University of Alberta

Depth Averaged and RANS Modeling of Open Channel Flow

by

A T M Hasan Zobeyer

A thesis submitted to the Faculty of Graduate Studies and Research

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Water Resources Engineering

Department of Civil and Environmental Engineering

Spring, 2012

Edmonton, Alberta

ABSTRACT

This study focuses on two issues of open channel flow modeling. First the convergence behavior of preconditioned Krylov subspace methods: Generalized Minimal Residual (GMRES) and Bi-Conjugate Gradient Stabilized (BiCGSTAB) is investigated for the open channel flow model River2D with Jacobi, Symmetric Gauss-Seidel (SGS) and Incomplete Lower Upper (ILU) factorizations with different levels of fill as preconditioners. A novel technique is developed where a matrix obtained from a lower time step than the simulation time step is used to compute the ILU factors. This technique increases the robustness and efficiency of the ILU preconditioners significantly. Applying this technique to ILU, the performance of preconditioned GMRES and BiCGSTAB are compared. In most cases ILU with no fill is found to be the most efficient preconditioner. A test to investigate the effect of mesh refinement on the convergence of the new ILU preconditioned solvers also shows promising results.

The second issue concerns with a coupled depth averaged (DA) and Reynolds averaged Navier-Stokes (RANS) model developed for open channel flow with or without the assumption of hydrostatic pressure. Initially the water surface and DA velocity are estimated by a DA model neglecting non-uniform velocity and non-hydrostatic pressure. Then in the RANS model, the horizontal momentum and the continuity equations are solved for the horizontal and vertical velocity respectively with the water surface as a fixed zero pressure boundary. For

the non-hydrostatic RANS model the pressure Poisson equation is solved for the non-hydrostatic pressure. A correction term is introduced in the RANS horizontal momentum equations for mass balance. Once the RANS model results are available, DA model results can be updated iteratively by incorporating the effects of non-uniform velocity and non-hydrostatic pressure. First the model is developed for two dimensional plane flow and verified against the experimental results of flow development, flow over a symmetric hump and a dune with good results and excellent computational efficiency. Then the model is extended and tested for three dimensional flow with promising results.

Acknowledgement

I would like to express my sincere gratitude to my supervisor Dr. Peter Steffler for his excellent supervision, innovative ideas, patience and continuous encouragement throughout this study. I would also like to thank the examination committee members for their comments and suggestions.

I am grateful to the University of Alberta Graduate Assistantship program, Provost Entrance Scholarships and the Queen Elizabeth II Doctoral Scholarship for providing financial assistance for this study. Financial support from the Natural Sciences and Engineering Research Council of Canada is gratefully acknowledged.

I am very thankful to Dr. Karen Dow for providing me the River2D model of the North Saskatchewan river. I am also thankful to Dr. Jose Vasquez of Northwest Hydraulic Consultants and Dr. Haitham Ghamry of Fisheries and Oceans, Canada for their help.

I would also like to convey my gratefulness to my parents and my wife Nasreen Jahan for their support and encouragements. Special thanks to my son Arish for his love which made every moment of my live joyful and enthusiastic.

Above all, I am thankful to Allah for giving me this excellent opportunity.

TABLE OF CONTENTS

List of Tables

List of Figures

List of Symbols

A =Jacobian matrix;

B = error matrix;

b = half width of curved channel.

C = preconditioning matrix;

c = wave celerity;

CFD = computational fluid dynamics;

CPU = central processing unit;

Cr = Courant number;

C_w = computational work;

C_* = dimensionless Chezy Coefficient;

d = distance from bed;

D = diameter of hemisphere;

Di = diagonal of a matrix;

E = strict lower part of a matrix;

F = strict upper part of a matrix;

g = acceleration due to gravity;

h = water depth;

H = water surface elevation;

k_s = bed roughness height;

K_m = Krylov subspace;

l = level of fill;

L_m = any subspace;

l_m= Prandtl's mixing length;

M = mass matrix,

N = number of iteration of the DA-RANS model;

n = number of time step;

P = instantaneous pressure;

p = pressure;

p_{hs} = hydrostatic pressure;

p^p = non-hydrostatic pressure;

q_x, q_x = discharge per unit width;

R = residual vector;

S = aquifer storativity,

T = aquifer transmissivity.

t = time;

U,V,W = instantaneous point velocity in the x, y and z directions;

u,v,w = time averaged point velocity in the x, y and z directions;

u', v', w' = fluctuating velocity;

$\bar{u}, \bar{v}, \bar{w}$ = depth averaged velocity;

u^p, v^p, w^p = perturbation velocity;

U_c, V_c, W_c = contravariant velocity components;

$\overline{U_c}, \overline{V_c}$ = depth averaged contravariant velocity components;

X, Y = correction terms;

Y_R = radial distance from the centerline of a bend;

x, y, z = Cartesian coordinates;

z_b = normal distance from bed;

β = exponent;

Δt = time step;

ρ = density of water;

κ = von Karman constant;

υ_t = turbulent eddy viscosity;

υ_t^d = depth averaged eddy viscosity;

ξ, η, ζ = curvilinear coordinates;

$\xi_x, \xi_y, \xi_z, \eta_x, \eta_y, \eta_z, \zeta_x, \zeta_x, \zeta_x$ = matrices of transformation;

$\sigma_x^v, \sigma_y^v, \sigma_z^v$ = viscous normal stress;

$\tau_{xy}^v, \tau_{yz}^v, \tau_{xz}^v$ = viscous shear stress;

$\sigma_x, \sigma_y, \sigma_z$ = turbulent normal stress;

$\tau_{xy}, \tau_{yz}, \tau_{xz}$ = turbulent shear stress;

Subscripts d and overbar denote depth averaged and prime denotes perturbation quantities.

Chapter 1

INTRODUCTION

Throughout history, human civilization has evolved around rivers. People use river water in different ways such as navigation, water withdrawal for domestic, commercial and industrial purposes, dumping of pollutant and hydropower generation. Man-made development activities often change the river flow, morphology and water quality seriously affecting the biological habitat and in some cases destroying some species completely. An optimum balance among these development activities, cost and environmental concerns requires a thorough understanding of the hydrodynamics related to contaminant transport, design of hydraulic structures, sediment transport and channel geomorphology.

In the early ages, development activities were based on the knowledge on hydraulics acquired through visual observation and mostly trial and error basis. In the last few centuries people started to understand the hydraulics in a more scientific way by developing mathematical expressions and performing flow measurements. Until the last century mathematical equation were difficult to solve except for very simplified cases. Flow measurements provided the basic knowledge to develop some empirical relationships useful for engineers.

In the last century, the advancement in flow measurement was very rapid with the invention of new measuring instruments and computers. Because rivers are very large measurements are generally limited to ‘physical models’ which are much reduced in scale. The difficulty in representing a river in a physical model is the scale difference in the horizontal and vertical directions in a river. Moreover measurements are costly and time consuming. However measurements are always needed for verifying the results obtained from the solutions of the mathematical equations.

The invention of computer and subsequent increase in power enabled the solution of the partial differential equations describing the flow by numerical methods where the partial differential equations are transformed into discrete equations for different points. Due to the scale issues in a natural river one, two and three dimensional equations were developed and numerical models were classified accordingly. One and two dimensional models are also known as ‘depth averaged’ models as they provide only an average velocity over a depth. Three dimensional models provide variation of the flow variables in the vertical direction in addition to the horizontal directions and are used where vertical variation is important such as flow over a dune, flow in a curved channel and flow around a boulder.

Numerical modeling is preferred due to its flexibility in simulating different scenarios and time and cost effectiveness over experiments. One and two dimensional models that are developed for modeling free surface open channel

flows have been studied extensively. Three dimensional modeling techniques are primarily developed for flows with almost same scale in all directions. Application of 3D models in open channel flow requires additional free surface modeling.

This study will investigate the behavior of iterative solvers which are usually needed to solve the large matrix that arises from the implicit modeling of the 2D depth averaged equations and develop a model by coupling the depth averaged and three dimensional equations with different modeling options for open channel flow.

In chapter 2, the governing equations for 3D flow known as the Reynolds averaged Navier-Stokes (RANS) equations and the equations for depth averaged flow also known as the St Venant equations are derived and the modeling techniques of the equations are described including the iterative solvers. Then the present state of open channel flow modeling is discussed. The knowledge gap in this area is identified and overview of this study is presented.

In chapter 3, different potential iterative solvers are presented and tested for the 2D depth averaged model River2D. A new technique to improve the solvers is also tested and the performance of different solvers is compared. The effect of mesh density on performance of the solvers is presented.

In chapter 4, the coupled model is developed and tested for two dimensional plane flow. The 2D RANS equations and the 1D depth averaged

equations are presented and the relations between the sets of equations are discussed. The modeling techniques for the coupled model are presented. The model is validated by simulating flow development in a rectangular channel, flow over a symmetric hump and flow over a dune.

In chapter 5, the coupled model is extended for 3D flow and validated by simulating the flow in curved open channel and the flow around a hemisphere. Also the model is applied in a natural river to test its computational efficiency.

In Chapter 6, conclusions of this study and recommendations for future work are presented.

Chapter 2

OPEN CHANNEL FLOW MODELING

2.1 INTRODUCTION

The flow in open channels is three dimensional (3D). However the models developed and used for open channel flow can broadly be classified as depth averaged or 3D models. The 3D models are based on the solution of the Reynolds averaged Navier-Stokes (RANS) equations while the depth averaged models solve the classical St. Venant equations. Another class of model exists that can be classified as the quasi-3D models. They are generally improved versions of the depth averaged models. In this chapter the model equations and modeling techniques are presented. Then the present state of open channel flow modeling is discussed and the rationale and objectives of the present study are pointed out.

2.2 THREE DIMENSIONAL FLOW EQUATIONS

The state of a fluid flow is described by the macroscopic properties such as velocity, pressure, density and temperature (Versteeg and Malalasekera, 2007). For isothermal condition the temperature is regarded as constant. Then the governing equations of three dimensional isothermal flow of any fluid consist of the continuity and three momentum equations also known as the Navier-Stokes equations. However the numerical modeling of turbulent flow by these equations

is restricted by the smallest turbulence scales. For modeling turbulent flow, these equations are time averaged to derive a new set of equations known as the Reynolds averaged Navier-Stokes (RANS) equations. These equations require modeling of turbulent stresses produced by the time averaging process. Generally fluid density is considered constant in the open channel flow modeling and special techniques are required for pressure modeling. Free surface computation also requires some special attention.

2.2.1 The 3D continuity equation

Figure 2.1 shows a fluid element with fluid inflow and outflow through different faces in the Cartesian coordinate system. The law of mass conservation states that the rate of increase of mass in the fluid element is equal to the net rate of flow of mass into the fluid element. Based on the law the 3D instantaneous continuity equation for incompressible flow can be written as

$$\frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} + \frac{\partial W}{\partial z} = 0 \qquad [2.1]$$

2.2.2 The 3D momentum equations

The 3D momentum equations are derived by applying the Newton's second law of motion in three directions. This law states that the rate of increase of momentum of a fluid particle in any direction is equal to the sum of forces on

the fluid particle in that direction. Figure 2.2 shows a fluid element in the Cartesian coordinate system with all the stress components in the *x* direction. The momentum equations in the *x*, *y* and *z* directions are

$$\frac{\partial U}{\partial t}+U\frac{\partial U}{\partial x}+V\frac{\partial U}{\partial y}+W\frac{\partial U}{\partial z}+\frac{1}{\rho}\frac{\partial P}{\partial x}-\frac{1}{\rho}\left(\frac{\partial \sigma_x^v}{\partial x}+\frac{\partial \tau_{yx}^v}{\partial y}+\frac{\partial \tau_{zx}^v}{\partial z}\right)=0, \quad [2.2]$$

$$\frac{\partial V}{\partial t}+U\frac{\partial V}{\partial x}+V\frac{\partial V}{\partial y}+W\frac{\partial V}{\partial z}+\frac{1}{\rho}\frac{\partial P}{\partial y}-\frac{1}{\rho}\left(\frac{\partial \tau_{xy}^v}{\partial x}+\frac{\partial \sigma_y^v}{\partial y}+\frac{\partial \tau_{zy}^v}{\partial z}\right)=0, \quad [2.3]$$

$$\frac{\partial W}{\partial t}+U\frac{\partial W}{\partial x}+V\frac{\partial W}{\partial y}+W\frac{\partial W}{\partial z}+\frac{1}{\rho}\frac{\partial P}{\partial z}-\frac{1}{\rho}\left(\frac{\partial \tau_{xz}^v}{\partial x}+\frac{\partial \tau_{yz}^v}{\partial y}+\frac{\partial \sigma_z^v}{\partial z}\right)+g=0, \quad [2.4]$$

where *x*, *y* and *z* = coordinate directions; *t* = time; *U*, *V* and *W* = instantaneous velocity components at any point in the *x*, *y* and *z* directions; *P* = the pressure; σ^v = normal stress, τ^v = shear stress, *g* =acceleration due to gravity, ρ = density of water. Superscript *v* indicates that the stresses are due to fluid viscosity.

The above momentum equations are in the non-conservative form. For different applications and derivation of the Reynolds averaged equations, the conservative form can be useful. They can be written in the conservative form using the continuity after some algebraic manipulation. Also in these equations viscous stresses can be related to the velocity gradient and kinematic viscosity by

the Newton's Law of viscosity. After replacing the viscous stress terms by the Newton's Law of viscosity the momentum equations in the conservative form can be written as

$$\frac{\partial U}{\partial t}+\frac{\partial U^2}{\partial x}+\frac{\partial UV}{\partial y}+\frac{\partial UW}{\partial z}+\frac{1}{\rho}\frac{\partial P}{\partial x}-\nu\left(\frac{\partial^2 U}{\partial x^2}+\frac{\partial^2 U}{\partial y^2}+\frac{\partial^2 U}{\partial z^2}\right)=0 \qquad [2.5]$$

$$\frac{\partial V}{\partial t}+\frac{\partial UV}{\partial x}+\frac{\partial V^2}{\partial y}+\frac{\partial VW}{\partial z}+\frac{1}{\rho}\frac{\partial P}{\partial y}-\nu\left(\frac{\partial^2 V}{\partial x^2}+\frac{\partial^2 V}{\partial y^2}+\frac{\partial^2 V}{\partial z^2}\right)=0 \qquad [2.6]$$

$$\frac{\partial W}{\partial t}+\frac{\partial UW}{\partial x}+\frac{\partial VW}{\partial y}+\frac{\partial W^2}{\partial z}+\frac{1}{\rho}\frac{\partial p}{\partial z}-\nu\left(\frac{\partial^2 W}{\partial x^2}+\frac{\partial^2 W}{\partial y^2}+\frac{\partial^2 W}{\partial z^2}\right)+g=0, \qquad [2.7]$$

where υ = kinematic viscosity of fluid. These equations are called the Navier-Stokes equations. If the fluid viscosity is neglected, these equations are called the Euler equations.

2.2.3 Derivation of the Reynolds averaged Navier-Stokes (RANS) equations

Most of flows of engineering importance, especially open channel flows, are turbulent. It is practically impossible to describe such a chaotic flow by the instantaneous continuity and the Navier-Stokes equations. Although turbulent flow exhibits very random nature, Osborne Reynolds suggested that the instantaneous flow variables can be split into a time averaged and a fluctuating

component. For example U can be split into a time averaged u and a fluctuating u' components

$$U = u + u'$$

u is defined as

$$u = \frac{1}{T}\int_{t_0}^{t_0+T} U dt$$

where t_0 = an arbitrary value of t and T =the time interval of time averaging which contains all time scales of turbulence. The time averaged component is also known as primary part of the turbulent flow.

Using similar splitting for V, W and P, substituting them into equations [2.1] and [2.8] to [2.10], and applying the time averaging rules, the following equations can be obtained.

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0, \qquad [2.8]$$

$$\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x} + \frac{\partial uv}{\partial y} + \frac{\partial uw}{\partial z} + \frac{1}{\rho}\frac{\partial p}{\partial x} - \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right) + \frac{\partial \overline{u'^2}}{\partial x} + \frac{\partial \overline{u'v'}}{\partial y} + \frac{\partial \overline{u'w'}}{\partial z} = 0, \qquad [2.9]$$

$$\frac{\partial v}{\partial t}+\frac{\partial uv}{\partial x}+\frac{\partial v^2}{\partial y}+\frac{\partial vw}{\partial z}+\frac{1}{\rho}\frac{\partial p}{\partial y}-\nu\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}+\frac{\partial^2 v}{\partial z^2}\right) + \frac{\partial \overline{u'v'}}{\partial x}+\frac{\partial \overline{v'^2}}{\partial y}+\frac{\partial \overline{v'w'}}{\partial z}=0, \quad [2.10]$$

$$\frac{\partial w}{\partial t}+\frac{\partial uw}{\partial x}+\frac{\partial vw}{\partial y}+\frac{\partial w^2}{\partial z}+\frac{1}{\rho}\frac{\partial p}{\partial z}-\nu\left(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2}+\frac{\partial^2 w}{\partial z^2}\right) + \frac{\partial \overline{u'w'}}{\partial x}+\frac{\partial \overline{v'w'}}{\partial y}+\frac{\partial \overline{w'^2}}{\partial z}+g=0, \quad [2.11]$$

Equations [2.12] to [2.11] are known as the Reynolds averaged Navier-Stokes (RANS) equations. Together with the continuity equation [2.8] they describe the primary part of the turbulent flow.

The averaging process yields six new unknowns $\overline{u'^2}$, $\overline{v'^2}$, $\overline{w'^2}$, $\overline{u'v'}$, $\overline{u'w'}$ and $\overline{v'w'}$. Therefore we have four equations but ten unknowns. This problem is known as the 'closure problem'.

The new terms in the RANS equations basically are the products of the fluctuating velocities and represent the effect of the turbulence on the primary flow. They are called Reynolds stresses as the turbulent fluctuations introduce additional stresses into the fluid flow. Of the Reynolds stresses, three are normal stresses

$$\sigma_x=-\rho\overline{u'^2}\quad \sigma_y=-\rho\overline{v'^2} \text{ and } \sigma_z=-\rho\overline{w'^2}$$

and three are shear stresses

$$\tau_{xy} = \tau_{yx} = -\rho\overline{u'v'},\ \tau_{yz} = \tau_{zy} = -\rho\overline{v'w'} \text{ and } \tau_{zx} = \tau_{xz} = -\rho\overline{u'w'}$$

In engineering flows the viscous stresses are most often neglected as they are very small as compared to the Reynolds stresses. The Reynolds equations in the non-conservative form can be written after neglecting the viscous stress terms as

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} + \frac{1}{\rho}\frac{\partial p}{\partial x} - \frac{1}{\rho}\left(\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z}\right) = 0 \qquad [2.12]$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z} + \frac{1}{\rho}\frac{\partial p}{\partial x} - \frac{1}{\rho}\left(\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} + \frac{\partial \tau_{zy}}{\partial z}\right) = 0 \qquad [2.13]$$

$$\frac{\partial w}{\partial t} + u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z} + \frac{1}{\rho}\frac{\partial p}{\partial z} - \frac{1}{\rho}\left(\frac{\partial \tau_{zx}}{\partial x} + \frac{\partial \tau_{zy}}{\partial y} + \frac{\partial \sigma_z}{\partial z}\right) + g = 0 \qquad [2.14]$$

2.2.4 Boundary conditions

To describe open channel flow using the RANS equations, water surface and bed boundary conditions are required. Referring to Figure 2.3 the water surface kinematic and dynamic boundary conditions are

Kinematic:

$$w_h = \frac{\partial h}{\partial t} + u_h\frac{\partial (h + z_b)}{\partial x} + v_h\frac{\partial (h + z_b)}{\partial y} \qquad [2.15]$$

Dynamic:

$$p_h = \tau_h = \sigma_h = 0, \qquad [2.16]$$

where subscript *h* indicates the quantities at the water surface.

At the bed a 'no slip' condition exist i.e.

$$u_b = v_b = 0 \qquad [2.17]$$

However these conditions are difficult to implement numerically due to the sharp velocity gradient near the channel bed and a slip velocity is allowed at a virtual bed located at the top of the very thin high velocity gradient layer (Steffler and Jin, 1993). Therefore a bed kinematic condition is applied.

$$w_b = u_b \frac{\partial z_b}{\partial x} + v_b \frac{\partial z_b}{\partial y} \qquad [2.18]$$

Bed dynamic conditions at $z = z_b$:

Bed stresses (τ_{bx} , τ_{by}) are tangential to the bed (Vreugdenhill, 1994):

$$\tau_{bx} = -\sigma_x \frac{\partial z_b}{\partial x} - \tau_{xy} \frac{\partial z_b}{\partial y} + \tau_{xz} \qquad [2.19]$$

$$\tau_{by} = -\tau_{xy} \frac{\partial z_b}{\partial x} - \sigma_y \frac{\partial z_b}{\partial y} + \tau_{yz} \qquad [2.20]$$

The bed stress is an unknown and usually is expressed as a function of other variables. Generally the bed stress is computed by the 'law of the wall' by expressing it as a function of the near bed velocity. The universal logarithmic law of the wall can be written as:

$$U^+ = \frac{1}{\kappa}\ln\frac{z_p}{k_s} + A_r \qquad [2.21]$$

$$A_r = \frac{1}{\kappa}\ln(k_s^+) + A \qquad [2.22]$$

where U^+ = dimensionless velocity = $\frac{u_b}{U_*}$; u_b =resultant near bed velocity parallel to the bed; U_* = Shear velocity = $\sqrt{\frac{\tau_b}{\rho}}$; τ_b =bed stress; k_s^+ = dimensionless roughness height = $\frac{k_s U_*}{\nu}$; k_s =bed roughness height; A=5.5;

A_r follows equation [2.22] over a smooth bed, however it deviates from the equation and decreases gradually as k_s^+ increases. For a completely rough bed A_r is equal to 8.5. For modeling purposes the normal distance z_p should be such that $z^+ = \frac{z_p U_*}{\upsilon}$ is within the range from 30 to 100 where the logarithmic law of the wall is valid (Nezu and Nakagawa 1993).

2.3 DEPTH AVERAGED FLOW EQUATIONS

Open channel flows are often described by the depth averaged equations known as the Saint Venant equations. De Saint-Venant (1771) derived the equations for one dimensional flow (Chaudhry, 1993). For two dimensional flow, the set of equations consists of a continuity and two horizontal depth averaged momentum equations and is usually derived by the application of conservations laws to a vertical column of fluid using total flux across each vertical face (Daubert and Graffe, 1967, Van Rijn, 1990, Ashraf et al., 1995). The important assumptions made in this derivation are that vertical distributions of *u* and *v* velocity are uniform and vertical acceleration is negligible thus the pressure distribution is hydrostatic. However the complete depth averaged equations can be derived without these assumptions by integrating the Reynolds equations over the flow depth (Weiyan 1992, Steffler and Jin 1993, Vreugdenhill, 1994). Then these assumptions can be applied to reduce the complete depth averaged equations to the Saint Venant equations.

2.3.1 Derivation of the DA equations from the RANS equations

In deriving the depth averaged equations, vertical profiles of *u, v* and *w* are split into depth averaged ($\bar{u}, \bar{v}, \bar{w}$) and perturbation ($u^p, v^p, w^p$) components while the vertical profile of *p* is split into hydrosatic (p^{hs}) and non-hydrostatic (

p^p) components. The stress terms (σ and τ) are also divided into depth averaged (σ^d and τ^d) and perturbation components (σ^p and τ^p)

$$u = \bar{u} + u^p;$$

$$v = \bar{v} + v^p;$$

$$w = \bar{w} + w^p;$$

$$p = p^{hs} + p^p;$$

$$p^{hs} = \rho g(H - z);$$

$$\sigma = \sigma^d + \sigma^p$$

$$\tau = \tau^d + \tau^p.$$

H= water surface elevation;

After substituting the above decomposition into the continuity equation [2.8], integration from the bed to the water surface and application of the Leibnitz Rule gives

$$\frac{\partial h}{\partial t} + \frac{\partial h\bar{u}}{\partial x} + u_b \frac{\partial z_b}{\partial x} + v_b \frac{\partial z_b}{\partial y} + \frac{\partial h\bar{v}}{\partial y} - u_h \frac{\partial (h + z_b)}{\partial x} - v_h \frac{\partial (h + z_b)}{\partial y} + w_h - w_b = 0$$

$$\left[\begin{array}{l} \textit{Leibnitz Rule :} \\ \\ \int_a^b \frac{\partial f}{\partial x} dz = \frac{\partial}{\partial x}\int_a^b f\, dz + f_a \frac{\partial a}{\partial x} - f_b \frac{\partial b}{\partial x} \end{array}\right]$$

By definition:

$$\int_{z_b}^{h+z_b} \bar{u} dz = \bar{u} h$$

$$\int_{z_b}^{h+z_b} u^p dz = 0 .$$

and similar for *v* and *w*.

Substituting the surface and bed kinematic equations yields the well-known depth averaged continuity or the mass conservation equation in two dimensions;

$$\frac{\partial h}{\partial t} + \frac{\partial h\bar{u}}{\partial x} + \frac{\partial h\bar{v}}{\partial y} = 0 \qquad [2.23]$$

Following the same procedure for the Reynolds momentum equations in the conservative form together with the surface and bed kinematic and dynamic conditions gives the depth averaged momentum equations in *x*, *y* and *z* directions.

$$\frac{\partial h\bar{u}}{\partial t}+\frac{\partial h\bar{u}^{2}}{\partial x}+\frac{\partial h\overline{u^{p}u^{p}}}{\partial x}+\frac{\partial h\bar{u}\bar{v}}{\partial y}+\frac{\partial h\overline{u^{p}v^{p}}}{\partial y}+gh\frac{\partial(h+z_b)}{\partial x}$$
$$+\frac{1}{\rho}\left(\frac{\partial h\overline{p^{p}}}{\partial x}+p_b^{p}\frac{\partial z_b}{\partial x}\right)-\frac{1}{\rho}\left(\frac{\partial h\left(\sigma_x^{d}+\overline{\sigma_x^{p}}\right)}{\partial x}+\frac{\partial h\left(\tau_{xy}^{d}+\overline{\tau_{xy}^{p}}\right)}{\partial y}\right) \qquad [2.24]$$
$$+\frac{1}{\rho}\left(\tau_{bx}^{d}+\tau_{bx}^{p}\right)=0,$$

$$\frac{\partial h\bar{v}}{\partial t}+\frac{\partial h\bar{u}\bar{v}}{\partial x}+\frac{\partial h\overline{u^{p}v^{p}}}{\partial x}+\frac{\partial h\bar{v}^{2}}{\partial y}+\frac{\partial h\overline{v^{p}v^{p}}}{\partial y}+gh\frac{\partial(h+z_b)}{\partial x}$$
$$+\frac{1}{\rho}\left(\frac{\partial h\overline{p^{p}}}{\partial x}+p_b^{p}\frac{\partial z_b}{\partial x}\right)-\frac{1}{\rho}\left(\frac{\partial h\left(\tau_{xy}^{d}+\overline{\tau_{xy}^{p}}\right)}{\partial x}+\frac{\partial h\left(\sigma_y^{d}+\overline{\sigma_y^{p}}\right)}{\partial y}\right) \qquad [2.25]$$
$$+\frac{1}{\rho}\left(\tau_{bx}^{d}+\tau_{bx}^{p}\right)=0,$$

$$\frac{\partial h\bar{w}}{\partial t}+\frac{\partial h\bar{u}\bar{w}}{\partial x}+\frac{\partial h\overline{u^{p}w^{p}}}{\partial x}+\frac{\partial h\bar{v}\bar{w}}{\partial y}+\frac{\partial h\overline{v^{p}w^{p}}}{\partial y}-\frac{1}{\rho}\left(\rho gh+p_b^{p}\right)$$
$$+\frac{1}{\rho}\left(\sigma_{bz}^{d}+\sigma_{bz}^{p}\right)-\frac{1}{\rho}\left(\frac{\partial h\left(\tau_{zx}^{d}+\overline{\tau_{zx}^{p}}\right)}{\partial x}-\left(\tau_{bx}^{d}+\tau_{bx}^{p}\right)\frac{\partial z_b}{\partial x}\right) \qquad [2.26]$$
$$-\frac{1}{\rho}\left(\frac{\partial h\left(\tau_{yz}^{d}+\overline{\tau_{yz}^{p}}\right)}{\partial y}-\left(\tau_{by}^{d}+\tau_{by}^{p}\right)\frac{\partial z_b}{\partial y}\right)+gh=0,$$

where overbar denotes depth averaged quantity.

The above depth averaged equations can provide water depth and three depth averaged velocities as accurately as a 3D RANS model if the perturbation terms are modeled perfectly.

2.3.2 Deduction of the St Venant Equations

Neglecting the primed terms, vertical acceleration and stress terms in equation [2.26] the hydrostatic pressure condition can be deduced. Neglecting the perturbation terms, equations [2.24] and [2.25] can be reduced to the classical St. Venant equations.

Continuity equation

$$\frac{\partial h}{\partial t}+\frac{\partial h\bar{u}}{\partial x}+\frac{\partial h\bar{v}}{\partial y}=0 \qquad [2.27]$$

St Venant-*x* momentum equation

$$\frac{\partial h\bar{u}}{\partial t}+\frac{\partial h\bar{u}^{2}}{\partial x}+\frac{\partial h\bar{u}\bar{v}}{\partial y}++gh\frac{\partial\left(h+z_b\right)}{\partial x} -\frac{1}{\rho}\left(\frac{\partial h\sigma_x^d}{\partial x}+\frac{\partial h\tau_{xy}^d}{\partial y}\right)+\frac{\tau_{bx}^d}{\rho}=0, \qquad [2.28]$$

St Venant- *y* momentum equation

$$\frac{\partial h\bar{v}}{\partial t}+\frac{\partial h\bar{u}\bar{v}}{\partial x}+\frac{\partial h\bar{v}^{2}}{\partial y}+gh\frac{\partial(h+z_b)}{\partial x} -\frac{1}{\rho}\left(\frac{\partial h\tau_{xy}^d}{\partial x}+\frac{\partial h\sigma_y^d}{\partial y}\right)+\frac{\tau_{by}^d}{\rho}=0, \qquad [2.29]$$

In all the equations presented so far the effect of wind stress and Coriolis forces are considered negligible which is generally valid for modeling flows in small domains such as rivers and streams. They may become important for large lakes and estuaries (Bertin, 1987).

The depth averaged equations are essentially hyperbolic in nature since the horizontal turbulent stress gradients are relatively small. For subcritical flow, one characteristic enters the domain through each of upstream and downstream boundaries therefore one boundary condition is needed in each of the boundaries. For supercritical flow two characteristics enter the domain through the upstream boundary requiring two boundary conditions.

2.4 MODELING OF THE RANS EQUATIONS

The RANS equations do not have any closed form solution except for very simplified cases; therefore numerical modeling techniques have been developed and implemented. Generally the three momentum equations provide three velocity components in their respective directions and we are left only with the continuity equation and the pressure. However it is interesting to note that there is no pressure term in the continuity equation. Therefore some special techniques have been developed for modeling the pressure. The discretization of the convective terms (terms with the first order derivative of the velocity components) in the

momentum equations requires particular attention regarding the direction of the flow. Additional models are required for turbulence modeling.

2.4.1 Discretization of the convective terms

The RANS momentum equations are generally convection dominated as the relative strength of diffusion from turbulent stresses is less than the convective terms. In any convection dominated flow, the property of fluid at any location is influenced by the fluid at the upstream location where the fluid is coming from. The discretization of the convective terms in the RANS momentum equations by a second order accurate center difference method usually leads to model instability as it does not consider the direction of flow. An upwind discretization is used where the influence of the direction of flow is considered. The simplest upwind method is the backward or forward difference methods depending on whether the fluid flow is in the positive or negative direction. This method is first order accurate and is known as the first order upwind method. A Taylor series expansion for this method shows that the leading truncated term contains a second derivative which is diffusive in nature. Therefore first order upwind methods may be very diffusive and diffusion increases with an increase in mesh size. This numerical diffusion may smear out sharp changes in the solution and provides stability of the numerical solution similar to physical viscosity.

The hybrid differencing method (Spadling, 1972) is a blend of first order upwind and centered differencing schemes. Depending on the Peclet number

which is a measure of relative strength of convection and diffusion, this method switches between the schemes. The power law (Patankar, 1980) is another hybrid method where a polynomial is used instead of the centered difference. However they are first order accurate based on the Taylor series expansion (Versteeg and Malalasekera, 2007).

Higher order discretization methods have been developed to minimize the numerical diffusion caused by the first order methods by involving more neighboring points and taking a wider influence. One of the widely used higher order upwind methods is the QUICK (Quadratic upstream interpolation for convective kinetics) scheme of Leonard (1979). This method uses 3 neighboring nodes, two from upstream and one from downstream in discretizing a convective term. In a regular grid this method gives third order accuracy. The higher order upwind methods can generally be considered as a first order method plus additional convection information which reduces the numerical diffusion caused by the first order method. However higher order discretization methods may produce oscillation especially near steep gradient of the flow variable. Therefore TVD (total variation diminishing) schemes (Harten, 1983) have been formulated to achieve oscillation free higher order accuracy which is attained by using limiter functions to limit the range of additional information added to the first order method for higher order accuracy. Different limiter functions have been developed such as Van Leer (Van Leer, 1974), Van Albada (Van Albada et al., 1982), Min-Mod (Roe, 1985) SUPERBEE (Roe, 1985), QUICK (Leonard, 1988)

and UMIST (Lien and Leschziner, 1993). These limiters provide second order accuracy.

2.4.2 Turbulence modeling

While information on the primary flow is needed for engineering applications, the effects of the turbulence on the primary flow are considered through the modeling of the Reynolds stress terms. Significant research has been devoted to develop turbulence models in order to calculate the Reynolds stresses and many models are now available (Rodi, 1984). The turbulence models currently used can be classified as the zero equation, one equation, two equations and very complicated seven equations Reynolds Stress models.

Of the turbulence models the zero to two equation models are generally used and are developed based on the Boussinesq hypothesis that relates the Reynolds stresses to the primary velocity gradients. Using an analogy to Newton's law of viscosity, Boussinesq proposed that the Reynolds stresses are proportional to the mean rates of deformation.

In a general form this can be expressed as

$$\tau_{ij} = -\rho\overline{u_i' u_j'} = \mu_t\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - \frac{2}{3}\rho k\delta_{ij} \qquad [2.30]$$

where *i, j* represents the coordinate directions, δ = Kronecker delta. *k* is the turbulent kinetic energy per unit mass.

Using this hypothesis the turbulent or eddy viscosity parameter μ_t or the kinematic eddy viscosity $\upsilon_t = \frac{\mu_t}{\rho}$ is calculated.

Zero equation models can be classified into following three categories (Vreugdenhil, 1994).

a) Constant eddy viscosity model:

In this model, the eddy viscosity is constant along the depth and sometimes related to the bed shear stress.

$$\upsilon_t = c_0 u_* h \qquad [2.31]$$

where u_* = shear velocity. c_0 = a constant. Therefore by this model, the eddy viscosity may vary in the horizontal directions.

b) Specified vertical profile of eddy viscosity model

In this model, the eddy viscosity is related to the bed stress and varies over the depth according to any specified vertical distribution.

$$\upsilon_t = 0.2\kappa u_* h \;\; for \;\; (z - z_b > 0.2h)$$
$$\upsilon_t = \kappa u_* (z - z_b) \quad for \;\; (z - z_b < 0.2h) \qquad [2.32]$$

This model gives a linear profile near the bed and a constant value above. Another model is (Shimizu et al, 1990):

$$\upsilon_t = \kappa u_* (z - z_b)\left(1 - \frac{z - z_b}{h}\right) \qquad [2.33]$$

Mixing length model

Prandtl (1925) suggested the first model to describe the distribution of eddy viscosity. In this model eddy viscosity is assumed to be proportional to the product of a velocity scale and a mixing length (l_m) (Rodi, 1984). Considering only one velocity gradient in the *z* direction $\left(\frac{\partial u}{\partial z}\right)$ and one turbulent stress (τ_{xz}) in shear flow, the velocity scale is taken as the product of the mean velocity gradient times the mixing length. Therefore the eddy viscosity can be related as

$$\upsilon_t = l_m^2 \left|\frac{\partial u}{\partial z}\right| \qquad [2.34]$$

For general flow with velocity gradients $\left(\frac{\partial u}{\partial z}\right)$ and $\left(\frac{\partial v}{\partial z}\right)$, the hypothesis is written as (Rodi, 1984)

$$\upsilon_t = l_m^2 \sqrt{\left(\frac{\partial u}{\partial z}\right)^2 + \left(\frac{\partial v}{\partial z}\right)^2} \qquad [2.35]$$

Different forms of l_m are available mainly based on experimental results of different types of flows. For open channel flow, following mixing length equations are available.

$$l_m = \kappa(z - z_b)\sqrt{\left(1 - \frac{z - z_b}{h}\right)} \qquad [2.36]$$

(Rajaratnam, 1987)

$$l_m = h\left(0.14 - 0.08\left(1 - \frac{z - z_b}{h}\right)^2 - 0.06\left(1 - \frac{z - z_b}{h}\right)^4\right) \qquad [2.37]$$

(Rodi, 1984)

Mixing length models are easy to implement and do not require additional computational efforts as no additional transport equation needs to be solved. The main disadvantage is that they only consider local flow conditions and ignore any transport of turbulence and therefore incapable of accurately modeling flows with separation or recirculation.

In a one equation model such as the Prandl-Kolmogorov model (Rodi, 1984), υ_t is computed as

$$\upsilon_t = C_\mu^{'}\sqrt{k}L \qquad [2.38]$$

where κ = the turbulent kinetic energy per unit mass, L = a length scale which may or may not be the same as l_m, $C_\mu^{'}$ = an empirical coefficient. k is obtained by solving the transport equation for turbulent kinetic energy.

In the two equation $k-\varepsilon$ model of Launder and Spadling (1974), υ_t is computed as

$$\upsilon_t = C_\mu \frac{k^2}{\varepsilon} \qquad [2.39]$$

where C_μ = an empirical coefficient and ε = energy dissipation rate. Two transport equations are solved to obtain k and ε.

2.4.3 Modeling pressure

In incompressible flow, the fluid density remains constant. This has led to the development of many techniques for the computation of pressure. These techniques can be divided into two categories: the artificial compressibility and the pressure correction methods.

Artificial compressibility method

In this method an artificial compressibility term similar to the time derivative of density is included into the continuity equation (Chorin, 1967)

$$\frac{\partial p}{\partial t} + a^2\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) = 0, \qquad [2.40]$$

where a is a parameter which can be interpreted as an artificial speed of sound and $p = a^2\rho$. For the steady state flow calculation the time derivative vanishes when steady state is achieved. The convergence of the flow calculation depends on the choice of a. Kwak *et al*. (1986) suggested the value of a to be between 0.1 and 10.

Pressure correction methods

In the pressure correction methods different forms of the Pressure Poisson equation are solved. The differential pressure Poisson equation can be derived from the RANS momentum and the continuity equations. The momentum equations 2.12- 2.14 are differentiated with respect to their respective coordinate direction, added up together and the continuity equation is applied. The pressure Poisson equation for 3D flow can be written as

$$\begin{aligned}\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} + \frac{\partial^2 p}{\partial z^2} &= 2\rho\left(\frac{\partial u}{\partial x}\frac{\partial v}{\partial y} + \frac{\partial v}{\partial y}\frac{\partial w}{\partial z} + \frac{\partial w}{\partial z}\frac{\partial u}{\partial x}\right) \\ &\quad - 2\rho\left(\frac{\partial u}{\partial y}\frac{\partial v}{\partial x} + \frac{\partial v}{\partial z}\frac{\partial w}{\partial y} + \frac{\partial w}{\partial x}\frac{\partial u}{\partial z}\right) \\ &\quad + \left(\frac{\partial^2 \sigma_x}{\partial x^2} + \frac{\partial^2 \sigma_y}{\partial y^2} + \frac{\partial^2 \sigma_z}{\partial z^2} + 2\frac{\partial^2 \tau_{xy}}{\partial x \partial y} + 2\frac{\partial^2 \tau_{yz}}{\partial y \partial z} + 2\frac{\partial^2 \tau_{zx}}{\partial z \partial z}\right)\end{aligned} \qquad [2.41]$$

In computational fluid dynamics, a discrete Poisson equation is derived from the discrete momentum and continuity equations with some assumptions. One of the widely used methods based on this technique is the Semi-Implicit Method for Pressure-Linked Equations (SIMPLE) method of Patankar and Spalding (1972). In the SIMPLE method pressure and velocity components are split into guessed and correction parts. First a guessed velocity field is obtained from the discrete momentum equations using a guessed pressure field. Then the discrete Poisson equation is then solved for the pressure correction. Finally the pressure is updated by adding the guessed and the correction values and the velocities are updated. This technique tends to overcorrect the pressure field requiring an under-relaxation in the pressure update for stability. This procedure is repeated after setting the updated flow field as guessed until the convergence is obtained.

An important aspect of the pressure correction methods is that the discretization of the continuity equation and the pressure terms in the momentum equations by a centered difference scheme usually leads to a ‘checker board’ type flow field in a regular collocated grid where all the variables are computed at the same location. This problem is alleviated by the use of a staggered grid in the control volume formulation where the velocities are computed at the nodes whereas the pressure is computed at the cell faces. To avoid the checker board pressure field in the collocated grid, Rhie and Chow (1983) modified the discretization of the velocities in the continuity equation by adding a higher order

term in the velocity interpolation. Majumder (1988) improved this technique to avoid the dependence of the final solution on the under-relaxation parameter.

Modifications of the SIMPLE method lead to the development of different algorithms such as SIMPLER (SIMPLE-Revised) (Patankar, 1980), SIMPLEC (SIMPLE-Consistent) (van Doormal and Raithby, 1984) and PISO (Pressure Implicit with Splitting of Operators) (Issa, 1986).

2.4.4 Free surface computation techniques

One of the most important aspects of open channel flow computation by the RANS equations is the free surface modeling. The set of the RANS equations does not contain any model equation to locate the free surface. Different techniques have been developed to model the free surface.

Rigid Lid approximation

The simplest method is to ignore the free surface variation in the steady flow calculation and use a 'rigid lid' approximation of the free surface. In this technique the water surface is approximated from measurements or set at a slope equal to the average bed slope. The zero pressure boundary condition no longer holds and pressure is computed at the surface which generally reflects the error in the water surface approximation.

Pressure and Bernoulli equations

The pressure at the water surface computed using the rigid lid approximation can be used to update the water surface by applying the Bernoulli equation for energy balance in potential flow. The Bernoulli equation can be written as

$$p + \rho g H + \frac{1}{2}\rho V_r = B \qquad [2.42]$$

where V_r is resultant velocity at the water surface, B is a constant.

This equation is applied along the water surface based on one fixed point usually at a downstream location (Olsen, 2009).

Kinematic free surface equation

In this technique free surface kinematic condition (equation [2.15]) is used to update the water surface elevations. RANS equations are solved to obtain the velocity components (u,v,w) and this equation is used to update the water level. CCHE3D model uses this technique for water surface computation (Jia *et al*. 2005).

Depth averaged continuity equation

In this technique the depth averaged continuity equation is used to compute the water surface. Depth averaged velocities are obtained by depth averaging the vertical profiles of the corresponding velocity obtained from the

RANS solution. Vreugdenhil (1994) mentioned that this technique is more robust that the kinematic free surface equations.

Depth averaged Poisson equation

Depth averaged Poisson equations can be derived by differentiating the depth averaged *x* and *y* momentum equations in the non-conservative form by their respective coordinate directions. Wu *et al.* (2000) used this technique to model the flow in a curved open channel. However, they mentioned that this technique may not be suitable for situations with steep water surface slopes.

Volume of fluid method (VOF)

The volume of fluid (VOF) of Hirt and Nicols (1981) is the most robust free surface computation technique. In this technique the volume fraction of a liquid (say water) is computed at each cell. The cell having a volume fraction of one is considered to be full while a value of zero indicates the cell to be empty. The cells having a volume fraction less than one but greater that zero contains the free surface. Once the cells containing the free surface are identified, the location and slope of the water surface at each cell are then computed.

2.4.5 Hydrostatic RANS model

The 3D model for open channel flow can also be applied with a hydrostatic pressure assumption. In this case the horizontal momentum equations

are solved for horizontal velocities with the continuity equation for the vertical velocity. Rigid lid assumption cannot be applied and the water surface elevation must be updated.

2.5 MODELING OF THE DA EQUATIONS

2.5.1 Discretization techniques

The modeling techniques are primarily developed for the St. Venant equations. These equations are predominantly hyperbolic and have similar mathematical properties to the compressible Navier-Stokes or the Euler equations. The numerical techniques developed for the compressible flow equations have successfully been extended for the depth averaged equations. Lax Wendroff (Lax and Wendroff 1960) and MacCormack (MacCormack 1969) schemes are two classical finite difference methods, give second order accuracy in space and time (Chung 2002). The MacCormack is a two-step predictor-corrector scheme and easy to program. However these centered difference methods may require artificial viscosity to dampen the high frequency oscillations (Jameson et al. 1981, Chaudhry 1993). Since information can propagate both directions in the Saint Venant equations, upwind methods are based on the characteristic directions. Generally the flux is split into positive and negative components based on the eigenvalues that represent the characteristics of the hyperbolic St Venant equations. In the first order accurate method, backward and forward differencing are used for the terms associated with the negative and positive eigenvalues

respectively. For higher order accuracy second order upwind methods based on the TVD concept have also been used (Garcia-Navarro et al. 1992, Louked and Hanich 1998, Wang et al. 2000).

The finite element method provides flexibility to represent the irregular geometry in modeling natural rivers. Upwind biased finite element methods are known as the Petrov Galerkin finite element methods. Hicks (1990) developed the characteristic dissipative Galerkin (CDG) finite element method for one dimensional open channel flow equations. She compared the CDG method with different finite element and finite difference methods extensively for a wide range of test problems and found the CDG method to be the best method both in terms of accuracy and stability. Ghanem (1995) extended the CDG method for two dimensional depth averaged flow. Subsequently a river modeling software 'River2D' (Steffler and Blackburn 2002) has been developed and is currently being used by the practitioners.

2.5.2 Modeling depth averaged turbulent stresses

The bed shear stresses are approximated according to

$$\tau_{bx}^{d} = \frac{\bar{u}\sqrt{\bar{u}^2 + \bar{v}^2}}{C_*^2}, \tau_{by}^{d} = \frac{\bar{v}\sqrt{\bar{u}^2 + \bar{v}^2}}{C_*^2} \qquad [2.43]$$

where C_* is the dimensionless Chezy Coefficient and is related to the effective roughness height k_s, through

$$C_* = 5.75\log(\frac{12h}{k_s}) \qquad [2.44]$$

$\sigma_x^d, \tau_{xy}^d, \tau_{yx}^d$ and σ_y^d are the depth averaged turbulent stresses and are modeled with an isotropic depth averaged eddy viscosity (υ_t^d) as

$$\tau_{xy}^d = \tau_{yx}^d = \rho\upsilon_t^d\left(\frac{\partial\bar{u}}{\partial y} + \frac{\partial\bar{v}}{\partial x}\right)$$

$$\sigma_x^d = 2\rho\upsilon_t^d\left(\frac{\partial\bar{u}}{\partial x}\right) \qquad [2.45]$$

$$\sigma_y^d = 2\rho\upsilon_t^d\left(\frac{\partial\bar{v}}{\partial y}\right)$$

The eddy viscosity (υ_t^d) is modeled by any of the following equations

$$\upsilon_t^d = (0.14 \pm 0.07)u_* h \qquad [2.46]$$

(Fisher, 1979)

$$\upsilon_t^d = \varepsilon_1 + \varepsilon_2\frac{h\sqrt{\bar{u}^2 + \bar{v}^2}}{C_*} + {\varepsilon_3}^2 h^2\sqrt{2\frac{\partial\bar{u}}{\partial x} + \left(\frac{\partial\bar{u}}{\partial y} + \frac{\partial\bar{v}}{\partial x}\right)^2 + 2\frac{\partial\bar{v}}{\partial y}} \qquad [2.47]$$

(Steffler and Balckburn 2002)

The default values of the parameters are ε_1=0, ε_2=0.5 and ε_3=0.1. These values can be adjusted by calibration.

2.6 LINEAR SOLVERS

The temporal discretization of the governing differential equations can be classified into two catagories: explicit and implicit methods. In general, explicit methods are easy to implement and require low memory storage. The time step size of an explicit method is restricted by the Courant-Fredrichs-Lewy (CFL) stability condition which often limits the steady state convergence speed of the method. To accelerate the convergence of an explicit method, techniques such as local time stepping and implicit residual smoothing with multigrid acceleration have been proposed (Lackey and Sotiropoulos 2005, Jameson 1985). However, for large scale problems the rate of convergence of an explicit method may deteriorate dramatically and become an inefficient solution technique. Moreover for an unstructured grid, multigrid acceleration technique with explicit methods may become very difficult to implement.

On the other hand, implicit methods allow large time steps for rapid convergence to the steady state solution. The implicit discretization of the governing equations results in nonlinear systems of equations which are linearized by the Newton-Raphson method giving a system of linear equations.

In general, there are two types of solver for the solution of the linear system: direct and iterative methods. The resulting linear systems of equations can be solved directly if the number of nodes is relatively small. However, for a very large number of nodes and large bandwidth of the resulting matrix, a direct

solution becomes unfeasible due to the precipitous growth in computation and storage ().

The most widely used methods to solve linear systems are the iterative methods. Iterative methods are further classified into basic and Krylov subspace methods. Basic iterative methods include Jacobi, Symmetric Gauss-Seidel (SGS) and Incomplete Lower Upper (ILU) factorizations. The convergence of the basic iterative solvers can sometimes be greatly enhanced by the multigrid method especially for elliptic and diffusion dominated problems (Trottenberg *et al*. 2001). Among different Krylov subspace methods, the Generalized Minimal Residual (GMRES) and some variants of Bi-Conjugate Gradient (BCG) methods are very popular iterative solvers for general unsymmetric linear systems arising from a wide variety of applications (Feng *et al.* 1997). However an efficient preconditioner is needed to improve the convergence of these Krylov iterative methods. The most commonly used preconditioners are the Jacobi, SGS and ILU factorizations.

Significant effort has been made to develop efficient iterative solution methods for the computational fluid dynamic (CFD) problems. Venkatakrishnan and Mavriplis (1992) developed a Newton-GMRES implicit solver for computing compressible flow around a multi-element airfoil in an unstructured grid. They investigated the Jacobi, SGS and ILU factorization with no fill (ILU(0)) as preconditioners. ILU(0) and SGS with over-relaxation have also been studied as iterative solvers. ILU(0) preconditioned GMRES showed the best performance.

Similar study was performed by Rogers (1995) in a structured mesh where ILU(0) preconditioned GMRES outperformed all other methods by at least a factor of two. A variant of GMRES is known as the matrix free GMRES where the matrix vector multiplication is computed by a first order forward difference. For a structured mesh, Pueo and Zingg (1998) presented a preconditioned matrix free GMRES algorithm for the computation of steady two dimensional aerodynamic flow. They investigated the ILU preconditioners with different levels of fill ILU(l) where l is the level of fill. After a thorough parametric study they found ILU (2) to be the best preconditioner. Using a similar approach, Nichols and Zingg (2005) found ILU(1) to be the best preconditioner for 3D flow. Manzano et al. (2003) used different levels of fill ranging from 1 to 3 in the ILU preconditioners depending on the test cases to develop an efficient matrix-free GMRES solver for 3D unstructured mesh. Nejat and Ollivier-Gooch (2008b) used ILU(1) preconditioner in the start-up phase with a ILU(4) preconditioner in the Newton phase of a GMRES solver for a fourth order accurate discretization of the inviscid compressible flow equations.

GMRES has widely been used for its smooth convergence, but it may be expensive in memory usage and computational work even with its restarted version. The Bi-Conjugate Gradient (BCG) method (Fletcher 1974) is another Krylov subspace method for unsymmetrical matrices, uses a different approach than GMRES and requires less memory than GMRES. Sonneveld (1989) proposed the Conjugate Gradient-Squared (CGS) method to improve the

convergence of BCG, but it has been found to be less stable than GMRES and the convergence was found to be very irregular (Shen and Wu 1995). van der Vorst (1992) proposed a stabilized version of the CGS algorithm, known as the Bi-Conjugate Gradient Stabilized (BiCGSTAB) to obtain smooth convergence. Lin *et al.* (1995) tested different variants of the BCG methods with an ILU preconditioner and recommended BiCGSTAB to be the best solver.

Although ILU preconditioners have successfully been used for improving the convergence of the Krylov subspace methods, they have some limitations. In the case of non-symmetric non-diagonally dominant matrices the incomplete factors may be more poorly conditioned than the original matrix and make the solver unstable (Chow and Saad 1997). One straightforward way to address these poorly conditioned factors is to add a diagonal perturbation to the original matrix to increase its diagonal dominance and perform incomplete factorization on this matrix (Manteuffel, 1980, Van der Vorst, 1981, Saad 1994). A major difficulty in this approach is to find the value of perturbation. Another way to increase the diagonal dominance is to reduce the off-diagonal dominance of the matrix (Pueyo and Zinng 1998). In aerodynamic computations, it is very common to use a lower order Jacobian obtained from a first order spatial discretization as preconditioner for higher order spatial discretizations to increase the stability of the ILU factorizations.

2.7 CURRENT STATUS OF OCF MODLEING

2.7.1 Depth averaged modeling

At the present time depth averaged (DA) models are widely used for river flow modeling instead of 3D CFD models. The classical St. Venant equations used for one and two dimensional depth averaged flow simulation in open channels are derived assuming uniform velocity, hydrostatic pressure and small channel slope (Chaudhry 1993). These equations cannot provide any vertical detail of the flow field and usually do not include effects which may be important for the study of rapidly varied flows where the velocity distribution is highly non-uniform and the pressure is significantly non-hydrostatic. Therefore they are only valid for flows with length scales much greater than the flow depth (Steffler and Jin 1993).

Most of the scientific effort in shallow water equations focused on resolving the flow discontinuities and achieving higher order accuracy of the simulation. But it is also important to achieve these solutions with computational efficiency and robustness. Very few literatures on the implicit iterative solvers for the solution of the shallow water equations exist. Of them, Barragy *et al*. (1993) applied Bi-Conjugate Gradient and Bi-Conjugate Gradient Squared methods with ILU preconditioner in tidal simulation. Their results indicate that with sufficient fill, the iterative methods are competitive with the direct frontal solver as they used a relatively small number of nodes. Fang and Sheu (1999) compared the relative performance of the GMRES method of Saad and Schultz (1986) and the

Bi-CGSTAB of van der Vorst (1992) without any preconditioning in simulating the bore wave propagation by the Taylor-Galerkin finite element method. The GMRES solver showed a much better convergence rate than the Bi-CGSTAB solver.

2.7.2 Quasi 3D modeling

Generally uniform velocity and hydrostatic pressure assumptions are valid in shallow water flow where the horizontal extent is very large as compared to the depth. In a natural river the flow it may exibit local 3D flow pattern where the velocity may be non-uniform and/or pressure may be non-hydrostatic due to bed and bank curvature. These flowq pattern may affect the depth averaged flow distributions. A number of studies have been performed to improve the depth averaged equations by including the effect of the vertical distribution of velocity and non-hydrostatic pressure. The Boussinesq equations are the improvements of the St. Venant equations by including the non-hydrostatic pressure and vertical velocity distributions, but only applicable for flows with wavelength to depth ratio greater than about ten (Steffler and Jin, 1993). These equations do not provide vertical distribution of longitudinal velocities better than the St. Venant equations. Dressler (1978) attempted to incorporate the bed curvature using curvilinear coordinates in the Euler equations assuming the water surface curvature is the same as the bed curvature. Hager and Hutter (1984) improved the model assuming a linear variation of flow angle and curvature between the bed and surface.

However all these potential flow assumptions do not include the turbulence and rotationality present in open channel flow.

Steffler and Jin (1993) proposed vertically averaged and moment equations (VAM) to take into account more vertical details in the depth averaged equations. The VAM equations were derived by integrating the RANS equations over the flow depth with a linear approximation for the longitudinal velocity distribution and a quadratic approximation for the vertical velocity and pressure distributions. Jin and Steffler (1993) used depth averaged and moment equations in a curved bend where the hydrostatic pressure distribution was assumed and the horizontal velocity was approximated by a power law distribution of mean velocity with a linear variation of perturbation (deviation from mean). Jin and Li (1996) applied the model of Jin and Steffler (1993) assuming a linear variation of the pressure perturbation (deviation from hydrostatic pressure). Khan (1995) applied the 1-D VAM equations in simulating the flow over curve beds and modeling overfalls. Ghamry (1999) extended the VAM equations in two dimensions with several assumed distributions. VAM equations can often give very good water surface profile and mean velocity, reasonable vertical velocity and pressure distribution and relatively poor distribution of horizontal velocity which is probably the most useful output from a 3D river modeling. Moreover VAM models may be prone to numerical instability in modeling complex flows such as the flow over a dune (Elgamal 2002). However none of the improved DA models have ever been successful in modeling a natural channel.

Multilayer models have been used by different researchers to compute the water surface and the velocity profiles (Lai and Yen 1993, Li and Yu 1996). The flow depth is divided in a number of stream wise layers and the depth averaged equations are solved within each layer assuming a hydrostatic pressure distribution. However, as the flow quantities are assumed uniform within a layer, the large number of layers required for accurate results may increase computation time. Xia and Jin (2006) used the moment equations with the layer averaged equations in each layer to improve the multilayer model by using a lesser number of flow layers. They assumed linear profiles of the flow variables within a layer.

2.7.3 3D modeling

To obtain detail and accurate velocity, pressure and bed stress distribution, 3D Computational fluid dynamic (CFD) models based on the Reynolds averaged Navier-Stokes (RANS) equations have been developed for open channel flow computation. CFD models based on the Reynolds averaged Navier-Stokes (RANS) equations with the free surface variation have also been developed for 2D plane open channel flow computation. John (1991) developed a free surface flow model where the pressure was decomposed into a hydrostatic and a hydrodynamic component that was computed from an integrated vertical momentum equation. Stansby and Zhou (1998) used the depth averaged continuity equation for water surface computation while a pressure correction technique was used for the non-hydrostatic pressure computation and mass

balance. Namin et al. (1999) used the kinematic free surface equation to compute the free surface variation in their implicit non-hydrostatic model for 2D plane flow.

Many studies have been performed to apply 3D models in simulating the flow in physical models with or without the free surface variations; however studies of natural rivers with complex geometries are limited. Sinha et al (1998) simulated the flow in a 4 km fairly straight stretch of the Columbia River downstream of the Wanapum Dam with complex bed topography. The flow domain was discretized using 0.33 million nodes with a mesh aspect ratio of as high as 20. The k-ε turbulence model was used for turbulence closure. The water surface boundary was treated as a fixed lid using measured data. To increase the computational efficiency, they used fully vectorizable algorithms to invert the implicit operators. Due to the large cell aspect ratio, they were unable to reduce the residual by more than three orders of magnitude and also they had to accept a mass imbalance of 3% through each stream wise cross section, yet their simulation required 150 IBM-380 CPU hours for converged solution.

Dargahi (2004) used the FLUENT model with 1 million unstructured elements to simulate the flow in a 6 km reach of the Klarälven river including a bifurcation. He attempted to use the volume of fluid (VOF) method of Hirt and Nichols (1981) but was unable to model the free surface variation; instead he modeled the water surface boundary as a rigid lid. However the simulation

required 6 weeks of computation to obtain a steady state flow condition in a double processor PC.

Lu and Wang (2009) used a 3D model to simulate the flow in a 16 km reach of the Yangtze river with 163(longitudinal)X81 (transversal)X15 (vertical) computational nodes. The free surface was computed by the 2D Poisson equation derived from the depth averaged equations. Jia *et al.* (2009) used CCHE3D to model the flow field affected by weirs in the 4.6 km stretch of the Victoria bendway with the flow depth varying from 15 to 35 m. They used over 0.4 million computational nodes with 322, 123 and 11 nodes in the longitudinal, transversal and vertical directions. The free surface was modeled using the free surface kinematic equation. The last three studies did not mention anything about the computational time.

The difficulties in using a general CFD model in a natural river arise mainly from the horizontal of the river extent as compared to the depth. Normally, the spatial resolution in all directions is governed by a requirement to have 10-20 nodes in the vertical dimension (approximately 10 cm spacing) whereas the horizontal dimensions may extend for several kilometers. Therefore it is very common to use a course mesh with a high mesh aspect ratio to discretize the domain, yet the use of a fully coupled implicit solver where all the flow variables are solved simultaneously may not be possible as a large matrix has to be solved. Sequential implicit solvers are generally used that also require the iterative matrix

solvers for the inversion of the implicit operators. These approach requires relatively large number of time steps as compared to a fully coupled model.

Another difficulty in 3D river modeling is the free surface modeling. Two options are available, dynamic free surface computation and rigid lid assumptions. For the first option, the VOF method is robust but may require high computational effort. Other methods are not very accurate or efficient and also require re-meshing of the domain each time the water surface is updated. This leads to increased computational effort even with a hydrostatic pressure assumption. Rigid lid assumptions from measured water surface have been used but may require many measurements to properly define the computational domain and are only applicable for the discharge for which the measurements are made. Non-hydrostatic pressure must be computed throughout the domain for mass balance by a pressure-velocity coupling technique even if it is not significant. Since the pressure equations are elliptic, information travels in all direction that may require a long time in a natural river. A large mesh aspect ratio may induce numerical stiffness and may deteriorate the convergence of the solution (Sinha et al 1998, Buelow et al 1994). Therefore it may be difficult to use an adequately refined descretization, a sophisticated turbulence model (e.g. shear stress transport), a higher order upwind method and allow the free surface variation due to computational time limitations in a natural channel.

2.8 PRESENT STUDY

The present study focuses on two issues of open channel flow modeling. One is the behavior of the linear matrix solvers in modeling the St. Venant equations by the implicit methods. The linearization of the St Venant equations produces an unsymmetric matrix. Therefore in this study, the behavior of different matrix solvers for unsymmetric matrices available in the literature will be tested for implicit modeling of the St Venant equations. Potential and limitations of these methods will be identified and possibility of improving the methods will be investigated. Generally the Krylov subspace methods with different preconditioners have been found to be very efficient for implicit modeling of the hyperbolic equations such as the Euler equations. Since the St Venant equations are predominantly hyperbolic, these methods will be tested in this study.

The other area in open channel flow that lacks comprehensive study is the 3 dimensional modeling. The problems in 3D modeling of natural rivers can be identified as the dynamic free surface computation, the mass balance by a pressure-velocity coupling for a large mesh aspect ratio and the performance of the iterative solver due to the large computational domain.

In this study a new concept is presented for modeling open channel flow which attempts to overcome the problems. The distribution of water depth and DA velocity in a natural channel is generally governed by the upstream, downstream and bed boundary conditions and partially depends on the non-uniform velocity (NUV) and the non-hydrostatic pressure (NHP). On the other

hand the vertical profiles of velocity and NHP mainly depend on the bed profile and roughness and local flow field. Therefore it might be useful to separate the DA variables from the 3D distributions and compute them by any DA model. Then keeping the DA outputs fixed, the 3D variation of velocity and NHP can be computed from the RANS model. Because 3D variations mainly depends on local DA flow field and bed variations which will be kept unchanged, and are not affected by the far upstream and downstream flow conditions, solution of the RANS model is expected to be faster.

The proposed model will consist of two parts, a depth averaged model and a Reynolds averaged Navier-Stokes (RANS) model. First a depth averaged model will be solved to the steady state to obtain the water surface elevation and depth averaged velocity neglecting the effects of NUV and NHP. Then for the hydrostatic RANS (RANS-H) model, the RANS horizontal momentum equations will be solved to obtain horizontal velocity profiles together with the RANS continuity equation for vertical velocity profiles using the computed water surface as a fixed upper boundary. Mass balance obtained in the DA computation will be retained by adding correction terms in the horizontal momentum equations. This will also enable modeling the horizontal velocity by only solving the horizontal momentum equations and neglecting the vertical velocity. In the non-hydrostatic model (RANS-NH), the pressure Poisson equation derived from the RANS equations will be solved for NHP assuming a zero pressure at the water surface with the z momentum equation for the bed pressure. The zero pressure condition

at the surface would have much stronger influence on the pressure field than the upstream and downstream boundaries as the depth of flow is significantly less than the horizontal dimensions. Consequently solution for pressure should be faster. Also any model with a fixed boundary condition tends to converge faster than a gradient type boundary that is commonly used for pressure solution. Again the mass balance will be retained by a correction term similar to the RANS-H model. As the 3D flow field depends on local flow conditions, the RANS horizontal momentum equations will be discretized explicitly in the horizontal direction with an implicit coupling in the vertical direction. Once the NUV and NHP are obtained from the RANS model, their effects will be included in the DA model to update the DA velocity and water depth. Then the 3D flow variations can again be computed from the RANS model based on the updated DA flow field. The model will first be developed and tested for 2D plane flow. Then it will be extended for 3D flow.

2.9 SPECIFIC OBJECTIVES OF THIS STUDY:

The specific objectives of this study are:

1. To develop an efficient and robust implicit iterative solver for modeling the St. Venant equations.
2. To develop a coupled depth averaged-RANS model for plane open channel flow.
3. To extend the coupled model for 3D flow.

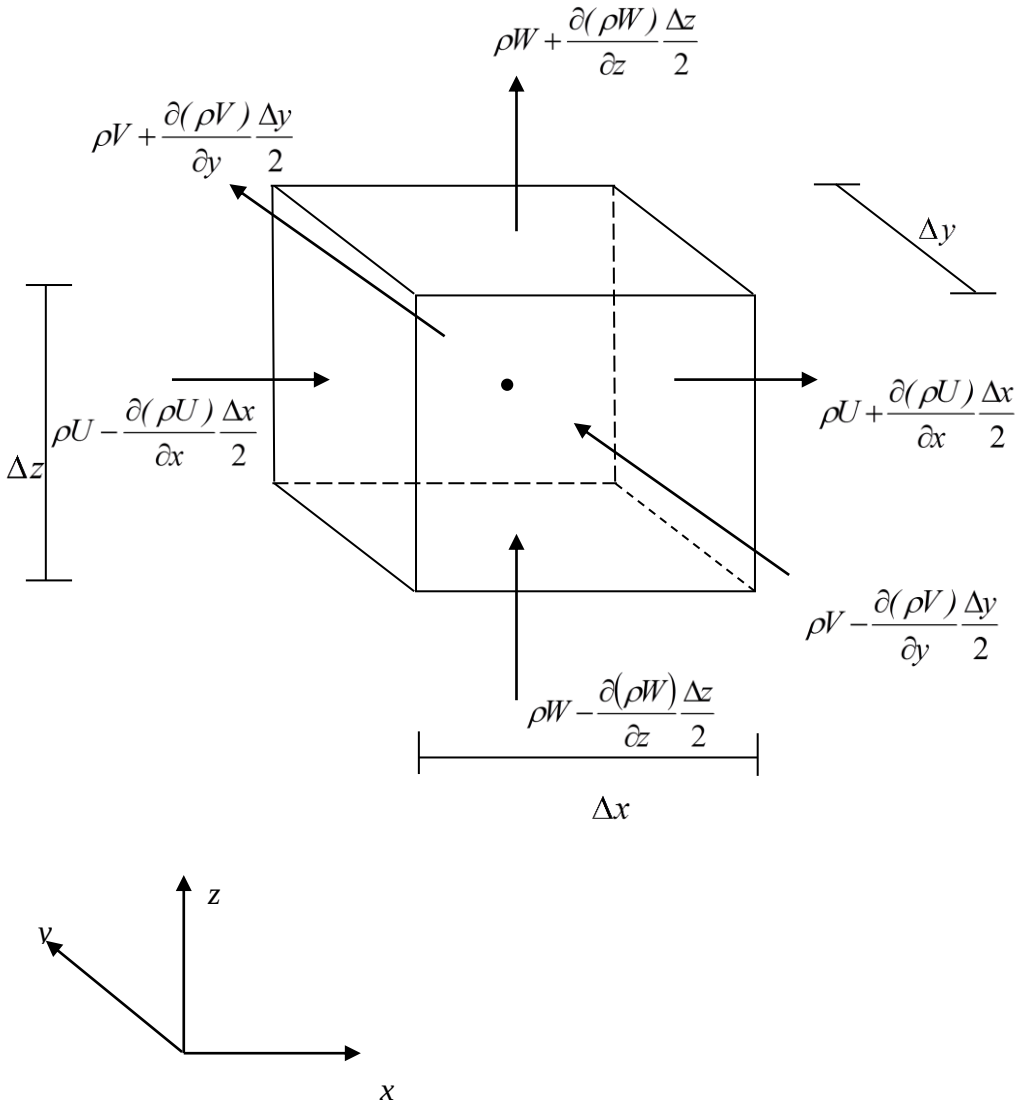

Figure 2.1 Mass flow through a fluid element

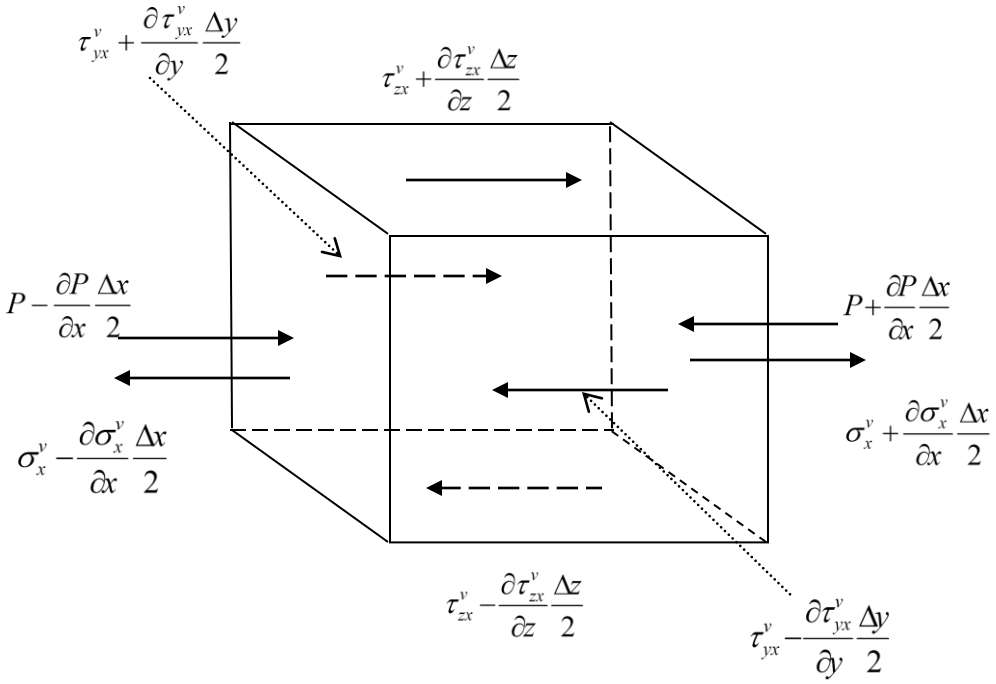

Figure 2.2 Stress components of a fluid element in the *x* direction

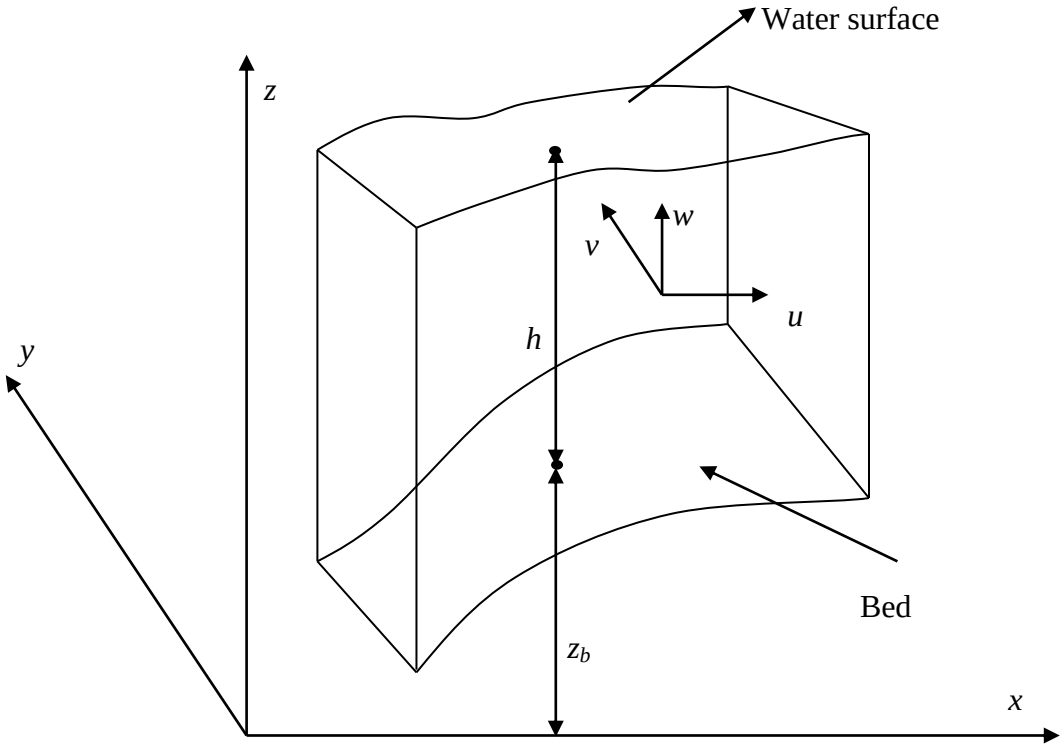

Figure 2.3 Definition sketch of 3D open channel flow

Chapter 3

ITERATIVE SOLVER FOR THE FINITE ELEMENT MODELING OF OPENCHANNEL FLOW

3.1 INTRODUCTION

The finite element (FE) method is attractive for simulating flow of water in a natural river with complex geometry due to its ability to represent the domain correctly using an unstructured mesh. Ghanem (1995) applied the characteristic dissipative Galerkin (CDG) FE method developed by Hicks and Steffler (1992) in modeling the St. Venant equations in two dimensions using triangular elements. Subsequently a river modeling software River 2D (Steffler and Blackburn 2002) has been developed. River2D has the ability to model the wetting-drying process. In the wet areas St. Venant equations which are essentially hyperbolic in nature are solved while in the dry areas parabolic Dupit equations are solved. Depending on the water level proportion of domain can be roughly equally divided.

In this study the convergence of the Krylov methods for the River2D model is investigated. The most important aspect of implementing any Krylov method is the preconditioning of the system. The main objective of this work is to test and develop preconditioned GMRES and BiCGSTAB solvers for the computation of

steady state solutions by the River 2D model. The preconditioners used are Jacobi, Gauss-Seidel and Incomplete lower-upper factorization (ILU) with different levels of fill. Also a new technique to improve the convergence and robustness of the ILU preconditioners is developed.

3.2 GOVERNING EQUATIONS OF RIVER2D MODEL

River2D solves the St. Venant equations (equations [2.27] to [2.29]) together with the Dupit equation:

$$\frac{\partial h}{\partial t} = \frac{T}{S}\left(\frac{\partial^2}{\partial x^2}\left(h + z_b\right) + \frac{\partial^2}{\partial y^2}\left(h + z_b\right) \right) \qquad [3.1]$$

In the River2D model, equations [2.27] and [3.1] are added together and solved in conjunction with equations [2.28] and [2.29] for the wet elements. For the dry elements equation [2.28] and [2.29] are replaced by

$$q_x = \bar{u}h = 0 \qquad [3.2]$$

$$q_y = \bar{v}h = 0 \qquad [3.3]$$

In the above equations q_x, q_x =discharge per unit width in the x and y directions, S=aquifer storativity, T= aquifer transmissivity.

The governing equations are discretized and then linearized by the Newton-Raphson method and can be expressed as

$$A\Delta u = R \qquad [3.4]$$

where A = Jacobian Matrix $= M\frac{1}{\Delta t} + \frac{\partial R}{\partial Q}$, Δu= change in h, q_x and q_y, Δt= time step, M = mass matrix, R= residual vector of the nonlinear system. For the steady state solution the mass matrix is lumped to obtain an identity matrix. Therefore Δt only contributes to the main diagonal of the Jacobian matrix.

In the FE modeling by the CDG method and linear interpolation function the resulting Jacobian matrix has a graph identical to the graph of the supporting unstructured mesh. Therefore this modeling technique provides higher order accuracy with the minimum possible storage for the Jacobian matrix. The Jacobian matrix can be computed analytically or numerically. In this study, the Jacobian matrix is computed numerically.

Equation [3.4] is a large linear system of equations which must be solved repeatedly at each time step to obtain an update for the vectors of unknown. When the time step is small, the Jacobian matrix is block diagonally dominant and the linear system can be solved with fewer iterations. As the time step increases the matrix tends to become off diagonally dominant and the linear system generally becomes difficult to solve requiring more iterations if the solver does not stall.

3.3 KRYLOV SUBSPACE METHODS

The Krylov subspace methods are considered to be the most effective iterative techniques for solving the large linear systems. Let x_0 be an initial approximation of the linear system $A\Delta u=R$. In this method an approximate solution x_m of the linear system is obtained from an affine subspace x_o+K_m such that

$$R - Ax_m \perp L_m$$

where K_m is the Krylov subspace of dimension m defined as

$$K_m(A, r_0) = span\{r_0, Ar_0, A^2 r_0, ..., A^{m-1} r_0\}, \text{ with } r_o = R - Ax_0$$

L_m is another subspace of dimension m.

Different choices for $L_{m,}$ give rise to different iterative methods (Saad, 1996).

In this study two Krylov subspace methods for unsymmetric matrix namely, GMRES and BiCGSTAB are used since they are found to be the best methods in other studies.

In the GMRES method, the Krylov subspace K_m is defined as

$$K_m(A, v_1) = span\{v_1, Av_1, A^2 v_1, ..., A^{m-1} v_1\}, \text{ where } v_1 = r_0 / \|r_0\|_2,$$

and the subspace L_m is taken as AK_m .

GMRES uses the Arnoldi algorithm to form an orthonormal basis of $K_m(A, v_1)$ by the modified Gram-Schimdt orthogonalization. The storage requirement of GMRES increases linearly and the CPU expense increases quadratically with the increase of the basis vectors in the Krylov subspace. To remedy this problem, GMRES is terminated when the dimension of the Krylov subspace is equal to *m* and then restarted using the latest solution as the initial guess. This is known as the restated version of GMRES and denoted as GMRES (*m*). Usually a value of 10 to 20 is used for *m.*

In the BCG methods, the subspace K_m is defined as before but L_m is taken as

$$L_m(A^T, w_1) = span\{w_1, A^T w_1, (A^T)^2 w_1, ..., (A^T)^{m-1} w_1\}$$

w_1 can be any vector provided v_1 and w_1 are not orthogonal but often chosen to be equal to v_1. The Lanczos bi-orthogonal algorithm is used to form a pair of bi-orthogonal bases for the two subspaces. The BCG algorithms are more economical than the GMRES algorithm in terms of memory usage. Among the BCG methods BiCGSTAB has more regular convergence behavior.

3.4 PRECONDITIONING

The convergence of the Krylov subspace methods is highly dependent on the condition number of the linear system. These methods work the best when the eigenvalues are clustered. The easiest way to improve their efficiency and robustness is to use preconditioning. This attempts to cluster the eigenvalues at a single value (Nejat and Ollivier-Gooch 2008a). Two types of preconditioning are usually used; left and right preconditioning. With the right preconditioning the system $A\Delta u = R$ becomes

$$AC^{-1}C = R \qquad [3.5]$$

where C is a preconditioning matrix and approximation to A which is much easier to solve than A. The idea is that the preconditioner transforms the original matrix as close as possible to the identity matrix. Finding an optimum preconditioner is not straight forward but something of an art. In this study Jacobi, Symmetric Gauss-Seidel (SGS) and Incomplete Lower Upper (ILU) facorization are used as preconditioners.

In the Jacobi preconditioning, C is taken as the diagonal of A.

In the SGS, $C=LU$

where $L = (Di+E)Di^{-1}$ and $U=Di+F$;

Here E=strict lower part of A, F=strict upper part of A and Di=Diagonal of A.

In the ILU factorization, matrix A is approximated as by a matrix C such that

$$A=C+B= LU+B$$

where L = lower triangular matrix, U = upper triangular matrix and B = error matrix. The factors in L and U are computed by a Gaussian elimination process of matrix A.

The accuracy of the factorization may depend on the number of nonzero entries retained in the factorization as compared to the original matrix. There are two commonly used approaches for forming the factorizations: ILU with level of fill-in (ILU(l)) and the threshold strategies. Of them, ILU(l) is the fastest to compute the factors and more widely used (Chapman et al, 2000). Therefore ILU(l) is used for this study. In ILU(l), the graph of the LU factorization is first determined and then the Gaussian elimination is performed to allow the nonzero entries based on the graph. The initial level of fill of an element a_{ij} of matrix A is defined by

$$lev_{ij} = \begin{cases} 0 & if\ a_{ij} \neq 0, \\ \infty & otherwise \end{cases}$$

During the Gaussian elimination, each time this element is modified by the element a_{kj} , the level of fill is updated as

$$lev_{ij} = \min\left\{lev_{ij,}\ lev_{ik} + lev_{kj} + 1\right\}$$

During this process, any element whose level of fill becomes greater than l is set to zero.

In ILU(0), no new entries are allowed and the factorization has the same pattern of nonzero entries as the original matrix. Since the wetting-drying process of the River2D model changes the nonzero structure of the original matrix, the nonzero structure of the wet elements is used for the dry elements. This allows the formation of the structure of ILU(l) at the beginning of simulation and the same structure can be used for the Gaussian elimination to compute the factors. Otherwise the formation of the structure of the level of fill may be very expensive each time the structure of the original matrix is changed with the wetting drying process.

Generally larger levels of fill in the ILU factorization may not be efficient when the CPU time to form the preconditioner is included (Pueyo and Zinng, 1998), therefore the fill level is restricted upto 2 in this study.

The increase in time step reduces the diagonal dominance of the Jacobian matrix. The preconditioner obtained from an off diagonal matrix may make the preconditioned system relatively difficult to solve. Therefore a new preconditioner obtained from a Jacobian matrix of lower time step will be tested.

This lower time step for each element will be obtained from a global Courant number (*Cr*). The effect of different Courant numbers will be tested to investigate its behavior and to find an optimum Courant number, if any.

The ordering of unknowns is an important factor in the convergence of the preconditioned iterative solvers (Dutto 1993). An efficient ordering is needed to reduce the bandwidth of a matrix and consequently the number of fill in the ILU factorization. The reverse Cuthill-McKee (RCM) method (Cuthill and McKee 1969) is well known for its bandwidth reduction algorithm and has widely been used by researchers (Pueyo and Zingg 1997, Nejat and Ollivier-Gooch 2008a). Therefore RCM method is used for the ordering of unknowns of this study.

3.5 RESULTS

To test the convergence performance of the preconditioned GMRES and BiCGSTAB methods, a portion of the South Platte River is chosen. This river has a very complex geometry. The depth, velocity and mesh size vary over a wide range. It contains many dry elements. The inflow discharge is 2.83 m^3/s. The mesh contains 9918 nodes and 18910 elements. To test the convergence behavior of the iterative methods, the flow is simulated upto 100 *s*. Then the Jacobian matrices corresponding to different time steps are obtained. The river with water depth at 100 *s* after the simulation started is shown in Figure 3.1.

To get an estimate of the Courant numbers corresponding to the time steps, the histogram of the number of elements against $(\overline{U}+c)/L_e$ is plotted in Figure 3.2. Here c = average celerity ($c=\sqrt{gh}$) of the three nodes of an element, $L_e=\sqrt{2A_e}$ and A_e=area of the element.

$(\overline{U}+c)/L$ can be considered as the Courant number corresponding to a time step of 1 *s* and is denoted as Cr^1. In the river, almost 5300 elements (28% of total elements) have Cr^1 less than 0.1 indicating the dry and very shallow areas. Almost 9000 elements (50% of total elements) have Cr^1 in between 0.5 to 1.5 with a median value of 1.0.

The convergence of the preconditioned solvers is investigated by the number of iterations required to reduce the L_2 norm of initial residual (obtained by setting Δu=0) by two orders of magnitude for different time steps. Previous studies have shown that although approximately solving the linear system may increase the number of nonlinear outer iterations to achieve a steady state solution, it reduces the overall computational time significantly (Najat and Ollivier-Gooch 2008b). For Jacobi preconditioner a maximum of 1000 iterations for both GMRES and BiCGSTAB is allowed while this number is 500 for the ILU preconditioners.

Results are presented by the number of iterations required by the Jacobi, ILU (0), ILU(1), and ILU(2) preconditioned GMRES and BiCGSTAB methods for the specified residual reduction in Table 3.1 and Table 3.2. 'NC' in the table

indicates that the solver did not converge within the maximum allowable iterations. The results of SGS are not included as this preconditioner cannot reduce the residual for any time step shown in the tables and thus found to be completely ineffective.

3.5.1 Results of GMRES

Table 3.1 shows the results of preconditioned GMRES for different time steps. For the Jacobi-GMRES solver, the rate of increase in the number of iteration with the increase of time step is almost linear. For ILU(0), the number of iterations when Δt=5 and 10 *s* are 6 and 9 respectively. Although the time step is doubled, the number of iterations has increased to 1.5 times which might be desirable. However the number of iterations increases to 79 when Δt=50 *s* and the solver stalls completely when Δt= 75 and 100 *s*. This shows a weakness of the ILU(0) preconditioner for larger time steps. The ILU(1) preconditioner shows similar behavior with increasing time steps. For the ILU(2) preconditioner an improvement in the convergence behavior upto Δt=50 *s* is observed. However this solver is also completely unable to converge when Δt= 75 and 100 *s*.

3.5.2 Results of BiCGSTAB

Table 3.2 shows the convergence behavior of the BiCGSTAB method with the preconditioners. Jacobi preconditioner shows a convergence behavior similar to Jacobi-GMRES. The ILU(0) preconditioner shows a better performance than

ILU(0)-GMRES. An increase from Δt=10 to 50 *s* increases the number of iterations from 5 to 32 and the solver is still convergent when Δt=75 and 100 *s*. However it requires 142 iterations when Δt=100 *s*. ILU(1) shows an inferior performance than ILU(0) with increasing time steps as an increase in time step from 10 to 50 *s* requires 10 times iterations and the solver does not converge when Δt=75 and 100 *s*. The ILU(2) preconditioner shows an almost linear increase in number of iterations upto Δt=50 *s*. It only takes 15 iterations when Δt=50 *s* as compared to 32 iterations of ILU(0), yet it cannot converge when Δt=75 and 100 *s*.

3.6 LOWER COURANT NUMBER ILU PRECONDITIONER (LCIP)

The results reveal that although the ILU preconditioners might be very effective for both GMRES and BiCGSTAB, they have the limitation of being very ill conditioned such that the solver becomes stalled for larger time steps. Increasing the amount of fill may improve the solver performance but it may not necessarily increase the robustness of the solver. The increase in time step makes the matrix off diagonally dominant. The ILU preconditioners obtained from such a matrix make the preconditioned system ill conditioned. As mentioned earlier that in the CFD computations with higher order spatial discretizations, a Jacobian matrix obtained from a first order spatial discretization is used for computing the

ILU preconditioners . Therefore this technique requires the computation of an additional matrix.

However this concept cannot be applied in the CDG FE modeling, since the spatial discretization uses the lowest possible graph of the mesh structure. To increase the diagonal dominance of the preconditioner a new concept is introduced where a preconditioner is obtained from a matrix A_m of lower time step than the original matrix. As mentioned before the time derivative of the governing equations contributes only to the diagonal of the Jacobian matrix *A*, therefore to compute the matrix A_m for the preconditioner *C*, it is only needed to replace the contributions of the simulation time step in matrix *A* by a lower time step. The local time step (Δt_e) of any element is computed from the Courant number as

$$\Delta t_e = \frac{Cr}{Cr^1}$$

Since a time step of any element greater than the simulation time step may decrease the efficiency of the preconditioner obtained from A_m, the maximum value of the time step Δt_e is limited to the simulation time step. This new Jacobian matrix A_m is tested for the Jacobi and the ILU preconditioners. No improvement is achieved for Jacobi although significant improvement is obtained for the ILU preconditioners. This preconditioner is named as the Lower Courant number ILU Preconditioner (LCIP). This new preconditioner is tested for three simulation time steps of 10, 50 and 100 *s*.

3.6.1 LCIP for GMRES

The results of LCIP for GMRES are shown in Figure 3.3. When Δt=10 *s*, Courant numbers of 10 and greater give the optimum result for all ILU preconditioners. Because the maximum time step is limited to the simulation time step, the higher Courant numbers essentially give a local time step equal to 10 *s* for most of the elements. For Courant numbers less than 10, the convergence deteriorates.

When Δt=50 *s*, ILU(0)-GMRES converges in 100 iterations for *Cr*=50. The number of iterations increases for *Cr*=35, then it decreases with decreasing Courant numbers till *Cr*=10, below which the number of iteration increases. An optimum range of *Cr*= 10 to 15 is obtained with the minimum number of iterations of 34 at *Cr*=10. For the same time step, ILU(1) and ILU(2) preconditioned GMRES requires 50 and 31 iterations respectively for *Cr*=50. The number of iterations decreases with decreasing Courant number with an optimum range of *Cr*=10 to 15 for ILU(1) and 15 to 20 for ILU(2). The minimum number of iterations are 24 for ILU(1) at *Cr*=15, and 16 for ILU(2) at both *Cr*=15 and 20.

When Δt=100 *s*, GMRES does not converge for any ILU preconditioner at *Cr*=50. At *Cr*=35 makes the solver convergent but requires approximately 200 iterations. A decrease in *Cr* decreases the number of iterations and minimum number of iterations are 63, 43 and 27 obtained at *Cr*= 10, 15 and 20 for ILU(0), ILU(1) and ILU(2) respectively. Further decrease in *Cr* increases the number of iterations and these numbers are greater than 100 for *Cr*=2.

3.6.2 LCIP for BiCGSTAB

Figure 3.4 shows the results of BiCGSTAB. Results similar to GMRES are obtained. A distinct range of *Cr* is observed for each of the ILU preconditioners. The optimum Courant numbers are 10, 15 and 20 for ILU(0), ILU(1) and ILU(2) respectively.

The above results show that the use of LCIP can significantly improve the convergence of GMRES and BiCGSTAB for larger time steps. General optimum values of *Cr*'s have been obtained. These values are 10, 15 and 20 for the ILU(0), ILU(1) and ILU(2) respectively, although a difference of ±5 does not affect the results for ILU(1) and ILU(2) with some small affect for ILU(0).

3.7 COMPARISON BETWEEN LCIP AND ORIGINAL PRECONDITIONERS

GMRES

Table 3 shows the comparison of results of the LCIP and the original ILU preconditioner for GMRES. For ILU(0), LCIP starts to improve the results from Δt=25 *s* and significant improvements are observed when Δt=35 *s* and above. When Δt=50 *s*, an improvement by a factor of more than 2 is obtained. For Δt=75 and 100 *s*, LCIP has stabilized the solver and an increase from Δt=50 to 100 *s* increases the number of iteration to a factor less than two.

For ILU(1) and ILU(2), the new technique improves the results by a factor of approximately 2 and 1.5 respectively. Similar to ILU(0), these preconditioners stabilizes the solver when Δt=75 and 100 *s* with an improvement in the convergence.

BiCGSTAB

Table 4 shows the comparison of results of the LCIP and the original ILU preconditioners for BiCGSTAB. For ILU(0), the effect of LCIP is observed from Δt=18 *s*. LCIP shows significant improvement in convergence for higher time steps.

3.8 COMPARISON AMONG DIFFERENT SOLVERS

In comparing different solvers, central processing unit (CPU) time for a given level of convergence is probably the best quantity. Figure 3.5 shows the CPU time in seconds required to reduce the L_2 norm of initial linear residual by two orders of magnitude by the preconditioned solvers. The CPU time also includes the time required to construct the ILU factors. From here on, ILU preconditioners are the lower Courant number ILU preconditioners using the optimum *Cr*.

For GMRES, ILU(0) shows better performance than ILU(1) and ILU(2) untill Δt=50 *s* when all preconditioners require the same CPU time. When Δt=5 *s* ILU(0) is better by a factor of 2 than ILU(2). When Δt is greater than 50, ILU(0) and ILU(1) show the same performance and ILU(2) is the best. The relatively poor performance of ILU(2) for the smaller time steps is due to the overhead of its construction time. Jac-GMRES requires almost the same CPU time as the ILU (2) preconditioner when Δt=5 *s*, however the difference increases with increasing time step and the ILU preconditioners are better than Jacobi by a factor of approximately 5 when Δt=100 *s*.

For BiCGSTAB, ILU(0) consistently shows better performance than other preconditioners for all time steps. ILU(0) is better than ILU(2) by a factor of 2 when Δt=5 *s* and 1.2 when Δt=100 *s*. Jac-BiCGSTAB requires the same CPU time as the ILU(1) preconditioner when Δt=5 *s*, however again the difference increases with increasing the time step and the ILU preconditioners far better than Jacobi by a factor of approximately 6 when Δt=100 *s*.

When comparing GMRES and BiCGSTAB preconditioned by ILU(0) in Figure 3.6, BiCGSTAB is better than GMRES by a factor of 1.25 when Δt=5 *s*, but this factor increases to 1.39 when Δt=100 *s*. The computational time of preconditioned GMRES increases almost linearly with the increase in time step. However this increase is less regular for BiCGSTAB.

3.9 EFFECT OF MESH REFINEMENT ON CONVERGENCE

To investigate the effects of the mesh refinement on the computational work of the newly preconditioned Krylov solvers, two more refined meshes are generated. Mesh2 (M2) and Mesh3 (M3) are generated by adding one additional node in each element in the original mesh (M1) and mesh M2 respectively. M2 and M3 contain 32169 and 101174 nodes respectively which are 3.25 and 10.2 times the number of nodes of M1. Similar to the original mesh, the flow is simulated untill 100 *s* then the Jacobian matrix and the residual vector are obtained for two time steps, 10 and 50 seconds. The ILU(0), ILU(1) and ILU(2) preconditioned GMRES and BiCGSTAB solvers are used for this test. The numbers of iteration required by the solvers to reduce the original linear residual by two orders of magnitude for all meshes are shown in Table 3.5. Since the computational cost of each iteration is proportional to the number of nodes, the equivalent numbers of iteration of mesh 1 are presented. From the table, a relationship of the following form is established

$$C_w \alpha N_n^{\beta};$$

where C_w = computational work, N_n = number of nodes and β = exponent.

The values of β have been computed based on mesh 1 and are presented in Table 3.6. β varies between 1.27 to 1.42 and 1.21 to 1.53 for GMRES and BiCGSTAB respectively with the ILU preconditioners. The average is 1.37

approximately for both the subspace methods. The variation in β is found to be higher for BiCGSTAB than GMRES.

3.10 CONCLUSION

In this chapter the convergence of the GMRES and BiCGSTAB iterative methods preconditioned by Jacobi, Symmetric Gauss-Seidel (SGS) and ILU with levels of fill varying from zero to two is presented for the finite element open channel flow model River2D. A new technique is developed to improve the convergence of the ILU preconditioned solvers. Thereafter a comparison among different solvers is performed. This shows that in most cases the ILU(0) preconditioner is better than the other preconditioners. Finally a mesh refinement analysis for the ILU preconditioned GMRES and BiCGSTAB is performed. The computational work is found to be approximately proportional to N_n^{β}, where N_n is the number of nodes and average values of β are 1.37.

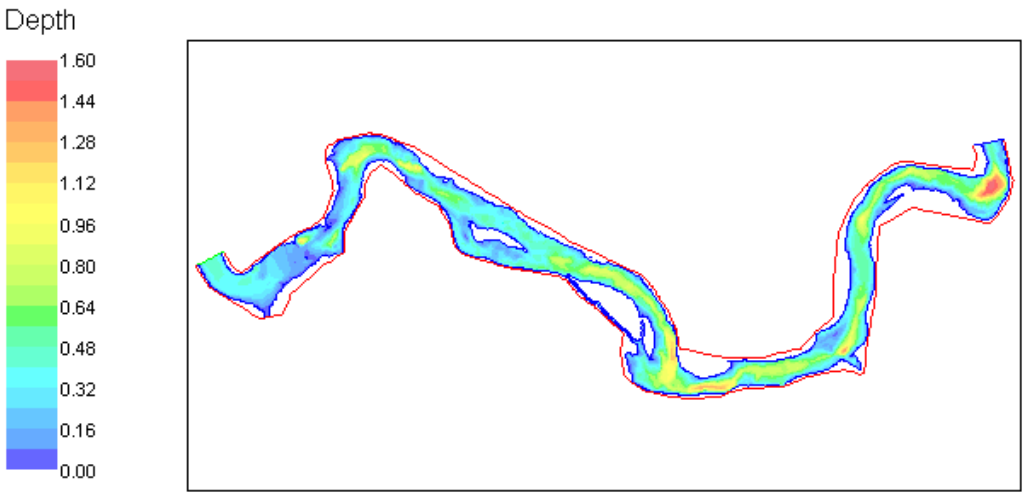

Figure 3.1 Contour of water depth in the natural channel

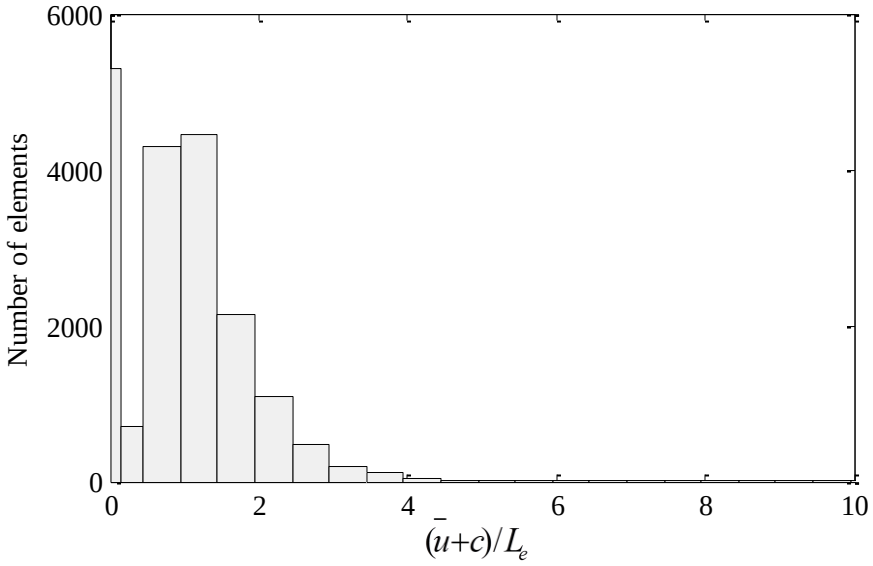

Figure 3.2 Histogram of number of elements

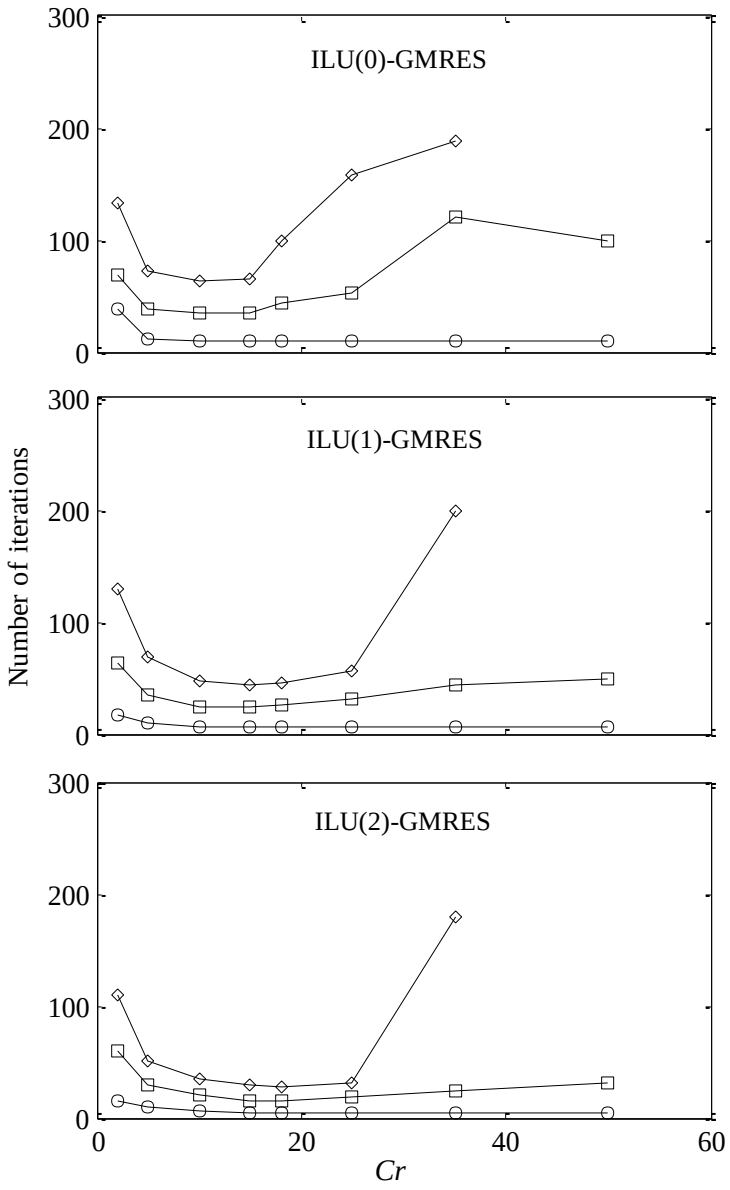

Figure 3.3 Effect of LCIP for GMRES with time steps (circle: 10 *s*, square: 50 *s* and diamond: 100 *s*)

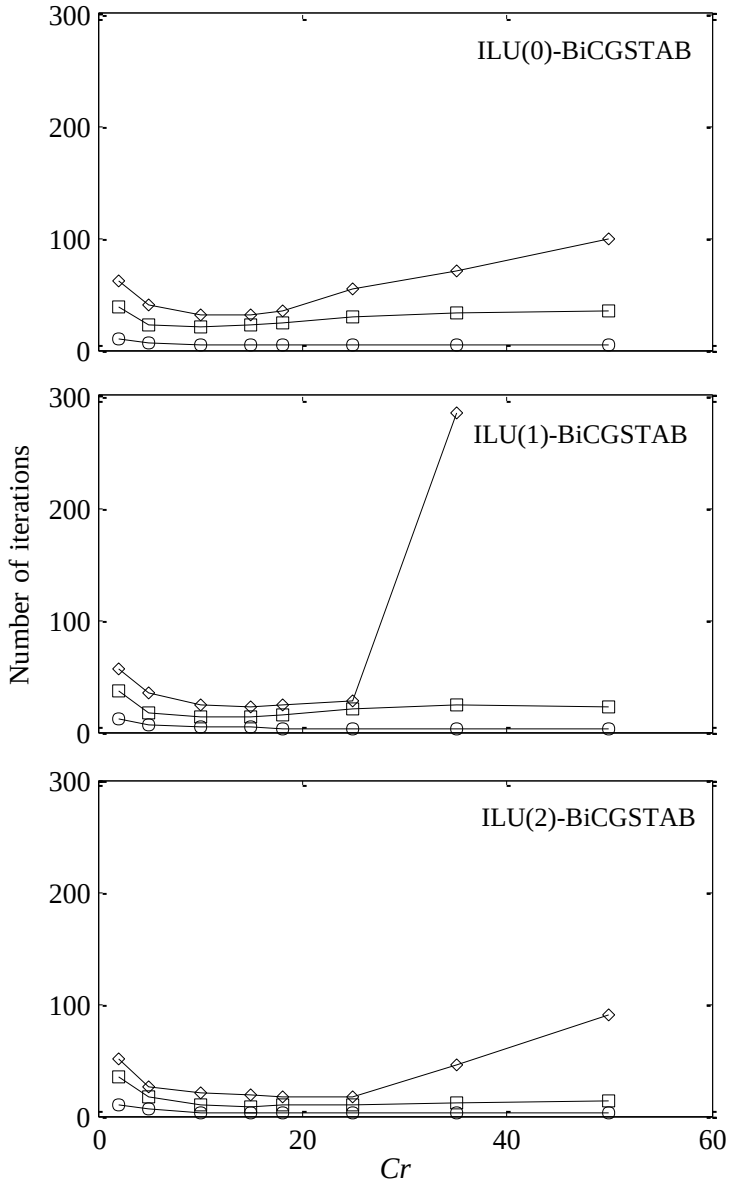

Figure 3.4 Effect of LCIP for BiCGSTAB with time steps (circle: 10 *s*, square: 50 *s* and diamond: 100 *s*)

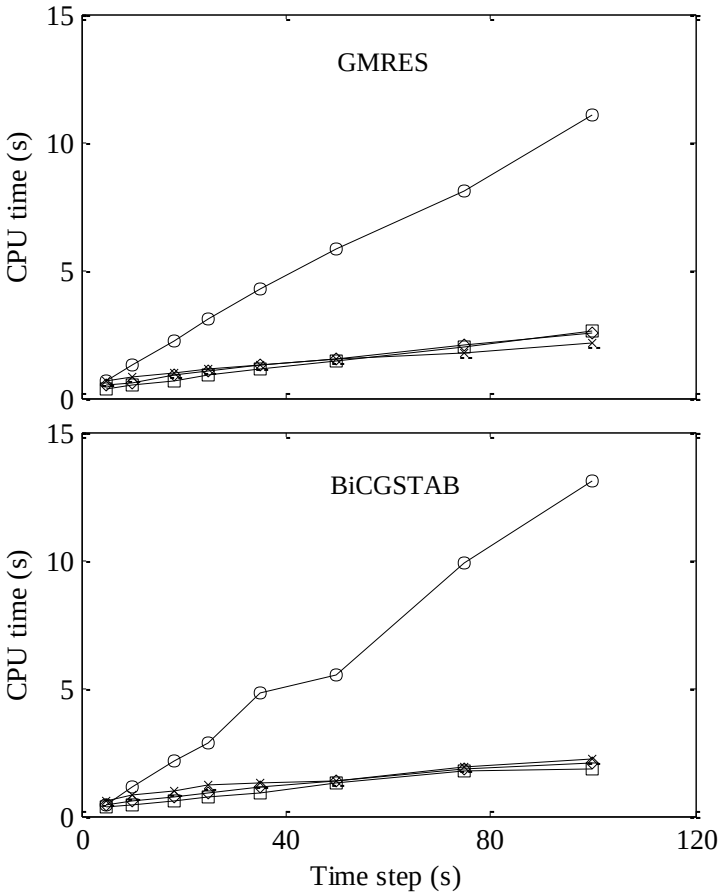

Figure 3.5 Comparison of computational work of preconditioned solvers. (circle: Jacobi, square: ILU(0), diamond: ILU(1), x: ILU(2))

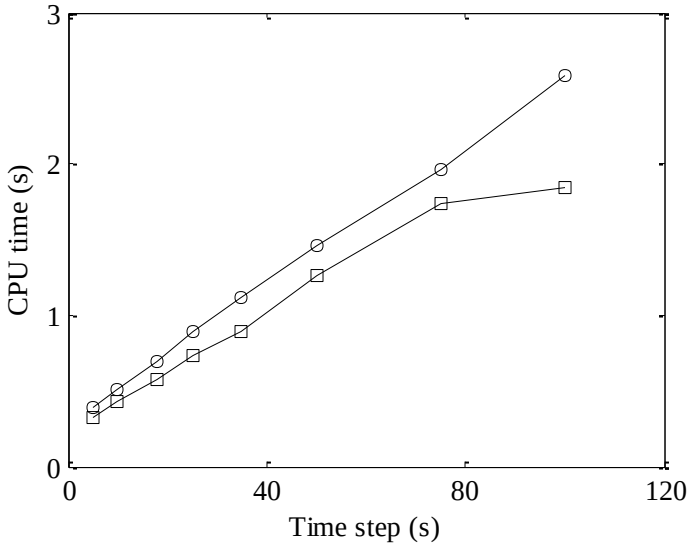

Figure 3.6 Comparison of computational work of GMRES (circle) and BiCGSTAB (diamond) preconditioned by ILU(0)

Table 3.1 Convergence behavior of GMRES with different preconditioners.

Time step (s)	Jacobi	ILU(0)	ILU(1)	ILU(2)
5	34	6	4	3
10	68	9	6	5
18	119	14	9	7
25	164	20	14	10
35	227	38	31	15
50	309	79	46	22
75	429	S	S	S
100	588	S	S	S

Table 3.2 Convergence behavior of BiCGSTAB with different preconditioners.

Time step (s)	Jacobi	ILU(0)	ILU(1)	ILU(2)
5	24	3	2	2
10	64	5	3	3
18	118	11	5	5
25	160	15	10	6
35	265	29	12	9
50	306	32	29	15
75	549	105	S	S
100	726	142	S	S

Table 3.3 Comparison of convergence of LCIP and original preconditioner for GMRES.

Time step	GMRES		
(s)	ILU(0	ILU(1	ILU(2
5	6 (6)	4 (4)	3 (3)
10	9 (9)	6 (6)	5 (5)
18	14	10 (9)	8 (7)
25	19	14	10
35	25	19	13
50	34	24	16
75	47 (S)	34 (S)	20 (S)
100	63 (S)	43 (S)	27 (S)

Table 3.4 Comparison of convergence of LCIP and original preconditioner for BiCGSTAB.

Time step	BICGSTAB		
(s)	ILU(0	ILU(1	ILU(2
5	3 (3)	2 (2)	2 (2)
10	5 (5)	3 (3)	3 (3)
18	8 (11)	6 (5)	5 (5)
25	11	8 (10)	7 (6)
35	14(29)	11	8 (9)
50	21	14	9 (15)
75	30	20 (S)	15 (S)
100	32	23 (S)	18 (S)

Table 3.5 Equivalent number of iterations required for different meshes

		ILU(0)		ILU(1)		ILU(2)	
		Δt=10s	Δt=50s	Δt=10s	Δt=50s	Δt=10s	Δt=50s
GMRES	M1	9	34	6	24	5	16
	M2	49	186	33	108	23	85
	M3	225	930	143	572	102	378
BiCGSTAB	M1	5	21	4	14	3	9
	M2	29	91	20	59	16	46
	M3	137	714	92	419	56	225

Table 3.6 Values of exponent β for mesh sensitivity analysis

		ILU(0)		ILU(1)		ILU(2)	
		Δt=10s	Δt=50s	Δt=10s	Δt=50s	Δt=10s	Δt=50s
GMRES	M1 to 2	1.44	1.44	1.43	1.27	1.29	1.42
	M1 to 3	1.39	1.42	1.36	1.36	1.29	1.36
BiCGSTAB	M1 to 2	1.49	1.24	1.34	1.21	1.42	1.38
	M1 to 3	1.43	1.53	1.35	1.46	1.26	1.39

Chapter 4

COUPLED DA-RANS MODEL FOR 2D PLANE FLOW

4.1 INTRODUCTION

The coupled DA-RANS model proposed in chapter 2 is aimed for modeling flow in rivers. However it is important to understand their behavior in simple yet representative flow problems. Therefore the objective of this chapter is to develop, test and verify the proposed model for two dimensional plane flow i.e. flow varies only in the longitudinal and vertical directions with no variation in the transverse direction. The governing depth averaged and the RANS equations and the relationship between the two sets of equations are presented in Section 4.2. Also the rationale for using a correction term for mass balance is explained. In section 4.3 the RANS equations in the generalized boundary-fitted coordinate are presented and a description of the numerical techniques is provided. Subsequently the details of the modeling technique of the correction term and the overall solution procedure are outlined. Next the results of the numerical model are presented and compared with the experimental results along with an analysis of

the behavior of the correction term. Finally the computational efficiency of the proposed model is presented.

4.2 MATHEMATICAL MODEL

Figure 4.1 shows the two-dimensional vertical plane flow where the flow variables vary only in the *x* and *z* directions. The continuity and the RANS *x* and *z* momentum equations in the conservative form can be written as

$$\frac{\partial u}{\partial x}+\frac{\partial w}{\partial z}=0 \qquad [4.1]$$

$$\frac{\partial u}{\partial t}+\frac{\partial u^2}{\partial x}+\frac{\partial uw}{\partial z}+\frac{1}{\rho}\frac{\partial p}{\partial x}-\frac{1}{\rho}\left(\frac{\partial \sigma_x}{\partial x}+\frac{\partial \tau_{zx}}{\partial z}\right)=0 \qquad [4.2]$$

$$\frac{\partial w}{\partial t}+\frac{\partial uw}{\partial x}+\frac{\partial w^2}{\partial z}+\frac{1}{\rho}\frac{\partial p}{\partial z}-\frac{1}{\rho}\left(\frac{\partial \tau_{zx}}{\partial x}+\frac{\partial \sigma_z}{\partial z}\right)+g=0 \qquad [4.3]$$

Boundary conditions at the water surface

Kinematic:

$$w_h=\frac{\partial h}{\partial t}+u_h\frac{\partial (h+z_b)}{\partial x} \qquad [4.4]$$

Dynamic:

$$p_h = \tau_h = \sigma_h = 0 \qquad [4.5]$$

subscript *h* indicates the quantities at the water surface.

Boundary conditions at the bed

Kinematic:

$$w_b = u_b \frac{\partial z_b}{\partial x} \qquad [4.6]$$

Dynamic:

$$\tau_{bx} = -\sigma_x \frac{\partial z_b}{\partial x} + \tau_{xz} \qquad [4.7]$$

The bed stress is computed by the 'law of the wall' as described in chapter 2.

The pressure Poisson equation derived from equation [4.1] to [4.3] can be written as

$$\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial z^2} = 2\rho\left(\frac{\partial u}{\partial x}\frac{\partial w}{\partial z} - \frac{\partial u}{\partial z}\frac{\partial w}{\partial x}\right) + \left(\frac{\partial^2 \sigma_x}{\partial x^2} + 2\frac{\partial^2 \tau_{xz}}{\partial x \partial z} + \frac{\partial^2 \sigma_z}{\partial z^2}\right), \qquad [4.8]$$

The solution of the Poisson equation requires boundary conditions at all the boundaries. For this study, zero non-hydrostatic pressure will be provided at all the boundaries except at the bottom where the pressure will be computed from the *z* momentum equation.

The depth averaged equations for one dimension can be written as

$$\frac{\partial h}{\partial t}+\frac{\partial h\bar{u}}{\partial x}=0 \qquad [4.9]$$

Depth averaged (DA) *x* momentum equation

$$\begin{aligned}&\frac{\partial h\bar{u}}{\partial t}+\frac{\partial h\bar{u}^{2}}{\partial x}+\frac{\partial h\overline{u^{p}u^{p}}}{\partial x}+gh\frac{\partial(h+z_b)}{\partial x}\\&+\frac{1}{\rho}\left(\frac{\partial h\overline{p^{p}}}{\partial x}+p_b^{p}\frac{\partial z_b}{\partial x}\right)-\frac{1}{\rho}\left(\frac{\partial h\overline{\sigma_x^{d}+\sigma_x^{p}}}{\partial x}\right)+\frac{1}{\rho}\left(\tau_{bx}^{d}+\tau_{bx}^{p}\right)=0,\end{aligned} \qquad [4.10]$$

For the hydrostatic model of this study, first equations [4.9] and [4.10] will be solved neglecting the primed terms to obtain *h* and $\bar{u}$. Then the water surface will be used as a fixed upper boundary in solving equation [4.2] neglecting the non-hydrostatic pressure together with equation [4.1] to obtain the velocity profiles. When the non-uniform velocity is neglected in equation [4.10], it creates a difference with equation [4.2]. Another difference between these two equations stems from the modeling of the stress terms. In equation [4.2] the bed stress is computed using near bed velocity and the normal stress is computed using point velocity. On the other hand in the DA model, all the stresses are computed using the DA velocity. Due to these differences the water surface and velocity obtained from such a depth averaged model contain some error. When this water surface is used as a fixed upper boundary to solve the RANS equations to the steady state, the horizontal discharge computed from the simulated horizontal velocity profiles may not conserve the mass balance. Therefore a modification in the *x* momentum RANS equation is needed, such that the profile of *u* obtained gives a mean equal

to the DA velocity from the DA computations. Once the velocity profiles and stresses are obtained from the RANS hydrostatic model (RANS-H), they can be used in equation [4.10] to recompute the water surface that can again be used for the RANS model.

For the non-hydrostatic model of this study, first the water surface elevation and the depth averaged velocity will be computed from the depth averaged model neglecting the primed terms. Then the water surface will be used as a fixed upper boundary in solving the RANS *x* momentum and the continuity equation together with a pressure Poisson equation for the non-hydrostatic pressure assuming a zero pressure at the surface. The pressure at the nodes near the bottom will be computed from the *z* momentum equation. A correction term will also be added to the *x* momentum equation for the mass balance. Similarly for the RANS non-hydrostatic (RANS-NH) model, additionally non-hydrostatic pressure will be used in equation [4.10] with the feedback between the depth averaged and the RANS-NH models.

Based on the above discussion a mathematical relationship between the correction term and the primed terms in equation [4.10] can be established. Let the effect of the primed terms be lumped in a term *hX*, where *X* is the correction. When the primed terms are neglected, equation [4.10] can be written as

$$\frac{\partial h\bar{u}}{\partial t}+\frac{\partial h\bar{u}^2}{\partial x}+\frac{\partial h\overline{u^p u^p}}{\partial x}+gh\frac{\partial (h+z_b)}{\partial x}$$
$$+\frac{1}{\rho}\left(\frac{\partial h\overline{p^p}}{\partial x}+p_b^p\frac{\partial z_b}{\partial x}\right)-\frac{1}{\rho}\left(\frac{\partial h\overline{\sigma_x^d+\sigma_x^p}}{\partial x}\right)+\frac{1}{\rho}\left(\tau_{bx}^d+\tau_{bx}^p\right)-hX=0, \qquad [4.11]$$

Therefore,

$$X \equiv \frac{1}{h}\left[\frac{\partial h\overline{u^p u^p}}{\partial x} - \frac{1}{\rho}\left(\frac{\partial h\overline{\sigma_x^p}}{\partial x}\right) + \frac{1}{\rho}\tau_{bx}^p\right] \qquad [4.12]$$

for the hydrostatic model and

$$X \equiv \frac{1}{h}\left[\frac{\partial h\overline{u^p u^p}}{\partial x} + \frac{1}{\rho}\left(\frac{\partial h\overline{p^p}}{\partial x} + p_b^p\frac{\partial z_b}{\partial x}\right) - \frac{1}{\rho}\left(\frac{\partial h\overline{\sigma_x^p}}{\partial x}\right) + \frac{1}{\rho}\tau_{bx}^p\right] \qquad [4.13]$$

for the non-hydrostatic model

The equivalent *x* momentum RANS equation can then be written as

$$\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x} + \frac{\partial uw}{\partial z} + \frac{1}{\rho}\frac{\partial p^p}{\partial x} + g\frac{\partial H}{\partial x} - \frac{1}{\rho}\left(\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{zx}}{\partial z}\right) - X = 0 \qquad [4.14]$$

Therefore the pressure Poisson equation is modified as

$$\frac{\partial^2 p^p}{\partial x^2} + \frac{\partial^2 p^p}{\partial z^2} = 2\rho\left(\frac{\partial u}{\partial x}\frac{\partial w}{\partial z} - \frac{\partial u}{\partial z}\frac{\partial w}{\partial x}\right) + \left(\frac{\partial^2 \sigma_x}{\partial x^2} + 2\frac{\partial^2 \tau_{xz}}{\partial x \partial z} + \frac{\partial^2 \sigma_z}{\partial z^2}\right) + \rho\frac{\partial X}{\partial x} - \rho g\frac{\partial^2 H}{\partial x^2} \qquad [4.15]$$

A modeling technique will be developed to compute the correction term such that the mean of the computed velocity profile in any vertical becomes equal

to the depth averaged velocity in that vertical. The Reynolds normal stresses in the RANS equations will be neglected as they are relatively small as compared to the shear stress due to the small horizontal velocity gradient and vertical velocity in open channel flow. The depth averaged normal stress will also be neglected for the similar reason as the Reynolds normal stresses.

4.3 NUMERICAL MODEL

For the depth averaged computation, equations [4.9] and [4.10] will be solved. All of the normal stresses are neglected for this study. Depth averaged modeling is accomplished by the McCormack scheme (Discretization by the McCormack scheme is provided in appendix B). For a subcritical upstream boundary, a known discharge is specified. For a subcritical downstream boundary, a known water depth is used. For a supercritical upstream boundary both water level and discharge are specified while no boundary condition is needed for a supercritical downstream boundary. The bed stress is computed from

$$\tau_{bx}^{d} = \frac{\bar{u}\left|\bar{u}\right|}{C_{*}^{2}} \qquad [4.16]$$

For the RANS model, the RANS x momentum and the pressure Poisson equations for incompressible fluid including the correction term (X) in the non-conservative form in the non-orthogonal boundary-fitted coordinate system are

solved neglecting the normal stress. In transforming the equations from Cartesian (x,z) to generalized boundary-fitted coordinate (ξ,ζ), ζ direction coincides with the vertical direction while ξ follows the bed and water surface (Figure 4.2). These equations after transforming from Cartesian to non-orthogonal boundary-fitted coordinate by the partial transformation approach *i.e* leaving the velocity components in Cartesian coordinate can be expressed as,

The x momentum equation,

$$\frac{\partial u}{\partial t}+U_c\frac{\partial u}{\partial \xi}+W_c\frac{\partial u}{\partial \zeta}=-\frac{1}{\rho}(\xi_x\frac{\partial p}{\partial \xi}+\zeta_x\frac{\partial p}{\partial \zeta})+\frac{1}{\rho}\left(\zeta_z\frac{\partial \tau_{xz}}{\partial \zeta}\right)+X \qquad [4.17]$$

The pressure Poisson equation,

$$\begin{aligned}&\left(\xi_x\frac{\partial}{\partial \xi}+\zeta_x\frac{\partial}{\partial \zeta}\right)\left(\xi_x\frac{\partial p^p}{\partial \xi}+\zeta_x\frac{\partial p^p}{\partial \zeta}\right)+\left(\xi_z\frac{\partial}{\partial \xi}+\zeta_z\frac{\partial}{\partial \zeta}\right)\left(\xi_z\frac{\partial p^p}{\partial \xi}+\zeta_z\frac{\partial p^p}{\partial \zeta}\right)\\&=2\rho\left(\xi_z\frac{\partial w}{\partial \xi}+\zeta_z\frac{\partial w}{\partial \zeta}\right)\left(\xi_x\frac{\partial u}{\partial \xi}+\zeta_x\frac{\partial u}{\partial \zeta}\right)-2\rho\left(\xi_x\frac{\partial w}{\partial \xi}+\zeta_x\frac{\partial w}{\partial \zeta}\right)\left(\xi_x\frac{\partial u}{\partial \xi}+\zeta_x\frac{\partial u}{\partial \zeta}\right)\\&+2\left(\xi_x\frac{\partial}{\partial \xi}+\zeta_x\frac{\partial}{\partial \zeta}\right)\left(\xi_z\frac{\partial \tau_{xz}}{\partial \xi}+\zeta_z\frac{\partial \tau_{xz}}{\partial \zeta}\right)+\rho\left(\xi_x\frac{\partial X}{\partial \xi}+\zeta_x\frac{\partial X}{\partial \zeta}\right)\\&-\rho g\left(\xi_x\frac{\partial H}{\partial \xi}+\zeta_x\frac{\partial H}{\partial \zeta}\right)\end{aligned} \qquad [4.18]$$

The z momentum equation,

$$U_c\frac{\partial w}{\partial \xi}+W_c\frac{\partial w}{\partial \zeta}=-\frac{1}{\rho}(\zeta_z\frac{\partial p^p}{\partial \zeta})+\frac{1}{\rho}\left(\xi_x\frac{\partial \tau_{xz}}{\partial \xi}+\zeta_x\frac{\partial \tau_{xz}}{\partial \zeta}\right) \qquad [4.19]$$

where U_c and W_c = contravariant velocity components ($U_c = u\xi_x + w\xi_z$ and $W_c = u\zeta_x + w\zeta_z$), ξ_x, ξ_z, ζ_x and ζ_z = components of the transformation matrix. (The equations for the components are provided in appendix A)

The RANS *x* momentum equation in the boundary-fitted coordinate system is modeled and discretized by a finite difference method. The convective terms are discretized a first order upwind method while the pressure gradient terms are approximated by a centered difference method. The gradient terms in the ξ direction are discretized explicitly with an implicit coupling in the ζ direction. This allows us to solve each vertical independently using the values of the neighboring verticals from the previous time step and avoids the solution of a large matrix, rather a tridiagonal matrix of size *m* is solved for each vertical, where *m* is the number of nodes in the vertical. The shear stress is computed by the eddy viscosity hypothesis. The eddy viscosity is computed by the mixing length model (equation [2.34]) where equation [2.36] is used for the mixing length. At the upstream boundary, vertical profile of *u* is specified and at the downstream boundary *u* is extrapolated from the interior nodes. Logarithmic velocity profiles based on the depth averaged velocities are specified as the initial condition at each vertical.

The RANS continuity equation is solved for the vertical velocity using known horizontal velocity by a control volume approach. The faces of the control volumes are located at the center of two adjacent nodes. The horizontal mass flux through a cell face is computed by a centered difference from *u* of two adjacent

nodes. Since the faces of the control volumes on two sides are vertical, vertical mass flux through these faces is zero and the flux through other two the faces is calculated by setting total mass flux through the bed equal to zero. At the upstream boundary vertical velocity is set to zero while at the downstream boundary *w* velocity profile is extrapolated from the interior nodes.

The pressure Poisson equation in the boundary-fitted coordinate is modeled and discretized by a finite difference method. At the upstream, downstream and water surface boundaries non-hydrostatic pressure is set to zero. For the bottom boundary nodes the *z* momentum is solved in the boundary-fitted coordinate also by a finite difference method. The pressure gradient term is discretized by a forward difference while all other terms are discretized by a centered difference.

4.4 MODELING THE CORRECTION TERM

In order to model the correction term, an approximate equation has been developed as follows.

Let us consider equation [4.17]. Considering the numerical scheme for the RANS model, this equation can be written as

$$\frac{u^{n+1}-u^{n}}{\Delta t}+\left(U_c\frac{\partial u}{\partial \xi}\right)^{n}+{W_c}^{n}\left(\frac{\partial u}{\partial \zeta}\right)^{n+1}=-\frac{1}{\rho}\left(\xi_x\frac{\partial p}{\partial \xi}+\zeta_x\frac{\partial p}{\partial \zeta}\right)^{n}+\frac{1}{\rho}\left(\zeta_z\frac{\partial \tau_{xz}}{\partial \zeta}\right)^{n+1}+X; \qquad [4.20]$$

With a guessed value of the correction term X_{gs}, equation [4.17] can be written as

$$\frac{u_g^{n+1}-u^{n}}{\Delta t}+\left(U_c\frac{\partial u}{\partial \xi}\right)^{n}+{W_c}^{n}\left(\frac{\partial u}{\partial \zeta}\right)_g^{n+1}=-\frac{1}{\rho}\left(\xi_x\frac{\partial p}{\partial \xi}+\zeta_x\frac{\partial p}{\partial \zeta}\right)^{n}+\frac{1}{\rho}\left(\zeta_z\frac{\partial \tau_{xz}}{\partial \zeta}\right)_g^{n+1}+X_{gs}; \qquad [4.21]$$

where superscripts n and $n+1$ refer to the solution at the current and next time level, the subscript gs refers to the values computed using X_{gs} and Δt is the discrete time step.

Subtracting equation [4.21] from equation [4.20], assuming that in the vertical direction the respective velocity and shear stress gradients in both equations are the same and integrating the remaining terms over the flow depth give the following relationship

$$X=\frac{<u^{n+1}>-<{u_g}^{n+1}>}{\Delta t}+X_g \qquad [4.22]$$

where the notation < > indicates that the velocity profiles are averaged over the depth. Since the purpose of the correction term is to adjust the velocity profile such that it gives a mean equal to the DA velocity computed from the DA model, therefore $< u^{n+1} >= \bar{u}$.

Due to the approximations made in this derivation more than one iteration may be required to obtain the appropriate value of the correction term. Therefore the above expression can be written as

$$X_{k+1} = \frac{\bar{u} - < u_k^{n+1} >}{\Delta t} + X_k \qquad [4.23]$$

where subscript *k* refers to the iteration level for the correction term. The iteration procedure is described in the next section.

4.5 SOLUTION PROCEDURE

Steady DA model:

For the depth averaged model, a time marching procedure is used to obtain the steady state solution. Since the McCormack scheme is explicit, time step should satisfy the Courant-Friedrichs-Lewy (CFL) criteria *i.e.*

$$N_{cfl} = \frac{\Delta t}{\Delta x}\left(\bar{u} + \sqrt{gh}\right) \leq 1$$

The steady state solution is achieved when the L_2 norms of the residuals of the depth averaged continuity and momentum equations are reduced to 10^{-4}.

Steady RANS model:

In the RANS model, at first the *x* momentum equation is solved by a time marching procedure. For each vertical a local time step is computed from a fixed Courant number (*Cr*),

$$\Delta t_i = \frac{Cr}{\max(U_{ci})}$$

where *i* refers to a particular vertical, U_{ci} = contravariant velocities at all nodes of the vertical *i*. For stability *Cr* must be less than or equal to 1.

At the beginning of the simulation, first the set of the discretized *x* momentum equations for each vertical is solved for one time step after setting the values of *X* equal to zero. During this simulation if the mean of the computed *u* velocity profile is not within a specified tolerance of their corresponding depth averaged velocities, equation [4.23] is used to compute *X* and the velocity profiles are recomputed. This procedure is repeated until the mean of the computed velocity profiles are within a specified tolerance (ε_k) of their depth averaged velocities. The tolerance is set as

$$\varepsilon_k = \frac{\left|\bar{u} - \bar{u}_k^{n+1}\right|}{\left|\bar{u}\right|} \leq \frac{\varepsilon}{10};$$

ε is the relative change in *u* velocity in previous two successive time steps and computed as

$$\varepsilon = \sqrt{\frac{\sum(\delta u)^2}{\sum u^2}}$$

Once the *x* momentum equation for all the verticals are solved for one time step, the RANS continuity equation is solved for the vertical velocity distributions by a control volume approach. For any vertical column of control volumes, computation starts from the bottom control volume and the vertical mass flux at the top face is computed explicitly setting the total flux to zero at the bed. This allows us to compute the vertical flux at the top face of the next control volume explicitly as the flux through all other faces are now known and continues up to the water surface. The nodal value of vertical velocity is computed from the linear interpolation of vertical velocity at two adjacent faces. As the mean of the computed *u* velocity profile conserves the continuity of the mean flow and the no flow condition is specified at the bottom, the computed vertical velocity at the water surface would satisfy the water surface kinematic condition automatically.

For the non-hydrostatic RANS model, then the pressure Poisson and the *z* momentum equations are also solved for the non-hydrostatic pressure (NHP). For the Poisson equation a vertical line implicit approach is used. After solving the

Poisson equation for one vertical, the *z* momentum equation is solved for the NHP at the bottom node of that vertical. In the vertical line implicit method the NHP at any vertical is expressed implicitly while latest available values of pressure are used for the neighboring nodes. First a sweep starts from upstream to downstream followed by a downstream to upstream sweep and the sweeps continues until either the ratio of current residual to the residual at the beginning of the line implicit approach of the Poisson equation is less than a certain tolerance or the total number of sweeps reaches a certain value. An under-relaxation may be needed for the computed pressure for stability.

The procedure is then advanced to the new time level using the newly computed values of u, w, p' and X and the above procedure is repeated. This solution procedure continues until the solution converges to the steady state. The convergence to the final steady state solution of the RANS model is achieved when the L_2 norm of residuals of the x momentum equation (L_2 (u)) becomes less than 10^{-4}.

Coupling between the DA and the RANS models:

The solution of both depth averaged and RANS to the steady state is named as N iteration. When one N is completed the perturbation terms are computed and feed back to the depth averaged equation to recompute the water level and depth averaged velocity. Then the RANS model is again solved using

the newly computed water level and depth averaged velocity. The whole solution procedure stops when the relative average change (ε_N) in water depth in two successive *N* iteration is less than 0.001. ε_N is computed as

$$\varepsilon_N = \sqrt{\frac{\sum \delta h^2}{\sum h^2}}$$

where δh is the difference in the water depths in two successive *N* iterations.

The whole solution procedure described above can be summarized as follows:

Step1: Solve DA model to the steady state neglecting the perturbation terms.

Step2: Solve RANS model to the steady state by solving:

a) the RANS *x* momentum equation for one time step including the modeling of the correction term
b) the RANS continuity equation
c) the Poisson equation (for RANS-NH model)

Repeat (a-c) until the steady state solution is achieved

Repeat step 1 (now including the perturbation terms) and 2 until converged.

The flow diagram of the solution procedure is given in Figure 4.3.

4.6 MODEL VERIFICATION

4.6.1 Case 1: Flow development in a rectangular channel

Ranga Raju *et al.* (2000) performed several experiments to investigate the flow development in a rectangular open channel with both smooth and rough beds. A glass-walled flume 15 m long, 0.75 m wide and 0.5 m deep was used. Of the experiments, run 8 and run 15 were conducted with smooth and rough beds respectively and are used for the model verification. For the depth averaged computation the inflow discharge is specified at the inflow boundary. For the downstream boundary, water depths of 0.078 m and 0.082 m are specified at the outlet for the smooth and rough beds respectively. For the RANS model, uniform velocity profile is specified at the upstream boundary. The bed roughnesses of 0.0003 m and 0.005 m are used for the smooth and rough beds respectively. In this case a hydrostatic pressure distribution is assumed. A horizontal spacing of 0.05 m with 15 nodes in the vertical direction is used giving a total of 4515 nodes. In the vertical direction the node spacing is smaller near the bed and increases towards the water surface. Three other meshes with a horizontal spacing of 0.05 m with 20 nodes in the z direction, 0.02 m in the horizontal and 15 and 20 nodes in the z directions have also been tested to ensure that the mesh used gives mesh independent solution by comparing the u velocity profiles at different locations.

First the behavior of the correction term and the effects of the perturbation terms on water level and correction term are analyzed. Figure 4.4 shows the correction term (X) along with the water level for the rough bed. The value of X is

high near the upstream boundary of the flow and gradually reduces as the flow moves downstream and then becomes relatively small after 2.0 m from the upstream boundary. This happens because a uniform velocity profile is used as the upstream boundary condition for the RANS model. For this flow development velocity profiles gradually change shapes from uniform to logarithmic profiles. Therefore the bed stress computed by the RANS model is higher than the bed stress computed by the DA model in the initial portion of the channel. The gradient of the dispersion term is also important in this region as the shape of the flow profiles change. When the water surface is corrected for the dispersion term and the bed stress separately in the DA model, the value of X near the entrance is reduced by approximately 40% and 60% respectively. The value of X is reduced to almost zero when both dispersion and bed stress are used together to correct the water surface. A comparison of the water levels for N=1 and 2 indicates that the relative change is 0.8% and both dispersion and bed stress increases the water level. Further correction by the new results from the RANS model does not change the water level by more than the tolerance

Figure 4.5a shows the vertical distribution of u velocity for the rough bed for N=1 and 2 which are compared with the experimental results. At x=1 m the whole profile changes its shape from uniform and further downstream it tend to become almost logarithmic. The model is also able to predict the profiles at all sections reasonably well. However the velocity profile at x=1 m is slightly under

predicted at the upper part and over predicted at the lower part of the profile. The relative differences between the simulated profiles for N=1 and 2 are negligible.

Figure 4.5b shows the vertical distribution of u velocity for the smooth bed for N=1 and 2. For this case the flow development is relatively slow. The model can predict the velocity profiles at all the three sections very well. At x=1 m the velocity profile changes from uniform only upto 0.2 m from bed above which the flow is almost uniform and the model is able to predict this changes quite well. After this section the velocity profiles gradually changes towards logarithmic profile. The relative differences between the simulated profiles for N=1 and 2 are also negligible.

4.6.2 Case 2: Flow over a symmetric hump

The experimental data of Sivakumaran *et at*. (1983) over a symmetric hump consists of water level and bed pressure. The experiment was conducted in a horizontal flume 9.15 m long, 0.3m wide and 0.65m high. The shape of the symmetric profile was created according to a normal distribution with a height of 0.2 m at the center (Figure 4.6b). The length of the hump was 1.2 m. This case is an example of transcritical flow where the non-hydrostatic pressure is very significant. Therefore the RANS non-hydrostatic model is used to simulate the flow variables. For the depth averaged computations, inflow boundary is specified with a constant discharge of 0.11 $m^3/s/m$. Since the downstream flow is supercritical no boundary condition is required. For the RANS model, a

logarithmic velocity profile is specified at the upstream boundary. A horizontal spacing of 0.02 m and 15 nodes in the vertical direction with smaller spacing near the bed is used for the simulation. For a mesh independence test, velocity profiles at different locations are compared for different mesh density. Further reduction in spacing both in the vertical and horizontal direction does not change the results.

Figure 4.6a and 4.6b show the correction term and the water level. The value of X is relatively high as compared to the first case. In this case non-hydrostatic pressure is significant and its effect on the correction term is the largest. When the water level is corrected by the dispersion, bed stress and non-hydrostatic pressure the correction term is reduced by one order of magnitude in two corrections (N=3). This figure also shows that when the water level is corrected, the agreement between the experimental and numerical results is very good.

The computed pressure for N=3 is compared with the experimental results in Figure 4.6c. The RANS model can predict the non-hydrostatic pressure quite well such that the total pressure around the hump crest and further downstream agrees well with the experimental data. In the upstream where non-hydrostatic pressure is not significant, total pressure shows a similar agreement as the water level.

4.6.3 Case 3: Flow over a dune

Van Mierlo and de Ruiter (1988) carried out a series of experiments for the flow field in a trench. The experimental flume had a width of 1.5 m with a slope of 0.00094 m/m. Dunes of 0.08 m high and 1.6 m long were placed in the flume to create the trench. The lee side slope was 0.5 m/m which was followed by a horizontal bed of 0.1 m. Then the luff side bed starts and consists of three slopes of 0.031, 0.087 and 0.031 m/m with 0.25, 0.75 and 0.26 m in length followed by a horizontal bed of 0.08m (Figure 4.7). The bed was paved with material (d_{50}=0.0016m) that had effective roughness of 0.0025 m. The results of experiment T5 are used for model verification. For the numerical simulation bed profile of three continuous dunes with a 1 m horizontal bed in upstream and downstream is created. The inflow discharge of 0.11m^2/s and the downstream depth of 0.21 m were used as upstream and downstream boundary conditions respectively. This is probably one of the most complicated cases where recirculation of the flow occurs. For the RANS non-hydrostatic model, as obtained from a mesh independence analysis a horizontal spacing of 0.02 m with 30 nodes in the vertical direction is used with smaller spacing near the bed. Similar to other cases velocity profiles at different locations is used to perform the mesh independence test. For the *x* momentum equation logarithmic velocity profile is specified at the upstream boundary. The horizontal velocity profiles in the middle dune are presented here. In this case non-hydrostatic pressure is considered.

Figure 4.7b shows the correction term for the flow over the dune. The value of the correction term is quite large near the start of the leeslope and the change is quite sharp. Afterwards along the slope the value decreases. Again near the start of the horizontal bed, a large change is observed. Thereafter the values are relatively small and small wavy shapes are observed at locations where change in slope occurs. When the water surface is corrected by the perturbation terms the relative change in water depth is 0.4% (Figure 4.7c). The values of the correction term does not change significantly except in some portion of the leeslope where the change in slope does not occur and the value of the dispersion term seem to be relatively high. Further correction does not change the water depth by more than the tolerance.

To test the effect of difference in the numerical discretization between the DA and the RANS models, water level in the RANS model is discretized by a forward difference instead of the centered difference and the correction term is plotted in Figure 4.7b. It shows that this correction term is almost twice the value of the correction term of the original model at the start of the lee slope. Therefore it appears that the correction term does not only account for the mathematical difference but also accounts for the numerical difference between two models.

Figure 4.8 shows the comparison of the experimental and simulated u velocity profiles for N=1 and 2. Visually there is no difference between these two sets of results. Just downstream of the sill crest (x=0.06 m), the simulated profile is in reasonably good agreement with the experimental result including the reverse

flow. At x= 0.13 m the model result is also reasonable. But further downstream the strength of the recirculation is underpredicted by the present model. However as the flow moves through the luffside slope the difference between the experimental and numerical results tend to decrease and the model can predict the velocity profiles accurately at x=1.12 and 1.58 m.

To test the effect of the non-hydrostatic pressure on the flow over the dune, the simulated near bed velocity of both the RANS-H and RANS-NH models are compared in Figure 4.9. The RANS-NH model can predict a longer recirculation zone than the RANS-H model due to the adverse pressure gradient in this zone. However in the luff slope where non-hydrostatic pressure is insignificant both models produce the same results.

In general the RANS models are fairly good at predicting the velocity profiles in the flow development and the pressure for the flow over the symmetric hump. The feedback between the DA and the RANS model improved the results for the flow over the hump noticeably. However the compared velocity profiles did not change for the other cases. The model showed a good performance in predicting the velocity profiles in the converging section of the dune, but it could not reproduce the recirculation zone very well. In the recirculation zone the flow is more complex and a better turbulence model may be needed to accurately predict the length of the recirculation.

4.7 COMPARISON OF COMPUTATIONAL EFFORTS

To get an estimate of the computational efficiency of the current RANS models, a comparison between the DA and the RANS models in terms of CPU time required to obtain the steady state solutions is made. All the computer programs of the DA and the RANS models are written in Matlab® and run in a PC with a 2.8 GHz processor.

Case 1

Modeling the flow development in a rectangular channel in the smooth and the rough beds by the hydrostatic RANS model requires approximately 14 and 12 *s* (168 and 118 time steps) respectively which are the less than the time required by the depth averaged model (75 s, 4500 time steps) in N=1. The time required by the RANS-H in N=2 is 8 *s* and the DA model requires same computational time as in N=1.

Case 2

For modeling the flow over a symmetric hump RANS-NH model requires 75 *s* (440 time steps) and DA model requires 135 *s* (5500 time steps) in N=1. In each of N=2 and 3 RANS-NH require approximately 55 *s* (340 time steps) and DA requires approximately 130 *s*.

Case 3

For the flow over the dunes, RANS-NH requires 100 *s* (580 time steps) and DA requires 150 *s* (6500 time steps) in *N*=1. RANS-NH and DA requires 80 *s*(470 time steps) and 120 *s* (5200 time steps) respectively in *N*=2. The RANS-H model requires almost the same number of time steps but requires 75% of the computational time as the RANS-NH model due to the absence of the Poisson equation.

The number of time steps required by the RANS model is approximately 0.5 to 2 times the number of verticals with an explicit discretization in the longitudinal direction. On the other hand this factor is 10 to 20 for the solution of the depth averaged model by the explicit McCormack scheme. Although the tolerance for the steady RANS solution is set to 10^{-4}, a tolerance of 10^{-2} can be used instead as the residuals are never increasing (monotonically decreasing) with the time steps as opposed to the depth averaged model where the residuals show a periodic pattern. This would allow achieving the steady state RANS solution almost at half the computational time reported above. For the test cases verified here, any typical 3D free surface model with a sequential implicit iterative solver might show similar convergence behavior as the DA models requiring time steps 10 to 20 times the number of nodes in the longitudinal direction (Ye and McCorquodale, 1998). The new RANS models have also been tested with a very large aspect ratio without compromising the convergence behavior. This could be achieved because the continuity of the mean flow is maintained at each vertical at

each time step by the correction term instead of a pressure velocity coupling technique.

4.8 CONCLUSIONS

A coupled depth averaged-RANS model with or without the hydrostatic pressure assumption is presented for 2D plane flow. A correction term was added to the *x* momentum equation such that the mean of the horizontal velocity profiles becomes equal to the depth averaged velocity. The performance in modeling 2D plane flow in experimental channels showed good promise. The computational efficiency of the models is also satisfactory as the effort for the RANS solution is of the same order as the DA solutions.

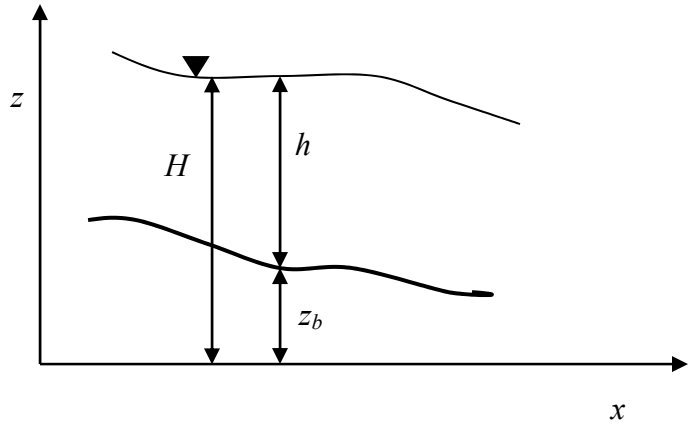

Figure 4.1 Definition sketch

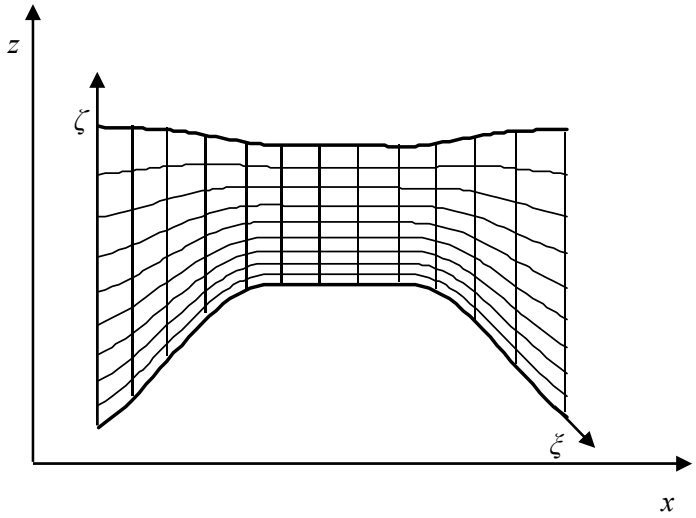

Figure 4.2 Schematic diagram of coordinate transformation in the *x* and *z* dimensions

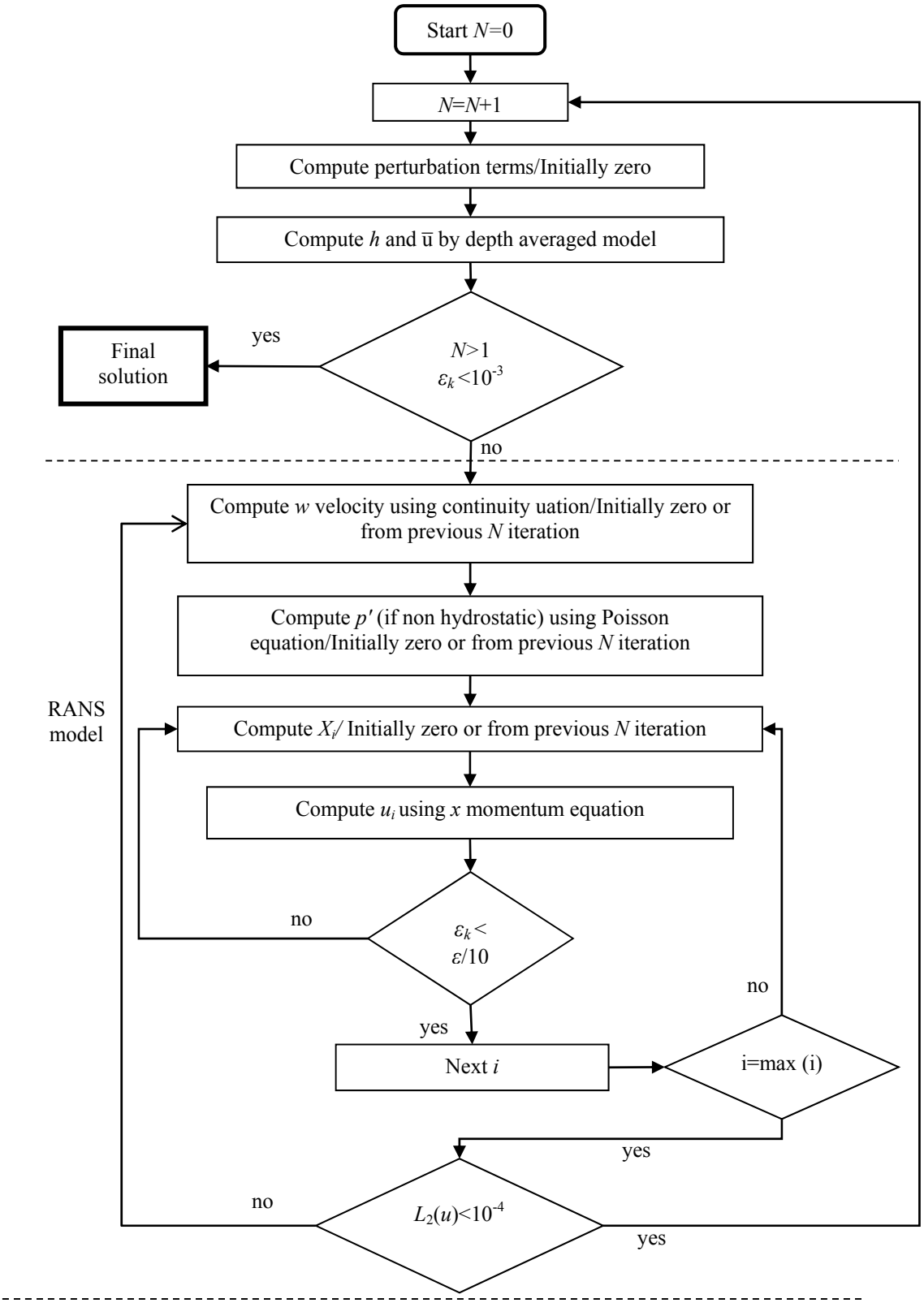

Figure 4.3 Flow diagram of the coupled DA-RANS model

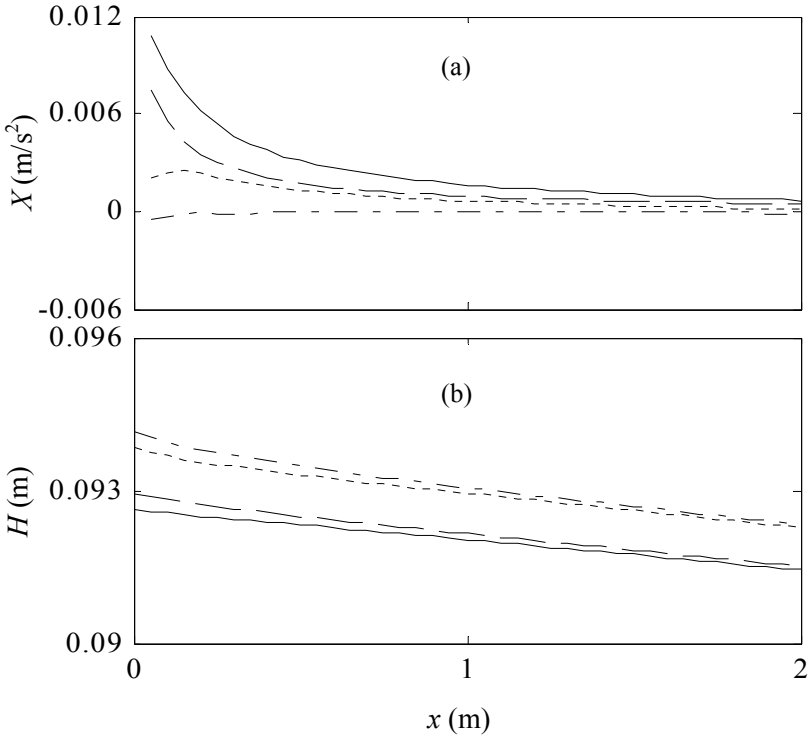

Figure 4.4 Correction term (X) and Water level (H) for the flow development over rough bed. (—) N=1, correction by (– –) dispersion, (--) bed stress, (– - –) dispersion and bed stress in N=2

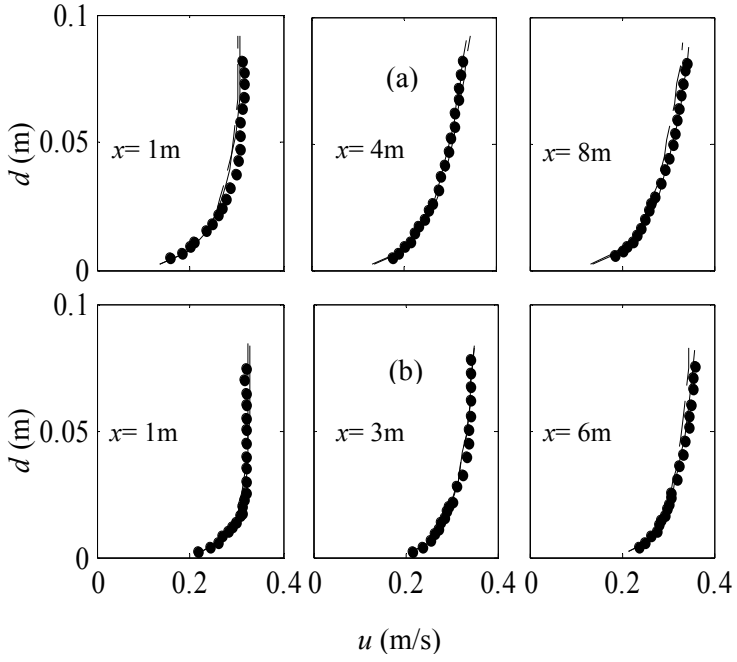

Figure 4.5 Velocity profiles in the flow development in a rectangular channel (●) Experimental, (—) N=1 and (— —) N=2 for numerical. (a) Rough bed (b) Smooth bed.

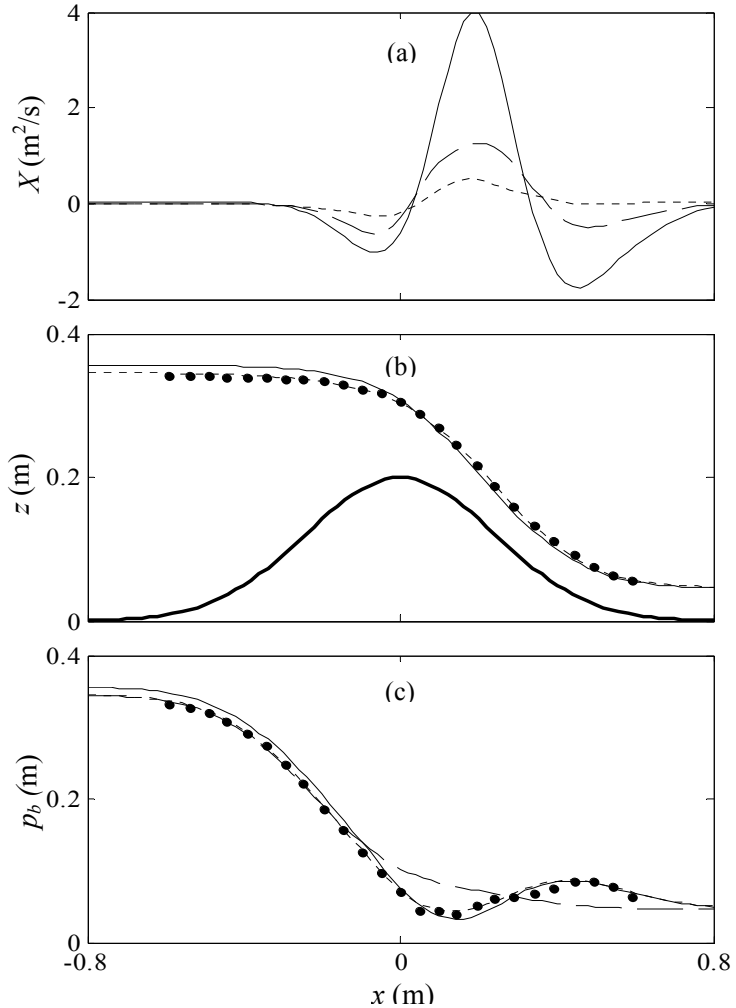

Figure 4.6 (a) Correction term and (b) water level for the flow over a symmetric hump. (●) Experimental, (—) N=1, (– –) N=2, (--) N=3 and (—) bed. (c) Bed pressure. Total pressure (●) Experimental, (—) N=1, (--) N=3. (– –), Hydrostatic pressure (numerical) for N=3.

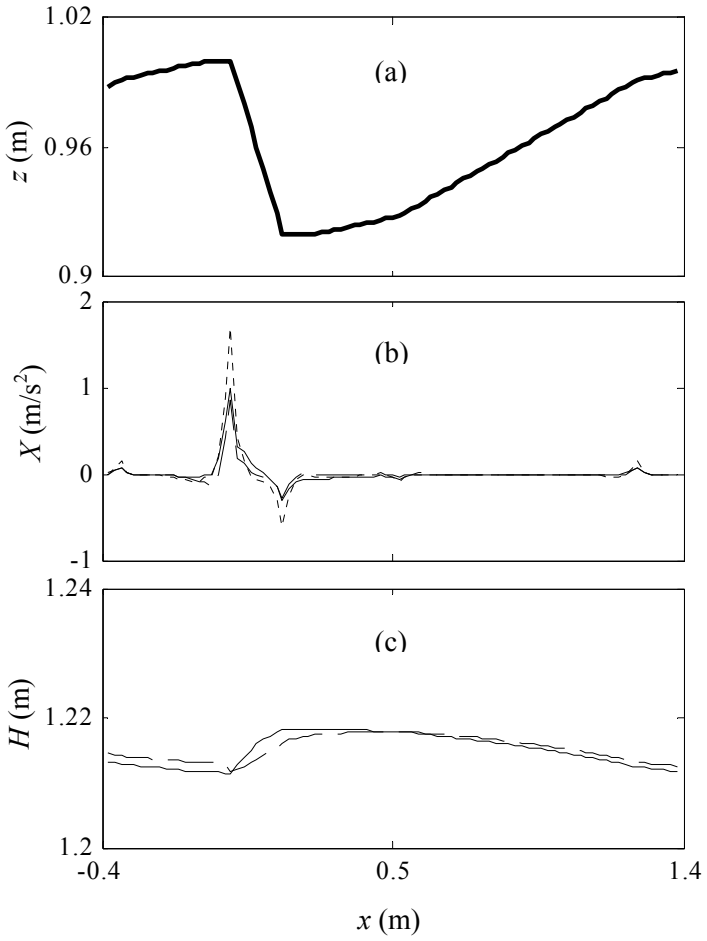

Figure 4.7 (a) Dune bed (b) Correction term and (c) water level for the flow over dunes (—) N=1, (– –) N=2, (--) $\frac{\partial H}{\partial \xi}$ discretized by Forward difference in the RANS model (N=1).

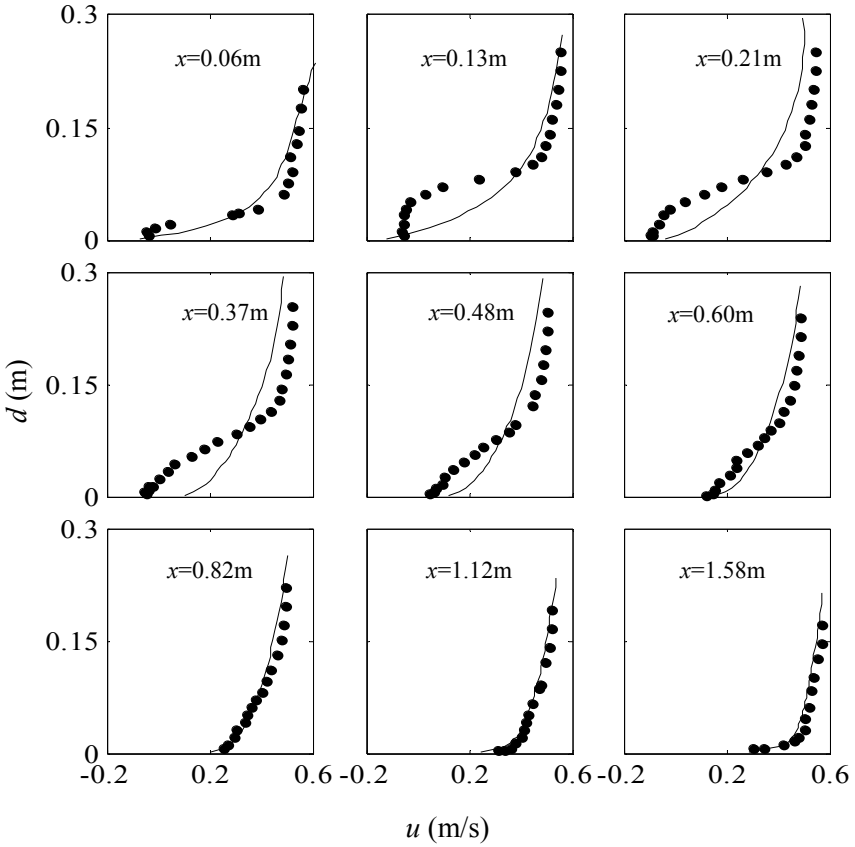

Figure 4.8 *u* velocity profiles for the flow over a dune (●) Experimental, (–) Numerical, *N*=2

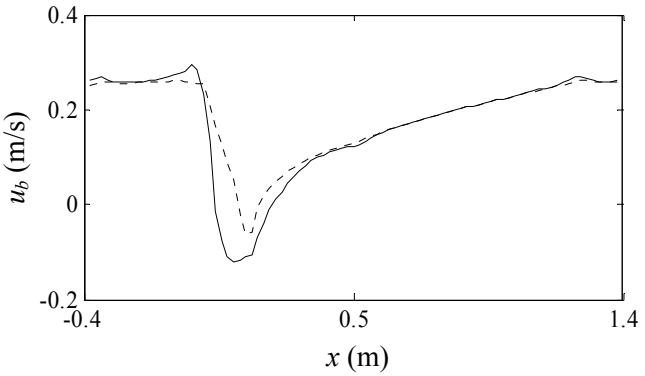

Figure 4.9 Comparison of simulated near bed velocity by (--) RANS-NH and (—) RANS-H models

Chapter 5

COUPLED DA-RANS MODEL FOR 3D FLOW

5.1 INTRODUCTION

In chapter 4, the coupled DA-RANS model was developed and tested for 2D plane flow. In this chapter the model is extended for 3D flow. The model is validated by simulating the flow in curved open channels and the flow around a hemisphere. Also the computational efficiency of the model is assessed by simulating the flow in a natural river.

5.2 MATHEMATICAL MODEL

The 3D RANS equations and the 2D depth averaged equations with their boundary conditions are presented in chapter 2.

Using the analogy from the 2D plane flow and one dimensional DA flow equations the correction terms (X and Y) for the RANS x and y momentum equations in 3D flow can be written as

$$X \equiv \frac{1}{h}\left[\begin{array}{c} \dfrac{\partial h\overline{u^p u^p}}{\partial x} + \dfrac{\partial h\overline{u^p v^p}}{\partial x} + \dfrac{1}{\rho}\left(\dfrac{\partial h\overline{p^p}}{\partial x} + p_b^p \dfrac{\partial z_b}{\partial x}\right) \\ -\dfrac{1}{\rho}\left(\dfrac{\partial h\overline{\sigma_x^p}}{\partial x} + \dfrac{\partial h\overline{\tau_{xy}^p}}{\partial y}\right) + \dfrac{1}{\rho}\tau_{bx}^p \end{array}\right] \qquad [5.1]$$

$$Y \equiv \frac{1}{h}\left[\begin{array}{c} \dfrac{\partial h\overline{u^p v^p}}{\partial x} + \dfrac{\partial h\overline{v^p v^p}}{\partial y} + \dfrac{1}{\rho}\left(\dfrac{\partial h\overline{p^p}}{\partial y} + p_b^p \dfrac{\partial z_b}{\partial y}\right) \\ -\dfrac{1}{\rho}\left(\dfrac{\partial h\overline{\sigma_y^p}}{\partial y} + \dfrac{\partial h\overline{\tau_{xy}^p}}{\partial x}\right) + \dfrac{1}{\rho}\tau_{bx}^p \end{array}\right] \qquad [5.2]$$

However as found from the results of 2D plane flow, the correction terms also represent the numerical differences between the RANS and the DA models.

The equivalent RANS *x* and *y* momentum equations can then be written as

$$\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x} + \frac{\partial uv}{\partial z} + \frac{\partial uw}{\partial z} + \frac{1}{\rho}\frac{\partial p}{\partial x} - \frac{1}{\rho}\left(\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z}\right) - X = 0 \qquad [5.3]$$

$$\frac{\partial v}{\partial t} + \frac{\partial uv}{\partial x} + \frac{\partial v^2}{\partial y} + \frac{\partial vw}{\partial z} + \frac{1}{\rho}\frac{\partial p}{\partial y} - \frac{1}{\rho}\left(\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} + \frac{\partial \tau_{zy}}{\partial z}\right) - Y = 0 \qquad [5.4]$$

Therefore the pressure Poisson equation is modified as

$$\frac{\partial^2 p^p}{\partial x^2}+\frac{\partial^2 p^p}{\partial y^2}+\frac{\partial^2 p^p}{\partial z^2}=2\rho\left(\frac{\partial u}{\partial x}\frac{\partial v}{\partial y}+\frac{\partial v}{\partial y}\frac{\partial w}{\partial z}+\frac{\partial w}{\partial z}\frac{\partial u}{\partial x}\right)$$
$$-2\rho\left(\frac{\partial u}{\partial y}\frac{\partial v}{\partial x}+\frac{\partial v}{\partial z}\frac{\partial w}{\partial y}+\frac{\partial w}{\partial x}\frac{\partial u}{\partial z}\right)$$
$$+\left(\frac{\partial^2\sigma_x}{\partial x^2}+\frac{\partial^2\sigma_y}{\partial y^2}+\frac{\partial^2\sigma_z}{\partial z^2}+2\frac{\partial^2\tau_{xy}}{\partial x\partial y}+2\frac{\partial^2\tau_{yz}}{\partial y\partial z}+2\frac{\partial^2\tau_{zx}}{\partial z\partial z}\right) \qquad [5.5]$$
$$+\rho\left(\frac{\partial X}{\partial x}+\frac{\partial Y}{\partial y}\right)-\rho g\left(\frac{\partial^2 H}{\partial x^2}-\frac{\partial^2 H}{\partial y^2}\right)$$

5.3 NUMERICAL SCHEME

For the depth averaged computation, equations [2.23], [2.24] and [2.25] are solved. All of the normal stresses are neglected for this study. Depth averaged modeling is accomplished by the McCormack scheme in the generalized boundary-fitted coordinate system. The DA continuity, x and y momentum equations can be expressed in the conservative form after transforming them from Cartesian to non-orthogonal boundary-fitted coordinate by the partial transformation approach, that is $(x,y)\rightarrow(\xi,\eta)$ (Figure 5.1) but leaving the velocity components in Cartesian coordinate,

$$\frac{1}{J}\frac{\partial h}{\partial t}+\frac{\partial}{\partial\xi}\left(\frac{h\overline{U_c}}{J}\right)+\frac{\partial}{\partial\eta}\left(\frac{h\overline{V_c}}{J}\right)=0 \qquad [5.6]$$

$$\frac{1}{J}\frac{\partial h\bar{u}}{\partial t}+\frac{\partial}{\partial\xi}\left(\frac{h\bar{u}\overline{U_c}}{J}+\frac{1}{2}\frac{gh^2}{J}\xi_x\right)+\frac{\partial}{\partial\eta}\left(\frac{h\bar{u}\overline{V_c}}{J}+\frac{1}{2}\frac{gh^2}{J}\eta_x\right)$$
$$+\frac{1}{J}\left(\xi_x\frac{\partial h\overline{u'u'}}{\partial\xi}+\eta_x\frac{\partial h\overline{u'u'}}{\partial\eta}\right)+\frac{1}{J}\left(\xi_y\frac{\partial h\overline{u'v'}}{\partial\xi}+\eta_y\frac{\partial h\overline{u'v'}}{\partial\eta}\right)$$
$$+\frac{1}{\rho J}\left(\xi_x\frac{\partial h\overline{p^p}}{\partial\xi}+\eta_x\frac{\partial h\overline{p^p}}{\partial\eta}\right)+\frac{p_b^p}{\rho J}\left(\xi_x\frac{\partial z_b}{\partial\xi}+\eta_x\frac{\partial z_b}{\partial\eta}\right) \qquad [5.7]$$
$$+gh\left(\xi_x\frac{\partial z_b}{\partial\xi}+\eta_x\frac{\partial z_b}{\partial\eta}\right)+\frac{1}{\rho J}\left(\tau_{bx}^d+\tau_{bx}'\right)=0$$

$$\frac{1}{J}\frac{\partial h\bar{v}}{\partial t}+\frac{\partial}{\partial\xi}\left(\frac{h\bar{v}\overline{U_c}}{J}+\frac{1}{2}\frac{gh^2}{J}\xi_y\right)+\frac{\partial}{\partial\eta}\left(\frac{h\bar{v}\overline{V_c}}{J}+\frac{1}{2}\frac{gh^2}{J}\eta_y\right)$$
$$+\frac{1}{J}\left(\xi_x\frac{\partial hu\overline{'v'}}{\partial\xi}+\eta_x\frac{\partial hu\overline{'v'}}{\partial\eta}\right)+\frac{1}{J}\left(\xi_y\frac{\partial hv\overline{'v'}}{\partial\xi}+\eta_y\frac{\partial hv\overline{'v'}}{\partial\eta}\right)$$
$$+\frac{1}{\rho J}\left(\xi_y\frac{\partial h\overline{p^{p'}}}{\partial\xi}+\eta_y\frac{\partial h\overline{p^{p'}}}{\partial\eta}\right)+\frac{p_b^p}{\rho J}\left(\xi_y\frac{\partial z_b}{\partial\xi}+\eta_y\frac{\partial z_b}{\partial\eta}\right) \qquad [5.8]$$
$$+\frac{gh}{J}\left(\xi_y\frac{\partial z_b}{\partial\xi}+\eta_y\frac{\partial z_b}{\partial\eta}\right)+\frac{1}{\rho J}\left(\tau_{by}^d+\tau_{by}'\right)=0$$

where $\overline{U_c}$, $\overline{V_c}$ = contravariant depth averaged velocity components ($\overline{U_c}=\bar{u}\xi_x+\bar{v}\xi_y$ and $\overline{U_c}=\bar{u}\eta_x+\bar{v}\eta_y$), J = Jacobian of the geometric transformation.

The DA equations are solved by the McCormack scheme in a finite volume formulation. As the second order accurate McCormack scheme may produce oscillations (Chaudhry, 1990), it may be necessary to eliminate the oscillations by adding explicit damping term. The artificial viscosity is modeled

by following the procedure of Martinelli and Jameson (1988) used by Lackey and Sotiropoulos (2005) for open channel flow modeling. The discretization by the McCormack scheme with the artificial viscosity is provided in appendix B**.**

For a subcritical upstream boundary, a known discharge is specified and the water level is extrapolated from interior nodes. For a subcritical downstream boundary, a known water depth is used and discharge is extrapolated from interior points. For a supercritical upstream boundary both water level and discharge are specified while no boundary condition is needed for a supercritical downstream boundary and both water level and discharge are extrapolated from interior points. For the side walls free-slip boundary conditions are used. They are implemented by setting the contravariant velocity component $\overline{U_{c\ \text{wall}}}$ equal to $\overline{U_c}$ at the first interior point while the condition of no flux perpendicular to the solid wall is applied by setting the contravariant velocity component $\overline{V_{c\ \text{wall}}}$ to zero. Water levels at the side walls are extrapolated from interior nodes.

The RANS continuity, x and y momentum equations for incompressible fluid including two correction terms (X and Y) can be expressed in the non-conservative form after transforming them from Cartesian to the non-orthogonal boundary-fitted coordinate system by the partial transformation approach, that is $(x,y,z)\rightarrow(\xi,\eta,\zeta)$ (Figure 5.2) but leaving the velocity components in Cartesian coordinate,

$$\xi_x \frac{\partial u}{\partial \xi} + \eta_x \frac{\partial u}{\partial \eta} + \zeta_x \frac{\partial u}{\partial \zeta} + \xi_y \frac{\partial v}{\partial \xi} + \eta_y \frac{\partial v}{\partial \eta} + \zeta_y \frac{\partial v}{\partial \zeta} + \xi_z \frac{\partial w}{\partial \xi} + \eta_z \frac{\partial w}{\partial \eta} + \zeta_z \frac{\partial w}{\partial \zeta} = 0 \qquad [5.9]$$

$$\frac{\partial u}{\partial t} + U_c \frac{\partial u}{\partial \xi} + V_c \frac{\partial u}{\partial \eta} + W_c \frac{\partial u}{\partial \zeta} = -\frac{1}{\rho}\left(\xi_x \frac{\partial p^p}{\partial \xi} + \eta_x \frac{\partial p^p}{\partial \eta} + \zeta_x \frac{\partial p^p}{\partial \zeta}\right) + g\left(\xi_x \frac{\partial H}{\partial \xi} + \eta_x \frac{\partial H}{\partial \eta}\right) + \frac{1}{\rho}\left(\xi_z \frac{\partial \tau_{xz}}{\partial \xi} + \eta_z \frac{\partial \tau_{xz}}{\partial \eta} + \zeta_z \frac{\partial \tau_{xz}}{\partial \zeta}\right) + X; \qquad [5.10]$$

$$\frac{\partial v}{\partial t} + U_c \frac{\partial v}{\partial \xi} + V_c \frac{\partial v}{\partial \eta} + W_c \frac{\partial v}{\partial \zeta} = -\frac{1}{\rho}\left(\xi_y \frac{\partial p^p}{\partial \xi} + \eta_y \frac{\partial p^p}{\partial \eta} + \zeta_y \frac{\partial p^p}{\partial \zeta}\right) + g\left(\xi_y \frac{\partial H}{\partial \xi} + \eta_y \frac{\partial H}{\partial \eta}\right) + \frac{1}{\rho}\left(\xi_z \frac{\partial \tau_{yz}}{\partial \xi} + \eta_z \frac{\partial \tau_{yz}}{\partial \eta} + \zeta_z \frac{\partial \tau_{yz}}{\partial \zeta}\right) + Y; \qquad [5.11]$$

The pressure Poisson equation

$$
\begin{aligned}
&\left(\xi_x\frac{\partial}{\partial\xi}+\eta_x\frac{\partial}{\partial\eta}+\zeta_x\frac{\partial}{\partial\zeta}\right)\left(\xi_x\frac{\partial p'}{\partial\xi}+\eta_x\frac{\partial p'}{\partial\eta}+\zeta_x\frac{\partial p'}{\partial\zeta}\right)\\
&+\left(\xi_y\frac{\partial}{\partial\xi}+\eta_y\frac{\partial}{\partial\eta}+\zeta_y\frac{\partial}{\partial\zeta}\right)\left(\xi_y\frac{\partial p'}{\partial\xi}+\eta_y\frac{\partial p'}{\partial\eta}+\zeta_y\frac{\partial p'}{\partial\zeta}\right)\\
&+\left(\xi_z\frac{\partial}{\partial\xi}+\eta_z\frac{\partial}{\partial\eta}+\zeta_z\frac{\partial}{\partial\zeta}\right)\left(\xi_z\frac{\partial p'}{\partial\xi}+\eta_z\frac{\partial p'}{\partial\eta}+\zeta_z\frac{\partial p'}{\partial\zeta}\right)\\
&=2\rho\left(\xi_x\frac{\partial u}{\partial\xi}+\eta_x\frac{\partial u}{\partial\eta}+\zeta_x\frac{\partial u}{\partial\zeta}\right)\left(\xi_y\frac{\partial v}{\partial\xi}+\eta_y\frac{\partial v}{\partial\eta}+\zeta_y\frac{\partial v}{\partial\zeta}\right)\\
&+2\rho\left(\xi_y\frac{\partial v}{\partial\xi}+\eta_y\frac{\partial v}{\partial\eta}+\zeta_y\frac{\partial v}{\partial\zeta}\right)\left(\xi_z\frac{\partial w}{\partial\xi}+\eta_z\frac{\partial w}{\partial\eta}+\zeta_z\frac{\partial w}{\partial\zeta}\right)\\
&+2\rho\left(\xi_z\frac{\partial w}{\partial\xi}+\eta_z\frac{\partial w}{\partial\eta}+\zeta_z\frac{\partial w}{\partial\zeta}\right)\left(\xi_x\frac{\partial u}{\partial\xi}+\eta_x\frac{\partial u}{\partial\eta}+\zeta_x\frac{\partial u}{\partial\zeta}\right)\\
&-2\rho\left(\xi_y\frac{\partial u}{\partial\xi}+\eta_y\frac{\partial u}{\partial\eta}+\zeta_y\frac{\partial u}{\partial\zeta}\right)\left(\xi_x\frac{\partial v}{\partial\xi}+\eta_x\frac{\partial v}{\partial\eta}+\zeta_x\frac{\partial v}{\partial\zeta}\right)\\
&-2\rho\left(\xi_z\frac{\partial v}{\partial\xi}+\eta_z\frac{\partial v}{\partial\eta}+\zeta_z\frac{\partial v}{\partial\zeta}\right)\left(\xi_y\frac{\partial w}{\partial\xi}+\eta_y\frac{\partial w}{\partial\eta}+\zeta_y\frac{\partial w}{\partial\zeta}\right)\\
&-2\rho\left(\xi_x\frac{\partial w}{\partial\xi}+\eta_x\frac{\partial w}{\partial\eta}+\zeta_x\frac{\partial w}{\partial\zeta}\right)\left(\xi_x\frac{\partial u}{\partial\xi}+\eta_x\frac{\partial u}{\partial\eta}+\zeta_x\frac{\partial u}{\partial\zeta}\right)\\
&+2\left(\xi_x\frac{\partial}{\partial\xi}+\eta_x\frac{\partial}{\partial\zeta}+\zeta_x\frac{\partial}{\partial\zeta}\right)\left(\xi_z\frac{\partial\tau_{xz}}{\partial\xi}+\eta_z\frac{\partial\tau_{xz}}{\partial\xi}+\zeta_z\frac{\partial\tau_{xz}}{\partial\zeta}\right)\\
&+2\left(\xi_y\frac{\partial}{\partial\xi}+\eta_y\frac{\partial}{\partial\zeta}+\zeta_y\frac{\partial}{\partial\zeta}\right)\left(\xi_z\frac{\partial\tau_{xy}}{\partial\xi}+\eta_z\frac{\partial\tau_{xy}}{\partial\xi}+\zeta_z\frac{\partial\tau_{xy}}{\partial\zeta}\right)\\
&+\rho\left(\xi_x\frac{\partial X}{\partial\xi}+\eta_x\frac{\partial X}{\partial\eta}+\zeta_x\frac{\partial X}{\partial\zeta}\right)+\rho\left(\xi_y\frac{\partial Y}{\partial\xi}+\eta_y\frac{\partial Y}{\partial\eta}+\zeta_y\frac{\partial Y}{\partial\zeta}\right)\\
&-\rho g\left(\xi_x\frac{\partial H}{\partial\xi}+\eta_x\frac{\partial H}{\partial\eta}+\zeta_x\frac{\partial H}{\partial\zeta}\right)-\rho g\left(\xi_y\frac{\partial H}{\partial\xi}+\eta_y\frac{\partial H}{\partial\eta}+\zeta_y\frac{\partial H}{\partial\zeta}\right)
\end{aligned}
\qquad [5.12]
$$

The RANS *z* momentum equation

$$U_c \frac{\partial w}{\partial \xi} + V_c \frac{\partial w}{\partial \eta} + W_c \frac{\partial w}{\partial \zeta} = -\frac{1}{\rho}\left(\xi_z \frac{\partial p^p}{\partial \xi} + \eta_z \frac{\partial p^p}{\partial \eta} + \zeta_z \frac{\partial p^p}{\partial \zeta} \right) + g\left(\xi_z \frac{\partial H}{\partial \xi} + \eta_z \frac{\partial H}{\partial \eta} + \zeta_z \frac{\partial H}{\partial \zeta} \right) + \frac{1}{\rho}\left(\xi_x \frac{\partial \tau_{xz}}{\partial \xi} + \eta_x \frac{\partial \tau_{xz}}{\partial \eta} + \zeta_x \frac{\partial \tau_{xz}}{\partial \zeta} \right) \quad [5.13]$$

where *u*, *v*, and *w* = Cartesian velocity components in the *x*, *y* and *z* directions respectively, U_c, V_c and W_c= contravariant velocity components ($U_c = u\xi_x + u\xi_y + w\xi_z$ $V_c = u\eta_x + v\eta_y + w\eta_z$, $W_c = u\zeta_x + v\zeta_y + w\zeta_z$), $\xi_x, \xi_y, \xi_z, \eta_x, \eta_y, \eta_z, \zeta_x, \zeta_y$ and ζ_z = components of transformation matrix. The ζ direction coincides with the vertical direction while ξ and η follows the bed and water surface profile and so ξ_z, η_z=0. Also $\frac{\partial H}{\partial \zeta}$, $\frac{\partial X}{\partial \zeta}$ and $\frac{\partial Y}{\partial \zeta} = 0$.

The convective terms are discretized by a first order upwind method. The pressure gradient terms are approximated by a centered difference method. The gradient terms in the ξ and η directions are discretized explicitly with an implicit coupling in the ζ direction which allows solving each vertical independently using the values of the neighboring verticals from the previous time step and avoids the solution of a large matrix. The shear stresses are computed by a zero equation turbulence model.

For the RANS *x* and *y* momentum equations vertical profiles of *u* and *v* are specified as the upstream boundary condition. For the side walls free-slip boundary conditions are used. They are implemented by setting the contravariant

velocity component $U_{c,wall}$ equal to U_c at the first interior point while the condition of no flux perpendicular to the solid wall is applied by setting the contravariant velocity component $V_{c,wall}$ to zero. The law of the wall is used as the bottom boundary condition and a zero shear stress is specified at the water surface. Logarithmic velocity profiles based on the depth averaged velocities are specified as the initial condition at each vertical.

5.4 MODELING THE CORRECTION TERM

Following the procedure described for 2D plane flow, the iterative formulae to model the correction terms can be written as

$$X_{k+1} = \frac{\bar{u} - \bar{u}_k^{n+1}}{\Delta t} + X_k \qquad [5.14]$$

$$Y_{k+1} = \frac{\bar{v} - \bar{v}_k^{n+1}}{\Delta t} + Y_k \qquad [5.15]$$

5.5 SOLUTION PROCEDURE

For the depth averaged model, a time marching procedure is used to obtain the steady state solution. Local time stepping is used by maintaining a constant Courant number for every node. Then the time step is computed as

$$\Delta t_{ij} = \frac{Cr}{\max\left(\left|\overline{U_c}\right| + c\sqrt{\xi_x^2 + \xi_y^2}, \left|\overline{V_c}\right| + c\sqrt{\eta_x^2 + \eta_y^2},\right)}$$

where, *i,j* refers to a particular node. Since the McCormack scheme is explicit, time step should satisfy the Courant-Friedrichs-Lewy (CFL) criteria *i.e* $Cr \leq 1$

The steady state solution is achieved when the L_2 norms of the residuals of the depth averaged continuity and momentum equations are reduced to 10^{-4}.

In the RANS model, a time marching procedure is also used to obtain the steady state solution. For each vertical a local time step is computed from a fixed Courant number (*Cr*),

$$\Delta t_i = \frac{Cr}{\max(U_{i,j}, V_{i,j})}$$

where *i,j* refers to a particular vertical.

The solution procedure is the same as the 2D plane flow except the RANS *y* momentum equation is also solved for *v* velocity together with the *x* momentum

equation and the correction term Y is modeled so that the mean of the modeled v velocity profile is equal to $\bar{v}$.

5.6 MODEL VALIDATION

5.6.1 Modeling flow in curved open channels

Among different types of flow in open channels the flow in meandering channels have drawn special attention from hydraulic engineers and researchers as it is practically harder to find a natural river with a straight reach longer than 10 channel width (Leopold and Wolman 1960) and the meandering channels exhibit special flow characteristics. The flow characteristics are the superelevation, flow acceleration in the inner bank and deceleration in the outer bank at the entrance of the bend, secondary flow and the redistribution of the longitudinal velocity. The first characteristic is mainly due to the centrifugal force acting on the fluid and the second one is a result of the first one. The secondary flow is due to the superelevation and non-uniform distribution of longitudinal velocity which in turn causes the flow redistribution.

The two dimensional (2D) depth averaged (DA) models based on the classical St Venant equations can only predict the first two characteristics. But they are unable to predict the other characteristics due to the uniform velocity assumption in its derivation. To model the secondary current and the flow redistribution many models ranging from the analytical, improved DA (quasi-3D), to full 3D models have been developed by different researchers.

Boussinesq developed a mathematical formulation of the laminar flow in a mildly curved channel to describe the secondary motion in a bend (Demuren and Rodi, 1986). Different varieties of perturbation techniques have been developed to analyze the flow in curved open channels (Rozovskii 1957, Yen 1965, Engelund, 1974). Johannesson and Parker (1987) developed an analytical model for calculating the lateral distribution of the depth averaged primary velocity by using the moment method and a linear perturbation technique. DeVriend (1976, 1981) and Kalkwijk and DeVriend (1980) modified the depth averaged equations by including local functions for the velocity deviations and developed mathematical models in the curvilinear coordinate system. They showed that it is important to include the convective influence of the secondary flow on the main flow. Jin and Steffler (1992) developed a finite element model which consists of the DA continuity equation, the momentum equations and two moment of the momentum equations. The vertically averaged and the moment (VAM) equations developed by Steffler and Jin (1992) were used by Ghamry and Steffler (2005) to simulate the flow in curved open channels and with different assumed distribution shapes for velocity and pressure (Ghamry and Steffler 2002).

Many computational fluid dynamic (CFD) models based on the RANS equations have been developed to simulate the flow in curved open channels and are generally capable of predicting the flow characteristics. Demuren and Rodi (1986) simulated the flow in a rectangular channel while Morvan et al. (2002) used a compound channel assuming the water surface to be a rigid lid. Ye and

McCorquodale (1998) used the DA continuity equation to update the water surface. Wu et al. (2000) used the 2D Poisson equation derived from the 2D DA equations while Meselhe and Sotipoulos (2000) used the free surface kinematic condition for modeling the water level variations. Nguyen et al. (2007) developed a three dimensional model in a finite element formulation with a segregated solver. A Lagrangian approach was used for modeling the free surface.

The depth averaged models based on the St Venant equations are incapable to produce the secondary flow and redistribution of longitudinal velocity. The dispersion terms ($\overline{u'u'}$, $\overline{u'v'}$, $\overline{v'v'}$) due to the non-uniform velocity in the complete depth averaged equations are responsible for the flow redistribution. In this study, the coupled DA-RANS model is validated to simulate the flow in curved open channels. Ghamry and Steffler (2005) showed that the vertical velocity and non-hydrostatic pressure may only be important very close to the side walls, therefore the vertical velocity and non-hydrostatic pressure are neglected here. The eddy viscosity is computed by the specified vertical profile of eddy viscosity model (Equation [2.33]). The experimental results of Rozovskii (1957) and Steffler (1984) are used for model verification.

Rozovskii's experiment

The experimental data of Rozovskii (1957) have widely been used to validate the capability of different numerical models to simulate the flow field and

predict the secondary flows in a bend that exhibits strongly three dimensional characteristics. Rozovskii performed his experiments in a 180° curved rectangular flume with a very strong degree of curvature ($Rc/2b$=1, Rc= radius of the centre of the bend, b= half channel width) The secondary velocities produced in his experiments were very strong due to the sharp curvature of the channel. The results of run 1 are used to test the numerical predictions of the present model.

Rozovskii's channel consisted of a 6m long straight approach and a 3m long straight exit with a 180° bend (Figure 5.3). The width of the channel was 0.8 m, and the radius of the channel centerline was 0.8 m for the circular reach. The channel bed was horizontal and smooth with a Chezy coefficient of 60 $m^{1/2}/s$. The inflow discharge was 0.0123 m^3/s with the flow depth at the entrance equal to 0.06 m. This gives a mean velocity at the entrance equal to 0.256 m/s. For the DA model inputs, a steady inflow discharge of 0.0123 m^3/s and a constant depth equal to 0.057m are used at the upstream and downstream boundaries respectively. Based on the measured Chezy coefficient the corresponding roughness height is estimated to be 0.0003 m.

To test the mesh independency of the numerical results, 3 meshes with 140 (longitudinal) X 17 (transverse), 140X25 and 180X33 are tested for the DA model. For the streamwise direction a denser spacing is used in the bend region with a coarser spacing in the straight reaches. For the transverse direction an equal spacing is used. The discharge per unit width at different sections of the bend is compared for the meshes. The results at 90^0 of the bend are plotted in Figure 5.4

which shows that the results of the second and third girds in the region Y_R/b= -0.75 to 0.75 are close. These results show similarity with those of Lackey and Sotiropoulos (2005) who verified their depth averaged model with a centered difference scheme and artificial viscosity by simulating this experimental data. They tested 3 meshes with 125 (longitudinal) X 21 (transverse), 253X41 and 505X81 nodes and compared the variation of the longitudinal velocity in the transverse direction at different sections. It showed that results due to the mesh refinement vary only near the side walls especially in the inner bank. This difference increases towards the downstream sections.

For the RANS model, mesh 2 and 3 are tested with 15 and 20 nodes in the vertical direction and the longitudinal velocity at different locations are compared. A typical comparison is shown in Figure 5.5. Generally 15 nodes in the vertical are sufficient for the RANS model. Therefore a mesh of 140X25X15 is used for simulating this experimental data.

The coupled model converged after 5 iterations (N=5). Figure 5.6 shows the depth averaged velocity at different sections for N=1 and N=5. The solution of the DA model at N=1 is actually the solution of the classical St Venant equations which ignores the effect of the dispersion terms. As compared in the figure, the feedback of the dispersion terms into the DA model can simulate the flow shifting from the inner towards the outer bank. So the results at N=5 generally give a better agreement with the experiment. As the flow moves towards the end of the bend the amount of flow shifting increases.

The computed longitudinal velocity (u_L) profiles at N=5 are shown in Figure 5.7 and compared with the experimental results. Generally all the profiles show good agreement with the experimental results. At section 6 which is located at 100 degree of the bend, the profiles at the center (Y_R/b=0), outer side (Y_R/b=0.5, 0.75) and inner side (Y_R/b=-0.5, -0.75) of the bend matches very well with the experimental results. However the model overpredicts the near bed velocity and underpredicts the near surface velocity near the inner bank at Y_R/b=-0.75. At section 8 (144 degree) the model predicts the center (Y_R/b=0) and the outer (Y_R/b=0.5, 0.75) profiles very well. In the inner side the model cannot predict the velocity dip at the surface. Just downstream of the exit (Section 10), the model results show similar agreement with the experiment as section 8.

The simulated transverse velocity (u_T) profiles are compared with the experimental results and shown in Figure 5.8. At section 6 the simulated profiles agree well with the experimental results in the center (Y_R/b=0.0) and outer side (Y_R/b=0.5, 0.75). At the inner side (Y_R/b=-0.5) the model underpredicts the near bed velocity and at Y_R/b=-0.75 the model underpredicts the strength of the secondary flow. At section 8 the model can reproduce the experimental results quite well expect it underpredicts the near bed velocity at Y_R/b=-0.5. At section 10, the model consistently underpredicts the near bed velocity. However the results near the surface are reasonably good.

Steffler's experiment

Steffler (1984) conducted experiments in a 270° bend with a moderate curvature ($Rc/2b$=3.4) at the Hydraulics laboratory of the University of Alberta, Edmonton. The flume was 1.07 m in width, 0.21 m in depth with a 13.4 m long straight inlet and 2.4 m long outlet sections (Figure 5.9). The radius of the centerline was 3.66 m. The bed slope was constant at a value of 0.00083 with a dimensionless Chezy coefficient equal to 16 giving a roughness height of 0.0013 m.

Four meshes of 180X15, 180X21, 180X29 and 240X29 are tested for the DA model and then mesh 2 and 4 are tested with 15 and 20 nodes in the vertical direction. Finally the mesh of 180X21X15 is selected for the DA-RANS model. This gives a total of 3780 and 56700 computational nodes for the DA and the RANS models respectively. For the DA model at the upstream boundary a steady inflow discharge of 0.0224 m^3/s is used. For the downstream boundary a constant depth equal to 0.061 m is used. Based on the measured Chezy coefficient the corresponding roughness height is estimated to be 0.0013 m.

The coupled model converged after 6 iterations (N=6). Figure 5.10 shows the depth averaged velocity at different sections for N=1 and N=6. At the entrance (Section 1) the St Venant equations (N=1) can simulate the flow acceleration near the inner bank correctly and the dispersion terms have no effect at this location (N=6). As the flow moves along the bend, the results at N=1 tend to deviate from the experimental results and the deviation increases with the distance from the

entrance. When the effects of the dispersion terms are introduced into the DA model, model results improve significantly. At section 2 the numerical and experimental DA velocity are in very good agreement at N=6. At section 3 the DA velocity is overpredicted near the inner bank. At section 4 the DA velocity is slightly overpredicted at the inner bank and slightly underpredicted at the outer bank. However the feedback of the dispersion terms can improve the DA results significantly.

The computed longitudinal velocity (u_L) profiles at N=6 are shown in Figure 5.11. All the profiles show good agreement with the experimental results. The model generally overpredicts the velocity and cannot reproduce the velocity dip near the surface in the inner bank. The model can predict the velocity profiles reasonably well near the center and the outer bank.

The simulated transverse velocity (u_T) profiles are compared with the experimental results and shown in Figure 5.12. At the entrance (Section 1), flow shifts from the outer toward the inner bank due to the acceleration at the inner bank and deceleration at the outer bank and the model can predict the unidirectional radially inward motion. At sections 2, 3 and 4 the predicted velocity profiles show the secondary motion from the inner towards the outer bank near the water surface with a relatively weaker counterflow near the bed. The model results also show good agreement with the experiment, however it underpredicts the near bed secondary motion at the centre (Y_R/b=0) and middle of the outer bank (Y_R/b=0.4) and overpredicts the near surface velocity at (Y_R/b=0.8).

Comparison of computational effort

To get an estimate of the computational efficiency of the DA-RANS model in simulating the flow in curved open channels, a comparison between the DA model and the RANS model in terms of CPU time required to obtain the steady state solutions is made. All the computer programs of the DA and RANS models are written in Matlab® with a 2.8 GHz processor.

Figure 5.13 shows the CPU time and number of time steps required for simulating the flow in Rozovoskii's bend which converged in 5 iterations (N=5). At N=1, the DA and the RANS models require 1000 *s* (7200 time steps) and 265 *s* (213 time steps) respectively. For the next four iterations, the DA model requires 950 *s* while the RANS model requires 800 *s* approximately. Figure 5.14 shows the comparison of computational efforts for simulating the flow in Steffler's bend which converged in 6 iterations. At N=1, the DA and the RANS models require 920 *s* (6500 time steps) and 190 *s* (135 time steps) respectively. For the next five iterations, the DA and the RANS models require 850 and 560 *s* respectively.

5.6.2 Modeling flow around a hemisphere

Shamloo (1996) conducted a series of experiments to investigate the flow around hemispheres with different flow and geometry at the Hydraulics Laboratory of the University of Alberta. The experimental flume was 1.22 m in

width, 0.2 m in height. The bed was made of smooth aluminum and side walls were made of Plexiglas. Experiment AS1 was conducted under a moderate relative depth to the height of the hemisphere of 1.85 and is used here for model verification. In this experiment, the diameter of the hemisphere was 0.130 m with an inflow discharge of 0.054 m^3/s and average water depth of 0.12m. The bed slope was 0.00147.

In the current RANS model, a non-orthogonal boundary-fitted coordinate system is used. It is generally recommended to limit the departure from orthogonality to less than 45 degree (Thompson et al. 1985). Therefore the geometry of the hemisphere is modified by extending the slope at 45^0 to the bed (Figure 5.15). The commercial CFD model CFX 12.0 which uses a fully coupled solver, is also used to simulate the flow around the modified geometry in order to compare with the current model results.

For the DA-RANS model, a domain of 4mX0.8m with the modified hemisphere at the center is discretized with a mesh of 165X65X15. The horizontal node spacing near the hemisphere is 0.01m. For the DA model, at the upstream boundary an inflow discharge of 0.054 m^3/s is provided. At the downstream boundary a water depth of 0.12 m is provided. For the RANS model, logarithmic velocity profiles corresponding to the DA velocity are provided at the upstream boundary. The feedback of the perturbation terms was not possible as it makes the DA model unstable.

For the model in CFX, only the half of the domain symmetric about the centerline along the *x* axis is modeled. The domain of 4mX0.4 m is discretized with 225,086 nodes and 1,233,315 elements. The horizontal node spacing near the hemisphere is 0.01m while in the vertical it is 0.005 m. A high resolution discretization technique and the standard $k-\varepsilon$ turbulence model are used with a smooth bed. The water surface is approximated with a rigid lid. At the upstream boundary logarithmic velocity profiles are used as the boundary conditions.

The simulated longitudinal velocity profiles of both models are compared in Figure 5.16. Experimental results at *x*/*D*=-0.77, 0, 0.25 and 0.4 are also compared. At *x*/*D*=-0.77 which is located just at the toe of the modified geometry, the profiles from both models are in good agreement. However they do not match with the experimental results near the bed as this location is upstream of the toe in the original geometry. At x/*D*=0.0, CFX results show better agreement than the DA-RANS model, although both models overpredict the results. At *x*/*D*=0.25 both model can predict the shape of the profile and however they overpredict it. At x/*D*=0.4, both models underpredict the near bed velocity and overpredict the upper profile. In general both models produce similar results.

Comparison of computational efforts

The DA model requires a CPU time of 30 minutes (3200 time steps) and the RANS model requires 20 minutes (350 time steps) to obtain the steady state

solution. On the other hand, the CFX model requires 120 minutes (120 time steps).

5.6.3 Flow modeling in the North Saskatchewan river

Dow (2003) studied the mixing characteristics in a 2 km stretch of the North Saskatchewan River near the Goldbar waste water treatment plant (GBWWTP) in Edmonton (Figure 5.17). She used the River2D model to simulate the steady two dimensional distribution of depth averaged velocity and water depth. The upstream boundary was located at about 0.5 km upstream of the GBWWTP outfall and the downstream boundary was located at about 1.5 km downstream of the outfall. The inflow discharge was 148.8 m^3/s and the flow through the outfall was 3.34 m^3/s. The domain was discretized by 63,424 nodes and 126,076 triangular elements. The results from the River2D model of Dow (2003) are used to run both the RANS-H and RANS-NH models. The purpose of this simulation is to get an estimate of the computational efficiency of the RANS model.

For the RANS models a reduced domain is used. Because it is difficult to discretize the region near the bridge piers by the boundary-fitted coordinate system, the reduced domain spans from the upstream boundary upto the upstream of the bridge. The length of this reduced domain is about 1.3 km. For locating the side wall boundaries cumulative discharge is used and the boundaries are located along cumulative discharge of 2 and 147 m^3/s. Since the side wall boundaries are

located along constant cumulative discharge, the depth averaged flow perpendicular to the walls is zero. The domain is first discretized with 783 and 65 nodes in the longitudinal and transverse directions respectively. The node spacing is 2 m approximately. The bed elevations, bed roughness, water level and the discharge intensity per unit width in the *x* and *y* directions are extracted at the node locations using the post processor of River2D. Then the domain was discretized by 15 nodes in the vertical direction. Thus the total number of node for the RANS model is 763,425. Because the sidewall boundaries are located inside the river water, the no flow boundary condition perpendicular to the wall is not applied for the RANS model. Rather the contravariant velocity perpendicular to the wall is set equal to the velocity at the first interior node and the computed velocity profiles are adjusted to get the mean of the computed velocity profile equal to the DA velocity. The RANS-H and RANS-NH models require approximately 1 and 1.5 hour of CPU time respectively to obtain the steady state solution.

5.7 CONCLUSION

The coupled DA-RANS model is verified by modeling the flow in curved open channels and flow around a hemisphere. In the curved open channel flow, the model showed very good performance when compared with the experimental results. For modeling the flow around the hemisphere, the model results are also

compared with the results from CFX. The computational efficiency of the RANS model is also verified by simulating the flow in the North Saskatchewan river which showed satisfactory results.

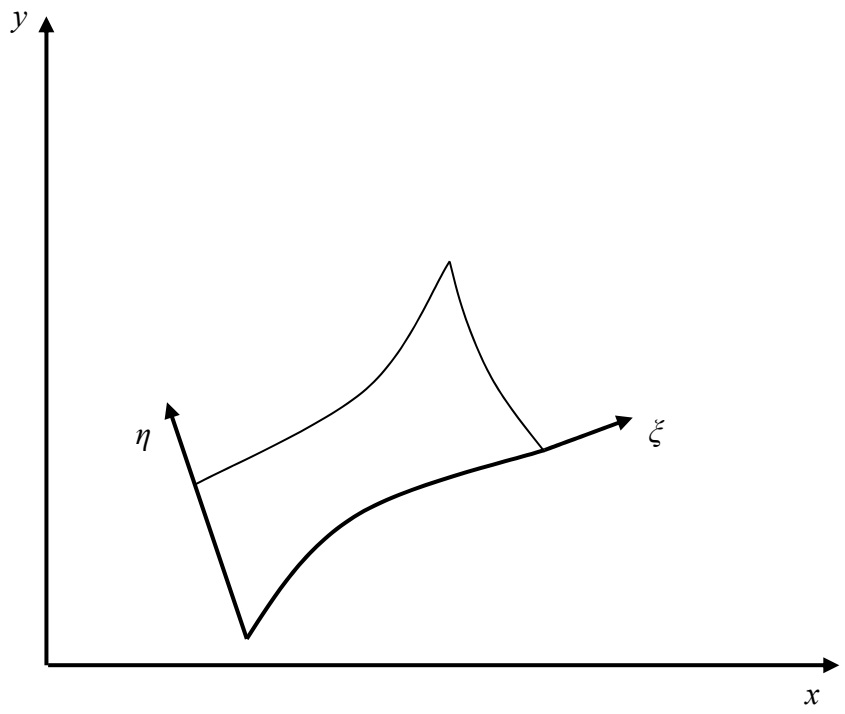

Figure 5.1 Schematic diagram of coordinate transformation in the x and y dimensions

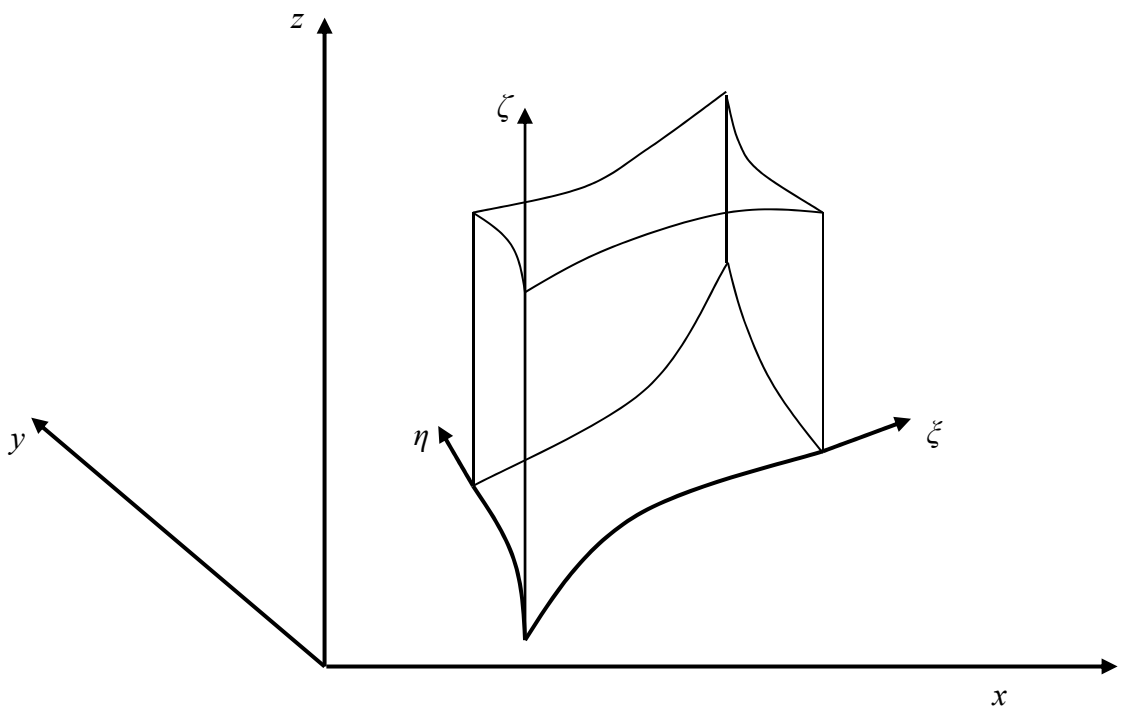

Figure 5.2 Schematic diagram of coordinate transformation in the *x*, *y* and *z* dimensions

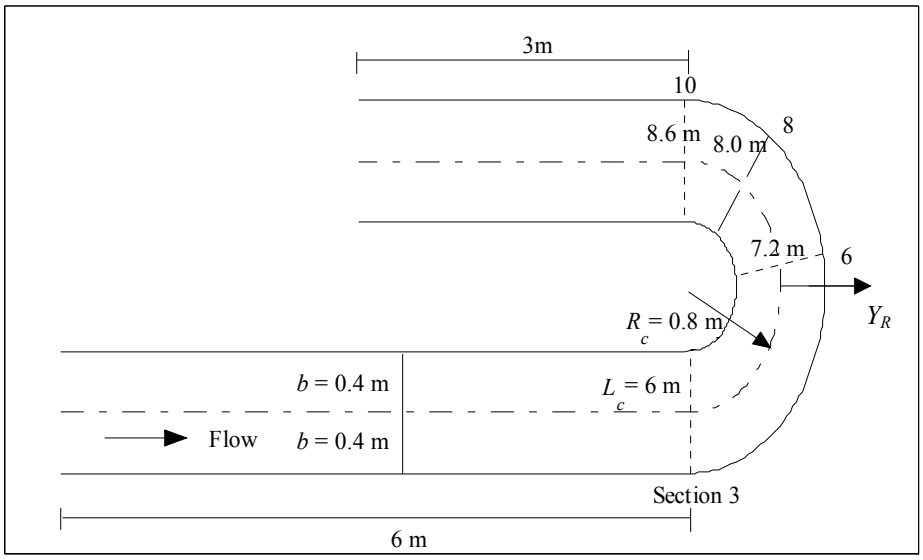

Figure 5.3 Layout of Rozovskii's experiment

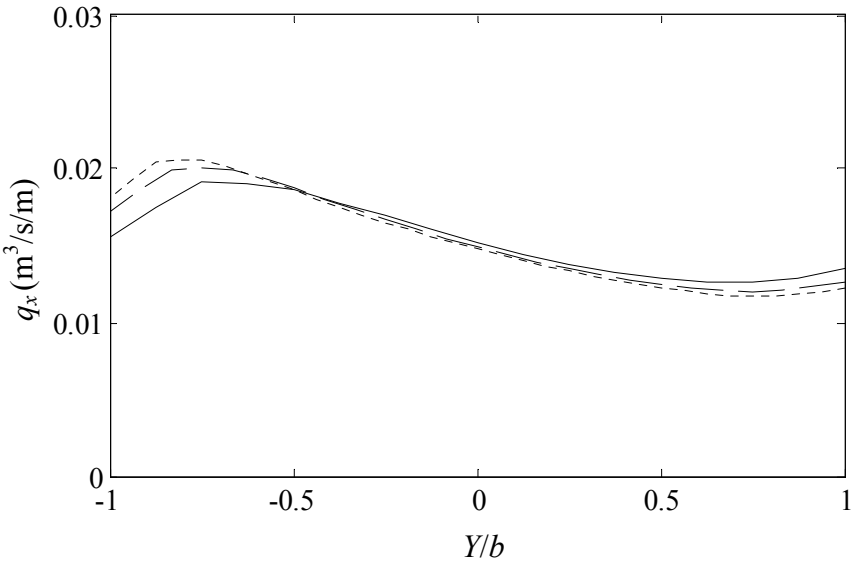

Figure 5.4 Comparison of discharge per unit width (m^3/s/m) for three different meshes in Rozovoskii's experiment. (—) 140X17, (— —) 140X25 and (---) 180X33.

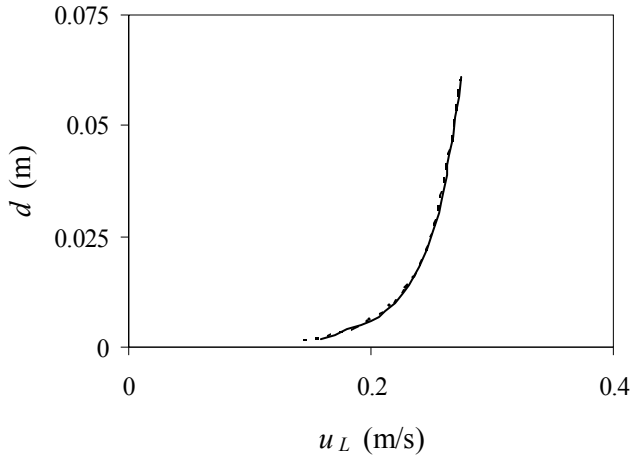

Figure 5.5 Comparison of longitudinal velocity profile at the center of 90^0 of Rozovoskii's bend for the meshes (—) 140X25X15 and (---) 180X33X20. .

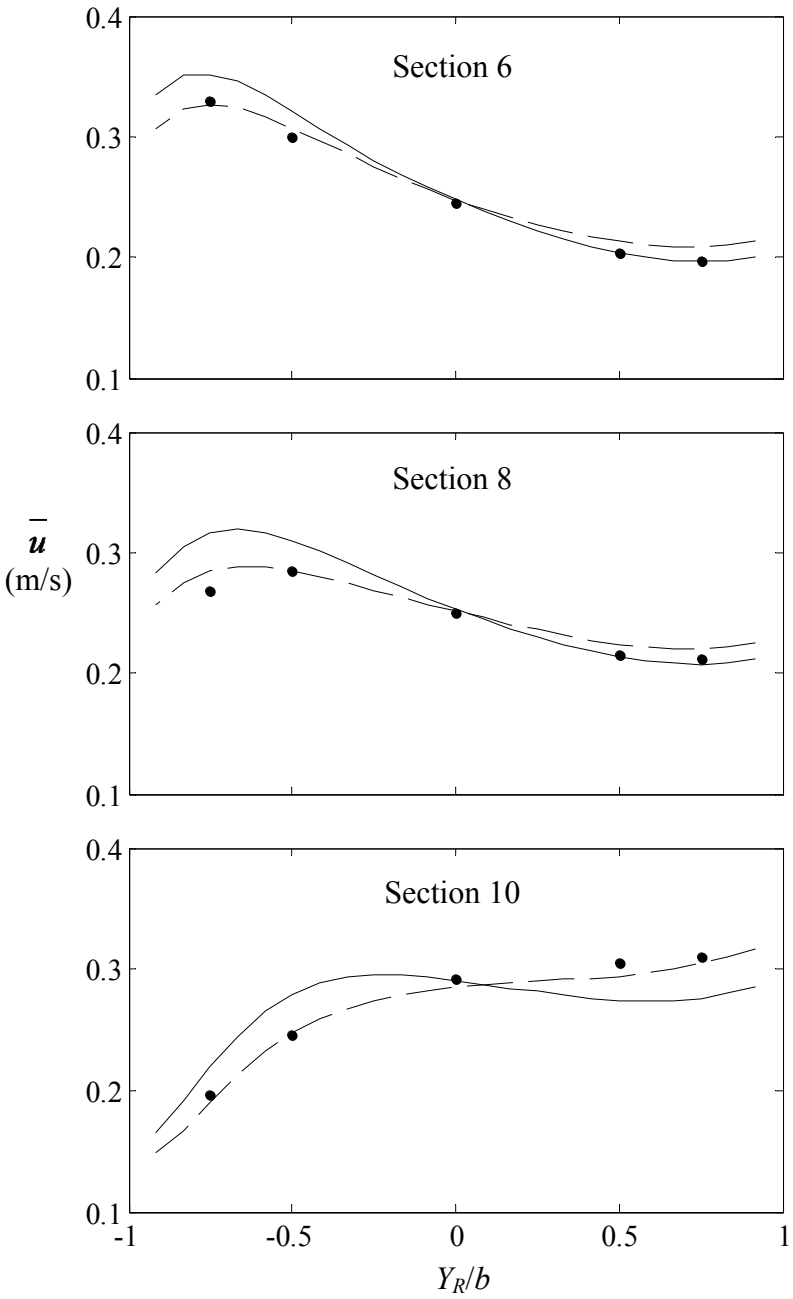

Figure 5.6 Comparison of depth averaged velocity at different sections of Rozovoskii's experiment. (Circle-Experimental, Solid: N=1, Dashed: N=5)

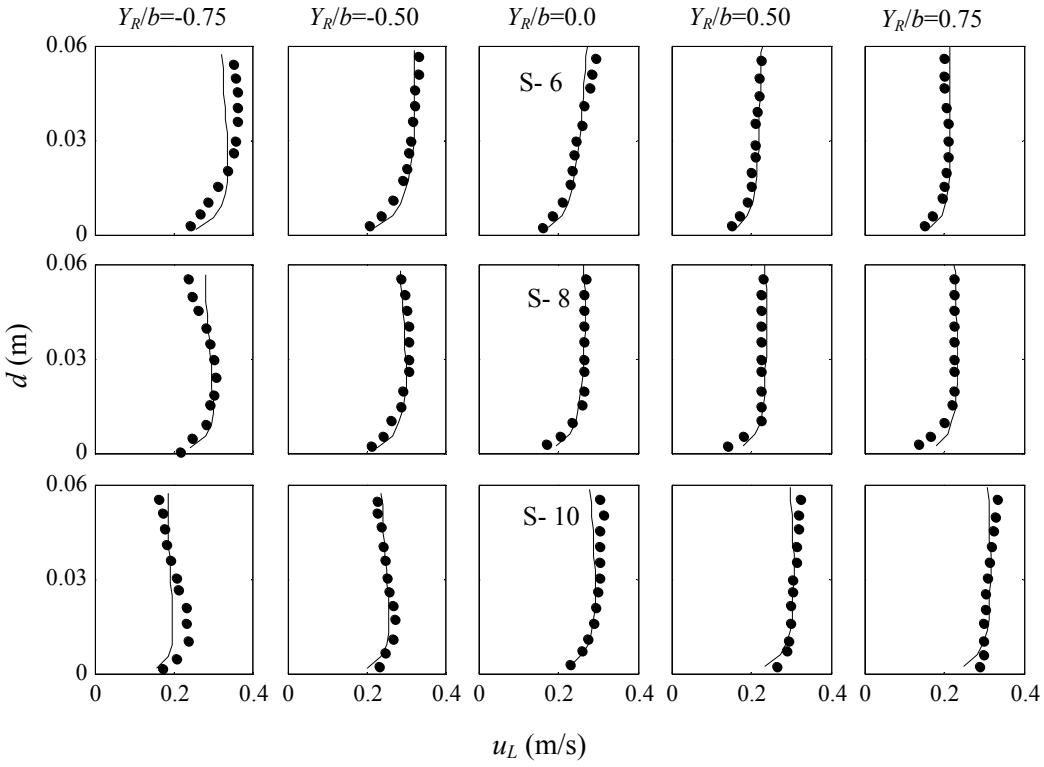

Figure 5.7 Comparison of longitudinal velocity at different sections (S) in Rozovoskii's experiment. (Circle: Experimental, Solid: Numerical N=5)

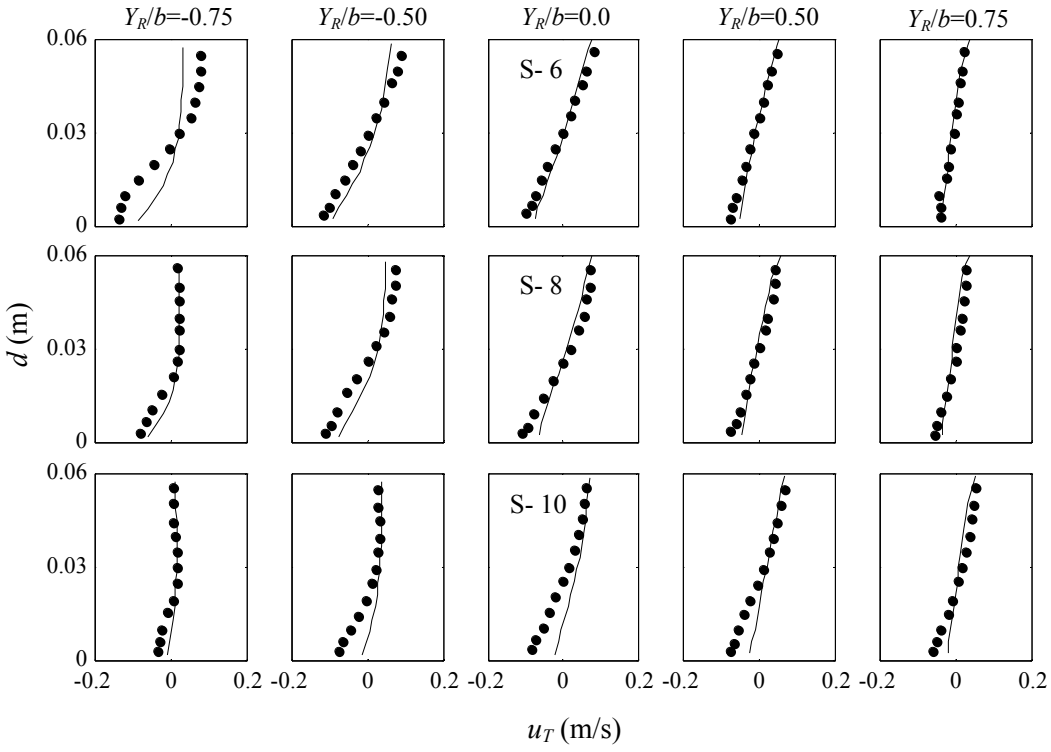

Figure 5.8 Comparison of transeverse velocity at different sections in Rozovoskii's experiment. (Circle: Experimental, Solid: Numerical N=5).

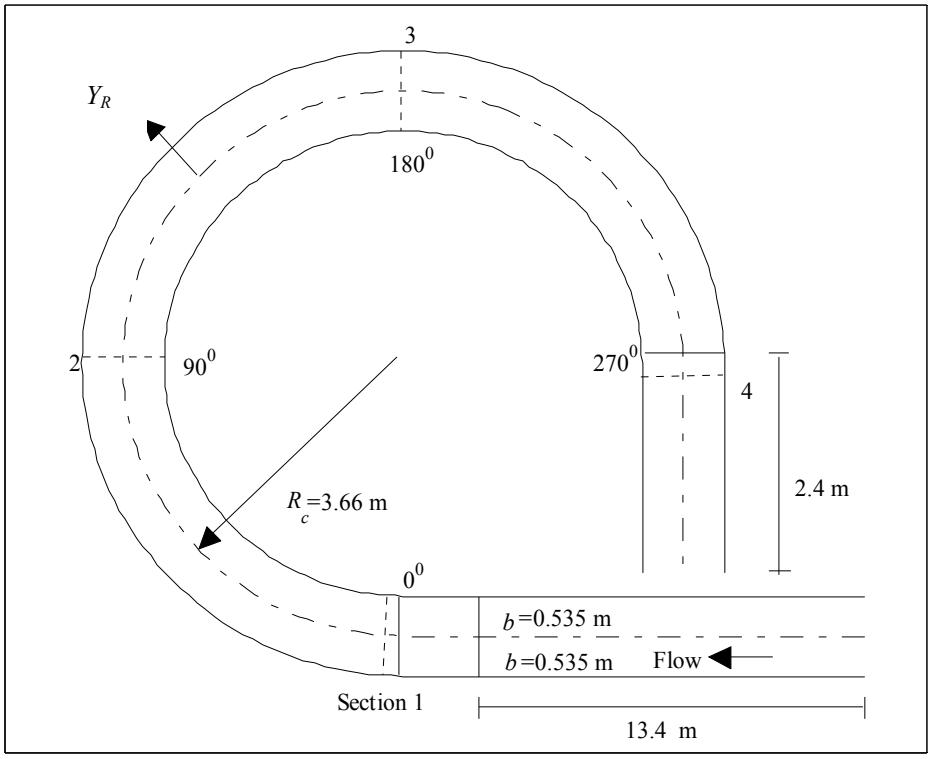

Figure 5.9 Layout of Steffler's experiment

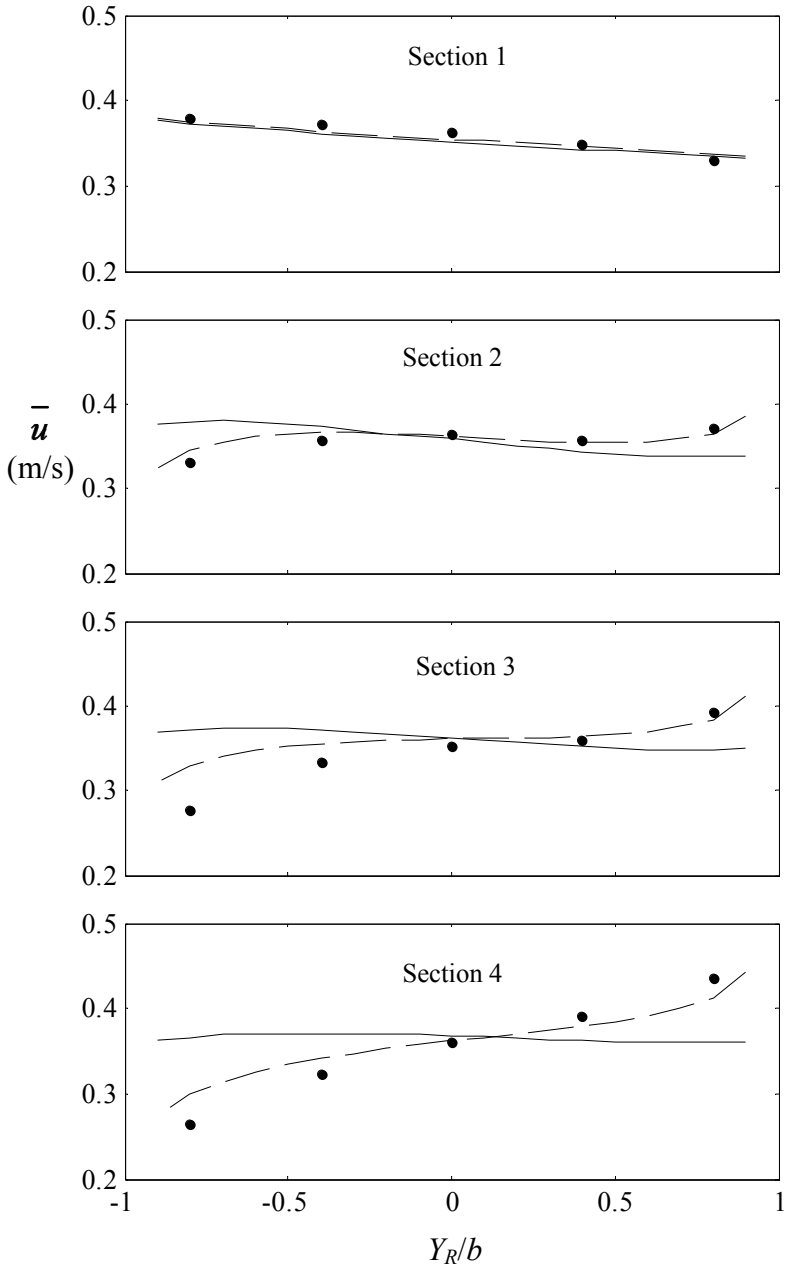

Figure 5.10 Comparison depth averaged velocity of Steffler's experiment. (Circle-Experimental, Solid: N=1, Dashed: N=6)

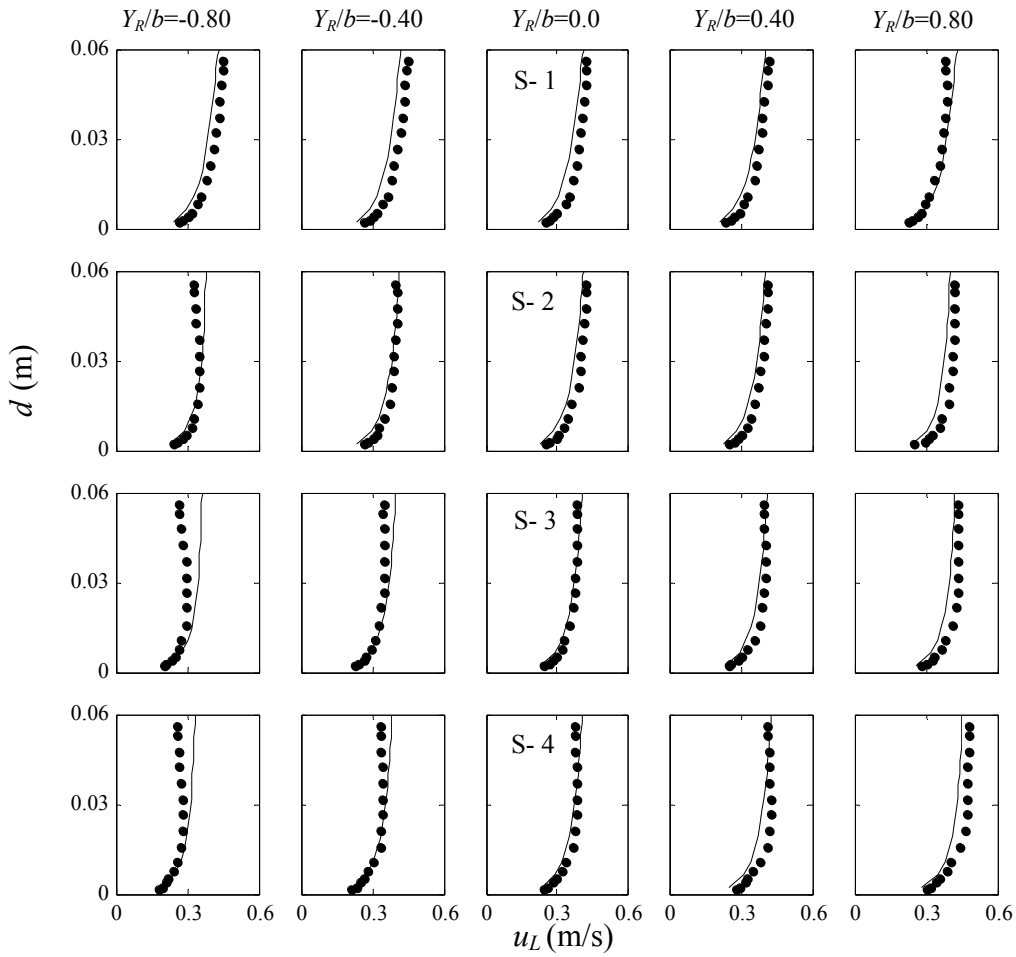

Figure 5.11 Comparision of longitudinal velocity at different sections of Steffler's experiment. (Circle: Experimental, Solid: Numerical N=6,)

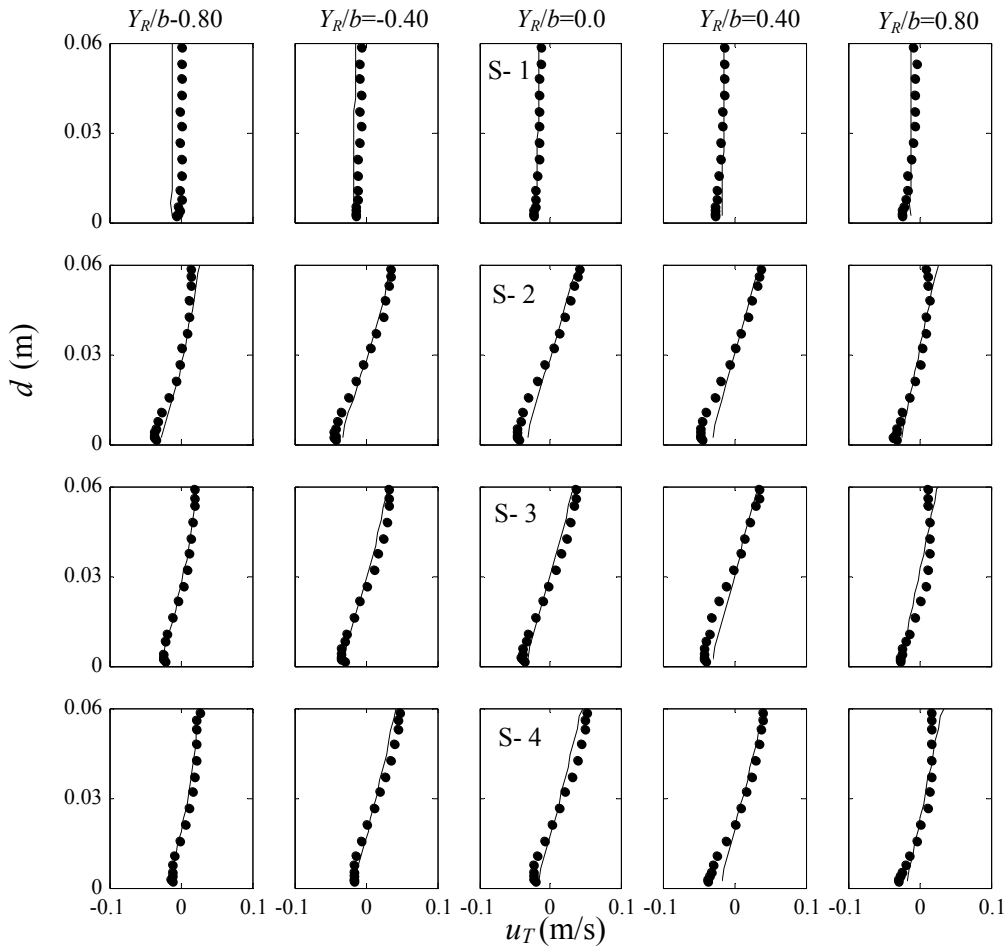

Figure 5.12 Comparison of transverse velocity at different sections of Steffler's experiment. (Circle: Experimental, Solid: Numerical N=6).

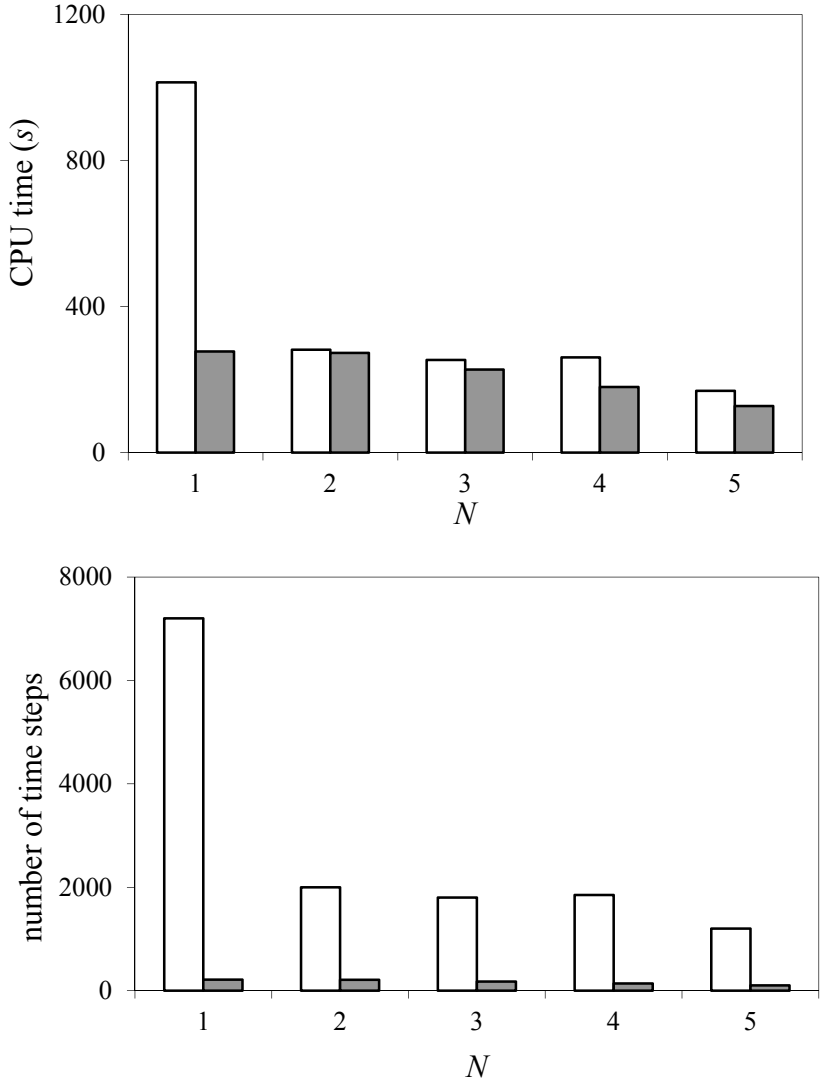

Figure 5.13 Comparison of computational work for Rozovoskii's experiment.

(White: DA model, Grey: RANS Model)

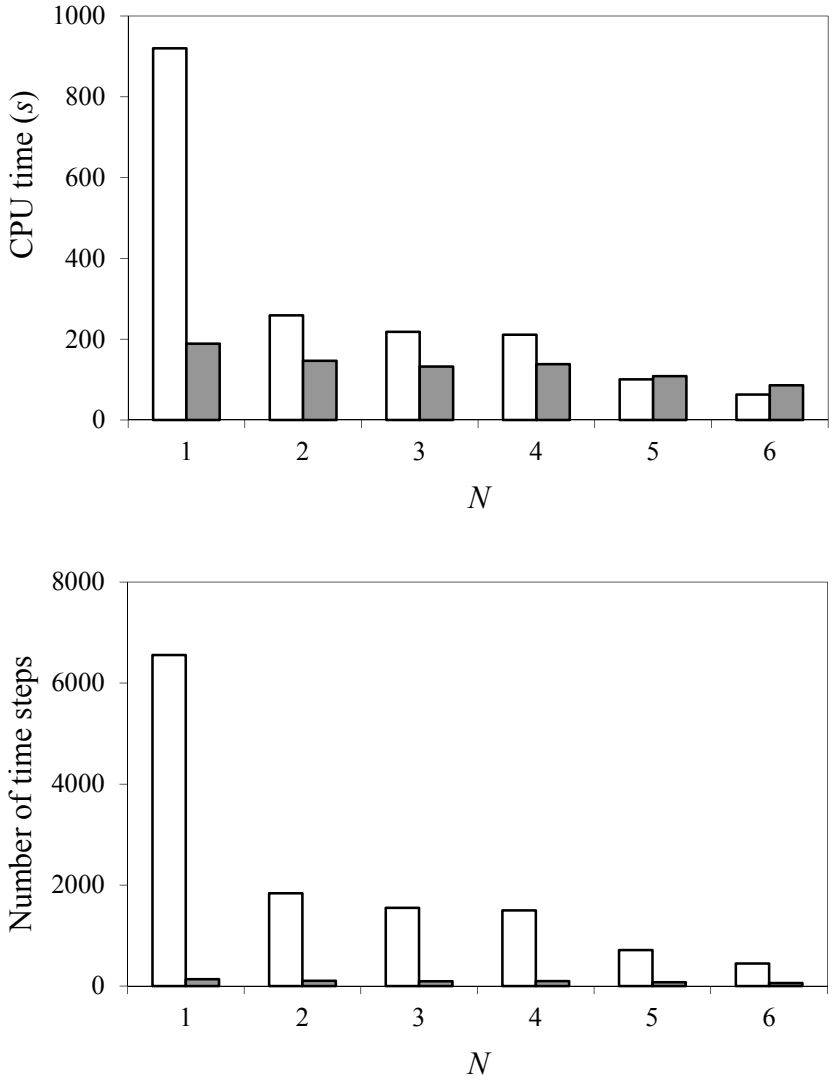

Figure 5.14 Comparison of computational work for Steffler's experiment (White:-DA model, Grey: RANS Model).

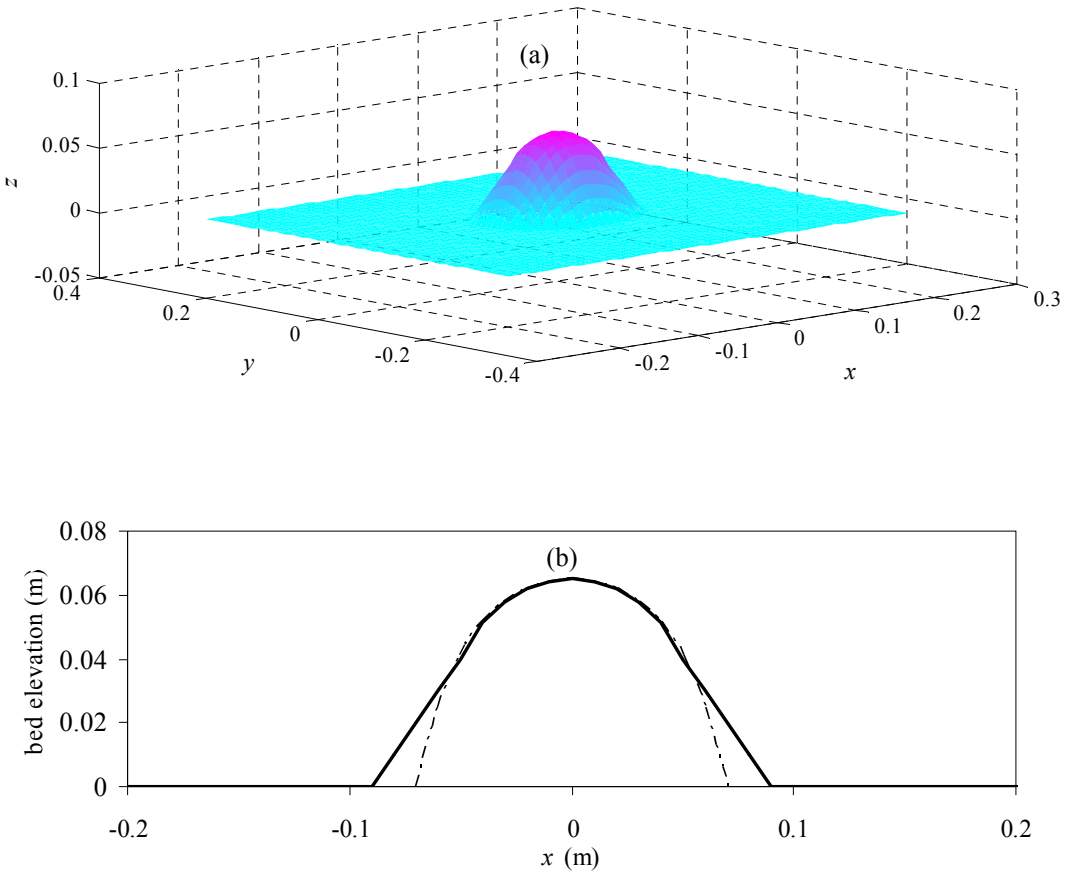

Figure 5.15 Layout of Shamloo's experirment. (a) 3D contour of bed. (b) cross section, Solid: Modified, Dashed: Original geometry.

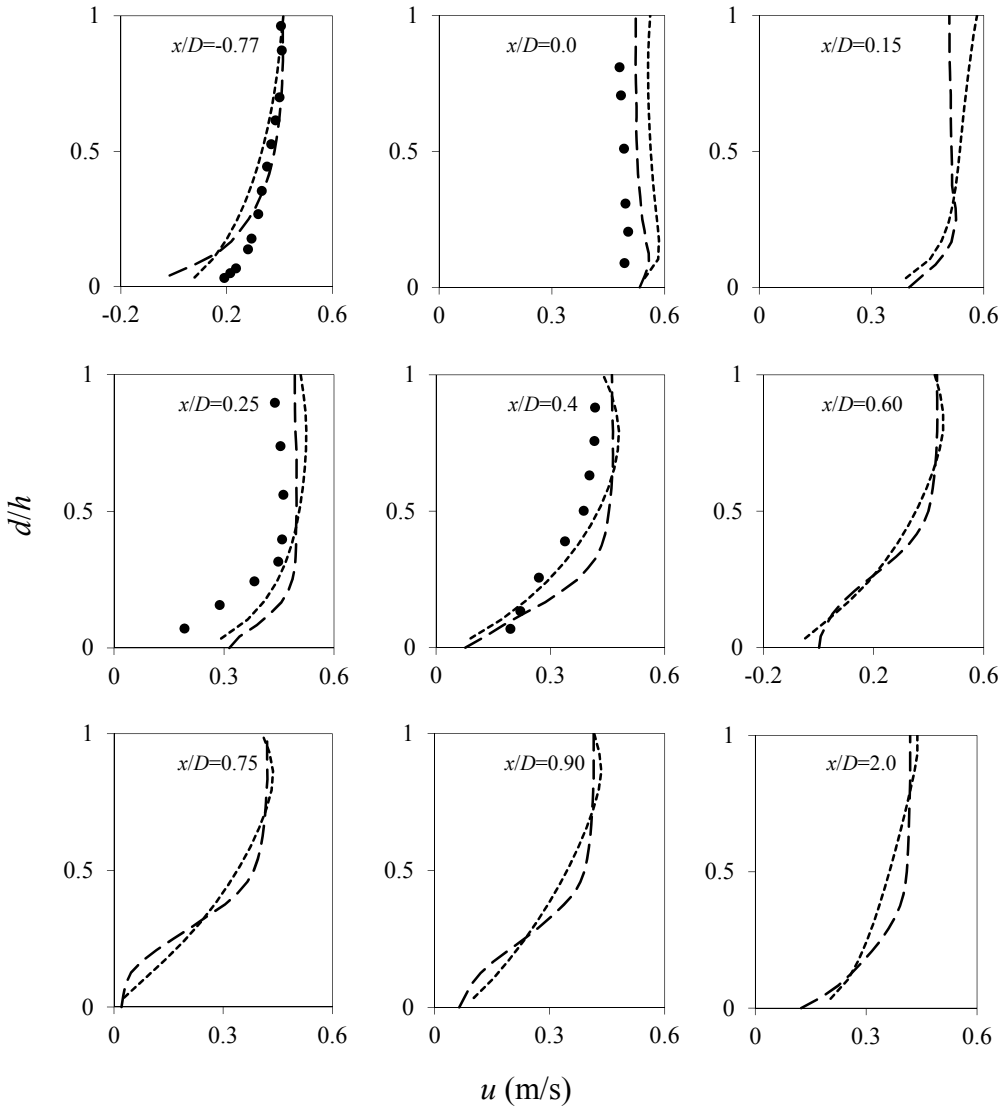

Figure 5.16 Comparison of velocity profiles in Shamloo's experiement.(Circle: experimental, dotted: DA-RANS, dash-CFX)

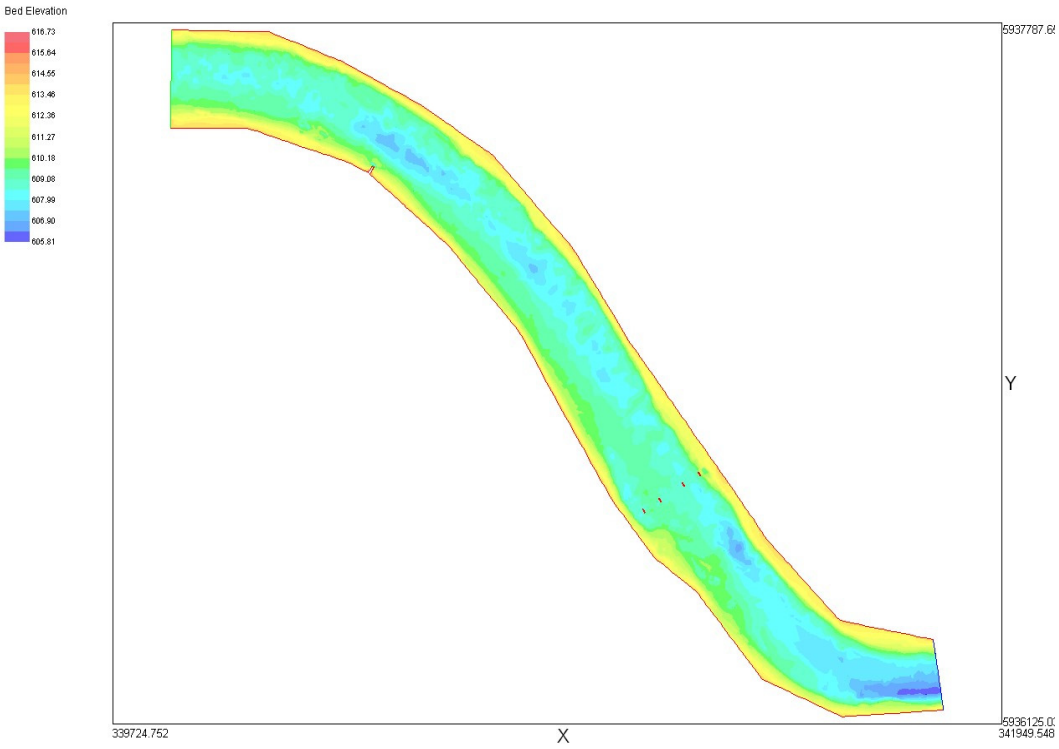

Figure 5.17 Bed contour of the study site of the North Saskatchewan river

Chapter 6

SUMMARY AND CONCLUSIONS

This study can be divided into two parts: developing iterative matrix solvers for a depth averaged river model and developing an open channel flow model by a combination of a depth averaged (DA) model and a Reynolds averaged Navier-Stokes (RANS) model.

Iterative solvers

The implicit modeling of the governing depth averaged equations for open channel flow modeling requires numerous solution iterative solutions of a large linear system. However no previous study has been performed on iterative solvers for the steady open channel flow computation. Therefore this study investigates the behavior of iterative linear solvers for the depth averaged model 'River2D'. This is an implicit model which solves the St. Venant equations discretized by the Characteristic Dissipative Galerkin finite element method. The matrix of the linear system is unsymmetric, therefore the techniques developed for solving an unsymmetric matrix are used. The most widely used methods for an unsymmetric

matrix are the Generalized Minimal Residual (GMRES) and the Bi-Conjugate Gradient Stabilized (BiCGSTAB) methods known as the Krylov subspace methods. For efficiency of these methods various preconditioners such as Jacobi, Symmetric Gauss-Seidel (SGS) and Incomplete Lower Upper (ILU) factorizations are used to improve the efficiency. After performing numerical experiments, it is found that although the ILU preconditioners are generally far better than the other preconditioners, they become unstable for larger time steps.

To improve the ILU preconditioned solvers, a novel technique is introduced where the preconditioners are obtained from the Jacobian matrix of lower time step based on Courant numbers and named as Lower Courant number ILU preconditioner (LCIP). It is found that the LCIP can substantially increase the efficiency and robustness of both GMRES and BiCGSTAB when large time steps are used to simulate the flow. The optimum values of the lower Courant numbers are obtained for the ILU preconditioners. The range is from 5 to 20. For test cases a choice of 15 gives near optimal performance for all ILU preconditioners. Using the optimum Courant numbers, the performance of the ILU preconditioned GMRES and BiCGSTAB solvers are compared, where in most cases ILU(0) preconditioner is found to be better than the other preconditioners except for larger time steps in GMRES where ILU(2) shows a better performance. When comparing the GMRES and BiCGSTAB methods with the ILU(0) preconditioner, BiCGSTAB is found to be better than GMRES. A mesh refinement analysis for ILU (0), ILU(1) and ILU(2) preconditioned GMRES and BiCGSTAB is

performed and the computational work is found to be approximately proportional to N_n^{β}, where N_n is the number of nodes and average value of β is 1.37.

DA-RANS model

A steady flow model is developed and tested by a combination of the depth averaged (DA) and the Reynolds averaged Navier-Stokes (RANS) equations. This model avoids the dynamic free surface computation and uses an explicit-implicit solver. This model also applies new techniques for mass balance and non-hydrostatic pressure computation. The model consists of two parts, a DA model and a RANS model with different modeling options. First a depth averaged model is solved to the steady state to obtain the water surface elevation and depth averaged velocity neglecting the non-uniform velocity (NUV) and non-hydrostatic pressure (NHP). Then for the hydrostatic RANS (RANS-H) model, the RANS horizontal momentum equations are solved to obtain horizontal velocity profiles together with the continuity equation for vertical velocity profiles using the computed water surface as a fixed upper boundary. The mass balance is retained by adding correction terms in the horizontal momentum equations such that the mean of the RANS horizontal velocity is equal to the DA velocity. This also enables modeling the horizontal velocity by only solving the RANS horizontal momentum equations and neglecting the vertical velocity. In the non-hydrostatic model (RANS-NH), the pressure Poisson equation derived from the RANS equations is solved for the non-hydrostatic pressure assuming a zero

pressure at the water surface and using the vertical momentum equation for the bottom node.

The model is first developed for two dimensional plane flow. The DA model is solved by the explicit McCormack scheme. The RANS momentum and the Poisson equations are discretized and solved in a non-orthogonal boundary-fitted coordinate system by the finite difference method with a zero equation turbulence model. For the Poisson equation a vertical line implicit approach is used. A technique is developed to model the correction term. A time marching procedure was used where first the RANS horizontal momentum equation is solved for one time step followed by the vertical velocity and NHP computation. This procedure continues until the steady state RANS solution is achieved. Once the RANS model results are obtained the effects of NUV, bed stress and/or NHP can be introduced in the DA model and the water surface and DA velocity can be recomputed. Then the RANS modeling can be performed using the newly obtained DA results.

The model is verified by simulating and comparing with the experimental results of flow development in a rectangular channel, flow over a symmetric hump and flow over a dune. The effect of the feedback on model performance and the relation between the DA and RANS models through the correction term are investigated. The correction term generally reflects the mathematical and numerical differences between the two models. The model can predict the horizontal velocity profiles and the non-hydrostatic pressure very well except the

flow recirculation in the flow over a dune. A comparison of the computational time of the DA and the RANS models are also performed which shows that the RANS model is as fast as the DA model.

The model is then extended for three dimensional (3D) flow using numerical techniques similar to the 2D plane flow. The 2D DA model requires artificial viscosity for stability. The model was first verified by simulating the flow in two curved open channels for which experimental data are available. For this case the NHP and vertical velocity are neglected. In the feedback process between the DA and RANS model, only the dispersion term that is the effect of the NUV is used. The flow shifting from the inner towards the outer bank is well simulated. Next the non-hydrostatic model is verified by simulating the flow over a modified hemisphere and comparing the results with the experiment and the results from a commercial CFD model (CFX). But in this case the feedback process cannot be performed as the DA model becomes unstable. Finally the computational efficiency of the RANS model is examined by simulating the flow in a 1.3 km reach of the North Saskatchewan river where the hydrostatic and the non-hydrostatic RANS models require 1.0 and 1.5 hours for convergence with over 700,000 computational nodes. All the computer programs for the DA-RANS model are written in Matlab® and run in a personal computer with a 2.8 GHz processor .

Several important contributions are achieved in this research:

a) A novel and robust technique is developed for the iterative solution of linear system by incomplete lower upper (ILU) preconditioned Krylov subspace methods. This technique can be adopted in any numerical model that uses a pseudo transient term to converge to steady state.

b) The computational work by the Krylov methods preconditioned by the new technique is proportional to the number of nodes with an exponent of 1.37.

c) The coupled DA-RANS model provides a wide range of modeling options. The user can make the decision based on the DA results, whether a RANS solution is required. If a RANS model is needed, it can be hydrostatic with or without considering the vertical velocity, or non-hydrostatic. Once the first RANS solution is obtained, the user can decide if there is merit in performing the feedback iteration and if so, the user can select the important factors such as the dispersion, NHP or stress. Conceivably, the software could be written to provide these refined solutions for only selected portions of the modelled domain.

d) Finally the excellent computation efficiency of the RANS model will enable the modeling of large domains with a refined mesh.

Also several issues have not been addressed in this research but can be done in future:

a) For the study on the linear solver, only the finite element model with the steady state solution is tested. Other techniques such as the finite volume methods with higher order discretizations can be investigated. Also the case of unsteady flow can be tested.

b) In the coupled model development, McCormack scheme is used for the DA model which is generally prone to oscillation and stability and requires artificial viscosity. On the other hand, the RANS model is discretized by a finite difference method. Moreover, only a first order upwind method is used in the RANS model. Other techniques especially the finite element technique can be used as it provides flexibility to model the complex natural geometry. As such the 3D model can be developed using the framework of the River2D model.

c) The coupled model is developed for steady flow computation. The model can be adapted for unsteady flow calculations where the coupling between the DA and the RANS model will take place at each time step.

d) Only zero equation turbulence models are used in the RANS model. One or two equation models can also be tested especially for the simulation of the flow over the dunes.

REFERENCES

Barragy, E., Carey, G.F. (1993). Application of conjugate gradient methods to tidal simulation. *Advances in Water Resources*, 16, 163-171.

Bertin, J.J. (1987) Engineering fluid mechanics. 2nd edition, Prentice Hall, Englewood Cliffs, NJ.

Buelow, P.E.O., Venkateswaran, S., Merkle,C.L. (1994). Effect of grid aspect ratio on convergence. *AIAA J.*, 32 (12), 2401-2409.

Chapman, A., Saad, Y. and Wigton, L. (2000). High-order preconditioners for CFD problems. *Int. J. Numer. Meth. Fluids.*, 33, 767-788.

Chaudhry, M.H. (1993). Open channel flow. Prentice-Hall, Englewood Cliffs, N.J.

Chorin, A.J. (1967). A numerical method for solving incompressible viscous flow problems. *J. Comput. Phys.*, 2, 12-26.

Chow, E., Saad, Y. (1997). Experimental study of ILU preconditioners for indefinite matrices. *J. Comp. Appl. Math.*, 86, 387-414.

Chung, T.J. (2002). Computational fluid dynamics. Cambridge University Press, Cambridge, UK.

Cuthill, E.H., McKee, J.M. (1969). "Reducing the bandwidth of Sparse Symmetric Matrices. *Proc. of the 24th National Conference of the Association for Computing Machinery*, Brondon Press, 157-172.

Dargahi, B. (2004). Three-dimensional flow modeling and sediment transport in the river Klaralven. *Earth Surf. Process. Landforms*, 29, 821-852.

Daubert, A., Graffe, O. (1967). Quelques aspects des ecoulements Presque horizontaux s deux dimensions en plan et non permanents application aux estuaries. La Houille Blanche, 8, 847-860.

Demuren, A.O., Rodi, W. (1986). Calculation of flow and pollutant dispersion in meandering channels. *J. Fluid Mech.* 172, 63-92.

De Saint-Venant, B. (1871). Theorie du movement non permanent des eausx, avec application aux crues de rivieras et a l'introduction des merces dans leur lit. Comptes Rendus de l'Academic des Science, 73: 147-154, 237-240.

DeVriend, H.J. (1976). A mathematical model of steady flow in curved shallow channels. *Report No. 76-1, Dept. of Civ. Eng.*, Delft University of Technology.

DeVriend, H.J. (1976). Steady flow in shallow channel bends- Part I. *Report No. 81-3, Dept. of Civ. Eng.*, Delft University of Technology.

Dow, K. (2003). Intermediate field mixing of wastewater effluent in the North Saskatchewan river at Edmonton Alberta. *M.Sc. thesis*, University of Alberta, Canada.

Dressler, R.F., (1978). New nonlinear shallow-flow equations with curvature. *J. Hydraul. Res.*, 16(3) 205-220.

Dutto, L.C. (1993). The effect of reordering on the preconditioned GMRES algorithm for solving the compressible Navier-Stokes equations. *Int. J. Numer. Meth. Eng.*, 36, 457-497.

Elgamal, M.H. (2002). Applications of moment approach to bedforms. *Ph.D. thesis*. University of Alberta, Canada.

Engelund, F. (1974). Flow and bed topography in channel bends. *J. Hydraul. Eng.*, 100 (11), 1631-1648.

Fang, C.C., Sheu, T.W.H. (2001). Two element by element iterative solutions for shallow water equations. *SIAM J. Sci. Comput.*, 22 (6), 2075-2092.

Feng, Y., Peric, D., Owen D., (1998) A Multigrid enhanced GMRES algorithm for elasto-plastic problems. *Int. J. Numer. Meth. Engng.*, 42, 1441-1462.

Fisher, H.B. (1979). Mixing in inland and coastal waters. Academy press.

Fletcher, R. (1975). Conjugate gradient methods for indefinite systems. In G.A. Watson editor, *Proc. of the Dundee Biennial Conference on Numerical Analysis*, Springer Verlag, NewYork.

Fletcher, C.A.J. (2006). Computational techniques for fluid dynamics- Volume II. Springer-Verlag.

Garcia-Navarro, P., Alcrudo, F., Saviron, J.M. (1992). 1-D open channel flow simulation using TVD-McCormack scheme. *J. Hydraul. Eng.*, 118(10), 1359-1372.

Ghamry, H. (1999). Two dimensional vertically averaged and moment equations for shallow free surface flow. PhD thesis, Department of Civil Engineering, University of Alberta.

Gramry, H., Steffler, P.M., (2002). Effect of applying different distribution shapes for velocities and pressure on simulation of curved open channels. *J. Hydraul. Eng.*, 2(11), 969-982.

Gramry, H., Steffler, P.M., (2005). Two-dimensional depth-averaged modeling of flow in curved open channels. *J. Hydraul. Res.*, 43(1), 44-55.

Ghanem, A. (1995). Two-dimensional finite element modeling of flow in aquatic habitats. *Ph.D. thesis*, Department of Civil Engineering, University of Alberta.

Ghanem, A., Steffler, P., Hicks, F., Katopodis, C. (1995). Two-dimensional finite element modeling of flow in aquatic habitats. Water Resources Engineering Report 95-S1. University of Alberta.

Hager, W.H., Hutter, K. (1984). Approximate treatment of plane channel flow. *Acta Mechanica*, 51, 31-48.

Harten, A. (1983). High resolution schemes for hyperbolic conservation laws. *J. Comput. Phys.* 49, 357-393.

Hicks, F.E. (1990). Finite element modeling of open channel flow. *Ph.D. thesis.* Department of Civil Engineering, University of Alberta.

Hicks, F. E., Steffler, P. M. (1992) Characteristic dissipative Galerkin scheme for open-channel flow. *J. of Hydraul. Eng.*, 118 (2), 337-352.

Hirt, C.W., Nichols, A. (1981). Volume of fluid (VOF) methods for the dynamics of free boundaries. *J. Comput. Phys.*, 39, 201-225.

Issa, R.I. (1986). Solution of the implicitly discretized fluid flow equations by operator-splitting. *J. Comput. Phys.*, 62, 40-65.

Jameson, A., Schimdt, W., Turkel, E. (1981). Numerical solutions of the Euler equations by finite volume methods using Runge-Kutta time stepping schemes. *Proc. AIAA 14th Fluid and Plasma Dynamics Conf., AIAA paper*-81-1259, Palo Alto, Calif.

Jameson, A. (1985). Multigrid algorithms for compressible flow calculations. *MAE Report*, *1743*, Princeton Univ., Princeton, N.J.

Jia, Y., Steve, S., Xu, Y., Huang, S., Wang, S.S.Y. (2005). Three-dimensional numerical simulation and analysis of flows around a submerged weir in a channel bendway. *J. Hydraul. Eng.*, 131 (8), 682-693.

Jia, Y., Scott, S., Xu, Y. and Wang, S.S.Y. (2009) Numerical study of flow affected by bendway weirs in Victoria bendway, the Mississippi river. *J. Hydraul. Eng.*, 135 (11), 902-916.

Jin, Y.-C., Steffler, P.M. (1993). Predicting flow in curved open channel by depth averaged method. *J. Hydraul. Eng.,* 119(1), 109-124.

Jin, Y.-C., Li, B. (1996). The use of a one-dimensional depth averaged moment of momentum equation for the non-hydrostatic pressure condition. *Can. J. Civ. Eng.,* 23(10), 150-156.

Johannesson, H., Parker, G. (1989). Velocity redistribution in meandering rivers. *J. Hydraul. Eng.,* 115(8), 1019-1039.

Kalkwijk, J.P.TH., DeVriend, H.J. (1980). Computation of the flow in shallow river bends. *J. Hydraul.Res.*18(4), 327-341.

Khan, A.A. (1995). Modeling rapidly varied open channel flows. *PhD thesis*, Department of Civil Engineering, University of Alberta, Edmonton.

Kwak, D., Chang, J., Shanks, S.P., Chakravarthy, S.R. (1986). A three dimensional incompressible Navier-Stokes solver using primitive variables. *AIAA J.*, 24, 390-396.

Lackey, T.C. and Sotiropoulos, F. (2005), Role of artificial dissipation scaling and multigrid acceleration in numerical solutions of the depth-averaged free-surface flow equations. *J. Hydraul. Eng.,* 131 (6), 476-487.

Lai, C.J., Yen, C.W. (1993). Turbulent free surface flow simulation using a multilayer model. *Int. J. Numer. Meth. Fluids*, 16(11), 1007-1025.

Launder, B.E., Spadling, D.B. (1974). The numerical computation of turbulent flows. *Comput. Meth. Appl. Mech. Eng.*, 3, 269-289.

Lax, P.D., Wendroff, B. (1960). Systems of conservation laws. *Comm. Pure Appl. Math.*, 15, 363.

Leonard, B.P. (1979). A stable and accurate convective modeling procedure based on quadratic upstream interpolation. *Comput. Meth. Appl. Mech. Eng.*, 19, 59-98.

Leonard, B.P. (1988). Simple high-accuracy resolution program for convective modeling of discontinuities. *Int. J. Numer. Meth. Fluids*

Li, C.W., Yu, T.S. (1996). Numerical investigation of turbulent shallow recirculating flow by a quasi-three-dimensional κ-ε model. *Int. J. Numer. Meth. Fluids*, 23(5), 485-501.

Lien, F.S., Leschziner, M.A. (1994). Upstream monotonic interpolation for scalar transport with application to complex turbulent flows. *Int. J. Numer. Meth. Fluids*, 19 (6), 527-548.

Lin, H., Yang, D.Y., Chieng, C., (1995). Variants of bi-conjugate gradient method for compressible Navier-Stokes solver. *AIAA J.*, 33(7), 1177-1184.

Louaked, M., Hanich, L. (1998). TVD schemes for shallow water equations. *J. Hydraul. Res.*, 36(3), 363-378.

MacCormack, R.W. (1969). The effect of viscosity in hypervelocity impact catering. *AIAA Paper*, 66-354.

Majumder, S. (1988). Role of underrelaxation in momentum interpolation for calculation of flow with non-staggered grids. *Numer. Heat Trans.* 13, 125-132.

Manteuffel, T.A. (1980). An incomplete factorization technique for positive definite linear systems. *Math. Comp.*, 34, 473-497.

Manzano, L., Lassaline, J., Wong, P. and Zingg, D. (2003). A Newton-Krylov algorithm for the Euler equations using unstructured grids. *41th AIAA Aerospace Sciences Meeting and Exhibit. AIAA paper*, 2003-0274.

Meselhe, E. A., Sotiropoulos, F. 2000. Three-dimensional numerical model for open-channels with free-surface variations. *J.Hydraul.Res.*, 38, 115–121.

Morvan, H., Pender, G., Wright, N.G., Ervine, D.A. (2002). Three-dimensional hydrodynamics of meandering compound channel. *J. Hydraul. Eng.*, 128 (7), 674-682.

Nejat, A., Ollivier-Gooch, C. (2008a). Effect of discretization order on preconditioning and convergence of a high-order unstructured Newton-GMRES solver for the Euler equations. *J. Comput. Phys.*, 227, 2366-2386.

Nejat, A., Ollivier-Gooch, C. (2008b). A high-order accurate unstructured finite volume Newton-Krylov algorithm for inviscid compressible flow. *J. Comput. Phys.*, 227, 2582-2609.

Nezu, I., Nakagawa, H. (1993). Turbulence in open-channel flows. IAHR monograph.

Nguen, V.TH., Nestmann, F., Scheuerlein, H. (2008). Three-dimensional computation of turbulent flow in meandering channels and rivers. *J. Hydraul. Res.*, 45(5), 595-609.

Nichols, J., Zingg, D.W. (2005). A three-dimensional multi-block Newton-Krylov flow solver for the Euler equations. *Proceedings of the Seventeenth AIAA Computational Fluid Dynamic Conference, American Institute of Aeronautics and Astronautics.*

Olsen, N.R. (2009). A three-dimensional numerical model for simulation of sediment movements in water intakes with multiblock option- User's manual.

Patankar, S.V. (1980). Numerical heat transfer and fluid flow. Hemisphere Publishing Corporation,New York, NY.

Prandtl, L. (1925). Uber die ausgebildete turbulenz. ZAMM, 5, 136.

Pueyo, A. and Zingg, D. (1998). Efficient Newton-Krylov solver for aerodynamic computations. *AIAA J.*, 36 (11), 1991-1997.

Rajaratnam, N. (1987). Engineering fluid mechanics. Department of Civil Engineering, University of Alberta, Edmonton.

Ranga Raju, K.G., Asawa, G.L., Mishra, H.K. (2000) Flow-establishment length in rectangular channels and ducts. *J. Hydraul. Eng.*, 126(7), 533-539.

Rhie, C.M., Chow, W.L. (1983). Numerical study of the turbulent flow past an airfoil with trailing edge separation. *AIAA J.*, 21 (11), 1525-1532.

Rodi, W. (1984). Turbulence models and their application in hydraulics- a state of the art paper. *IAHR*, Delft, the Netherlands.

Roe, P.L. (1985). Some contributions to the modeling of discontinuous flows. *Lectures in Applied Mechanics*, 22, Springer-Verlag, Berlin, 163-193.

Rogers, S.E. (1995). Comparison of implicit schemes for the incompressible Navier-Stokes equations. *AIAA J.*, 33 (11), 2066-2072.

Rozovskii, I. L. 1961. Flow of water in bends of open channels. *Acad. Sci. Ukrainian SSR.*, Translated from Russian, Israel Program for Science Translation, 1–233.

Saad, Y., Schultz, M. (1986). GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear System. *SIAM J. Sci. Stat. Comput.*, 7 (3), 856-869.

Saad, Y. (1994). Preconditioned Krylov subspace methods for CFD applications. *In: W.G Habashi (Ed.), Proceedings of International Workshop on Solution Techniques for Large-Scale CFD Problems*, Montréal, Québec, 179-195.

Saad, Y. (1994). ILUT: A dual threshold incomplete ILU factorization. *Numer. Linear Algebra Appl.*, 1, 387-402.

Saad, Y. (1996). Iterative methods for sparse linear systems. PWS publishing: New York.

Schlichting, H. (1979). Bounday Layer Theory. Mcgraw-Hill, New York.

Shamloo, H., (1996). Hydraulics of simple habitat structures in open channels. *PhD thesis*. University of Alberta, Edmonton, Canada.

Sheen, S.C., Wu, J.L. (1998). Preconditioning techniques for the BiCGSTAB algorithm used in convection-diffusion problems. *Numer. Heat Transfer, Part B,* 34, 241-256.

Shimizu, Y., Yamaguchi, H., Itakura, T. (1990). Three-dimensional computation of flow and bed deformation. *J. Hydraul. Eng.*, 116(9) 1090–1108.

Sinha, S. K., Sotiropoulos, F., Odgaard, J. (1998). Three-dimensional numerical model for flow through natural rivers. *J. Hydraul. Eng.*, 124 (1), 13-24.

Sivakumaran, N.S., Tingsanchali, T. ans Hosking, R.J. (1983) Steady shallow flow over curved beds. *J. Fluid Mech.* Vol. 128, pp 469-487.

Sonneveld, P. (1989). CGS, a fast Lanczos-type solver for nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 10, 36-52.

Spadling, D.B. (1972). A novel finite difference formulation for differential expressions involving both first and second derivatives. *Int. J Numer. Meth. Eng.*, 4, 55-559.

Steffler (1984). Turbulent flow in a curved rectangular channel. *PhD thesis*, University of Alberta, Edmonton, Canada.

Steffler, P.M., Jin, Y. (1993). Depth-averaged and moment equations for moderately shallow free surface flow. *J. Hydraul. Res.*, 31(1), 5-17.

Steffler, P., Blackburn, J. (2002). River2D: Two-dimensional depth-averaged model of river hydrodynamics and fish habitat- Introduction to depth averaged modeling and user's manual. University of Alberta, Canada.

Thompson, J.F., Warsi, Z.U.A., Martin, C.W. (1985). Numerical grid generation, foundations and applications. North-Holland, Amsterdam.

Trottenberg, U., Oosterlee, C.W., Schüller, A. (2001). Multigrid. Academic Press. London, UK.

van Albada, G.D., van Leer, B. Roberts, W.W. (1982). A comparative study of computational methods in cosmic gas dynamics. *Astron. Astrophys.*, 108, 76-84.

Van der Vorst, H.A. (1981). Iterative solution methods for certain sparse linear systems with a non-symmetric matrix arising from PDE-problems. *J. of Comput. Phys.*, 44, 1-19.

Van der Vorst, H.A. (1992). A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 13 (2), 631-644.

van Doormal, J.P., Raithby, G.D. (1984). Enhancements of the SIMPLE method for predicting incompressible flows. *Numer. Heat Transfer*, 7, 147-163.

van Leer, B. (1974). Towards the ultimate conservative differencing II. Monotinicity and conservation combined in a second-order scheme. *J. Comput. Phys.*, 14, 361-370.

van Mierlo, M.C.L.M., de Ruiter, J.C.C. (1988). Rivers: Turbulence measurements above artificial dunes. Report Q-798, delft Hydraulics Laboratory, Delft, the Netherlands.

van Rijn, L.C. (1990) Principles of fluid flow and surface waves in rivers, estuaries, seas, and oceans. Aqua publications, Amsterdam, the Netherlands.

Venkatakrishnan, V., Mavriplis, D. (1993). Implicit solvers for unstructured meshes. *J. of Comput. Phys.*, 105, 83-91.

Versteeg, H.K., Malalasekera, W. (2007). An introduction to computational fluid dynamics. 2nd Edition, Pearson Education Limited, Essex, England.

Vreugdenhill, C.B. (1994). Numerical methods for shallow-water flow. Kluwer Academic Publishers, Dordrecht, the Netherlands.

Wang, J.S., Ni, H.G., He, Y.S. (2000). Finite-difference TVD scheme for computation of dam-break problems. *J. Hydraul. Eng.*, 126(4), 253-262.

Weiyan, T. (1992). Shallow water hydrodynamics. Elsevier Oceanography Series 55, Elsevier, Ansterdam.

White, F.M. (1986). Fluid mechanics. 2nd Edition, McGraw-Hill Book Company, New York, NY.

Wilson, C. A. M. E., Boxall, J. B., Guymer, I., Olsen, N. R. B. (2003). Validation of a three-dimensional numerical code in the simulation of pseudo-natural meandering flows. *J. of Hydraul. Eng.*, 129 (10), 758-768.

Wu, W., Rodi, W., Wenka, T. (2000). 3D numerical modeling of flow and sediment transport in open channels. *J. Hydraul. Eng.*, 126 (1), 4-15.

Ye, J., McCorquadale, J.A. (1998) Simulation of curved open channel flows by 3D hydrodynamic model. *J. Hydraul. Eng.*, 124 (7), 687-697.

Yen, B.B. (1965). Characteristics of subcritical flow in a meandering channel. *Inst. Of Hydr. Res.,* University of Iowa.

Xia, C., Jin, Y.-C. (2006). Multilayer averaged and moment equations for one-dimensional open-channel flows. *J. Hydraul. Eng.*, 132 (8), 839-849.

APPENDIX A

BOUNDARY-FITTED COORDINATE TRANSFORMATION

In this thesis, the RANS equations for 2D plane flow and 3D flow, and the 2D DA equations are transformed from the physical Cartesian (x, y, z) to the generalized boundary-fitted coordinate system (ξ, η, ζ). ξ follows the bed and the water surface in the longitudinal direction while η follows them in the transverse direction. ζ is in the vertical direction and parallel to the z axis (Figure 5.2). The new coordinates are generally non-orthogonal i.e. the angle between the new coordinates may deviate from 90^0.

Based on the functional relationships, $\xi = \xi(x,y,z)$, $\eta = \eta\ (x,y,z)$ and $\zeta = \zeta\ (x,y,z)$, any partial derivative can be transformed into the derivatives corresponding to the boundary-fitted coordinate system (ξ, η, ζ).

For example the first derivative of the velocity component u with respect to x can be written as

$$\frac{\partial u}{\partial x} = \frac{\partial u}{\partial \xi}\frac{\partial \xi}{\partial x} + \frac{\partial u}{\partial \eta}\frac{\partial \eta}{\partial x} + \frac{\partial u}{\partial \eta}\frac{\partial \eta}{\partial x}$$

Similarly the first derivative of the three velocity components (u,v,w) with respect to x, y and z can be written in the matrix form as:

$$\begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} & \frac{\partial u}{\partial z} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} & \frac{\partial v}{\partial z} \\ \frac{\partial w}{\partial x} & \frac{\partial w}{\partial y} & \frac{\partial w}{\partial z} \end{bmatrix} = \begin{bmatrix} \frac{\partial u}{\partial \xi} & \frac{\partial u}{\partial \eta} & \frac{\partial u}{\partial \zeta} \\ \frac{\partial v}{\partial \xi} & \frac{\partial v}{\partial \eta} & \frac{\partial v}{\partial \zeta} \\ \frac{\partial w}{\partial \xi} & \frac{\partial w}{\partial \eta} & \frac{\partial w}{\partial \zeta} \end{bmatrix} \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \xi}{\partial y} & \frac{\partial \xi}{\partial z} \\ \frac{\partial \eta}{\partial x} & \frac{\partial \eta}{\partial y} & \frac{\partial \eta}{\partial z} \\ \frac{\partial \zeta}{\partial x} & \frac{\partial \zeta}{\partial y} & \frac{\partial \zeta}{\partial z} \end{bmatrix}$$

where the Jacobian matrix of transformation is defined as

$$J \equiv \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \xi}{\partial y} & \frac{\partial \xi}{\partial z} \\ \frac{\partial \eta}{\partial x} & \frac{\partial \eta}{\partial y} & \frac{\partial \eta}{\partial z} \\ \frac{\partial \zeta}{\partial x} & \frac{\partial \zeta}{\partial y} & \frac{\partial \zeta}{\partial z} \end{bmatrix}$$

Usually it is not possible to evaluate the elements of the Jacobian matrix directly as no explicit analytical relationship of the transformed coordinates with the Cartesian coordinates is available (Fletcher, 2006). Therefore the inverse Jacobian matrix is used which is defined as

$$J^{-1} \equiv \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \eta} & \frac{\partial z}{\partial \zeta} \\ \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \eta} & \frac{\partial z}{\partial \zeta} \\ \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \eta} & \frac{\partial z}{\partial \zeta} \end{bmatrix}$$

The elements of the inverse Jacobian matrix J^1 can be evaluated numerically. Then the elements of the Jacobian matrix can be calculated as

$$J = \frac{Transpose\, of\ cofactor\ of\ J^{-1}}{\left|J^{-1}\right|}$$

where $\left|J^{-1}\right|$ = determinant of the inverse Jacobian.

The elements of the Jacobian are expressed as

(Using the notation $\xi_x = \frac{\partial \xi}{\partial x}$, $\xi_y = \frac{\partial \xi}{\partial y}$, $x_\xi = \frac{\partial x}{\partial \xi}$ etc.)

$$\xi_x = \frac{y_\eta z_\zeta - y_\zeta z_\eta}{\left|J^{-1}\right|}$$

$$\xi_y = \frac{z_\eta x_\zeta - x_\zeta z_\eta}{\left|J^{-1}\right|}$$

$$\xi_z = \frac{x_\eta y_\zeta - y_\zeta x_\eta}{\left|J^{-1}\right|}$$

$$\eta_x = \frac{y_\zeta z_\xi - z_\zeta y_\xi}{\left|J^{-1}\right|}$$

$$\eta_y = \frac{z_\zeta x_\xi - x_\zeta z_\xi}{\left|J^{-1}\right|}$$

$$\eta_z = \frac{x_\zeta y_\xi - y_\zeta x_\xi}{\left|J^{-1}\right|}$$

$$\zeta_x = \frac{y_\xi z_\eta - z_\eta y_\xi}{\left|J^{-1}\right|}$$

$$\zeta_y = \frac{x_\xi z_\eta - z_\eta x_\xi}{\left|J^{-1}\right|}$$

$$\zeta_z = \frac{x_\xi y_\eta - y_\xi x_\eta}{\left|J^{-1}\right|}$$

$$\left|J^{-1}\right| = x_\xi\left(y_\eta z_\zeta - y_\zeta z_\eta\right) - x_\eta\left(y_\xi z_\zeta - y_\zeta z_\xi\right) - x_{\zeta\eta}\left(y_\xi z_\eta - y_\eta z_\eta\right)$$

Since the coordinate ζ is parallel to the z direction,

$$\xi_z = \eta_z = 0$$

The elements of the inverse Jacobian matrix are evaluated by the finite difference method. For the interior nodes a center differencing is used while for the boundary nodes either a forward or a backward differencing is used.

For 2D plane flow where the flow varies only in the x and z directions, coordinate transformation is performed by the ξ and ζ coordinates. Therefore

$$y_\eta = 1,\ x_\eta = z_\eta = 0$$

$$\text{So, } \eta_x = \eta_y = \eta_z = 0$$

For 2D depth averaged flow where the flow varies only in the x and y directions, coordinate transformation is performed by the ξ and η coordinates. Therefore

$$z_\zeta = 1,\ x_\zeta = y_\zeta = 0$$

$$\text{So, } \zeta_x = \zeta_y = \zeta_z = 0$$

APPENDIX B

MACCORMACK SCHEME

The DA equations in the generalized coordinate system can be written in the flux-vector format as:

$$\frac{1}{J}\frac{\partial Q}{\partial t}+\frac{\partial E}{\partial \xi}+\frac{\partial F}{\partial \eta}+H=0 \qquad \text{(B-1)}$$

$$Q=\begin{bmatrix} h \\ h\bar{u} \\ h\bar{v} \end{bmatrix}$$

$$E=\frac{1}{J}\begin{bmatrix} h\overline{U_c} \\ h\bar{u}\overline{U_c}+\xi_x\dfrac{gh^2}{2} \\ h\bar{v}\overline{U_c}+\xi_y\dfrac{gh^2}{2} \end{bmatrix}$$

$$F=\frac{1}{J}\begin{bmatrix} h\overline{V_c} \\ h\bar{u}\overline{V_c}+\eta_x\dfrac{gh^2}{2} \\ h\bar{v}\overline{V_c}+\eta_y\dfrac{gh^2}{2} \end{bmatrix}$$

$$H = \begin{bmatrix} 0 \\ \frac{1}{J}\left(\xi_x \frac{\partial h\overline{u'u'}}{\partial \xi} + \eta_x \frac{\partial h\overline{u'u'}}{\partial \eta}\right) + \frac{1}{J}\left(\xi_y \frac{\partial h\overline{u'v'}}{\partial \xi} + \eta_y \frac{\partial h\overline{u'v'}}{\partial \eta}\right) + \frac{1}{\rho J}\left(\xi_x \frac{\partial h\overline{p'}}{\partial \xi} + \eta_x \frac{\partial h\overline{p'}}{\partial \eta}\right) \\ + \frac{p_b'}{\rho J}\left(\xi_x \frac{\partial z_b}{\partial \xi} + \eta_x \frac{\partial z_b}{\partial \eta}\right) + gh\left(\xi_x \frac{\partial z_b}{\partial \xi} + \eta_x \frac{\partial z_b}{\partial \eta}\right) + \frac{1}{\rho J}\left(\tau_{dbx}' + \tau_{bx}'\right) \\ \frac{1}{J}\left(\xi_x \frac{\partial h\overline{u'v'}}{\partial \xi} + \eta_x \frac{\partial h\overline{u'v'}}{\partial \eta}\right) + \frac{1}{J}\left(\xi_y \frac{\partial h\overline{v'v'}}{\partial \xi} + \eta_y \frac{\partial h\overline{v'v'}}{\partial \eta}\right) + \frac{1}{\rho J}\left(\xi_y \frac{\partial h\overline{p'}}{\partial \xi} + \eta_y \frac{\partial h\overline{p'}}{\partial \eta}\right) \\ + \frac{p_b'}{\rho J}\left(\xi_y \frac{\partial z_b}{\partial \xi} + \eta_y \frac{\partial z_b}{\partial \eta}\right) + \frac{gh}{J}\left(\xi_y \frac{\partial z_b}{\partial \xi} + \eta_y \frac{\partial z_b}{\partial \eta}\right) + \frac{1}{\rho J}\left(\tau_{dby}' + \tau_{by}'\right) \end{bmatrix}$$

In a semi discrete form equation (B-1) can be written as

$$\frac{\partial Q_{i,j}}{\partial t} + R_{i,j} = 0$$

where indices *i* and *j* represent the location of any node.

$$R_{i,j} = \frac{\overline{E}_{i+1/2,j} - \overline{E}_{i-1/2,j}}{\Delta \xi} + \frac{\overline{F}_{i,j+1/2} - \overline{E}_{i,j-1/2}}{\Delta \eta} + H_{i,j}$$

MacCormack scheme is a two-step predictor corrector scheme.

Predictor:

$$Q_{i,j}^{*} = Q_{i,j}^{n} - \Delta t J R_{i,j}^{n}$$

where *n* indicates the values in previous time step.

In the predictor step the convective flux vectors at the cell interface are evaluated as

$$\overline{E}_{i+1/2,j} = E_{i+1,j} + D^{\xi}_{i+1/2,j} \text{ and } \overline{F}_{i+1/2,j} = F_{i+1,j} + D^{\eta}_{i+1/2,j} \text{ and so on.}$$

$D^{\xi}_{i+1/2,j}$ =artificial dissipation flux for numerical diffusion and explained later.

The derivatives of the source term H are evaluated by a forward difference.

Corrector:

$$Q^{**}_{i,j} = Q^{n}_{i,j} - \Delta t J R^{*}_{i,j}$$

In the corrector step the convective flux vectors at the cell interface are evaluated as

$$\overline{E}_{i+1/2,j} = E_{i,j} + D^{\xi}_{i+1/2,j} \text{ and } \overline{F}_{i+1/2,j} = F_{i,j} + D^{\eta}_{i+1/2,j} \text{ and so on.}$$

The derivatives of the source term H are evaluated by a forward difference.

The new values of the variables are calculated as the average of the values obtained in the predictor and corrector steps:

$$Q_{i,j}^{n+1} = \frac{Q_{i,j}^{*} + Q_{i,j}^{*}}{2}$$

Artificial dissipation flux modeling:

$$D_{i+1/2,j}^{\xi} = \frac{1}{J_{i,j}} \left[\left| \overline{U_c} \right| + c\sqrt{\xi_x^2 + \xi_y^2} \right]_{i+1/2,j} \left[-\varepsilon_{i+1/2}^{(2)}, j\left(Q_{i+1,j} - Q_{i,j}\right) \right]$$

$$D_{i,j+1/2}^{\eta} = \frac{1}{J_{i,j}} \left[\left| \overline{V_c} \right| + c\sqrt{\eta_x^2 + \eta_y^2} \right]_{i,j+1/2} \left[-\varepsilon_{i,j+1/2}^{(2)} \left(Q_{i,j+1} - Q_{i,j}\right) \right].$$

where $c = \sqrt{gh}$. $\varepsilon^{(2)}$=coefficient of second order artificial dissipation (Lackey and Sotiropoulos, 2005). In this thesis its value is taken as a constant.