DEVELOPMENT OF A 3-D LAMINAR NAVIER-STOKES SOLVER FOR SEPARATED CHANNEL FLOWS

A Thesis Submitted to
the Graduate School of Engineering and Sciences of
İzmir Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of

MASTER OF SCIENCE

in Mechanical Engineering

by
Umut CEYHAN

June 2011

İZMİR

We approve the thesis of **Umut CEYHAN**

Assist. Prof. Ünver ÖZKOL
Supervisor

Assoc. Prof. Moghtada MOBEDİ
Committee Member

Assoc. Prof. Şebnem ELÇİ
Committee Member

3 June 2011

Prof. Metin TANOĞLU
Head of the Department of
Mechanical Engineering

Prof. Durmuş Ali DEMİR
Dean of the Graduate School of
Engineering and Sciences

ACKNOWLEDGMENTS

The preparation of this study would not have been possible without the endless support and belief of my parent in me. I would very much like to thank them. I would also very much like to thank my supervisor Ünver Özkol for providing me with the chance of studying on this subject. He gave me the chance of creating my own pace and style while his advises and valuable comments led my vision. He has a great share on the success of this study. I thank to him since he created an ideal environment for my academic studies and researches with more than enough freedom. I also thank to all the other Mechanical Engineering professors who teach and support me. Finally, I am delighted and grateful to the Scientific and Technological Research Council of Turkey (TUBITAK) for their support throughout my Master's studies.

ABSTRACT

DEVELOPMENT OF A 3-D LAMINAR NAVIER-STOKES SOLVER FOR SEPARATED CHANNEL FLOWS

This study involves the development of a 3-D laminar Navier-Stokes solver with finite volume method in C++ language and investigation of 3-D separated channel flows. The missing parts of 3-D numerical implementation in Computational Fluid Dynamic books and articles are tried to be revealed within this study. To achieve these, the details of the discretization methods, implementation of boundary conditions and solution algorithm are explained. Besides, a more generalized form of the coefficients of the discretized momentum and pressure correction equations including boundary nodes are proposed. The use of artificial viscosity method achieves converting the conventional channel geometry into different channel geometries. Validation of the code is made investigating developing channel flow, artificial viscosity method and backward facing step flow. There exists an excellent agreement between present study and analytical results and experimental data. The simulation of a 3-D backward facing step is given in detail and the flow structure behind the step geometry is investigated. It was showed that complex three-dimensional flow develops behind the step with reverse and swirling flow regions. The "jet-like" flow and the impingement to the bottom wall are found to be responsible from the minimum on the reattachment line where the streamwise component of the wall shear stress is zero. The effect of channel expansion ratio on flow structure and pressure recovery is investigated and it is found that as expansion ratio increases, the reattachment line moves toward downstream of channel and the expansion loss coefficient increases.

ÖZET

DUVARDAN AYRILMIŞ KANAL AKIŞLARI İÇİN 3 BOYUTLU LAMİNER NAVIER-STOKES ÇÖZÜCÜSÜ GELİŞTİRİLMESİ

Bu çalışma 3 boyutlu laminar Navier-Stokes denklemleri çözücüsünün sonlu hacim metodu kullanılarak C++ dilinde geliştirilmesini ve 3 boyutlu duvardan ayrılan kanal akışlarının incelenmesini içermektedir. Bu çalışma ile hesaplamalı akışkanlar dinamiği kitaplarında ve ilgili makalelerdeki 3 boyutlu nümerik uygulamalardaki eksik kısımlar açığa çıkarılmaya çalışılmıştır. Bunun için, diskritizasyon metodlarının, sınır koşullarının uygulamasının ve çözüm algoritmasının detayları verilmiştir. Bunların yanında, diskritize edilmiş momentum ve basınç düzeltme denklemlerindeki katsayılar için, sınır koşullarını içeren, daha genel bir metod önerilmiştir. Yapay viskozite metodunun kullanımı, standart bir kanal geometrisini, farklı kanal geometrilerine çevirmeyi sağlamıştır. Kodun doğrulanması ise gelişmekte olan kanal akışı, yapay viskozite metodu ve geri basamak akışı incelenerek yapılmıştır. Şu anki çalışma ve deneysel veriler ile analitik sonuçlar arasında çok tutarlı sonuçlar görülmüştür. 3 boyutlu geri basamak akışının simülasyanu ayrıntılı olarak verilmiş ve basamağın arkasındaki karmaşık akış yapısı incelenmiştir. Basamak arkasında 3 boyutlu geri dönen ve kıvrılan akış yapıları görülmüştür. Akış içinde görülen jet tarzı ve duvara çarpan yapı, kanal alt duvarında akış yönündeki kayma geriliminin sıfır olduğu noktaların oluşturduğu, akışın duvara tekrardan birleşme çizgisinde görülen minimum noktanın sebebi olarak yorumlanmıştır. Kanal genişleme oranının akış yapısına ve basınç geri kazanımına olan etkisi incelenmiş ve genişleme oranı arttıkça akışın tekrardan birleşme çizgisinin akıntı yönünde ilerlediği ve genişleme kayıp katsayısının da arttığı görülmüştür.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

a	Momentum coefficient
ap	Pressure correction coefficient
A	Surface area, m^2
A_c	Cross-sectional area, m^2
AR	Aspect ratio
b	Momentum source term
bp	Pressure correction source term
CFD	Computational fluid dynamics
C_p	Pressure recovery coefficient
C. V.	Control volume
D_0	Substantial derivative
D_i	Diffusion conductance flux, kg/s
D_h	Hydraulic diameter, m
ER	Expansion ratio
f_e	Interpolation factor
f_{darcy}	Darcy friction factor
$f_{fanning}$	Fanning friction factor
$F_{i,b}$	Body force, N
F_i	Convective mass flux, kg/s
H	Upstream height of channel for backward facing step flow, m
g	Gravitational acceleration, m/s^2
K_e	Expansion loss coefficient
L	W/2, half width of the backward facing step channel, m
L_d	Length of the downstream of step, m
L_{dc}	Developing channel length, m
L_s	Length of step, m
LDA	Laser Doppler Anemometer
M	Mass, kg
MR	Material region
n	Normal direction

NS	Numerical simulation
NSS3D	Navier-Stokes Solver 3D
N_x	Number of control volumes in x-direction
N_y	Number of control volumes in y-direction
N_z	Number of control volumes in z-direction
O	Outlet plane
p	Pressure, Pa
P	Cell Peclet number or Pressure
P_r	Related pressure for neighbors, Pa
r	Expansion factor
R	Residual
Re	Reynolds number
R_i	Surface force, N
s	Step height, m
S	Half height of the channel, m
SP	Symmetry plane
T	Time, sec.
T_{ij}	Stress tensor
u	Spanwise velocity, m/s
y*	y/h
v	Transverse velocity, m/s
V	Volume, m^3
w	Streamwise velocity, m/s
w_0	Average axial velocity, m/s
W	Width of the the backward facing step channel, m
w*	w/w_0
z*	z/L_{dc}

Greek Symbols

a	Under-relaxation factor
Γ	Diffusion coefficient
d_{ij}	Kronecker delta function
Δ	Difference

e_{ij}	Strain rate tensor
l	Bulk viscosity, kg/ms
m	Dynamic viscosity, kg/ms
f	Dependent variable
ρ	Density, kg/m^3
t	Shear stress, Pa
t_{ij}	Shear stress tensor

Subscripts

b	Back side face of the control volume
B	Back nodal point
e	East side face of the control volume
E	East nodal point
f	Front side face of the control volume
F	Front nodal point
n	North side face of the control volume
N	North nodal point
P	General nodal point
s	South side face of the control volume
S	South nodal point
w	West side face of the control volume
W	West nodal point
0	Initial or average value

Superscripts

-	Average value
‘	Correction value
*	Guess value
i	ith item or index notation

CHAPTER 1

INTRODUCTION

The science has devoted itself to understand the nature and, sometimes, to control it which are possible with deep researches and investigations. Numerical simulation is one of the tools to accomplish the aim of science and it is a practical and viable alternative to experimental and theoretical tools with the aid of increasing computer power. With this increase, more complex models can be added for the problem or more discretization points to the simulation to increase the accuracy.

Numerical simulation avoids using complicated experimental setups, and the theoretical tools are limited to solve some problems in closed form. Nevertheless, they are dependent on each other, i.e., good numerical simulations require good physical and mathematical models describing the problem which are based on the experiments and some theoretical background. The most reliable information about a physical phenomenon is obtained with experimentation and therefore numerical solutions should be validated against the results of the experiments. However, it should be noted that some experiments are extremely expensive and sometimes impossible to obtain complete information. Besides these, the measuring equipments are not free from errors as well. It is also difficult or even sometimes impossible to obtain all the needed information by experiments. If the mathematical description is adequate then a numerical approach may be superior to experiment but if the mathematical description has not completed yet, there is an uncertainty about the extent to which the computed results would agree with reality. It is doubtless that for investigation of a new basic phenomenon, experiment leads and computation follows. The numerical simulation is the intersection of the physical and mathematical models, numerical methods and computer science (Figure 1.1).

Currently numerical simulations are utilized in several scientific, engineering and industrial areas such as the analysis of stability in mechanical structures, optimization of chemical reactions and combustion processes, simulation of 3-D DNA structures, meteorological prediction and design of engineering devices involving fluid

flow and/or heat transfer phenomenon as in stated in the study of Kaufmann *et al*. (1993).

The aim of this study is to develop a three-dimensional Navier-Stokes flow solver written in C++ language and investigate the separated flows in a channel. The main objective is to reveal all the details of a flow solver which are missing in the literature and commercial codes. Commercial codes are easy to use programs, however they can be called as "black boxes" requiring inputs and giving outputs. Development of a homemade solver enables us to understand the basics of it as being inside the "black box". It gives the control to the developer and after that it is easier to add new steps to the code. To work in new application areas requires a good knowledge of the fundamentals and the aim of this study is to reveal these fundamentals in computational fluid dynamics area.

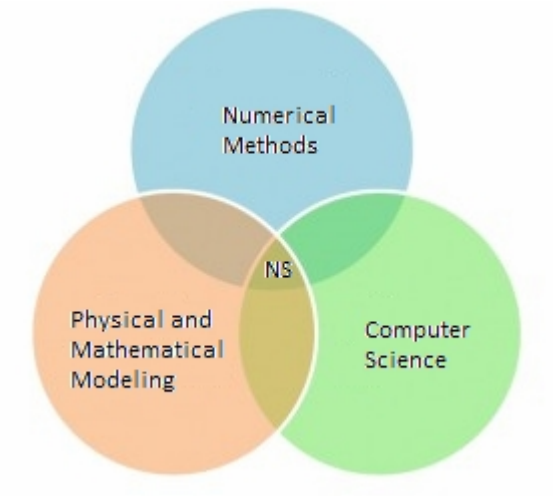

Figure 1.1. The location of numerical simulation (NS) in science.

The beginning of twentieth century is the time when the numerical methods such as finite difference to solve ordinary and partial differential equations had been developed. But the interest in computational study has increased in 1960s with the power of high-speed computers.

The beginning of computational fluid dynamics (CFD), is attributed to the first definitive work of Richardson (1910) who introduced the point iterative schemes to solve Laplace's Equation and biharmonic equation numerically. Some sources show the

beginning of numerical analysis as the paper of Courant, Friedrichs and Lewy (1928) in which the questions of “uniqueness” and “existence” were addressed for the numerical solution of partial differential equations.

After World War II, the number of researches on numerical solution of fluid dynamics problems has increased. Professor Jhon von Neumann worked on the stability of numerical methods for solving time marching problems and he introduced artificial viscosity scheme with Richtemyer (1950). In 1970s and 1980s, different discretization schemes are introduced. Not only the techniques to discretize the equations had been developed, but also the solution algorithms to solve the resulting equations had been introduced and the studies of Patankar (1980) are very important in the solution algorithm literature. The studies of J.C. Tannehill, R. W. MacCormack, P. J. Roache, T. J. Chung, A. J. Baker, D. A. Anderson and K. A. Hoffman have added many tools in to the Computational Fluid Dynamics area.

Since 1960, the Navier-Stokes Equations have been solved numerically and today there are numerous solvers dealing with different types of flows. They are both commercial and open source codes. The well known commercial codes are FLUENT, STAR-CD, ANSYS CFX. They can handle very complicated problems. The problem with these codes is the difficulty to understand the code and implement the needed subroutines. There exist also open-source codes that can handle Navier-Stokes or similar type equations. They are developed by the users and the most common one is OPEN-FOAM in which you should use their style to add new things and it is difficult to understand the code also the number of the open-source codes are very limited. However, a code written for academic purposes is more valuable than using commercial software or a code written by someone else. One of the powerful academic study purposed flow solvers is the “Navier-Stokes Stanford University Solver (NSSUS)”. It solves the three dimensional Unsteady Reynolds Averaged Navier-Stokes (URANS) type equations.

Problem tackled in this study is channel flows with separation and recirculation regions. The nature of the separation and the consequent flow recirculation are problems of the most fundamental interest in the fluid dynamics. The flow over obstacles, fences, steps have been investigated both numerically and experimentally since these problems are the basic configurations of the separation and recirculation. Although the geometry boundaries are simple, the flows are rather complex. As the free stream passes over the obstacle, the flow separates at the corner and then reattaches at the bottom-wall

downstream. The flow reverses behind the obstacle and a boundary layer redevelops downstream of the reattachment. A secondary reversed flow on the roof wall occurs at some low or moderate Reynolds numbers as can be seen in Figure 1.2. The deviation exists between the two-dimensional numerical simulations and the experimental data in some low-Reynolds-number flows, because of the side wall effects. This makes the flow three dimensional (3-D).

The investigation of separated and reattached flows is very important. They occur in many heat exchanging devices such as electronic and power generating equipments, in pieces of electronic cooling equipment, cooling of nuclear reactors, cooling of turbine blades, flow in vertical plates with ribs, flow in wide angle diffusers and valves. The mixing of high and low energy fluids occur mostly in the reattached flow region therefore improving the heat transfer performances of these devises. The separation phenomenon is also used for understanding of onset of transition to turbulence.

For several decades, separated flows are under investigation and the most popular one is backward facing step geometry. Figure 1.2 shows the step geometry, separation region, primary recirculation bubble, reattachment point and recirculation attached to the roof. Since the reattachment length is given as a global feature of this flow, it is used for both validation and comparison of the numerical codes. Most of the published work is for 2-D and 3-D flow simulations are rare both because of its complexity and high computational cost.

A popular study for the backward facing step geometry was done by Armaly *et al.* (1983). This study based on both experimental and theoretical investigations and they reported the velocity measurements and reattachment points for laminar flows in channels utilizing Laser-Doppler (LDA) measurement technique up to Reynolds number of 8000. They showed that a secondary recirculation zone attached to the roof (Figure 1.2) occurs beyond the Re>400. Armaly *et al.* (2003) published their measurements in three-dimensional laminar separated flow revealing a swirling "jet-like" flow near the side wall in the separating shear layer and the impingement of flow on the stepped wall. The measurement in that study covered a range of Reynolds number between 98.5 and 525. Nie and Armaly (2003) presented numerical simulations for incompressible three-dimensional laminar forced convection flow adjacent to backward facing step in rectangular duct to examine the reattachment region of the separated flow on the stepped wall. The effects of step height for the same flow is

studied by Nie and Armaly (2002) showing that the size of the primary recirculation region and the maximum on the Nusselt number distribution near the side wall increased as step height increased for Reynolds number of 343.

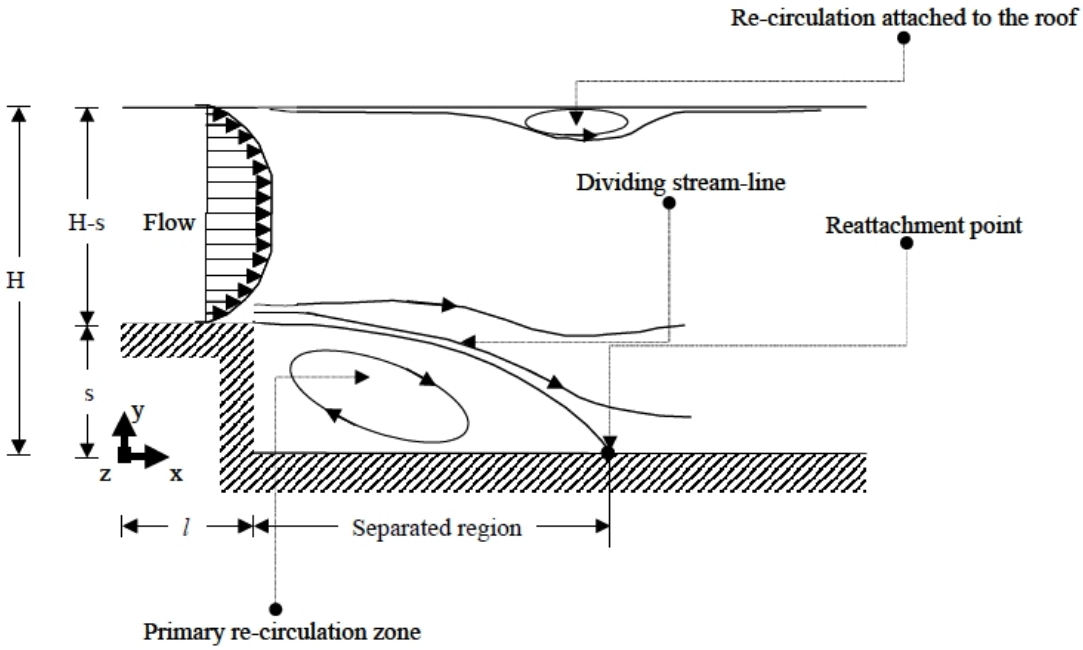

Figure 1.2. Flow over a backward facing step
(Source: Saldana, 2005).

Kim and Moin (1985) computed the flow over a backward-facing step showing a relation between the reattachment length and Reynolds number and they compared their numerical results with the experimental data of Armaly *et al.* (1983) in a good agreement.

Gartling (1990) solved steady and incompressible backward facing step flow as a test problem for outflow boundary condition. He developed a solution procedure utilizing Galerkin-based finite element method and compared his results with the numerical results of Kim and Moin (1985).

Williams and Baker (1997) simulated laminar flow over a 3D backward facing step. The main interest of this steady was the characterization of three-dimensional vortices in the primary separation region for $100 \leq Re \leq 800$. They showed that the side walls are responsible for the creation of wall-jet pointing from the side-wall to the center of the channel.

Chiang and Sheu (1999) made a numerical revisit of backward facing step flow. They solved steady-laminar Navier Stokes equations for a channel geometry and flow conditions reported by Armaly *et al.* (1983) in the range of $100 \leq Re \leq 1000$. They revealed that the flow at the symmetry plane developed into two-dimensional for Re=800 and the channel width is 100 times of the upstream step-height. They also presented the topological features of the flow using critical point theory.

Vogel and Eaton (1985) worked experimentally on the backward step geometry. They measured combined heat transfer and fluid dynamics in a separated and reattaching boundary layer with emphasis on the near wall region for $13000 \leq Re \leq 42000$. They showed that reattachment increases the heat transfer coefficient by a factor of two and they obtained the maximum heat transfer coefficient slightly upstream of reattachment where the highest turbulence intensity was measured. Adams and Eaton (1988) published their LDA study of the backward facing step including the effects of velocity bias. The aim of the study was providing structural information and test data for modelers. Fessler and Eaton (1997) investigated the flow field in a vertically downward, single sided sudden expansion with a fully developed channel flow inlet of Reynolds number 13800. They used dense particles to measure their response via LDA to turbulent flow field after the sudden expansion.

Le and Moin (1997) performed direct numerical simulations of turbulent flow over a backward facing step at a Reynolds number of 5100. They showed the strong stream-wise vorticity structures and gave the instantaneous velocity field to reveal the variation of oscillating reattachment length.

Saldana (2005) worked on the numerical simulation of mixed convection over a three-dimensional horizontal backward facing step. He investigated the effects of buoyancy force to the velocity field and temperature distribution. He used the velocity profiles of a developing channel flow to validate its code against the data given in the study of Shah and London (1978).

CHAPTER 2

MATHEMATICAL MODEL DESCRIPTION

The equations governing the incompressible Newtonian fluid flow are conservation of mass, also known as continuity and conservation of momentum. The equations contain mechanical pressure and three velocity components to be determined from these set of equations.

2.1. Conservation of Mass

Conservation of mass is called as continuity equation due to the continuum assumptions, i.e., density and velocity may be defined at every point in space. The basic idea behind this law is that the mass of a material region is constant, in other words the time rate of change of mass of a material region is zero as given in Eq. (2.1).

$$\frac{dM_{MR}}{dt} = \frac{d}{dt}\int_{MR} r\,dV = 0 \tag{2.1}$$

The surface of the material region (Figure 2.1) moves with the local fluid velocity v_i . The time differentiation at right hand side of Eq. (2.1) can be moved into the integration by using Leibnitz's theorem, i.e., the derivative with respect to time can be moved inside the integral if a surface integral is added to compensate for the motion of the boundary moving with velocity w which is equal to v.

$$\int_{MR} \partial_0 r\,dV + \int_{MR} n_i v_i r\,dS = 0 \tag{2.2}$$

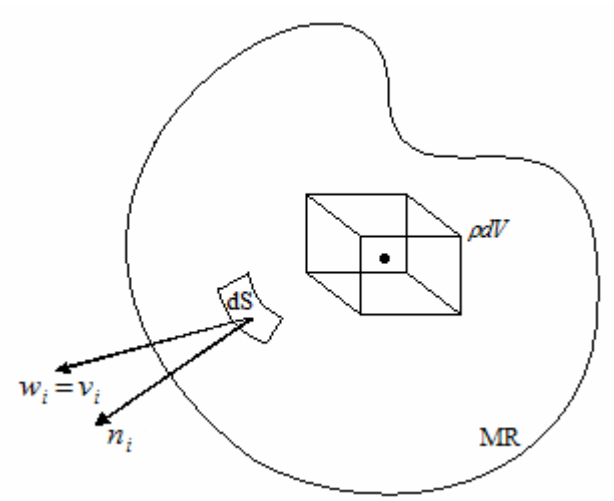

Figure 2.1. Material region (MR) and local fluid.

The second term in Eq. (2.2) is a surface integral and it can be converted into volume integral applying the Gauss's theorem (Eq. (3.3)) and the sum of the two integrals gives Eq. (2.3).

$$\int_{MR} [\partial_0 r + \partial_i (r v_i)] dV = 0 \tag{2.3}$$

Since the choice of the integration region is arbitrary, Eq. (2.3) is zero if the integrand is zero. Then the differential form of the conservation of mass can be written in index notation as follow:

$$\partial_0 r + \partial_i (r n_i) = 0 \tag{2.4}$$

The first term in Eq. (2.4) is the rate of change of mass per unit volume at a fixed point in space. The second term includes three parts in it; these are the net mass flow rates out of the element in x, y and z directions. They are also called as the convective terms.

If the flow is steady, the first term in Eq. (2.4) drops out, and then the equation turns into the following form.

$$\partial_i \left(r n_i \right) = 0 \tag{2.5}$$

If the density is constant (incompressible flow) in addition to steady flow, Eq. (2.4) turns into even simpler form of velocity solenoidality.

$$\partial_i (n_i) = 0 \tag{2.6}$$

2.2. Conservation of Momentum: The Navier-Stokes Equations

The conservation of momentum principle is that the time rate of change of the linear momentum of a material region is equal to the sum of the forces on the region. Newton's second law, which relates the acceleration of a particle to the applied force, is a special case of the conservation of momentum principle. The force can be stated as the sum of two types of forces which are body and surface forces. The body forces apply to the entire mass (bulk) of the material and they arise usually from external fields such as gravity, electromagnetic and magnetohydrodynamic effects. On the other hand, the surface forces act on the boundary surface and simulate the effect of the outside fluid upon the inside fluid as an actual surface force. In other words, the effect of the outside fluid is replaced with a surface force acting on the boundary of the inside fluid.

If $F_{i,b}$ represents the body force per unit mass and R_i represents the surface force per unit area on the bounding surface of material region (MR), then the sum of two forces on the region gives the total force as given in Eq. (2.7).

$$\int_{MR} r F_{i,b} dV + \int_{MR} R_i dS = Net\ Force\ on\ a\ material\ region \tag{2.7}$$

Since the net force on a material region is equal the time rate of chance of momentum of the region, then the conservation of momentum principle can be written as given in Eq. (2.8).

$$\frac{d}{dt} \int_{MR} r v_i dV = \int_{MR} r F_{i,b} dV + \int_{MR} R_i dS \tag{2.8}$$

The time differentiation on the left-hand side of the Eq. (2.8) can be moved into the integral by Leibnitz's and Gauss' theorem. The result is given in Eq. (2.9).

$$\int \left[\partial_0 \left(r v_i\right) + \partial_j \left(r v_j v_i\right)\right] dV = \int r F_{i,b} dV + \int R_i dS \tag{2.9}$$

The surface forces are the external stresses on the sides of the fluid element. The stress is defined in terms of pressure which creates a normal stress and nine viscous stress components for a three dimensional fluid element. The stress tensor T_{ij} can be broken into two parts by subtracting out the thermodynamic pressure, p_t, as given in Eq. (2.10).

$$T_{ij} = -p_t d_{ij} + t_{ij} \tag{2.10}$$

The term d_{ij} in Eq. (2.10) is the Kronecker delta function and viscous stresses are denoted by the stress tensor t_{ij}. The stress tensor is given as

$$t_{ij} = \begin{pmatrix} t_{xx} & t_{xy} & t_{xz} \\ t_{yx} & t_{yy} & t_{yz} \\ t_{zx} & t_{zy} & t_{zz} \end{pmatrix} \tag{2.11}$$

The rows represent the forces in each direction. Taking the divergence of the row vectors, (t_{xx}, t_{xy}, t_{xz}), (t_{yx}, t_{yy}, t_{yz}) and (t_{zx}, t_{zy}, t_{zz}) gives the surface force vector in x, y and z directions. Then the term representing the surface forces in Eq. (2.9) can be written in terms of pressure and viscous stress tensor.

$$\int R_i dS = \int n_i T_{ij} dS = \int n_i \left(-p_t d_{ij} + t_{ij}\right) dS \tag{2.12}$$

After converting the surface integral in Eq. (2.12) into control volume integral using Gauss' theorem and equating the thermodynamic pressure to mechanical pressure using the Stoke's assumption that the average normal stress is zero, Eq. (2.9) can be

written as in Eq. (2.13) and collecting the terms under the same control volume integration results with Eq. (2.14).

$$\int\left[\partial_0\left(rv_i\right)+\partial_j\left(rv_jv_i\right)\right]dV=\int rF_{i,b}dV+\int\left(-\partial_i p+\partial_j t_{ji}\right)dV \tag{2.13}$$

$$\int\left[\partial_0\left(rv_i\right)+\partial_j\left(rv_jv_i\right)-rF_{i,b}+\partial_i p-\partial_j t_{ji}\right]dV=0 \tag{2.14}$$

Since the choice of the integration region is arbitrary, Eq. (2.14) is zero if the integrand is zero. Considering the body force as just gravitational, the conservation of momentum can be written as in Eq. (2.15).

$$rD_0n_i=rg_i+\partial_j T_{ij} \tag{2.15}$$

In Eq. (2.15), the term D_0 is called the total derivative which signifies the Lagrangian acceleration of a material particle. Stress tensor T_{ij} should be expressed in terms of velocity and this expression depends on some physical models. This study deals with the Newtonian fluids for which the viscous stresses are linearly proportional to the rates of deformation. Stokes' assumptions set in at this point for all gases and most common fluids:

i. The fluid is continuous and the stress tensor T_{ij} is at most a linear function of the strain rate tensor e_{ij}.

ii. The fluid has isotropic properties, i.e., its properties are independent of direction; therefore the deformation law is independent of the coordinate axes in which it is expressed.

iii. If the strain rates are zero, the deformation law must reduce to the hydrostatic pressure condition, $T_{ij}=-pd_{ij}$.

Considering these assumptions, the general form of the deformation law for a Newtonian fluid can be written as follow.

$$T_{ij}=-pd_{ij}+m\left(\frac{\partial n_i}{\partial x_j}+\frac{\partial n_j}{\partial x_i}\right)+d_{ij}\,l\,\partial_i n_i \tag{2.16}$$

In Eq. (2.16), d_{ij} is the Kronecker delta function, m is the first viscosity which is also called as dynamic viscosity and l is the second viscosity which is also known as bulk viscosity. The detailed derivations for the stress tensor components, strain rate tensor components and the relation between first and second viscosities can be found in books of Panton (1984) and White (2006).

The momentum equation for a Newtonian fluid can be obtained by substituting Eq. (2.16) into Eq. (2.15). The outcome is the famous Navier-Stokes Equations which are given in the following set of equations.

$$r\frac{Du}{Dt} = rg_x - \frac{\partial p}{\partial x} + \frac{\partial}{\partial x}\left(2m\frac{\partial u}{\partial x} + l\,\vec{\nabla}\cdot\vec{V}\right) + \frac{\partial}{\partial y}\left[m\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)\right] + \frac{\partial}{\partial z}\left[m\left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}\right)\right] \tag{2.17a}$$

$$r\frac{Dv}{Dt} = rg_y - \frac{\partial p}{\partial y} + \frac{\partial}{\partial y}\left(2m\frac{\partial v}{\partial y} + l\,\vec{\nabla}\cdot\vec{V}\right) + \frac{\partial}{\partial x}\left[m\left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right)\right] + \frac{\partial}{\partial z}\left[m\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right)\right] \tag{2.17b}$$

$$r\frac{Dw}{Dt} = rg_z - \frac{\partial p}{\partial z} + \frac{\partial}{\partial z}\left(2m\frac{\partial w}{\partial z} + l\,\vec{\nabla}\cdot\vec{V}\right) + \frac{\partial}{\partial x}\left[m\left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}\right)\right] + \frac{\partial}{\partial y}\left[m\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right)\right] \tag{2.17c}$$

Rewriting the equation set (2.18) in index notation;

$$rD_0 n_i = rg_i + \partial_i p + \partial_j[m(\partial_j n_i + \partial_i n_j) + d_{ij}\, l\, \partial_i n_i] \tag{2.18}$$

If the fluid is incompressible, the term $\partial_i n_i$ vanishes in Eq. (2.18) due to the continuity equation given in Eq. (2.6). If the dynamic viscosity is constant too, then the equation takes a simple form of the Navier-Stokes equation for constant viscosity and density as in Eq. (2.19).

$$rD_0 n_i = rg_i + \partial_i p + m\partial_j \left(\partial_j n_i + \partial_i n_j\right) \tag{2.19}$$

This study uses the following assumptions for the mathematical model described.

i. Steady state

ii. Laminar and incompressible flow

iii. Constant property, i.e., constant density and constant viscosity

iv. No gravitational effect

v. No viscous dissipation (heating)

With these assumptions the continuity and momentum equations take the following forms within their symbolic notation.

Continuity equation:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \tag{2.20}$$

Momentum equations:

$$\frac{\partial}{\partial x}(ruu) + \frac{\partial}{\partial y}(rvu) + \frac{\partial}{\partial z}(rwu) = m\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right) - \frac{\partial P}{\partial x} \tag{2.21a}$$

$$\frac{\partial}{\partial x}(ruv) + \frac{\partial}{\partial y}(rvv) + \frac{\partial}{\partial z}(rwv) = m\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}\right) - \frac{\partial P}{\partial y} \tag{2.21b}$$

$$\frac{\partial}{\partial x}(ruw) + \frac{\partial}{\partial y}(rvw) + \frac{\partial}{\partial z}(rww) = m\left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2}\right) - \frac{\partial P}{\partial z} \tag{2.21c}$$

Even with the assumptions given above, the Eq. (2.20) and (2.21) are not easy to solve because the momentum equations involve nonlinear terms, they are coupled within itself and with continuity equation and there is a pressure term in momentum equations, which has not a separate equation. Therefore, numerical approximation is the only way to reach the solution in most of the applications.

CHAPTER 3

NUMERICAL IMPLEMENTATION

3.1. Introduction

The mathematical description of a physical problem generally is a set of partial differential equations which are nonlinear in nature in case of flow problems. The nonlinearity of the governing equations makes impossible to obtain an analytical solution. Fortunately numerical methods promise that the implications of the mathematical model can be worked out for almost any practical problem. With the numerical methods, what is solved is not the differential equations directly, but the algebraic equations derived from the discretized form of these equations and this makes the numerical methods so powerful and applicable. Numerical method treats its basic unknowns as the values of the dependent variable at a finite number of locations in the computational domain. The obtained algebraic linear system of equations is solved to find the values at those points with a proper solution algorithm.

3.2. The General Transport Equation

The steady form of the general transport equation is given in Eq. (3.1). In this equation, ϕ represents the dependent variable. If ϕ takes the value of 1, it turns out to be the conservation of mass equation and if it takes the value of u, v, w, then it represents the x, y and z direction momentum equations respectively. In Eq. (3.1), the left hand side includes the convective terms, and the right hand side includes the diffusive and source terms. The diffusion coefficient Γ and the source term S vary depending on the meaning of the dependent variable ϕ.

$$\frac{\partial}{\partial x_i}\left(r\, f\, u_i\right) = \frac{\partial}{\partial x_i}\left(\Gamma_f \frac{\partial}{\partial x_i} f\right) + S_f \tag{3.1}$$

The dependent variables are calculated at the chosen grid points by solving the linear algebraic equations derived from Eq. (3.1). In this derivation, the variation of dependent variable ϕ among the grid points should be employed. This variation is generally calculated including a small region in terms of ϕ values at the grid points within and around the region. To achieve this, the calculation domain is subdivided into a number of sub-domains or elements. There are many methods employed for the division of the domain into sub-domains and regions. The most well known methods are finite difference method, finite element method, method of weighted residuals and finite volume method.

3.3. The Finite Volume Method

Finite volume method is one of the discretization methods used to solve the partial differential equations. The basic idea behind the finite volume method is to divide the computational domain into a number of non-overlapping control volumes. The computational domain (e.g. channel volume) and a few exaggerated sub-control volumes are shown in Figure 3.1. The sub-control volumes are surrounded by their neighboring control volumes.

The finite volume method leads itself to direct physical interpretation. The differential equation is integrated over all the sub-control volumes and the piecewise profiles expressing the variation of ϕ between the grid points are used to evaluate the required integrals. The result is the discretized equation containing the values of ϕ for a group of grid points.

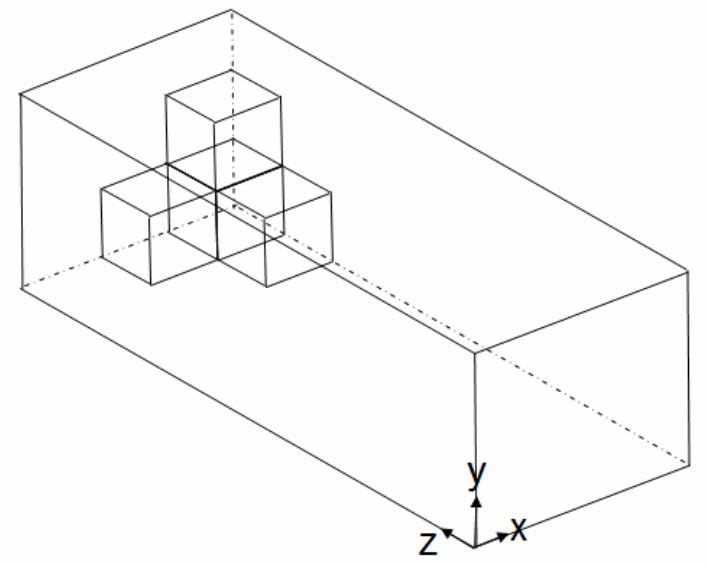

Figure 3.1. The basic control volume division of the computational domain.

Since the differential equation is integrated over the control volumes, the integral for the sub-control volumes are conservative, i.e., the integral conservation of mass, momentum and energy are exactly satisfied over any group of control volume and over the whole calculation domain. There is an exact balance among the control volume and its neighbors. Finite volume methods, due to this property, are called conservative in their nature.

The schematic representation of one of the control volumes is given in Figure 3.2 showing its neighbors. The capital letters show the centers of the neighboring control volumes and the small letters represent the locations of the faces of the control volumes. As shown in Figure 3.2, **w** represents the west face of the control volume represented by the P nodal point. A three dimensional control volume has six neighboring control volumes which are west (W), east (E), north (N), south (S), front (F), back (B) control volumes, and it has six control volume faces which are west (w), east (e), north (n), south (s), front (f) and back (b) faces.

The key step of the finite volume method is the integration of Eq. (3.1) over a three-dimensional control volume. Taking the integral of both sides of the Eq.(3.1) results with Eq. (3.2). This procedure is applied to all the nodal points in the domain.

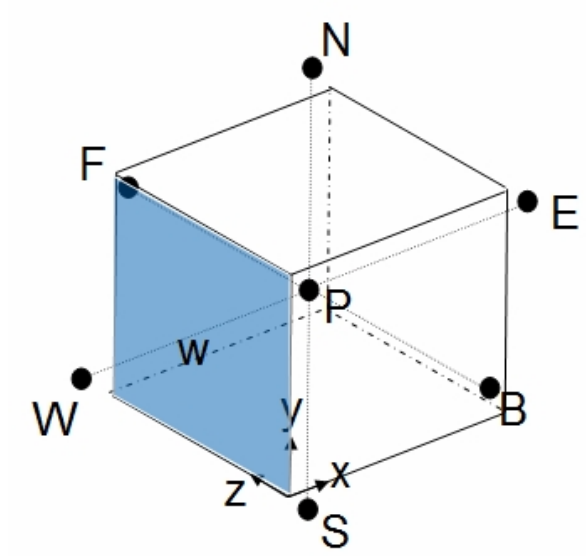

Figure 3.2. Schematic view of a control volume with neighboring points and faces.

$$\int_{C.V} \frac{\partial}{\partial x_i}(r\, f\, u_i) = \int_{C.V} \frac{\partial}{\partial x_i}\left(\Gamma_f \frac{\partial}{\partial x_i} f\right) + \int_{C.V} S_f\, dV \tag{3.2}$$

The divergence theorem of Gauss states that the volume integrals can be transformed into surface integrals over the entire bounding surface of the control volume. This theorem can be written for any vector x as;

$$\int_{C.V} div(\vec{x})dV = \int_{A} \vec{n} \bullet \vec{x} dA \tag{3.3}$$

Equation (3.3) has a direct physical interpretation. The term $\vec{n} \bullet \vec{x}$ is the component of $\vec{x}$ in the direction of the vector $\vec{n}$ normal to surface element dA. The integral of the divergence of a vector over a volume is equal to the component of that vector in the direction normal to the surface bounding the volume integrated over the whole surface A. With this theorem, the order of the derivative is reduced by one and this makes the numerical solution procedure easier. Applying this method to the volume integrals of Eq. (3.2) turns them into surface integrals as in Eq. (3.4).

$$\int_A n_i \bullet (r\, f\, u_i) = \int_A n_i \bullet \left(\Gamma_f \frac{\partial}{\partial x_i} f \right) + \int_{C.V} S_f \, dV \tag{3.4}$$

The term on the left hand side of Eq. (3.4) is the net rate of change of ϕ in the fluid element due to convection and the first term on the right hand side of the Eq. (3.4) is the net rate of change of ϕ in the fluid element due to diffusion. The last term on the Eq. (3.4) is the net rate of creation of ϕ inside the control volume due to sources.

The integration in Eq. (3.4) yields the variation of ϕ in the control volume due to interaction with its neighboring nodal points, the convective and diffusive flux across the control faces and the change in the source term as given in Eq. (3.5). It is a statement of the conservation of a fluid property for a finite size control volume.

$$\begin{aligned} &\left[ruf - \Gamma_f \frac{\partial f}{\partial x} \right]_e dzdy - \left[ruf - \Gamma_f \frac{\partial f}{\partial x} \right]_w dzdy \\ &+ \left[rvf - \Gamma_f \frac{\partial f}{\partial y} \right]_n dxdz - \left[rvf - \Gamma_f \frac{\partial f}{\partial y} \right]_s dxdz \\ &+ \left[rwf - \Gamma_f \frac{\partial f}{\partial z} \right]_f dxdy - \left[rwf - \Gamma_f \frac{\partial f}{\partial z} \right]_b dxdy = \bar{S}_f \, dxdydz \end{aligned} \tag{3.5}$$

The effect of the combined convective and diffusive fluxes is given in the square brackets in Eq. (3.5). The values of the dependent variables are calculated at the nodal points, however in Eq. (3.5), the values are for the control volume faces and therefore they should be evaluated at the faces utilizing the value of the neighboring nodal points.

Evaluation of the values of the dependent variable ϕ is very important in Finite Volume Method. There exist several methods to evaluate the values at the faces and these methods are called discretization schemes. The simplest one is approximating the values at the faces assuming linear changes. One of the simple forms is known as central differencing scheme and it evaluates the value at the face as the average of the two neighboring nodal points' values. This procedure can be applied to both convection and gradient terms. The other well-known yet slightly more complicated schemes are upwind, hybrid, power law and exponential (exact) schemes which are compared in later sections.

To make the calculations easier, two new variables F and D are introduced here. F represents the convective mass flux and D represents the diffusion conductance flux at the cell faces. The “i” refers to the corresponding face in Eq. (3.6) and Eq. (3.7).

$$F_i = ru_i \Delta A \tag{3.6}$$

$$D_i = \frac{\Gamma_f}{\Delta x_i} \Delta A \tag{3.7}$$

The values of the dependent variable at different faces are written in terms of the neighboring nodal points with the chosen discretization scheme. Then Eq. (3.5) includes only the values of the dependent variable ϕ at the nodal points, i.e., ϕ_E , ϕ_W, ϕ_N, ϕ_S, ϕ_F, ϕ_B and ϕ_P. With the aid of discretization scheme, the Eq. (3.5) turns into an equation including ϕ at the nodal points and the new variables F and D. The resulting equation can be combined into common parenthesis of the dependent variables at the nodes with their related coefficient in the parenthesis. Then the general form of the nodal equation is written in the form of Eq.(3.8).

$$a_P f_P = a_E f_E + a_W f_W + a_N f_N + a_S f_S + a_F f_F + a_B f_B + b \tag{3.8}$$

The sub index “P” refers to the nodal point whose value calculated according to the values of the neighboring variables and source term. The last term in Eq. (3.8) is the source term where

$$b = \bar{S}_f dxdydz \tag{3.9}$$

In practical situations, the source term may be a function of dependent variable ϕ. In the finite volume method, the source term is approximated by means of linear form.

$$\bar{S} dxdydz = S_u + S_p f_P \tag{3.10}$$

The coefficients of the dependent variables are related with the discretization scheme applied to the differential equation. The general formula for these coefficients can be written as

$$a_i = D_i A(|P_i|) + [|\pm F_i, 0|] \tag{3.11}$$

where P is the cell Peclet number which is the ratio of F to D which gives the strength of convection over diffusion. The Function A is determined according to the discretization scheme chosen. Table 3.1 gives the expressions of this function with respect to various schemes.

The derivation of Eq. (3.11) can be found in the work of Patankar (1980). Figure 3.3 compares the consequences of various schemes for a 1-D convection diffusion problem. The values of ϕ_P is plotted for given values of ϕ_E=1 and ϕ_W=0 by different schemes for various values of the cell Peclet number P.

Table 3.1. The function A(|P|) for different schemes
(Source: Patankar, 1980).

Scheme	Formula for A(|P|)
Central difference	1-0.5|P|
Upwind	1
Hybrid	[| 0, 1-0.5|P| |]
Power law	[| 0, $(1-0.1\|P\|)^5$ |]
Exponential (exact)	|P|/[exp(|P|)-1]

The results of the schemes suggest that all the schemes except central differencing scheme gives physically meaningful results. As can be deduced from Fig. 3.3, the results of the power law scheme and the exact solution are too close to one another, therefore, in this study, power law discretization scheme is used for the discretization of convective terms. It should be noted that, the central differencing scheme can be used in the solutions by refining the grid dimension. This refinement reduces the cell Peclet number (P) below two in which range the central differencing scheme is stable. However, this procedure requires use of too much grid points in the computational domain which is not demanded for the practical applications. The

physically meaningful results with finite number of grid points depend on some properties such as conservativeness, boundedness and transportiveness. The definitions of these properties can be found in the book of Malalasekera (2007). It should be noted that the power law scheme is fully conservative, unconditionally bounded and it satisfies the transportiveness requirement. One disadvantage of the scheme is that the accuracy in terms of Taylor series truncation error is only first order.

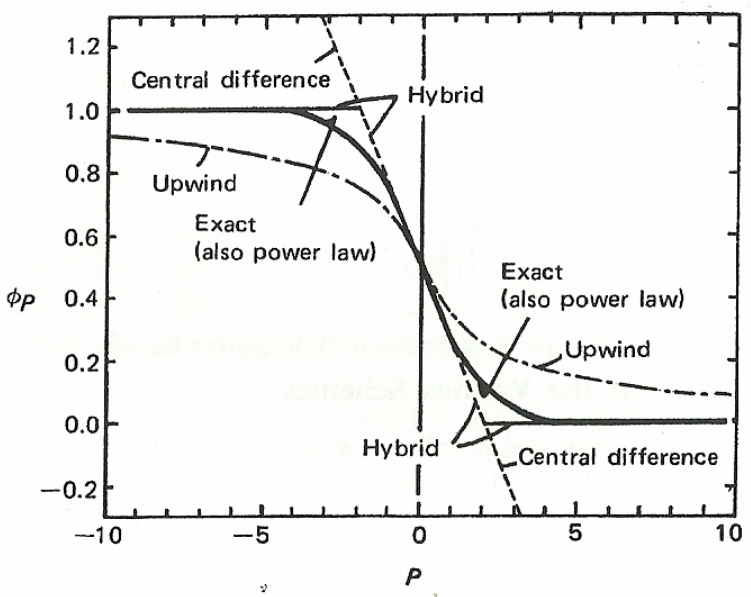

Figure 3.3. Prediction of ϕ_P by the various schemes for a range of Peclet numbers (Source: Patankar, 1980).

The Power Law scheme is given by Eq. (3.12). In this equation, the function A takes the larger value between the two choices, i.e., if zero is greater than the other option, function A takes the value of zero.

$$A_i = \left[0, \left(1-\left(0.1 * P_i\right)\right)^5\right] = \left[0, \left(1-0.1*\left(\frac{F_i}{D_i}\right)\right)^5\right] \tag{3.12}$$

The function A can be put into Eq.(3.11) to evaluate the **a** coefficients of neighboring nodes. In Eq.(3.11), the sign of the convective term depends on the flow direction. Normally, it is assumed that the flow is from west to east, or south to north or

back to front with respect to the flow conditions. Therefore, the **a** coefficient can be written for the neighboring nodal points with its proper sign in front of the convective term as in equations from Eq. (3.13) to Eq. (3.14).

$$a_E = D_e A(|P_e|) + [|-F_e, 0|] \tag{3.13a}$$

$$a_W = D_w A(|P_w|) + [|+F_w, 0|] \tag{3.13b}$$

$$a_N = D_n A(|P_n|) + [|-F_n, 0|] \tag{3.13c}$$

$$a_S = D_s A(|P_s|) + [|+F_s, 0|] \tag{3.13d}$$

$$a_F = D_f A(|P_f|) + [|-F_f, 0|] \tag{3.13e}$$

$$a_B = D_b A(|P_b|) + [|+F_b, 0|] \tag{3.13f}$$

After evaluating the coefficients of neighbors, the coefficient of node P which is the point that is calculated at that instant should be evaluated. Since the **a** coefficients include the dependent variable ϕ at nodal point P because of the evaluation of the variation of it, the coefficient of dependent variable at P is also dependent on its neighboring nodal points. The formula for a coefficient at P is given generally for the internal nodal points in the studies of Patankar (1980) and Malalasekera (2007). In this study, we propose more generalized form of this coefficient including the effects of the boundary nodes. The evaluation of these coefficients are not so clear in literature, because every author assumes its own style and do not give every detail. The proposed **a** coefficient at the nodal point P is given in Eq. (3.14).

$$a_P = a_E + a_W + a_N + a_S + a_F + a_B + \Delta F - S_P \tag{3.14}$$

In Eq. (3.14), the term S_P is the source term coming from its linear form which is given in Eq. (3.10). This term changes according to the boundary conditions. The evaluation of this term will be explained under the Boundary Conditions section.

The general form of the discretized general transport equation (Eq. (3.1)) can be written as in the following form.

$$a_p f_P = \sum_{i=1}^{6} a_i f_i + b_P \tag{3.15}$$

Note that the term b_P in Eq. (3.15) should not be confused with the term b in Eq. (3.8). Here, the coefficient b is the source term coming from the linearized form of it and this term does only include the portion which is independent from ϕ.

3.3. Discretization of Navier-Stokes Equations

3.3.1. Momentum Equation

The momentum equation for a Newtonian fluid can be written in index notation as given in Eq. (3.16).

$$\frac{\partial}{\partial x}(ruf_i)+\frac{\partial}{\partial y}(rvf_i)+\frac{\partial}{\partial z}(rwf_i)=\frac{\partial}{\partial x}\left(m\frac{\partial f_i}{\partial x}\right)+\frac{\partial}{\partial y}\left(m\frac{\partial f_i}{\partial y}\right)+\frac{\partial}{\partial z}\left(m\frac{\partial f_i}{\partial z}\right)-\frac{\partial P}{\partial x_i} \tag{3.16}$$

In Eq. (3.16), sub-index i refers to the direction interested. For z-direction ϕ_i becomes w (velocity component in z-direction) and x_i becomes z. Discretized form of this equation can be obtained by applying the procedure explained in section 3.2 as;

$$\begin{aligned}&(ruA)_e f_e-(ruA)_w f_w+(rvA)_n f_n-(rvA)_s f_s\\&+(rwA)_f f_f-(rwA)_b f_b=\left(mA\frac{\partial f}{\partial x}\right)_e-\left(mA\frac{\partial f}{\partial x}\right)_w+\left(mA\frac{\partial f}{\partial y}\right)_n\\&-\left(mA\frac{\partial f}{\partial y}\right)_s+\left(mA\frac{\partial f}{\partial z}\right)_f-\left(mA\frac{\partial f}{\partial z}\right)_b+(P_r-P_P)*A\end{aligned} \tag{3.17}$$

This equation is valid for the three dimensions. The differences in these three equations are the dependent variable term ϕ and the pressure term P_r, which is the related pressure and is the main momentum source term in most of the engineering

problems. The term P_P is the pressure at nodal point P which is the node whose dependent value is computed at that instant. The P_r would take the three values as given in Table 3.2. The evaluation of the pressure gradient term is given in section 4.2.

Table 3.2. The values of P_r in Eq. (3.17).

Direction	P_r
x-direction	P_W
y-direction	P_S
z-direction	P_B

The velocity gradients seen in the right hand side of Eq. (3.17) is evaluated using the central differencing scheme since these gradient terms are diffusive terms and the distribution of the transported quantity by diffusion occurs along its gradients in all directions whereas convection occurs in the flow direction. Although central differencing works very well for diffusion terms, the convection terms require discretization schemes identifying flow direction and the relative strength of convection to diffusion. Therefore, the convective terms are discretized using Power Law scheme, while diffusion terms are discretized using central differencing scheme in this study. After evaluating the velocity gradient terms and replacing related terms with F and D, Eq. (3.17) is written as,

$$\begin{aligned} &F_e f_e - F_w f_w + F_n f_n - F_s f_s + F_f f_f - F_b f_b = D_e(f_E - f_P) - D_w(f_P - f_W) \\ &+ D_n(f_N - f_P) - D_s(f_P - f_S) + D_f(f_F - f_P) - D_b(f_P - f_B) + (P_r - P_P)A \end{aligned} \quad (3.18)$$

Note that the small letters indicate the values at the faces, while capital ones refer to the values at the related nodes. The values of the F at each of the faces are given through Eq. (3.19a) and Eq. (3.19f). The D coefficients at each of the faces are given through Eq. (3.20a) and Eq. (3.20c) for constant density and constant dynamic viscosity. The evaluation of the D's will be different for case of a non-constant viscosity.

$$F_e = (ruA)_e = 0.5 * r * (\Delta y * \Delta z) * [u_P + u_E] \tag{3.19a}$$

$$F_w = (ruA)_w = 0.5 * r * (\Delta y * \Delta z) * [u_P + u_W] \tag{3.19b}$$

$$F_n = (rvA)_n = 0.5 * r * (\Delta x * \Delta z) * [v_P + v_N] \tag{3.19c}$$

$$F_s = (rvA)_s = 0.5 * r * (\Delta x * \Delta z) * [v_P + v_S] \tag{3.19d}$$

$$F_f = (rwA)_f = 0.5 * r * (\Delta x * \Delta y) * [w_P + w_F] \tag{3.19e}$$

$$F_b = (rwA)_b = 0.5 * r * (\Delta x * \Delta y) * [w_P + w_B] \tag{3.19f}$$

$$D_e = D_w = \frac{(mA)}{\Delta x} = \frac{m}{\Delta x} \Delta y * \Delta z \tag{3.20a}$$

$$D_n = D_s = \frac{(mA)}{\Delta y} = \frac{m}{\Delta y} \Delta x * \Delta z \tag{3.20b}$$

$$D_f = D_b = \frac{(mA)}{\Delta z} = \frac{m}{\Delta z} \Delta x * \Delta y \tag{3.20c}$$

Defining all the required terms, the general form of the discretized momentum equations can be written as in Eq. (3.21).

$$a_P f_P = \sum_{i=1}^{6} a_i f_i + (P_r - P_P) A + b_P \tag{3.21}$$

3.3.2. Continuity Equation

Three dimensional continuity equation is given as in Eq. (3.22). Integrating Eq. (3.22) over the three dimensional control volume and applying the Gauss' Divergence theorem, the discretized form of the continuity equation is written as in Eq. (3.23).

$$\frac{\partial}{\partial x}(ru)+\frac{\partial}{\partial y}(rv)+\frac{\partial}{\partial z}(rw)=0 \tag{3.22}$$

$$(ruA)_e-(ruA)_w+(rvA)_n-(rvA)_s+(rwA)_f-(rwA)_b=0 \tag{3.23}$$

3.3. Grid Arrangement

The arrangement of grid is very crucial in finite volume method. At first glance, keeping all the dependent variables at the same nodal point is seemed to be the simplest method. This arrangement is known as “Collocated Grid” arrangement. It enables to easy implementation for complex geometries and it reduces the memory requirement as all the variables are kept at the same location. However, this grid arrangement can produce unrealistic wavy velocity and pressure distributions. These problems can be avoided by using different control volumes for different dependent variables, also known as “Staggered Grid”, which is our choice for this study and it is widely used in computational fluid dynamics applications. In this arrangement the nodal locations for the velocity components are arranged such that they are located on the control volume faces of the main grid in which the pressure and other scalar variables are stored. The schematic three dimensional view of this grid arrangement is shown in Figure 3.4.

The control volumes for the three velocity components are shifted from the main control volume of P in the direction of arrows shown in Figure 3.4. Since every control volume keeps its variable on its geometric center, the locations of the velocities coincide with faces of the control volume for P. The grid arrangement can be understand easily if the arrangement is given in the two dimensional planes of the computational domain. Figures 3.5, 3.6 and 3.7 are given for this purpose. The shaded areas shown in these figures are the representative control volumes for pressure and velocities.

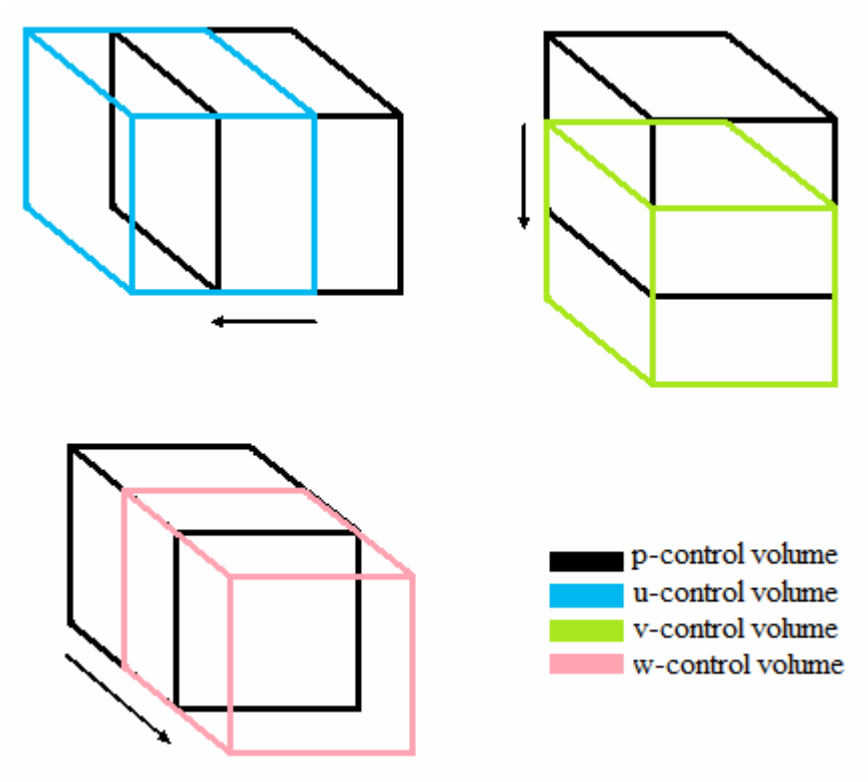

Figure 3.4. Three-dimensional staggered grid arrangement.

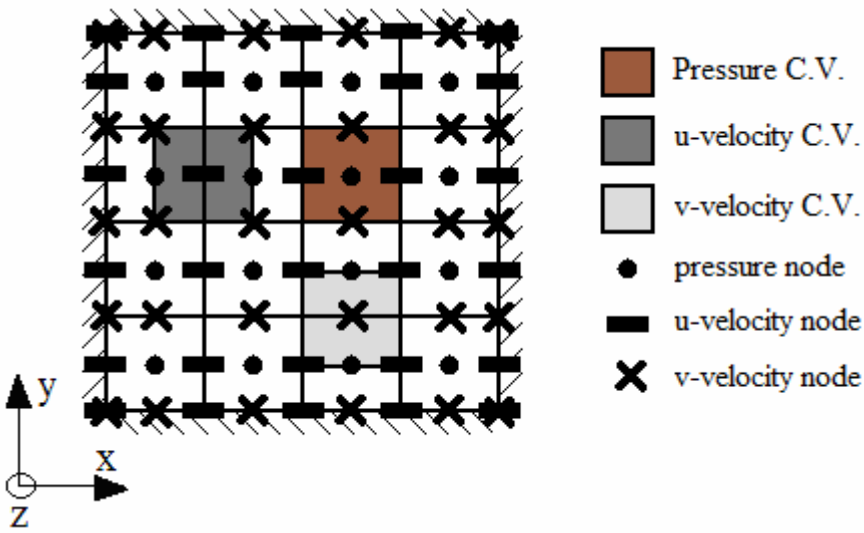

Figure 3.5. Staggered grid arrangement on x-y plane.

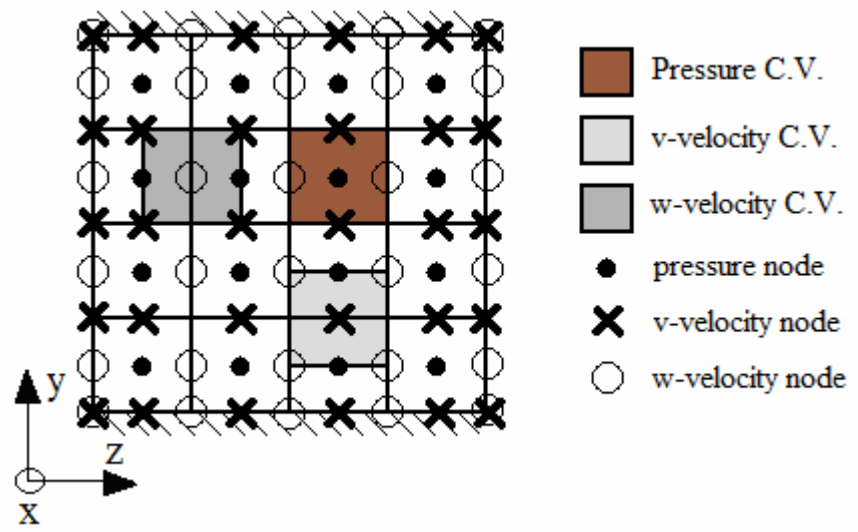

Figure 3.6. Staggered grid arrangement on y-z plane.

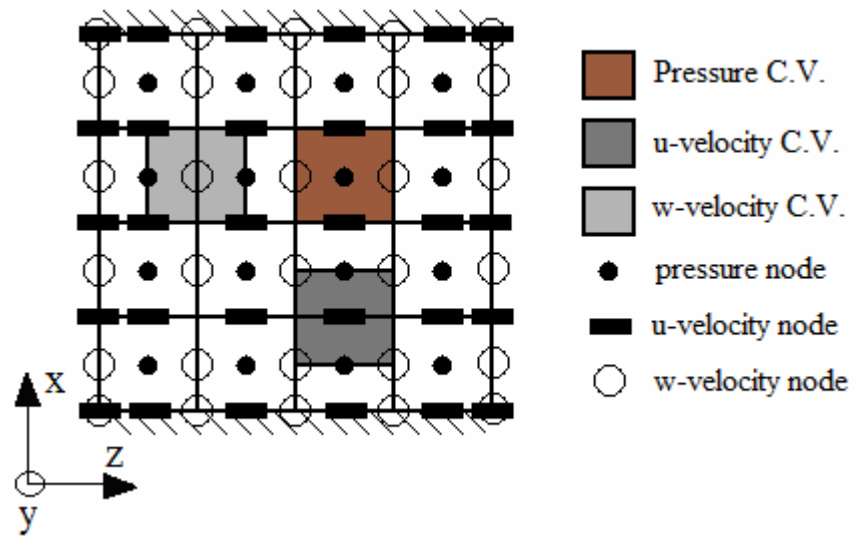

Figure 3.7. Staggered grid arrangement on x-z plane.

The use of staggered grid arrangement has several advantages:

1) Any wavy or unrealistic velocity distributions that satisfy the mass conservation equation are avoided.
2) The natural driving force for any velocity component is the result of pressure difference between two adjacent grid points.
3) The staggered grid arrangement generates the velocity components at the exact locations where they are required for the computation of the transport of the convection and diffusion; therefore there is no need for extra interpolation.

CHAPTER 4

SOLUTION ALGORITHM

4.1. Introduction

Numerical solution of Navier-Stokes equations is not direct as the just convection and conduction problems since the latter does not contain any coupling, extra unknown such as pressure in Navier Stokes and they can be solved directly as stated in the numerical implementation section without any special solution algorithm. Solution for the Navier-Stokes equations, on the other hand, involves more as it has several problems:

i. There is pressure gradient term in the momentum equations which is very important due to physical implementation for the momentum equations, but there is no separate transport equation to compute the pressure.
ii. The convective terms in the momentum equation contain nonlinear quantities.
iii. The equations are internally coupled as the velocity components are presented in each of the momentum and continuity equations.

There are basically two different solution methods used in CFD. These are segregated and coupled solution methods. In segregated solution method, momentum equations in each direction and pressure equation are solved sequentially, i.e., they are segregated from one another. On the other hand, in coupled solution method, the equations are solved simultaneously instead of sequentially. Besides these, the segregated solver bases on the pressure while the coupled solver bases on the density. Therefore, the coupled solver is recommended if a strong inter-dependence exists among density, energy, momentum, and/or species. Within this study, the segregated solver is preferred as the problems of this study are low speed flows, there is no coupling between density and momentum equations and the segregated solver gives faster solution for this type of problems.

The non-linearity and internal coupling can be handled by solving these equations iteratively if the pressure field is known. However, it is not known in almost all problems. If the problem is compressible flow problem, then the continuity equation can be used directly for the computation of the density distribution and from that the pressure field can be calculated as it can be expressed in terms of an equation of state. In the case of incompressible flow, there is no such a link connecting pressure via a state equation since the density is constant. Nevertheless a proper pressure field distribution must satisfy the momentum equation and the velocity field found from the momentum equation then satisfies the continuity equation. From this statement it can be deduced that pressure field is indirectly specified via the continuity equation. All of these problems can be achieved by a proper solution algorithm. One of the solution algorithms was presented by Patankar and Spalding (1972) as SIMPLE algorithm which stands for Semi-Implicit Method for Pressure Linked Equations. In this study, The SIMPLE algorithm is utilized to solve the set of equations.

4.2. Simple Method

The SIMPLE algorithm is a guess and correct procedure for the computation of velocity and pressure field in the domain. The convective fluxes at the volume faces are evaluated from the guessed velocity components. A guessed pressure field is used to solve the momentum equations. The continuity equation is used for the pressure correction equation which is used to correct the velocity and pressure field. At the start of the solution, the values are guessed values, after that point on the velocity components and pressure field are corrected to reach the desired values. Throughout the computations, the next guessed values are the ones obtained from the previous iteration of the computation algorithm. In this manner, the velocity and pressure field can be obtained up to the defined convergence criteria.

4.2.1. Derivation of Pressure Correction Equation

The general discretized form of the momentum equation is given in Eq. (3.12). It can be rewritten for the three velocity components as in Eq. (4.1a, b, c).

$$a_P u_P = \sum a_{nb} u_{nb} + (P_W - P_P) \cdot \Delta y \cdot \Delta z + bu \tag{4.1a}$$

$$a_P v_P = \sum a_{nb} v_{nb} + (P_S - P_P) \cdot \Delta x \cdot \Delta z + bv \tag{4.1b}$$

$$a_P w_P = \sum a_{nb} w_{nb} + (P_B - P_P) \cdot \Delta x \cdot \Delta y + bw \tag{4.1c}$$

To initiate the solution, the velocity field and the pressure field are guessed first and from these guessed values, the Eq. (4.1) turns into guessed form of these equations which are given in Eq.(4.2a, b and c).

$$a_P u_P^* = \sum a_{nb} u_{nb}^* + (P_W^* - P_P^*) \cdot \Delta y \cdot \Delta z + bu \tag{4.2a}$$

$$a_P v_P^* = \sum a_{nb} v_{nb}^* + (P_S^* - P_P^*) \cdot \Delta x \cdot \Delta z + bv \tag{4.2b}$$

$$a_P w_P^* = \sum a_{nb} w_{nb}^* + (P_B^* - P_P^*) \cdot \Delta x \cdot \Delta y + bw \tag{4.2c}$$

Define the pressure correction, p’, as the difference between the correct pressure field, p, and guessed pressure field, p*.

$$p = p^* + p' \tag{4.3}$$

With the same convention, define the velocity corrections u’, v’ and w’ to relate to the correct velocities and guessed ones.

$$u = u^* + u' \tag{4.4a}$$

$$v = v^* + v' \tag{4.4b}$$

$$w = w^* + w' \tag{4.4c}$$

The guessed form of the momentum equations which are given in Eq. (4.2) is subtracted from the correct form of the discretized momentum equations given in Eq. (4.1).

$$a_P(u_P - u_P^*) = \sum a_{nb}(u_{nb} - u_{nb}^*) + \left[(P_W - P_W^*) - (P_P - P_P^*)\right] \cdot \Delta y \cdot \Delta z \qquad (4.5a)$$

$$a_P(v_P - v_P^*) = \sum a_{nb}(v_{nb} - v_{nb}^*) + \left[(P_S - P_S^*) - (P_P - P_P^*)\right] \cdot \Delta x \cdot \Delta z \qquad (4.5b)$$

$$a_P(w_P - w_P^*) = \sum a_{nb}(w_{nb} - w_{nb}^*) + \left[(P_B - P_B^*) - (P_P - P_P^*)\right] \cdot \Delta x \cdot \Delta y \qquad (4.5c)$$

From the definitions of Eq. (4.3) and Eq. (4.4), the Eq.(4.5) can be written as in Eq.(4.6).

$$a_P u_P^{'} = \sum a_{nb} u_{nb}^{'} + (P_W^{'} - P_P^{'}) \cdot \Delta y \cdot \Delta z \qquad (4.6a)$$

$$a_P v_P^{'} = \sum a_{nb} v_{nb}^{'} + (P_S^{'} - P_P^{'}) \cdot \Delta x \cdot \Delta z \qquad (4.6b)$$

$$a_P w_P^{'} = \sum a_{nb} w_{nb}^{'} + (P_B^{'} - P_P^{'}) \cdot \Delta x \cdot \Delta y \qquad (4.6c)$$

The main approximation of SIMPLE method takes action at this point: for the computational convenience, the method drops the terms involving the neighboring nodal points in Eq. (4.6). If the neighboring relations, ($a_{nb}u'_{nb}$) are kept in the Eq.(4.6), they have to be expressed in terms of the pressure and velocity corrections of their neighbors. All the neighbors require their neighbors and the velocity correction equations which are given in Eq. (4.6) would involve all the nodal points in the calculation domain and eventually the resulting pressure correction equation would become unmanageable. With this omission, the pressure correction equation would obey the rule of one-variable at a time procedure as in the general ϕ equation.

Actually, the omission of this term is the basis of the name of the SIMPLE method; i.e., this term represents an implicit effect of the pressure correction velocity field. The pressure corrections at the neighboring locations correct them and the

corrected values of the neighbors alter the velocity of the node under consideration, therefore the effect of the neighbors are included partially, i.e., implicitly.

At the end of the computation what is obtained is the correct solution. As long as the final velocity field satisfies the continuity, the procedure for the pressure correction equation becomes irrelevant with the converged solution.

With this main approximation of SIMPLE method, Eq. (4.6) is written as follow.

$$a_P u_P^{'} = (P_W^{'} - P_P^{'}) \cdot \Delta y \cdot \Delta z \tag{4.7a}$$

$$a_P v_P^{'} = (P_S^{'} - P_P^{'}) \cdot \Delta x \cdot \Delta z \tag{4.7b}$$

$$a_P w_P^{'} = (P_B^{'} - P_P^{'}) \cdot \Delta x \cdot \Delta y \tag{4.7c}$$

Within Eq. (4.7), the corrected velocities are in terms of P’ and from Eq. (4.4), the correct velocities can be written in terms of the corrected version of velocities and pressure. Then the correct velocity components can be written as;

$$u_p = u_P^* + \frac{(P_W^{'} - P_P^{'}) \cdot \Delta y \cdot \Delta z}{a_p} \tag{4.8a}$$

$$v_p = v_P^* + \frac{(P_S^{'} - P_P^{'}) \cdot \Delta x \cdot \Delta z}{a_p} \tag{4.8b}$$

$$w_p = w_P^* + \frac{(P_S^{'} - P_P^{'}) \cdot \Delta x \cdot \Delta y}{a_p} \tag{4.8c}$$

The discretized form of the continuity equation is given in Eq. (3.23). The continuity equation is evaluated at the nodal point P and as can be seen in the discretized form of the continuity equation, it uses the velocity values at the six faces of the control volume. The use of Staggered Grid arrangement enables to use these values directly where they are calculated. Since the velocity components are calculated at the faces of the control volume for pressure, they are ready to use without any interpolation.

To form the pressure correction equation, the velocities in terms of corrected velocity and corrected pressure which are given in Eq. (4.8) are put into Eq. (3.23). Collecting the related terms and rearranging the resulting equation, the pressure correction equation can be written as in the form of the equation for velocities.

$$ap_P P_P^{'} = ap_E P_E^{'} + ap_W P_W^{'} + ap_N P_N^{'} + ap_S P_S^{'} + ap_F P_F^{'} + ap_B P_B^{'} + bp \qquad (4.9)$$

The term **ap** refers the related coefficient of pressure correction term and **bp** is the source term arising from any error in the continuity due to the guessed pressure field. The coefficients in front of the pressure correction values are given in Eq. (4.10).

$$ap_E = r\,\frac{(\Delta y \cdot \Delta z)^2}{a_E} \qquad (4.10a)$$

$$ap_W = r\,\frac{(\Delta y \cdot \Delta z)^2}{a_W} \qquad (4.10b)$$

$$ap_N = r\,\frac{(\Delta x \cdot \Delta z)^2}{a_N} \qquad (4.10c)$$

$$ap_S = r\,\frac{(\Delta x \cdot \Delta z)^2}{a_S} \qquad (4.10d)$$

$$ap_F = r\,\frac{(\Delta x \cdot \Delta y)^2}{a_F} \qquad (4.10e)$$

$$ap_B = r\,\frac{(\Delta x \cdot \Delta y)^2}{a_B} \qquad (4.10f)$$

The **a** coefficients in Eq. (4.10) are given in the momentum equations. The coefficient of the pressure correction at nodal point P is the summation of the coefficients of the neighboring nodal points' coefficients (Eq. (4.11)).

$$ap_P = ap_E + ap_W + ap_N + ap_S + ap_F + ap_B \tag{4.11}$$

For internal nodes, the source term in the pressure correction equation is given in Eq. (4.12). The evaluations of these coefficients will be different for the boundary nodes and they are explained in the related section.

$$bp = r \cdot \Delta y \cdot \Delta z \cdot \left(u_w^* - u_e^*\right) + r \cdot \Delta x \cdot \Delta z \cdot \left(u_s^* - u_n^*\right) + r \cdot \Delta x \cdot \Delta y \cdot \left(u_b^* - u_f^*\right) \tag{4.12}$$

4.2.2. Solution Algorithm of SIMPLE Method

The execution order of the SIMPLE method is given below:

1. Guess the pressure field, p*, and velocity field, u*, v* and w* to initiate the method.
2. Solve the momentum equations to find the new guessed velocity field using the ones at step 1.
3. Evaluate the pressure correction equation by using the values found in step 2. With this step, the pressure correction, p’, values are calculated for all the nodes in the computational domain.
4. Correct the pressure using Eq. (4.3).
5. Correct the velocity components using Eq. (4.8).
6. Solve all other discretised transport equations such as temperature distribution and turbulence quantities if they influence the flow field through fluid properties, source terms, etc.
7. Set the pressure as a new guessed pressure field in the domain and return to step 2 until the convergence criteria is satisfied, in that case the continuity equation should be satisfied.

4.3. On the Iterative Nature of the SIMPLE Method and Pressure Correction Equation

The iterative procedure applied in this study is to solve coupled nonlinear equations. The equations are turned into linear algebraic equations and they are solved

by iteration. During the iterations, the related coefficients are recalculated and the equations are solved again and again until reaching the final solution. Every iteration is an intermediate stage and at every stage all the algebraic equations are solved to find the dependent variables. The correct solution may be obtained with a direct solution method if applicable but this would be very time consuming. The iterative technique greatly simplifies the construction of the numerical method and provides a way in which one can handle any nonlinearity and coupling.

The pressure correction equation is an intermediate stage that leads to the correct pressure field but it has no direct effect on the final solution. Therefore the formulation of the pressure correction equation is not so important as long as the converged solution is reached. However, the convergence rate depends on the formulation of the pressure correction equation. Omission of too many terms may result with divergence.

The pressure correction equation derived in Eq. (4.9) is also prone to divergence if no under-relaxation is applied. Therefore, the pressure correction and the velocity corrections are both under-relaxed according to the following equations.

$$P = P^* + a_P P' \tag{4.13}$$

$$f = (1 - a_f) f + a_f [f^* + f_P (P_r' - P_P') A] \tag{4.14}$$

The under-relaxation factor, α, is normally should be kept between 0 and 1. A value close to 1 results in a fast convergence but is also prone to divergence; on the other hand, a value close to 0 keeps the iterations converging but may result in an unnecessarily extensive computation time. Therefore, in this study, an algorithm called “auto-relaxation” is developed in which relaxation factor is adjusted in a controlled manner. With the aid of auto-relaxation algorithm, the under-relaxation factors are adjusted during the computation time. Checking the convergence history of the current program, it tries to adjust the under-relaxation factor in such a way that both the program is prevented from divergence and the computation time is reduced. The algorithm automatically saves the results at some iteration intervals and uses these values to recover the program in case of any probable divergence. Normally, the under-relaxation factors for the velocities are taking the value of 0.7 while the one for pressure is limited to maximum of 0.6.

In the case of constant density and steady flow situation, the normal velocities are given at all boundaries. Considering the staggered grid arrangement, no calculation is made at boundaries therefore no boundary pressure is specified and all the boundary coefficients related with that boundary becomes zero. As a result, the pressure correction P' and (P' + constant) would both satisfy the pressure correction equation which means that no unique absolute value of P' is obtained. However, during the momentum equation calculations, the differences in the pressure nodes are used, not their absolute values. Therefore, adding a constant to pressure does not change the result. The pressure in this case may be called as relative variable.

The direct question that may arise from this statement is that the values of the pressure correction equation are not unique and the question is that can it be solved or not. Mathematically, if you set up the resulting linear system of pressure correction equations in the form of Ax=b, the coefficient matrix A will be singular and when trying to find a direct solution, it gives the singularity problem and gives no solution with a direct methods. In that case, if the absolute value of it is not unique, does the pressure correction equation converge to a solution? Fortunately, the iterative procedure ensures this. But, in this case, the absolute value depends on the initial guess of the pressure correction equation in the computational domain. To prevent any of these problems, the following is what is done in this study.

i) The values of the pressure correction at every node is made equal to zero before entering the calculation loop of the pressure correction to prevent p' to have large absolute values.

ii) Pressure at a point P of one of the control volumes is set to some reference value and all the other values of p are forced to be calculated relative to the reference value. This also makes the solution unique.

4.4. Solution of Linear System of Equations

The linear system of equations for three components of velocity and pressure correction equations should be solved during the computation. Any linear system of equations can be solved directly or indirectly. The most common direct methods are Inverse matrix method, Cramer's rule and Gaussian Elimination. These methods try to

give the exact solution of the system. The problem with these methods is that if the size of the system is very large and sparse as in the case of this study, it is very time and computer memory consuming. Besides this, if the system has a singularity, the direct methods give no solution.

The iterative methods are more suitable in these cases. They are good at solving large and sparse systems. The iterative methods can be divided into two categories as stationary and non-stationary methods. The non-stationary methods use successive approximations to obtain the solution of the set using orthogonal vectors as in conjugate gradient methods. Most of the computational fluid dynamics methods use the stationary methods which include Gauss-Seidel point by point iteration method, Jacobi iteration method, successive over relaxation and successive under relaxation methods. They are easier to implement since there is no need to construct the set of equations in the form of Ax=b, but they are slower compared with the other methods especially for the sets arising from the three-dimensional computational domains. The solver-subroutines of the developed code include the previously mentioned stationary-iterative solution methods.

4.5. Convergence Criterion

An iterative process is said to have converged if the successive iterations will not produce any change in the value of the dependent variables. The termination is determined by a convergence criterion which is specified by the user. There are several methods to find a proper convergence criteria and it depends on the nature of the problem and aim of the computation. The most common one is looking at the relative change in the calculated values between two successive iterations being greater than a certain small number. The relative change of all the grid point values can be used as a convergence criterion. If heavy under-relaxation is applied during the calculation, this method may be misleading as the change between the successive iterations is slowed down by under-relaxation. In that case, the solution may seem as converged although the calculated values between two successive iterations differ.

In this study, physically more meaningful method is applied to monitor the convergence. What is monitored is how perfectly the discretized equations are satisfied by the current values of the dependent variables. This may be called as “residual” of

that equation. The criterion for the dependent variable and pressure are given in Eq. (4.15) and Eq. (4.16).

$$R_f = \frac{\sum_{nodes} \left| a_P f_P \right| - \left(\sum \left| a_{nb} f_{nb} \right| + A(P_{f-1} - P_f) + b_f \right)}{\sum_{nodes} \left| a_P f_P \right|} \leq e_f \tag{4.15}$$

$$R_P = \sum_{nodes} \left| r \left[A_{y-z}(u_w - u_e) + A_{x-z}(v_s - v_n) + A_{x-y}(w_b - w_f) \right] \right| \leq e_P \tag{4.16}$$

The epsilon value is chosen generally as 10^{-8} in this study. The choice of the epsilon is very crucial during the calculations. It should be noted that if the under-relaxation factors are too small, the epsilon values should be small too to compensate this effect. It would be wise to check during the runs whether the solution has converged or not, because there may be some situations that these epsilon values may not be sufficient for the correctly converged solution. Since the problems of this study are in a channel and the flow is allowed to develop fully, one of the checks for the convergence may be to look whether the flow develops fully at the channel exit or not.

CHAPTER 5

BOUNDARY CONDITIONS AND THEIR IMPLEMENTATION

The boundary conditions used in this study are mostly no-slip boundary condition, inlet boundary condition, outflow boundary condition and symmetry boundary condition. Their use and implementation are crucial in the numerical application because the problem is solved with these boundary conditions. A simple methodology is developed to implement these boundary conditions for the boundary nodes. Even if the boundary conditions are the same, their implementation is different for different locations. For instance, for one of the velocity components, there exist 8 corner, 12 edge and 6 face nodes using the boundary values whose locations are different (Figure 5.1). Since 3 velocity components and 1 pressure correction equation are solved, this means 104 different implementations at the boundary nodes.

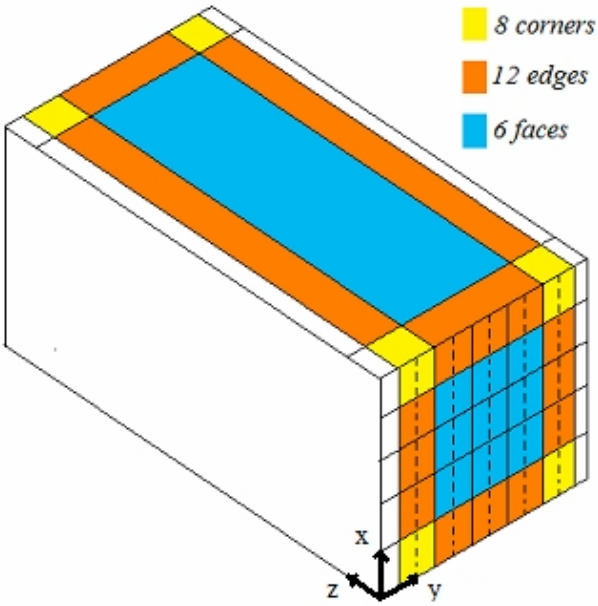

Figure 5.1. Boundary nodes for the u-velocity component.

5.1. No Slip Boundary Condition

No slip boundary condition is the condition applied at the stationary solid walls. If the problem is confined, as the walls of the channel in this study, the velocity components on the wall are set to zero as given in Eq. (5.1).

$$f\big|_{wall} = 0 \tag{5.1}$$

The configuration of the location of the wall and the velocity components are shown in Figure 5.2 for w and v velocity components.

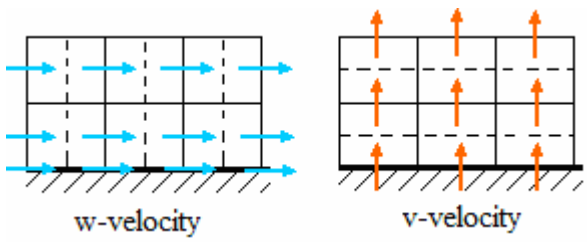

Figure 5.2. Configuration of w and v-velocity components adjacent to the wall. Dashed lines show the edges of the velocity control volumes.

The values of the velocities are set to zero for this boundary condition. Since the velocities are known at these locations, there is no need to solve pressure correction equation there. While solving the equations adjacent to the boundary conditions, the related coefficients are adjusted to implement the related boundary condition. The procedure is as follow:

i. If one of the neighboring value is known, set the momentum coefficient **a** of that neighbor to zero.

ii. Add D±F coefficients to the **a** momentum coefficient at P.

iii. Add u_i*(D±F) to source term **b** for the known neighbor value. Here, since the velocity is zero, this will add nothing to source term **b**. However, if the wall is moving or there is a suction and/or injection on the wall, it is implemented directly with this procedure.

iv. For the pressure correction equation, set the coefficient for pressure **ap** at related boundary to zero and use velocity directly, not the guessed one in the calculations. No extra implementation is required.

5.2. Inlet Boundary Condition

At the inlet boundary, flow variables are specified. At the inlet either the pressure or inlet velocities are specified. If the exact details of the flow distribution are unknown, but the boundary values of pressure are known, the solution may be obtained from this pressure inlet boundary condition. If the velocity components are known on the inlet section instead of pressure, then inlet velocity boundary condition is used as in this study. That is the inlet velocity profile is prescribed at the inlet boundary (Eq. (5.2)). Since the velocity components are given, the pressure corrections at these points are zero meaning that no calculation of any momentum equation at these points is needed. The schematic diagram of the inlet boundary is given in Figure 5.3 for w and v velocity components.

$$f\big|_{inlet} = f_0 \tag{5.2}$$

The procedure is same as with the procedure for no-slip boundary condition. The only difference is that at least one of the velocity components must be different from zero. Figure 5.4 shows where the related velocity components are calculated with their symbols.

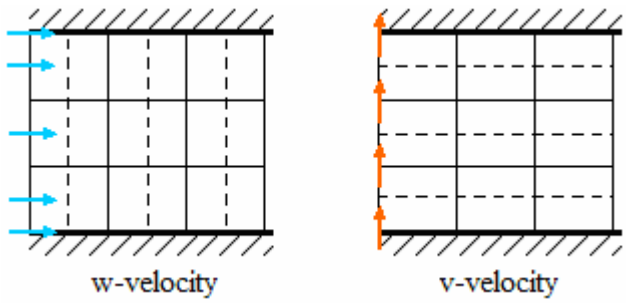

Figure 5.3. Schematic diagram of the inlet boundary. Dashed lines show the edges of the velocity control volumes.

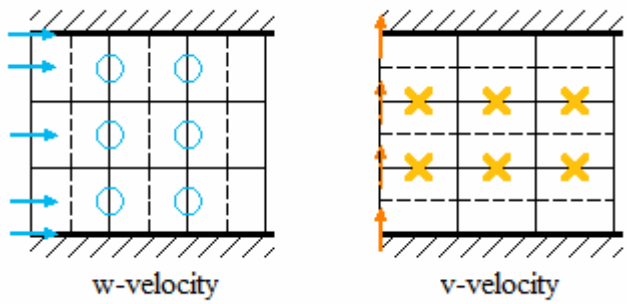

Figure 5.4. Schematic diagram of the velocity nodes adjacent to the inlet boundary, symbols show where the related velocities are computed. Dashed lines show the edges of the velocity control volumes.

5.3. Outflow Boundary Condition

When the location of the outlet section is far enough from any disturbance, the flow reaches its fully developed condition and at that location the outflow boundary condition can be used. With the outflow boundary condition, it can be stated that there is no change in the flow direction for the dependent variables. Practically, the gradients of all the variables in the flow direction are set to zero at that location as given in Eq. (5.3).

$$\left.\frac{\partial f}{\partial z}\right|_{outlet} = 0 \tag{5.3}$$

With this boundary condition, there is no calculation for the velocity components and pressure correction at the outlet. No change in the gradient is implemented as making the velocity components equal to the previous nodes in the flow direction as can be seen in Figure 5.5. The procedure is as follow:

i. Make the gradients of dependent variables in flow direction equal to zero, i.e., $\frac{\partial f}{\partial z} = 0$. This means that $\frac{f_{out} - f_{previous}}{\Delta z} = 0 \Rightarrow f_{out} = f_{previous}$. In other words, the values of the dependent variables at the outlet are made equal to the values of the previous nodes.

ii. For the boundary nodes in which the velocities are calculated adjacent to the outlet boundary, the momentum coefficient **a** at the front neighbor is set to zero.

iii. There is nothing to add to the momentum coefficient **a** for the outflow side and nothing to add for the source term **b**.

iv. Since the outflow condition used, the velocity components are taken from the previous nodes, therefore there is no pressure correction equation to be solved at these locations; the pressure correction is zero there.

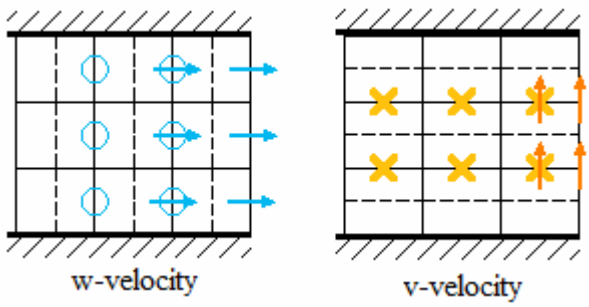

Figure 5.5. Outflow velocity components for w and v-velocity, symbols show where the related velocities are computed. Dashed lines show the edges of the velocity control volumes.

5.4. Symmetry Boundary Condition

The symmetry boundary condition is used when there is a flow in which the flow pattern is symmetric in one or more directions. In this study, the advantage of symmetry boundary condition is used since it decreases the computational node number at least by half. Physically, a symmetry boundary condition is a location where there is no flow across the boundary and no scalar flux across the boundary. The schematic view of this boundary is shown in Figure 5.6.

The ghost cells seen in Figure 5.6 are used in the calculation, however they are not calculated separately, their value is taken from its symmetric peer (The blue velocity arrows shown in Figure 5.6. are symmetric peers for w-velocity). The values on the ghost cells are taken from the neighbors adjacent to the symmetry plane. The normal velocity is zero on the symmetry plane; that is u-velocity on the symmetry plane is zero.

The related velocity component here is the orange one. The symmetric pairs can bee seen in Figure 5.6.

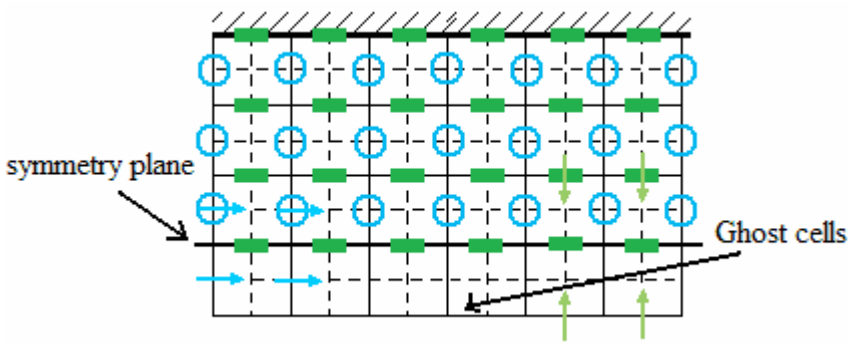

Figure 5.6. Velocity components adjacent to symmetry plane, circles represent w velocity, rectangles represent u-velocity components. Representative velocity vectors show the symmetry about the symmetry plane.

Since the velocity value on the symmetry plane is zero, there is no velocity vector on it. The mathematical definition is given in Eq. (5.4). It states that there is zero normal velocity and zero normal gradients of all variables at a symmetry plane.

$$f_n\big|_{symmetry\ plane} = 0$$
$$\frac{\partial f_t}{\partial n}\bigg|_{symmetry\ plane} = 0 \qquad (5.4)$$

The procedure for the implementation is as follow:

i. For the calculation of the momentum equation adjacent to the symmetry plane, set the related **a** momentum coefficient to zero since the values on the ghost cells are known from the previous iteration.

ii. Add D±F coefficients to the momentum coefficient **a** at P.

iii. Add u_i*(D±F) to source term **b** for the known neighbor value.

iv. In the pressure correction equation, set the related **ap** coefficient for pressure to zero; no further modification is required.

CHAPTER 6

CODE VALIDATION

Due to the inherent assumptions, linearizations and discretizations, every CFD code is prone to inaccuracies whether small or large. After developing the code, numerical validation must be done in order to justify its correctness. The ultimate way of testing a CFD code is a comparison between its output and experimental data and/or the analytical solution if exists. One of the methods reporting the validation study is to choose a target quantity and to compare the results. If the difference between computed and experimental results and/or analytical solution looks sufficiently small, the CFD code is considered to be validated. In this chapter, only the problem descriptions and results are given, the details of the developed code, the properties of the computer used throughout this study and the average iteration time can be found in Appendix A.

6.1. Developing Channel Flow

The problems of this study are for separated channel flows. Therefore, validating the code for a three dimensional developing channel flow is the first task to show how the code deals with this relatively simpler flow. The most valuable work done on the three dimensional laminar flows in ducts are the work of Shah & London (1978) and the work of Ravi Shankar *et al*. (1993). Numerical results for a developing channel flow are also given in the study of Saldana (2005).

6.1.1. Problem Description

The problem is a pure forced convective developing flow in a square channel with a uniform inlet velocity. The geometry of the computational domain is shown in Fig. 6.1. The Reynolds number based on the hydraulic diameter is chosen as 100 with the purpose of comparing the computational results with literature. To make the flow reach to its fully developed condition, the channel length, L_{dc}, is set to 40 times to its

hydraulic diameter and the results of the computation show that this length is sufficient to assume a fully developed flow at outlet of the channel.

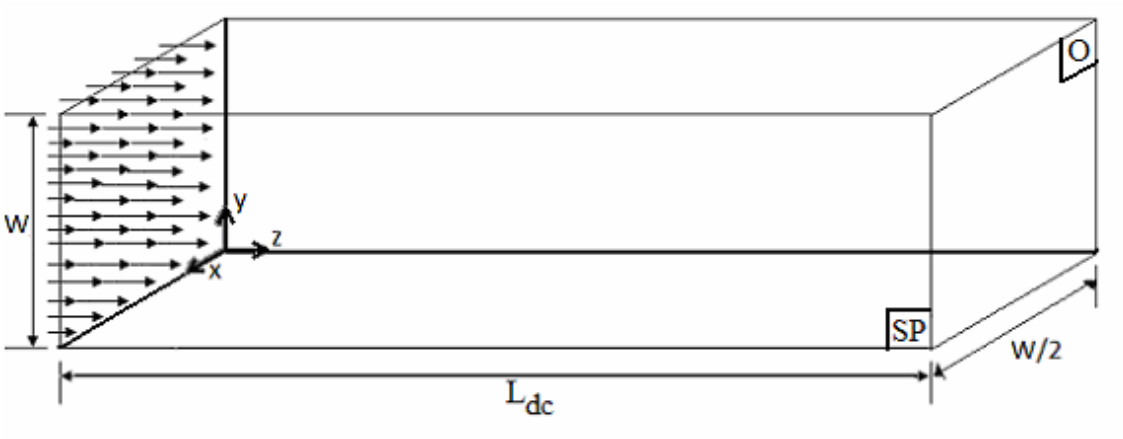

Figure 6.1. Developing channel flow geometry, SP: symmetry plane, O: outlet plane.

6.1.2. Boundary Conditions

A uniform inlet velocity boundary condition is given at the inlet. The axial velocity component at all the inlet nodes is set to uniform velocity and the other two components of the velocity are set to zero. The problem is geometrically symmetric, therefore only the half of the domain is solved and symmetry boundary condition is applied at the symmetry plane shown in Figure 6.1. At the walls, no slip boundary condition is applied and at the outlet, outflow boundary condition is applied. The mathematical definitions of the boundary conditions are given in Table 6.1.

Table 6.1. Boundary conditions of developing channel flow.

	Boundary condition	**Mathematical definition**
Inlet	Inlet velocity	$u=v=0,\ w=w_0$
Walls	No slip	$u=v=w=0$
Symmetry plane	Symmetry	$u=0,\ \frac{\partial v}{\partial x}=\frac{\partial w}{\partial x}=0$
Outlet	Outflow	$\frac{\partial u}{\partial z}=\frac{\partial v}{\partial z}=\frac{\partial w}{\partial z}=0$

6.1.3. Results and Discussion

The computational grid is first chosen to be uniform and the control volume numbers in x, y and z directions are chosen to be 20, 40 and 200 respectively, making a total of 1.6×10^5 control volumes. If the symmetry boundary condition was not implemented then 3.2×10^5 would be required. The convergence criterion is set to be 10^{-8} for all three velocity components and pressure.

Theoretical studies of Shah and London (1978) give the values of Fanning Friction Factor and maximum velocity at the outlet. These type of integral quantities and specific values of velocity are classical choices for CFD code validation. Fanning friction factor, named after J. T. Fanning, is a dimensionless parameter used in fluid flow calculations and it is defined as in Eq. (6.1) where t, r, w_0 are wall shear stress, density and average velocity respectively.

$$f_{Fanning} = \frac{2t}{r w_0^2} \tag{6.1}$$

The Fanning Friction Factor should not be confused with Darcy Friction Factor which are very similar quantities. Darcy Friction Factor is four times the Fanning Friction Factor.

$$f_{Darcy} = \frac{-\frac{dp}{dz} D_h}{r w_0^2 / 2} \tag{6.2}$$

where D_h is the hydraulic diameter which is defined in Eq. (6.3).

$$D_h = \frac{4A_c}{P} \tag{6.3}$$

Table 6.2 gives the comparison between current numerical computations and theses theoretical results. The percent error is found to be less than 0.4 and this shows that the code is trustable for the forced convection in a channel.

Table 6.2. Comparison of w_{max} /w_0 for the developing square channel flow.

	fRe (Fanning)	fRe(Darcy)	w_{max}/w_0
Shah and London(1978)	14.2270	-	2.0962
Incropera *et al.* (2007)	-	57	-
Present study	14.1970	56.789	2.0908
Error (%)	0.2113	0.3716	0.2583

In addition to the comparison made in Table 6.2, we compare the axial velocity at the outlet with the analytical velocity distribution given in the book of Frank M. White (2006). The constant velocity contours are given in Figure 6.2 at the outlet x-y plane. The present numerical computations match very well with the analytical velocity distribution for the fully developed non-circular duct flow and this one way validates the developed code.

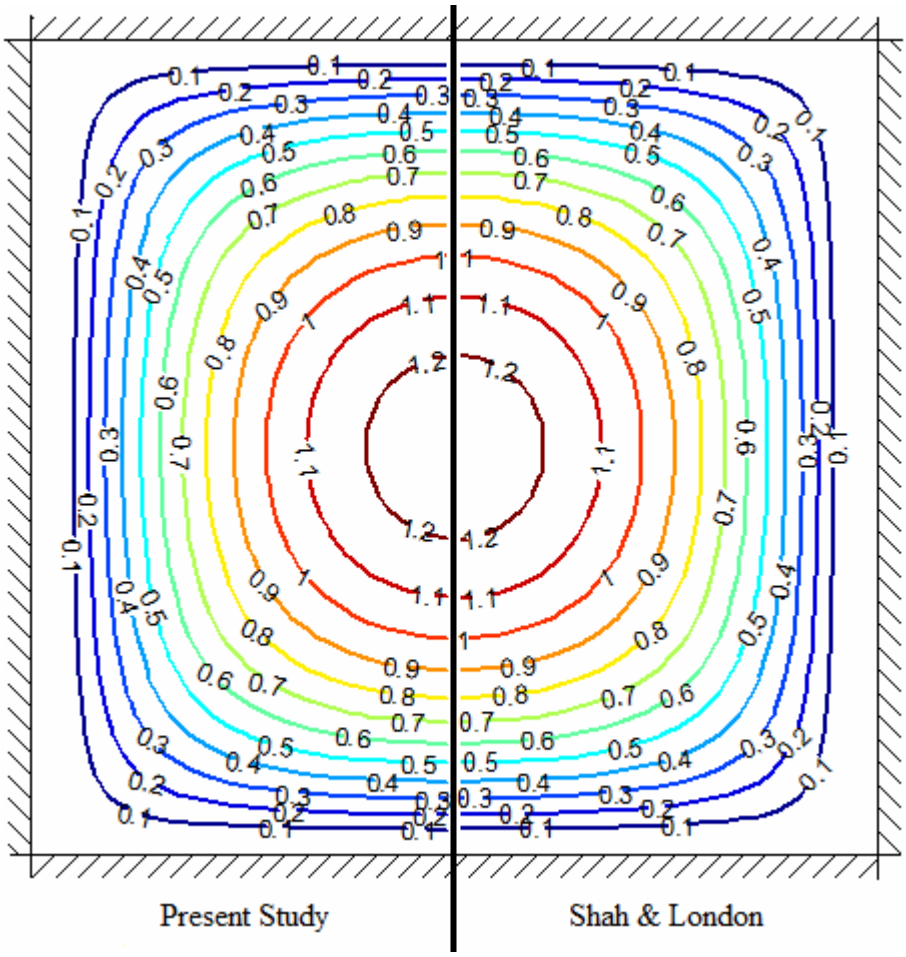

Figure 6.2. Axial velocity contour comparison of present study and the analytical velocity distribution given by Shah & London (1978) in the book of White (2006), velocity labels are in m/s.

Figure 6.3 shows the axial velocity profiles on the symmetry plane at different z locations (flow direction). The velocity profile is not parabolic in the developing region, but at the channel exit where it develops fully, the profile is parabolic.

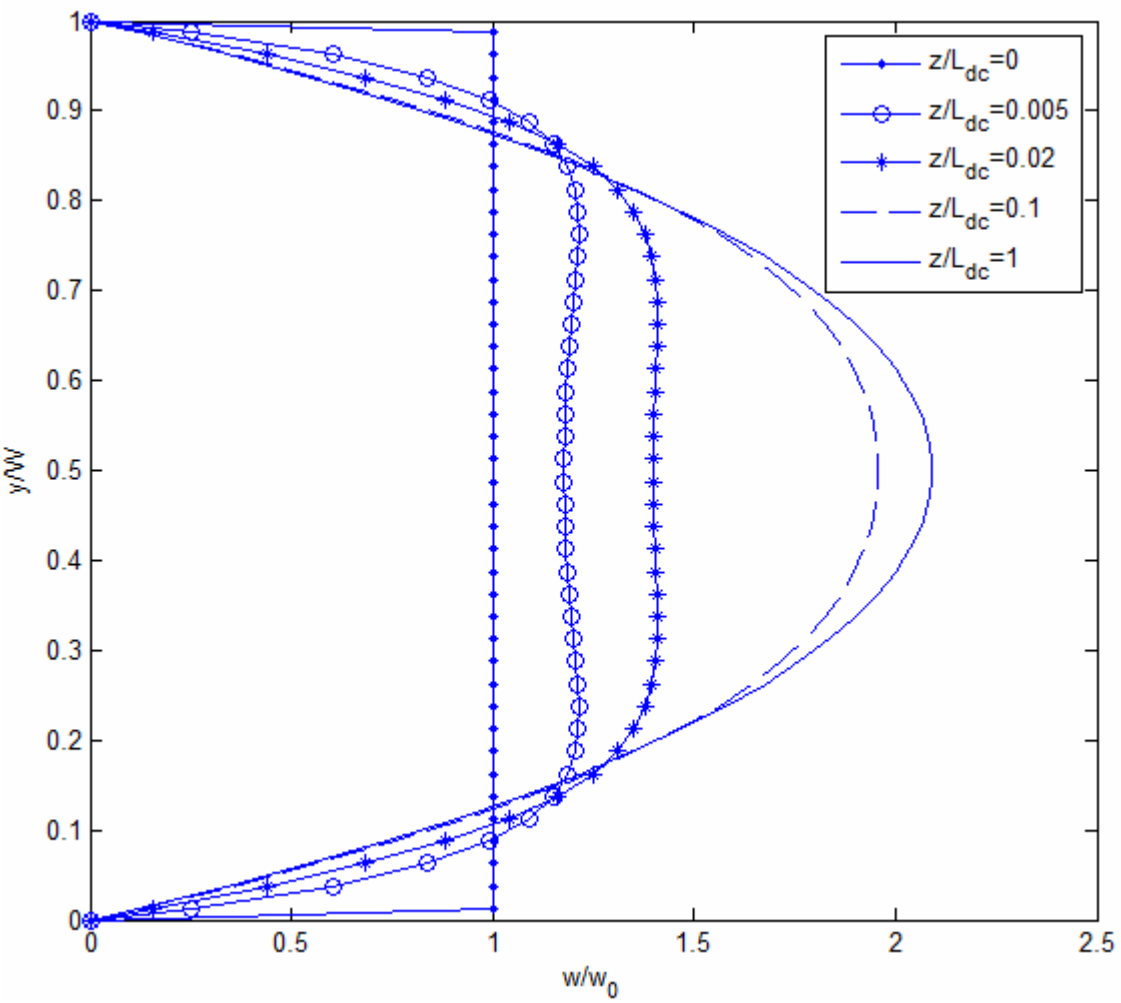

Figure 6.3. Stream-wise velocity profiles on the symmetry plane for different locations.

6.2. Artificial Viscosity Method

A code developed for a constant area channel flow can easily be adapted to any geometry with rectangular features by artificial viscosity method in which walls are modeled as a fluid with very high viscosity. For example, to simulate a solid region in the domain, it is known that the velocity components in this domain are zero. One of the two methods to achieve this is use of large source terms and the other method is artificial viscosity method which is the use of very high dynamic viscosity in that control volumes such that the control volumes behave as if they are solid in that region.

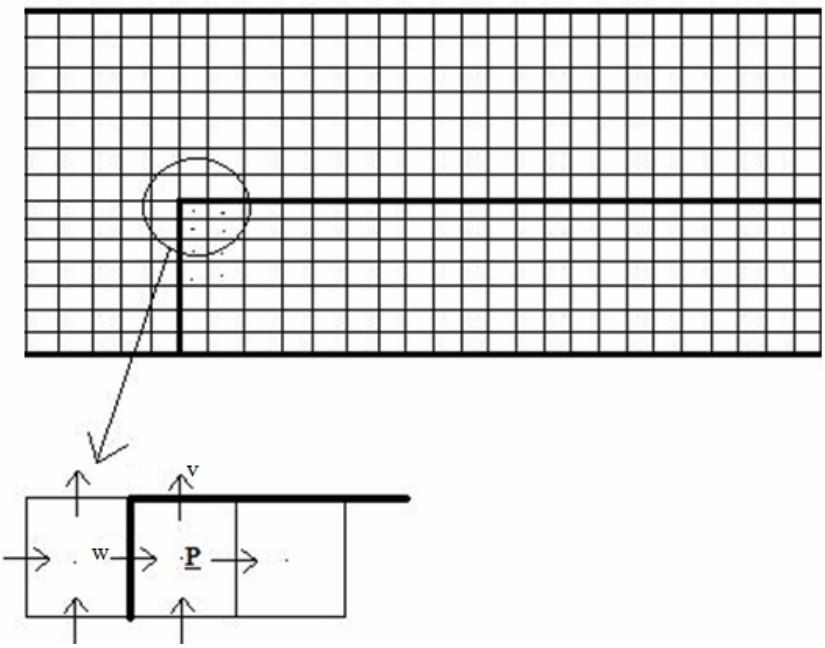

Figure 6.4. Location of control volumes in artificial viscosity method.

An example for this type of arrangement is shown in Figure 6.4. The control volumes shown in the figure are for the scalar quantities, such as pressure density and viscosity. Using very high viscosity value inside the inactive or artificially solid region, the desired solid effect can be created. A few representative control volumes for the high viscosity values are shown in Figure 6.4 with dots inside the control volumes. The locations of the velocity components and scalars (point P) are shown in the magnified portion of the corner of the forward facing step. The high viscosity values are kept inside the domain at location P while the velocity components are kept on the walls of the step. If the viscosity values at these points evaluated properly, the velocities on the step should be zero because of the no-slip condition.

6.2.1. Evaluation of Viscosity at Interface

The problem of artificial viscosity method is the large discontinuities between the zones of normal viscosity and artificial viscosity. The interface viscosity should be evaluated in such a way that the flow should not be affected from this discontinuity.

The interface viscosity can be calculated by assuming a linear variation between related control volumes. The interface viscosity of control volumes P and E (Figure 6.5) can be calculated as in Eq. (6.4).

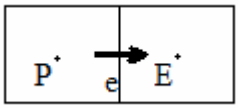

Figure 6.5. Interface for the P and E control volumes.

$$m_e = f_e m_P + (1 - f_e) m_E \tag{6.4}$$

where f_e is the interpolation factor defined as follows:

$$f_e = \frac{dz_{eE}}{dz_{PE}} \tag{6.5}$$

where dz in Eq. (6.5) is the horizontal distance between the points defined in sub-index. If the grid is uniform, f_e becomes 0.5 meaning that interface viscosity is calculated taking the arithmetic mean of the two control volumes' viscosity values or a linear variation for non-uniform grid arrangement adjacent to the artificial boundary. This type of calculation would result with a very high viscosity at the interface which is not desired since intended wall location would move up between the interface and the first boundary node in this case. Fortunately, taking the harmonic mean of related control volumes can handle the viscosity at the interface correctly and handles the abrupt change without requiring an excessively fine grid in the vicinity of the change. The harmonic mean is calculated as given in Eq. (6.6).

$$m_e = \left(\frac{1 - f_e}{m_P} + \frac{f_e}{m_E} \right)^{-1} \tag{6.6}$$

Since very high viscosity value is used with this method and let this be m_P, then the first term in Eq. (6.6) becomes very small and the high viscosity does not affect the calculations as in the case of linear variation method. Here the interface viscosity is not all dependent on the viscosity value at P. Besides this, m_e is not equal directly to m_E, but 1/ f_e times of it.

For the three dimensional calculations, the locations of the six neighboring control volume locations are very different for different interfaces and different velocity components. While calculating the interface velocities, the neighboring control volumes enter into the calculation. For example, Figure 6.6 shows the control volumes where viscosity is kept and the location of u-velocity on the x-y plane. While evaluating the momentum equation for the u-control volume (shaded control volume in Fig. 6.6), one of the required interface viscosity is, for instance, the north face of the u-control volume.

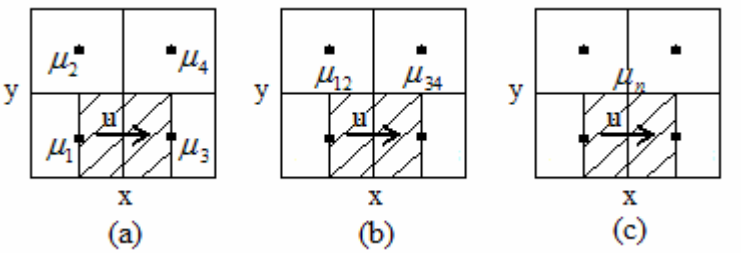

Figure 6.6. Location of viscosity values and u-velocity component, (a) viscosities at their local points, (b) and (c) at interfaces.

To calculate the interface viscosity at the north face of u-control volume, first the interface viscosity between nodes for m_1 and m_2 should be calculated using Eq. (6.6), and then the interface viscosity between nodes for m_3 and m_4 is computed which result in m_{12} and m_{34} respectively (Figure 6.6 (b)). The interface viscosity at the north face, m_n (Figure 6.6 (c)), of the u-control volume is obtained using again Eq. (6.6) with interface viscosities m_{12} and m_{34}. All the velocity components and faces require this type of interface viscosity evaluation.

6.2.2. Validation of the Artificial Viscosity Method

To validate the artificial viscosity method, developing channel flow is used. The same problem is prepared once with a true no-slip boundary condition and then again with an artificially solidified wall with a very high viscosity. The geometries of the two test cases are shown in Figure 6.7. The height of the channel for the normal test case is set to h and the one with artificial viscosity is set to 2h in which the bottom portion is filled with high viscosity of 10^{50} kg/ms. With this arrangement, it is expected that the region filled with high viscosity behaves as if solid and two solutions give the same exact answer.

The number of control volumes used in normal test case is 1×10^4 and the number of control volumes used in the test case with artificial viscosity is 2×10^4. With this control volume arrangement, it is ensured that all the control volume sizes are equal in both of the test cases; therefore the number of control volumes in the second test case is twice the number of control volumes of the normal test case.

The boundary conditions for the normal test are same as with the developing channel flow except the Reynolds number is set to 400 in this case. As for the test case with artificial viscosity method, the inlet velocity profile is uniform for the range of y=h to y=2h and zero for the range of y=0 to y=h. If shaded region would not be filled with very high viscosity fluid, it would be expected that the flow behaved as if there would be a sudden expansion.

The validation of the application of the method is made comparing the axial velocities on the symmetry plane at different z locations. The second test case does not behave as if there is a sudden expansion, but it gives the same result with the normal test case. The streamwise velocity profiles on the symmetry plane at different z locations are shown in Figure 6.8. The results from the both cases match perfectly well and this validates the application of the method for any irregular shape inside the computational domain converting the regular channel geometry artificially into geometrically different channels.

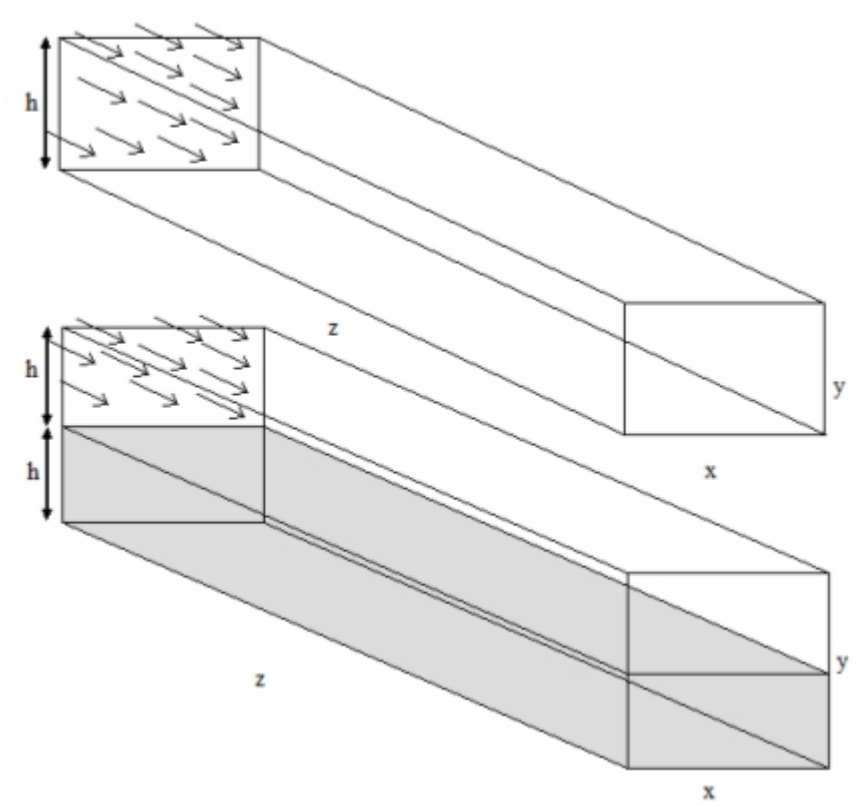

Figure 6.7. The geometry of the test problems for artificial viscosity method, top: normal test, bottom: test with artificial viscosity.

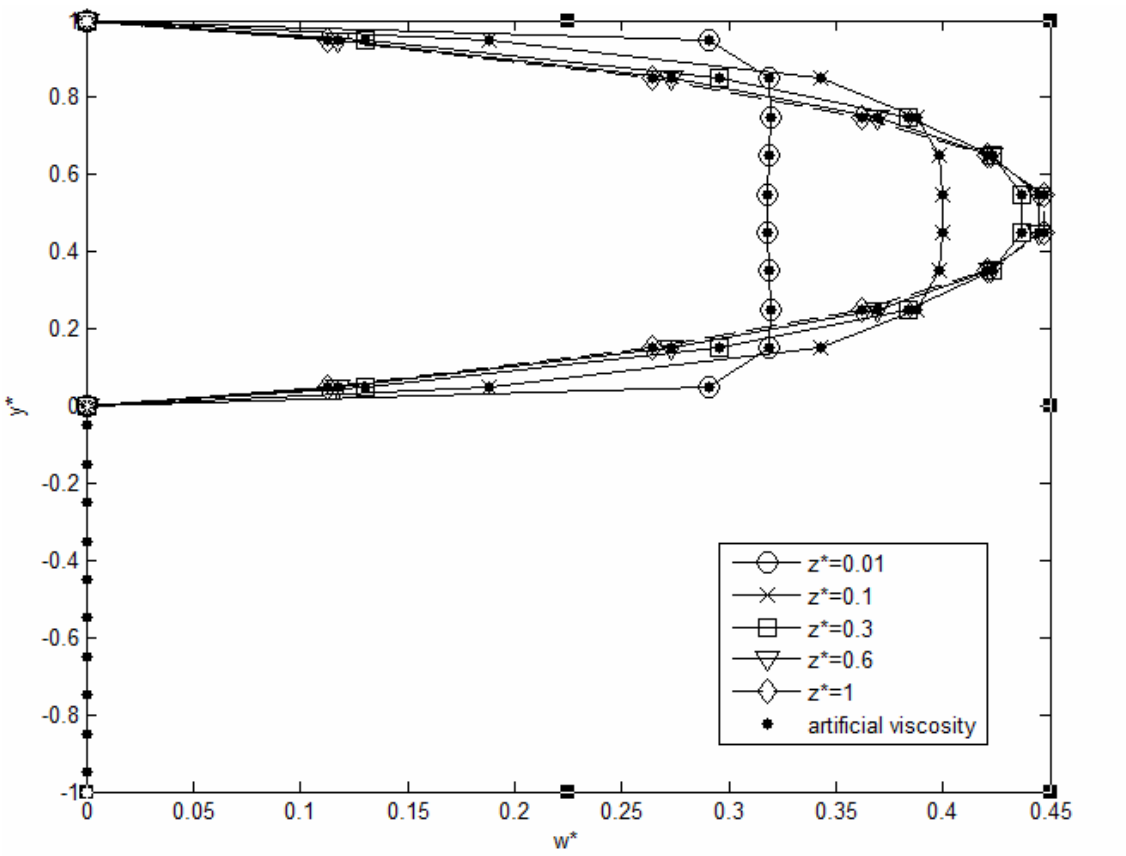

Figure 6.8. Streamwise velocity profiles on the symmetry plane for different z-locations.

6.3. Backward Facing Step Flow

Flow separation and the subsequent reattachment is one of the fundamental research topics in fluid dynamics. To investigate the details of separation and reattachment phenomenon, some set of geometric test cases has been developed for the researchers. These are flow in a pipe with sudden expansion, flow in a pipe including an obstacle such as an orifice, flow over an obstacle such as step or fence in a channel and flow in a channel with a sudden expansion which is also called as backward facing step flow in a channel. Among these geometric configurations, the backward facing step is very popular benchmarking problem and validation test problem for the numerical studies. Although the geometry is simple, the flow contains most of the complexities that are encountered in other separated flow geometries. Most of the studies for backward facing step flow is in 2-D, however all the actual systems containing the step geometry is 3-D, therefore there is a need to understand the 3-D vortical flow structure behind the step.

The main aim of investigation of this type of flow problem is due to its simplicity in its geometry, complexity in its flow structure and availability of reliable experimental data. The objective of the current test case is investigate steady laminar flow over 3-D backward facing step and make comparison with the experiments as a validation of the developed solver.

6.3.1. Problem Description

The problem is a forced convection over a 3-D backward facing step in a channel. The schematic view of the computational domain is shown in Figure 6.9. The step geometry and flow conditions are chosen according to the study of Armaly ***et al.*** (2003) due to the ease of comparison of the results and the validation of the code developed. The upstream height of the duct, h, is 0.01 m, the step height, s, is 0.01 m and the width of the channel, W, is 0.08 m. The aspect ratio and expansion ratio are given in Eq. (6.1) and Eq. (6.2).

The length of the step, L_s, is 0.02 m and the length of the downstream of the step, L_d, is set to 0.5 m which is 50 times of the step height to ensure a fully developed conditions at the outlet plane. The choice of the upstream length of the step is for preventing the inflow from the effects of sudden expansion.

The problem is geometrically symmetric in span-wise direction (x-direction), therefore the half of the computational domain is solved using L=W/2 portion of the channel. The coordinate system is put at the bottom corner of the step and the y-z plane at x=W/2 is the symmetry plane.

$$AR = \frac{W}{s} = 8 \tag{6.1}$$

$$ER = \frac{h+s}{h} = 2 \tag{6.2}$$

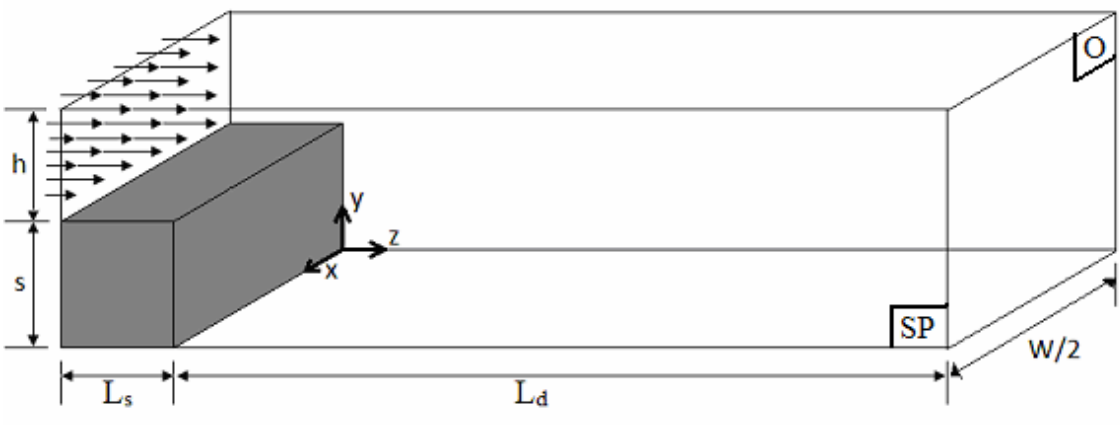

Figure 6.9. Geometry of the backward facing step channel, SP: symmetry plane, O: outlet plane.

The flow of air is simulated in this problem and its physical properties are evaluated as constants throughout the simulation as density, r , is 1.205 kg/m^3 and dynamic viscosity, m is 1.81 x 10^{-5} kg/ms. The validation tests are made for Reynolds number (defined in Eq. 6.3) of 343.

$$\mathrm{Re} = \frac{2 r w_0 h}{m} \tag{6.3}$$

6.3.2. Boundary Conditions

At the inlet section of the channel, fully developed velocity profile is used. The velocity distribution is given analytically in the book of White (2006) and Eq. (6.4) gives the fully developed velocity distribution used for the inlet boundary condition. The u and v velocity components are set to zero at inlet. For a rectangular cross section shown in Figure 6.10, $-a \le x \le a, -b \le y \le b$:

$$w(x,y) = \frac{16a^2}{mp^3}\left(-\frac{dp}{dz}\right)\sum_{i=1,3,5...}^{\infty}(-1)^{(-i-1)/2}\left[1-\frac{\cosh(ipy/(2a))}{\cosh(ipb/(2a))}\right]\cdot\frac{\cos(ipx/(2a))}{i^3} \tag{6.4a}$$

$$Q = \frac{4ba^3}{3m}\left(-\frac{dp}{dz}\right)\left[1 - \frac{192a}{p^5 b}\sum_{i=1,3,5...}^{\infty}\frac{\tanh(ipb/(2a))}{i^5}\right] \tag{6.4b}$$

In Eq. (6.4b), Q is the volumetric flow rate and it is known from the inlet boundary condition. The pressure gradient term in Eq. (6.4a) is calculated from Eq. (6.4b) and then the velocity distribution is computed with Eq. (6.4a).

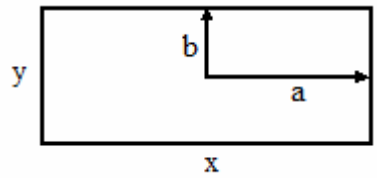

Figure 6.10. Rectangular cross section for fully developed channel flow.

Because of the symmetry of the problem in the span-wise direction, symmetry boundary condition is applied at x=W/2 y-z plane. At the walls of the channel no-slip boundary condition is applied and at the outlet plane, outflow boundary condition is used. The locations of the boundaries and the mathematical definitions of the boundary conditions are given in Table 6.3.

Table 6.3. Boundary conditions of backward facing step flow.

	Boundary condition	**Mathematical definition**
Inlet	Inlet velocity	u=v=0, w=w(x,y) (Eq. 6.2)
Walls	No slip	u=v=w=0
Symmetry plane	Symmetry	u=0, $\frac{\partial v}{\partial x} = \frac{\partial w}{\partial x} = 0$
Outlet	Outflow	$\frac{\partial u}{\partial z} = \frac{\partial v}{\partial z} = \frac{\partial w}{\partial z} = 0$

6.3.3. Grid Setup

Non-uniform staggered grid arrangement is used for the backward facing step flow. The grid is highly concentrated close to the step and adjacent to the walls in order to assure the accuracy of the numerical simulation. What is desired for non-uniform grid generation is constructing a fine grid in the regions of strong gradients such as walls and step corner and a coarse grid for the regions where the variations of the dependent variables are small, for example far from the step. There are several non-uniform grid generation schemes such as successive ratio, exponential and logarithmic schemes and they are used in commercial softwares too. The successive ratio schemes may produce very big sized coarser grids compared to exponential and logarithmic ones if the computational domain is too long. The choice of the grid generation scheme, however, should depend on the details of the problem. In this study, we utilized a successive ratio logic because the size of the coarser grids produced near the outlet plane do not affect the solution of the problems of this study. But it should be noted that, if any problem is suspected from the size of the coarser grids, the exponential or logarithmic schemes may be tried.

To generate non-uniform grid, a basic mathematical approach is applied. At this point, where the control volume faces are located should be defined. The two practices are seen in the literature: practice A and practice B (Patankar, 1980). In practice A, the grid points are located first, and then the faces are located midway between the grid points. In Practice B, the faces are located first and then the grid points are placed at the centers of the control volumes as shown in Figure 6.11 (dots are the grid points and the vertical lines are the faces.). For uniform grid, both of the practices are identical. For non-uniform grid arrangement, both of them have pros and cons. For example in Practice A, the evaluation of the gradients at the faces are very accurate, since the faces are midway between grid points, however the nodes are may not be at the center of the control volume therefore the value of the variable kept at this location may not be a good representative of the control volume. In Practice B, there is not a problem like this since the nodes are at the geometric center of the control volumes. However, the control volume faces are not midway between the nodes and the evaluation of the gradients requires special attention. Although there is a disadvantage like this, it is more convenient to use Practice B as the control volumes are the basic elements of the finite

volume method. Besides this, Practice A requires half control volumes around the boundary grid points while the computational domain is filled with regular control volumes with Practice B, since the locations of the faces are determined first. It is easier to fill the domain, more importantly; the boundary control volume faces coincide with the boundary or any discontinuity in the material as in the case of artificial viscosity method. Considering all these pros and cons, Practice B is selected to be used in this study.

To generate the non-uniform grid with Practice B, i.e., to locate the control volume faces first; we should find a relation for the distance of the faces to the boundary. If the first control volume is the starting control volume with length of Δx for this arrangement, then the length of the next control volumes are multiplied with an expansion factor "r" which is kept between 1 and 1.5. Therefore the size of the next control volume is always greater than the previous one. The schematic view is shown in Figure 6.11.

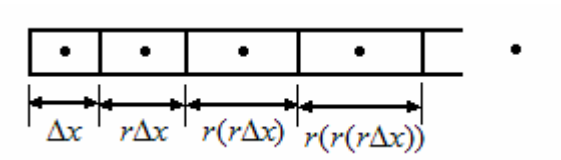

Figure 6.11. The schematic view of the non-uniform grid arrangement.

If a distance of L is divided into N control volumes with Δx being the length of the first control volume, then the sum of the lengths of the sub-control volumes should give the total length L (Eq. (6.5)).

$$\Delta x + r\Delta x + r(r\Delta x) + r(r(r\Delta x)) + ... = L \tag{6.5}$$

Eq. (6.5) can be written as in Eq. (6.6).

$$r^0\Delta x + r^1\Delta x + r^2\Delta x + r^3\Delta x + ... + r^{N-1} = L \tag{6.6}$$

Writing Eq. (6.6) in summation form, Eq. (6.7) is obtained. The summation is the finite geometric series and it can directly be written in terms of r and N as in the right hand side of Eq. (6.7).

$$\frac{L}{\Delta x} = \sum_{n=0}^{N-1} r^n = \frac{1-r^N}{1-r} \tag{6.7}$$

From Eq. (6.7), an equation for Δx is written as follows.

$$\Delta x = L\frac{1-r}{1-r^N} \tag{6.8}$$

Since Eq. (6.8) gives the distance between the starting boundary and the end of the control volume (right end of control volume in Figure 6.11), Eq. (6.8) can be generalized to give the distance of the end of the ith control volume to the starting boundary, L_i. The non-uniform grid setup is made with Eq. (6.9).

$$L_i = L\frac{1-r^i}{1-r^N} \tag{6.9}$$

The 3-D grid arrangement using Eq. (6.9) is shown in Figure 6.12. The location z=0 m is the end of the step and y=0.01 m is the corner of the step. As can be seen from the magnified portion of the grid in Figure 6.12, the size of control volumes are bigger in z-direction from inlet to the edge of the step since before the edge of the step, the flow is coming as fully developed and there is nothing special to investigate before the edge of the step.

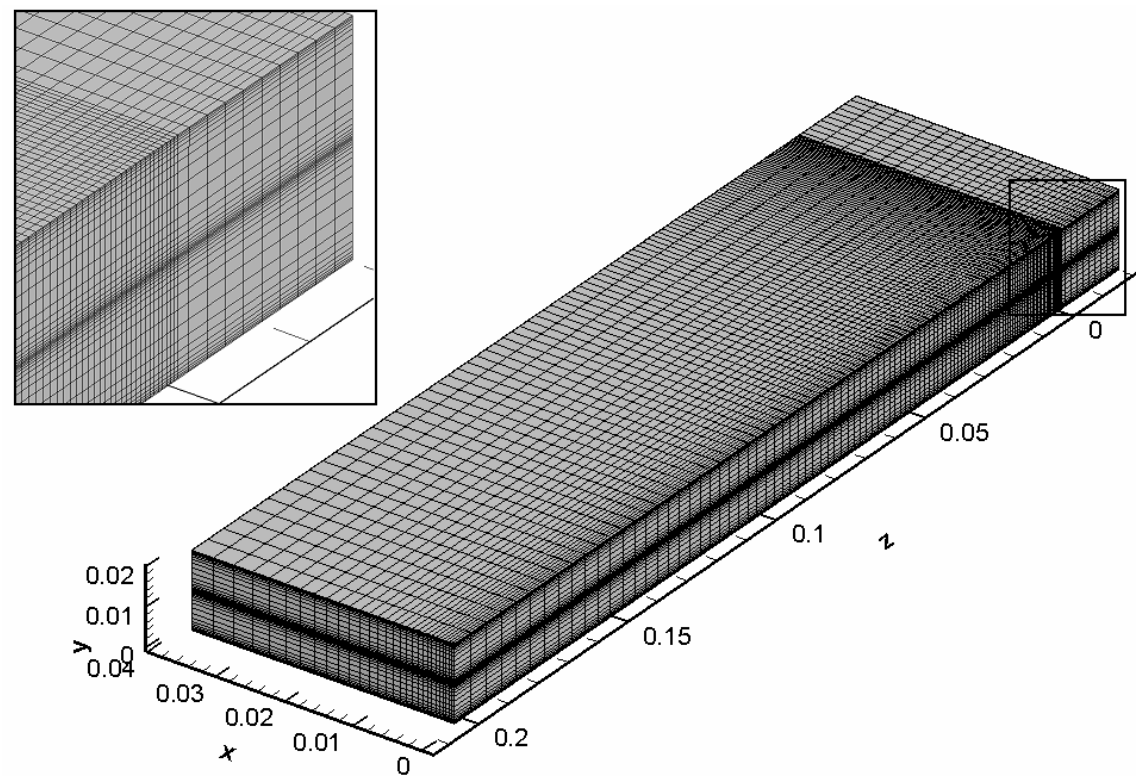

Figure 6.12. Grid arrangement of backward facing step flow geometry.

6.3.4. Results and Discussion

The simulations performed here are for Re=343 and most of the researchers have used the experimental data of Armaly *et al*. for computational validation studies. The target quantity for the validation is the reattachment length for 3-D flow. For 2-D simulation there is only one value for the reattachment point but for 3-D case, the reattachment points form a line called "z_w-line" on the wall. This is the location where the stream-wise component of the wall shear stress ($m\partial w/\partial y|_{y=0}$) is equal to zero and at this location the stream-wise velocity component changes sign from positive to negative. For the grid independence study, the z_w-line is chosen as target quantity.

Figure 6.13 shows the comparison for the so-called z_w-line for both uniform and non-uniform grid arrangements. Although the control volume number for both of the cases is equal, the uniform grid cannot catch the flow details. Fortunately, there is a perfect agreement for non-uniform grid arrangement with experimental data of Armaly *et al.* (2003) and this validates the present numerical study.

The plot in Figure 6.13 is obtained after investigating 17 different grid arrangements of which constitutes the grid independence study.

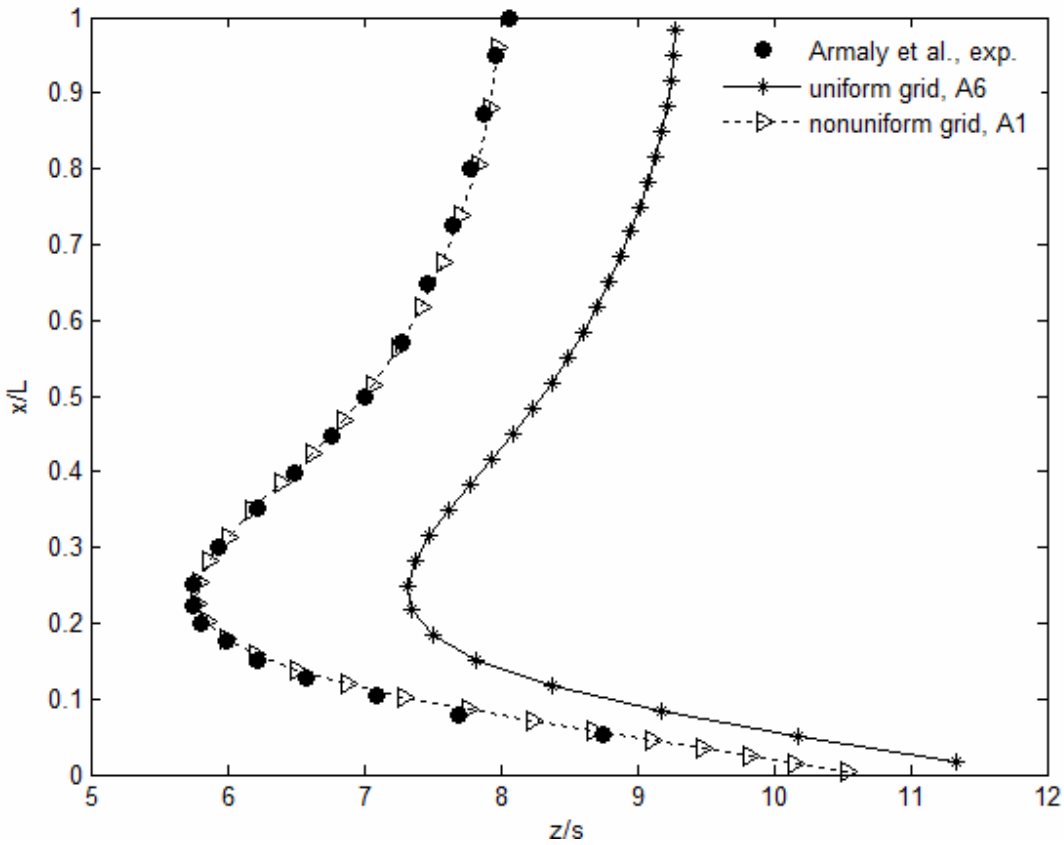

Figure 6.13. Comparison of uniform and non-uniform grid on the z_w-line.

Firstly, the effect of number of control volumes N_x, N_y and N_z in x, y and z-directions are investigated. Table 6.4 gives the details of different grid densities for the comparison of effect of N_x.

Table 6.4. Grid details of the different grid sets for the effect of N_x.

Grid	N_x, N_y, N_z	Δx_{min}	Δx_{max}
Grid C1	20, 40, 150	8.7409e-04	0.0038
Grid A1	30, 40, 150	3.5310e-04	0.0033
Grid C2	40, 40, 150	1.5441e-05	0.0031
Grid C3	50, 40, 150	6.9714e-05	0.0030

The effect of N_x on target quantity z_w-line is compared in Figure 6.14. As shown in the magnified plot, the grid with 20 control volumes in x-direction cannot catch the correct location of zero wall shear stress at a location around z/s takes its minimum value. For Grid A1, C2 and C3, computed locations are very close to the experiment and since Grid A1 can catch the detail with less number of control volumes in x-direction than Grid C2 and C3, N_x is chosen to be 30 in x-direction.

After determining the number of control volumes in x-direction, the next comparison is made on the effect of number of control volume in y-direction, N_y. The details of the grids comparing the effect of N_y can be found in Table 6.5.

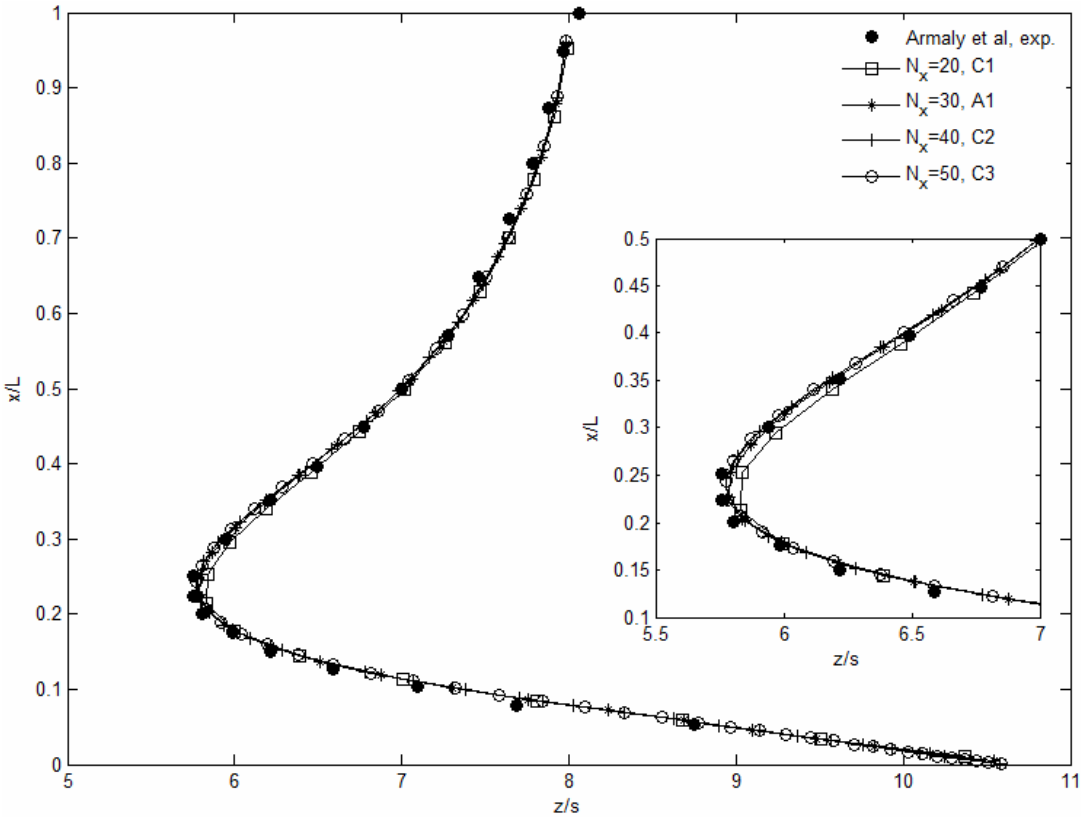

Figure 6.14. Effect of N_x on the z_w-line distribution.

Figure 6.15 shows the effect of Ny on the z_w-line for different grid densities in y-direction. There is a shift to the left of the z_w-line for the results of grids B1 and B2. However, The results for grids A1 and B3 matches very well with the experimental data and there is not so much difference between the results of Grid A1 and B3, therefore the

number of control volumes in y-direction is chosen as 40 so as to decrease computational memory compared for a number of control volumes of 60 for grid B3.

Table 6.5. Grid details of the different grid sets for the effect of N_y.

Grid	N_x, N_y, N_z	Δy_{min}	Δy_{max}
Grid B1	30, 20, 150	5.0229e-04	0.0017
Grid B2	30, 28, 150	2.4400e-04	0.0015
Grid A1	30, 40, 150	9.1592e-05	0.0014
Grid B3	30, 60, 150	1.9628e-05	0.0013

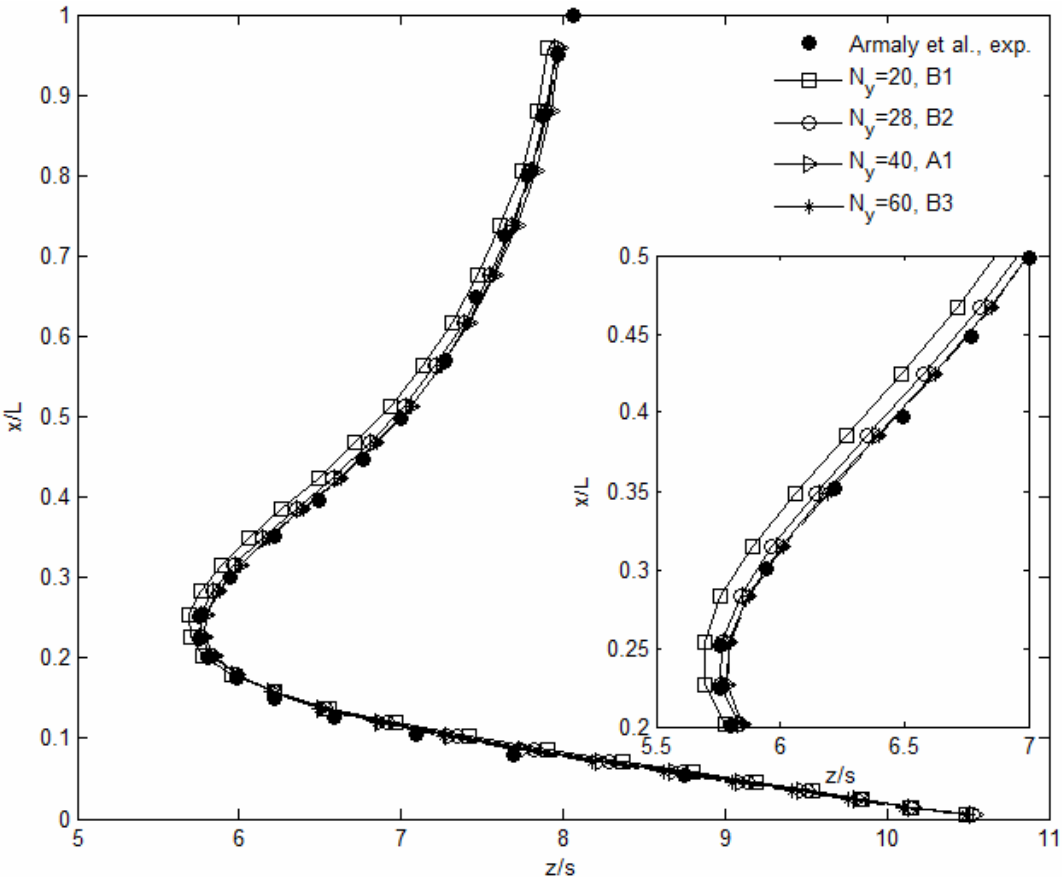

Figure 6.15. Effect of N_y on the z_w-line distribution.

The effect of number of control volume in z-direction is compared with three different grid densities in z-direction and details of them are given in Table 6.6. For this comparison, the number of control volumes in x and y directions are fixed to 30 and 40, but the effect of Ny is investigated. The results are shown in Figure 6.16. While grid D1 cannot catch the z_w-line, the grids A1 and D2 matches very well. Since the results for grids A1 and D2 are very close to each other, the less number of control volume which is 150 in z-direction is chosen.

Table 6.6. Grid details of the different grid sets for the effect of N_z.

Grid	N_x, N_y, N_z	Δz_{min}	Δz_{max}
Grid D1	30, 40, 100	1.5000e-03	0.0130
Grid A1	30, 40, 150	4.0689e-04	0.0126
Grid D2	30, 40, 180	1.9074e-04	0.0124

After determining the number of control volumes in x, y and z-directions, the expansion factors for non-uniform grid arrangement should be determined. To choose the factors, the effects of different expansion factors for the three directions are compared investigating the effect of the expansion factors on the target quantity.

Table 6.7 and 6.8 give the details of the different grid densities arising from the different expansion factor combinations. While an expansion factor very close to 1 means a grid like a uniform grid, an expansion factor close 1.5 creates a grid arrangement very concentrated near the boundary, in other word as the expansion factor increases, a dense grid is obtained near boundaries but a coarse grid is created far from the boundary. Therefore, a suitable combination of expansion factors should be chosen such that it is dense enough near boundaries and not too much coarse far from the boundary, but this is not the only factor affecting the choice; the target quantity should be match with the experiment as well.

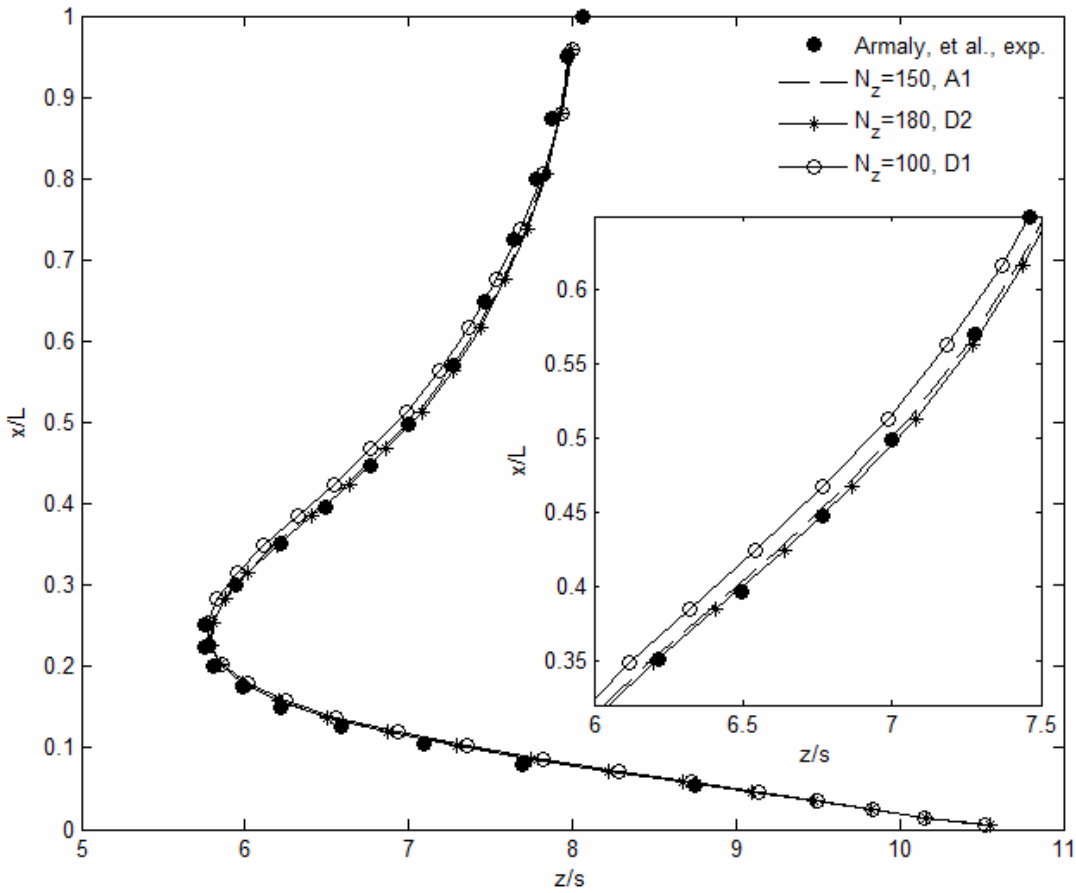

Figure 6.16. Effect of N_z on the z_w-line distribution.

Table 6. 7. Expansion factors of the different grid sets for N_x=30, N_y=40, N_z=150.

Grid	r_x	r_y	r_z
Grid A1	1.08	1.35	1.025
Grid A2	1.08	1.35	1.035
Grid A3	1.08	1.45	1.025
Grid A4	1.15	1.35	1.025
Grid A5	1.20	1.35	1.025
Grid A6	1.00	1.00	1.000
Grid A7	1.08	1.05	1.025
Grid A8	1.08	1.25	1.025

Table 6.8. Minimum and maximum dimensions of control volumes for Grid A.

Grid	Δx_{min}	Δx_{max}	Δy_{min}	Δy_{max}	Δz_{min}	Δz_{max}
Grid A1	3.5310e-04	0.0033	9.1592e-05	0.0014	4.0689e-04	0.0126
Grid A2	3.5310e-04	0.0033	9.1592e-05	0.0014	1.4286e-04	0.0170
Grid A3	3.5310e-04	0.0033	5.6131e-05	0.0016	4.0689e-04	0.0126
Grid A4	9.2008e-05	0.0053	9.1592e-05	0.0014	4.0689e-04	0.0126
Grid A5	3.3844e-05	0.0067	9.1592e-05	0.0014	4.0689e-04	0.0126
Grid A6	0.00133	0.00133	0.0005	0.0005	0.00346	0.00346
Grid A7	3.5310e-04	0.0033	3.9752e-04	6.1669e-04	4.0689e-04	0.0126
Grid A8	3.5310e-04	0.0033	1.5036e-04	0.0011	4.0689e-04	0.0126

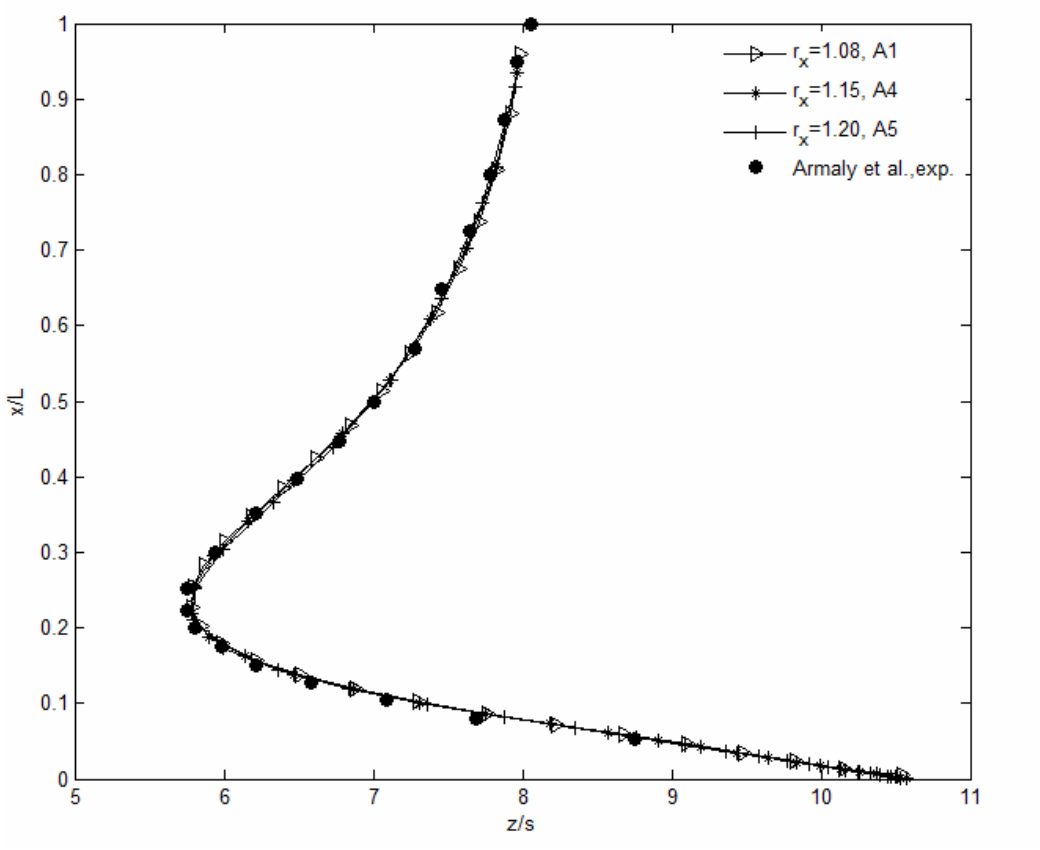

Figure 6.17. Effect of r_x on the z_w-line distribution.

The effect of different expansion factors in x-direction is showed investigating grids A1, A4 and A5 in Figure 6.17. As can be deduced out from this plot, there is no so much difference for these grid densities, therefore one of the three can be chosen, but the smallest one is chosen because it creates grid closer to a uniform grid preventing any effect of aspect ratio differences in x-direction for all of the control volumes.

The comparison for the effect of expansion factor in y-direction is made using the grids A1, A3, A7 and A8. The expansion factor for grid A7 is 1.05 and very close to 1, therefore it cannot use the advantage of non-uniform grid as much as others. The difference between grid A7 and experimental data can be seen in Figure 6.18. The expansion factor for grid A3 is 1.45, which is found as big for this run as shown in figure. However, the results of grids A1 and A8 are very close to each other. The choice of expansion factor between the two is made based on the magnified view of the plot near symmetry plane. Since the results for grid A1 matches at the center better than grid A8, the expansion factor is determined as 1.35 in y-direction.

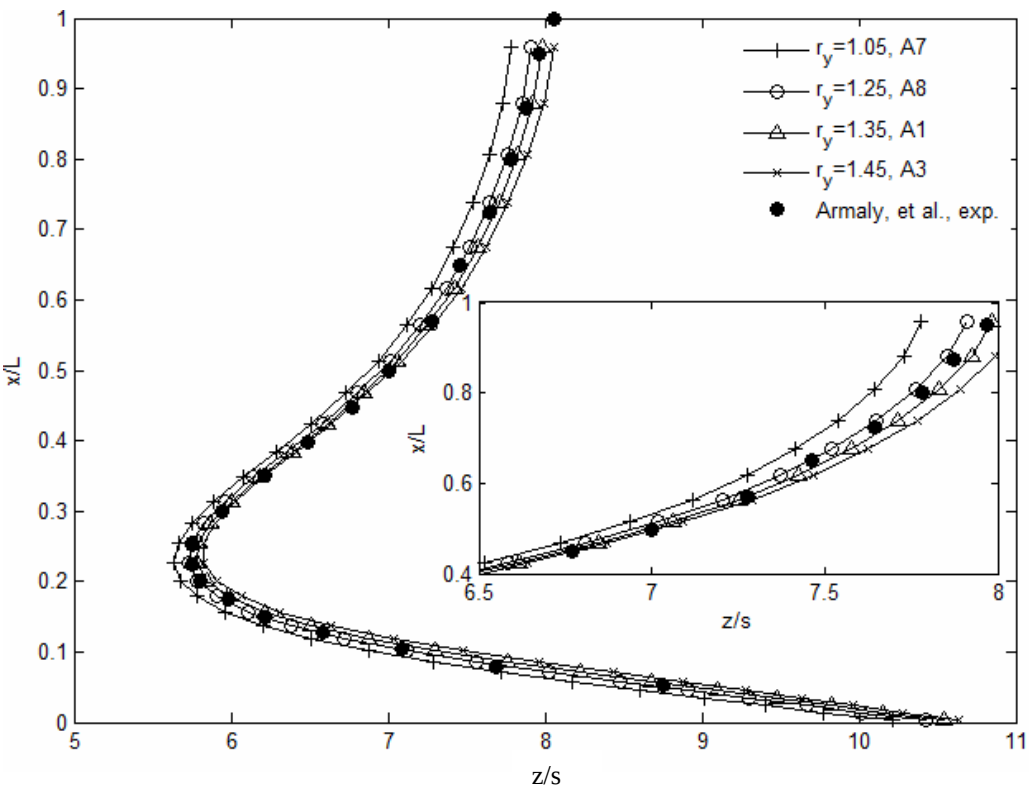

Figure 6.18. Effect of r_y on the z_w-line distribution.

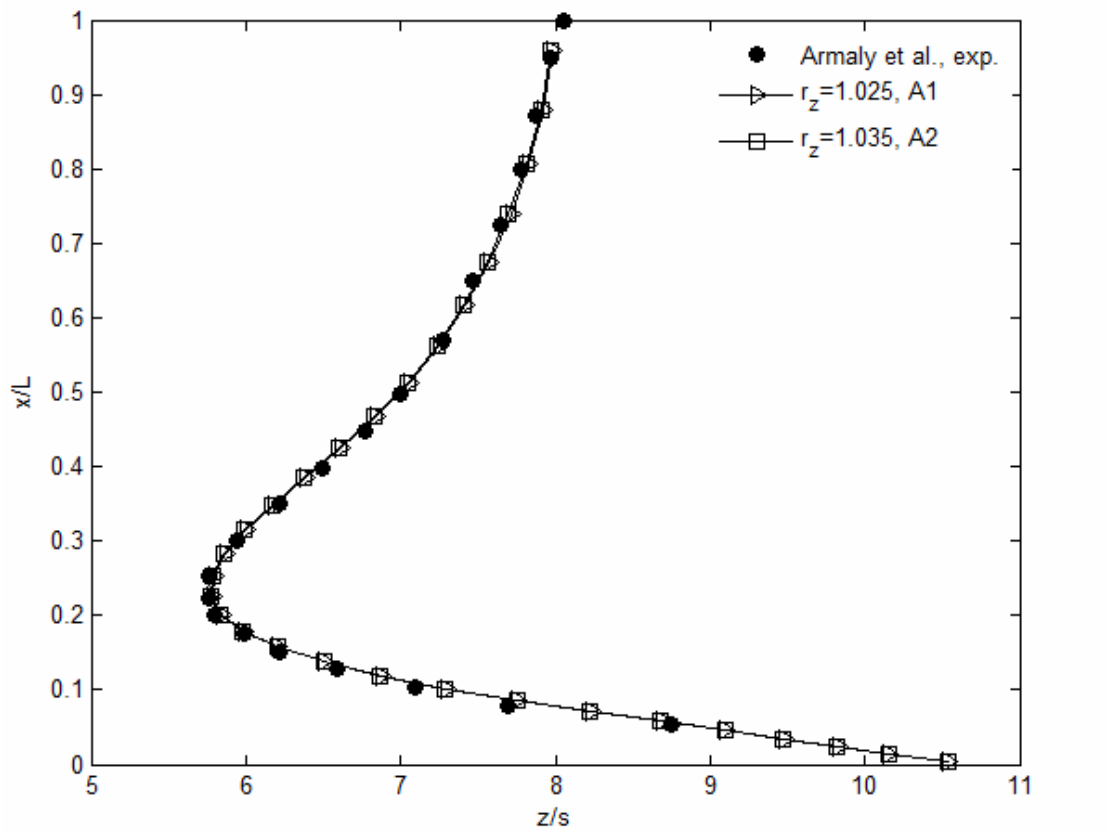

Figure 6.19. Effect of r_z on the z_w-line distribution.

The final comparison is made to determine the expansion factor in z-direction which is the flow direction and the longest dimension. For this comparison, grids A1 and A2 are compared and the results are shown in Figure 6.19. Since there is no so much difference between the two grid arrangements, the smaller expansion factor which is 1.025 chosen for z-direction for the same reason stated for x-direction expansion factor.

Completing the decision for the expansion factor in z-direction, the grid independence study is over and the other flow structures in the backward facing step flow are investigated with Grid A1. At this point it should be stated that Grid A1 uses 1.8×10^5 grid points. If the results of this grid arrangement is tried to reach with uniform grid arrangement, 3.15×10^7 grid points should be used to catch the same details of the smallest control volumes size of the non-uniform grid at each direction.

Figure 6.20 shows the streamwise velocity (w) profiles at different z-locations on symmetry plane. The present study agrees very well with the study of Saldana & Anand (2004). The location z/s=-2 is the inlet and the velocity profile is fully developed

there. The recirculation zone can be identified clearly at location z/s=5. The location z/s=0 is the edge of the step and up to y/s=1 is the wall of the step on which the velocity component should be zero. The present study agrees very well with this boundary condition such that the velocity is zero on the vertical wall of the step. This also proves the correctness of the artificial viscosity method applied for this solution. The location z/s=25 is the mid section of the channel. Since the flow has passed the recirculation zone, it starts to redevelop throughout the outlet of the channel.

Streamwise distributions of w-velocity component on different y-planes are given in Figure 6.21 and the numerical computations agree with the experimental data of Armaly *et al.* (2003). From the velocity distributions on figure 6.21, it can be deduced that there is velocity peak at around x/L=0.25 at y-planes below the step. This peak is seen for the y-planes equal to y=s and below of it. There is no such a peak for the locations above the corner of the step. These peaks show that there is a jet-like flow adjacent to the side wall and this type of flow is the reason of the minimum that appears on the z_w-line in Figure 6.14. At location y/s=1.6, there is a reverse flow for x/L<0.1.

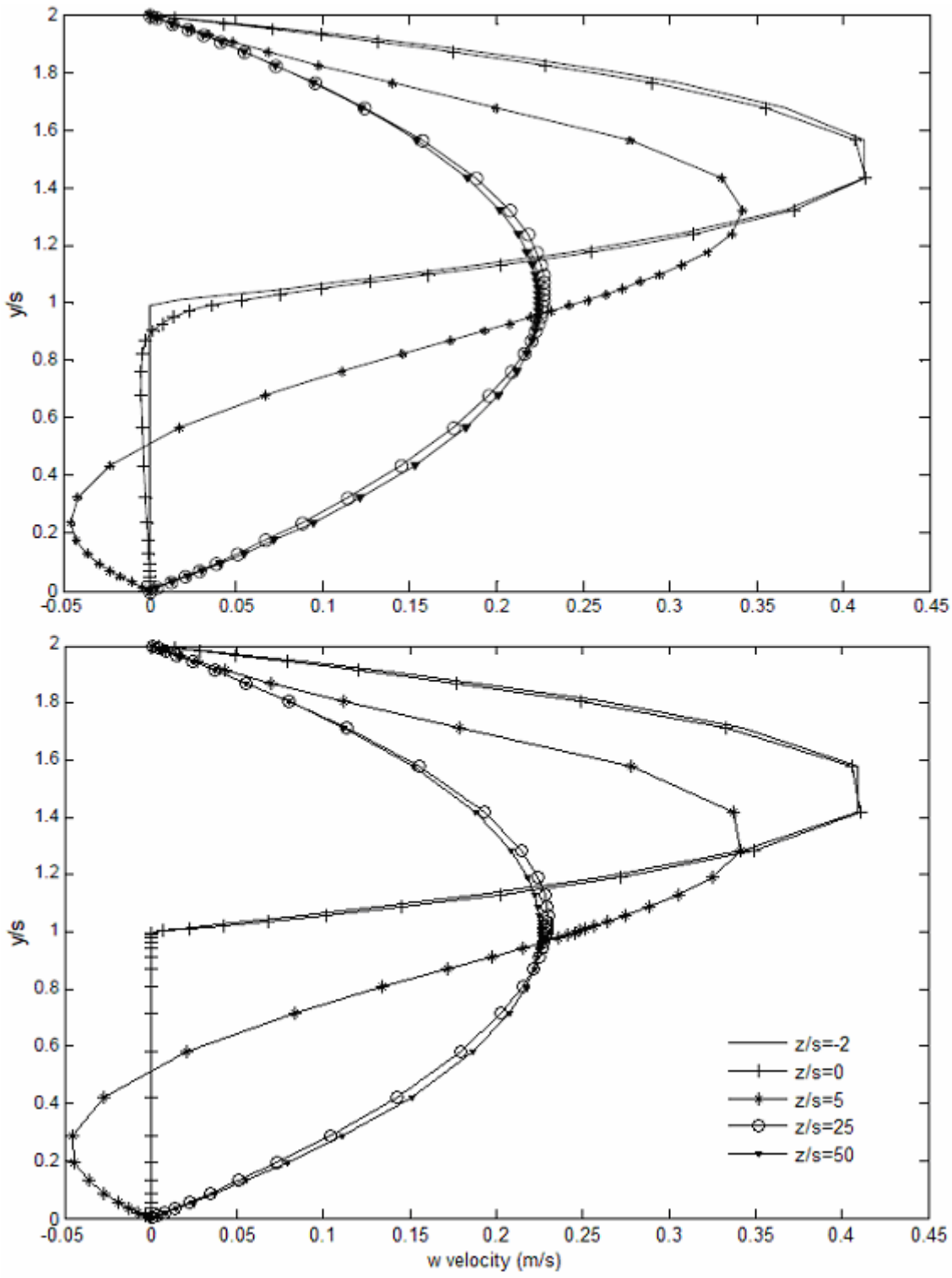

Figure 6. 20. Stream-wise velocity component at different z planes on the symmetry plane, (top): Saldana & Anand (2004), (bottom): present study.

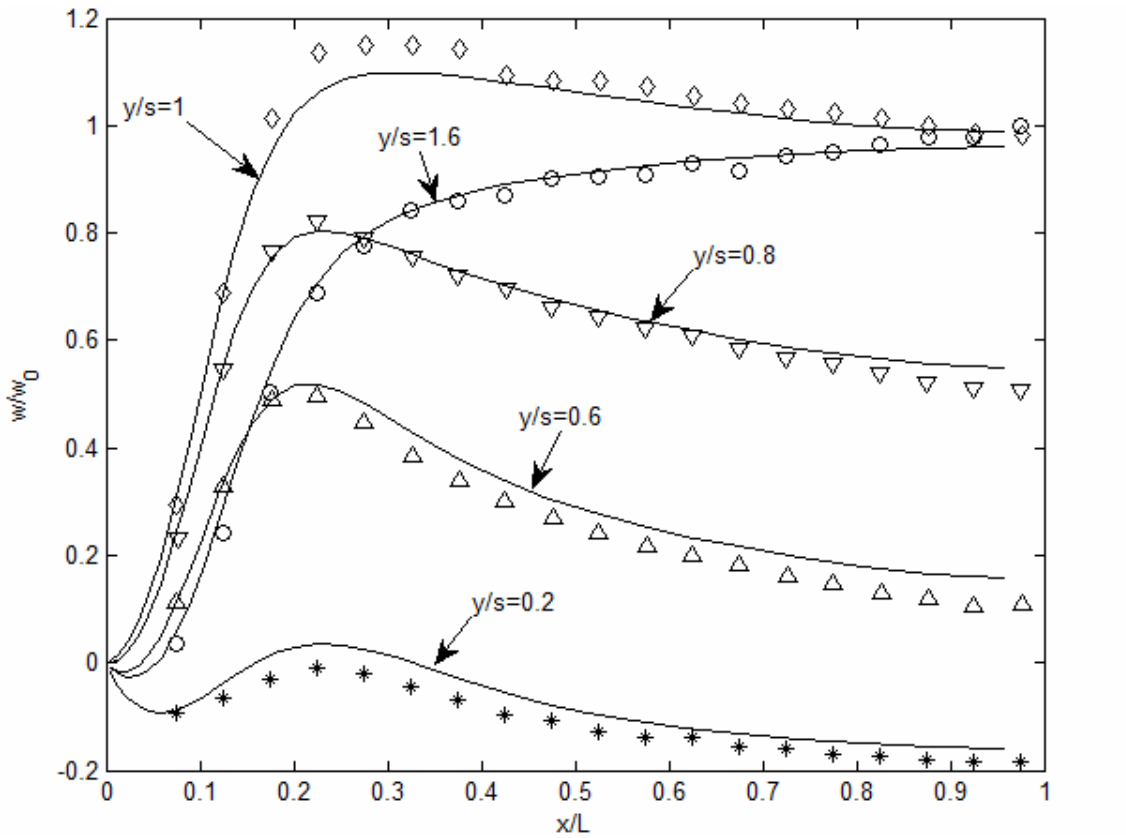

Figure 6.21. Span-wise distribution of w-velocity components on different y-planes, Lines: present study, Symbols: Armaly *et al.* (2003).

Streamlines are the lines that are instantaneously tangent everywhere to the velocity vector of the flow. These lines show the direction a fluid element will travel at any point in any time for steady flows. Figure 6.22 shows a few representative 3D streamlines representing the flow path of particles with no mass. The streamlines start from the inlet plane and they take different forms downstream of the step. Streamlines A and B impinges on the bottom wall and this "jet-like" flow structure is a significant feature of 3-D backward facing step flow. After the impingement, streamline B continues its flow redeveloping downstream of the flow while the streamline A moves upwards, goes back and return to the flow direction back forming a swirling flow region near the side wall. This "jet-like" flow is the reason of the maximum and minimum on the z_w-line. The particles on streamline D move into the primary recirculation zone directly and they make a swirling motion increasing in size to the center of the channel (symmetry plane) and then join to the flow downstream of the channel. There are 2 more streamlines adjacent to streamline C. While streamline C joins to the recirculation after impingement, the two streamlines adjacent to streamline C join to the redeveloping

flow. Figure 6.23 shows three different rake of streamlines. Though the starting positions of the streamlines are close to one another, the flow patterns are very different. Especially, streamlines closer to the side wall move rapidly inside the primary recirculation zone.

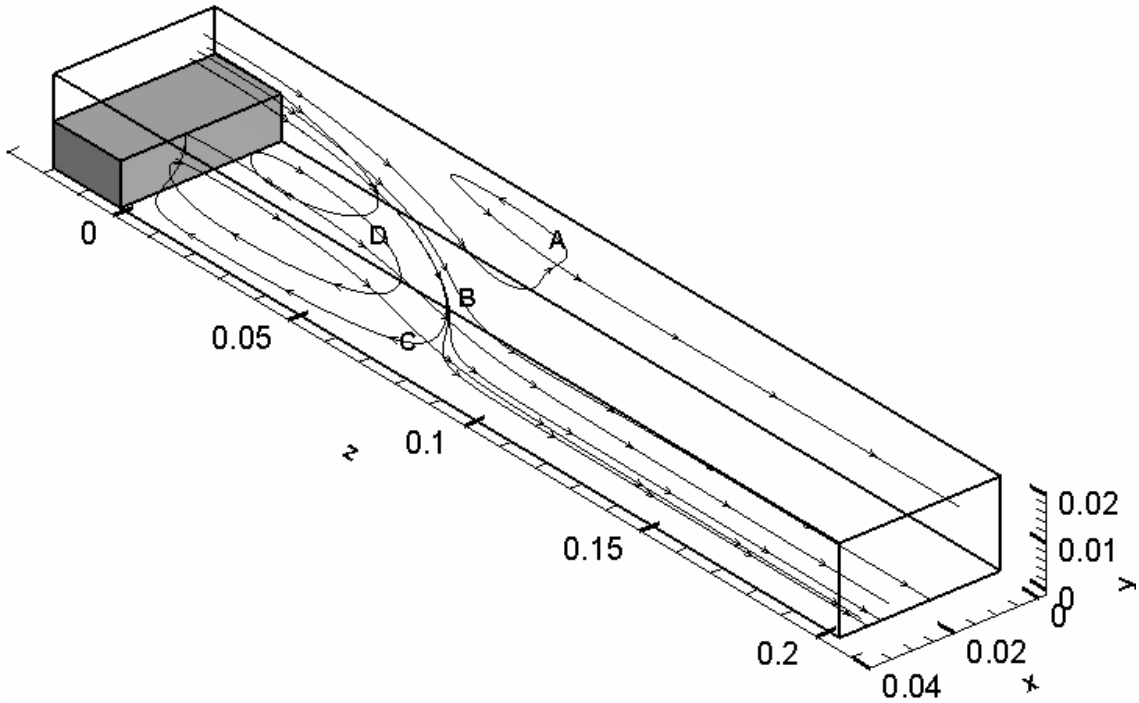

Figure 6.22. Streamlines showing the flow structure behind the step.

The primary recirculation bubble can be seen in the stream-wise velocity contours given in Figure 6.24. The recirculation seen in streamline A in figure 6.22 can be seen in the w-velocity contour at x=0.0025 m and the shape of z_w-line is very clear. The location of reattachment point starting from center decreases to its minimum at location x=0.01 m corresponding to x/L=0.25 in Figure 6.13 and then increases near the side wall taking its maximum. The profile of the recirculation bubble in span-wise direction can be seen in Figure 6.25. The blue colored contours show the reverse flow. The recirculation region ends before z=0.1 m as seen in both Figures 6.24 and 6.25. After that location, flow redevelops throughout the outlet plane.

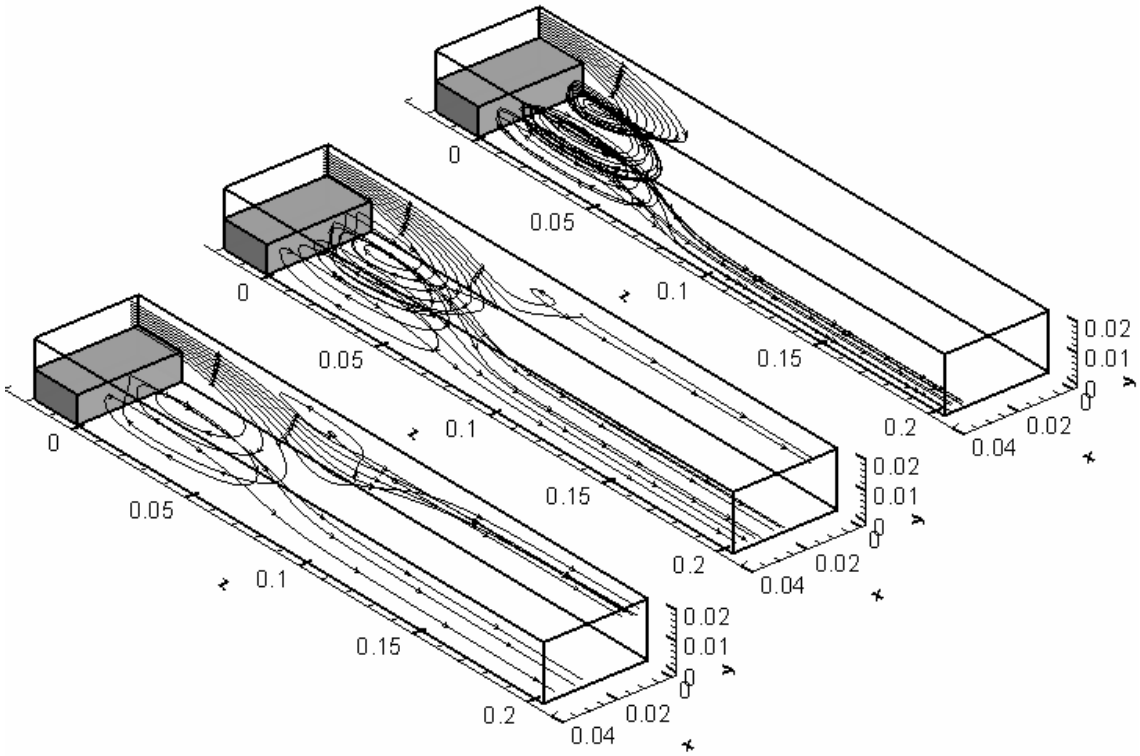

Figure 6.23. Rake of streamlines starting from x=0.001 m, x=0.002 m and x=0.003 m from top to bottom.

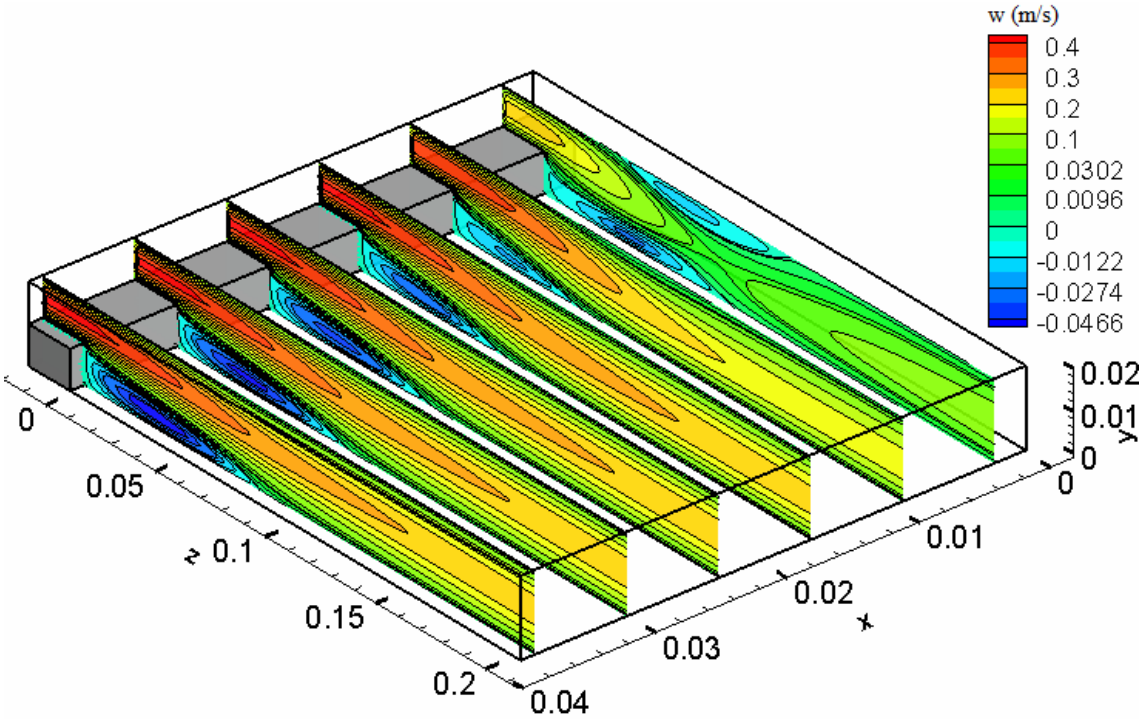

Figure 6.24. Streamwise velocity contours on different y-z planes.

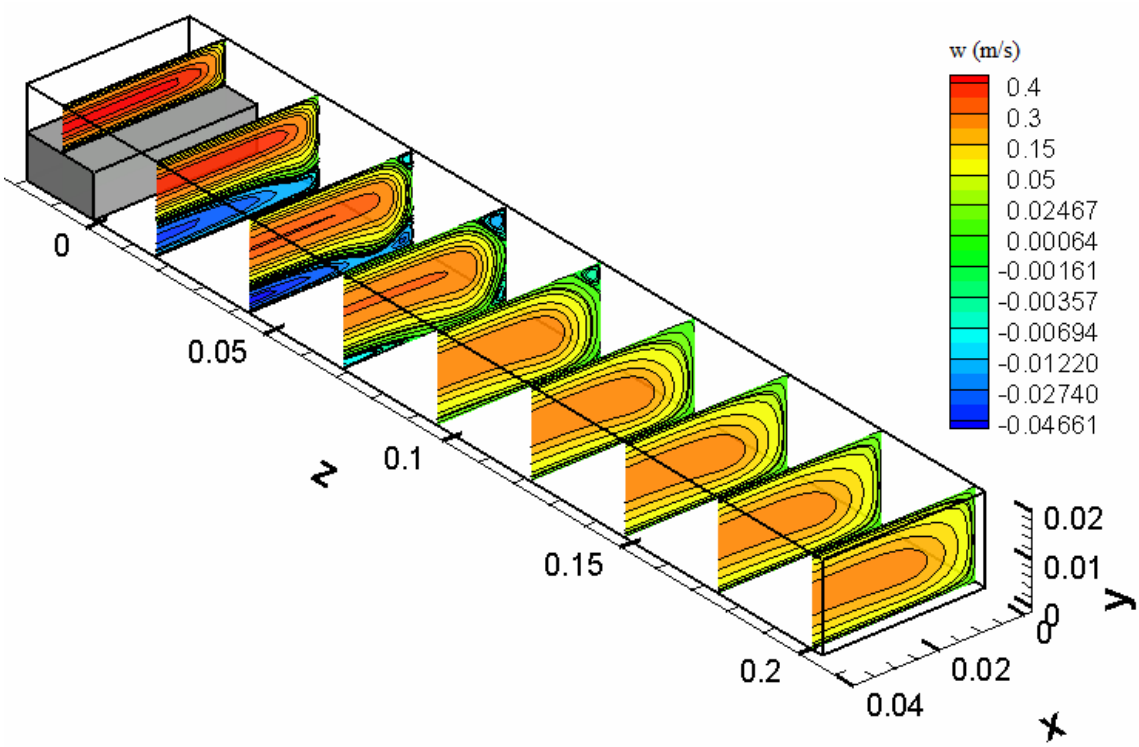

Figure 6.25. Axial velocity contours at different x-y planes.

The separating flow at the corner of the step reattaches at the bottom wall forming the primary recirculation zone. The swirling flow in the span-wise direction and the other recirculation zones in the channel are the cause of adverse pressure gradient developing in the flow due to the sudden expansion at the step. Figure 6.26 and 6.27 show pressure distribution on several planes. The high pressure seen at location around z=0.1 m is due to the impinging flow at that region. The flow direction is affected from a few parameters. These are pressure gradient, inertial forces and viscous forces. If the pressure is enough to overcome the effects of other forces, some of the flow goes back and some continue to its way in the redeveloping region of the downstream of the channel as seen in the very high pressure regions near the bottom wall. However, although the pressure values are high at location around z=0.1 m from bottom to top wall, there are some regions where the flow does not change its direction because at these locations inertial and viscous forces dominate over high pressure there.

The vorticity lines which are the lines tangent to the vorticity vectors ($\vec{\nabla} x \vec{u}$) for three different rakes are shown in figure 6.28. Using right-hand rule, the direction of the

flow can be deduced looking at the vorticity lines. The lines just outside the recirculation bubble show the direction of the flow to out, i.e., the flow is in the axial direction; however the vorticity lines inside the bubble show that the flow direction near bottom wall is to inside, i.e., the flow direction is reverse.

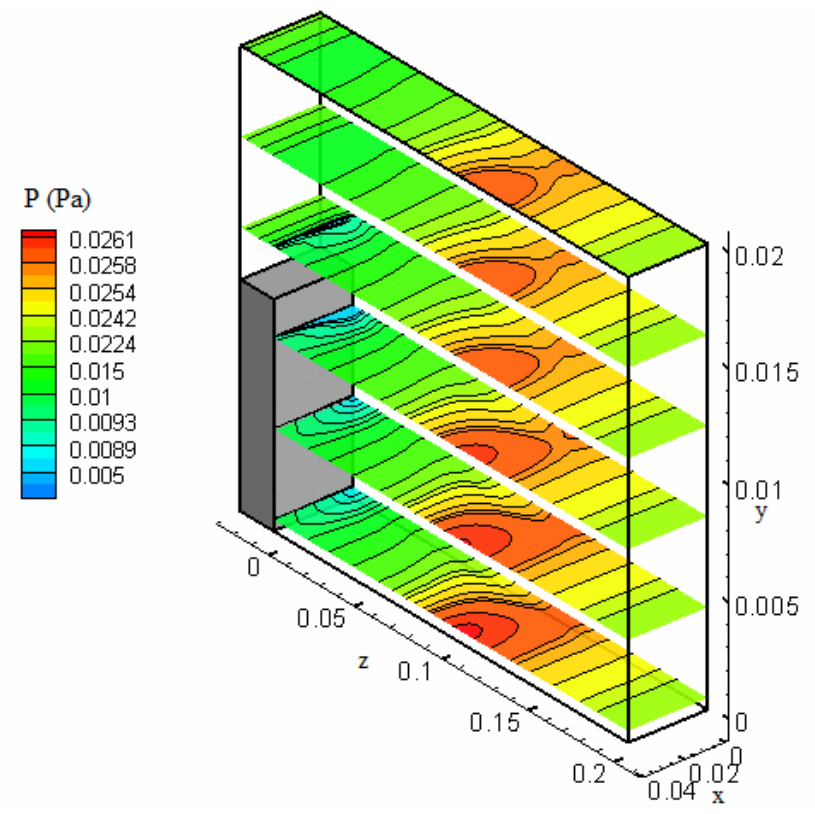

Figure 6.26. Pressure distributions on x-z planes.

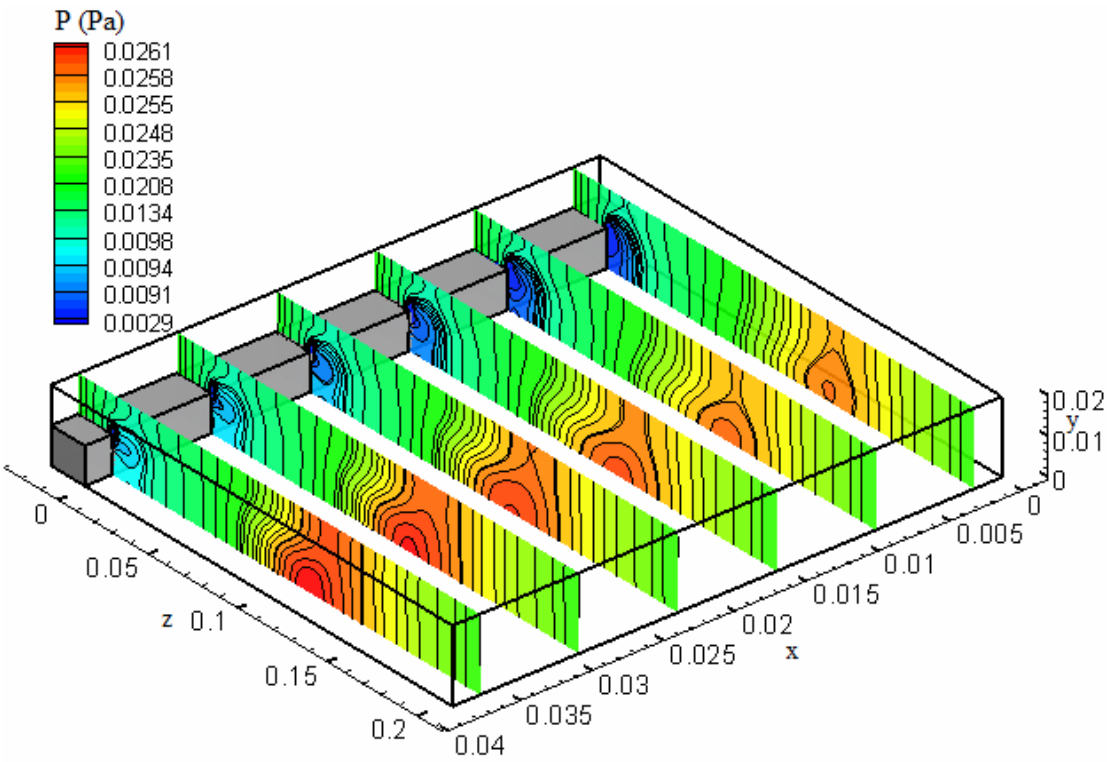

Figure 6.27. Pressure distributions on y-z planes.

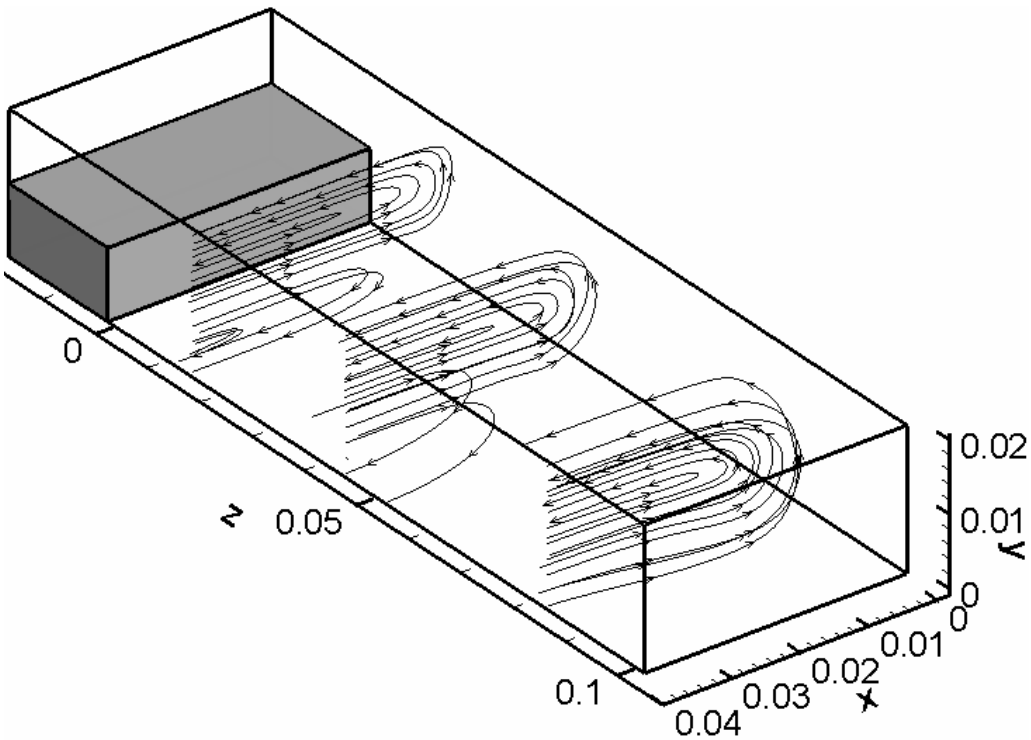

Figure 6.28. Vorticity lines inside and just outside the recirculation bubble.

The final investigation is made on the effects of expansion ratio (ER) on the flow structure. The simulations are made using four different ER with a constant Reynolds number of 343 which is achieved by setting the same average inlet velocity and total channel height for all of the cases. Table 6.9 gives the details of the step height (s), upstream channel length (h), ER and AR values for different geometries. The effect of ER on the primary reattachment length is shown in Figure 6.29 in which the current computational results are compared with the numerical results of Nie *et al.* (2002) for ER 2.5, 2.0, 1.67 and the results match very well. The axial velocity contours for different ER values are shown in Figure 6.30 where the difference in the reattachment length can be seen easily. From both of the figures, it is easily deduced that as ER increases, the primary reattachment length increases as well.

Table 6.9. Details of the geometries for different expansion ratios.

Geometry	s (m)	h (m)	ER	AR
1	0.004	0.016	1.25	20
2	0.008	0.012	1.67	10
3	0.010	0.010	2.00	8.00
4	0.012	0.008	2.50	6.67

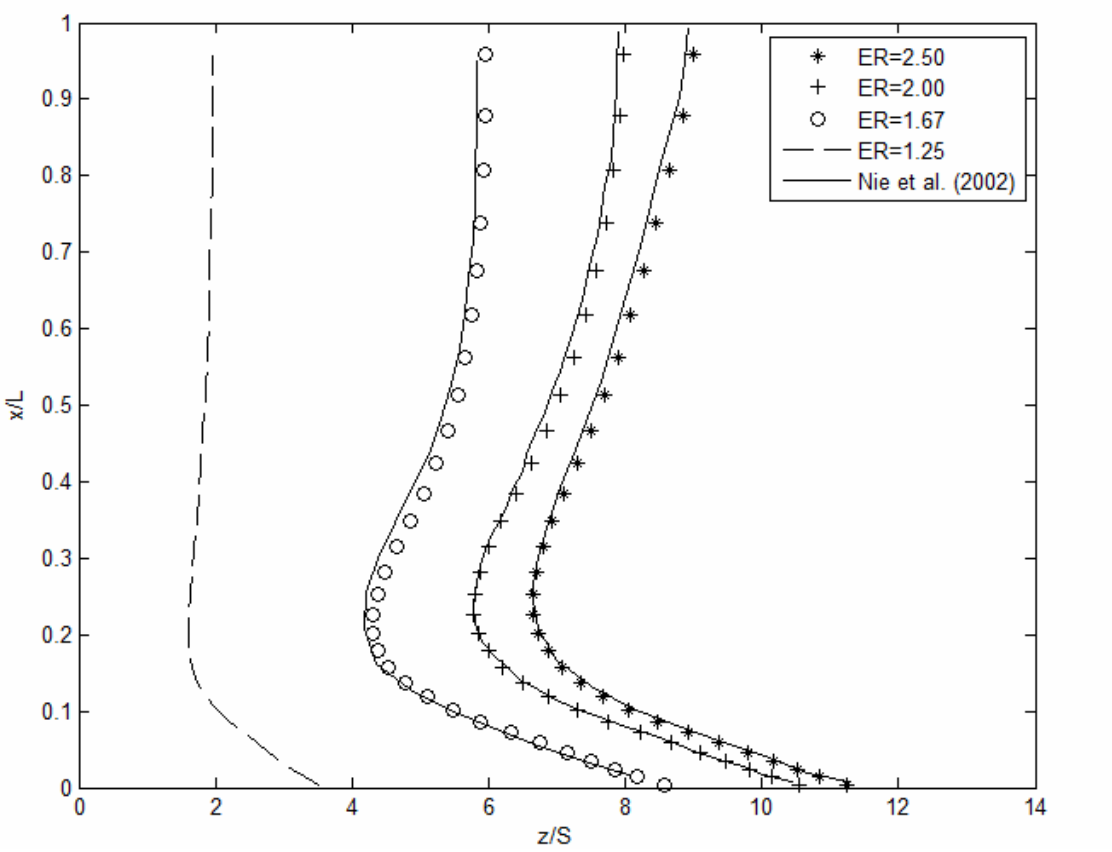

Figure 6. 29. Effect of ER on z_w-line.

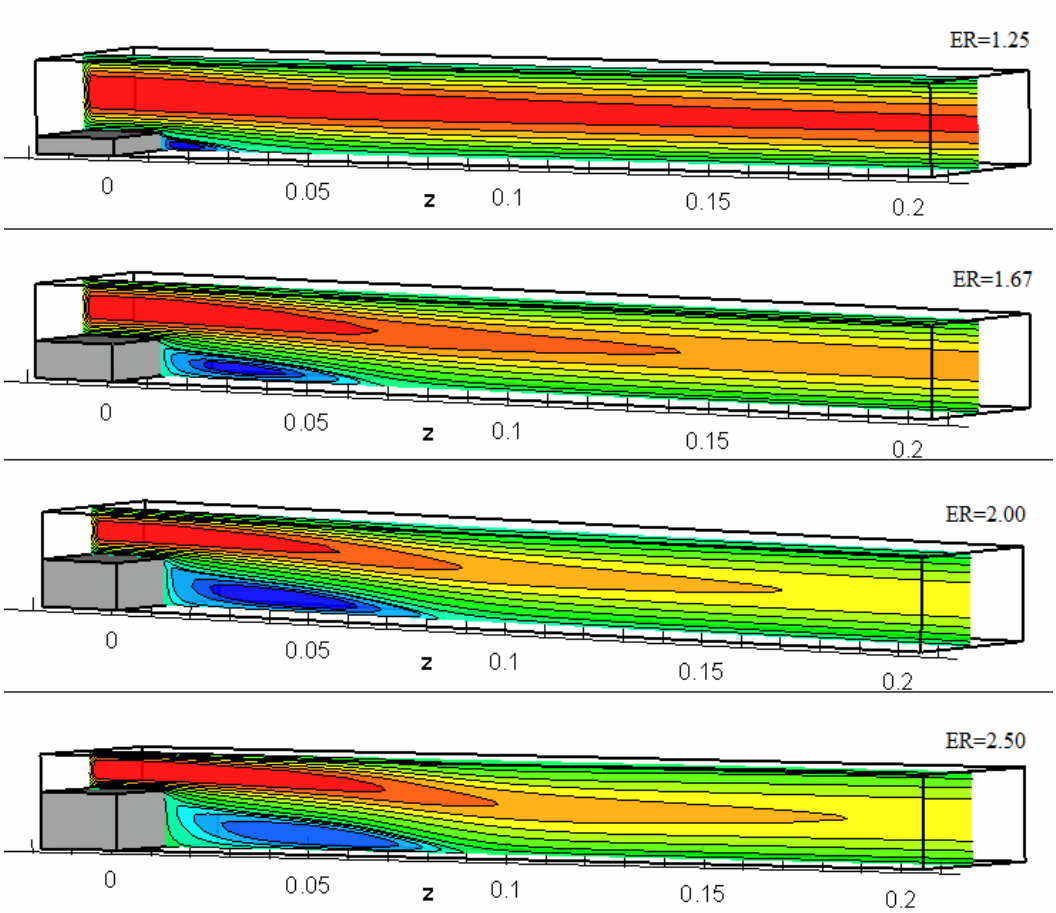

Figure 6.30. Axial velocity contours at x=0.02 m for different ERs.

To show the flow structure behind the step, some streamlines starting from inlet are presented in Figure 6.31. When ER=1.25, there is no reverse flow adjacent to the side wall, however there exist for the others. As ER increases, the size of the recirculation bubble increases, the location of the impingement of “jet-like” flow on the bottom wall moves further downstream of the step. For ER bigger than 1.25, the impinging flow rebounds and moves upward and the location of the impingement on the top wall is also moves further downstream as ER increases.

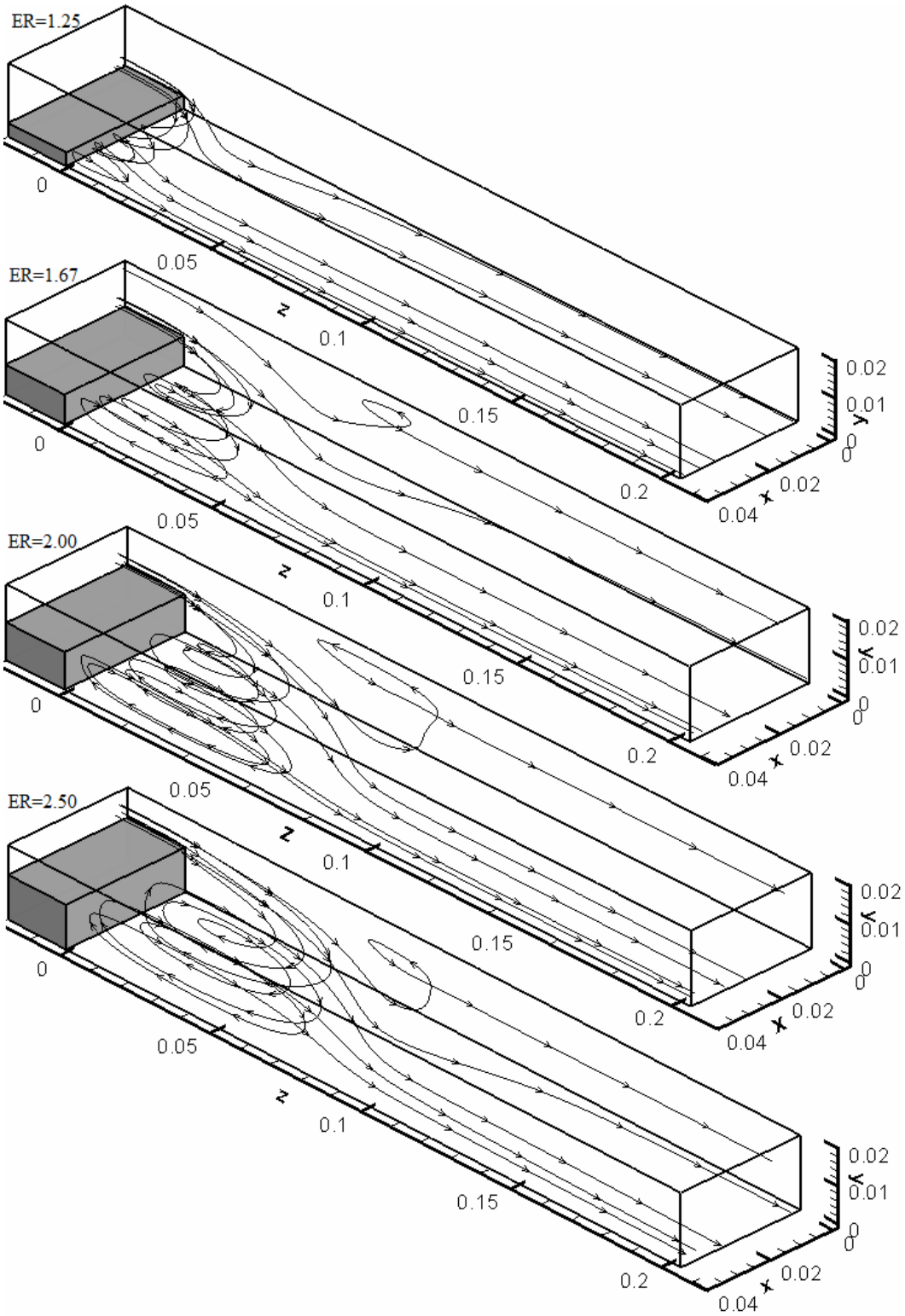

Figure 6.31. Streamlines showing the flow structure for different ERs.

Due to sudden expansion, there is a pressure recovery in channel with the decrease in mean velocity and it is represented by a pressure recovery coefficient, C_P, which is the ratio of static pressure rise to inlet dynamic pressure (Eq. (6.10)).

$$C_P \equiv \frac{p_2 - p_1}{\frac{1}{2} r w_0^2} \tag{6.10}$$

The sudden change in geometry increases the loss of recovery and the expansion loss coefficient, K_e, is defined as given in Eq. (6.11).

$$K_e = \left(1 - \frac{1}{ER^2}\right) - C_p \tag{6.11}$$

The term in parenthesis in Eq. (6.11) represents the ideal (frictionless) pressure recovery coefficient; therefore the loss is the difference between ideal and actual values. The data of the change of the expansion loss coefficient with respect to ER for a 2-D sudden expansion flow is given by Fox et al. (2003). Although the problems are not the same, they are very similar to each other. The calculations for present study are made at the symmetry plane and the results are compared in Figure 6.32. The present results are consistent with the data given by Fox, therefore it can be stated that, as expansion ratio increases, the expansion loss coefficient increases as well.

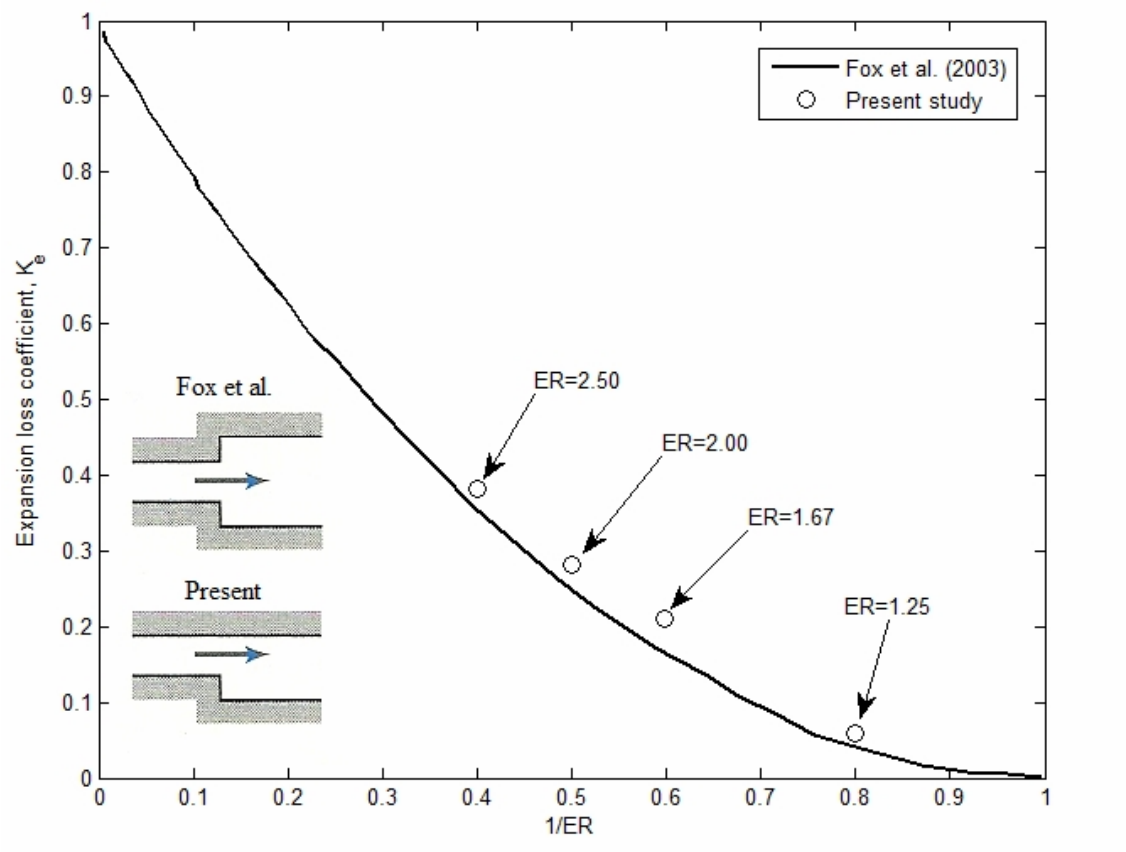

Figure 6.32. Expansion loss coefficient for flow through sudden area changes.

CHAPTER 7

CONCLUSION

The development of a 3-D laminar Navier Stokes solver and the validation of it were presented in this study. The solver has been designed in such a way that it can deal with separated channel flows by means of artificial viscosity method. The development of this solver involves starting from a scratch and dealing with all the subroutines of the solver and gives the chance of understanding all the details of the flow solver. Since the commercial versions of this type of programs are "black boxes", this study has revealed the important facts inside that black box. That the developed solver being three-dimensional makes the development a bit complex but it gives the chance of examining actual flows in 3-D.

The textbooks and courses on the CFD area tend to be theoretical and the simplest examples such as 1-D and 2-D conduction and convection problems are given. The details of 3-D cases are always overlooked. In articles, on the other hand, the authors assume that the reader has the basic knowledge and they do not give the details of the numerical application. They discuss mostly on the computational results obtained by their code. With this study, we have tried to close these gaps.

The solver code is developed with C++ language and it utilizes one main routine and twenty one subroutines. All the routines are responsible from the calculation of a part of the solution algorithm. The resulting linear algebraic equations are solved via iterative methods and it is terminated when the convergence criteria is achieved. Finite volume method is applied throughout this study and the details of discretization methods, boundary conditions' implementation and the solution algorithm were given in detail. We proposed a more generalized form of the coefficients of discretized momentum and pressure correction equations including the effects of boundaries.

The first validation of the code was made testing it for a developing channel flow to show that the code is capable of dealing with forced convection in a channel. Although the flow is simple, it is a valuable test case to just show that the code can handle three dimensional problems. The results agree very well with the analytical results given by Shah and London (1978) and Incorpera *et al.* (2007).

To augment the use of code, we applied the artificial viscosity method with which the code for conventional channel geometry is easily converted into geometrically different channel geometries. Since this method uses very high viscosity in the regions where an artificial solid region is created, the evaluation of the interface viscosity in 3-D case is given in detail which is missing in literature. To validate the use of artificial viscosity method, the results of a developing channel flow with filled bed of high viscosity region were compared against the normal case and the agreement was perfect validating the use of the method.

Another test problem was chosen to be more complex to show how the code handles complex flow structures in 3-D. This problem is a backward facing step flow. The reason behind the choice of this problem was the simplicity of the geometry, inherent complex flow structures of the separated flow from the edge of the step, to verify experimental data to validate the current code and to show the application of the artificial viscosity method in this relatively more complex flow. After grid independence study for 17 different grid density, the target quantity, z_w-line, was compared against the experimental data given by Armaly *et al*. (1983), Li (2001) and Armaly *et al*. (2003) to justify the use of our developed computer code. Results showed that there was generally excellent agreement between the present results and the experimental data. The investigation of the flow structure was made with 3-D streamlines, velocity and pressure contours, velocity profiles and vorticity lines. It was showed that complex three-dimensional flow develops behind the step with reverse and swirling flow regions. The "jet-like" flow and the impingement to the bottom wall were found to be responsible from the minimum on the z_w-line. The investigation of effects of expansion ratio (ER) on the flow structure revealed that, as the ER increases, the size of primary recirculation bubble increases, reattachment length moves downstream of the channel and because of the sudden expansion, the expansion loss coefficient increases as well.

All in all, this study has provided us with revealing the details of a 3-D flow solver and with the investigation of complex flow structures behind a backward facing step separated from the edge in 3-D. With the aid of artificial viscosity method, the conventional code written for a simple channel flow can handle very different geometries in a channel. Since all the subroutines of the code are known, it is easy to implement new variations on it.

For future studies with this solver;

- The grid generator subroutine should be improved to deal with unstructured grids.
- The adaptive grid refinement technique may be implemented in to the grid generator subroutine in order to save computational time and user does not have to have apriori knowledge of the flow during the grid generation.
- The iterative linear equation solver choices should be increased to decrease the run-times especially the pressure correction solver since it is the most time consuming part of the simulations.
- Parallel programming may be implemented to speed up the solvers. Although the Massage Passing Interface (MPI) and OpenMP were applied for parallelizing the code in this study, it requires deeper research for 3-D applications.
- Heat and chemical transport can be added.
- The code should be improved to solve unsteady problems.
- The code may be expanded to solve compressible, two phase and turbulent flows.

BIBLIOGRAPHY

Acharya, S. *et al*., 2007, *Pressure Based Finite Volume Methods in Computational Fluid Dynamics*, 129, Transactions of ASME.

Adams, E. W., Eaton, J. K., 1988. "*An LDA Study of the Backward-Facing Step Flow Including the Effects of Velocity Bias*", Journal of Fluids Engineering, 1988.

Armaly, B. F., Durst, F., Pereira, J. C. F., and Schonung, B., 1983, *Experimental and Theoretical Investigation of Backward-Facing Step Flow*, J. Fluid Mech., 127, pp. 473–496.

Armaly, B. F., Li, A., Nie, J. H., 2003, *Measurements in Three-Dimensional Laminar Separated Flow*, International Journal of Heat and Mass Transfer, 46, pp. 3573-3582.

Biswas, G., Breuer, M., Durst, F., 2004, *Backward-Facing Step Flows for Various Expansion Ratios at Low and Moderate Reynolds Numbers*, 126, Transactions of ASME.

Chiang, T.P., Sheu, T.W.H., and Tsai, S.F., 1997, *Topological Flow Structures in Backward-Facing Step Channels*, Computers and Fluids, 26(4), pp. 321-337.

Chiang, T. P., Sheu, T.W.H, 1997, *Vortical Flow Over a 3-D Backward-Facing Step*, Numerical Heat Transfer, 31:2, pp. 167-192.

Chiang, T. P, Sheu, T. W. H., 1999, *A Numerical Revisit of Backward-Facing Step Flow Problem*, Physics of fluids, 11(4), pp. 862-874.

Courant, R., Friedrichs, K. O., Lewy, H., 1928, *Über die Partiellen Differenzengleichungen der Mathematischen Physik*, Math. Ann., vol. 100, pp. 32-74.

Ferziger, J. H., Peric, M., 2001, *Computational Methods for Fluid Dynamics*, 3rd Edition, Springer.

Fessler, J. R., Eaton, J. K., 1997, *Particle Response in a Planar Sudden Expansion Flow*, Experimental Thermal and Fluid Science.

Fox, R. W., McDonald, A. T., Pritchard, P. J., 2003, *Introduction to fluid mechanics*, 6th Edition, Wiley.

Frank M. White, 2006,*Viscous Fluid Flow*, 3rd Edition, McGraw-Hill.

Gartling, D. K., 1990, *A Test Problem for Outflow Boundary Conditions - Flow Over a Backward-Facing Step*, International Journal for Numerical Methods in Fluids, 11, pp. 953-967.

Incropera, DeWitt, Bergman, Lavine, 2007, *Fundamentals of Heat and Mass Transfer*, 6th Edition, Wiley.

Iwai, H., Nakabe, K., and Suzuki, K., 2000, *Flow and Heat Transfer characteristics of Backward-Facing Step Laminar Flow in a Rectangular Duct*, Int. J. Heat and Mass Transfer, 43, pp. 457-471.

Kaufmann, W., and Smarr, L., 1993, *Supercomputing and the Transformation of Science*, Scientific American Library, New York.

Kim, J., Moin, P., 1985, *Application of a Fractional-Step Method to Incompressible Navier-Stokes Equations*, Journal of Computational Physics, 59, pp. 308-323.

Le, H., Moin, P., 1997, *Direct Numerical Simulation of Turbulent Flow Over a Backwarding-Facing Step*, Journal of fluid mechanics, 330, pp. 349-474.

Lee, T., Mateescu, D., 1998, *Experimental and Numerical Investigation of 2-D Backward-Facing Step Flow*, 12, Journal of Fluids and Structures, pp. 703-716.

Li, A., 2001, *Experimental and Numerical Study of Three Dimensional Laminar Separated Flow Adjacent to Backward Facing Step*, Ph.D. thesis, University of Missouri-Rolla.

Nie, J. H., Armaly, B. F., 2002, *Three-Dimensional Convective Flow Adjacent to Bacward-Facing Step-Effects of Step Height*, International Journal of Heat and Mass Transfer, 45, pp. 2431-2438.

Nie, J. H., Armaly, B. F., 2003, *Reattachment of Three-Dimensional Flow Adjacent to Backward-Facing step*, 125, Transactions of ASME.

Patankar, S. V., Spalding, D. B. (1972), *A Calculation Procedure for Heat, Mass and Momentum Transfer in Three Dimensional Parabolic Flows*, International Journal of Heat Mass Transfer, 15, pp. 1787.

Patankar, S. V. ,1980, *Numerical Heat Transfer and Fluid Flow*, Taylor&Francis.

Ronald L. Panron, 1984, *Incompressible Flow*, John Wiley & Sons, Inc.

Saldana, J. G. B., Anand, N. K., 2004, *Forced Convection Over a Three-Dimensional Horizontal Backward Facing Step*, International Journal for Computational Methods in Engineering Science and Mechanics, 6 (4), pp. 225-234.

Saldana, J. G. B., 2005, *Numerical Simulation of Mixed Convection over a Three-Dimensional Horizontal Backward-Facing Step*, Texas A&M University Doctoral Dissertation, College Station.

Sankari, S. R., Mees, P. A. J, Nandakumarz, K., 1993, *Development of Three-Dimensional Streamwise-Periodic Flows in Mixed-Convection Heat Transfer*, Journal of Fluid Mechanics, 255, pp. 683-705.

Shah, R.K., and London, A.L., 1978, *Laminar Flow Forced Convection in Ducts*, Academic Press Inc., New York.

Tannehill, J. C., Anderson, D. A., Pletcher, R. H., 1997, *Computational Fluid Mechanics and Heat Transfer*, 2nd Edition, Taylor & Francis.

Williams, P. T., Baker, J., *Numerical Simulations of Laminar Flow Over a 3-D Backward facing step*, International Journal for Numerical Methods in Fluids, 24, pp. 1159-1183.

Versteeg, H. K., Malalasekera, W., 2007, *An Introduction to Computational Fluid Dynamics, The Finite Volume Method*, 2nd Edition, Pearson, Prentice Hall.

Vogel, J. G., Eaton, J. K., 1985, *Combined Heat Transfer and Fluid Dynamic Measurements Downstream of a Backward-Facing Step*, Transactions of the ASME.

APPENDIX A

NAVIER STOKES SOLVER 3D (NSS3D)

NSS3D is a solver code has written in C++ language with one main routine and twenty one subroutines to solve 3-D laminar Navier-Stokes equations. The solution algorithm of NSS3D is build upon the SIMPLE algorithm of Patankar (1980). Table A.1 gives the properties of the computer on which the programs are worked and the average run time of one iteration for the grid A-1 of backward facing step flow.

Table A.1. Properties of the computer used and average time per iteration.

Properties	
Processor	Intel(R) Xeon(R) CPU E5450 @ 2.50 GHz
Installed memory (RAM)	8.00 GB
System type	64-bit operating system
Average time per iteration	3.2 sec

Figure A-1 shows the flowchart of the program in which the names and locations of the subroutines can be found. The explanations of all the routines are given below.

main.cpp: Main routine is the master of the subroutines. The variables and arrays are defined in it. It calls the required subroutines according to the flowchart in Figure A-1. Besides, the auto-relaxation is made on this routine.

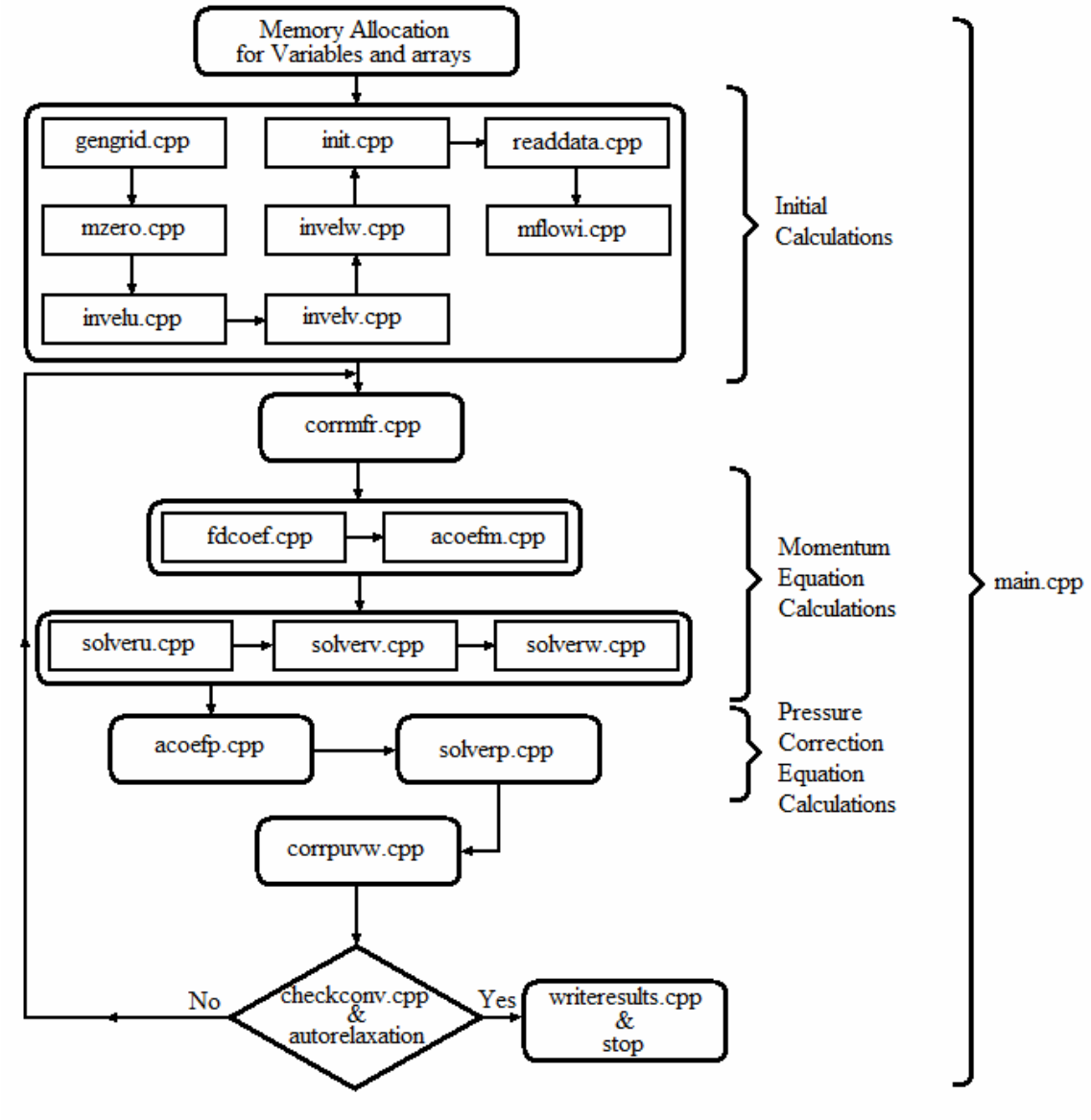

Figure A.1. Flow chart of NSS3D.

gengrid.cpp: The grid set up is made within this subroutine. It is capable of setting both uniform and nonuniform grids with choices of symmetric, diverging and converging non-uniform grids. With this routine, any desired non-uniform grid can be created.

mzero.cpp: It equates all the elements of the three-dimensional arrays to zero.

invelu.cpp & invelv.cpp & invelw.cpp: These subroutines set the velocity values at the inlet boundary. They can either set to a constant value or read the velocity profile from a file.

init.cpp: It initializes the velocity components at all nodes to either the inlet velocity profile or to a constant value.

readdata.cpp: This subroutine reads velocity and pressure data from a file and it enables to read computed data in case of any interruption to continue to run from point of interruption.

mflowi.cpp: It calculates mass flow rate at inlet to use it for mass flow rate correction at outlet.

corrmfr.cpp: It computes the outlet mass flow rate at intermediate stages and corrects it with respect to inlet mass flow rate.

fdcoefm.cpp: It computes the **F** and **D** coefficients of momentum equations.

acoefm.cpp: It computes the **a** coefficients of momentum equations and the source term **b**.

solveru.cpp & solverv.cpp & solverw.cpp: These are the iterative solvers for u, v and w velocity components.

acoefp.cpp: It computes the **ap** coefficients of pressure correction equation and the source term **bp**.

solverp.cpp: It solves the pressure correction equation iteratively.

corrpuvw.cpp: This subroutine corrects the pressure, and velocity components computed in the solvers.

checkconv.cpp: It checks whether the computed pressure and velocity components reached to the convergence criteria or not.

autorelaxation: It is a code arranging the relaxation factors checking the convergence history. It uses two subroutines which are autorecover.cpp and autosave.cpp.

autosave.cpp: It saves the results automatically in user defined intervals to use them in case of any divergence.

autorecover.cpp: In case of divergence, it recovers all the pressure and velocity values to the last auto saved ones and the program reduces the relaxation factors and continues with these data.

writeresults.cpp: It writes the converged results to file.