j}\right)\left(U_{i,j} - U_{i-1,j}\right) \tag{3.19}$$

$$U^R_{i+1/2,j} = U_{i+1,j} - \frac{1}{2}\varphi\left(r_{i+1/2,j}\right)\left(U_{i+2,j} - U_{i+1,j}\right) \tag{3.20}$$

and similarly for the cell face in y-direction

$$U^L_{i,j+1/2} = U_{i,j} + \frac{1}{2}\varphi\left(r_{i,j+1/2}\right)\left(U_{i,j} - U_{i,j-1}\right) \tag{3.21}$$

$$U^R_{i,j+1/2} = U_{i,j+1} - \frac{1}{2}\varphi\left(r_{i,j+1/2}\right)\left(U_{i,j+2} - U_{i,j+1}\right) \tag{3.22}$$

There are many common limiting functions in literature. Their function within the scheme is to limit forward and backward gradients by adding some dissipation, and consequently preserve stability and monotonicity (Alcrudo & Garcia-Navarro, 1993). Superbee (Roe, 1986), van Leer (van Leer, 1974), van Albada (van Albada, 1982) and Minmod (Roe, 1986) limiters have been tried in this study.

After the MUSCL extrapolation with limiting functions is applied, and the intermediate values U_R and U_L are expressed, the numerical fluxes can be obtained from

$$E^*_{i+1/2,j} = E^*\left(U^L_{i+1/2,j}, U^R_{i+1/2,j}\right) \tag{3.23}$$

$$G^*_{i,j+1/2} = G^*\left(U^L_{i,j+1/2}, U^R_{i,j+1/2}\right) \tag{3.24}$$

where the numerical fluxes $E^*_{i+1/2,j}$ and $G^*_{i,j+1/2}$ have second-order spacial accuracy.

Roe scheme numerical fluxes in x- and y-directions are approximated by the equations

$$E^*_{i+1/2,j} = \frac{1}{2}\left[E^*\left(U^L_{i+1/2,j}\right) + E^*\left(U^R_{i+1/2,j}\right) - |A|_{i+1/2,j}\left(U^R_{i+1/2,j} - U^L_{i+1/2,j}\right)\right] \tag{3.25}$$

or

$$E^*_{i+1/2,j} = \frac{1}{2}\left[E^R + E^L - \Delta E^*\right] \tag{3.26}$$

and

$$G^*_{i,j+1/2} = \frac{1}{2}\left[G^*\left(U^L_{i,j+1/2}\right) + G^*\left(U^R_{i,j+1/2}\right) - |B|_{i,j+1/2}\left(U^R_{i,j+1/2} - U^L_{i,j+1/2}\right)\right] \tag{3.27}$$

or

$$G^*_{i,j+1/2} = \frac{1}{2}\left[G^R + G^L - \Delta G^*\right] \tag{3.28}$$

$E^*\left(U^R_{i+1/2,j}\right)$ and $E^*\left(U^L_{i+1/2,j}\right)$ are the interface values of fluxes calculated in terms of interface values of variables in x-direction, i.e.,

$$U^R = \begin{bmatrix} h_R \\ (hu)_R \\ (hv)_R \end{bmatrix} \rightarrow E^*(U^R) = E^R = \begin{bmatrix} h_R u_R \\ (hu)_R u_R + g h_R^2/2 \\ (hv)_R u_R \end{bmatrix} \tag{3.29}$$

$$U^L = \begin{bmatrix} h_L \\ (hu)_L \\ (hv)_L \end{bmatrix} \rightarrow E^*(U^L) = E^L = \begin{bmatrix} h_L u_L \\ (hu)_L u_L + g h_L^2/2 \\ (hv)_L u_L \end{bmatrix} \tag{3.30}$$

and the interface values of y-direction fluxes are

$$U^R = \begin{bmatrix} h_R \\ (hu)_R \\ (hv)_R \end{bmatrix} \rightarrow G^*(U^R) = G^R = \begin{bmatrix} h_R v_R \\ (hu)_R v_R \\ (hv)_R v_R + g h_R^2/2 \end{bmatrix} \tag{3.31}$$

$$U^L = \begin{bmatrix} h_L \\ (hu)_L \\ (hv)_L \end{bmatrix} \rightarrow G^*(U^L) = G^L = \begin{bmatrix} h_L v_L \\ (hu)_L v_L \\ (hv)_L v_L + g h_L^2/2 \end{bmatrix} \tag{3.32}$$

$\Delta\boldsymbol{E}^*$ and $\Delta\boldsymbol{G}^*$ are called as the flux variations or may be considered as artificial diffusive fluxes. These variations are expressed with a summation of simple wave contributions.

$$\Delta E^* = \sum_{k=1}^{3} \tilde{\alpha}^k |\tilde{a}^k| \tilde{e}^k \tag{3.33}$$

$$\Delta G^* = \sum_{k=1}^{3} \tilde{\gamma}^k |\tilde{b}^k| \tilde{f}^k \tag{3.34}$$

where $\tilde{e}^1$, $\tilde{e}^2$, $\tilde{e}^3$ are the eigenvectors of the matrix A with the eigenvalues $\tilde{a}^1$, $\tilde{a}^2$, $\tilde{a}^3$, and $\tilde{\alpha}^1, \tilde{\alpha}^2, \tilde{\alpha}^3$ are the wave strengths (variations in x-direction), and $\tilde{f}^1$, $\tilde{f}^2$, $\tilde{f}^3$ are the eigenvectors of the matrix B with the eigenvalues $\tilde{b}^1$, $\tilde{b}^2$, $\tilde{b}^3$, and $\tilde{\gamma}^1, \tilde{\gamma}^2, \tilde{\gamma}^3$ are the wave strengths (variations in y-direction).

They are all written in terms of so-called Roe averages (Roe, 1981).

$$\tilde{u} = \frac{u_R\sqrt{h_R} + u_L\sqrt{h_L}}{\sqrt{h_R} + \sqrt{h_L}} \qquad \tilde{v} = \frac{v_R\sqrt{h_R} + v_L\sqrt{h_L}}{\sqrt{h_R} + \sqrt{h_L}} \qquad \tilde{c} = \sqrt{\frac{g(h_R + h_L)}{2}} \tag{3.35}$$

where intermediate states of velocities are calculated as follows,

$$\begin{aligned} u_R &= (hu)_R/h_R \\ u_L &= (hu)_L/h_L \\ v_R &= (hv)_R/h_R \\ v_R &= (hv)_L/h_L \end{aligned} \tag{3.36}$$

So, the average eigenvalues are

$$\begin{aligned} \tilde{a}^1 &= \tilde{u} + \tilde{c} \qquad & \tilde{b}^1 &= \tilde{v} + \tilde{c} \\ \tilde{a}^2 &= \tilde{u} & \tilde{b}^2 &= \tilde{v} \\ \tilde{a}^3 &= \tilde{u} - \tilde{c} & \tilde{b}^3 &= \tilde{v} - \tilde{c} \end{aligned} \tag{3.37}$$

and the corresponding eigenvectors are

$$\tilde{e}^1 = \begin{pmatrix} 1 \\ \tilde{u} + \tilde{c} \\ \tilde{v} \end{pmatrix} \qquad \tilde{e}^2 = \begin{pmatrix} 0 \\ 0 \\ \tilde{c} \end{pmatrix} \qquad \tilde{e}^3 = \begin{pmatrix} 1 \\ \tilde{u} - \tilde{c} \\ \tilde{v} \end{pmatrix} \tag{3.38}$$

$$\tilde{f}^1 = \begin{pmatrix} 1 \\ \tilde{u} \\ \tilde{v} + \tilde{c} \end{pmatrix} \qquad \tilde{f}^2 = \begin{pmatrix} 0 \\ -\tilde{c} \\ 0 \end{pmatrix} \qquad \tilde{f}^3 = \begin{pmatrix} 1 \\ \tilde{u} \\ \tilde{v} - \tilde{c} \end{pmatrix} \tag{3.39}$$

The wave strengths are given as,

$$\begin{aligned} \tilde{\alpha}^1 &= \frac{\Delta h}{2} + \frac{1}{2\tilde{c}}[\Delta(hu) - \tilde{u}\Delta h] \qquad & \tilde{\gamma}^1 &= \frac{\Delta h}{2} + \frac{1}{2\tilde{c}}[\Delta(hv) - \tilde{v}\Delta h] \\ \tilde{\alpha}^2 &= \frac{1}{\tilde{c}}[\Delta(hv) - \tilde{v}\Delta h] & \tilde{\gamma}^2 &= \frac{1}{\tilde{c}}[-\Delta(hu) + \tilde{u}\Delta h] \\ \tilde{\alpha}^3 &= \frac{\Delta h}{2} - \frac{1}{2\tilde{c}}[\Delta(hu) - \tilde{u}\Delta h] & \tilde{\gamma}^3 &= \frac{\Delta h}{2} - \frac{1}{2\tilde{c}}[\Delta(hv) - \tilde{v}\Delta h] \end{aligned} \tag{3.40}$$

where $\Delta(\,) = (\,)_R - (\,)_L$ are the jumps in conserved quantity

$$\begin{aligned} \Delta h &= h_R - h_L \\ \Delta(hu) &= (hu)_R - (hu)_L \\ \Delta(hv) &= (hv)_R - (hv)_L \end{aligned} \tag{3.41}$$

And finally the flux variations are

$$\Delta E^* = \begin{bmatrix} \Delta E_1 \\ \Delta E_2 \\ \Delta E_3 \end{bmatrix} = \begin{bmatrix} \tilde{\alpha}^1|\tilde{a}^1|\tilde{e}^1(1) + \tilde{\alpha}^2|\tilde{a}^2|\tilde{e}^2(1) + \tilde{\alpha}^3|\tilde{a}^3|\tilde{e}^3(1) \\ \tilde{\alpha}^1|\tilde{a}^1|\tilde{e}^1(2) + \tilde{\alpha}^2|\tilde{a}^2|\tilde{e}^2(2) + \tilde{\alpha}^3|\tilde{a}^3|\tilde{e}^3(2) \\ \tilde{\alpha}^1|\tilde{a}^1|\tilde{e}^1(3) + \tilde{\alpha}^2|\tilde{a}^2|\tilde{e}^2(3) + \tilde{\alpha}^3|\tilde{a}^3|\tilde{e}^3(3) \end{bmatrix} \tag{3.42}$$

$$\Delta G^* = \begin{bmatrix} \Delta G_1 \\ \Delta G_2 \\ \Delta G_3 \end{bmatrix} = \begin{bmatrix} \tilde{\gamma}^1|\tilde{b}^1|\tilde{f}^1(1) + \tilde{\gamma}^2|\tilde{b}^2|\tilde{f}^2(1) + \tilde{\gamma}^3|\tilde{b}^3|\tilde{f}^3(1) \\ \tilde{\gamma}^1|\tilde{b}^1|\tilde{f}^1(2) + \tilde{\gamma}^2|\tilde{b}^2|\tilde{f}^2(2) + \tilde{\gamma}^3|\tilde{b}^3|\tilde{f}^3(2) \\ \tilde{\gamma}^1|\tilde{b}^1|\tilde{f}^1(3) + \tilde{\gamma}^2|\tilde{b}^2|\tilde{f}^2(3) + \tilde{\gamma}^3|\tilde{b}^3|\tilde{f}^3(3) \end{bmatrix} \tag{3.43}$$

The numerical fluxes are evaluated as (Toro, 2009)

$$E^*_{i+1/2,j} = \frac{1}{2}\left[E^R + E^L - \sum_{k=1}^{3} \tilde{\alpha}^k|\tilde{a}^k|\tilde{e}^k\right] \tag{3.44}$$

$$G^*_{i,j+1/2} = \frac{1}{2}\left[G^R + G^L - \sum_{k=1}^{3} \tilde{\gamma}^k|\tilde{b}^k|\tilde{f}^k\right] \tag{3.45}$$

Then, the discretized form of the governing equations are written as a two-step predictor-corrector procedure.

For conservation of mass,

$$h^P_{i,j} = h^n_{i,j} - \frac{\Delta t}{2\Delta x}\left(E^*_{i+1/2,j} - E^*_{i-1/2,j}\right) - \frac{\Delta t}{2\Delta y}\left(G^*_{i,j+1/2} - G^*_{i,j-1/2}\right) \tag{3.46}$$

$$h^{n+1}_{i,j} = h^n_{i,j} - \frac{\Delta t}{\Delta x}\left(E^{*P}_{i+1/2,j} - E^{*P}_{i-1/2,j}\right) - \frac{\Delta t}{\Delta y}\left(G^{*P}_{i,j+1/2} - G^{*P}_{i,j-1/2}\right) \tag{3.47}$$

For conservation of x-momentum,

$$(hu)_{i,j}^{P} = (hu)_{i,j}^{n} - \frac{\Delta t}{2\Delta x}\left(E_{i+1/2,j}^{*} - E_{i-1/2,j}^{*}\right) - \frac{\Delta t}{2\Delta y}\left(G_{i,j+1/2}^{*} - G_{i,j-1/2}^{*}\right) + \frac{\Delta t}{2} S_{i,j}^{n} \tag{3.48}$$

$$(hu)_{i,j}^{n+1} = (hu)_{i,j}^{n} - \frac{\Delta t}{\Delta x}\left(E_{i+1/2,j}^{*P} - E_{i-1/2,j}^{*P}\right) - \frac{\Delta t}{\Delta y}\left(G_{i,j+1/2}^{*P} - G_{i,j-1/2}^{*P}\right) + \Delta t S_{i,j}^{P} \tag{3.49}$$

and for conservation of y-momentum,

$$(hv)_{i,j}^{P} = (hv)_{i,j}^{n} - \frac{\Delta t}{2\Delta x}\left(E_{i+1/2,j}^{*} - E_{i-1/2,j}^{*}\right) - \frac{\Delta t}{2\Delta y}\left(G_{i,j+1/2}^{*} - G_{i,j-1/2}^{*}\right) + \frac{\Delta t}{2} S_{i,j}^{n} \tag{3.50}$$

$$(hv)_{i,j}^{n+1} = (hv)_{i,j}^{n} - \frac{\Delta t}{\Delta x}\left(E_{i+1/2,j}^{*P} - E_{i-1/2,j}^{*P}\right) - \frac{\Delta t}{\Delta y}\left(G_{i,j+1/2}^{*P} - G_{i,j-1/2}^{*P}\right) + \Delta t S_{i,j}^{P} \tag{3.51}$$

3.3. Stability Criteria and Boundary Conditions

A limitation is introduced on the computational time step to maintain stability of numerical solution. In order to damp out artificial oscillations while applying a high-resolution scheme, the CFL number should be kept less than 0.75 (Mohammadian et al, 2005). In two dimensional space, the stability condition is defined as

$$\Delta t_{i,j} = \frac{CFL\ \min(\Delta x, \Delta y)}{\left(c + \sqrt{u^2 + v^2}\right)_{i,j}} \tag{3.52}$$

where CFL value is fixed to control the time step size. In the present computations time step is a function of local velocity since a constant mesh size in space is used, and it is selected to satisfy the CFL criterion and ensure stability.

In order to start the time evolution of computation on the two dimensional domain, initially uniform flow conditions has been provided by specifying the values of three dependent variables $\mathrm{h} = \mathrm{H}_0, \mathrm{u} = \mathrm{U}_0, \mathrm{v} = 0$ at every grid point i, j for the time t=0.

The numerical treatment of the boundary conditions was explained in the topic of Boundary Conditions in Chapter 2. (In addition, it should be commented that even though the numerical scheme is second order accurate, boundary cells have first-order accuracy).

Around wall boundaries, ghost cell values are assigned by setting normal velocity on the wall surface to zero. With this treatment, convective fluxes normal to the wall boundaries are forced to be zero. In addition, characteristic boundary conditions from Roe's method is applied to the left and right states (Dadone and Grossman, 1994).

Periodic boundary condition is well imposed by copying inflow boundary interior computational cells to the outflow boundary ghost cells and by copying outflow boundary interior computational cells to the inflow boundary ghost cells.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1. 1D Test cases

In order to test the shock capturing ability of the numerical solution method, one-dimensional test cases with known analytical solutions (Stoker, 1957; Wu et al., 1999; Zoppou & Roberts, 2003) are considered. The most commonly used, 1D dam-break problem is selected and the code is run for sub and supercritical states of flow. In this test case, total length of the horizontal wide channel is 2000 m. Initially water levels on the left ($x<1000$ m) and on the right ($x>1000$ m) are constants, velocity is zero everywhere and an imaginary wall is fixed at $x = 1000$ m. At $t=0$ the imaginary wall is removed and propagation of the water surface discontinuity in upstream ($x<1000$) and downstream ($x>1000$) directions is computed. Comparisons of numerical solutions at $t=50$ s with corresponding analytical solutions are shown in Figure 4.1 and Figure 4.2. It is observed that the code can successfully reproduce the rarefaction and shock wave propagations in the computational domain.

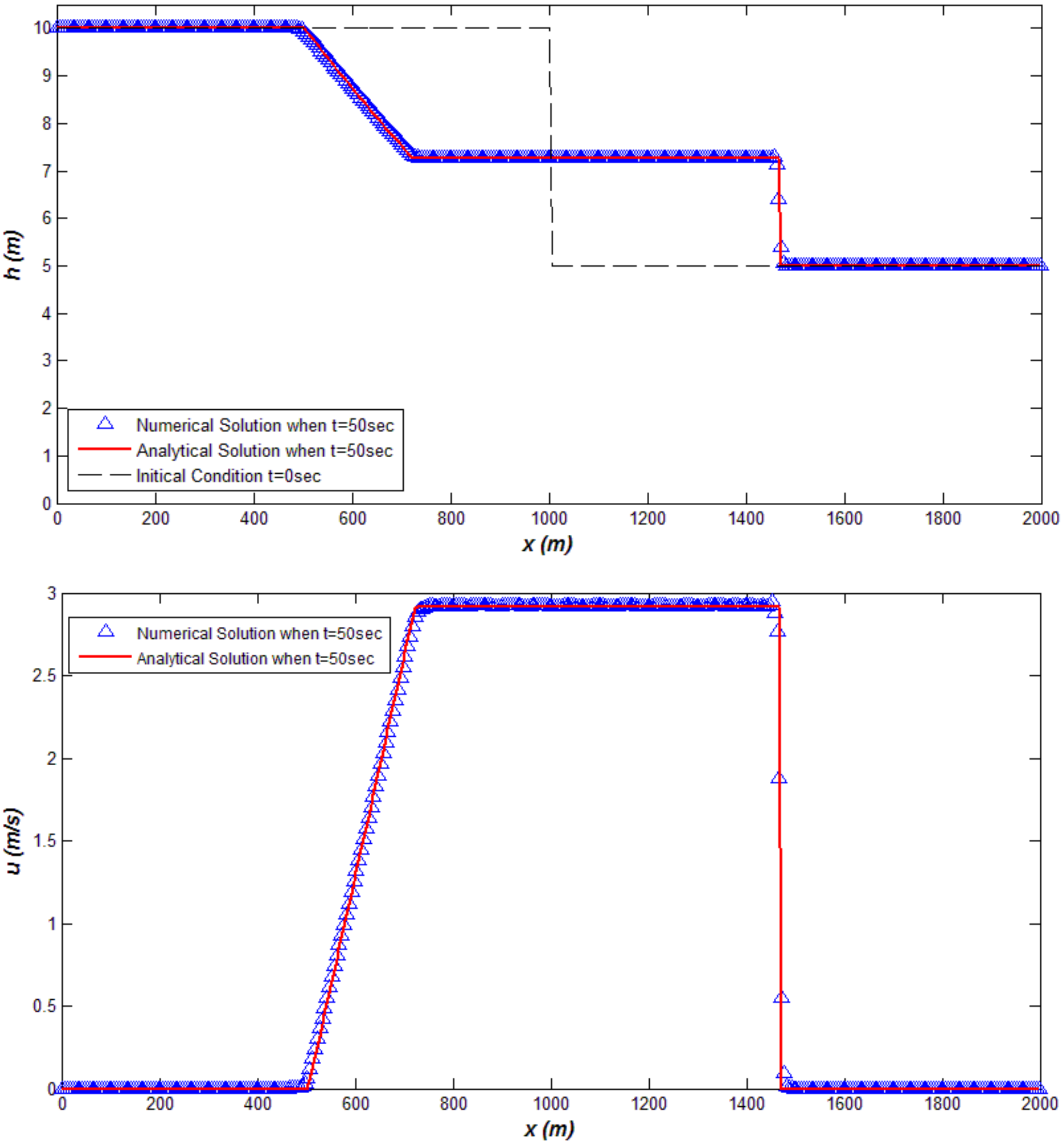

Figure 4.1. 1D test case for the analytical and numerical solutions of dam-break problem for subcritical state where HL=10 m and HR=5 m obtained at t=50sec using Roe scheme

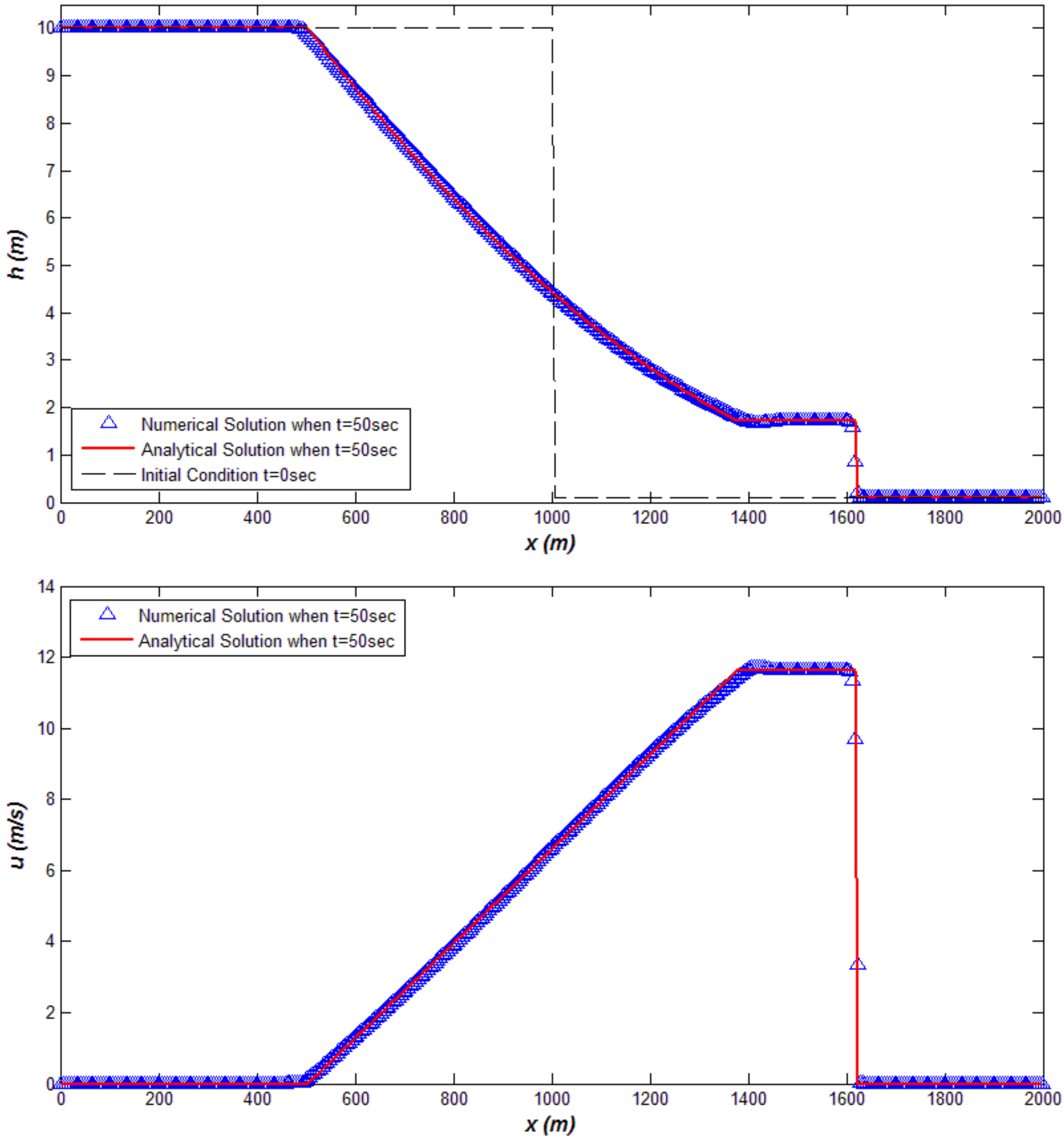

Figure 4.2. 1D test case for the analytical and numerical solutions of dam-break problem for supercritical state where HL=10 m and HR=0.1 m obtained at t=50sec using Roe scheme

4.2. 2D Solutions

Performance of the two dimensional depth-averaged numerical model is verified by defining several test cases for the generic flow domain (Figure. 4.3) considered in this study. Subcritical and supercritical flow states are tested with special attention paid on chocking phenomenon. The effects of different mesh sizes, number of iterations, different types of limiters on numerical accuracy are observed. Results are given with a mesh size of 0.02 m since the model having mesh sizes of 0.04, 0.01 and 0.05 m produces no significant difference in results.

Numerical stability of computations is maintained by using appropriate time steps in the iterative computations. The practical tool for determination of the time step is the CFL number that should theoretically be less than 1 to avoid any disturbance signal moving more than one mesh size in one time step. However, in practice the limiting value may be smaller than the theoretical value depending on complexity of the flow type and nonlinearity of the governing equations. In the literature (Mohammadian et al., 2005) the range of recommended values is 0.7~0.9. For the present study the CFL value has been fixed as 0.5, after some preliminary tests to avoid any numerical oscillations or error accumulation due to large time step. In this study, average runtime of the program is measured as 15 minutes. The test channel dimensions, are shown in the Figure 4.3, where the channel length, CL = 6.2 m, obstruction block width and length, BW = 0.2 m, section (1) channel width, CW = 1.2 m, at section (2) the reduced channel width is 1 m, and Manning roughness, n = 0.01 for all cases.

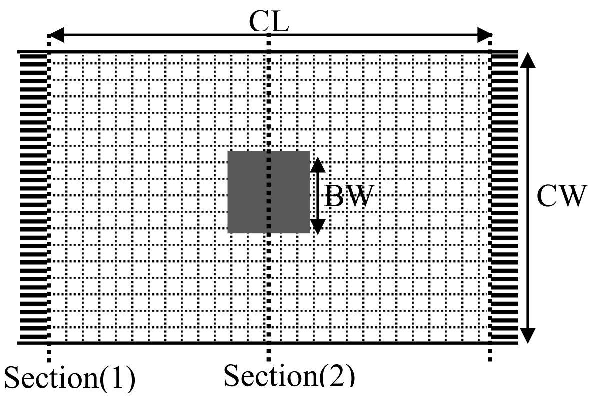

Figure 4.3. Channel dimensions for all test cases

4.1.1. Supercritical Flow Test Case

In open channel flows behavior of water surface in sub and supercritical flows is quite different. As a first test case, two dimensional flow around a square block located at the middle of the rectangular prismatic channel is considered. The channel bed slope and the water depth and average velocity for uniform flow conditions (without the block) are determined and applied as initial data all over the flow domain. The data set chosen for the supercritical flow test case is given in Table 4.1. As it can be seen from the computed specific energies, normally, there shouldn't be any choking, flow can continue in supercritical state through the contracted section around the block.

Table 4.1. Supercritical flow data

Discharge, Q			15 m^3/s
Initial data: uniform flow depth, H_0			1.6 m
Initial data: uniform flow velocity, U_0			7.813 m/s
Initial data: y-direction velocity, V			0 m/s
Channel slope, S_0			0.0221
Froude Number			1.972
State of the flow			Supercritical
SECTION (1) flow data		SECTION (2) flow data	
b_1	1.2 m	b_2	1 m
q_1	12.5 $m^3/s/m$	q_2	15 $m^3/s/m$
E_1	4.711 m	$E_2=E_1$	4.711 m
Ec_1	3.774 m	Ec_2	4.262 m $< E_2$

An important issue in numerical solution to hyperbolic problems is the number and type of boundary conditions to be applied at the inflow and outflow sections. Different applications of boundary conditions will be required for sub and supercritical flows (Brufau et al. 2002). For supercritical flows, all of the dependent variables should be specified at the inflow section. In the present study, the water depth, h, and the unit discharges, q_x=hu, and q_y=hv are fixed at the inflow boundary (Section-1). At the outflow, there is no requirement as a boundary condition for supercritical case, therefore, all variables are left free. This type of free boundary application, practically results in constant slope for all variables at the outflow section.

The free surface profile for supercritical test case obtained from numerical solution at the end of 8000 iterations is shown in Figure 4.4. Very sharp edged cross waves at the downstream of the block are observed. Flow depth where the cross waves are formed varies between 1.53 m and 1.74 m. It is observed that the average flow depth is very close to the uniform flow depth given as initial condition.

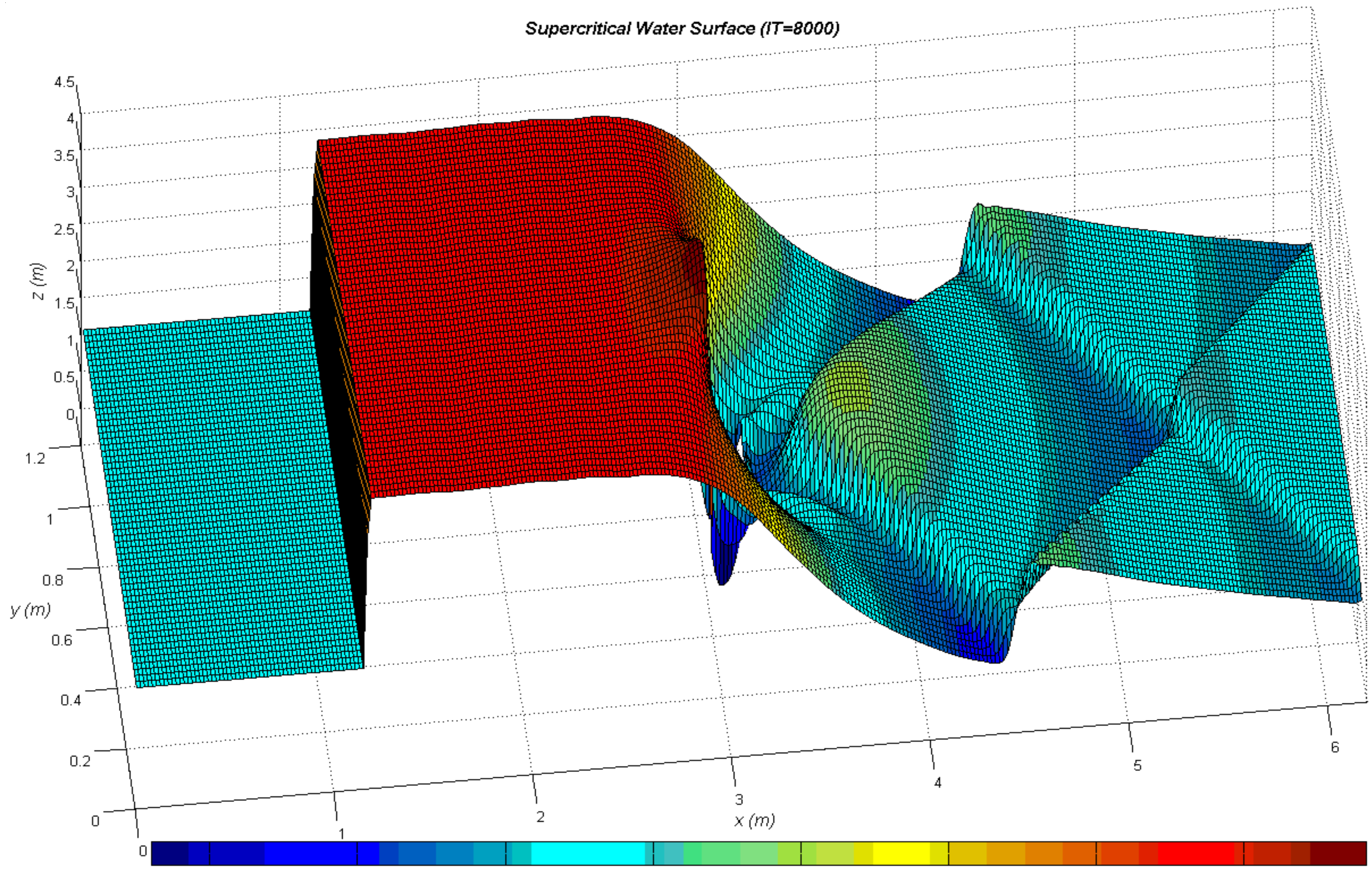

Figure 4.4. Supercritical surface profile (IT=8000 & mesh size=0.02 m)

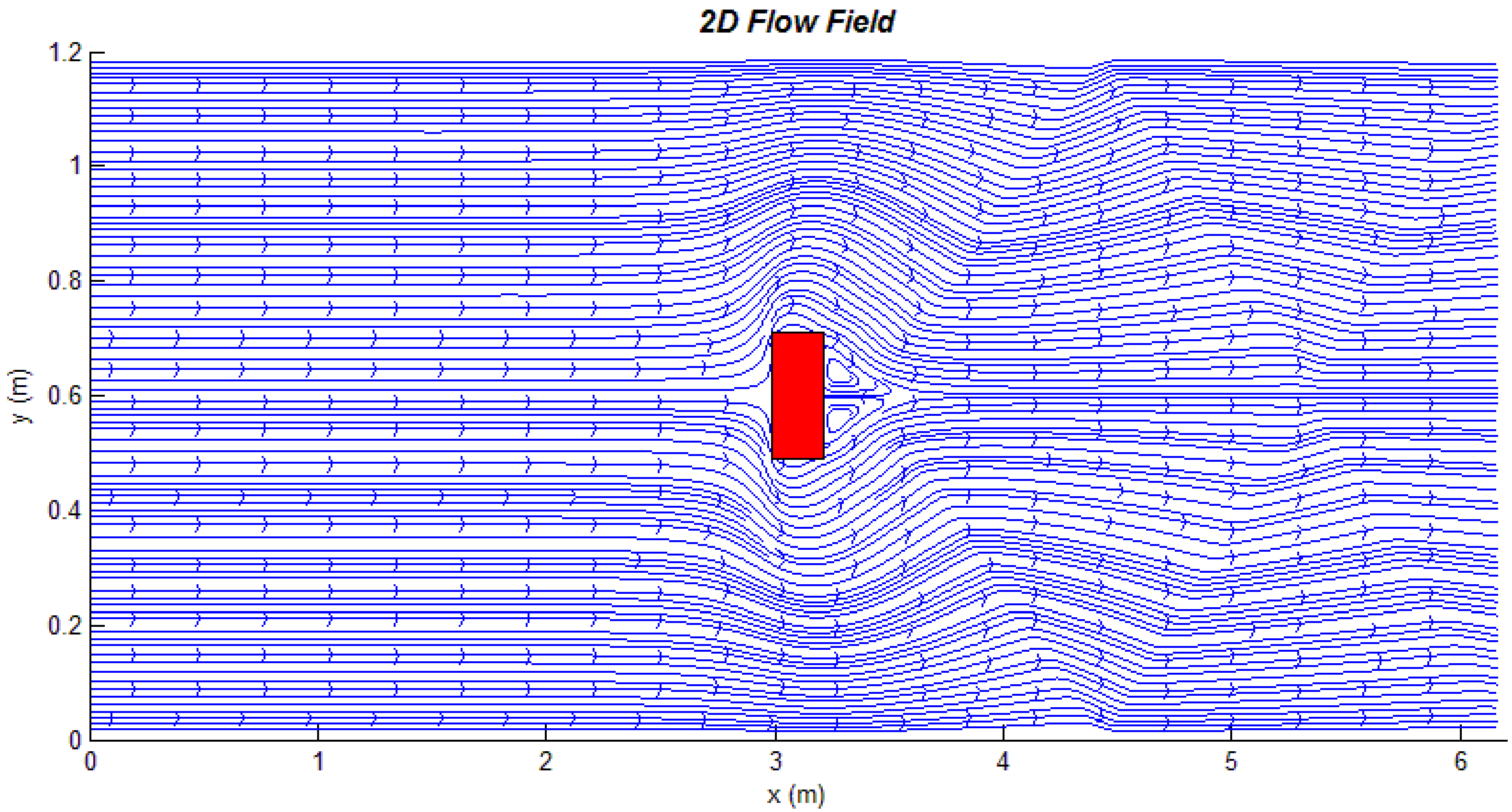

Figure 4.5. 2D Flow field (Supercritical, IT=8000 & mesh size=0.02 m)

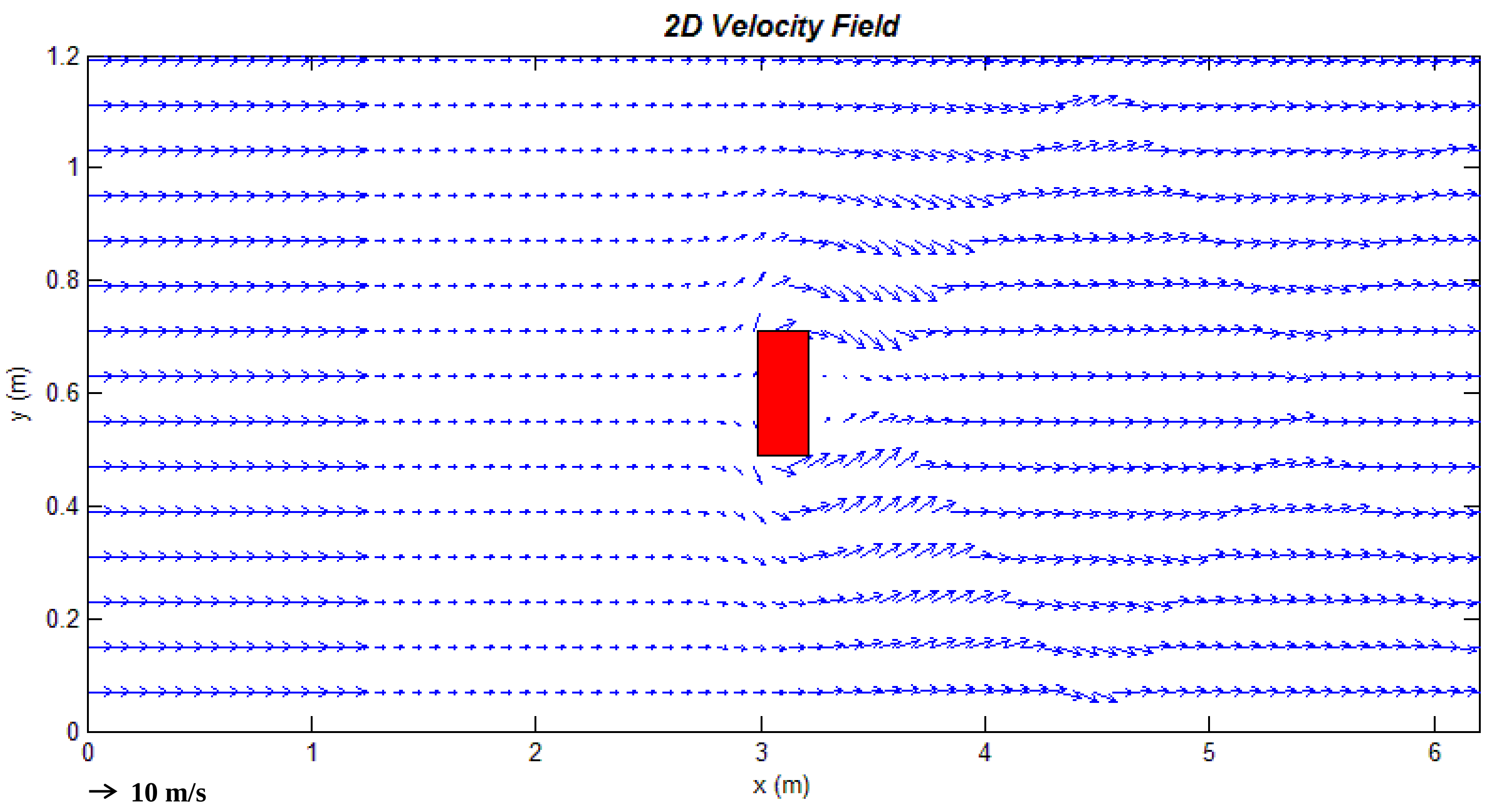

Figure 4.6. 2D Vector field (Supercritical, IT=8000 & mesh size=0.02 m)

However, it is not possible to observe whether the cross waves persist for long distances or disappear since the location of the outflow boundary is very close to the obstruction block.

On the upstream side there is a hydraulic jump occurred in front of the block and therefore, the flow is subcritical between the jump and the block. From the calculated water depths on the sequent depth side, it is understood that, there is a temporary choking and water depth in front of the block is greater than the sequent depth of the hydraulic jump. The streamline patterns around the block and the velocity vector field are shown in Figures 4.5 and 4.6, respectively. The separation vortex behind the block and the crest lines of the cross waves are clearly observed.

Although the residual errors are small and therefore convergence is achieved, iterations are continued and another printout after 12000 iterations is obtained and the corresponding water surface profile is shown in Figure 4.7. An obvious difference is the position of the hydraulic jump which moves in upstream direction. At the same time, the water depth between the jump and the block is decreased by 2 cm. There is no difference between Figure 4.4 and 4.7 at the downstream of the block. There is another printout obtained after 20000 iterations shown in Figure 4.8. The jump starts exactly at the location of the inflow boundary and water depth in front of block decreased further another 2 cm. To continue the iterations to find the equilibrium position of the jump, it is necessary to move the location of the inflow boundary farther upstream. If such a test can be done it will be possible to see that the theoretical value of the sequent depth will be reached between the jump and the block.

Tests done for the supercritical flow show that the numerical solution procedure and the computer code developed are successful in computing discontinuities due to hydraulic jump and the cross waves produced by an obstruction in the flow field. Furthermore, the moving discontinuity due to hydraulic jump with changing location is also reproduced without any oscillations in the flow variables.

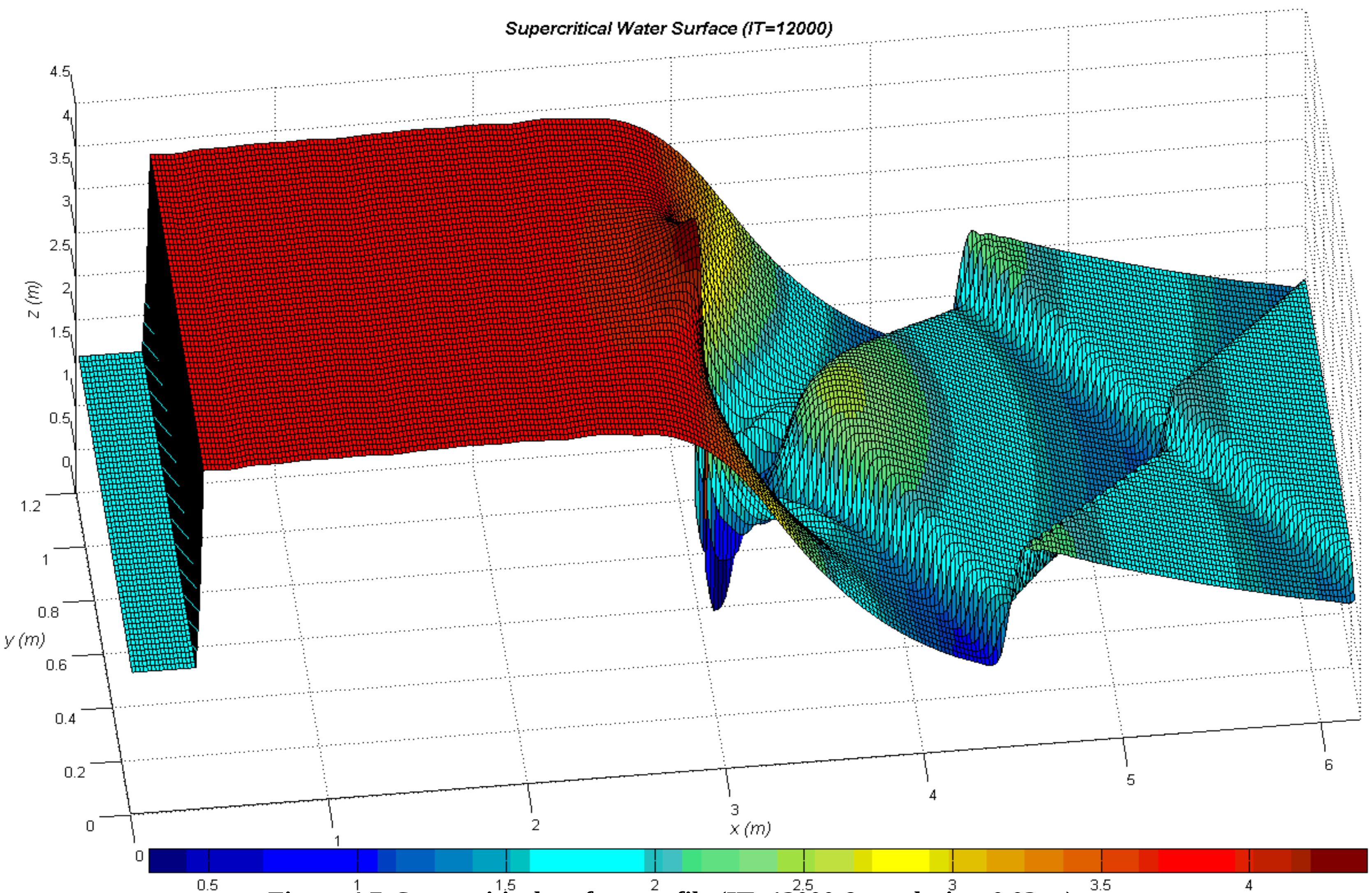

Figure 4.7. Supercritical surface profile (IT=12000 & mesh size=0.02 m)

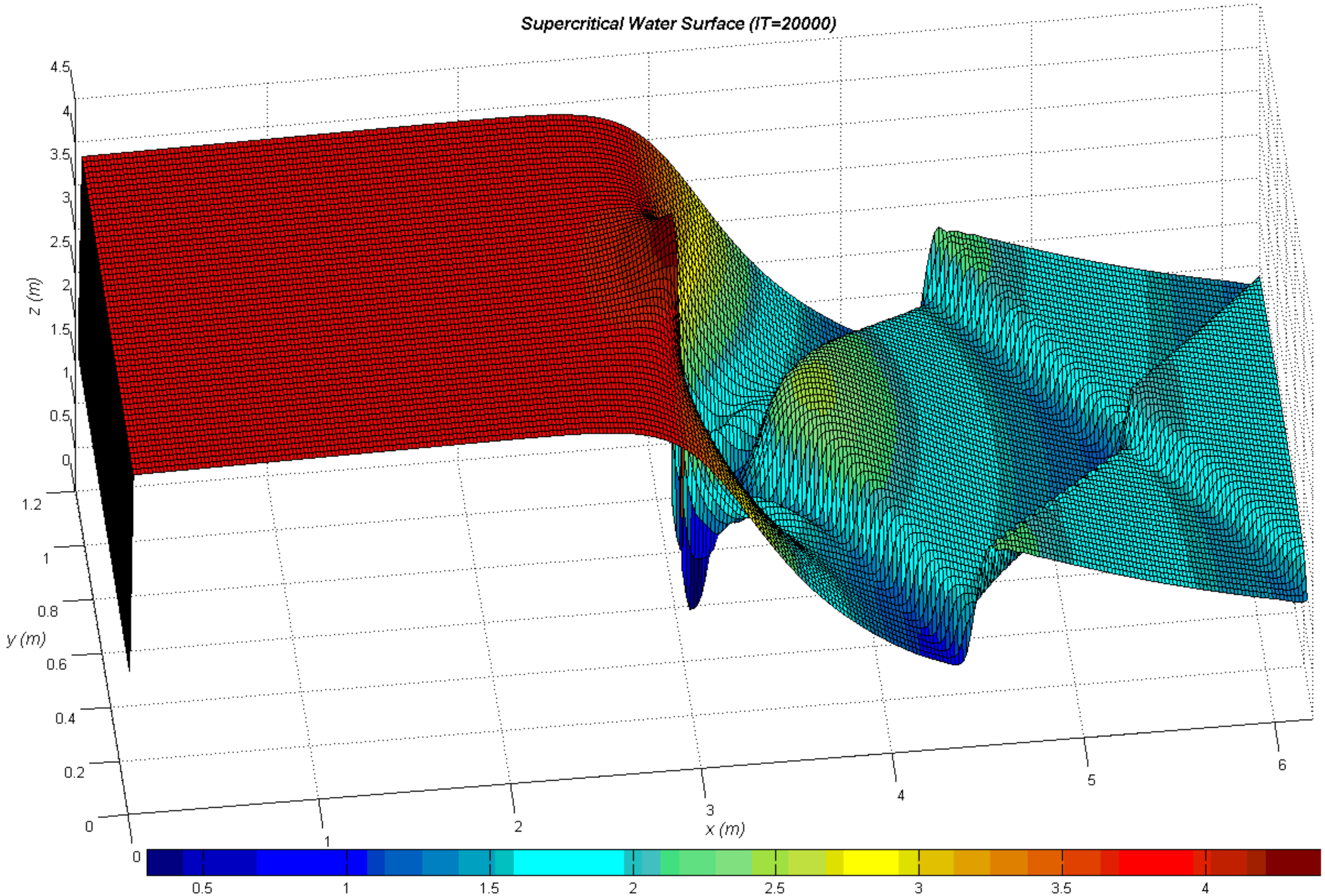

Figure 4.8. Supercritical surface profile (IT=20000 & mesh size=0.02 m)

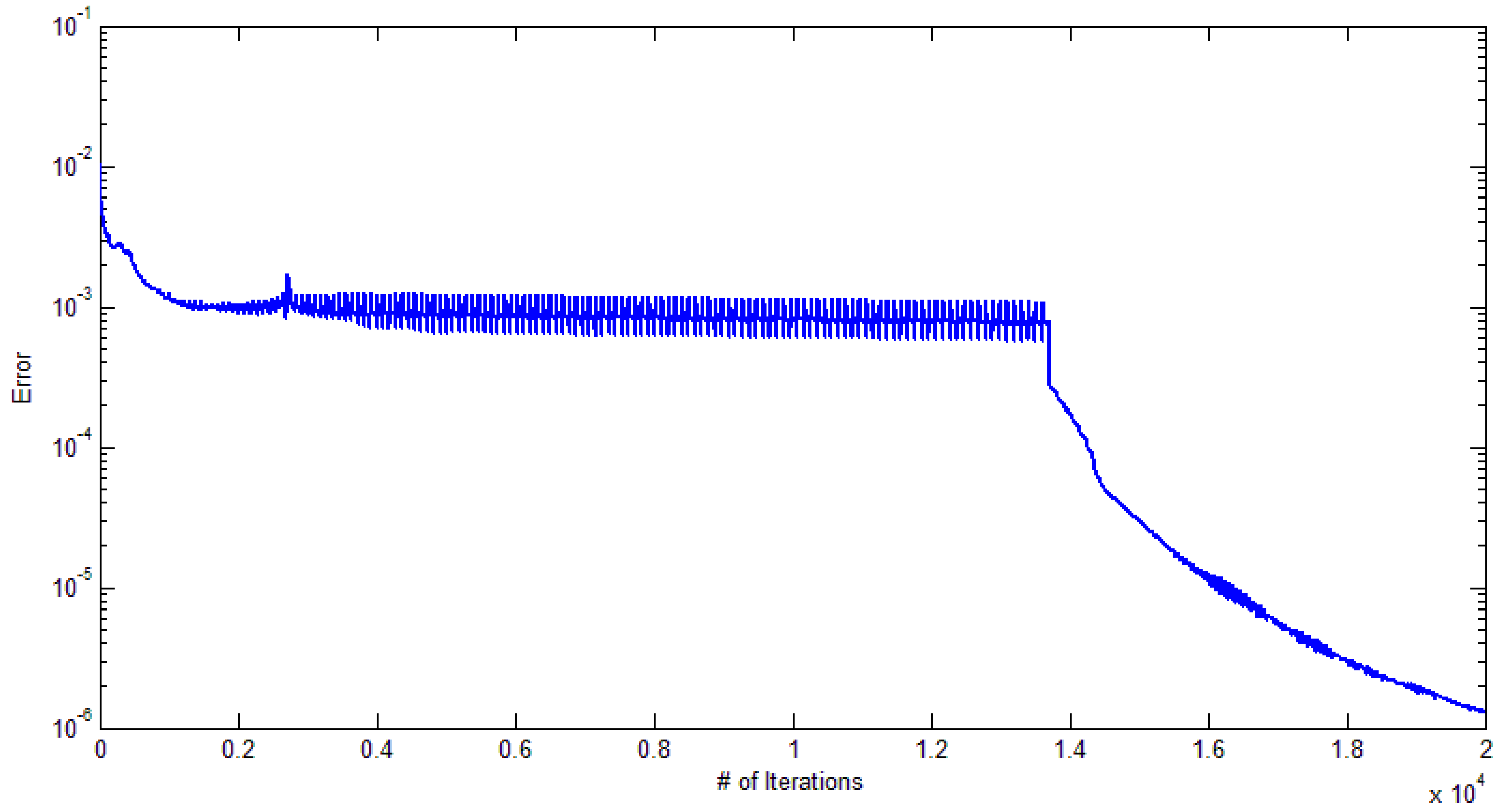

Figure 4.9. Error of supercritical flow case (IT=20000 & mesh size=0.02 m)

In an iterative numerical solution, it is important to see that the changes in dependent variables vanish and no change in further iterations is observed. To observe this feature, the numerical error normalized by uniform flow depth is defined as

$$Error = \frac{\sqrt{\sum\left(h_{I,J}^{n} - h_{I,J}^{n+1}\right)^{2}}}{NC * H_0} \tag{4.1}$$

where *NC* is the number of computational cells in the domain.

The changes in numerical error with the number of iterations is shown in Figure 4.9. At first, there is a rapid decrease in error upto 2000 iterations. Then, the error starts to oscillate around 0.001 upto approximately 14000 iterations. From beginning to 14000 iterations the jump formed in the flow field continuously moves upwards and this moving discontinuity causes the oscillations in error without any permenant decrease. When the jump reaches to inflow boundary, its location is fixed by the inflow boundary condition an the numerical error decreases rapidly to 10^{-6}.

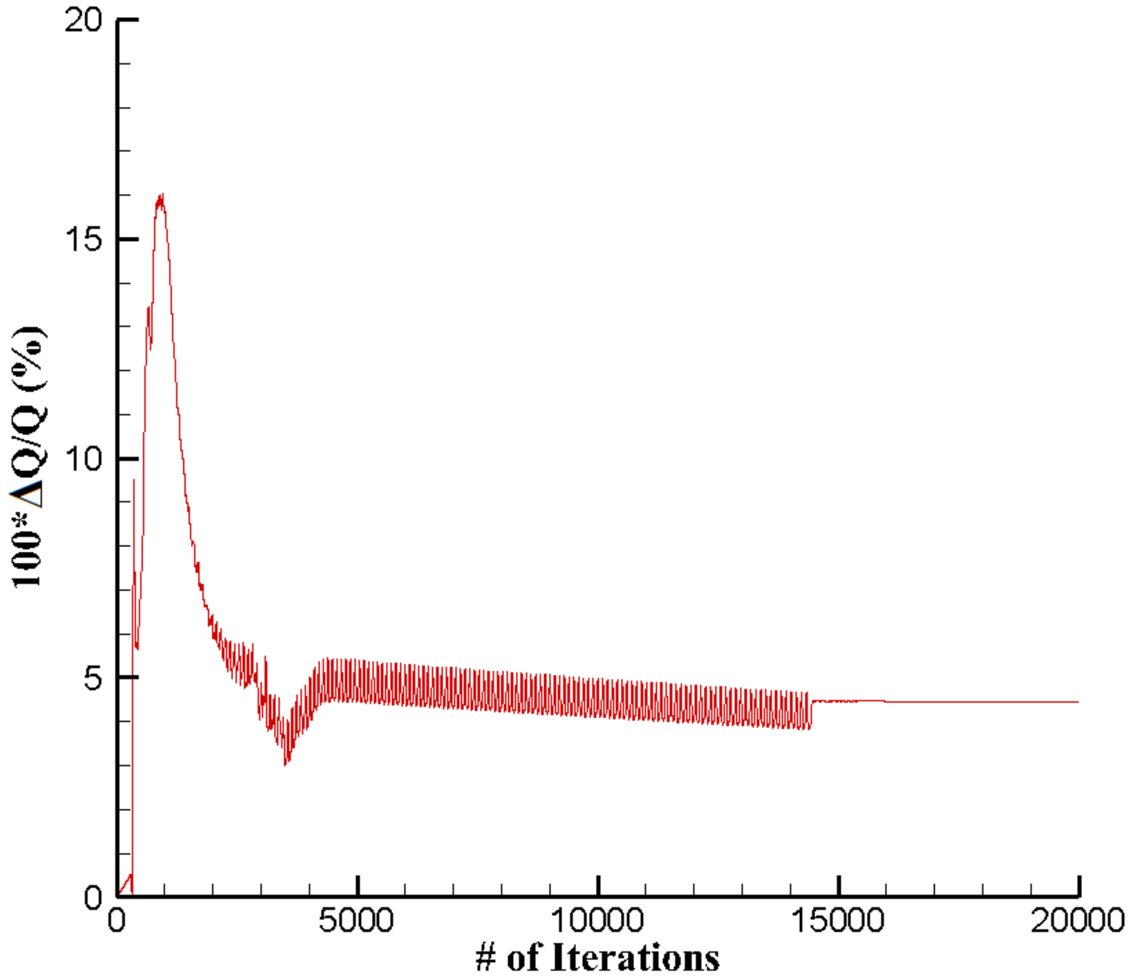

Figure 4.10. The change in discharge with the # of iterations (supercritical flow case & mesh size=0.02m)

Another important feature of a numerical flow computation is the conservativeness of the solution algorithm. In a typical incompressible flow field, the volume flowrates at the inflow and outflow sections should be equal when there are no side in or outflows. The percent difference of discharge at the inflow and outflow boundaries is computed after each iteration step and plotted in Figure 4.10. At early stages of computation the relative error in discharge goes upto 16 %, then decreases rapidly by improved convergence and the final value is about 5 % after 20000 iterations and thenafter remain constant. This rate of error (5 %) in volume flowrate is not negligible and should be resolved. It was not possible wihin the time available for this study to find the source of this error. It is anticipated that the error may be due to incomplete boundary condition applications around the block, especially at the computational cells around the corners of the block.

4.1.2. Subcritical Flow Test Case

To have a subcritical flow, channel bottom slope is decreased and a mild slope is chosen. As in the supercritical flow case, water depth and velocity component in x-direction are initialized to uniform flow conditions. For the subcritical flow test, chosen data is given in Table 4.2.

Table 4.2. Subcritical flow data

Discharge, Q	15 m^3/s
Initial data: uniform flow depth, H_0	3.884 m
Initial data: uniform flow velocity, U_0	3.218 m/s
Initial data: y-direction velocity, V	0 m/s
Channel slope, S_0	0.0030
Froude Number	0.521
State of the flow	Subcritical

SECTION (1) flow data		SECTION (2) flow data	
b_1	1.2 m	b_2	1 m
q_1	12.5 $m^3/s/m$	q_2	15 $m^3/s/m$
E_1	4.412 m	$E_2=E_1$	4.412 m
Ec_1	3.774 m	Ec_2	4.262 m $< E_2$

For the subcritical flows any two of the variables must be specified at the inflow boundary and one variable must be specified at the outflow boundary. In the present study, the unit discharges are fixed at the inflow and water depth, h, is fixed at the outflow. The free surface profile at the end of 20000 iterations is shown in Figure 4.11. There are no discontinuities for this case. Water depth upstream of the block has been increased to nearly 4.5 m from the initial value of 3.88 m. The history of numerical error is shown in Figure 4.12 in which there are no severe oscillations as was the case in supercritical flow with moving hydraulic jump.

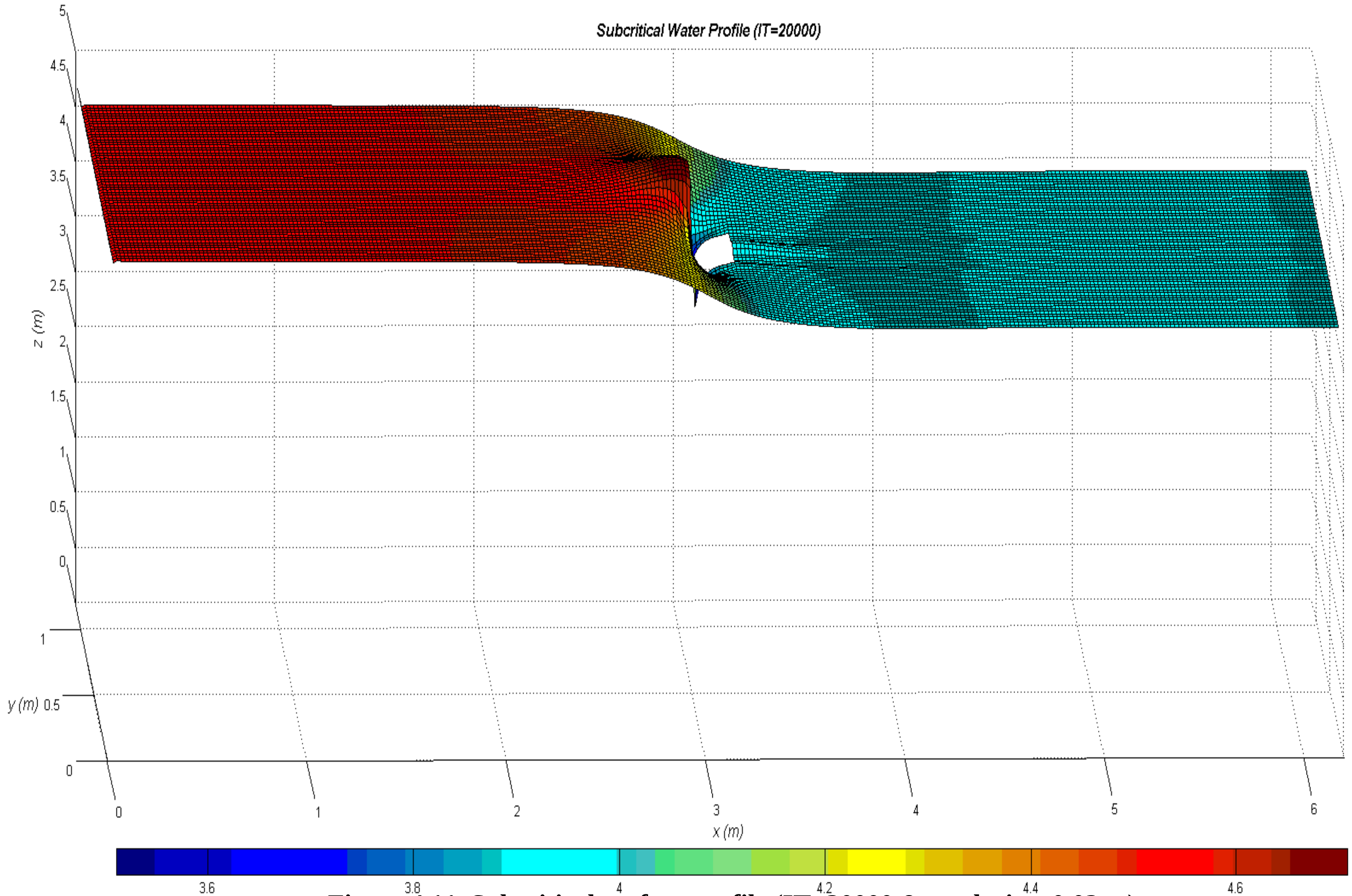

Figure 4.11. Subcritical surface profile (IT=20000 & mesh size=0.02 m)

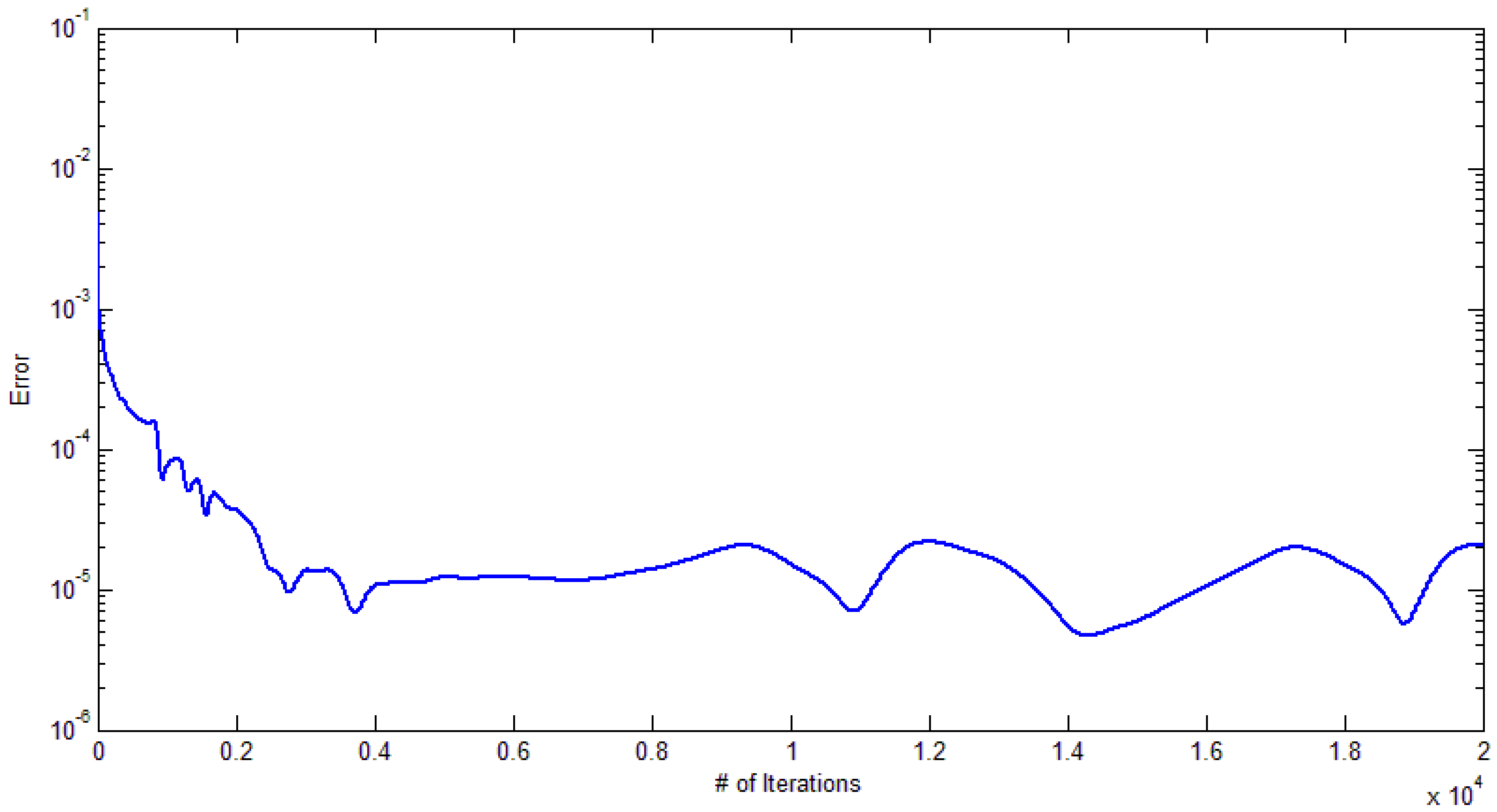

Figure 4.12. Error of subcritical flow case (IT=20000 & mesh size=0.02 m)

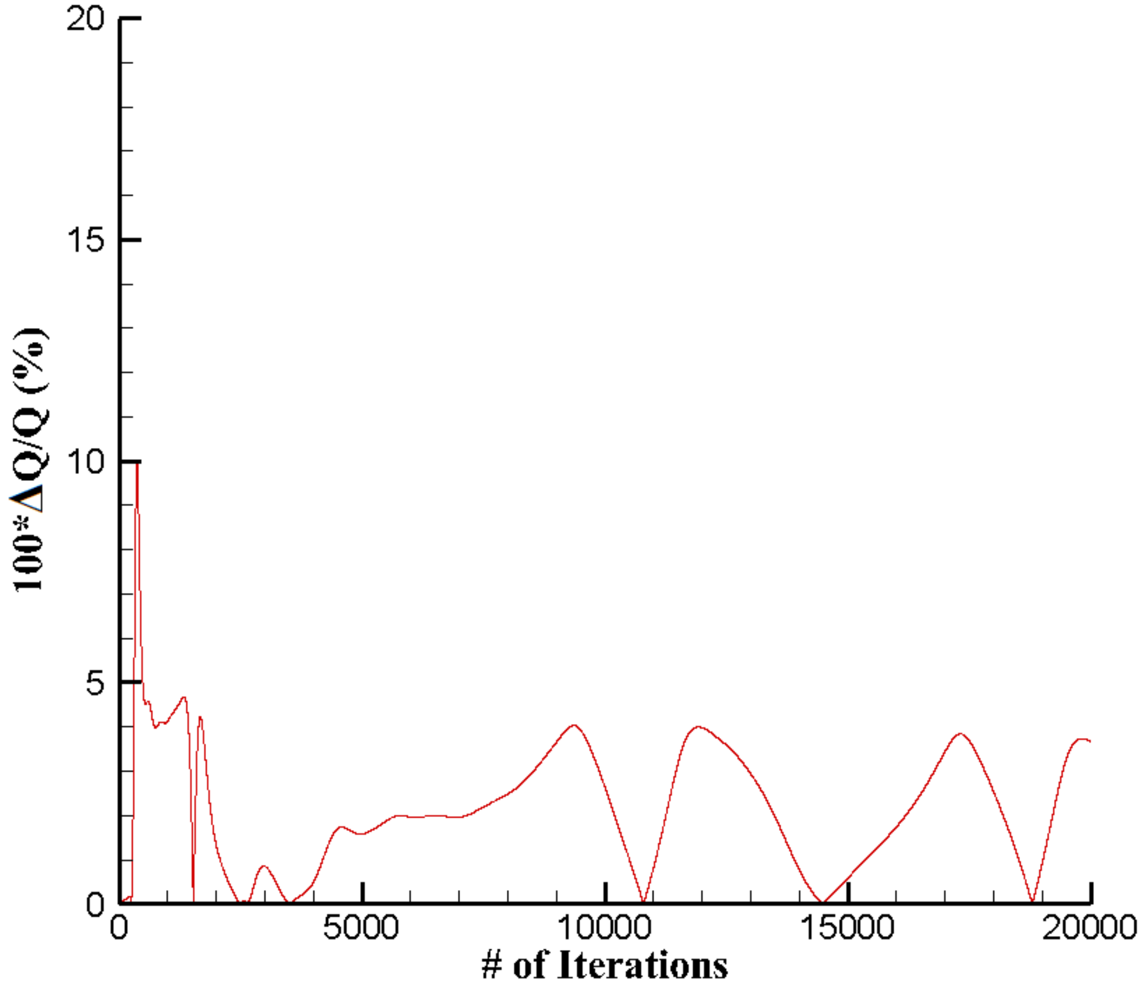

Figure 4.13. The change in discharge with the # of iterations (subcritical flow case & mesh size=0.02m)

The difference of inflow and outflow discharges is shown in percentage of the inflow discharge in Figure 4.13. The final error percentage oscillates between 0 and 4 for subcritical case which was 5 in the supercritical case.

Another case considered is subcritical flow with chocking. As can be seen from the data given in Table 4.3, approaching subcritical flow energy is not adequate to pass the contracted section. Therefore, for the same discharge, an increase in water depth in the upstream is expected to pass the given discharge around the block where critical flow will occur. Boundary conditions at the inflow and outflow are the same as the previous subcritical case. Free surface profile after 20000 iterations is shown in Figure 4.14. The upstream water depth is increased to 4.08 m from the initial uniform flow value of 3.4 m

Table 4.3. Subcritical choked flow data

Discharge, Q	15 m^3/s
Initial data: uniform flow depth, H_0	3.405 m
Initial data: uniform flow velocity, U_0	3.671 m/s
Initial data: y-direction velocity, V	0 m/s
Channel slope, S_0	0.0040
Froude Number	0.635
State of the flow	Subcritical

SECTION (1) flow data		SECTION (2) flow data	
b_1	1.2 m	b_2	1 m
q_1	12.5 $m^3/s/m$	q_2	15 $m^3/s/m$
E_1	4.092m	$E_2=E_1$	4.092 m
Ec_1	3.774 m	Ec_2	4.262 m $> E_2$

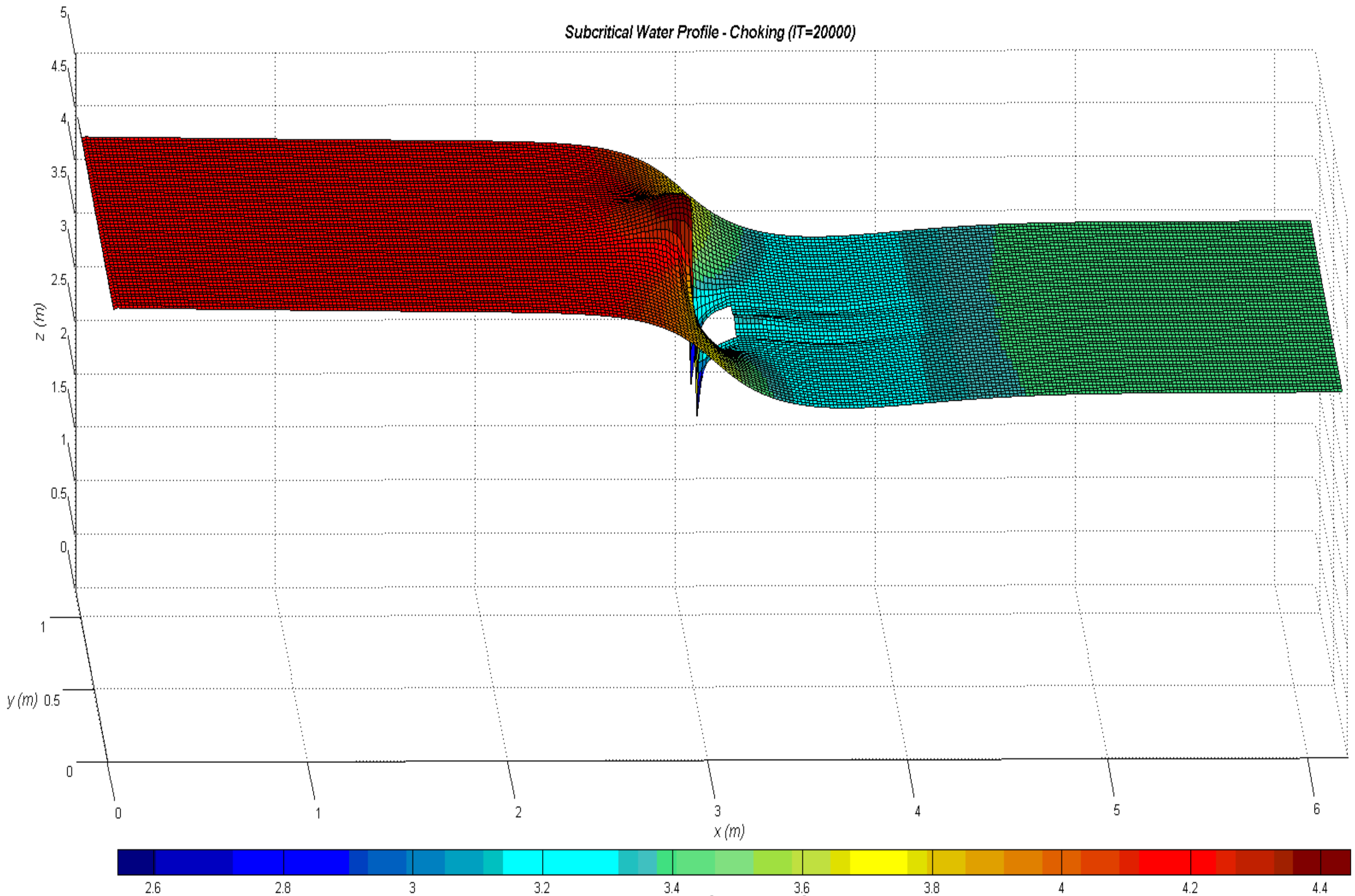

Figure 4.14. Subcritical (choked) surface profile (IT=20000 & mesh size=0.02 m)

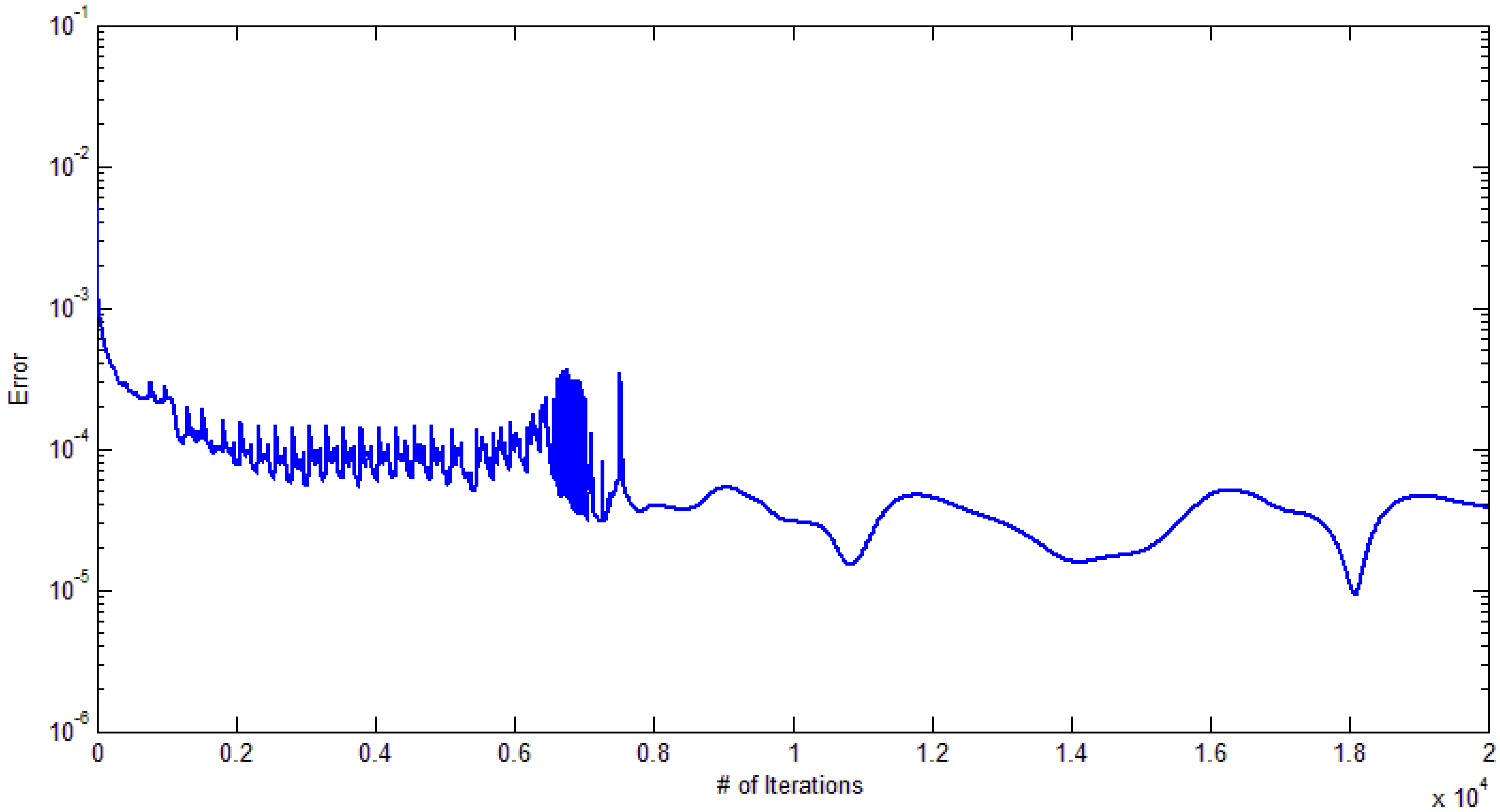

Figure 4.15. Error of subcritical choked flow case (IT=20000 & mesh size=0.02 m)

The numerical error plot of this test case is given in Figure 4.15. Very similar to supercritical case with moving hydraulic jump, there are oscillations in error until the choking process is completed at about 7700 iterations. After completion of choking, high frequency oscillation in error disappeared.

4.1.3. 2D Problem with Periodic Boundary Conditions

Periodic boundary conditions are valid when big number of identical blocks are placed at the same configuration with a constant spacing in the flow direction as shown in Figure 4.16. The periodic boundary condition can be applied on any two sections at a distance equal to spacing between the blocks (CL).

The values of the conserved variables at the first interior row of computational cells of the inflow boundary are copied to ghost cells at the first outside row of the computational cells of outflow boundary. Similarly, the last row of interior cells of outflow boundary are copied to the first row of ghost cells on the upstream side of the inflow boundary. It is also known that fluxes at the periodic boundaries are identical since the inflow into a periodic domain equals the outflow of the previous periodic domain. This equality is also implemented in the solution as part of the periodic the boundary condition.

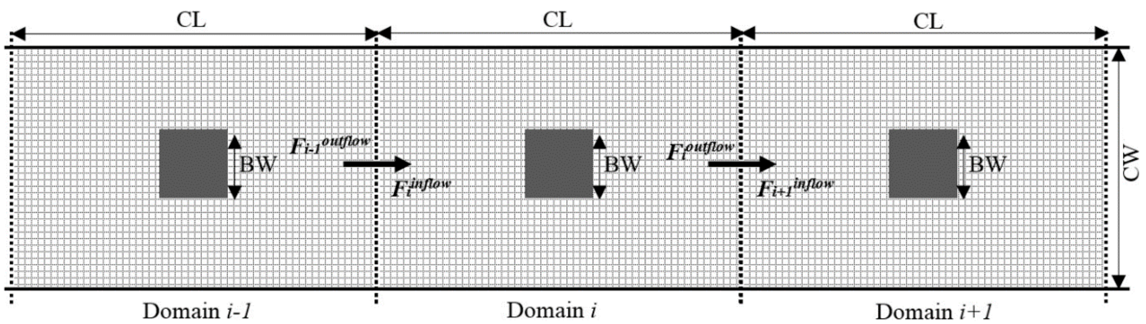

Figure 4.16. Channel with periodic arrangement of the blocks

Although periodic boundary conditions has been applied with many variations in its implication, it was not possible to reach to a steady-state solution. There are no clear descriptions for application of such periodic boundary condition. Lucas (2012) states that the order of the scheme has an effect on how boundary conditions are implemented for periodic domains. First order solutions can be accomplished with periodic boundary conditions but, for the second or higher order solutions further investigations may be necessary to describe a convergent scheme with periodic boundary conditions.

Because of time limitations of this study, the research on the implementation of periodic boundary conditions is not completed yet.

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

Depth-averaged equations for two dimensional shallow flow are derived from the Reynolds Averaged Navier-Stokes Equations. A numerical solution model for the two dimensional depth-averaged equations is developed using the Godunov's method to study free surface flows. The approximate Riemann solver of Roe is adopted to evaluate the intercell fluxes so that the shock capturing property is preserved in the numerical scheme. Finite volume method is used in discretization of the governing equations. MUSCL scheme is employed for interpolation of flux terms to achieve second order accuracy. In the estimation of interface fluxes, slope limiters are applied to gradients of dependent variables to damp out unphysical oscillations in the numerical solution due to second order accuracy.

Flow around a square block fixed in a rectangular prismatic channel is considered to test the solution algorithm and the computer code developed. Several boundary conditions and their implication procedures are reviewed and tested for sub and supercritical flows. Important findings of this study are summarized below.

1) The numerical solution method adopted to solve the depth integrated equations of free surface flow is able to calculate discontinuities in flow variables in 1D and 2D, sub and supercritical flows.
2) In supercritical state, the hydraulic jump in front of the block is successfully computed and its displacement in upstream directions was also simulated.

3) The cross waves involving sharp variations in water depth in the downstream of the block are reproduced successfully without any numerical oscillations.
4) The numerical model can successfully reproduce choking phenomena when appropriate boundary conditions are selected.
5) There is a non-negligible difference in between the inflow and outflow discharges that can be due to inaccuracies introduced in implication of the wall boundary conditions around the block. This problem requires further investigations and improvements in treatment of the wall boundary conditions.
6) It was one of the primary aims of this study to solve the flow around periodic block arrangements. All possible boundary condition treatments applied for this case failed to give a converged solution resulting in continuous change of water depth in the domain. Although it was recommended that the first order methods can handle such domains, no numerical cure for such a false variation was reported in the literature for second order solution algorithms. This is another topic to be investigated further.
7) Several slope limiters available in the literature are implemented in the present code and tested. No significant difference in the results were observed.
8) There are no viscous terms included in the present solution. Only the bottom friction is modeled appropriately using Manning's formula for uniform flow. Viscous terms and turbulent stresses may be included in a future study to better simulate the energy losses and thus satisfy the energy conservation.

REFERENCES

Alcrudo, F. and Garcia-Navarro, P. (1993). *A High-Resolution Godunov Type Scheme in Finite Volumes for the 2D Shallow Water Equations*, International Journal for Numerical Methods in Fluids, Volume 16, Pages 489-505.

Anastasiou, K. and Chan, C. T. (1997). *Solution of the 2D Shallow Water Equations Using Finite Volume Method on Unstructured Triangular Meshes*, International Journal for Numerical Methods in Fluids, Volume 24, Pages 1225–1245.

Brufau, P. and Garcia-Navarro, P. (2000). *Two-Dimensional Dam Break Flow Simulation*, International Journal for Numerical Methods in Fluids, Volume 33, Pages 35–57.

Brufau, P., Vazquez-Cendon, M.E., Garcia-Navarro, P. (2002). *A Numerical Model for the Flooding and Drying of Irregular Domains*, International Journal for Numerical Methods in Fluids, Volume 39, Pages 247–275.

Dadone, A. and Grossman, B. (1994). *Surface Boundary Conditions for the Numerical Solution of the Euler Equations*, AIAA Journal, Volume 32, No. 2, Pages 285-293.

Flanders, H. (1973). *Differentiation Under the Integral Sign*, American Mathematical Monthly, Volume 80(6), Pages 615–627.

George, D.L. (2004). *Numerical Approximation of the Nonlinear Shallow Water Equations with Topography and Dry Beds: A Godunov Type Scheme*, Master of Science Thesis, University of Washington.

Guinot V. (2010). *Wave Propagation in Fluids, Models and Numerical Techniques*, Wiley.

Hirsch, C. (1990), *Numerical Computation of Internal and External Flows (Volume II. Computational Methods for Inviscid and Viscous Flows)*, John Wiley & Sons, Chichester.

LeVeque, R. J. (2002). *Finite Volume Methods for Hyperbolic Problems*, Cambridge University Press.

Lucas C. (2012) *FullSWOF : A Software for Overland Flow Simulation, Advances in Hydroinformatics*, Proceedings SimHydro12, Pages 221-231.

Mohammadian, A., Le Roux, D.Y., Tajrishi, M., Mazaheri, K. (2005). *A Mass Conservative Scheme for Simulating Shallow Flows Over Variable Topographies Using Unstructured Grid*, Advances in Water Resources, Volume 28, Issue 5, Pages 523–539.

Roe, P. L. (1981). *Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes*, Journal of Computational Physics, Volume 43, Pages 357–372.

Stoker, J. J. (1957). *Water Waves, the Mathematical Theory with Applications*, Interscience, London.

Toro, E.F. (2000), *Numerica. A Library of Source Codes for Teaching, Research and Applications*, Numeritek Ltd., Cheshire.

Toro, E.F. (2001), *Shock-Capturing Methods for Free-Surface Shallow Flows*, John Wiley & Sons, Chichester.

Toro, E.F. (2009), *Riemann Solvers and Numerical Methods for Fluid Dynamics. A Practical Introduction*, Springer-Verlag, Berlin, Heidelberg.

Van Leer, B. (1979). *Towards the Ultimate Conservative Difference Scheme*, V. A Second Order Sequel to Godunov's Method, Journal of Computational Physics, Volume 32, Pages 101–136.

Vazquez-Cendon, M.E. (1999). *Improved Treatment of Source Terms in Upwind*

Schemes for the Shallow Water Equations in Channels with Irregular Geometry, Journal of Computational Physics, Volume 148, Pages 497–526.

Versteeg, H. K., Malalasekera, W. (2007). *An Introduction to Computational Fluid Dynamics - The Finite Volume Method*, Pearson, Prentice Hall.

Wu, C., Huang, G., and Zheng, Y. (1999). *Theoretical Solution of Dam Break Shock Wave*, Journal of Hydraulic Engineering, Volume 125(11), Pages 1210–1215.

Zoppou, C. and Roberts, S. (2003). *Explicit Schemes for Dam-Break Simulations*, Journal of Hydraulic Engineering, Volume 129(1), Pages 11–34.