NUMERICAL STUDY ON AIRFOILS IN STEADY AND OSCILLATING STREAMS

by

Mehmet Rifat Özalp

BS. in M.E., Boğaziçi University, 1995

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science
in
Mechanical Engineering

Boğaziçi University

1999

ACKNOWLEDGMENTS

I would like to thank my thesis supervisor Asst. Prof. Ali Ecder for his guidance and cooperation during my study.

I would like to thank Prof. Akın Tezel and Asst. Prof. Emre Otay for their valuable and important comments, which helped me a lot while constituting the final shape of this work.

Special thanks and appreciation go to my friends A. Devrim Pakkan, S. Utku Ergün, Mine Kaçan and Bülent İmamoğlu for their help and sincereness.

Finally, I would like to express my special thanks to all the members of my family, especially my wife, Zeynep, who provided endless encouragement, support and enthusiasm throughout my studies.

ABSTRACT

In this study, a numerical method has been developed in order to simulate steady and unsteady flow fields around airfoils. The unsteady flow field around the airfoil has been obtained by imposing periodically oscillating far field boundary conditions. In the interior domain, the Navier-Stokes equations have been solved by using Krylov subspace methods. The nonlinear part of the problem has been linearized by Newtons method and GMRES is used as the linear solver with ILU preconditioner for efficient calculations. In order to reduce storage requirements, matrix free method is employed.

The method of body-fitted curvilinear coordinate generation has been used to construct a finite-difference solution of the full, incompressible Navier-Stokes equations for the laminar viscous flow about arbitrary two-dimensional airfoils or any other two dimensional bodies. The governing equations in generelized curvilinear coordinate system were solved by using the stream function-vorticity formulation, which eliminates the pressure term. The velocity components have been calculated from the dependent variables, namely stream function and vorticity.

ÖZET

Bu çalışma ile, kanat profilleri etrafında zamana bağlı ve zamana bağlı olmayan akışı canlandırmak üzere, sayısal bir yöntem geliştirilmiştir. Zamana bağlı akış, sınır koşullarında, akışkanın kendini tekrarlayan harmonik salınım hareketi ile sağlanmıştır. Sınır koşullarının çevrelediği iç bölgede, Navier-Stokes denklemleri, Krylov altuzay yöntemlerinin uygulanması ile çözülmüştür. Problemin doğrusal olmayan kısımlarının çözümünde Newton yöntemi kullanılmıştır. Newton iterasyonunda elde edilen doğrusal denklem sistemi ise, ILU ön-iyileştirme tekniği kullanılarak verimi arttırılan GMRES yöntemi ile çözülmüştür. Hesaplamalardaki hafıza gereksinimini azaltmak üzere, GMRES yönteminin matrisden bağımsız özelliğinden faydalanılmıştır.

İki boyutlu bir kanat veya herhangi bir nesnenin kesiti etrafındaki, sıkıştırılmaz, viskoz ve kararlı akışın sonlu farklar yöntemi ile çözümünü oluşturmak üzere, kesitin geometrisini sınır değeri olarak kabul eden eğrisel koordinat sistemi kullanılmıştır. Problemin ana denklemlerinin çözümünde, basınç değişkenini yok etmek üzere, akım fonksiyonu – çevrinti formülasyonu kullanılmıştır. Bir sonraki aşamada, akışkan hızları, çözümden elde edilen akım fonksiyonu ve çevrinti değerlerinin kullanılmasıyla bulunmuştur.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

A	Coefficient matrix
B	Right hand side of a non-linear system
c	Chord length of an airfoil
D	Diagonal matrix
f	Right hand side of a linear system
H_k	Hessenberg matrix
I	Jacobian of inverse transformation
J	Jacobian of transformation
k	Iteration step
Km	Krylov subspace
L	Lower triangular matrix of LU factorization
$\underline{L}$	Lower triangular matrix of ILU factorization
M	Preconditioner matrix
m	Number of Krylov subspace
n	time step
P	Grid control function
p	Minimal polynomial
p	Pressure
R	Residual vector of a non-linear system
Re	Reynolds number
r_m	Residual vector of a linear system
Q	Grid control function
T	Period of oscillation
t	Time
U	free stream velocity
U	Upper triangular matrix of LU factorization
$\underline{U}$	Upper triangular matrix of ILU factorization
u	Velocity component in x-direction
V	Unknown vector
V_c	Converged solution of the system
V_m	Orthonormal system

$\vec{V}$	Velocity vector
v	Velocity component in y-direction
v_0	Arbitrary vector
x	Unknown vector of a linear system
α	Geometric coefficient of transformation
β	Geometric coefficient of transformation
β^*	Norm of initial guess
γ	Geometric coefficient of transformation
η	Coordinate axis of computational domain
θ	Angle of attack
ρ	Density
υ	Kinematic viscosity
ω	Angular velocity
ω	Relaxation parameter
$\vec{\omega}$	Angular velocity vector
ξ	Coordinate axis of computational domain
ψ	Stream function
$\psi^n{}_{BC}$	Boundary values of stream function at time step n
ζ	Vorticity
$\zeta^n{}_{BC}$	Boundary values of vorticity at time step n

1. INTRODUCTION

The phenomenon of an airfoil in an oscillating stream (that is, a stream of which the angle of attack varies periodically) encountered in many phases of aeronautics. For example, the effect of a set of prerotation vanes upon a wind tunnel fan blade is to produce periodic disturbances through which the fan blades pass. Furthermore, a consideration of counter-rotating propellers shows that the rearward blades operating in the helical wake produced by the forward blades are in a stream of varying angle of attack. As another example, the horizontal tale of an airplane may be subjected to forces induced by fluctuations in the angle of the wing wake [1].

The aerodynamic characteristics (eg. lift, drag and moment) of an airfoil in unsteady flow field are significantly affected by the viscous boundary layer. At certain combinations of oscillation amplitude and period, the boundary layer on an airfoil in pitching motion separates from the airfoil surface, forming a large concentrated region of vorticity known as the dynamic stall vortex. This phenomenon is characterised by dramatic changes in the aerodynamic performance (eg. a rapid change in the moment) of the airfoil. In addition, the production of vibrations and existence of a propulsive force are other problems, which arise in oscillating flows.

In this study, a numerical method for computing massively separated, steady and unsteady flow fields around airfoils is developed on the basis of incompressible viscous flow theory. The periodically oscillating flow field is created by imposing time-dependent boundary conditions at the far field of the domain. The calculation of the operating characteristics of an airfoil near stall is accomplished most efficiently by solving the Navier-Stokes equations on an arbitrary coordinate system. Therefore a computer algorithm using boundary fitted curvilinear coordinates, which is called GRIDGEN, is developed in order to solve the finite difference equations arising from the discretization of the Navier-Stokes equations in transformed plane. In the present study, a numerical solver that can handle non-linear coupled partial differential equations is used. The solver uses Newtons method to linearize the system of equations and GMRES is used as the linear

solver. A powerful preconditioner (ILU) is employed for better spectral properties of the system matrix. For efficient calculations, a matrix free method is employed since the storage requirements and flexibility are very important for the solver to be effective and common.

Experimental and numerical studies on airfoils have been established by a number of researchers. Hodge, Stone and Miller [2] conducted a numerical study on incompressible, laminar, and unsteady flow field around NACA airfoils near stall. In their study boundary fitted curvilinear coordinate systems were optimised for viscous flows at different angles of attack such that boundary layer dependent grid systems for high Reynolds numbers were generated efficiently. The grid systems were utilised in implicit finite difference solutions. The predicted flows around a NACA 6412 airfoil near stall at Reynolds numbers of 4×10^4 and 2×10^5 were compared with experimental observations.

An implicit central difference code was used by Barton and Pulliam [3] to calculate two-dimensional inviscid and thin layer Navier-Stokes solutions for flow about an NACA 0012 airfoil at high angles of attack. It was investigated whether separation could occur in an inviscid calculation and what the causes would be such separation. Examples were shown of inviscid shocked flow with and without separation and shock-free flow with separation. An Euler solution with self induced oscillation and separation driven by a strong shock was contrasted with a shock free solution whose separation was caused by numerical error. Computed solutions to Euler equations were compared to those of potential equations. Comparisons were also made between experimental data from wind tunnel tests and viscous flow calculations at similar conditions.

Halim [4] used a marching procedure for solving approximate Navier-Stokes equations (ANS) in the viscous flow region coupled in a fully implicit manner with the elliptic inviscid equation. In order to solve the system of non-linear algebraic equations, which result from finite difference approximation of Navier-Stokes equations, Newton's iteration method was used and Thomas algorithm was applied to the obtained linear system. To demonstrate the independence of the scheme, as to the way the grid was generated, different grid generation techniques were employed (parabolic and algebraic grid generation algorithms). Both orthogonal and non-orthogonal grids were used to

predict the flow solutions for the J102 and NACA-0012 airfoils. Results were obtained for the J025 airfoil and a comparison with the experiment was shown.

Kandil and Chuang [5] derived the unsteady Euler equations for the flow relative motion with respect to a frame of reference that is rigidly attached to the moving airfoil. The grid was generated once by an elliptic solver without any need for dynamic grid computation. An implicit approximately factored finite-volume scheme was developed and implemented through a fully vectorized computer program. The scheme was based on the spatial approximate factorisation (AF) of Beam and Warming and applied to steady and unsteady transonic flows around NACA 0012 airfoil. For forced harmonic airfoil motions, periodic solutions were achieved within the third cycle of oscillation. Their results were in good agreement with the experimental data, with the exception of the high angle of attack range, where the inviscid Euler equations couldn't handle smooth surface separation.

Schroeder and Fleeter [6] developed an analysis that models the unsteady aerodynamics of a harmonically oscillating flat-plate airfoil, including the effects of mean flow incidence angles, in an incompressible laminar flow at moderate values of Reynolds number. The unsteady viscous flow was assumed to be a small perturbation to the steady viscous flow field. The non-uniform and non-linear steady flow field was described by the Navier-Stokes equations and was independent of the unsteady flow. The small perturbation unsteady viscous flow was described by a system of linear partial differential equations that were coupled to the steady flow field, thereby modelling the strong dependence of the unsteady aerodynamics on the steady flow. Solutions for both steady and unsteady viscous flow field were obtained by a locally analytical method in which the discrete algebraic equations representing the flow field equations were obtained from analytical solutions in individual local grid elements.

An experimental study for visualising the vortical patterns in the wake of an oscillating airfoil was conducted by Koochesfahani [7]. In order to show flow patterns in the wake of NACA 0012 airfoil pitching at small amplitudes, a low speed water channel was used. It was shown that a great deal of control can be exercised on the structure of the wake by the control of frequency, amplitude, and also the shape of the oscillation waveform. An important observation in this study was the existence of an axial flow in the

cores of the wake vortices. Estimates of the magnitude of the axial flow suggested a linear dependence on the frequency and amplitude.

In the study of Tuncer, Wu and Wang [8] unsteady flow fields around NACA 0012 airfoil oscillating in pitch and associated dynamic stall phenomena were investigated. A viscous flow analysis and a simplified vortical flow analysis, both based on an integro-differential formulation of the Navier-Stokes equations were developed and calibrated. A two-layer algebraic turbulence model was also incorporated. In this method, knowing that vorticity is absent in the potential flow zone, vorticity transport equation was solved only in the viscous flow zone and furthermore attached boundary layer and detached recirculating flow regions in the viscous flow zone were treated individually. This method confined computations into the viscous flow zone only. This confinement of the computations and the zonal approach greatly reduced the computational demands. The numerical solution of the discretized equations was based on the successive line under-relaxation (SLUR) method along the radial lines of the O shaped grid and performed in a transformed complex plane. The integral equations for the velocity and surface vorticity were solved by using finite Fourier series expansions of the field variables at each circular ring of the O grid and partially evaluated analytically.

Tung, McAlister and Wang [9] and Panda and Zaman [10] conducted experimental studies on oscillating airfoils in order to investigate their unsteady flow behaviour and load characteristics. The types of the airfoils were VR-7 and NACA 0012 respectively. Another experimental study was made by Ahmed and Chandrasekhara [11] to describe the reattachment process of the separated flow field around oscillating NACA 0012 airfoil.

Ekaterinaris and Menter [12] presented the ability of one- and two-equation turbulence models to predict unsteady separated flows over airfoils. An implicit, factorised, upwind-biased numerical scheme was used for the integration of the compressible, Reynolds averaged Navier-Stokes equations. One- and two-equation turbulence models were first tested for a separated airfoil flow at fixed angle of incidence. The same models were then applied to compute the unsteady flow fields about airfoils undergoing oscillatory motion at low subsonic Mach numbers. Their conclusion was that the leading-edge transitional flow has a significant effect on the development of the unsteady flow field about oscillating airfoils.

As a result of the continuous interest on the application of the Navier-Stokes methods to complex, unsteady flow, Srinivasan, Ekaterinaris and McCroskey [13] calculated the two-dimensional flow field of an oscillating NACA 0015 airfoil using implicit, finite-difference method of Beam-Warming for the solution of the Navier-Stokes equations in the body fitted C-grid domain. The main intent was to evaluate the accuracy of five widely used turbulence models, namely Baldwin-Lomax (B-L), Renormalization Group (RNG), Johnson-King (J-K), Baldwin-Barth (B-B) and Spalart-Allmaras (S-A).

A numerical investigation of unsteady transitional flow over an oscillating NACA 0012 airfoil at a Reynolds number 44,000 and reduced frequency of 0.2 was carried out by Kim, Zaman and Panda [14] and the results were compared with experimental data. In their study the Navier-Stokes equations defined on arbitrary Lagrangian-Eulerian coordinates were solved by a time-accurate finite volume method that incorporates a pressure staggered mesh and an incremental pressure equation for the conservation of mass. The transitional turbulence field is described by multiple-time-scale turbulence equations. In most of the previous studies, either an alternating-direction-implicit (ADI) scheme or a flux vector splitting scheme was used to solve the Navier-Stokes equations and the moving mesh is implicitly taken into account through the use of time-varying grid transformation Jacobians. In this study the time-varying grid was treated explicitly by solving the Navier-Stokes equations defined on arbitrary Lagrangian-Eulerian coordinates and the calculated streaklines, the velocity profiles and vorticity contours were in good agreement with the measured data.

Understanding the effects of compressibility, pitch rate and Reynolds number on the initial stages of unsteady leading-edge boundary layer separation for a pitching NACA 0012 airfoil was the main objective of Choudhuri and Knight [15]. The governing equations were the two-dimensional unsteady, compressible, laminar Navier-Stokes equations written in strong conservation form of Steger [16]. The flow equations were written in an inertial frame of reference and the motion of the airfoil is included through a general time-dependent coordinate transformation. The grid was fixed with respect to the airfoil and therefore pitched with it. For the generation of the C-type grid, hyperbolic grid

generation technique was used. The computations were performed using the approximate-factorisation algorithm of Beam and Warming employing a boundary-fitted grid system.

A new physically consistent method was presented by Guilmineau, Piquet and Queutey [17] for the reconstruction of the velocity fluxes, which arise from discrete equations for the mass and momentum balance. They discussed the computation of the time-mean, turbulent, two-dimensional incompressible viscous flow past an airfoil at fixed incidence. The closure method for fluxes made possible the use of a cell-centred grid in which the velocity and pressure unknowns shared the same location, while circumventing the occurrence of spurious pressure modes. The influence of several turbulence models was investigated. The method was tested on NACA 4412, AS-240B, GAW-(1) commercial type airfoils, for which documented experimental data are available.

Very recently Bhaskaran and Rothmayer [18] have conducted a study of the unsteady, leading-edge flow over stationary, pitching and oscillating NACA-0012 airfoil. Their governing equations consisted of the unsteady Navier-Stokes equations in terms of stream function ψ and vorticity ζ and the body-fitted parabolic coordinate system (ξ, η) was chosen as the computational domain. The parabolic coordinate system was used since it is body-fitted and orthogonal. The governing equations were finite-differenced using a variant of the well-known ADI scheme. The non-linear terms were Newton-linearized about guessed values for and converged by making alternate-direction sweeps in the ξ direction. This approach is essentially the same as a globally iterated parabolized Navier-Stokes problem. The results compared favourably with the previous full-airfoil computations.

2. NUMERICAL GRID GENERATION

2.1. Introduction

In order to numerically solve governing partial differential equations (PDEs) of fluid mechanics, approximations to the partial differentials are introduced. These approximations convert the PDEs to finite difference expressions. The approximate algebraic equations referred to as finite difference equations are subsequently solved at discrete point within the domain of interest. Therefore, a set of grid points within the domain, as well as the boundaries of the domain, must be specified. The creation of such grid system is known as grid generation [19].

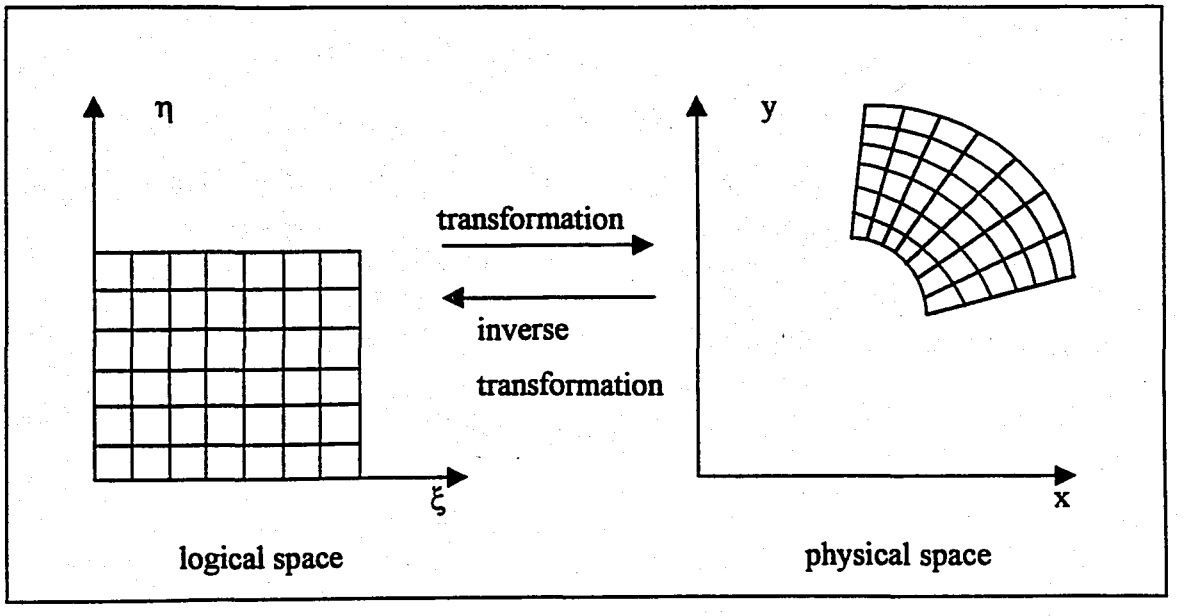


FIGURE 2.1. Mapping from logical space to physical space.

Grid generation is the process of determining the coordinate transformation that maps the body-fitted non-uniform, non-orthogonal physical space x, y, z and t into the transformed uniform orthogonal computational space ξ, η, ϕ and τ. In numerical grid generation, the Jacobian of the transformation is required to be non-zero, and consequently

the transformation has an inverse. Because grids are first chosen in logical space, it is natural to view the transformation as mapping from logical space to physical space, where they form a grid.

The coordinate transformation must satisfy several requirements. The following list includes the most common requirements [20].

- The grid in the transformed computational space must be uniform and orthogonal.
- The transformation must be one to one.
- The transformation must be singular; that is, the Jacobian determinants J and I both must be zero.
- The transformation must yield a body-fitted grid.
- Grid points must be closely spaced in regions of large gradients and widely spaced in regions of small gradients.
- The transformation must be smooth so that the transformation metrics is continuous.
- The maximum and minimum values of the transformed coordinates must occur on the boundaries of the physical space.
- Coordinate lines (or surfaces) of the same family must not cross.
- Complete control of the spacing of points on the boundaries of the physical space must be possible.
- Reasonable control of the spacing of points within the physical space must be possible.
- The transformation must be applicable to both two- and three-dimensional physical spaces.

The procedures of various means of generating curvilinear coordinate systems are of the general types [21];

1.Algebraic grid generation

2.Conformal mapping

3.Grid generation using differential equations

Grid generation for the latter can be classified as follows;

3.a. Elliptic generators

3.b. Nonelliptic generators

3.b.1. Orthogonal grid generation

3.b.2. Hyperbolic grid generation

3.b.3. Parabolic grid generation

3.b.4. Biharmonic grid generation

Conformal mapping is based on complex variable theory, which is limited to two dimensions. Consequently, that method is not as general as the other methods. Algebraic methods and differential equation methods can be applied both two- and three-dimensional spaces. In the present study elliptic grid generation method is used since it satisfies the requirements mentioned formerly.

2.2. Methodology and Mathematical Formulation

Grid generation using differential equations involves the generation of a body fitted coordinate transformation using differential equations.

The problem can be stated as follows:

Determine the coordinate transformation

$$\xi = \xi(x, y), \quad \eta = \eta(x, y) \tag{2.1}$$

that satisfies the boundary point distributions

$$\xi = F(x, y), \quad \eta = G(x, y) \tag{2.2}$$

where, the functions specified in Eq. (2.2) satisfy some differential equation.

Grid generation using differential equations involves the following steps:

1. Determination of the grid point distribution on the boundaries of the physical space.

2. An interior grid point distribution which is specified by a differential equation that satisfies the grid point distributions specified on the boundaries and yields an acceptable interior grid point distribution.

Any of the three classical types of partial differential equation (elliptic, parabolic, or hyperbolic) can be used as the governing grid generation differential equation. The list

of requirements that the coordinate transformations must satisfy suggests the use of an elliptic differential equation as the governing differential equation.

The most common partial differential equation used for grid generation is the Poisson equation:

$$\nabla^2 \xi = P(\xi, \eta) \tag{2.3}$$

$$\nabla^2 \eta = Q(\xi, \eta) \tag{2.4}$$

with boundary conditions specified by Eq. (2.2).

The functions P and Q are selected according to a specific need. The requirement may be a clustering of grid points at some prescribed location or forcing orthogonality at the surface. In this study these source terms P and Q are used for grid point clustering.

Taking the form of the control functions to be [22],

$$\begin{aligned} P(\xi,\eta) &= -\sum_{i=1}^{n} a_i \, sign(\xi - \xi) \exp - (c_i |\xi - \xi|) \\ &- \sum_{j=1}^{m} b_j \, sign(\xi - \xi_j) \exp\left\{ - d_j \left[(\xi - \xi_j)^2 + (\eta - \eta_j)^2 \right]^{1/2} \right\} \end{aligned} \tag{2.6}$$

$$\begin{aligned} Q(\xi,\eta) &= -\sum_{i=1}^{n} a_i \, sign(\eta - \eta_i) \exp - (c_i |\eta - \eta_i|) \\ &- \sum_{j=1}^{m} b_j \, sign(\eta - \eta_j) \exp\left\{ - d_j \left[(\eta - \eta_j)^2 + (\eta - \eta_j)^2 \right]^{1/2} \right\} \end{aligned} \tag{2.7}$$

where n is the number of lines and m is the number of points the grid is to be attracted to, and where a_i, b_j, c_i, d_j, ξ_i, and η_i are parameters. Notice that P and Q are logical-space functions.

The effects of P and Q on the interior grids are shown in Figure 2.2. It is observed that positive values of P cause ξ-lines to move in the positive x-direction, and negative values of P cause ξ-lines to move in the negative x-direction. Non-zero values of P do not effect the η-lines. Q has analogous effect on η-lines and no effects on ξ-lines.

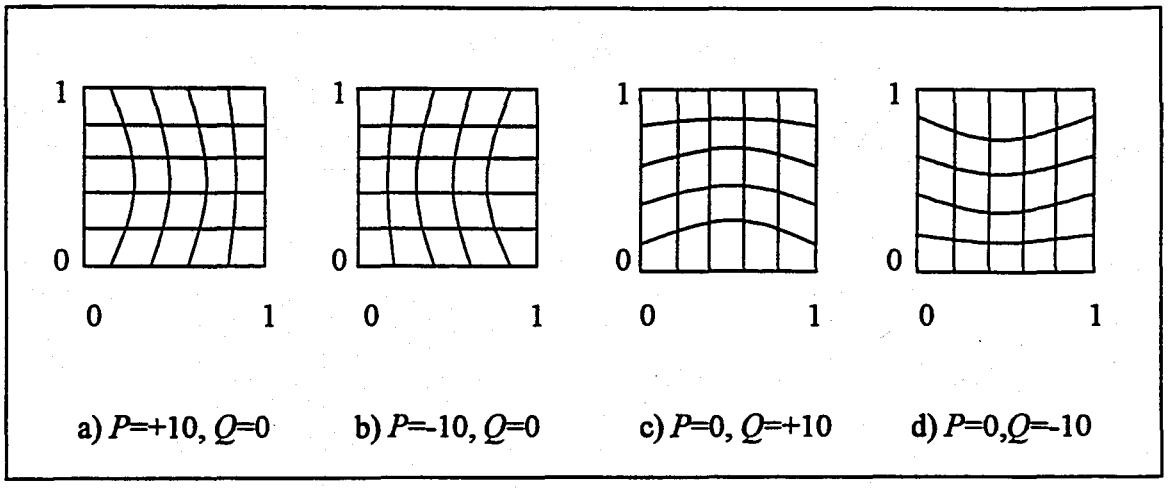


FIGURE 2.2. Effects of P and Q on the interior grid.

Solving Eq. (2.3) and Eq. (2.4) yields the direct transformation, Eq. (2.1). However the non-uniform non-orthogonal physical grid x, y is not known. Consequently Eq. (2.3) and Eq. (2.4) cannot be solved directly. The inverse transformation

$$x = x(\xi, \eta) \tag{2.5}$$

$$y = y(\xi, \eta) \tag{2.6}$$

must be solved for.

Now consider steady 2-D grid generation problem.

$$\nabla^2 \xi = \xi_{xx} + \xi_{yy} = P(\xi, \eta) \tag{2.7}$$

$$\nabla^2 \eta = \eta_{xx} + \eta_{yy} = Q(\xi, \eta) \tag{2.8}$$

After applying chain rule for pds. and some algebraic manipulations:

$$(x_\eta^2 + y_\eta^2)x_{\xi\xi} - 2(x_\xi x_\eta + y_\xi y_\eta)x_{\xi\eta} + (x_\xi^2 + y_\xi^2)x_{\eta\eta} = -I^2(Px_\xi + Qx_\eta) \tag{2.9}$$

$$(x_\eta^2 + y_\eta^2)y_{\xi\xi} - 2(x_\xi x_\eta + y_\xi y_\eta)y_{\xi\eta} + (x_\xi^2 + y_\xi^2)y_{\eta\eta} = -I^2(Py_\xi + Qy_\eta) \tag{2.10}$$

In condensed form two quasilinear elliptic partial equations

$$\alpha x_{\xi\xi} - \beta x_{\xi\eta} + \gamma x_{\eta\eta} = -I^2(Px_\xi + Qx_\eta) \tag{2.11}$$

$$\alpha y_{\xi\xi} - \beta y_{\xi\eta} + \gamma y_{\eta\eta} = -I^2(Py_\xi + Qy_\eta) \tag{2.12}$$

where,

$$\alpha = x_\eta^2 + y_\eta^2 \tag{2.13}$$

$$\beta = x_\xi x_\eta + y_\xi y_\eta \tag{2.14}$$

$$\gamma = x_\xi{}^2 + y_\xi{}^2 \tag{2.15}$$

The boundary conditions for Eq. (2.11) and Eq. (2.12) are (Dirichlet type):

$$x = F(\xi, \eta) \tag{2.16}$$

$$y = F(\xi, \eta) \tag{2.17}$$

Writing Eq. (2.11) and Eq. (2.12) in finite difference form using second-order centred difference approximations yields respectively [20],

$$\begin{aligned} & x_{i-1,j-1}\left(-\frac{\beta_{i,j}}{2}\frac{\Delta\xi}{\Delta\eta}\right) + x_{i-1,j}\left(\alpha_{i,j} - I_{i,j}{}^2 P_{i,j}\frac{\Delta\xi}{2}\right) + x_{i-1,j+1}\left(\frac{\beta_{i,j}}{2}\frac{\Delta\xi}{\Delta\eta}\right) \\ & + x_{i,j-1}\left(\gamma_{i,j}\frac{\Delta\xi^2}{\Delta\eta^2} - I_{i,j}{}^2 Q_{i,j}\frac{\Delta\xi^2}{2\Delta\eta}\right) - 2x_{i,j}\left(\alpha_{i,j} + \gamma_{i,j}\frac{\Delta\xi^2}{\Delta\eta^2}\right) \\ & + x_{i,j+1}\left(\gamma_{i,j}\frac{\Delta\xi^2}{\Delta\eta^2} - I_{i,j}{}^2 Q_{i,j}\frac{\Delta\xi^2}{2\Delta\eta}\right) + x_{i+1,j-1}\left(\frac{\beta_{i,j}}{2}\frac{\Delta\xi}{\Delta\eta}\right) \\ & x_{i+1,j}\left(\alpha_{i,j} + I_{i,j}{}^2 P_{i,j}\frac{\Delta\xi}{2}\right) + x_{i+1,j+1}\left(-\frac{\beta_{i,j}}{2}\frac{\Delta\xi}{\Delta\eta}\right) = 0 \end{aligned} \tag{2.18}$$

$$y_{i-1,j-1}\left(-\frac{\beta_{i,j}}{2}\frac{\Delta\xi}{\Delta\eta}\right)+y_{i-1,j}\left(\alpha_{i,j}-I_{i,j}{}^{2}P_{i,j}\frac{\Delta\xi}{2}\right)+y_{i-1,j+1}\left(\frac{\beta_{i,j}}{2}\frac{\Delta\xi}{\Delta\eta}\right)$$

$$+y_{i,j-1}\left(\gamma_{i,j}\frac{\Delta\xi^{2}}{\Delta\eta^{2}}-I_{i,j}{}^{2}Q_{i,j}\frac{\Delta\xi^{2}}{2\Delta\eta}\right)-2y_{i,j}\left(\alpha_{i,j}+\gamma_{i,j}\frac{\Delta\xi^{2}}{\Delta\eta^{2}}\right) \qquad (2.19)$$

$$+y_{i,j+1}\left(\gamma_{i,j}\frac{\Delta\xi^{2}}{\Delta\eta^{2}}-I_{i,j}{}^{2}Q_{i,j}\frac{\Delta\xi^{2}}{2\Delta\eta}\right)+y_{i+1,j-1}\left(\frac{\beta_{i,j}}{2}\frac{\Delta\xi}{\Delta\eta}\right)$$

$$y_{i+1,j}\left(\alpha_{i,j}+I_{i,j}{}^{2}P_{i,j}\frac{\Delta\xi}{2}\right)+y_{i+1,j+1}\left(-\frac{\beta_{i,j}}{2}\frac{\Delta\xi}{\Delta\eta}\right)=0$$

These finite difference equations constitute the governing equations of the grid generation using partial differential equations.

2.3. Computational Procedure

The equations (2.18) and (2.19) can be solved as follows [20]:

1. Prescribe a uniform orthogonal grid on the ξ, η plane.

2. Specify the distributions of $x = F(\xi, \eta)$ and $y = G(\xi, \eta)$ on the boundaries of the ξ, η plane.

3. Choose an initial approximation $x^{(0)}(\xi, \eta)$ and $y^{(0)}(\xi, \eta)$ for the interior point distribution.

4. Evaluate α, β, γ and I based on the initial approximations $x\,(0)\,(\xi,\eta)$ and $y\,(0)\,(\xi,\eta)$ or on a subsequent approximation $x\,(k)\,(\xi,\eta)$ and $y\,(k)\,(\xi,\eta)$.

5. Solve the linearized system of FDEs, Eq. (2.18) and Eq. (2.19) by an SOR (Successive Over-Relaxation) iterative method.

6. Periodically update the coefficients as the solution proceeds, and repeat step 5.

7. Repeat steps 5 and 6 to convergence.

In the present study, the above algorithm is used and a subroutine called GRIDGEN has been developed. To illustrate grid generation using GRIDGEN some examples have been given in the next section.

2.4. Application of GRIDGEN to Some Arbitrary Two-Dimensional Geometries

2.4.1. Simply Connected Domains

Examples of grid generation for simply connected domains by GRIDGEN are demonstrated in Figures 2.3, 2.4 and 2.5. By definition a simply connected region is one which is reducible and can be contracted to a point. Thus, for a simply connected region, there are no objects within the domain.

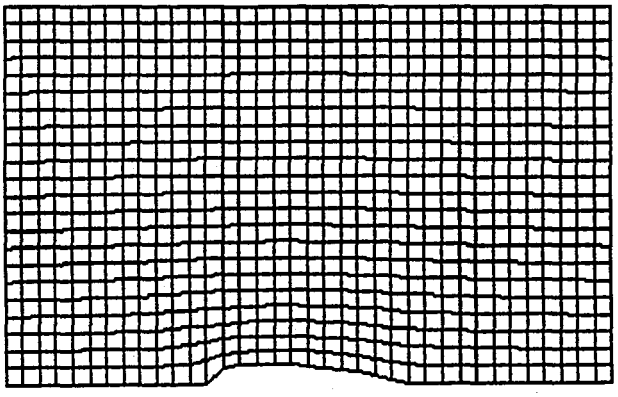


FIGURE 2.3. Grid generated for the upper half of the symmetrical NACA0020 airfoil.

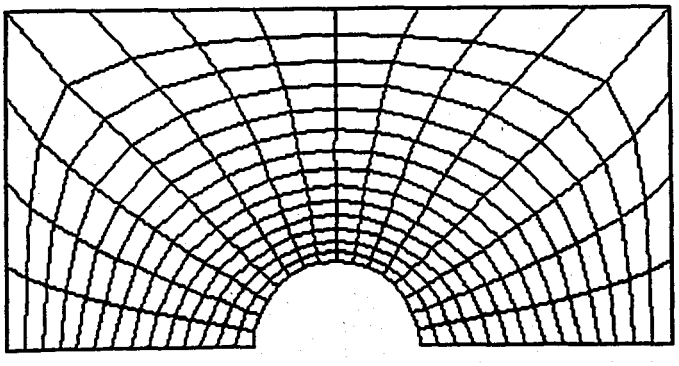


FIGURE 2.4. Grid generated for the upper half of the circular cylinder.

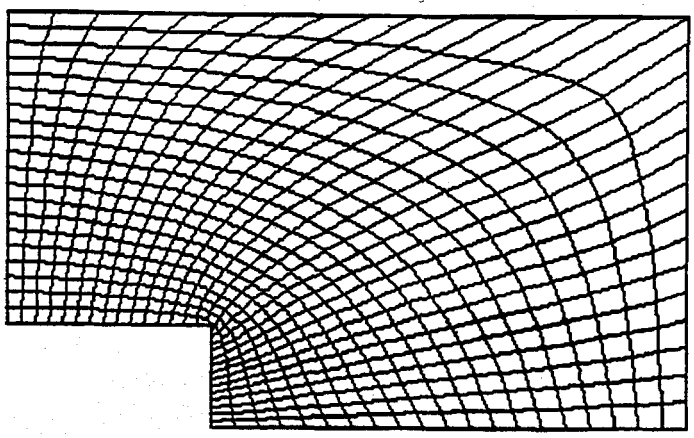


FIGURE 2.5. Grid generated for the backward facing step.

2.4.2. Doubly Connected Domains

A doubly connected domain is defined as a region, which is not reducible. Thus, for a doubly connected region there is one configuration within the domain. A doubly connected region may be rendered simply connected by introducing a suitable branch cut. This procedure is accomplished by inserting a branch cut that extends from a point on the body (interior boundary) to a point on the outer boundary. Examples of this type grid generation by GRIDGEN are given in Figures 2.6, 2.7 and 2.8.

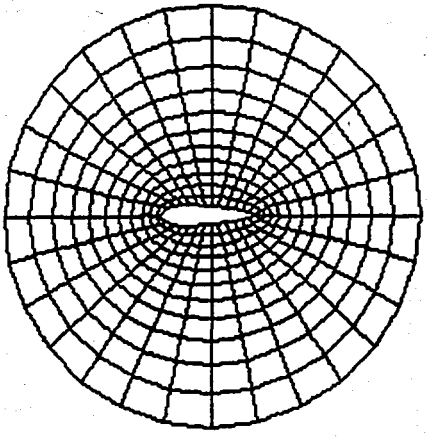


FIGURE 2.6. O-type grid generated for the NACA 0020 airfoil.

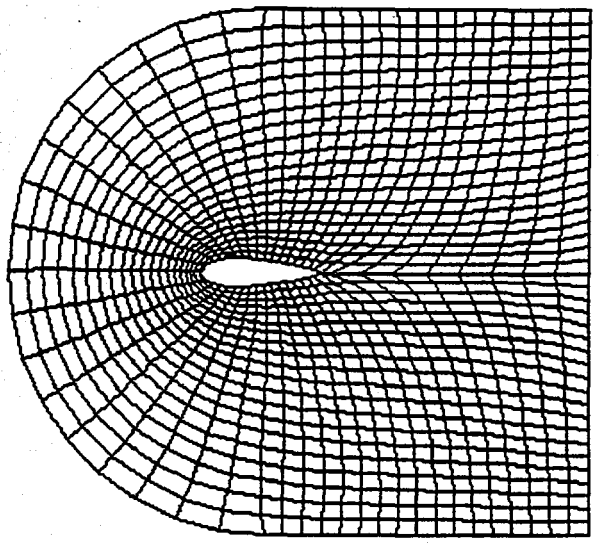


FIGURE 2.7. C-type grid generated for the NACA 0020 airfoil.

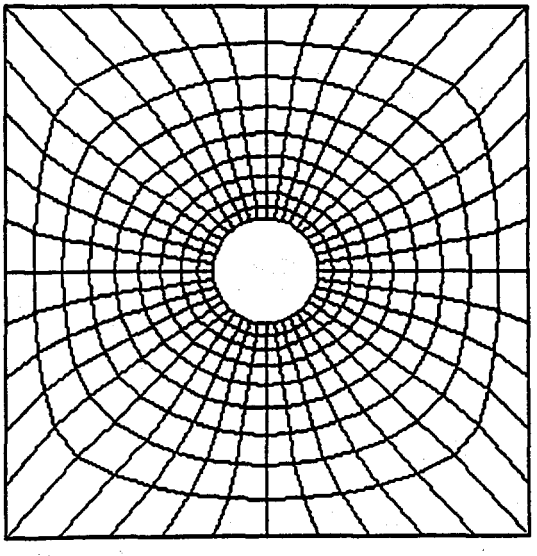


FIGURE 2.8. Grid generated for the circular cylinder.

3. THEORY AND MATHEMATICAL MODELING

3.1. Airfoils

At the beginning of twentieth century, the importance of aerodynamics blew up with the advent of successful powered flight. Then interest grew in the understanding of the aerodynamic action of such lifting surfaces as fixed wings on airplanes and later rotors on helicopters. Between 1912 and 1918, the studies of German scientist Ludwig Prandtl showed that the aerodynamic consideration of wings could be split in two parts: first the study of a section of a wing, an airfoil, and the modification of such airfoil properties for the complete finite wing [23].

When the wing as drawn in perspective in Figure 3.1 is considered, the wing extends in the y direction. The free-stream velocity U is parallel to the *x-z* plane. Any section cut by a plane parallel to the *x-z* plane is called an airfoil.

3.1.1. Airfoil Nomenclature

The first airfoil shapes, that were patented, were developed by Horaito F. Phillips in 1884. He was the first man to carry out the first wind tunnel experiments on airfoils. In 1902 Wright brothers developed their own wind tunnel tests, with developing more efficient shapes, that contributed to their successful first flight on December 17, 1903. In early 1930's, the forerunner of NASA, the National Advisory Committee for Aeronautics (NACA) began on a series of definite airfoil experiments using rational and systematic airfoil shapes. The well-known standard since that day is by NACA that is still in use [23].

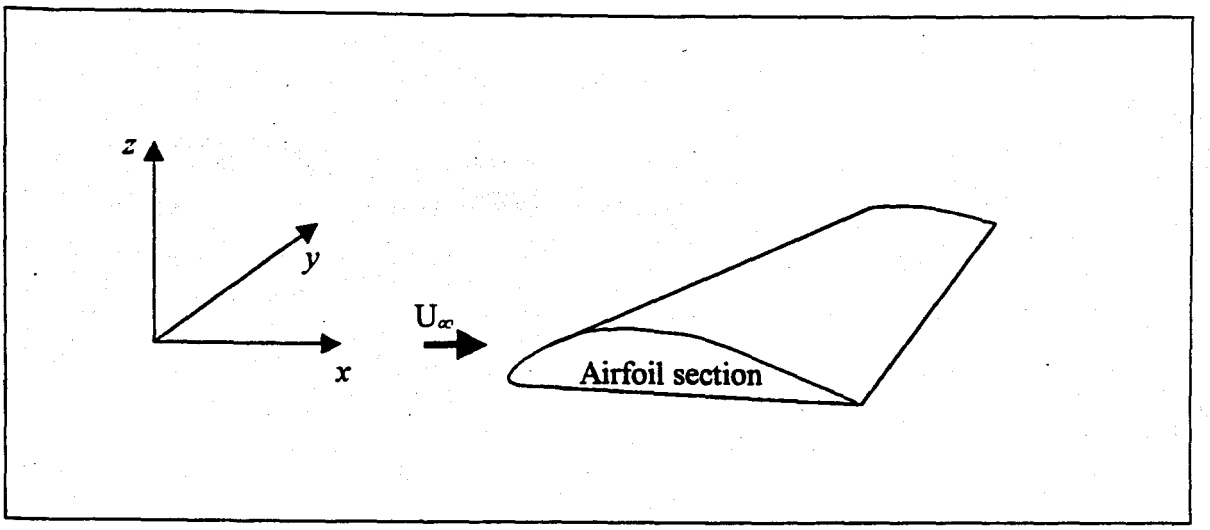


FIGURE 3.1. Definition of an airfoil.

When Figure 3.2 is considered, the mean camber line is the focus of points halfway between the upper and lower surfaces as measured perpendicular to the mean camber line itself. The point, that is most forward of the mean camber line is called the leading edge and the most rearward is called the trailing edge. The chord line is the straight line connecting the leading and the trailing edges. The distance from the trailing to the leading edge measured along the chord line is simply named the chord, c of the airfoil. The maximum distance between the mean camber line and the chord line is called the camber. The thickness is the perpendicularly measured distance between the upper and lower surfaces. The shape of the airfoil at the leading edge is usually circular with a radius of 0.02c. All standard NACA airfoils are generated by specifying the shape of the mean camber line and then wrapping a symmetrical thickness distribution around the mean camber line.

The airfoil series developed in the 1930's was the "four digit series", such as NACA 2412 airfoil. The first digit stands for the maximum camber in hundredths of chord, the second stands for the location of the maximum camber along the chord from the leading edge in tenths of chord, and the last two digits give the maximum thickness in hundredths of chord. For the NACA 2412 airfoil, maximum camber is 0.02c located at 0.4c from the leading edge and the maximum thickness is 0.12c. While stating these numbers, one uses in percentage of chord like, two per cent camber at 40 per cent chord, with 12 per cent thickness.

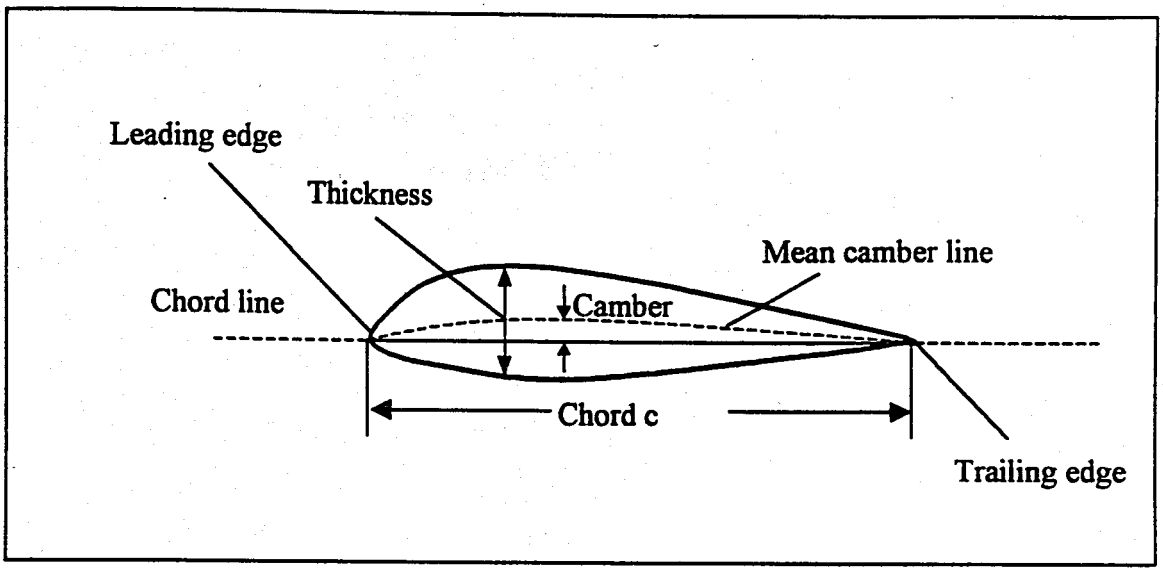


FIGURE 3.2. Airfoil nomenclature [23].

The so-called symmetrical airfoil is the one with the camber line and the chord line coincident. For example, the NACA 0013 airfoil is a symmetric airfoil with a same shape both above and below the chord line and with a maximum thickness of 13 per cent [24].

The second airfoil series developed is the "five digit series" such as NACA 23012 airfoil. In these series, if the first digit is multiplied by 3/2, it gives the design lift coefficient in tenths. Then if the next two digits are divided by two, it gives the location of the maximum camber along the chord from the leading edge in hundredths of chord and the final two digits give the maximum thickness in hundredths of chord. So for NACA 23012 airfoil, the design lift coefficient is 0.3, the location of maximum camber is at 0.15c, and the airfoil has 12 per cent maximum thickness [24].

One other airfoil series developed during the World War II is the "6-series" laminar flow airfoils, for example NACA 65-218 airfoil. The first digit simply stands for the series, the second gives the location of the minimum pressure in tenths of chord from the leading edge, the third gives the design lift coefficient in tenths and the last two digits give the maximum thickness in hundredths of chord. For the NACA 66-220 airfoil, the first six is simply the series designation, the minimum pressure occurs in 0.5c for the basic symmetric thickness distribution at zero lift, the design lift coefficient is 0.2 and the airfoil has a maximum thickness of 20 per cent [24].

3.1.2. Finite Wing

Airfoils have the same properties of a wing of infinite span; indeed airfoil data are frequently denoted as "infinite wing" data. However all real airplanes have wings of finite span.

An airfoil is simply a section of a wing. At first one can think of the same behaviour of airfoils and wings, however flow over an airfoil is two-dimensional and the finite wing is a three-dimensional body, and consequently the flow over the finite wing is three dimensional. The physical mechanism for generating lift on the wing is the existence of high pressure at the bottom surface and a low pressure at the top surface. This net difference creates the lift. A by-product of this pressure imbalance is the tendency to curl around the tips, being forced from the high-pressure region just underneath the tips to the low-pressure region on top. As a result, on the top surface of the wing there is generally a span-wise component of flow from the root toward the tip, causing the streamlines over the bottom surface to bend toward the tip. Clearly the flow over the finite wing is three-dimensional and therefore one would expect the overall aerodynamic properties of such a wing differ from those of its airfoil.

Today many large aircraft companies design their own special airfoils, by the aid of airfoil design computer programs utilizing either panel technologies or direct finite difference solutions of the governing partial differential equations for the flow field. For instance Boeing 727, 737, 747, 757 and 767 all have their specially designed Boeing airfoils [23].

3.2. Separation and Vortex Formation

The decelerated fluid particles in the boundary layer do not, in all cases, remain in the thin layer, which adheres to the body along the whole wetted length of the wall. In

some cases the boundary layer increases its thickness considerably in the downstream direction and the flow in the boundary layer becomes reversed. This causes the decelerated fluid particles to be forced outward, which means that the boundary layer is separated from the wall. We than speak of boundary layer separation. This phenomenon is always associated with the formation of vortices and large energy losses in the wake of the body. It occurs primarily near blunt bodies, such as circular cylinders and spheres. Behind such a body there exists a region of strongly decelerated flow (so-called wake), in which the pressure distribution deviates considerably from that in a frictionless fluid. The large drag of such bodies can be explained by the existence of this large deviation in pressure distribution, which is a consequence of boundary layer separation [25].

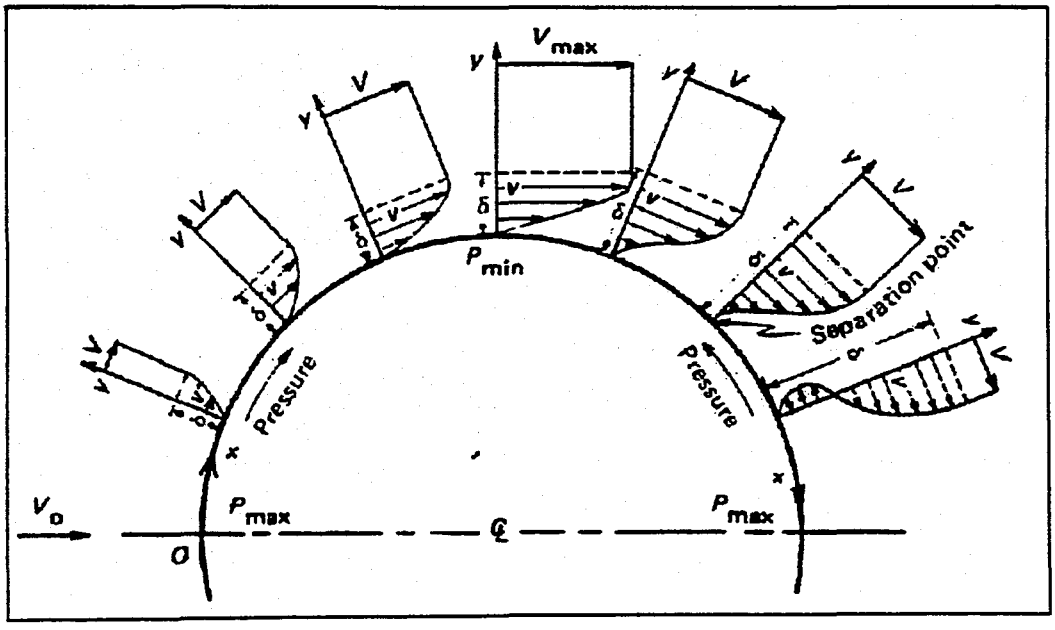


FIGURE 3.3. Development of boundary layer profiles around a circular cylinder [26].

In order to explain the important phenomenon of boundary layer separation in detail let us consider the flow about a blunt body, e.g. about a circular cylinder, as shown in Figure 3.3. The velocity varies from zero, or stagnation, at the front of the cylinder to a maximum at the top of the cylinder. The velocity then decreases along the aft face, reaching zero at the aft stagnation point. The corresponding pressures are maximum at the front and rear stagnation points and a minimum at the top of the cylinder. At this time it is suitable to introduce a basic consideration in the boundary layer theory. The pressure is constant across a boundary layer and equal to the value just outside the boundary layer.

This fallows from the fact that the decreasing velocities within the boundary layer are because of the viscous shearing effects and not because of the exchange of pressure and kinetic energy associated with velocity changes in the essentially perfect fluid region [26].

The boundary layer profiles in Figure 3.3 start at the forward stagnation point. As the flow accelerates outside the boundary layer because of the decreasing pressure, the same favorable pressure gradient acts on the boundary layer fluid. Pressure gradients tending to accelerate the flow are called favorable gradients, where as the tending to slow the flow are called adverse gradients. Layers of fluid closer to the surface are moving slower, and therefore are favorable gradients act on them for a longer time. Thus these slower layers acquire larger velocity increment. The result is that the boundary layer profile becomes somewhat fuller and the total thickness becomes less in a favorable pressure gradient. Aft of the top of the cylinder, the pressure starts to increase (ie., the pressure gradients becomes adverse). In the stream just outside the boundary layer, the velocity and the associated kinetic energy of the fluid are just high enough to convey the fluid against the adverse gradient to the rear stagnation point. In the boundary layer the velocity and the kinetic energy are less, however, since the viscous friction has been consuming kinetic energy which appears as heat. The fluid velocity reaches zero before getting to the rear stagnation point. Furthermore, the slower fluid near the wall, the greater the deceleration it suffers because of the longer exposure to the adverse gradient. Thus the shape of the velocity profile changes, producing a reversal in curvature near the wall. When the profile becomes vertical at the wall (ie., v becomes zero for some distance out from the wall), the flow can no longer continue around the body.

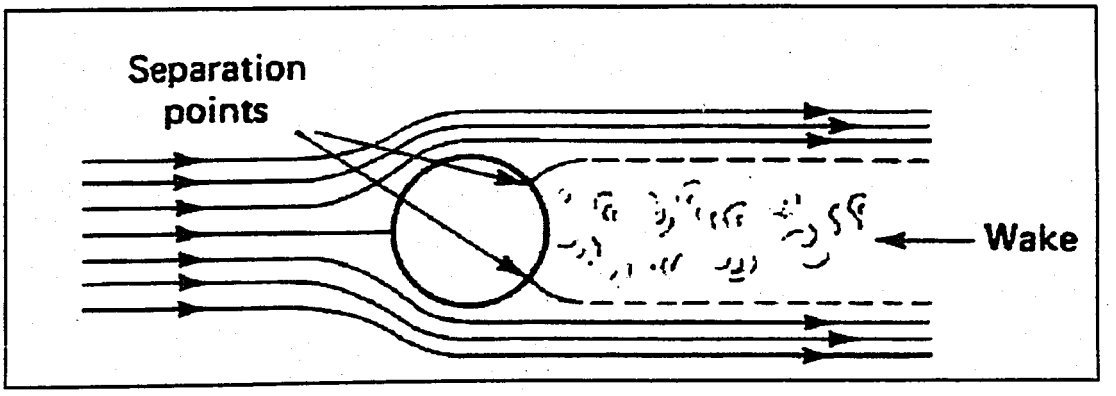


FIGURE 3.4. Separated flow behind a circular cylinder [26].

This is known as the separation point. Beyond this, the flow along the surface will actually be reversed in direction, since no fluid flowing from the front of the body is filling the space. The main flow leaves the body and flows downstream, as indicated in Figure 3.4.

This general description of the flow around a circular cylinder applies to any body sufficiently blunt that the pressure gradient on the rear portion is high enough to cause separation. In fact, a bluff body is defined as one, in which separation occurs well ahead of the trailing edge. Conversely, a streamline body is a body with a pressure increase downstream of the minimum pressure point sufficiently gradual so that separation does not occur until the trailing edge is almost reached. Airfoils at normal flight attitudes are streamline bodies. If, however, the airfoil angle of attack is increased until the pressure gradient downstream of the peak suction is so high that separation occurs well before the trailing edge, the airfoil is said to be stalled and it has really become a bluff body, which is shown in Figure 3.5.

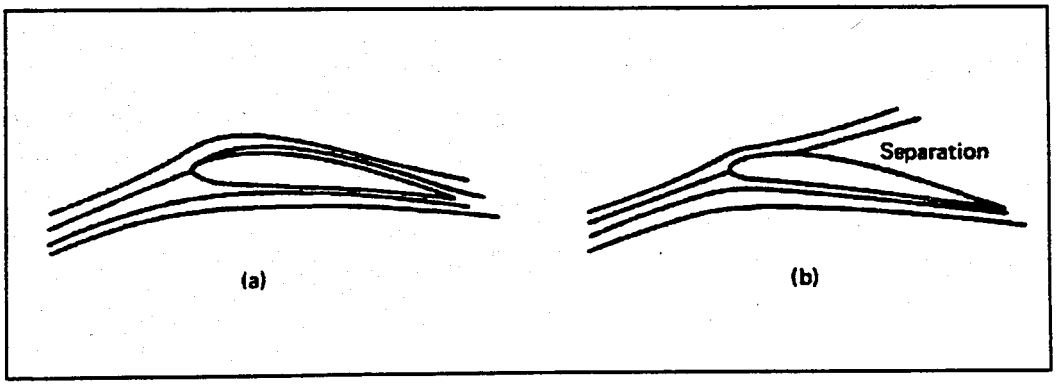


FIGURE 3.5. Flow around an airfoil: (a) unseparated flow; (b) separated stalled flow [26].

At the stall, the Kutta condition for the location of the rear stagnation point at the trailing edge is no longer met, and the circulation and therefore the lift decrease. Airfoils that stall with a separation that first occurs just forward the trailing edge and slowly moves forward as the angle of attack is increased above the angle for the initial stall have a relatively gradual loss of lift. Airfoils for which the first separation is far forward on the airfoil suffer a large immediate loss of lift, which leads to a more violent stall.

3.3. Governing Equations of the System

In order to obtain the flow field around an arbitrary shaped object exposed to external flow, one has to solve a set of partial differential equations arising from the conservation of mass and conservation of momentum. The governing equations for a unsteady, incompressible, viscous flow without body forces is stated in vector form as follows:

Continuity:
$$\vec{\nabla}\cdot\vec{V}=0 \tag{3.1}$$

Momentum:
$$\frac{\partial\vec{V}}{\partial t}+\left(\vec{V}\cdot\vec{\nabla}\right)\vec{V}+\frac{\vec{\nabla}p}{\rho}=\upsilon\nabla^2\vec{V} \tag{3.2}$$

If we write these equations in dimensional, non-conservative form for two dimensions, they become:

Continuity:
$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}=0 \tag{3.3}$$

x-Momentum:
$$\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+\frac{1}{\rho}\frac{\partial p}{\partial x}=\upsilon\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right) \tag{3.4}$$

y-Momentum:
$$\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}+\frac{1}{\rho}\frac{\partial p}{\partial y}=\upsilon\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}\right) \tag{3.5}$$

The governing equations for an incompressible flow may be expressed in two different formulations based on dependent variables used. The first formulation is expressed in terms of pressure and velocity. The second form of the equations is the so-called vorticity-stream function equations, which are derived from the Navier-Stokes equations by incorporating the definitions for the vorticity and the stream function. This formulation is generally used in two-dimensional applications. It should be noted that for a three dimensional flow, it is possible to extent the approach of stream function, but additional complication arise.

The governing equations for the primitive variable formulation are a mixed system of equations, which are solved simultaneously. The unknowns in the equations are velocity and pressure. Since there is no direct link for the pressure between the continuity and momentum equations, one has to establish some mathematical manipulations. Generally speaking there are two procedures for this purpose. The first one is the Poisson equation for pressure and the second is the introduction of artificial compressibility into the continuity equation. On non-staggered grids, Poison formulation can lead to odd-even decoupling of the pressure. In addition to this, the pressure boundary condition must be carefully selected to achieve proper level of convergence and accuracy. Staggered grid formulations can satisfy boundary conditions exactly. However, they may fail to satisfy the discrete continuity equation. The vorticity-stream function formulation does not include pressure term, therefore, first the velocity field is determined and subsequently the Poisson equation for pressure is employed to obtain the pressure field [19].

3.4. Vorticity-Stream Function Formulation

The vorticity of a fluid particle is defined as twice the angular velocity and is

$$\vec{\zeta} = 2\vec{\omega} = \vec{\nabla} \times \vec{V} \tag{3.6}$$

which, for a two dimensional flow, is reduced to

$$\zeta = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \tag{3.7}$$

For a two dimensional, incompressible flow, a function may be defined which satisfies the continuity equation. Such a function is known as the stream function and in Cartesian coordinate system, is given by

$$u = \frac{\partial \psi}{\partial y} \tag{3.8}$$

$$v = -\frac{\partial \psi}{\partial x} \tag{3.9}$$

From a physical point of view, the lines of constant ψ represent streamlines, and the difference in the values of ψ between two streamlines gives the volumetric flow rate between two.

In order to derive the vorticity transport equation, the pressure is eliminated from the momentum equations by cross-differentiation. The following equation must be constructed in order to obtain the desired result.

$$\frac{\partial}{\partial y}\left(\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+\frac{1}{\rho}\frac{\partial p}{\partial x}=\upsilon\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)\right)$$

$$-\frac{\partial}{\partial x}\left(\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}+\frac{1}{\rho}\frac{\partial p}{\partial y}=\upsilon\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}\right)\right)=0 \tag{3.10}$$

After the substruction the following equation is obtained,

$$\frac{\partial\left(\frac{\partial u}{\partial y}-\frac{\partial v}{\partial x}\right)}{\partial t}+u\frac{\partial\left(\frac{\partial u}{\partial y}-\frac{\partial v}{\partial x}\right)}{\partial x}+v\frac{\partial\left(\frac{\partial u}{\partial y}-\frac{\partial v}{\partial x}\right)}{\partial y}+\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}\right)\left(\frac{\partial u}{\partial y}-\frac{\partial v}{\partial x}\right)$$

$$=\upsilon\left[\frac{\partial^2\left(\frac{\partial u}{\partial y}-\frac{\partial v}{\partial x}\right)}{\partial x^2}+\frac{\partial^2\left(\frac{\partial u}{\partial y}-\frac{\partial v}{\partial x}\right)}{\partial y^2}\right] \tag{3.11}$$

The fourth term on the left-hand side is zero by continuity and upon substitution of vorticity defined by Eq. (3.7), one obtains

$$\frac{\partial\zeta}{\partial t}+\left(\frac{\partial\psi}{\partial y}\right)\left(\frac{\partial\zeta}{\partial x}\right)-\left(\frac{\partial\psi}{\partial x}\right)\left(\frac{\partial\zeta}{\partial y}\right)=\upsilon\left(\frac{\partial^2\zeta}{\partial x^2}+\frac{\partial^2\zeta}{\partial y^2}\right) \tag{3.12}$$

Eq. (3.12) is known as the vorticity transport equation and is classified as elliptic type partial differential equation with the unknown being the vorticity ζ. Now reconsidering the definition of vorticity, if we substitute the relations in equations (3.8) and (3.9) in Eq. (3.7) we obtain,

$$\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = -\zeta \tag{3.13}$$

This equation is known as the stream function equation and is classified as an elliptic partial differential equation. The unknown is the stream function ψ, whose ζ is provided from the solution of Eq. (3.12). Once the stream function has been computed, the velocity components may be determined from Eq. (3.8) and Eq. (3.9).

The vorticity transport and stream function equations may be expressed in a non-dimensional form by using the non-dimensional quantities defined below,

$$x^* = \frac{x}{L} \tag{3.14}$$

$$y^* = \frac{y}{L} \tag{3.15}$$

$$\psi^* = \frac{\psi}{u_\infty L} \tag{3.16}$$

$$\zeta^* = \frac{\zeta L}{u_\infty} \tag{3.17}$$

$$\mathrm{Re} = \frac{\rho_\infty u_\infty L}{\mu_\infty} \tag{3.18}$$

where, L is the characteristic length, and ρ_∞ and u_∞ are the reference (eg., free stream) density and velocity, respectively. Substituting the non-dimensional quantities in Eq. (3.11) and Eq. (3.12), the non-dimensional and non-conservative form of the vorticity transport equation and stream function equation, dropping the superscript *, can be expressed as respectively,

$$\frac{\partial \zeta}{\partial t}+\left(\frac{\partial \psi}{\partial y}\right)\left(\frac{\partial \zeta}{\partial x}\right)-\left(\frac{\partial \psi}{\partial x}\right)\left(\frac{\partial \zeta}{\partial y}\right)=\frac{1}{\mathrm{Re}}\left(\frac{\partial^2 \zeta}{\partial x^2}+\frac{\partial^2 \zeta}{\partial y^2}\right) \tag{3.19}$$

$$\frac{\partial^2 \psi}{\partial x^2}+\frac{\partial^2 \psi}{\partial y^2}=-\zeta \tag{3.20}$$

These are the governing equations for unsteady, incompressible, two dimensional viscous flow in terms of the vorticity and stream function.

3.5. Transformation of the Governing Equations

In order to enhance the efficiency and accuracy of the numerical scheme and to simplify the implementation of boundary conditions, a transformation from physical space to computational space is performed in section 2 (Numerical Grid Generation). The computational domain is a rectangular shape, which is divided into equally spaced grid system. In order to solve the governing equations of motion in computational space, a transformation of the governing equations from physical space into computational space is required. Therefore, we obtain the governing equations in terms of the generalised coordinates as independent variables, which makes the computations to be performed effectively.

3.5.1. Generalized Coordinate Transformation

It is assumed that there is a unique, single valued relationship between the generalized and the physical coordinates and the equations of motion are transformed by the following relations:

$$\begin{aligned} \xi &= \xi(x, y, z) \\ \eta &= \eta(x, y, z) \\ \gamma &= \gamma(x, y, z) \end{aligned} \tag{3.21}$$

and by implication,

$$\begin{aligned} x &= x(\xi, \eta, \gamma) \\ y &= y(\xi, \eta, \gamma) \\ z &= z(\xi, \eta, \gamma) \end{aligned} \tag{3.22}$$

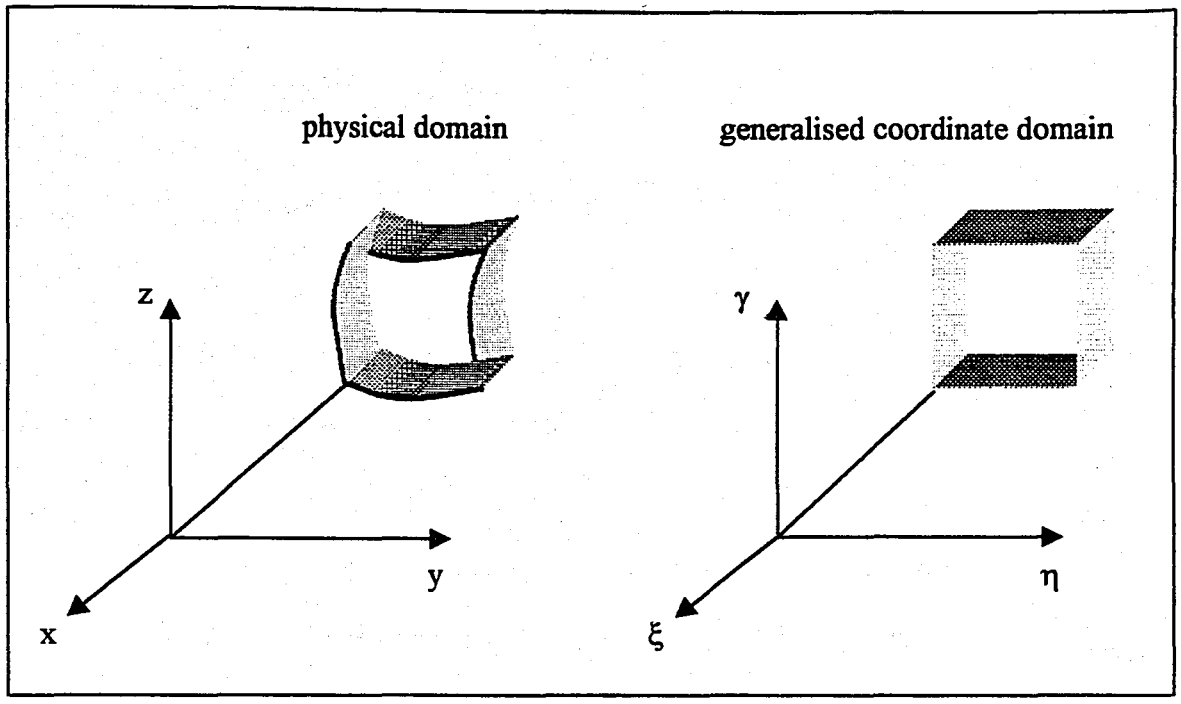


FIGURE 3.6. Correspondence of the physical and generalised-coordinate domains.

The chain rule of partial differentiation provides the following expressions for the two-dimensional Cartesian derivatives:

$$\frac{\partial}{\partial x} = \xi_x \frac{\partial}{\partial \xi} + \eta_x \frac{\partial}{\partial \eta} \tag{3.23}$$

$$\frac{\partial}{\partial y} = \xi_y \frac{\partial}{\partial \xi} + \eta_y \frac{\partial}{\partial \eta} \tag{3.24}$$

where, ξ_x, ξ_y, η_x and η_y are the metrics of transformation and defined as

$$\xi_x = \frac{1}{I}\frac{\partial y}{\partial \eta} \tag{3.25}$$

$$\xi_y = -\frac{1}{I}\frac{\partial x}{\partial \eta} \tag{3.26}$$

$$\eta_x = -\frac{1}{I}\frac{\partial y}{\partial \xi} \tag{3.27}$$

$$\eta_y = \frac{1}{I}\frac{\partial x}{\partial \xi} \tag{3.28}$$

and I is the inverse Jacobian matrix defined as:

$$I = J^{-1} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{bmatrix} \tag{3.29}$$

Using the transformations given in Table B.1, and substituting the metrics from Eq.(3.25-29) into Eq.(3.19) and Eq.(3.20), and after rearranging the sequence of terms, the governing equations of the system can be obtained.

The governing equation (3.19) in transformed plane has the following form [27]:

$$\frac{1}{I^2}\left[\left(\alpha\frac{\partial^2\psi}{\partial\xi^2}\right) - \left(2\beta\frac{\partial^2\psi}{\partial\xi\partial\eta}\right) + \left(\gamma\frac{\partial^2\psi}{\partial\eta^2}\right)\right] + P^*\left(\frac{\partial\psi}{\partial\xi}\right) + Q^*\left(\frac{\partial\psi}{\partial\eta}\right) = -\zeta \tag{3.30}$$

where α, β and γ are the scale factors of transformation and given in detail in section two.

In addition, the vorticity-transport equation (3.20) in transformed plane has the following form:

$$\frac{\partial \zeta}{\partial t}+\frac{1}{I}\left[\left(\frac{\partial \psi}{\partial \eta}\right)\left(\frac{\partial \zeta}{\partial \xi}\right)-\left(\frac{\partial \psi}{\partial \xi}\right)\left(\frac{\partial \zeta}{\partial \eta}\right)\right]$$

$$=\frac{1}{\mathrm{Re}}\left\{\frac{1}{I^2}\left[\left(\alpha\frac{\partial^2 \zeta}{\partial \xi^2}\right)-\left(2\beta\frac{\partial^2 \zeta}{\partial \xi \partial \eta}\right)+\left(\gamma\frac{\partial^2 \zeta}{\partial \eta^2}\right)\right]+P^*\left(\frac{\partial \zeta}{\partial \xi}\right)+Q^*\left(\frac{\partial \zeta}{\partial \eta}\right)\right\} \tag{3.31}$$

where

$$P^*(\xi,\eta)=\frac{1}{J^3}\begin{bmatrix}\left(\frac{\partial x}{\partial \eta}\left(\alpha\frac{\partial^2 y}{\partial \xi^2}-2\beta\frac{\partial^2 y}{\partial \xi \partial \eta}+\gamma\frac{\partial^2 y}{\partial \eta^2}\right)\right)\\ -\left(\frac{\partial y}{\partial \eta}\left(\alpha\frac{\partial^2 x}{\partial \xi^2}-2\beta\frac{\partial^2 x}{\partial \xi \partial \eta}+\gamma\frac{\partial^2 x}{\partial \eta^2}\right)\right)\end{bmatrix} \tag{3.32}$$

$$Q^*(\xi,\eta)=\frac{1}{J^3}\begin{bmatrix}\left(\frac{\partial y}{\partial \xi}\left(\alpha\frac{\partial^2 x}{\partial \xi^2}-2\beta\frac{\partial^2 x}{\partial \xi \partial \eta}+\gamma\frac{\partial^2 x}{\partial \eta^2}\right)\right)\\ -\left(\frac{\partial x}{\partial \xi}\left(\alpha\frac{\partial^2 y}{\partial \xi^2}-2\beta\frac{\partial^2 y}{\partial \xi \partial \eta}+\gamma\frac{\partial^2 y}{\partial \eta^2}\right)\right)\end{bmatrix} \tag{3.33}$$

The velocity components u and v, which have been given in equation (3.8) and (3.9) respectively, have the following form in transformed plane:

$$u = \frac{\partial \psi}{\partial y} = \frac{1}{I}\left[\left(\frac{\partial \psi}{\partial \eta}\right)\left(\frac{\partial x}{\partial \xi}\right) - \left(\frac{\partial \psi}{\partial \xi}\right)\left(\frac{\partial x}{\partial \eta}\right)\right] \tag{3.34}$$

$$v = -\frac{\partial \psi}{\partial x} = \frac{1}{I}\left[\left(\frac{\partial \psi}{\partial \eta}\right)\left(\frac{\partial y}{\partial \xi}\right) - \left(\frac{\partial \psi}{\partial \xi}\right)\left(\frac{\partial y}{\partial \eta}\right)\right] \tag{3.35}$$

3.6. Boundary Conditions in Transformed Plane

In order to solve the governing equations by the present numerical scheme boundary conditions must be prescribed. In the present study there are body surface, far field, inflow, outflow and branch cut boundary conditions, which are presented in Figure 3.7 and 3.8.

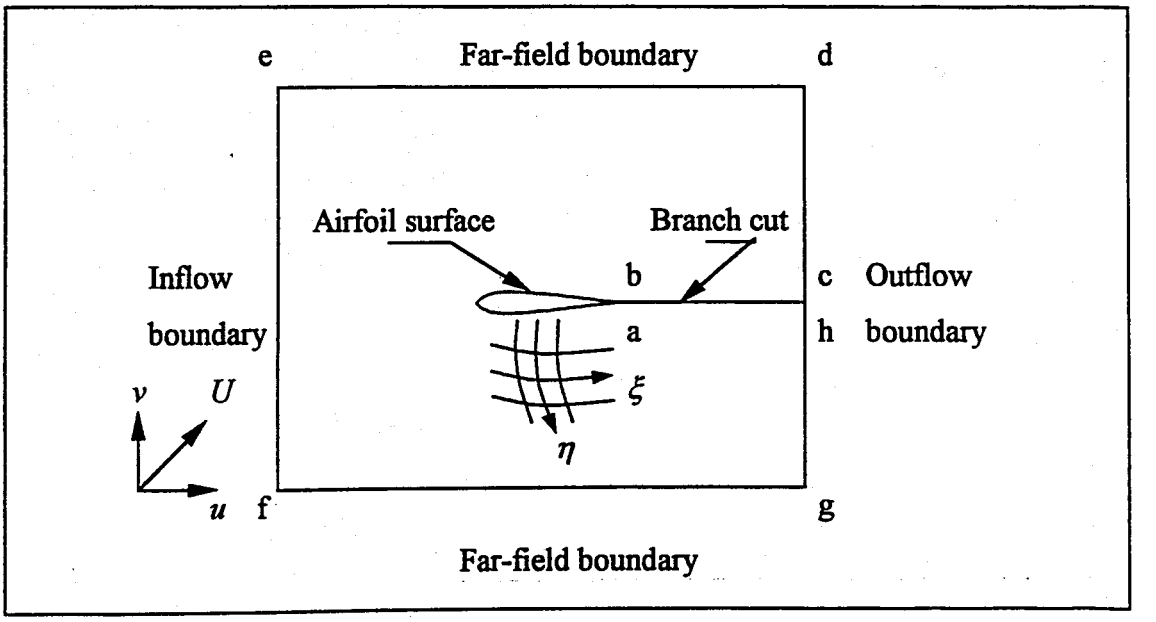


FIGURE 3.7. Grid configuration and physical domain.

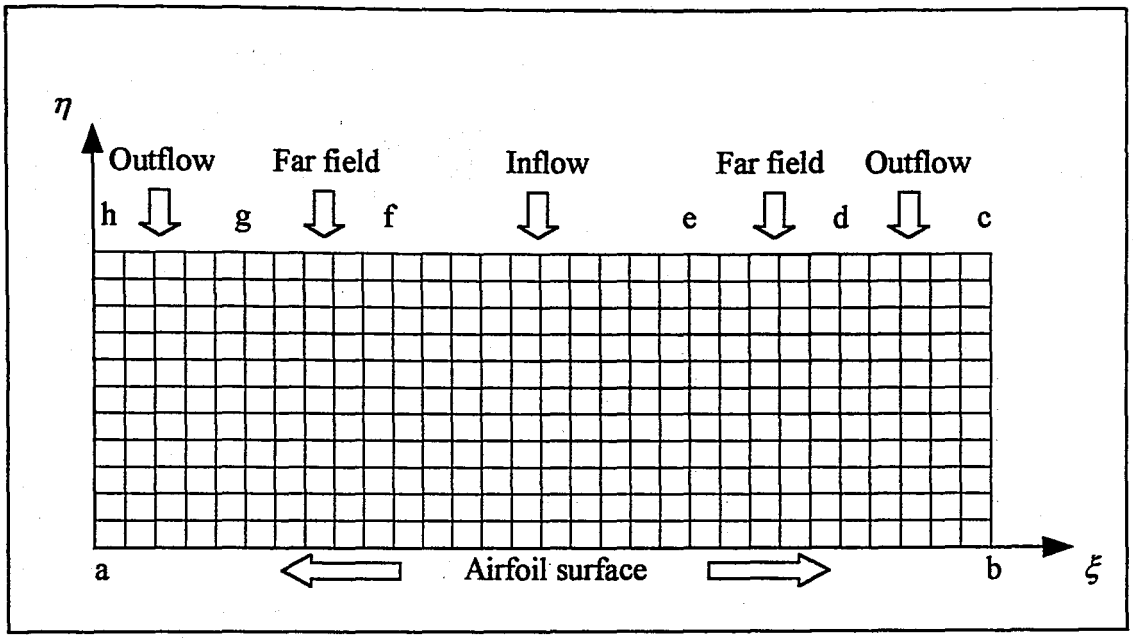


FIGURE 3.8. Computational domain with equal step sizes $\Delta\xi$ and $\Delta\eta$.

3.6.1. Airfoil Surface (a – b)

Stream function: No slip boundary conditions (u=0, v=0) are applied to the airfoil surface. A solid surface can be considered as a streamline, therefore the stream function is constant, and its value may be assigned arbitrarily. In the present study stream function value on the airfoil surface is chosen as

$$\psi = 0 \tag{3.36}$$

Vorticity: The boundary condition for vorticity can not be specified directly, since it does not exist. Therefore, it must be constructed by the use of the stream function equation with Taylor's series expansion of the stream function. Recall that stream function equation is,

$$\frac{1}{J^2}\left[\left(\alpha\frac{\partial^2\psi}{\partial\xi^2}\right)-\left(2\beta\frac{\partial^2\psi}{\partial\xi\partial\eta}\right)+\left(\gamma\frac{\partial^2\psi}{\partial\eta^2}\right)\right]+P^*\left(\frac{\partial\psi}{\partial\xi}\right)+Q^*\left(\frac{\partial\psi}{\partial\eta}\right)=-\zeta$$

Since the stream function is constant along the boundary of the airfoil surf
is zero for the present study Eq. (3.37) simplifies to

$$\frac{1}{J^2}\left(\gamma\frac{\partial^2\psi}{\partial\eta^2}\right)=-\zeta$$

This boundary condition is also given in Eq. (30a) of reference [28].

Now to obtain an expression for the second order derivative in Eq. (3.
Taylor's series expansion is considered as follows:

$$\psi_{i,j+1}=\psi_{i,j}+\left.\frac{\partial\psi}{\partial\eta}\right|_{i,j}\Delta\eta+\left.\frac{\partial^2\psi}{\partial\eta^2}\right|_{i,j}\frac{\Delta\eta^2}{2}+\ldots$$

Along the boundary velocity components, u and v are equal to zero, therefore

$$\frac{\partial\psi}{\partial\eta}=0$$

Substituting Eq.(3.38) into the Eq.(3.39) and considering Eq.(3.40)
boundary condition for vorticity is obtained:

$$\zeta = \frac{2}{\Delta\eta^2}\frac{\gamma}{I^2}\left(\psi_{i,j} - \psi_{i,j+1}\right) \tag{3.41}$$

3.6.2. Far Field Boundary (g - f, e - d)

$$\psi = U(y(\xi,\eta)\cos\theta - x(\xi,\eta)\sin\theta) \tag{3.42}$$

$$\zeta = 0 \tag{3.43}$$

3.6.3. Inflow boundary (e – f)

$$\psi = U(y(\xi,\eta)\cos\theta - x(\xi,\eta)\sin\theta) \tag{3.44}$$

$$\zeta = 0 \tag{3.45}$$

3.6.4. Outflow boundary (h - g, d - c)

$$\psi = U(y(\xi,\eta)\cos\theta - x(\xi,\eta)\sin\theta) \tag{3.46}$$

$$\zeta = 0 \tag{3.47}$$

These above conditions can only be applied if the outflow boundary is sufficiently far away from the body in domain. The reason for this assumption is that, these conditions should not influence the variables close to the airfoil surface [27, 29].

3.6.5. Branch Cut (a - h, b - c)

The periodicity condition is imposed along the boundary.

4. COMPUTATIONAL TECHNIQUES

The task of obtaining solutions to the governing equations of fluid mechanics represents one of the most challenging problems in science and engineering. In most instances, the mathematical formulations of the fundamental laws of fluid mechanics are expressed as partial differential equations (PDE). Second order partial differential equations appear frequently and, therefore, are of particular interest in fluid mechanics and heat transfer. Generally, the governing equations of fluid mechanics form a set of coupled, non-linear PDEs, which must be solved within an irregular domain subject of various initial and boundary conditions.

In many instances, analytical solutions of the equations of fluid mechanics are limited. This is further restricted due to the imposed boundary conditions. For example, a PDE subject to a Dirichlet boundary condition (ie. values of the dependent variable on the boundary are specified) may have an analytical solution. However, the same PDE with Neumann Boundary condition (where normal gradients of the dependent variable on the boundary are specified) may not have an analytical solution.

The Navier-Stokes equations are non-linear because of the convective terms. If the governing equations of fluid flow are discretised by the use of the finite difference method, the following non-linear system of equations is obtained:

$$A(V)V=B \tag{4.1}$$

where V is the vector of unknown nodal values or coefficients, A is the algebraic coefficients arising from the discretisation and, in general, may depend on the solution (V) itself, and B is made up of algebraic coefficients associated with discretisation and known values of V, eg. that are given by boundary conditions.

$A(V)$ is a full or a sparse matrix (few non-zero elements) due to the method used for the discretisation. In the present study finite difference method is used for discretisation and as a result $A(V)$ became a sparse matrix, in which the non-zero terms are close to the diagonal.

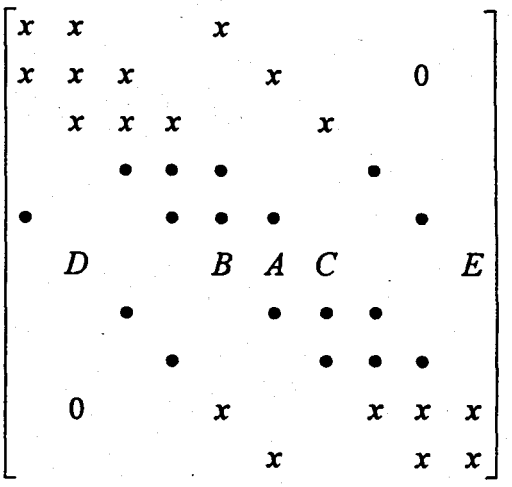


FIGURE 4.1. Typical non-zero element in A after a finite difference discretisation.

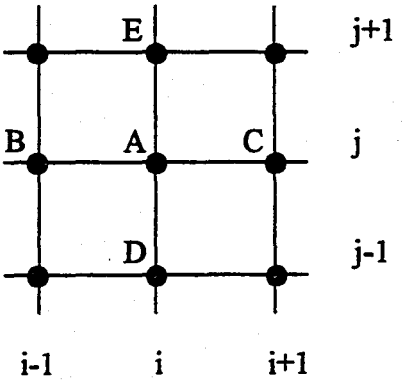


FIGURE 4.2. Stencil associated with the matrix shown in Figure 4.1.

It is possible to solve the Eq. (4.1) by using the common iterative techniques, but the solution can be more efficient if the system is converted to a linear system each step by Newton's method and linear solver GMRES is used in order to obtain the unknowns in the system.

4.1. Newton's Method

Newton's method is a powerful technique for solving non-linear systems as discussed above. In order to introduce this method let us consider the non-linear system in Eq. (4.1). It can be written as follows:

$$R=A(V)V-B=0 \tag{4.2}$$

Then the Newton's method can be written as,

$$V^{(k+1)}=V^{k}-(J^{(k)})^{-1}R^{(k)} \tag{4.3}$$

where k is the iteration level and $J(k)$ is the Jacobian. An element of $J(k)$ is

$$J_{ij}=\frac{\partial R_i^{(k)}}{\partial V_j^{(k)}} \tag{4.4}$$

Equation (4.3) can be written in the form

$$J^{(k)}\Delta V^{(k+1)}=-R^{(k)} \tag{4.5}$$

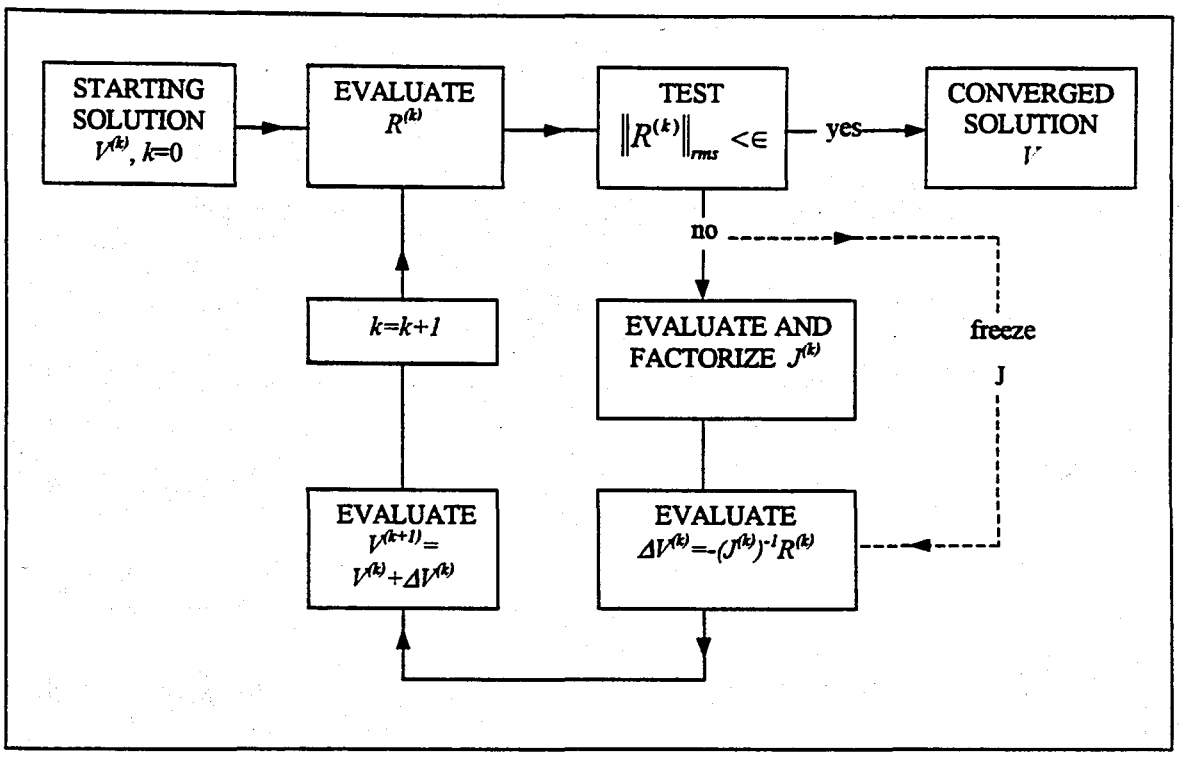


FIGURE 4.3. Schematic representation of Newton's method [30].

Equation (4.5) is a linear system of equations that is to be solved for the correction vector,

$$\Delta V^{(k+1)} = V^{(k+1)} - V^{(k)} \tag{4.6}$$

at each stage of the iteration. In the present study, the Krylov Subspace method GMRES, with different type of preconditioners such as ILU or Jacobi, is applied to the linear system.

If an initial guess for the unknown vectors isn't made, the initial unknown vector can be considered zero. If the current iteration $V^{(k)}$ is sufficiently close to the converged solution V_c, Newton's method demonstrates quadratic convergence and quadratic convergence implies,

$$\left\| V^{(k+1)} - V_c \right\| \approx \left\| V^{(k)} - V_c \right\|^2 \tag{4.7}$$

The important disadvantage of the Newton's method is that, its convergence is not guaranteed. The correction factor may be quite high that it can throw the current solution outside of the radius of the convergence, if a damping factor is not utilized. This means that converged solution can't be achieved.

The damping factor can be applied to Eq. (4.6) and the resultant equation is as follows:

$$V^{(k+1)} = V^{(k)} + \omega \Delta V^{(k+1)} \tag{4.8}$$

The damping factor ω may be chosen between zero and one ($0<\omega<1$). For small values of ω the convergence rate can be drastically slow and time is sacrificed. In order to obtain optimum values for ω, techniques have been developed for dynamic achievement of the damping factor as the iteration proceeds.

4.2. Conventional Techniques for the Solution of Linear Systems

The solutions of linear systems that arise at each correction of non-linear solvers often constitute the principal time and storage bottleneck. Techniques for the solution of the linear systems of equations, which has the form of $AV=B$, can be classified as direct and iterative methods.

4.2.1. Direct Methods

In the absence of round-off errors, direct methods can compute the exact solution of any non-singular problem in a finite number of operations. They use some form of factorisation of the system matrix and are well suited for the solution of dense systems.

If we consider the following linear system

$$AV=B \tag{4.9}$$

The coefficient matrix A is factorised by the application of one of the direct methods, namely Gaussian elimination method, into lower triangular, L, and the upper triangular U. The factored form of (4.9) is solved as

$$UV=L^{-1}B \tag{4.10}$$

However, the systems generated by the discretisation of partial differential equations are sparse. When direct methods are applied, the resulting factored forms have more non-zero entries than the original matrix. Although direct methods can still be practicable for some two-dimensional problems, the relative fill-in for the case of three dimensions makes it imperative to consider iterative techniques.

Examples of the direct methods are,

1. Gaussian Elimination
2. Gauss-Jordan Elimination
3. Matrix Inversion
4. LU decomposition

The major disadvantage of these methods is the enormous amount of arithmetic operations required to produce a solution. Some advanced direct methods have been proposed which require moderate computation time, but almost all of them have disadvantages. Usually these methods are limited by one or more restrictions such as the Cartesian coordinate system, a rectangular domain, the size of the coefficient matrix, a large storage requirement, boundary conditions, or difficulty of programming.

Another drawback of direct methods is the possible contamination of round-off errors in ill-conditioned systems. In contrast, iterative methods have the self-correcting feature that they use the matrix in its original form, although round-off errors can still delay convergence.

4.2.2. Iterative Methods

Iterative methods are distinguished by the manner in which a new iterate is obtained from the previous ones. They start with an initial guess and compute a sequence of approximate solutions converging to the exact solution. They preserve the sparsity of the original matrix. It is important, for an effective iterative method, to decide when to terminate iterations. The number of iterations required to achieve convergence depends on:

- The dominance of the diagonal coefficients,
- The initial solution vector,
- The algorithm used,
- The convergence criterion specified.

Examples of the iterative methods are,

1. Jacobi Iteration
2. Gauss-Seidel Iteration
3. Successive Over Relaxation (SOR)

4. Symmetric Successive Over Relaxation
5. Krylov Subspace Methods

4.2.2.1. Jacobi Iteration Method

The Jacobi method is easy to implement, but it is uneconomic since the convergence rate is slow. For diagonally dominant systems this method can be rated as efficient and convergence is guaranteed.

By the use of the method, the coefficient matrix A is split as follows:

$$A=D-L-U \tag{4.11}$$

where, D is the diagonal, $-L$ is the lower triangular part and, $-U$ is the upper triangular part of A.

After some manipulations Jacobi method can be described in matrix terms as

$$V^{k+1}=D^{-1}(L+U)V^{k}+D^{-1}B \tag{4.12}$$

The algorithm for Jacobi method is given in Figure 4.4.

In the present study, the Jacobi method is used as an optional preconditioner to ILU to improve the convergence of GMRES.

Given vector V_0

a) For $k=0,1,2,...$

b) *For* $i=1,2,...,N$

c) $t_i=0$

d) *For* $j=1,2,3,...,i-1,i+1,...,N$

e) $t_i = t_i + a_{ij} V_j^k$

f) $t_i=(b_i-t_i)/a_{ii}$

g) $V^{k+1}=t$

h) *Check for convergence: stop if satisfied*

FIGURE 4.4. Algorithm of Jacobi method [31].

4.3. GMRES

In the present study Generalized Minimal Residual Method (GMRES) has been used to solve the linear system arising from the Newton's method. GMRES algorithm, introduced by Saad and Schultz [32], is a projection method on Krylov subspaces. Before introducing GMRES method in details Krylov methods should be explained.

4.3.1. Krylov Methods

Krylov methods are effective for solving large sparse systems of linear equations that arise in many computational fluid dynamics applications. In contrast to some conventional iterative methods like relaxation methods or approximate factorisations, they do not require a priori estimates of parameters. They are easy to implement and their convergence can further be improved by preconditioning techniques [31].

4.3.1.1. Krylov Subspaces

The Krylov subspace can be explained by considering the eigenvector s of matrix A and an arbitrary vector v_0. The eigenvectors of A are characterised by a linear relation between As and s. However, for an arbitrary vector v_o this type of linear relation does not exist. Instead, the following type of relation can be written.

$$v_r=A^r v_0 \tag{4.18}$$

Let the minimum value of r, which makes these vectors dependent, be denoted by m, then the form between the vector v_o and the matrix A becomes:

$$p(A)\, v_0=(A^m+p_{m-1}A^{m-1}+p_{m-2}A^{m-2}+\ldots\ldots\ldots+p_0 I)v_0=0 \tag{4.19}$$

This monic polynomial p is called the minimal polynomial of the vector v_0 and m is called the grade of v_0 with respect to A. The subspace described by the sequence of Eq. (4.18) is referred as the Krylov subspace and represented as;

$$K_m(A,v_0)=span\{v_0,Av_0,A^2v_0,\ldots\ldots\ldots, A^{m-1}v_0\} \tag{4.20}$$

Most Krylov methods require the process of building an orthogonal basis of the subspace K_m [31].

To understand the algorithm of GMRES we should first introduce the Arnoldi process, which builds an orthonormal basis of the subspace K_m for general non-Hermitian matrices using Gram-Schmidt orthogonalization method. Arnoldi presented his method in

1951 as a means of reducing a dense matrix into Hessenberg form. Later it was discovered that, this method leads to an efficient technique for approximating the eigenvalues of large sparse matrices.

At each step, the algorithm multiplies the previous Arnoldi vector v_j by A and then orthonormalizes the resulting vector w_j against all previous v_j's by a standard Gram-Schmidt procedure. After the k steps of the process, k×(k+1) matrix H_k and an orthonormal system V_{k+1} are generated. Here, H_k is the same as with the Hessenberg matrix plus a row whose elements are zero except $(k+1,k)^{th}$ element.

The vectors v_j and the matrix H_k satisfy the relation;

$$AV_k = V_{k+1} H_k \tag{4.21}$$

Recall that, we were trying to obtain the solution of the following type of linear system

$$Ax = f \tag{4.22}$$

using the orthonormal system V_m, we seek an approximate solution which has the form, $x_m = x_0 + z_m$ where x_0 is initial guess of the solution and z_m is an element of the Krylov subspace $K_m = span\{ r_0, Ar_0, A^2r_0, ..., A^{m-1}r_0 \}$, where $r_0 = f - A x_0$. Now it is time to recognise that the residual vector $r_m = f - A x_m$ is orthogonal to K_m.

In order to obtain the approximate solution x_m, the following least square problem must be solved;

$$\min_{z \in K_k} \| f - A[x_0 + z] \| = \min_{z \in K_k} \| r_0 - Az \| \tag{4.23}$$

If $z=V_k y$, the above norm can be written as a function of y;

$$F(y) = \left\| \beta^* v_1 - A V_k y \right\| \tag{4.24}$$

where $\beta^* = \left\| r_0 \right\|$, then using Eq. (4.22), Eq. (4.24) becomes;

$$F(y) = \left\| V_{k+1} \left[\beta^* e_1 - H_k y \right] \right\| \tag{4.25}$$

The vector e_1 is the first column of the (k+1) × (k+1) identity matrix and as V_{k+1} is orthonormal, we get

$$F(y) = \left\| \beta^* e_1 - H_k y \right\| \tag{4.26}$$

Then the solution of the least square problem is obtained;

$$x_k = x_0 + V_k y_k \tag{4.27}$$

where y_k minimizes the function F(y), defined by Eq. (4.26) over $y \in R^k$ [31].

1. Compute $r_0 = b - Ax_0, \beta^* = \|r_0\|_2$, and $v_1 = r_0 / \beta^*$
2. Define the $(k+1) \times k$ matrix $H_k = \{h_{ij}\}_{1 \le i \le k+1, 1 \le j \le k}$. Set $H_k = 0$
3. For $j = 1,2,.....,k$ Do:
4. Compute $w_j = A\, v_j$
5. For $i = 1,.....j$ Do:
6. $h_{ij} = (w_j, v_i)$
7. $w = w_j - h_{ij}\, v_i$
8. End Do
9. $h_{j+1,j} = \|w_j\|_2$. If $h_{j+1,j} = 0$ set $k = j$ and go to 12
10. $v_{j+1} = w_j / h_{j+1,j}$
11. End Do
12. Compute y_k the minimizer of $\|\beta^* e_1 - H_k y\|_2$ and $x_k = x_0 + V_k y_k$

FIGURE 4.6. The Algorithm of GMRES [32].

The advantages of the GMRES method can be written as follows:

1. GMRES does not require the Jacobian matrix A. The matrix-vector product is sufficient.

2. The solution vector can be always obtained even for the non-symmetric systems if the dimension of the Krylov subspace is chosen as the original dimension of the Jacobian matrix.

3. The storage requirement and operations per step are less than some mathematically equivalent algorithms.

Besides these advantages, there exists the problem of increasing number of stored vectors as k increases. In order to overcome this undesired situation the algorithm can be restarted every m steps by setting $x_0 = x_n$ or a powerful preconditioner can be utilized. In the next section, the concept of preconditioner will be introduced.

4.4. ILU Preconditioning Technique

Numerical experience and existing theory show that the convergence of iterative methods depends on the distribution of the eigenvalues. The better the clustering of the eigenvalues, the faster these methods will converge. Transforming the linear system into one that has the same solution, but a more favourable eigenvalue spectrum does this. This transformation is done by the operation of a matrix call the preconditioner [31].

A preconditioner is any form of implicit or explicit modification of an original linear system, which makes it easier to solve by a Krylov subspace method or a given iterative method and require fewer steps to converge than the original system.

Now consider the following system ;

$$Ax = f \tag{4.28}$$

where A is the coefficient matrix and x is the unknown vector. If we multiply both sides of the system by non-singular matrix, the resulting system will still has the same solution.

$$(M)^{-1} Ax = (M)^{-1} f \tag{4.29}$$

where M is the preconditioner. It may be said that M is a matrix that approximates A and more easily invertible than A.

The simplest of all preconditioners is the Jacobi preconditioner, where M is the diagonal matrix whose elements on the diagonal are equal to those of the matrix A. The Jacobi preconditioner is considered to be the simplest one since obtaining the inverse of a

diagonal matrix is a simple process. In the present study, Jacobi and ILU have been used as preconditioner.

LU factorization is a direct method, which consists of factorization of original system matrix into the product of lower triangular and upper triangular matrices by using the Gaussian elimination process.

We can factorize the matrix A in Eq. (4.28) by

$$A=LU \tag{4.30}$$

Then it can be written that $LUx=f$, and we obtain two staggered systems, which can be solved easily because of the lower and upper triangular structures of the matrices namely L and U by applying a forward and a backward substitutions;

$$Ly=f \tag{4.31}$$

$$Ux=y \tag{4.32}$$

Since L and U are usually less sparse than A, the LU decomposition of A requires a large number of calculations and memory storage. Therefore we have to get matrices $K=\underline{L}\ \underline{U}$ which resemble A with $\underline{L}$ and $\underline{U}$ almost as spares as matrix A. This can be obtained by neglecting some predetermined elements in the $\underline{L}$ and $\underline{U}$ matrices during Gaussian elimination process. Zeros may occur in arbitrary off-diagonal places in $\underline{L}$ and $\underline{U}$. By choosing some non-zero elements the elements to drop at each step must be specified. The only restriction on the zero pattern set P, such that

$$P \subset \{(i,j) | i \neq j, 1 \leq i,j \leq n\}, \tag{4.33}$$

This method is called Incomplete LU factorization (ILU) and its procedure is shown in Figure 4.7.

The algorithm produces factors $\underline{L}$ and $\underline{U}$ such that,

$$A = \underline{L}\,\underline{U} - R \tag{4.34}$$

inwhich R is the matrix of the elements that are dropped during the incomplete elimination process.

1. *For k=1,2,.....,n-1 Do:*
2. *For i=k+1,n and if (i,k) ∉P Do:*
3. $a_{ik} = a_{ik}/a_{kk}$
4. *For j=k+1,.....,n and for (i,j) ∉P Do:*
5. $a_{ij} = a_{ij} - a_{ik} * a_{kj}$
6. *End do*
7. *End do*
8. *End do*

FIGURE 4.7. Algorithm of Static Pattern ILU.

We can make some manipulations to obtain a linear system that the coefficient matrix is a very close approximation of the identity matrix. Identity matrix can be achieved by multiplying the matrix by its reciprocal but usually this not a simple procedure. Let us consider the following linear system that the coefficient matrix A is a general sparse matrix;

$$Ax=f \tag{4.35}$$

A general incomplete LU factorization process computes a sparse lower triangular matrix $\underline{L}$ and a sparse upper triangular matrix $\underline{U}$ so that the product of $\underline{L}$ and $\underline{U}$ have the same elements of A at some predetermined locations depending on the non-zero pattern of the matrix A.

If both sides of the Eq. (4.35) are multiplied by $(\underline{L}\,\underline{U})^{-1}$ we obtain the following equation:

$$(\underline{L}\,\underline{U})^{-1} Ax = (\underline{L}\,\underline{U})^{-1} f \tag{4.36}$$

The resultant equation can be solved very easily compared to the solution procedure of Eq. (4.35) although the solution of Eq. (4.36) and Eq. (4.35) are the same. Now let us write the Eq. (4.36) in a different form, such that

$$\underline{U}^{-1}\underline{L}^{-1} Ax = \underline{U}^{-1}\underline{L}^{-1} f \tag{4.37}$$

By denoting $(\underline{L}^{-1} A)$ as Y, the matrix Y can be obtained very easily by making a forward substitution since $\underline{L}$ is a lower triangular matrix. After substitution of Y in Eq. (4.37), the system has the following form:

$$\underline{U}^{-1} Yx = \underline{U}^{-1}\underline{L}^{-1} f \tag{4.38}$$

And now let us denote $(\underline{U}^{-1} Y)$ by Z, the matrix Z can be obtained by making a backward substitution since $\underline{U}$ is an upper triangular matrix.

$$\underline{U}^{1} Y = Z \tag{4.39}$$

If we denote the right hand side of the Eq. (4.37) by t, it can be obtained by applying the same procedure.

$$t = \underline{U}^{1}\underline{L}^{1} f \tag{4.40}$$

After the substitution of Z and t in Eq. (4.38), the new preconditioned system gets the following form:

$$Zx = t \tag{4.41}$$

Finally we obtained a new linear system, which is more easy to solve since its coefficient matrix (Z) is a very close approximation to the identity matrix and whose solution is the same as the original one.

4.5. Finite Difference Discretization

The differentials of the dependent variables in the governing equations must be expressed as approximate expressions, so that the flow field variables can be solved at the specific, discrete grid points in the computational flow domain. This can be achieved by converting the governing equations of the problem into a system of algebraic equations.

Recall that, the governing equations of the present study in computational domain have been given in section three and these are;

Stream function equation,

$$\frac{1}{I^2}\left[\left(\alpha\frac{\partial^2\psi}{\partial\xi^2}\right)-\left(2\beta\frac{\partial^2\psi}{\partial\xi\partial\eta}\right)+\left(\gamma\frac{\partial^2\psi}{\partial\eta^2}\right)\right]+P^*\left(\frac{\partial\psi}{\partial\xi}\right)+Q^*\left(\frac{\partial\psi}{\partial\eta}\right)=-\zeta \tag{4.42}$$

Vorticity transport equation,

$$\frac{\partial\zeta}{\partial t}+\frac{1}{I}\left[\left(\frac{\partial\psi}{\partial\eta}\right)\left(\frac{\partial\zeta}{\partial\xi}\right)-\left(\frac{\partial\psi}{\partial\xi}\right)\left(\frac{\partial\zeta}{\partial\eta}\right)\right]$$

$$=\frac{1}{\mathrm{Re}}\left\{\frac{1}{I^2}\left[\left(\alpha\frac{\partial^2\zeta}{\partial\xi^2}\right)-\left(2\beta\frac{\partial^2\zeta}{\partial\xi\partial\eta}\right)+\left(\gamma\frac{\partial^2\zeta}{\partial\eta^2}\right)\right]+P^*\left(\frac{\partial\zeta}{\partial\xi}\right)+Q^*\left(\frac{\partial\zeta}{\partial\eta}\right)\right\} \tag{4.43}$$

In this study, these equations have been discretized by using the finite difference method. This method is based on the Taylor's series expansions. In the interior domain second order central difference and at the boundaries first order difference formulations have been used. These formulations are given schematically in Figure 4.8 and tabulated in Appendix C.

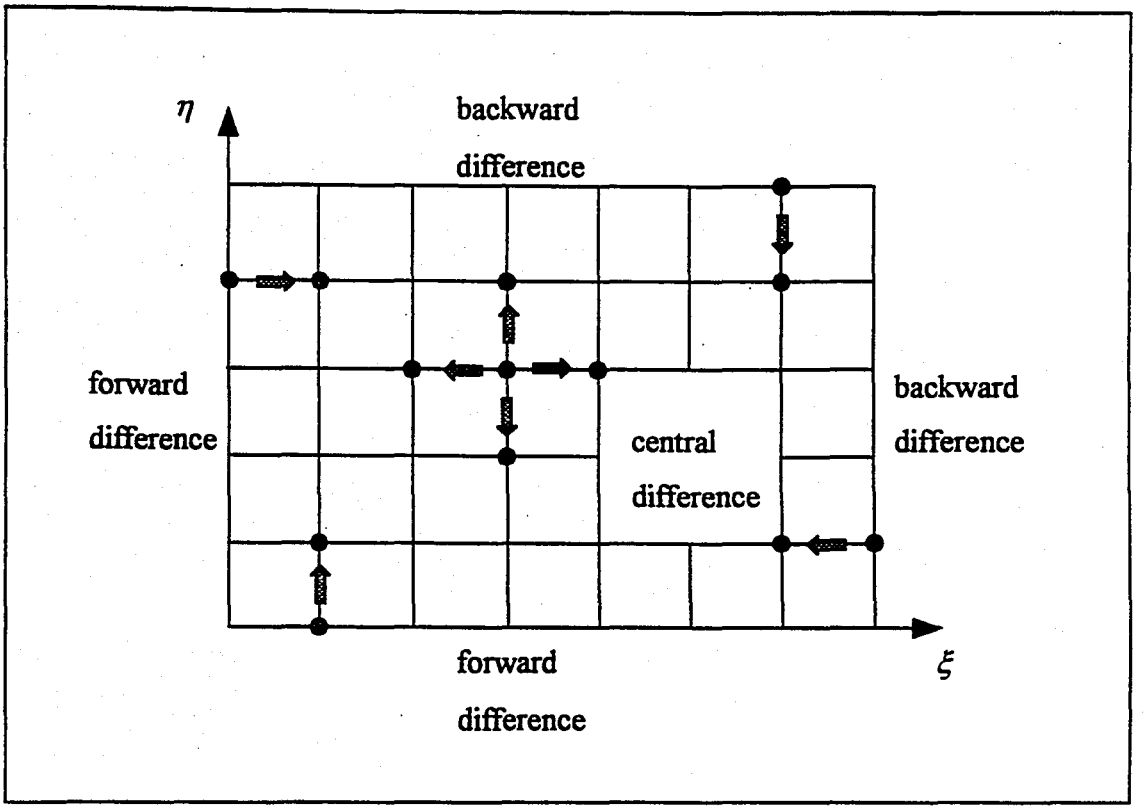


FIGURE 4.8. Finite difference formulations in computational domain.

Using the expressions in Appendix C with stream function-vorticity formulation, the following algebraic equations for the governing equations in the interior computational domain are obtained:

Discretized stream function equation,

$$\frac{1}{I^2}\left[\alpha\left(\frac{\psi_{i+1,j}-2\psi_{i,j}+\psi_{i-1,j}}{\Delta\xi^2}\right)-2\beta\left(\frac{\psi_{i+1,j+1}-\psi_{i+1,j-1}-\psi_{i-1,j+1}+\psi_{i-1,j-1}}{4\Delta\xi\Delta\eta}\right)\right.$$

$$\left.+\gamma\left(\frac{\psi_{i,j+1}-2\psi_{i,j}+\psi_{i,j-1}}{\Delta\eta^2}\right)\right]+P^*\left(\frac{\psi_{i+1,j}-\psi_{i-1,j}}{2\Delta\xi}\right)+Q^*\left(\frac{\psi_{i,j+1}-\psi_{i,j-1}}{2\Delta\eta}\right)=-\zeta \tag{4.44}$$

Discretized vorticity transport equation,

$$\frac{\zeta_{i,j}^{n+1}-\zeta_{i,j}^{n}}{\Delta t}+\frac{1}{I}\left[\left(\frac{\psi_{i,j+1}^{n}-\psi_{i,j-1}^{n}}{2\Delta\eta}\right)\left(\frac{\zeta_{i+1,j}^{n}-\zeta_{i-1,j}^{n}}{2\Delta\xi}\right)-\left(\frac{\psi_{i+1,j}^{n}-\psi_{i-1,j}^{n}}{2\Delta\xi}\right)\left(\frac{\zeta_{i,j+1}^{n}-\zeta_{i,j-1}^{n}}{2\Delta\eta}\right)\right]$$

$$=\frac{1}{\mathrm{Re}}\left\{\frac{1}{I^{2}}\left[\alpha\left(\frac{\zeta_{i+1,j}^{n}-2\zeta_{i,j}^{n}+\zeta_{i-1,j}^{n}}{\Delta\xi^{2}}\right)-2\beta\left(\frac{\zeta_{i+1,j+1}^{n}-\zeta_{i+1,j-1}^{n}-\zeta_{i-1,j+1}^{n}+\zeta_{i-1,j-1}^{n}}{4\Delta\xi\Delta\eta}\right)\right.\right. \tag{4.45}$$

$$\left.\left.+\gamma\left(\frac{\zeta_{i,j+1}^{n}-2\zeta_{i,j}^{n}+\zeta_{i,j-1}^{n}}{\Delta\eta^{2}}\right)\right]+P^{*}\left(\frac{\zeta_{i+1,j}^{n}-\zeta_{i-1,j}^{n}}{2\Delta\xi}\right)+Q^{*}\left(\frac{\zeta_{i,j+1}^{n}-\zeta_{i,j-1}^{n}}{2\Delta\eta}\right)\right\}$$

4.6. Computational Procedure

For the simulation of the unsteady flow around NACA 0020 airfoil, a mesh of 41x35 grid size has been generated. In the numerical formulation of the vorticity transport equation FTCS (forward time/ central space) explicit method is used. The time derivative has been discretized using the forward difference approximation and space derivatives have been discretized using central differencing.

Since explicit method is used in the formulation of the vorticity transport equation, vorticity is solved algebraically at each time step. The stream function equation, which is classified as elliptic equation, is solved by GMRES.

The following computational procedure is performed for the solution of the stream function and vorticity-transport equations:

a) ψ^{n} and ζ^{n} are the solutions of the steady state problem. Now they are used as the initial conditions for the unsteady problem. The boundary conditions must be specified.

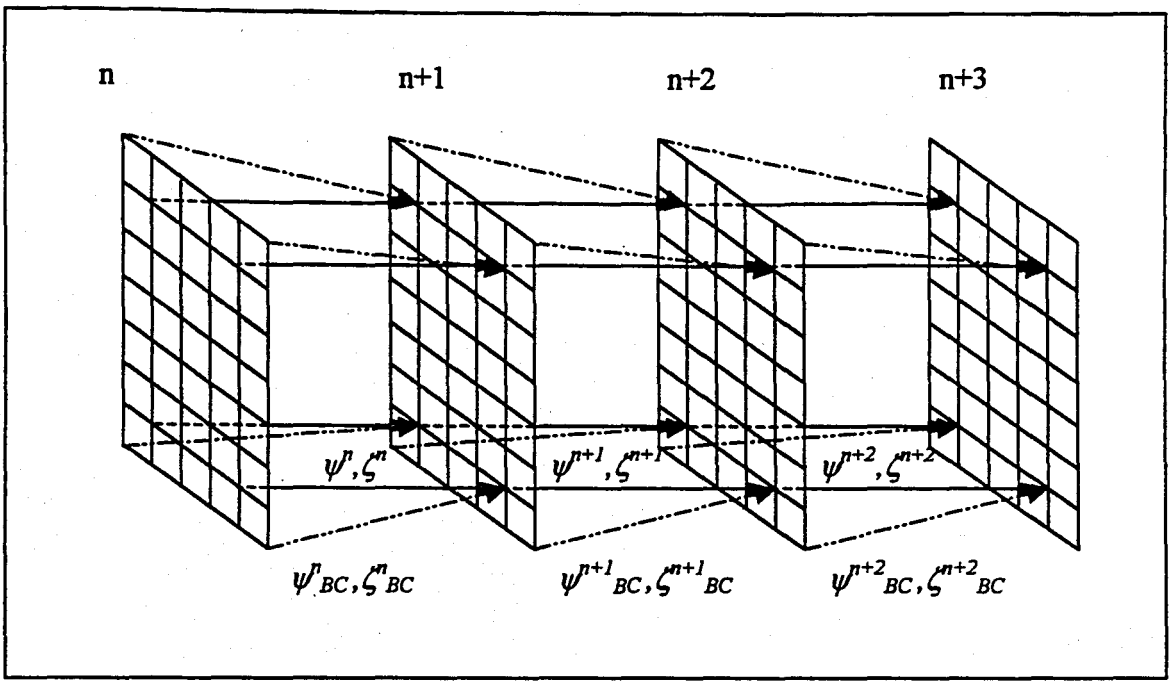


FIGURE 4.9. Computational procedure of FTCS explicit method.

b) The computation begins with the solution of Eq. (4.45) within the domain. ζ^{n+1} is calculated from the initial and boundary conditions of the problem.

c) The newly calculated values of vorticity (ζ^{n+1}) are inserted into the Eq. (4.44) in order to obtain stream function at level n+1 (ψ^{n+1}).

The same procedure repeats itself until the desired time step is reached.

5. RESULTS AND DISCUSSION

In the present study, the technique of generating the transformed coordinates as solutions of an elliptic differential system in the physical plane has been applied to arbitrarily shaped bodies. The elliptic equations for the coordinates are solved in finite-difference approximation by successive overrelaxation (SOR) iteration.

This method of body fitted curvilinear coordinate generation has been used to construct a finite-difference solution of the full, incompressible, steady and unsteady Navier-Stokes equations for the laminar viscous flow around two-dimensional NACA 0020 airfoil.

In this section some problems with published data are solved in order to validate the governing equations and numerical code used in the solution of the present study. The first problem is the separating flow over a backward facing step and the second one is the steady viscous flow around a circular cylinder. Finally, steady and unsteady, incompressible two-dimensional, viscous flow fields around NACA 0020 airfoil have been calculated and simulated.

5.1. Test Problem #1 - Flow past Backward Facing Step

In this study steady, two dimensional, laminar, incompressible viscous flow past a backward facing step has been solved. The continuity and vorticity transport equations in transformed plane in terms of stream function and vorticity (Eq.(3.30) and Eq.(3.31) respectively) constitute the governing equations for this problem. In order to solve the system of the finite difference equations, a transformation from the L shaped physical domain to rectangular computational domain has been performed. The geometry of the problem and the grid point distribution within the domain are shown in Figures 5.1 and 5.2.

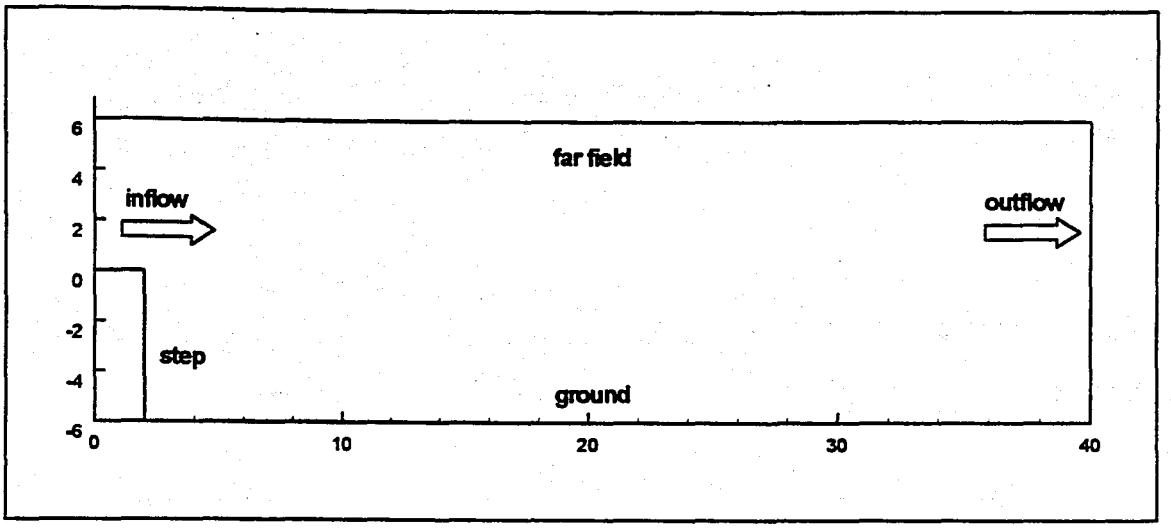


FIGURE 5.1. Geometry and boundary conditions of the backward facing step.

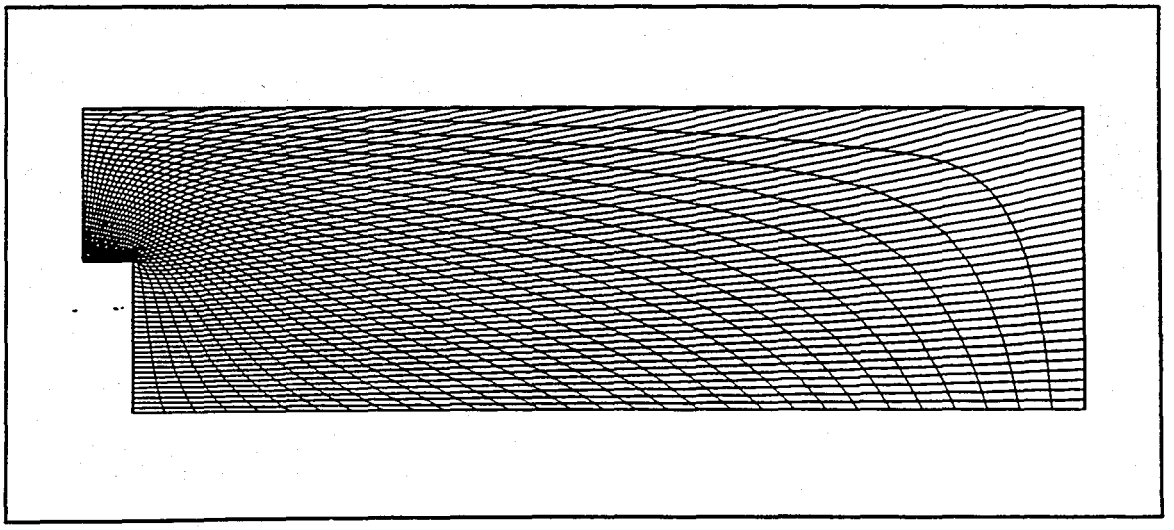


FIGURE 5.2. Grid point distribution within the L shaped physical domain, 61x 31 grid.

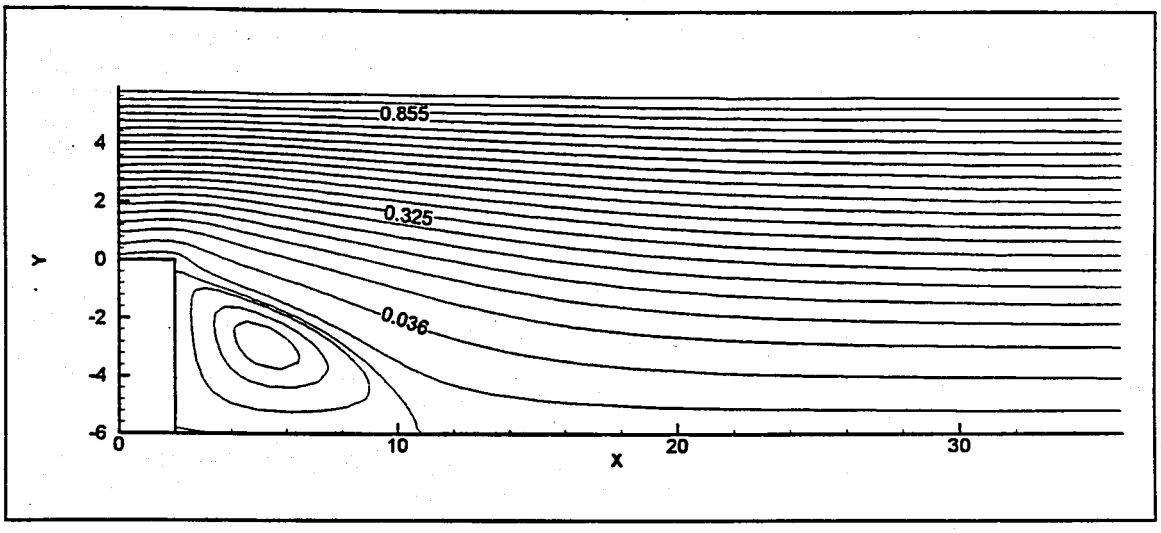


FIGURE 5.3. Streamline contours at Re=10, with 61x31 grid points.

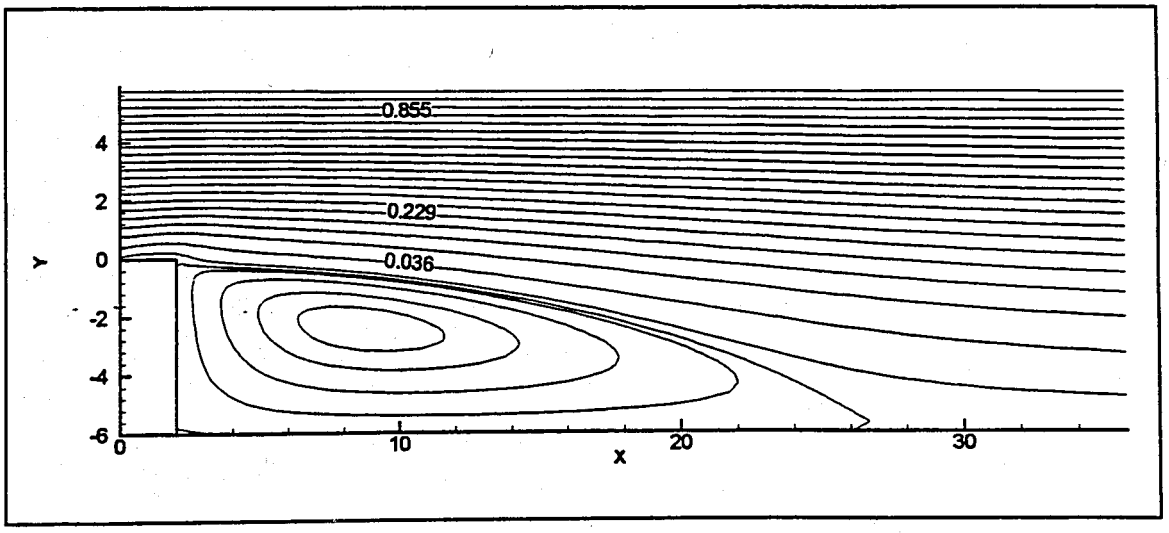


FIGURE 5.4. Streamline contours at Re=50, with 61x31 grid points.

The problem has been solved using the 61 x 31 grid at Reynolds numbers 10 and 50. The resulting streamline contours are shown in Figures 5.3 and 5.4. Furthermore, the calculated velocity field vector plots at Re=10 and 50 are shown in Figure 5.5.

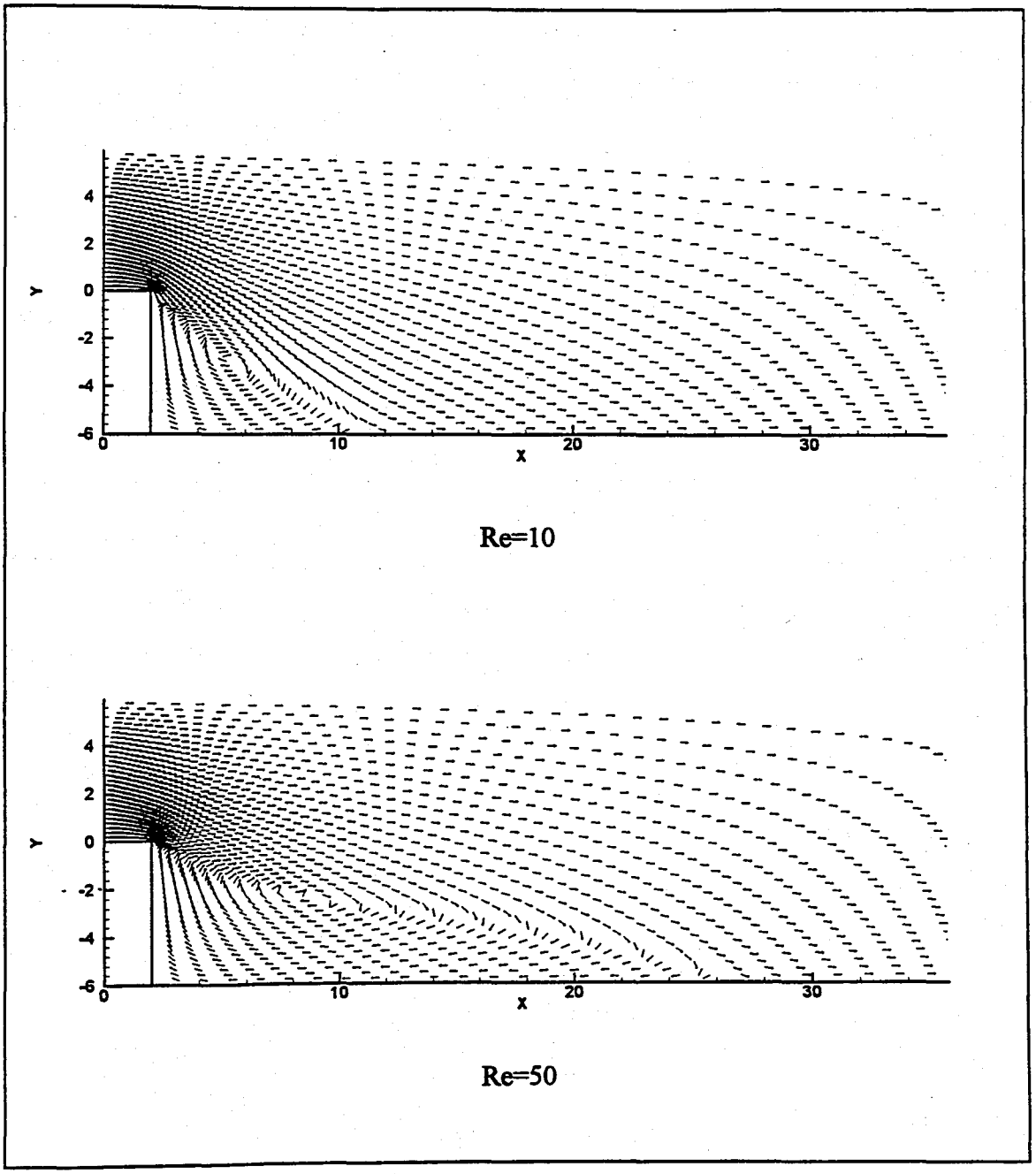


FIGURE 5.5. Velocity field vector plots at Re=10 and 50, with 61x31 grid points.

5.2. Test Problem #2 - Flow around a Circular Cylinder

In this study numerical solutions have been obtained for steady viscous flow past a circular cylinder at Reynolds numbers 20 and 200. Again the stream function and vorticity transport equations in transformed plane (Eq.(3.30) and Eq.(3.31) respectively) constitute the governing equations for this problem.

As in the former study, a transformation from the physical domain to rectangular computational domain has been performed. The geometry of the problem and the grid points distribution are shown in Figures 5.6 and 5.7.

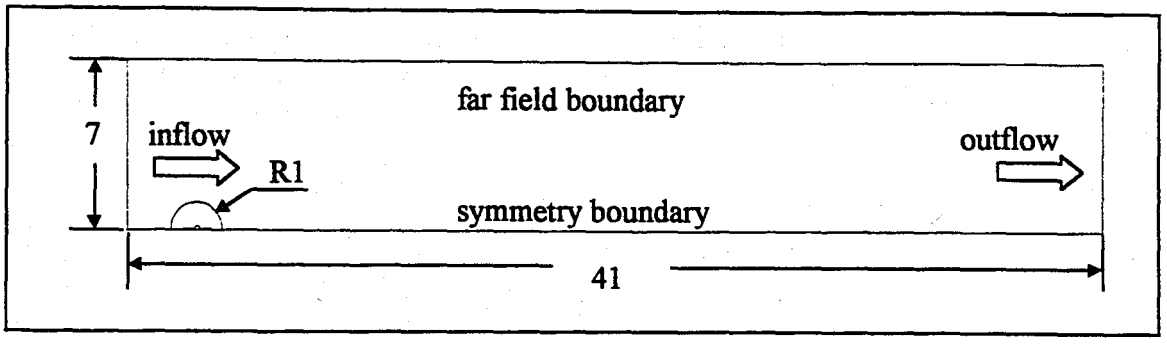


FIGURE 5.6. Geometry and boundary conditions of the problem.

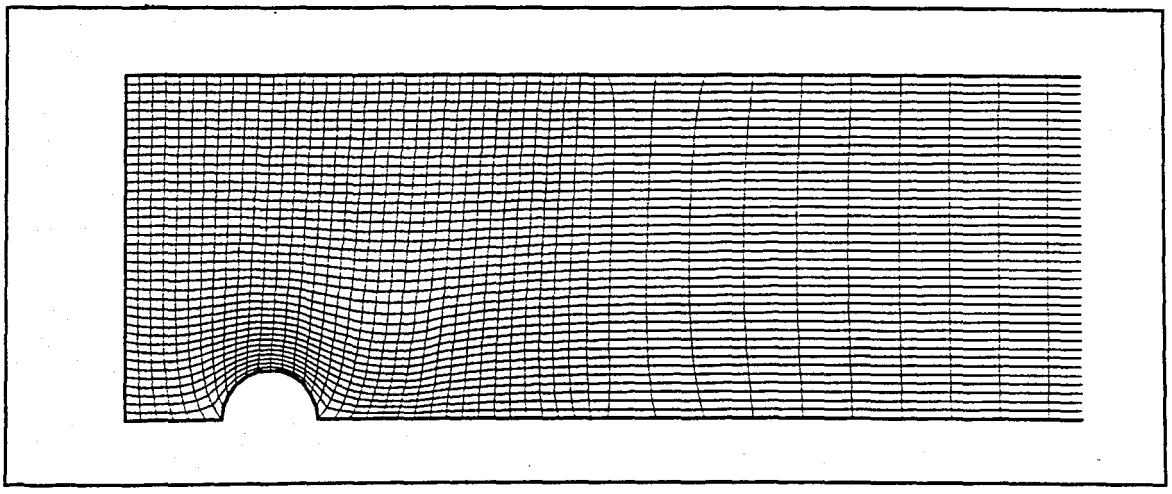


FIGURE 5.7. Grid point distribution in physical domain, 69x 40 grid.

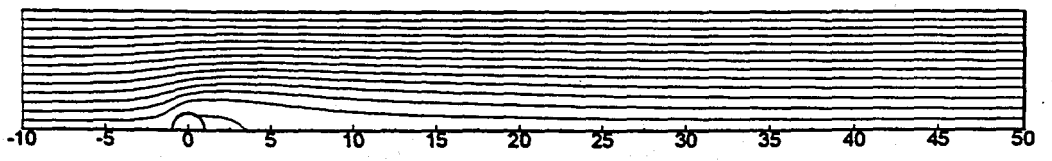


FIGURE 5.8. Streamline contours at Re=20, grid size 69 x 40.

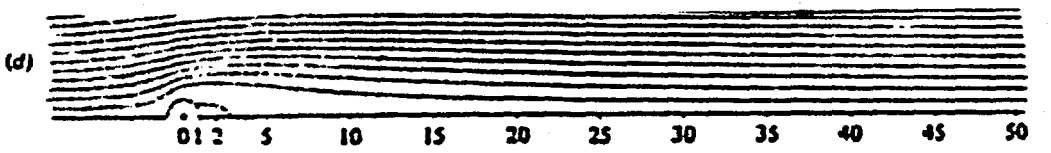


FIGURE 5.9. Streamline contours at Re=20 [33].

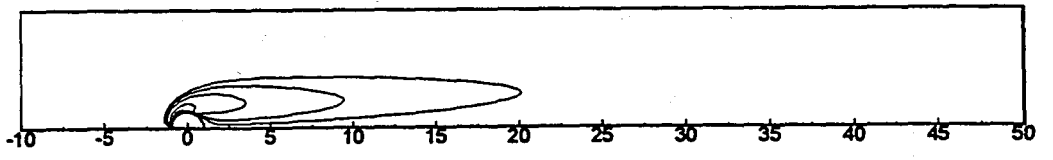


FIGURE 5.10. Vorticity contours at Re=20, grid size 69 x 40.

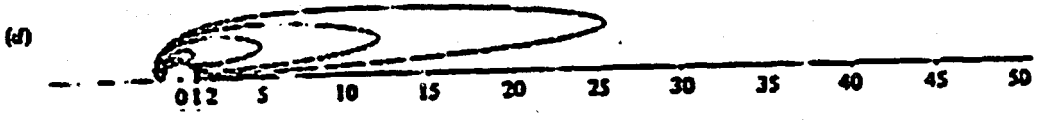


FIGURE 5.11. Vorticity contours at Re=20 [33].

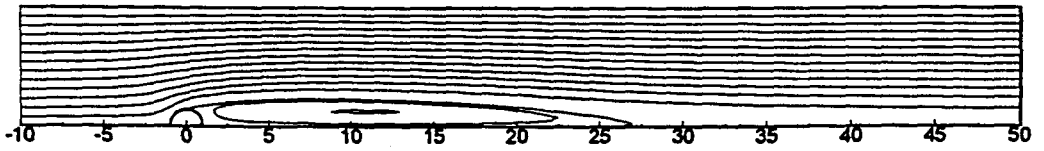


FIGURE 5.12. Streamline contours at Re=200, grid size 69 x 40.

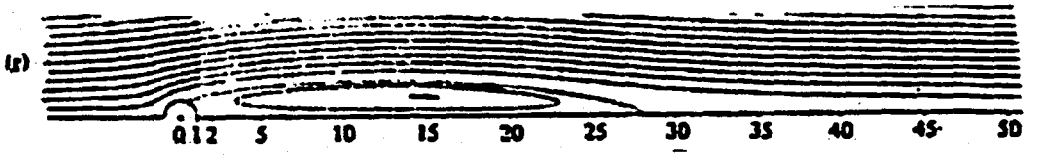


FIGURE 5.13. Streamline contours at Re=200 [33].

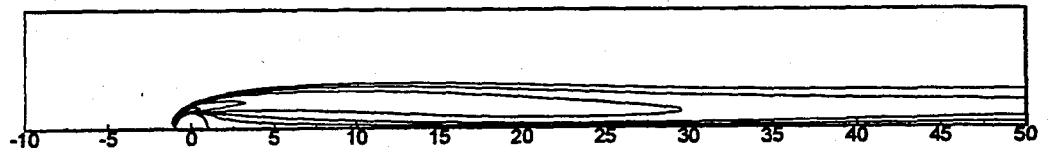


FIGURE 5.14. Vorticity contours at Re=200, grid size 69 x 40.

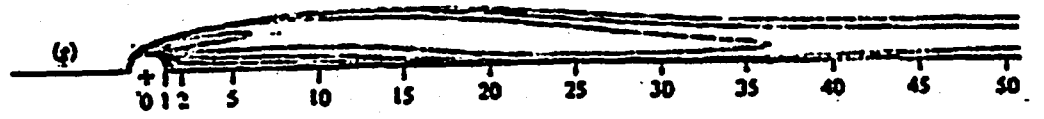


FIGURE 5.15. Vorticity contours at Re=200 [33].

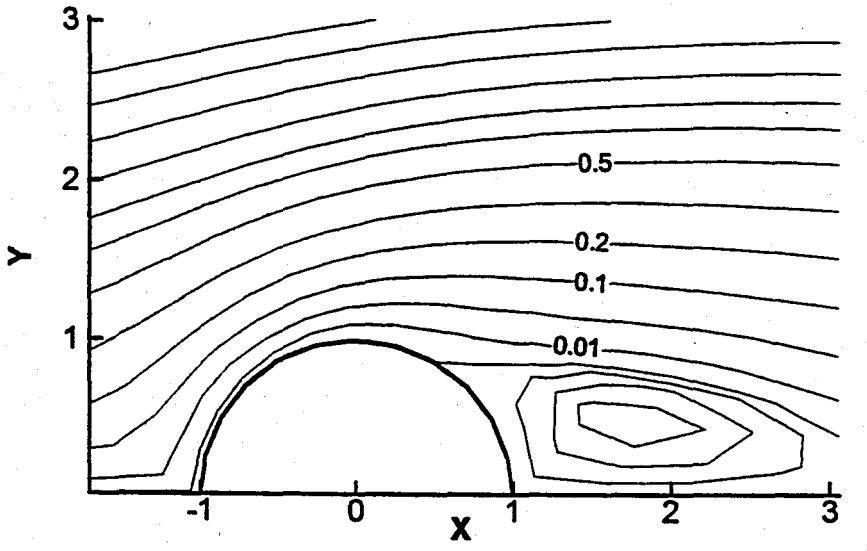


FIGURE 5.16. Details of Figure 5.8.

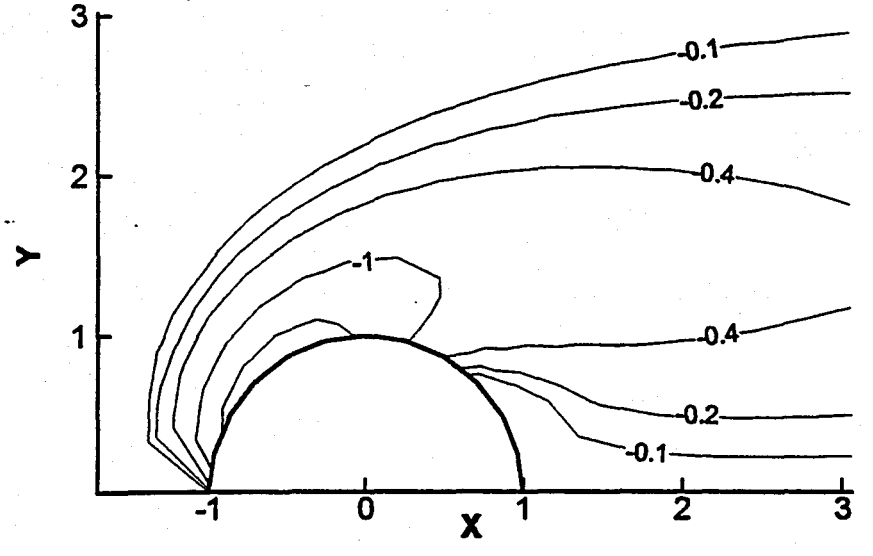


FIGURE 5.17. Details of Figure 5.10.

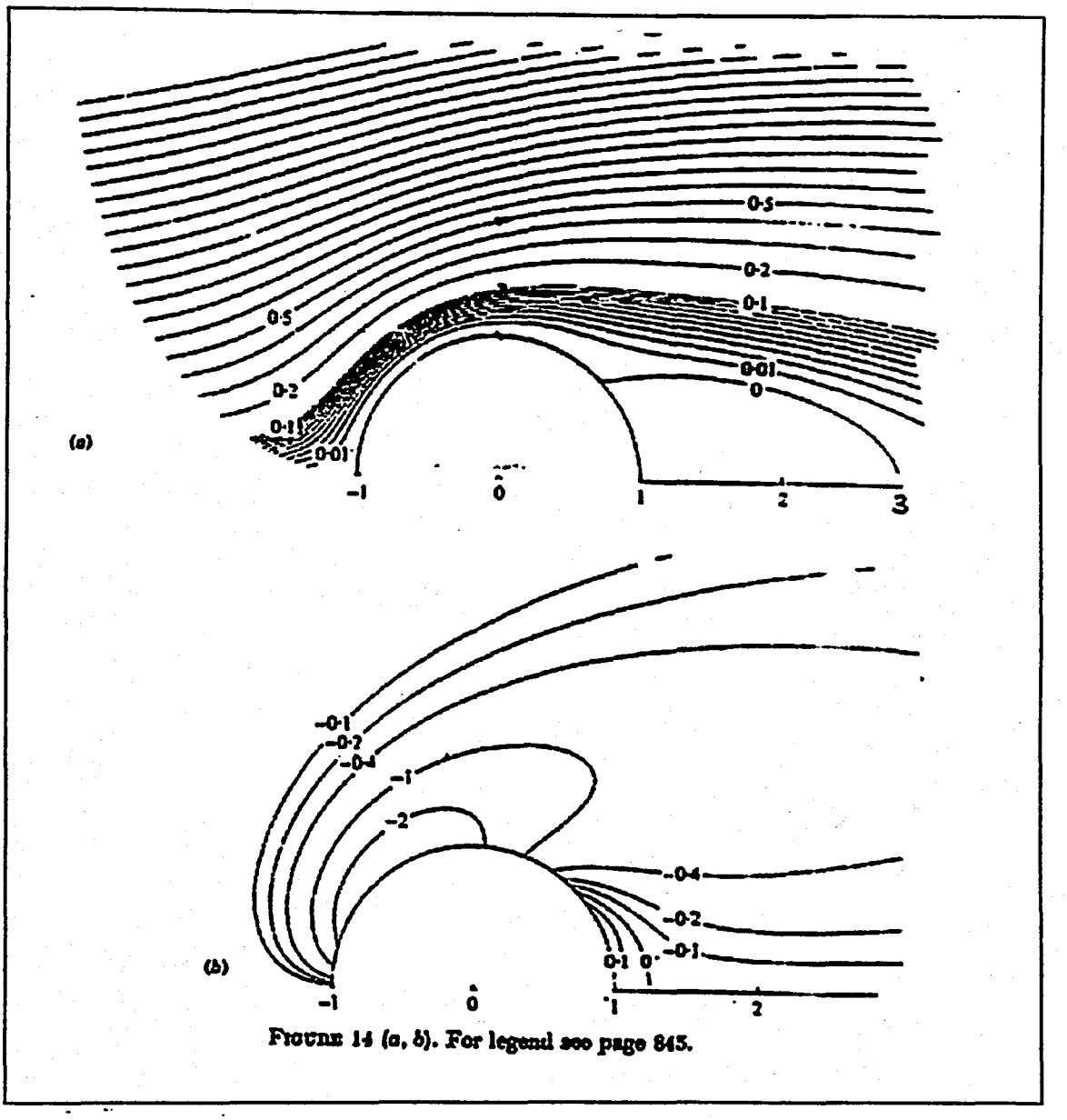


FIGURE 14 (a, b). For legend see page 845.

FIGURE 5.18. Details of Figures 5.9 and 5.11.

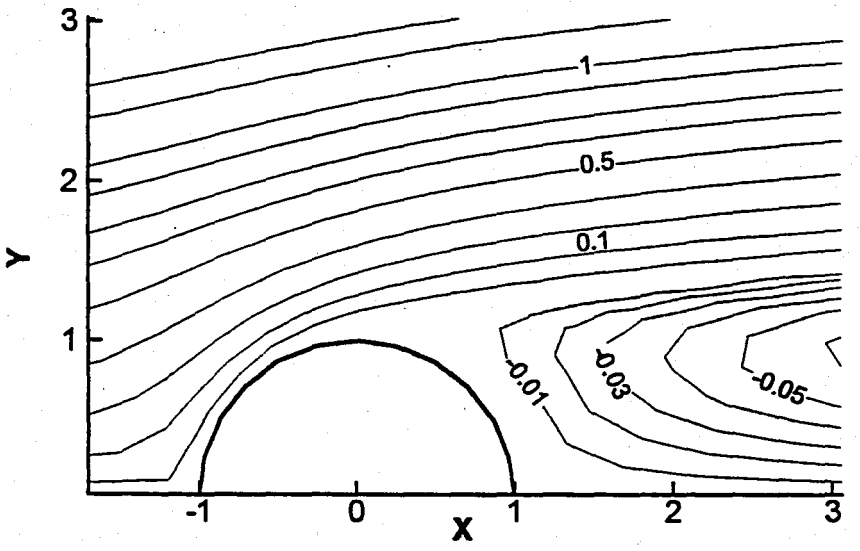


FIGURE 5.19. Details of Figure 5.12.

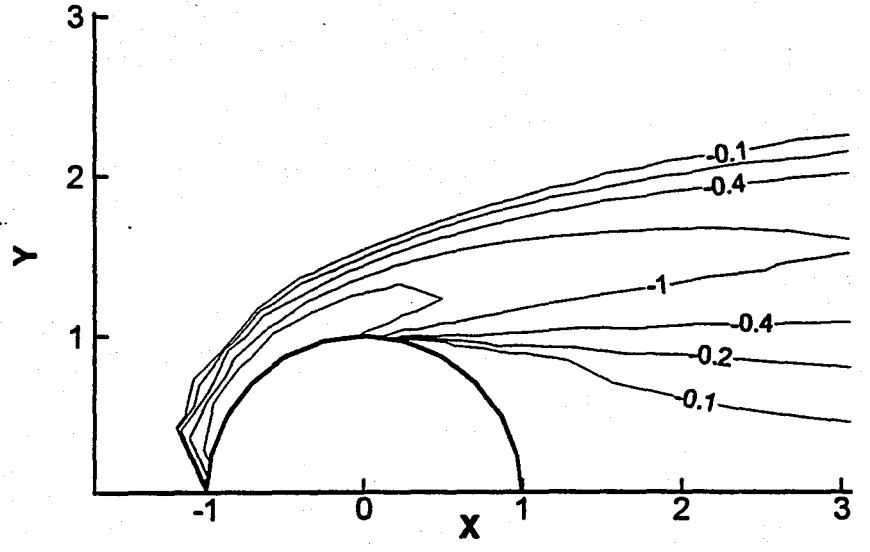


FIGURE 5.20. Details of Figure 5.14.

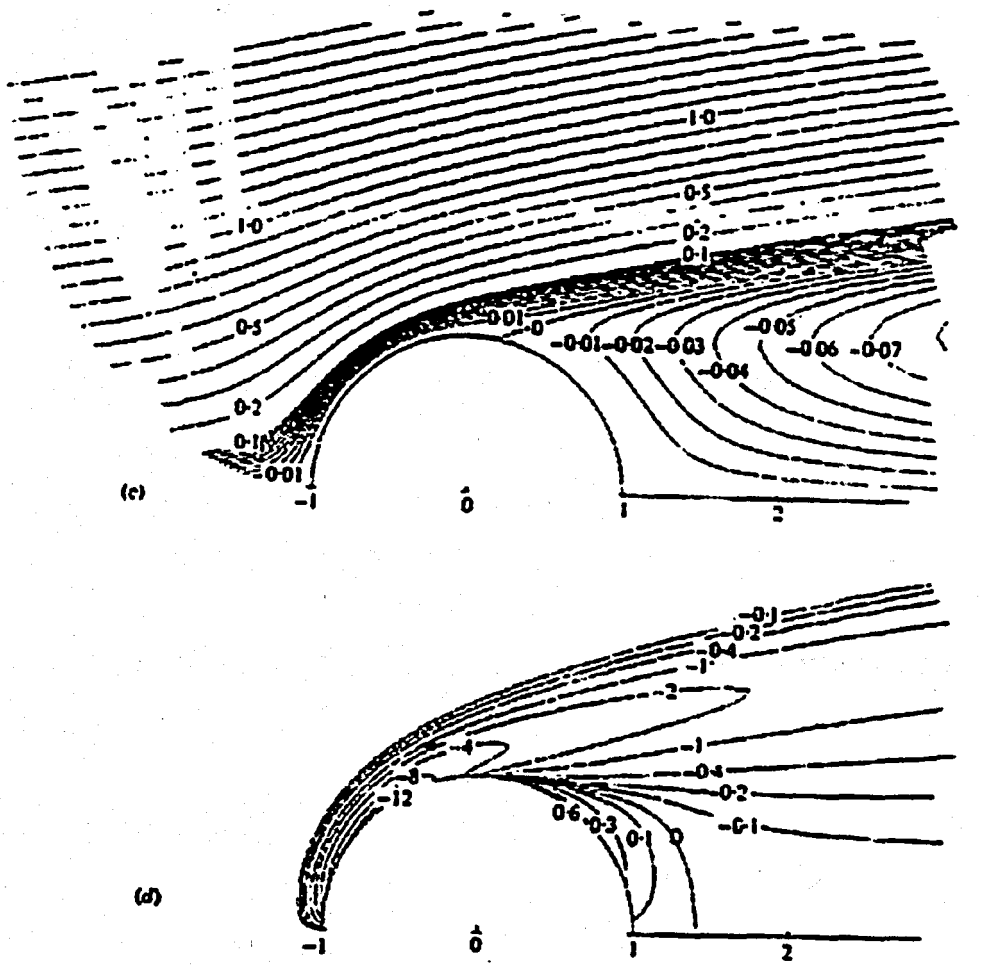


FIGURE 14. Details of the flow pictures for $Re = 20$ and $Re = 200$. $Re = 20$, (a) stream function, (b) vorticity. $Re = 200$, (c) stream function, (d) vorticity.

FIGURE 5.21. Details of Figures 5.13. and 5.15.

From these Figures it can be observed that the results of the present study are consistent with the study of Fornberg Bengt [33].

5.3. Steady Flow Field around NACA 0020 Airfoil

The Navier-Stokes equations are written in the vorticity-stream function formulation with the no slip boundary condition on the body is satisfied. The solution is designed to provide the velocity field distribution. Results are shown for flow over NACA 0020 airfoil and compared to NACA 0012 airfoil.

For the simulation of the steady flow field, a mesh of 57x57 grid points has been constructed with GRIDGEN numerical code. The airfoil surface is chosen as the first row of the computational domain. Therefore, there are 57 points on the airfoil surface. The inflow and far field boundary conditions have been selected three chord lengths away from the midpoint of the airfoil chord. The outflow boundary condition is six chord lengths away from the midpoint of the airfoil. The generated mesh is shown in Figure 5.22.

The steady flow field around NACA 0020 airfoil is simulated using four discrete angles of attack at different Reynolds numbers. Figures 5.23 and 5.24 show the streamline patterns for θ=5° at Reynolds numbers 100, 200, 300, 400, 500 and 600. A small flow separation near the trailing edge can be seen clearly at Reynolds number 300. As expected, as the Reynolds number increases, the flow separation occurs closer to the leading edge.

At an angle of attack θ=10° flow separation occurs somewhere between Reynolds number 100 and 200. At an angle of attack θ=15° and 20° flow separation occurs before reaching the Reynold number 100. Massive flow separations over the upper surface of the airfoil can be observed at the angle of attacks θ=15° and θ=20° with Reynolds numbers 500 and 300 respectively.

In Figure 5.32 the streamline patterns around NACA 0012 airfoil for θ=20° at Reynolds numbers 100, 200 and 300 are shown. It is obvious that, smaller flow separations occur at the upper surface of the airfoil in comparison to the case of the NACA 0020 airfoil with the same angle of attack and Reynolds numbers. This gives an idea about the thickness effect of the symmetrical airfoils on the flow field.

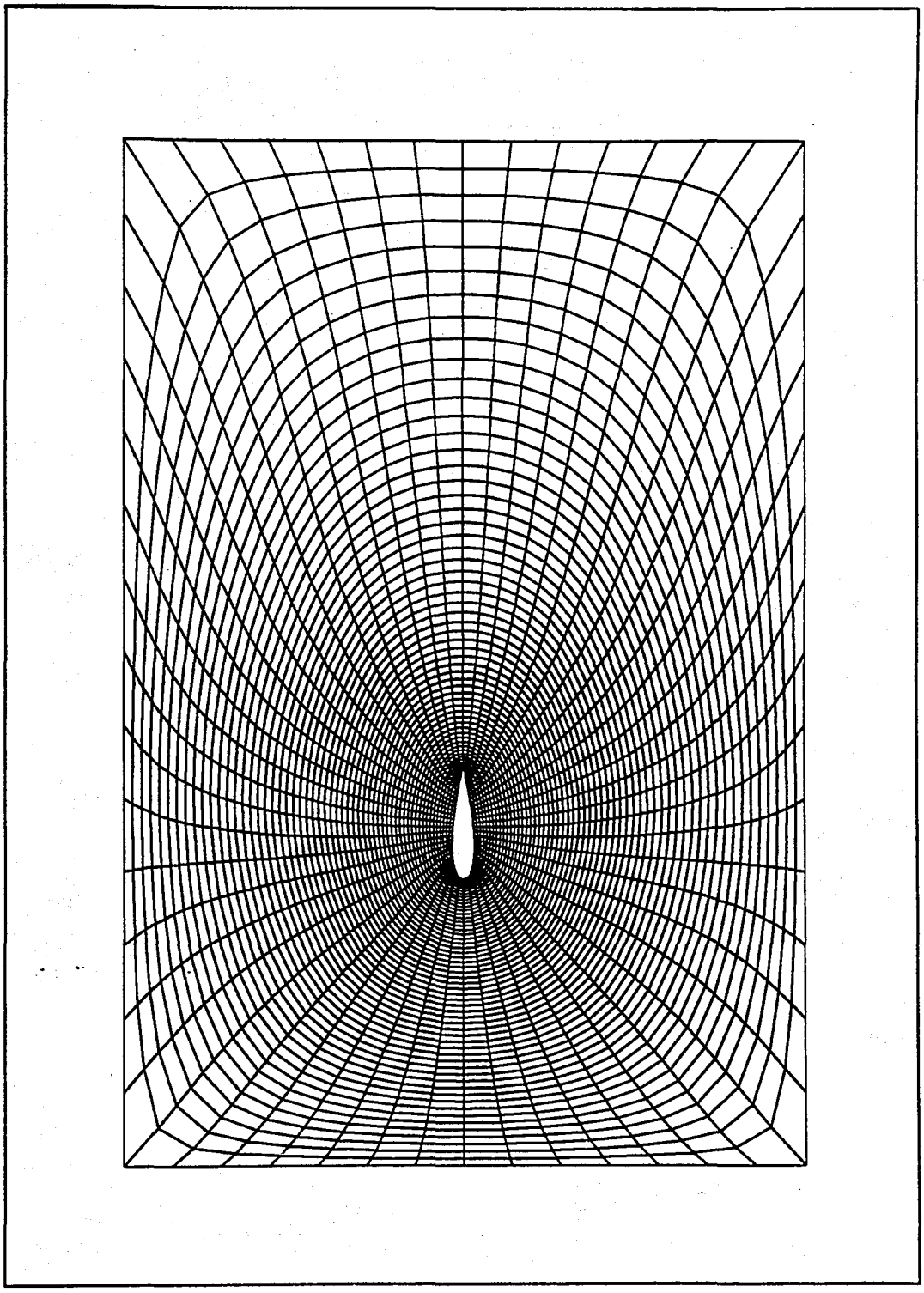
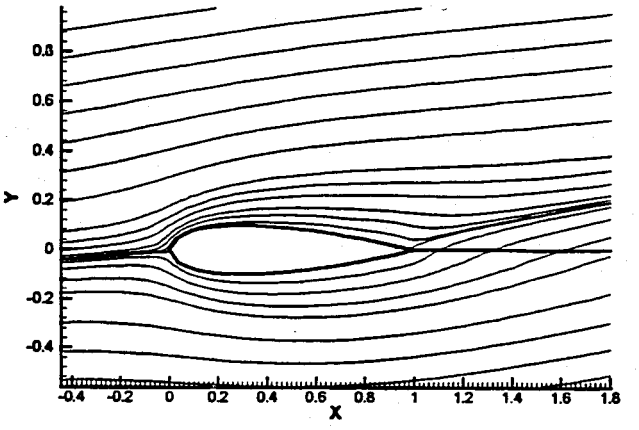
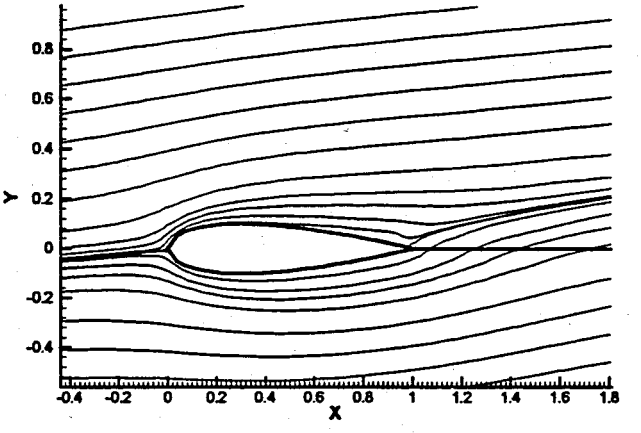


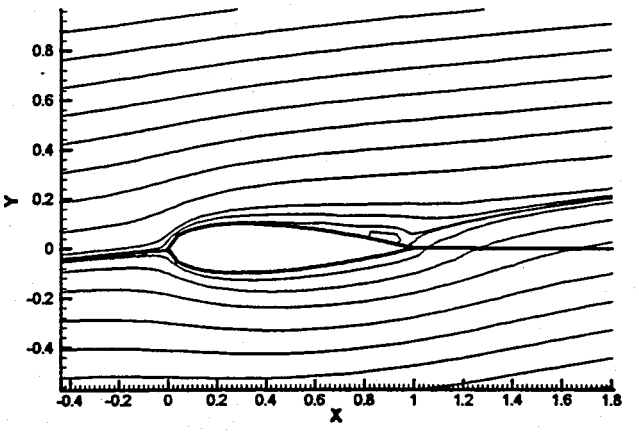
FIGURE 5.22. Mesh generated by GRIDGEN with 57x57 grid points for the steady flow field calculations of NACA 0020 airfoil.



(a) Re=100

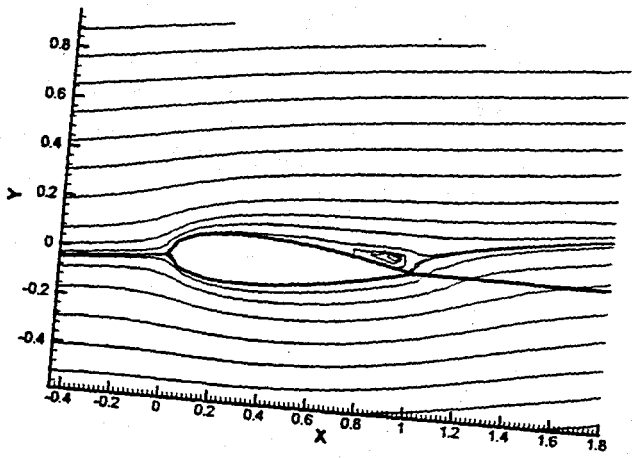


(b) Re=200

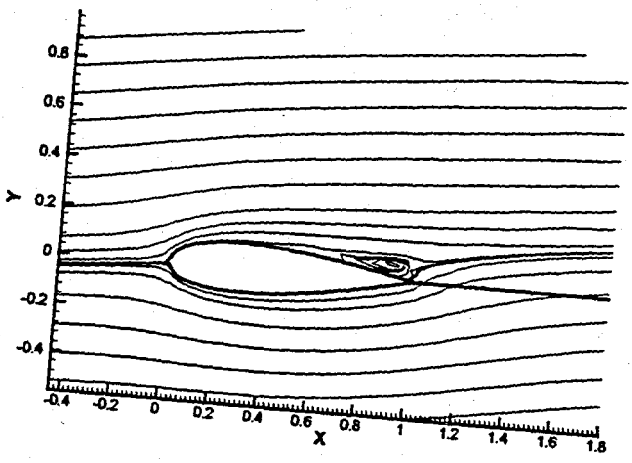


(c) Re=300

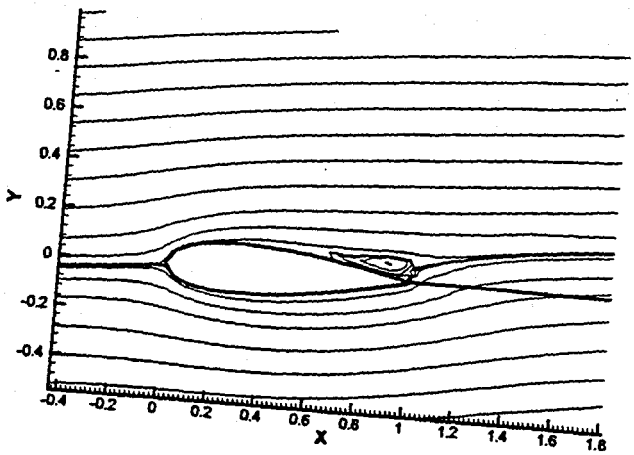
FIGURE 5.23. Steady state streamlines around NACA 0020 airfoil at angle of attack 5°, Re=100, 200 and 300.



(d) Re=400

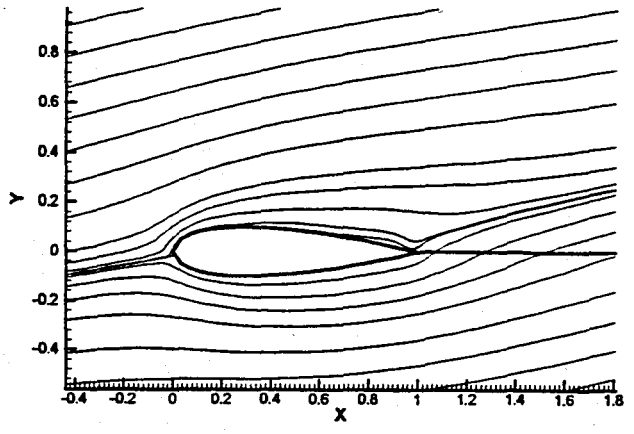


(e) Re=500

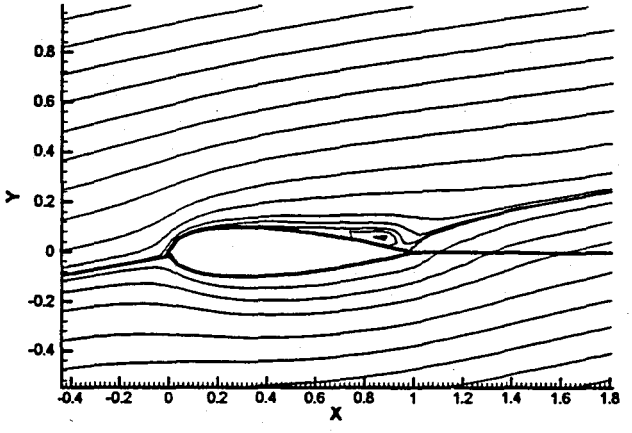


(f) Re=600

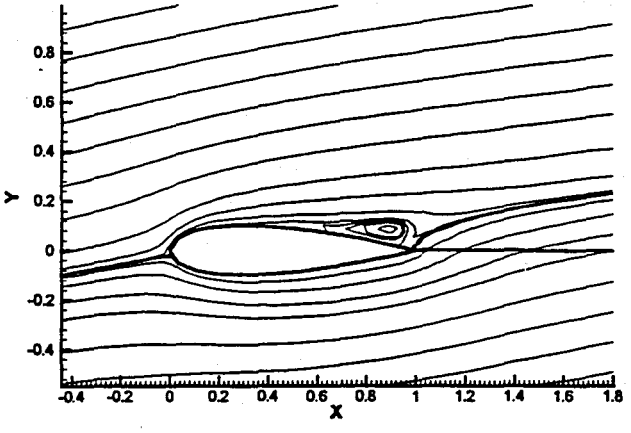
FIGURE 5.24. Steady state streamlines around NACA 0020 airfoil at angle of attack 5°, Re=400, 500 and 600.



(a) Re=100

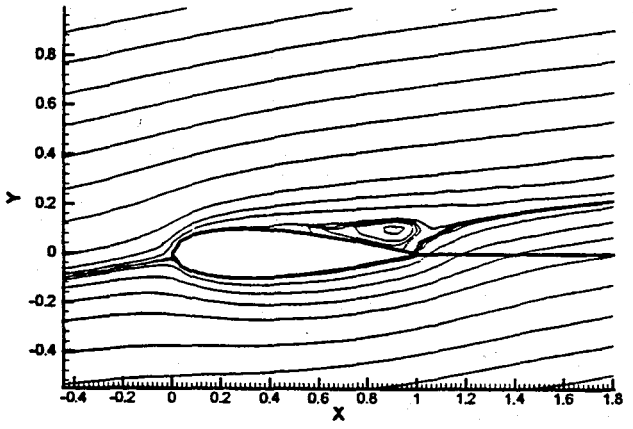


(b) Re=200

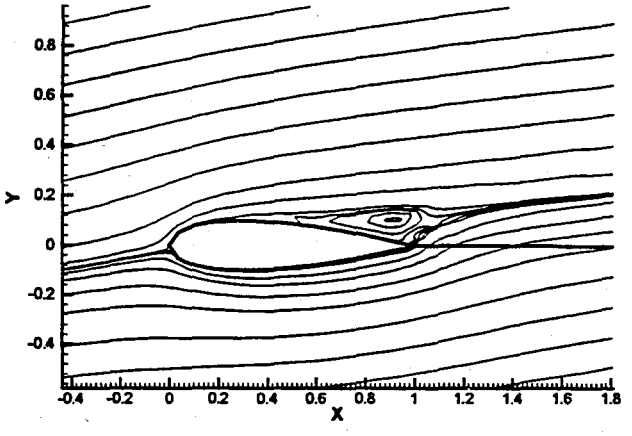


(c) Re=300

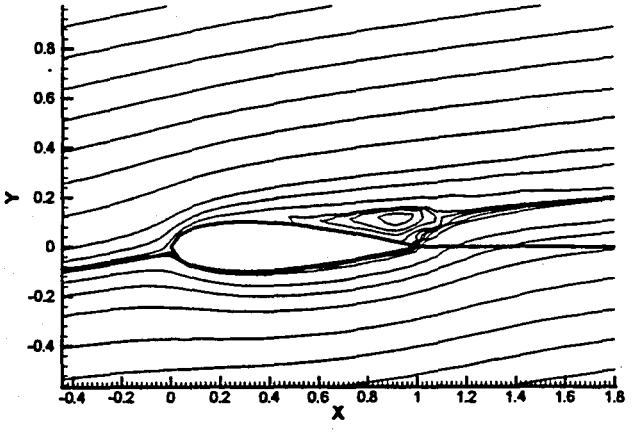
FIGURE 5.25. Steady state streamlines around NACA 0020 airfoil at angle of attack 10°, Re=100, 200 and 300.



(d) Re=400

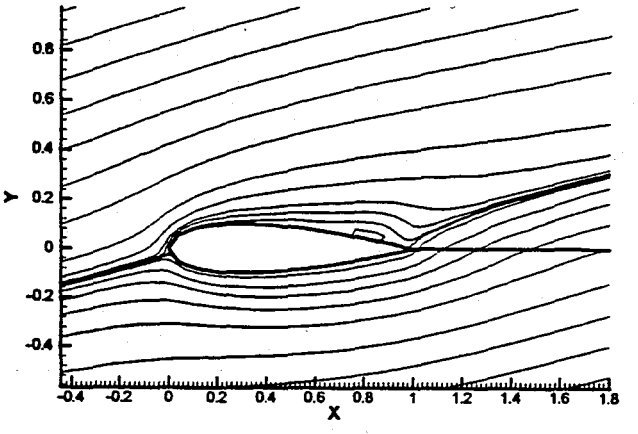


(e) Re=500

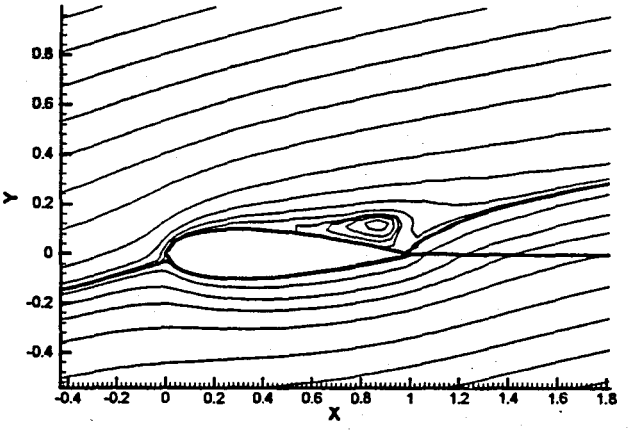


(f) Re=600

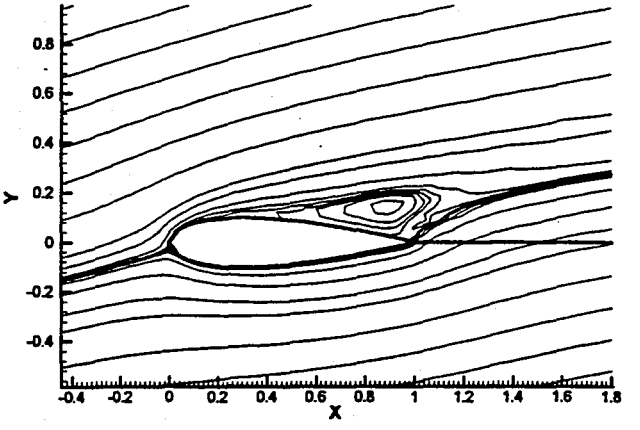
FIGURE 5.26. Steady state streamlines around NACA 0020 airfoil at angle of attack 10°, Re=400, 500 and 600.



(a) Re=100

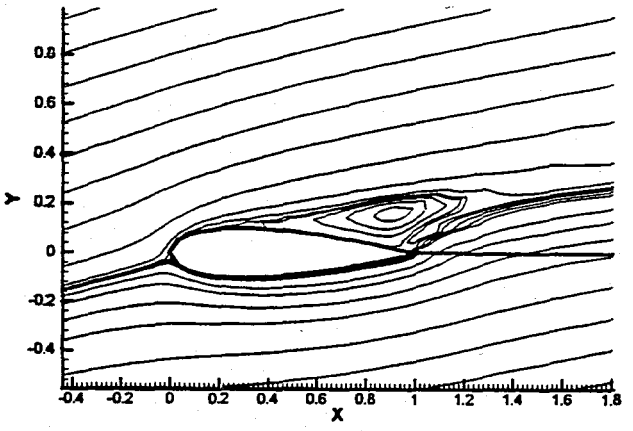


(b) Re=200

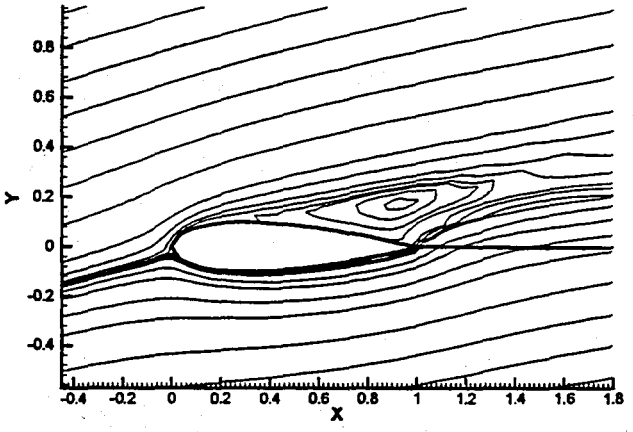


(c) Re=300

FIGURE 5.27. Steady state streamlines around NACA 0020 airfoil at angle of attack 15°, Re=100, 200 and 300.

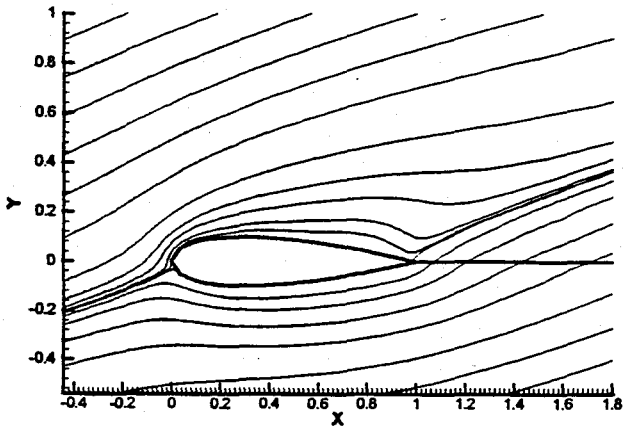


(d) Re=400

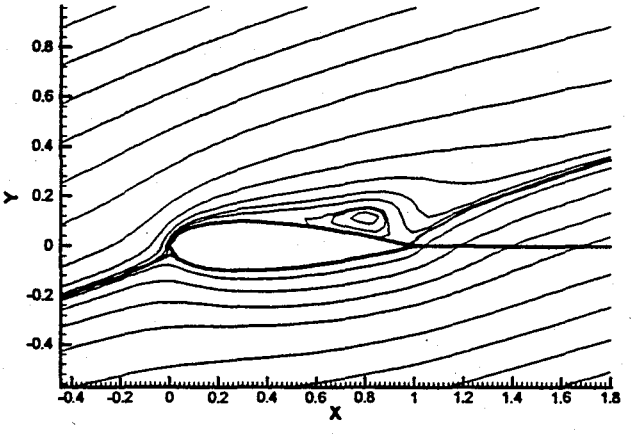


(e) Re=500

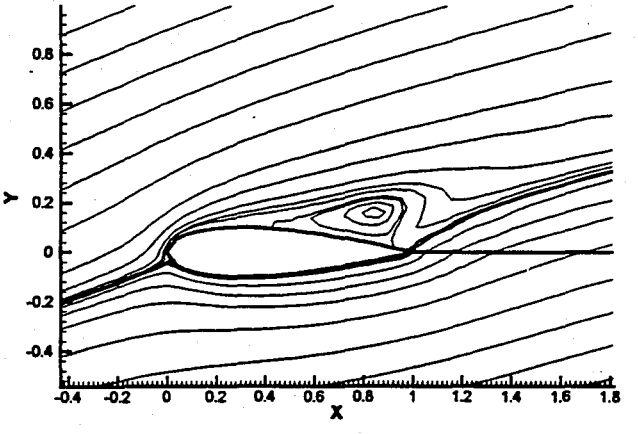
FIGURE 5.28. Steady state streamlines around NACA 0020 airfoil at angle of attack 15°, Re=400 and 500.



(a) Re=50

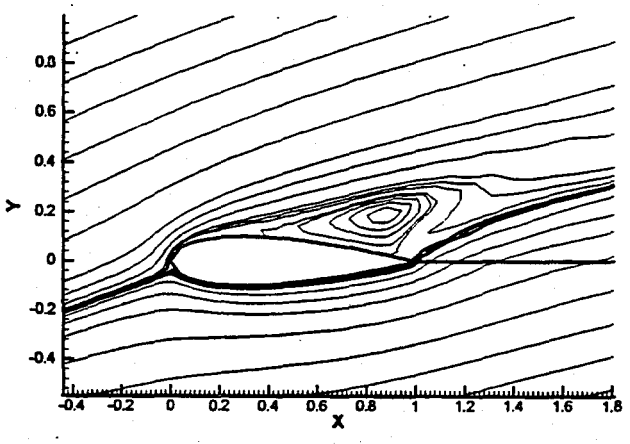


(b) Re=100

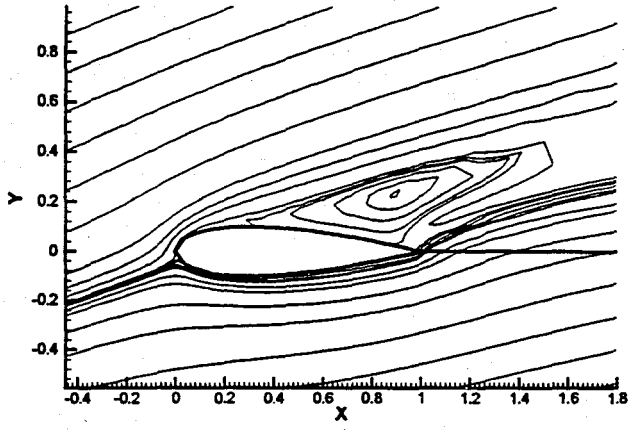


(c) Re=150

FIGURE 5.29. Steady state streamlines around NACA 0020 airfoil at angle of attack 20°, Re=50, 100 and 150.

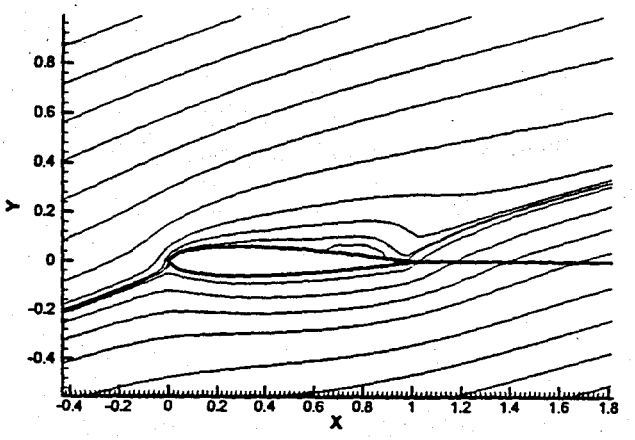


(d) Re=200

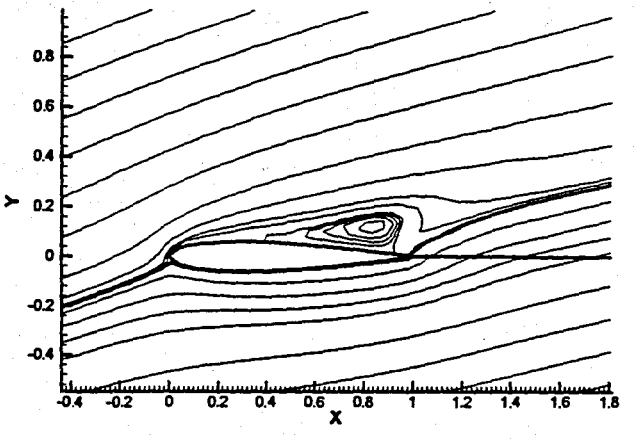


(e) Re=300

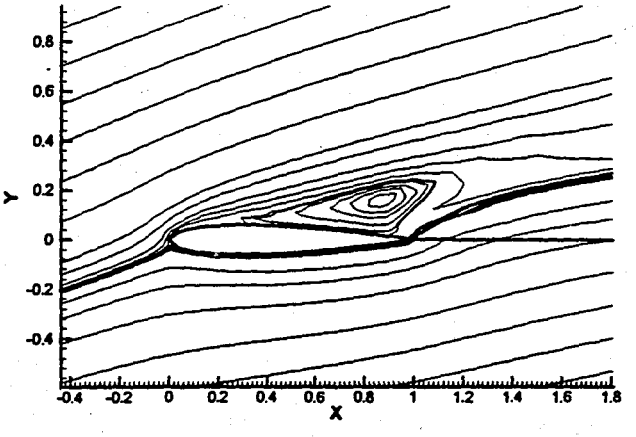
FIGURE 5.30. Steady state streamlines around NACA 0020 airfoil at angle of attack 20°, Re=200 and 300.



(a) Re=100



(a) Re=200



(a) Re=300

FIGURE 5.32. Steady state streamlines around NACA 0012 airfoil at angle of attack 20°.

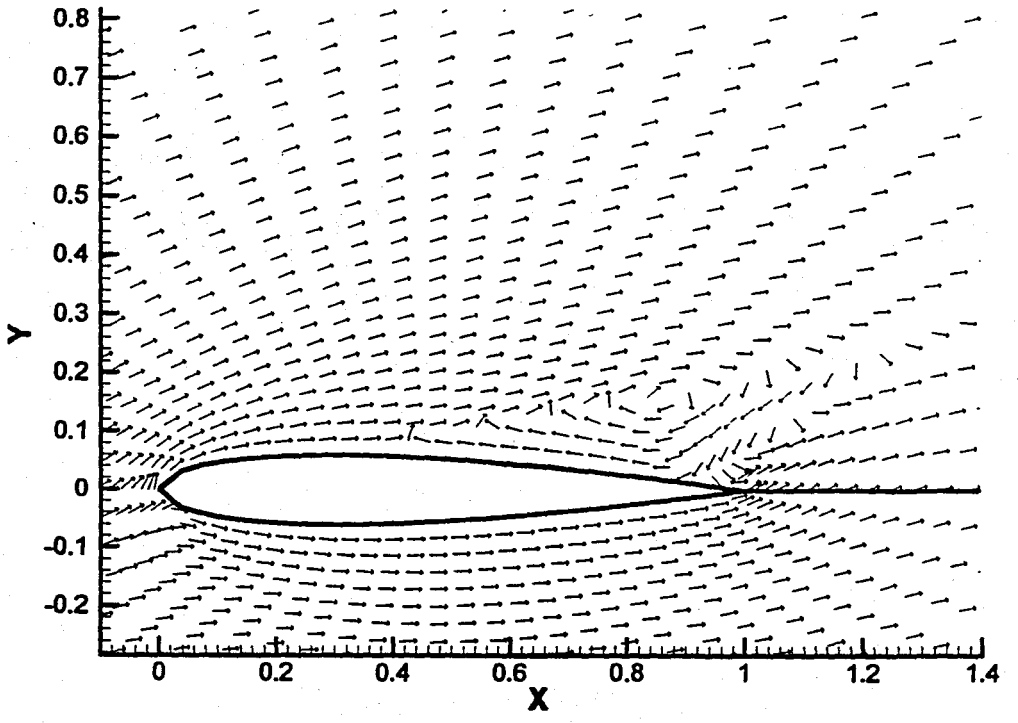


FIGURE 5.33. Vector plot around NACA 0012 airfoil at Re=300 and angle of attack 20°.

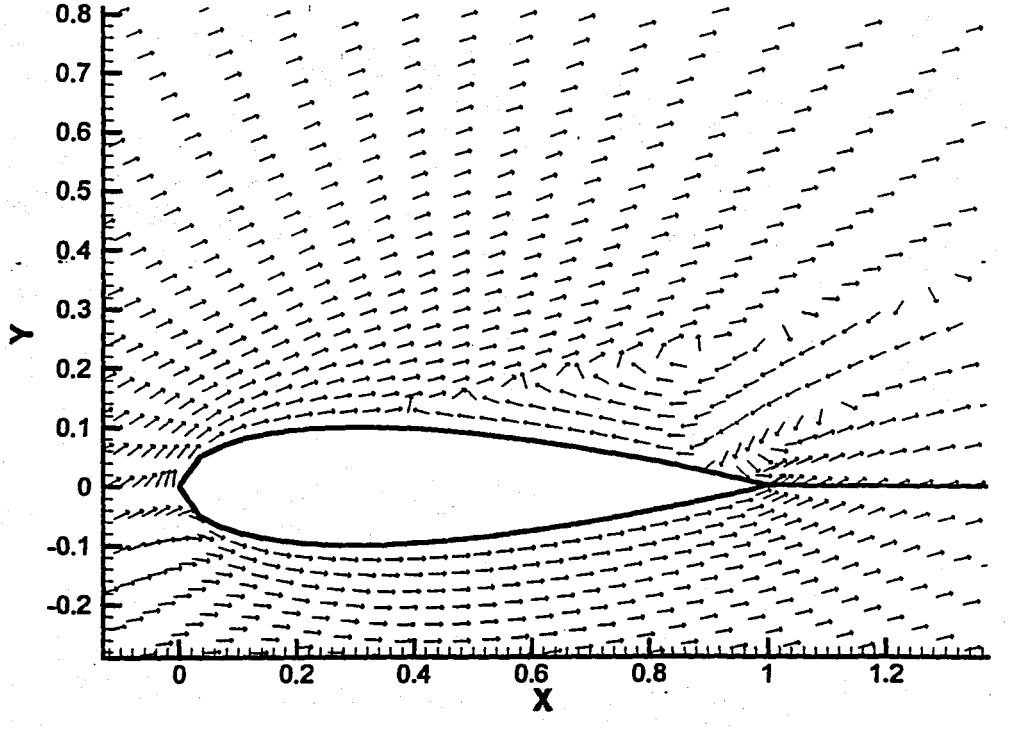


FIGURE 5.34. Vector plot around NACA 0020 airfoil at Re=300 and angle of attack 20°.

5.4. Oscillating Flow Field around NACA 0020 Airfoil

For the simulation of the unsteady flow field around NACA 0020 airfoil at Re=100, a mesh of 41x35 grid points has been generated. In the numerical formulation of the vorticity transport equation, FTCS (forward time/central space) explicit method is used. The oscillatory motion of the flow is defined by the time-dependent angle $\theta(t)$:

$$\theta(t) = \theta_0 + \theta_A \sin(\omega t) \tag{3.48}$$

where $\theta_0 = 20°$ is the mean angle, $\theta_A = 7°$ is the amplitude of the oscillation and $\omega = 2\pi/T$ is the angular velocity, where T = 7 seconds.

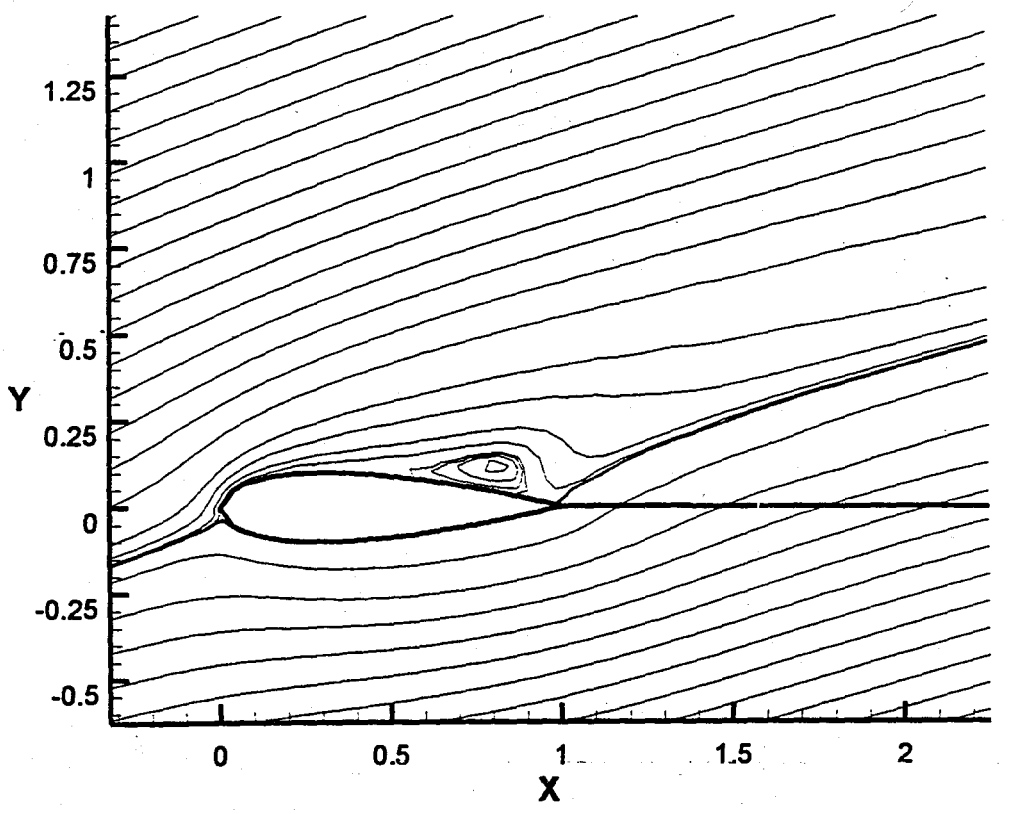


FIGURE 5.35. Steady state streamlines at Re=100 and θ_0=20°.

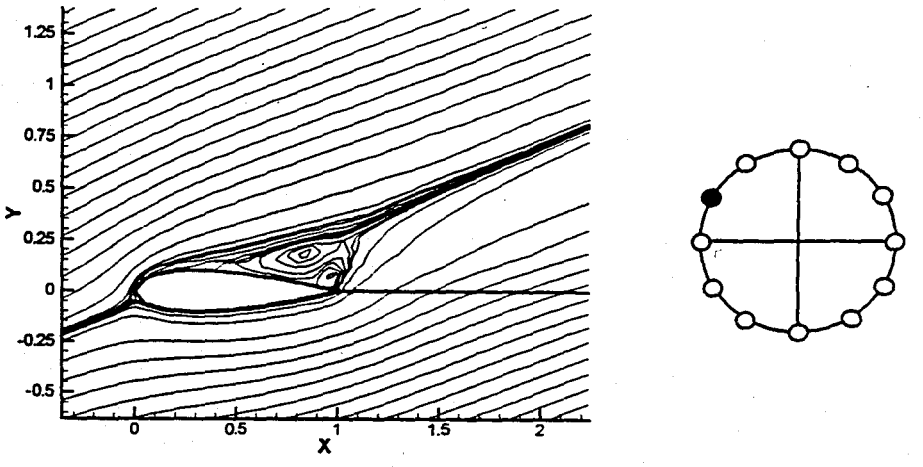


FIGURE 5.36. Streamlines at time t = 0.58 s and instant angle of attack $\theta(t) = 23.48°$.

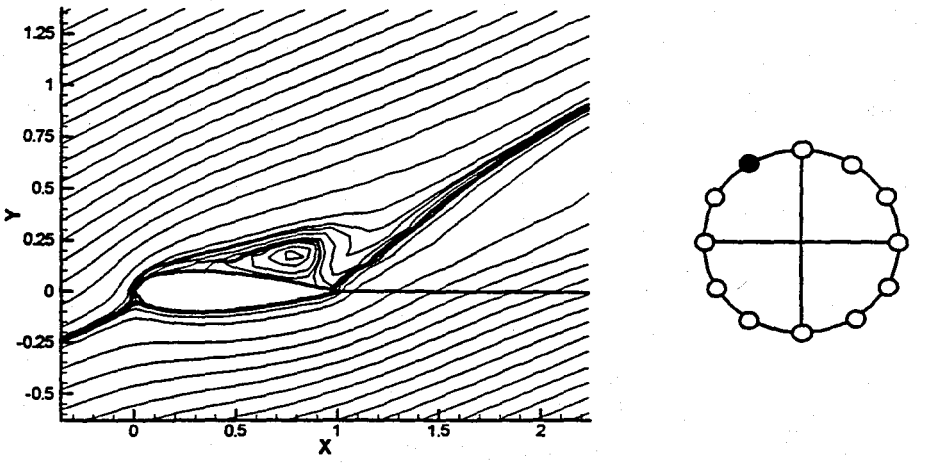


FIGURE 5.37. Streamlines at time t = 1.17 s and instant angle of attack $\theta(t) = 26.07°$.

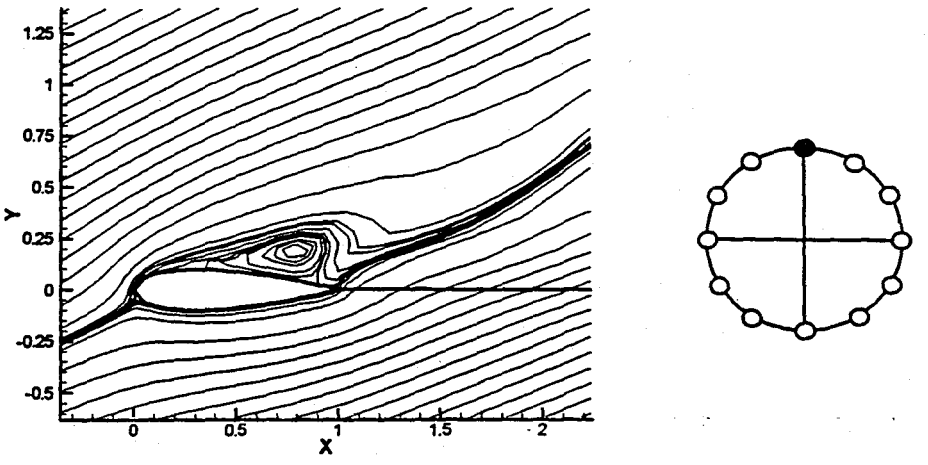


FIGURE 5.38. Streamlines at time t = 1.75 s and instant angle of attack $\theta(t) = 27°$.

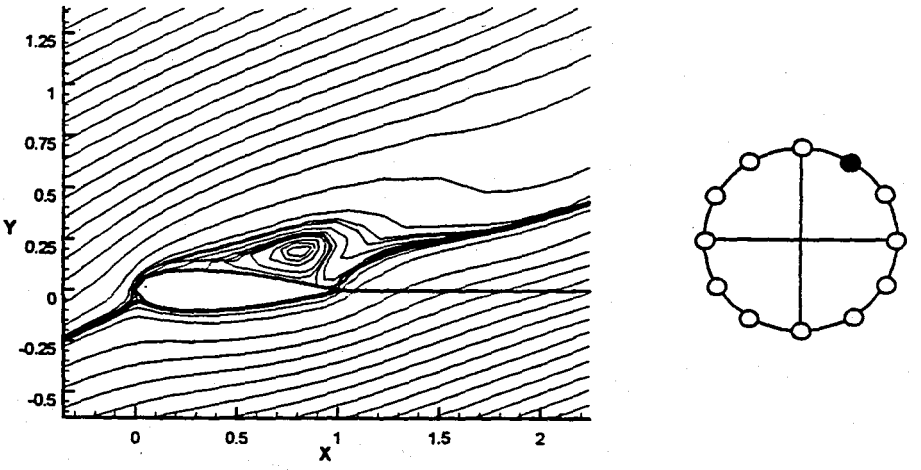


FIGURE 5.39. Streamlines at time t = 2.33 s and instant angle of attack $\theta(t) = 26.07°$.

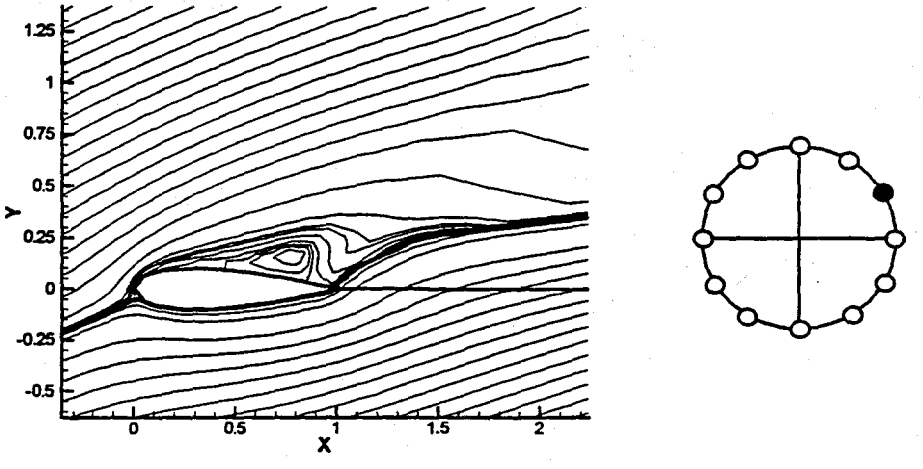


FIGURE 5.40. Streamlines at time t = 2.91 s and instant angle of attack $\theta(t) = 23.48°$.

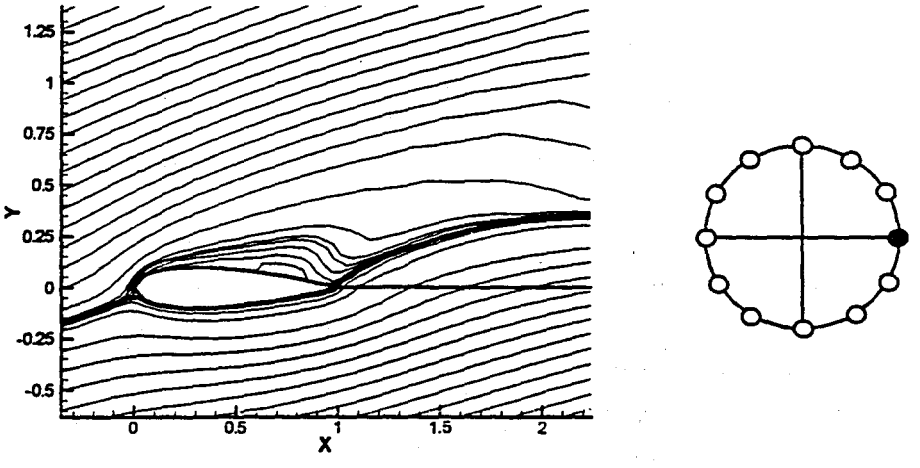


FIGURE 5.41. Streamlines at time t = 3.50 s and instant angle of attack $\theta(t) = 20°$.

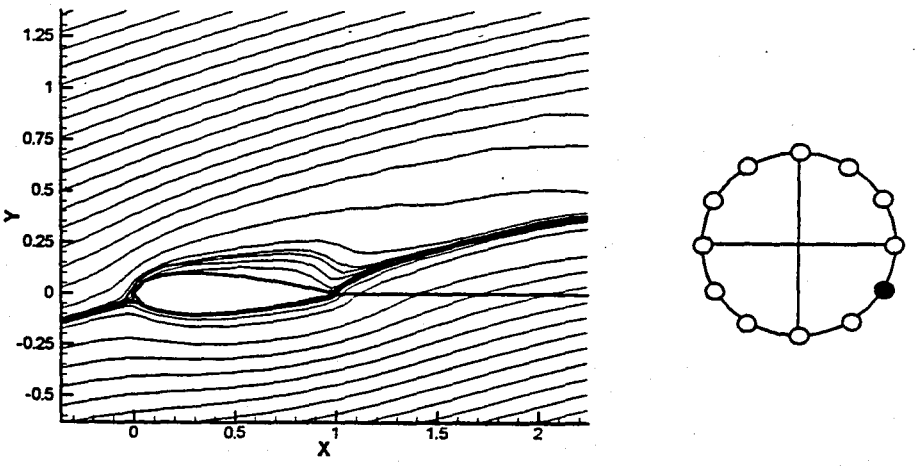


FIGURE 5.42. Streamlines at time t = 4.08 s and instant angle of attack $\theta(t) = 16.51°$.

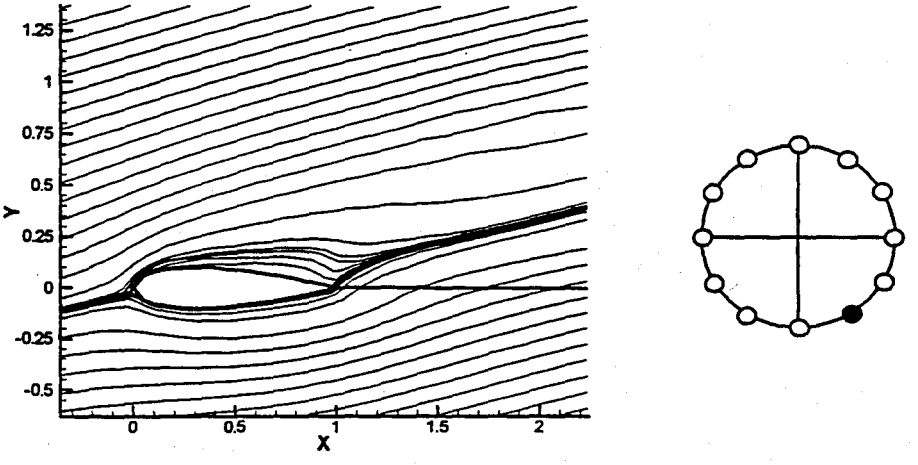


FIGURE 5.43. Streamlines at time t = 4.67 s and instant angle of attack $\theta(t) = 13.93°$.

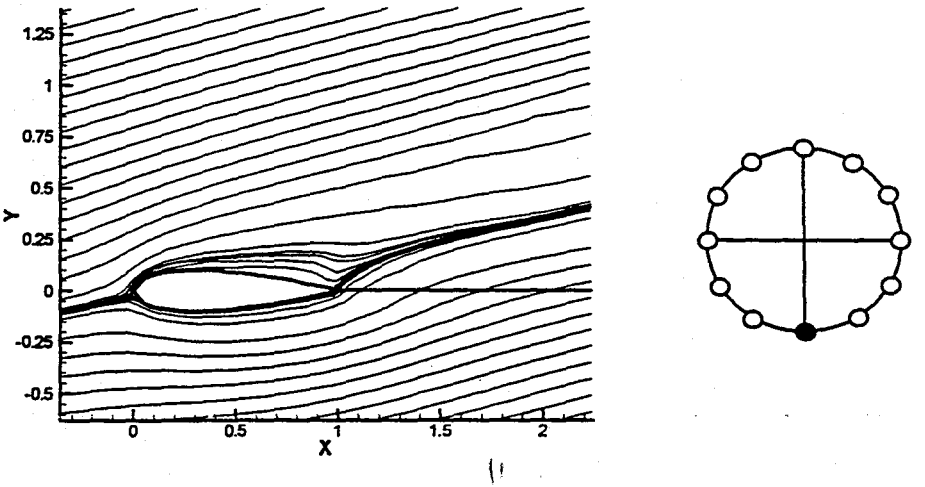


FIGURE 5.44. Streamlines at time t = 5.25 s and instant angle of attack $\theta(t) = 13°$.

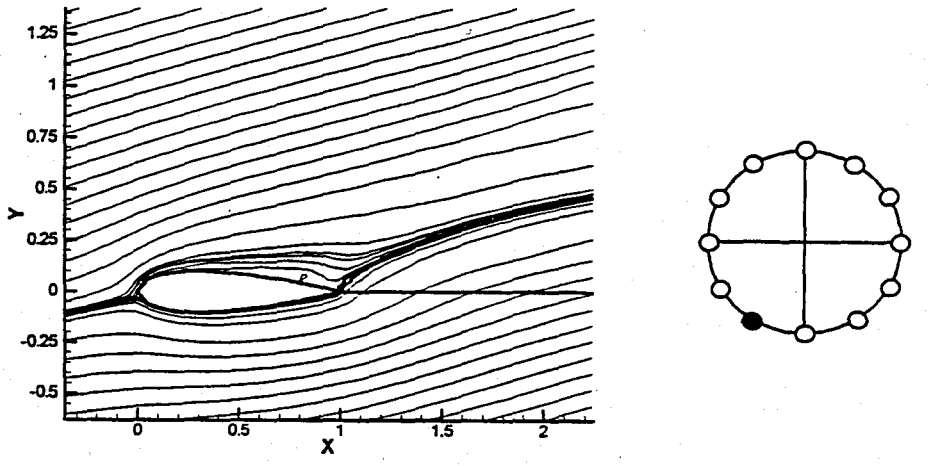


FIGURE 5.45. Streamlines at time t = 5.83 s and instant angle of attack $\theta(t) = 13.93°$.

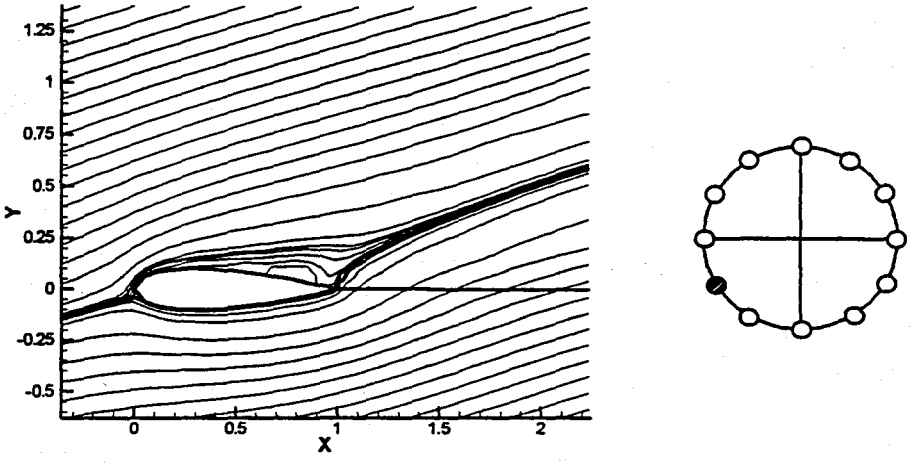


FIGURE 5.46. Streamlines at time t = 6.42 s and instant angle of attack $\theta(t) = 16.51°$.

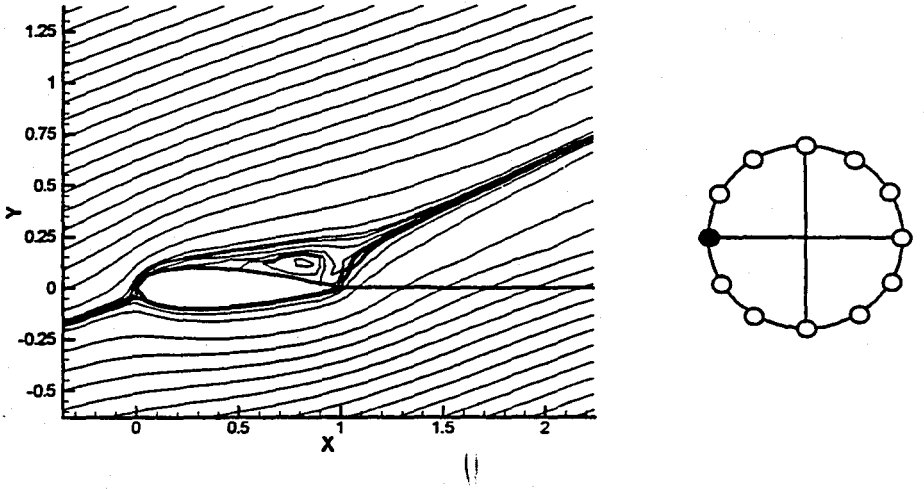


FIGURE 5.47. Streamlines at time t = 7.00 s and instant angle of attack $\theta(t) = 20°$.

6. CONCLUSIONS

Regardless of the shape and spacing of the curvilinear coordinate lines, all numerical computations, both to generate the coordinate system and subsequently to solve partial differential equations on the coordinate system, are done on a rectangular grid with a rectangular mesh, that is, in the transformed plane. The Navier-Stokes equations have been solved by using modern Krylov subspace methods, which are well suited for parallel computers. Parallelizability of the numerical code is an important feature of the present study since parallelism is an important criterion for assessing the suitability of numerical algoritms in today's rapidly advancing technology [31]. In addition to that, among the other studies given in literature survey section, the present study is the first one in application of the Krylov subspace methods for the solution of viscous flow fields around airfoils.

In order to validate the present study, steady viscous flow past a backward facing step and viscous flow around a circular cylinder have been solved. Results of the viscous flow around a circular cylinder have qualitatively been compared to study of Bengt [33] and it is observed that the results are very consistent.

Steady and unsteady, incompressible two-dimensional, viscous flow fields around NACA 0020 airfoil have been calculated and simulated by the use of the validated numerical code. Results of the steady problem showed that the flow is fully attached at low angle of attacks and Reynolds numbers. Separations have been observed at certain combination of these parameters and massive flow separations have occurred at high angle of attacks and increasing Reynolds numbers. As expected, flow separation have started first at the trailing edge and extended with the increasing values of the parameters closer to leading edge, which means the airfoil experiences gradually the loss of lift and becomes a bluff body.

In the present study, converged results up to Reynolds number of 700 could be achieved. The following explanation can be probably the reason for the Reynolds number

limitation. In order to treat high Reynolds number flow the coordinate lines must concentrate near the surface to a greater degree as the Reynolds number increases. If the number of the curvilinear lines encircling the body are increased and the method of clustering is applied, the distances between the grid points close to the airfoil, where large gradients exist, decreases. Because of the hardware limitations, the grid size has been chosen small as compared to other studies in literature and therefore the requirement for large Reynolds number treatment has been not achieved. Furthermore, in other studies calculations were performed in supercomputers, whereas in the present study a Pentium 100 MHz CPU with 16 MB. RAM has been used.

The numerical code utilizes Newton's method for the linearization of non-linear systems and preconditioned Gmres algorithm for solving the linear system arising at each Newton's iteration. GMRES only requires matrix-vector products with the coefficient matrix, therefore, the code does not need Jacobi matrix explicitly and no storage requirements for the Jacobi (coefficient matrix) are needed. With the combination of the matrix free method, GMRES needs less storage capacity and fewer operations per step than some similar algorithms.

It is found that, without a powerful preconditioner, it is very troublesome to obtain a converged solution in some ill-conditioned systems by using GMRES algorithm. Therefore using GMRES with a powerful preconditioner is very crucial. The code of the present study uses left ILU as the preconditioner. With the use of the ILU preconditioner, the convergence property of the numerical algorithm is improved remarkably.

APPENDIX A

a) TABLE A.1. List of some currently in service airplanes and their airfoils.

Airplane	Airfoil
Beechcraft Sundowner	NACA 63415
Cessna 150	NACA 2412
Fairchild A-10	NACA 6716 (at root)
	NACA 6713 (at rip)
General Dynamics F-16	NACA 64A204
Lockheed C-5 Galaxy	NACA 0012 (modified)

b) The profile of the NACA-00't' family is given by,

$$y = t\left(a_1 x^{1/2} + a_2 x + a_3 x^2 + a_4 x^3 + a_5 x^4\right)$$

where,

a_1=1.4779155,

a_2=-0.624424,

a_3=-1.727016,

a_4=1.384087,

a_5=-0.510563

and t is the airfoil thickness and indicates the percentage thickness as a two digit number. NACA-0020 airfoil implies a symmetric airfoil with a 20 per cent thickness based on a unit chord.

APPENDIX B

TABLE B.1. General transformation of the equations [34].

Physical space (x, y)	Computational space (ξ, η)
$\frac{\partial}{\partial x}$	$\left(\frac{\partial}{\partial \xi}\right)\left(\frac{\partial \xi}{\partial x}\right)+\left(\frac{\partial}{\partial \eta}\right)\left(\frac{\partial \eta}{\partial x}\right)$
$\frac{\partial}{\partial y}$	$\left(\frac{\partial}{\partial \xi}\right)\left(\frac{\partial \xi}{\partial y}\right)+\left(\frac{\partial}{\partial \eta}\right)\left(\frac{\partial \eta}{\partial y}\right)$
$\frac{\partial^2}{\partial x^2}$	$\left(\frac{\partial}{\partial \xi}\right)\left(\frac{\partial^2 \xi}{\partial x^2}\right)+\left(\frac{\partial}{\partial \eta}\right)\left(\frac{\partial^2 \eta}{\partial x^2}\right)+\left(\frac{\partial^2}{\partial \xi^2}\right)\left(\frac{\partial \xi}{\partial x}\right)^2$ $+\left(\frac{\partial^2}{\partial \eta^2}\right)\left(\frac{\partial \eta}{\partial x}\right)^2+2\left(\frac{\partial^2}{\partial \eta \partial \xi}\right)\left(\frac{\partial \eta}{\partial x}\right)\left(\frac{\partial \xi}{\partial x}\right)$
$\frac{\partial^2}{\partial y^2}$	$\left(\frac{\partial}{\partial \xi}\right)\left(\frac{\partial^2 \xi}{\partial y^2}\right)+\left(\frac{\partial}{\partial \eta}\right)\left(\frac{\partial^2 \eta}{\partial y^2}\right)+\left(\frac{\partial^2}{\partial \xi^2}\right)\left(\frac{\partial \xi}{\partial y}\right)^2$ $+\left(\frac{\partial^2}{\partial \eta^2}\right)\left(\frac{\partial \eta}{\partial y}\right)^2+2\left(\frac{\partial^2}{\partial \eta \partial \xi}\right)\left(\frac{\partial \eta}{\partial y}\right)\left(\frac{\partial \xi}{\partial y}\right)$
$\frac{\partial^2}{\partial x \partial y}$	$\left(\frac{\partial}{\partial \xi}\right)\left(\frac{\partial^2 \xi}{\partial x \partial y}\right)+\left(\frac{\partial}{\partial \eta}\right)\left(\frac{\partial^2 \eta}{\partial x \partial y}\right)$ $+\left(\frac{\partial^2}{\partial \xi^2}\right)\left(\frac{\partial \xi}{\partial x}\right)\left(\frac{\partial \xi}{\partial y}\right)+\left(\frac{\partial^2}{\partial \eta^2}\right)\left(\frac{\partial \eta}{\partial x}\right)\left(\frac{\partial \eta}{\partial y}\right)$ $+\left(\frac{\partial^2}{\partial \eta \partial \xi}\right)\left[\left(\frac{\partial \eta}{\partial x}\right)\left(\frac{\partial \xi}{\partial y}\right)+\left(\frac{\partial \xi}{\partial x}\right)\left(\frac{\partial \eta}{\partial y}\right)\right]$

APPENDIX C

TABLE C.1. Central difference formulations.

$\frac{\partial f}{\partial \xi}$	$\frac{f_{i+1,j} - f_{i-1,j}}{2\Delta\xi}$
$\frac{\partial f}{\partial \eta}$	$\frac{f_{i,j+1} - f_{i,j-1}}{2\Delta\eta}$
$\frac{\partial^2 f}{\partial \xi^2}$	$\frac{f_{i+1,j} - 2f_{i,j} + f_{i-1,j}}{\Delta\xi^2}$
$\frac{\partial^2 f}{\partial \eta^2}$	$\frac{f_{i,j+1} - 2f_{i,j} + f_{i,j-1}}{\Delta\eta^2}$
$\frac{\partial^2 f}{\partial \xi \partial \eta}$	$\frac{f_{i+1,j+1} - f_{i+1,j-1} - f_{i-1,j+1} + f_{i-1,j-1}}{4\Delta\xi\Delta\eta}$

TABLE C.2. Forward difference formulations.

$\frac{\partial f}{\partial \xi}$	$\frac{f_{i+1,j} - f_{i,j}}{\Delta\xi}$
$\frac{\partial f}{\partial \eta}$	$\frac{f_{i,j+1} - f_{i,j}}{\Delta\eta}$
$\frac{\partial^2 f}{\partial \xi^2}$	$\frac{f_{i,j} - 2f_{i+1,j} + f_{i+2,j}}{\Delta\xi^2}$
$\frac{\partial^2 f}{\partial \eta^2}$	$\frac{f_{i,j} - 2f_{i,j+1} + f_{i,j+2}}{\Delta\eta^2}$
$\frac{\partial^2 f}{\partial \xi \partial \eta}$	$\frac{f_{i+1,j+1} - f_{i,j+1} - f_{i+1,j} + f_{i,j}}{\Delta\xi\Delta\eta}$

TABLE C.3. Backward difference formulations.

$\frac{\partial f}{\partial \xi}$	$\frac{f_{i,j} - f_{i-1,j}}{\Delta\xi}$
$\frac{\partial f}{\partial \eta}$	$\frac{f_{i,j} - f_{i,j-1}}{\Delta\eta}$
$\frac{\partial^2 f}{\partial \xi^2}$	$\frac{f_{i,j} - 2f_{i-1,j} + f_{i-2,j}}{\Delta\xi^2}$
$\frac{\partial^2 f}{\partial \eta^2}$	$\frac{f_{i,j} - 2f_{i,j-1} + f_{i,j-2}}{\Delta\eta^2}$
$\frac{\partial^2 f}{\partial \xi \partial \eta}$	$\frac{f_{i-1,j-1} - f_{i,j-1} - f_{i-1,j} + f_{i,j}}{\Delta\xi\Delta\eta}$

REFERENCES

1. Greenberg, J. Mayo, "Some Considerations on an Airfoil in an Oscillating Stream", 1947, http://naca.larc.nasa.gov/reports/1947/naca-tn-1372/index.cgi?page0002.gif

2. Hodge, James K., Alan L. Stone, and Thomas E. Miller, "Numerical Solution for Airfoils Near Stall in Optimised Boundary Fitted Curvilinear Coordinates", *AIAA Journal*, Vol. 17, No. 5, pp. 458-464, October 1978.

3. Barton, T. John, and Thomas H. Pulliam, "Airfoil Computation at High Angels of Attack, Inviscid and Viscous Phenomena", *AIAA Journal*, Vol. 24, No.5, pp. 705-712, May 1986.

4. Halim, Ahmad A. M., "Global Marching Technique for Predicting Separated Flows over Arbitrary Airfoils", *AIAA Journal*, Vol. 25, No. 9, pp. 1263-1266, September 1987.

5. Kandil, Osama A., and Chuang H. Andrew, "Unsteady Transonic Airfoil Computation Using Implicit Euler Scheme on Body-Fixed Grid", *AIAA Journal*, Vol. 27, No. 8, pp. 1031-1037, August 1989.

6. Schroeder, Linda M., and Fleeter Sanford, "Viscous Aerodynamic Analysis of an Oscillating Flat-Plate Airfoil", *AIAA Journal*, Vol. 27, No. 8, pp. 1021-1022, August 1989.

7. Koochesfahani, Manoochehr M., "Vortical Patterns in the Wake of an Oscillating Airfoil", *AIAA Journal*, Vol. 27, No. 9, pp. 1200-1205, September 1989.

8. Tuncer, Ismail H., James C. Wu, and C. M. Wang, "Theoretical and Numerical Studies of Oscillating Airfoils", *AIAA Journal*, Vol. 28, No. 9, pp. 1615-1624, September 1990.

9. Tung, Chee, Kenneth W. McAlister, and Clin M. Wang, "Unsteady Aerodynamic Behaviour of an Airfoil with and without a Slat", *Computers and Fluids*, Vol. 22, No. 4/5, pp. 529-547, 1993.

10. Panda, J., and K. B. M. Q. Zaman, "Experimental Investigation of the Flow Field of and Oscillating Airfoil and Estimation of Lift from Wake Surveys", *Journal of Fluid Mechanics*, Vol. 265, pp. 65-95, 1994.

11. Ahmed, S., and M. S. Chandrasekhara, "Reattachment Studies of an Oscillating Airfoil Dynamic Stall Flow Field", *AIAA Journal*, Vol. 32, No. 5, pp. 1006-1012, May 1994.

12. Ekaterinaris, J. A., and F. R. Menter, "Computation of Oscillating Airfoil Flows with One- and Two-Equation Turbulence Models", *AIAA Journal*, Vol. 32, No. 12, pp. 2359-2365, December 1994.

13. Srinivasan, G. R., J. A. Ekaterinaris, and W. J. McCroskey, "Evaluation of Turbulence Models for Unsteady Flows of an Oscillating Airfoil", *Computers and Fluids*, Vol. 24, No. 7, pp. 833-861, 1995.

14. Kim, S. W., K. B. M. Q. Zaman, and J. Panda, "Numerical Investigation of Unsteady Transitional Flow over Oscillating Airfoil", *Journal of Fluids Engineering*, Vol. 117, pp. 10-16, March 1995.

15. Choudhuri, P. Ghosh, and D. D. Knight, "Effects of Compressibility, Pitch Rate, and Reynolds Number on Unsteady Incipient Leading-Edge Boundary Layer Separation over a Pitching Airfoil", *Journal of Fluid Mechanics*, Vol. 308, pp. 195-217, 1996.

16. Steger, Joseph L., "Implicit Finite-Difference Simulation of Flow About Arbitrary Two-Dimensional Geometries", *AIAA Journal*, Vol. 16, No. 7, pp. 679-686, July 1978.

17. Guilmineau, E., J. Piquet, and P. Queutey, "Two-Dimensional Turbulent Viscous Flow Simulation past Airfoils at Fixed Incidence", *Computers and Fluids*, Vol. 26, No. 2, pp. 135-162, 1997.

18. Bhaskaran, Rajesh, and Alric P. Rothmayer, "Separation and Instabilities in the Viscous Flow over Airfoil Leading Edges", *Computers and Fluids*, Vol. 27, No. 8, pp. 903-921, 1998.

19. Hoffmann, Klaus A. and Steve T. Chiang, *Computational Fluid Dynamics for Engineers-Vol. I*, Engineering Education System, Wichita Kansas, Fourth Print August 1997.

20. Ecder, Ali, "Grid Generation", Lecture Notes in ME 683, Computatioanl Fluid Dynamics, Bogazici University, 1997.

21. Thompson, Joe F., Zahir U. A. Warsi, and C. Wayne Mastin, "Boundary-Fitted Coordinate Systems for Numerical Solution of Partial Differential Equations-A Review", *Journal of Computational Physics*, Vol. 47, pp. 1-108, 1982.

22. Thompson, Joe F., Frank C. Thames, and C. Wayne Mastin, "TOMCAT-A Code for Numerical Generation of Boundary Fitted Curvilinear Coordinate Systems on Fields Containing Any Number of Arbitrary Two-Dimensional Bodies", *Journal of Computational Physics*, Vol. 24, pp. 274-302, 1977.

23. Anderson, John D., *Fundamentals of Aerodynamics*, Second Edition, Mcgraw-Hill, New York, 1991.

24. McCormick, Barnes W., *Aerodynamics, Aeronautics, and Flight Mechanics*, John Wiley and Sons, New York, 1979.

25. Schlichting, Hermann, *Boundary-Layer Theory*, McGraw-Hill, Inc., Seventh Edition, New York, 1979.

26. Shevell, Richard S., *Fundamentals of Flight*, Prentice Hall, Second Edition, New Jersey, 1989.

27. Thames, Frank C., Joe F. Thompson, C. Wayne Mastin, and Ray L. Walker, "Numerical Solutions for Viscous and Potential Flow about Arbitrary Two-Dimensional Bodies Using Body-Fitted Coordinate Systems", *Journal of Computational Physics*, Vol. 24, pp. 245-273, 1977.

28. Hwang, Robert R., and S. H. Lin, "On Laminar Wakes Behind a Circular Cylinder in Stratified Fluids", *Journal of Fluids Engineering*, Vol. 114, pp. 20-28, March 1992.

29. Thompson, Joe F., Frank C. Thames, and C. Wayne Mastin, "Automatic Numerical Generation of Body-Fitted Curvilinear Coordinate System for Field Containing Any Number of Arbitrary Two-Dimensional Bodies", *Journal of Computational Physics*, Vol. 15, pp. 299-319, 1974.

30. Fletcher, C.A. J., *Computational Technics for Fluid Dynamics Vol. 1 – 2*, Springer – Verlag, Berlin, 1991.

31. Ecder, Ali, "Summer Short Course Series on CFD Techniques in Engineering", İstanbul, YTU, July 18-30, 1994.

32. Saad, Y., and M. H. Schultz, "GMRES: A General Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems", *SIAM Journal of Science and Statistical Computing*, Vol. 7, No. 3, pp. 856-869, July 1986.

33. Fornberg, Bengt, "A Numerical Study of Steady Viscous Flow past a Circular Cylinder", *Journal of Fluid Mechanics*, Vol. 98, Part 4, pp. 819-855, June 1979.

34. Wendt, John F., *Computational Fluid Dynamics*, Springer–Verlag, New York, 1992.