Hybrid 3-D Euler/Navier-Stokes Calculations of Transonic Vortex Flows over the NTF Delta Wing

by

Todd M. Becker

B.S. Aerospace Engineering
University of Missouri at Rolla (1987)

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
Master of Science
in
Aeronautics and Astronautics
at the
Massachusetts Institute of Technology

11 August 1989

Signature of Author _____
Department of Aeronautics and Astronautics
11 August 1989

Certified by _____
Professor Earll M. Murman
Thesis Supervisor, Department of Aeronautics and Astronautics

Accepted by _____
Professor Harold Y. Wachman
Chairman, Department Graduate Committee

Hybrid 3-D Euler/Navier-Stokes Calculations of Transonic Vortex Flows over the NTF Delta Wing

by

Todd M. Becker

Submitted to the Department of Aeronautics and Astronautics
on 11 August 1989
in partial fulfillment of the requirements for the degree of
Master of Science in Aeronautics and Astronautics

A hybrid Euler/Navier-Stokes solver is developed for computing the laminar, transonic vortex flow over the blunt-edged NTF delta wing. The flow domain is split into two zones; an *inner* zone in the near the wing and wake where viscous effects are significant, and an *outer* zone away from the wing and wake where the flow is inviscid. A grid is generated in each zone, with the two grids fitting together to form a composite grid. The thin layer Navier-Stokes (TLNS) equations are solved on the *inner grid* and the Euler equations are solved on the *outer grid*. The two solvers are coupled at the interface between grids at each stage of the Runge-Kutta time integration which was used. The TLNS solver functions in two modes; *explicit* and *semi-implicit*. With the TLNS solver running in explicit mode, results were obtained at $M_\infty = 0.85$, $\alpha = 15°$ for Reynolds numbers of 2×10^6 and 7.5×10^6 on a grid with 276,000 cells. Explicit results at $M_\infty = 0.75$, $\alpha = 16°$ and $Re = 1 \times 10^5$ were also obtained on a coarse grid of 36,000 cells. The semi-implicit TLNS mode was not found to be robust, and only one solution at $M = 0.75$, $\alpha = 16°$, Re $= 1 \times 10^5$ was found on a very coarse grid of 11,400 cells. The results are qualitatively correct, but quantitative assessment is impossible, as no experimental data is currently available.

Thesis Supervisor: Earll M. Murman,
Professor of Aeronautics and Astronautics

Acknowledgements

Thank you to all those who contributed, directly or indirectly, to the completion of this thesis.

There are a few people who deserve special mention for their contribution. First, a big "thank you" to Earll Murman for giving me the opportunity to attend an institution of the caliber of MIT, and for his insights on the philosophy and process of research. Thanks to Richard Taylor for your friendship and for providing some really nice drawings for this thesis. Thanks also to Sean Tavares for many interesting discussions about life, the universe, and vortical flows. To my "opera buddy" Teva Regule, thank you for your friendship and the "culturizing" I received over the past two years. Last, but not least, thanks to my parents for *everything*.

This research was supported by the National Aeronautics and Space Administration under Grants No. NAG-1-507 and NAG-1-855, monitored by Mr. Duane Melson, NASA Langley Research Center, Hampton VA.

Contents

List of Figures

Chapter 1
Introduction

1.1 Background

Since the first powered flight by Orville and Wilbur Wright in 1903, the science of flight has progressed at an extremely fast pace, greatly aided by the occurence of two world wars. Had it not been for the wars, aviation would probably not be where it is today, since the amount of money spent on aviation research and development in times of war is significant.

Early aircraft were awkward, flimsy contraptions which flew rather slowly and often were a hazard to the person flying them, due to inherent instability of the design. As aviation progressed, advances were made in many areas. Planes were better built, more stable, had longer ranges and endurances and, most of all, greater speeds.

The delta wing came into existence as speeds attained by aircraft approached and eventually surpassed the speed of sound. Designers found that the important parameter for drag in the transonic and supersonic flow regimes was the Mach number *normal* to the leading edge, rather than the freestream Mach number. By sweeping the leading edges of the wing, the normal Mach number, and thus the drag, could be reduced. This concept has been used extensively, as shown by the fact that most supersonic aircraft have delta wings. In addition, all modern aircraft travelling at transonic or greater speeds have wings which, though not delta wings, are swept.

Experience with delta wings showed that this type of wing had a certain phenomenon associated with it at high angles of attack, namely the formation of leading edge vortices. The phenomenon of leading edge vortices is of interest because of its complexity and its

direct practical application in enhancing the high α characteristics of flight vehicles. It also has the potential for seriously degrading the performance of a flight vehicle and causing it to spin and crash. For these reasons, an understanding of leading edge vortices is essential.

1.1.1 Occurrence of Leading Edge Vortices

The flow over a given delta wing depends greatly upon the geometry of the wing and the flow conditions which are involved. A general classification suggested by Newsome and Kandil [23] places delta wing flows into four diferent categories as follows:

1. **Low α.** In this flow regime, the flow is attached over the wing and no vortices are present.

2. **Moderate to High α.** At the higher angles of attack, 3-D boundary layer separation takes place and large lee-side vortices are shed. The flow is stable and symmetric.

3. **Very high α.** Here, the vortices are still being shed but there is a loss of stability of the vortex system, resulting in the onset of asymmetry, unsteadiness, vortex bursting, or a combination of these.

4. **Extreme α's.** This regime includes angles of attack up to 90°. An unsteady, diffuse wake is present. Periodic vortex shedding may also be present, depending on flow parameters and geometry.

The flows of interest in this investigation fall into Category 2, i.e. stable and symmetric flows.

1.1.2 Structure of a Leading Edge Vortex System

Stanbrook and Squire [30] devised a grouping for the different vortical flow topologies which occur in the "stable and symmetric" regime described in the previous section. The grouping is based on the Mach number and angle of attack *normal* to the leading edge,

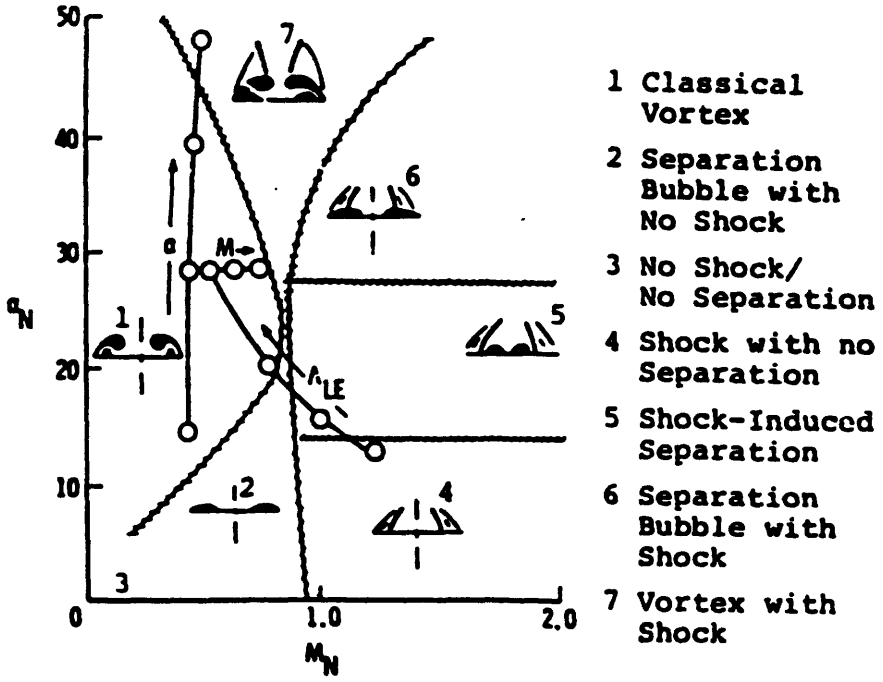

Figure 1.1: Classification of leading edge vortex flows, based on M_N and α_N.

defined respectively by

$$M_N = M_\infty \sqrt{1 - \sin^2 \Lambda \cos^2 \alpha} \tag{1.1}$$

and

$$\alpha_N = \tan^{-1}\left(\frac{\tan \alpha}{\cos \Lambda}\right). \tag{1.2}$$

This classification was later improved and expanded by Miller and Wood [20] as a result of their extensive experimental investigation on sharp-edged delta wings. Their classification in terms of the normal Mach number and angle of attack is shown in Figure 1.1. Although this figure is for sharp edged delta wings, the occurence of vortices over blunt edged delta wings may be classified in the same way, the only difference being that, due to the bluntness, the exact flow conditions for separation may vary and the vortices will not be as strong. The flows of interest in this investigation fall into Region 1 of Figure 1.1, labeled the "classical vortex" regime.

Figure 1.2 shows the structure of a typical leading-edge vortex from Region 1 of Figure 1.1. As it flows around the leading edges, the wing boundary layer is unable to negotiate the adverse pressure gradient it encounters on the leeward side of the wing just inboard of the

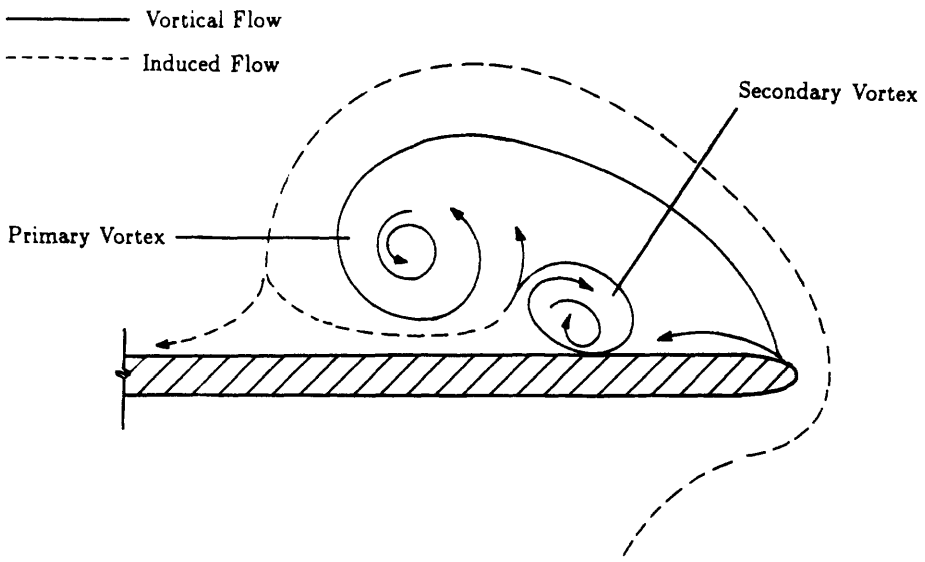

Figure 1.2: Structure of a leading edge vortex system.

leading edge. It therefore separates from the body, producing a free shear layer of distributed vorticity. In contrast to sharp edged delta wings, for blunt delta wings the separation point is not fixed at the leading edge, but varies with the particular flow conditions.

The shear-layer shed from the leading edge proceeds to roll up into a spiral under its own influence and forms the core of the leading edge vortex. The high fluid velocities in the vortices sitting above the wing create low pressure regions on the lee-side of the wing. This is shown in Figure 1.3.

The swirling flow about the vortex core induces a spanwise velocity in the boundary layer fluid on the wing surface. This outward flow is subject to an adverse pressure gradient as it moves from the low pressure area beneath the leading edge vortex (known as the "primary" vortex) to the recompression area just inboard of the leading edges. Unable to negotiate this pressure gradient, the boundary layer again separates into a free vorticity layer, giving rise to a second, smaller vortex (known as the "secondary" vortex) located underneath the large primary vortex. The secondary vortex rotates in the sense opposite to the primary. The location of separation of the secondary vortex is very dependent on

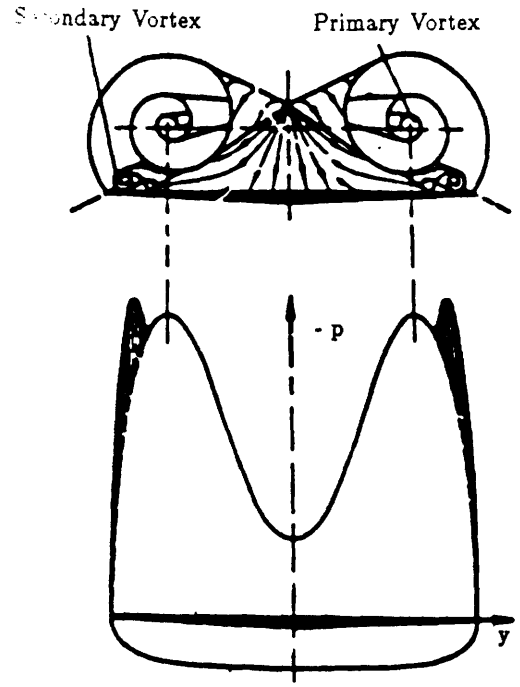

Figure 1.3: Pressure distribution on the lee-side of the wing.

the state of the boundary layer on the wing. In transonic flows the secondary vortex has a large effect on the location of the primary vortex compared with subsonic and transonic regimes. The secondary vortex produces a smaller and less powerful low pressure region on the wing than the primary vortex, as shown in Figure 1.3.

1.1.3 Practical Consequences of Leading Edge Vortices

Leading edge vortices may have positive or negative effects on the wing from which they are shed, depending on the details of their occurrence. Some associated phenomena, such as vortex bursting, may also occur and influence the characteristics of the wing.

When they are symmetric and steady, leading edge vortices have a positive effect on the wing, providing regions of low pressure on the wing surface below where the vortex is located. These low pressure regions formed on the upper side of the wing make the lift significantly higher than would be obtained if the flow remained attached, especially at higher angles of attack. The vortices are also responsible for the non-linear lift characteristics of delta wings at high angles of attack.

Leading edge vortices may be used to provide thrust by using a wing modification known as a "vortex flap". Vortex flaps are downward deflections of the leading edges. This creates a part of the wing surface which has a normal vector with a component in the upstream direction. The intent is to place the shed vortex over this surface, so that the low pressure on the vortex flap creates a net forward force.

In the case of asymmetric vortices, the effects on the wing are negative. The different vortex strengths cause the low pressure regions on either side of the wing to vary in magnitude and size, creating a net rolling moment about the wing centerline. When the asymmetric vortices are unsteady, they may lead to the phenomenon of "wing rock" where the rolling moment alternates in sign, causing the wing to oscillate in roll and possibly become unstable. For an arbitrary configuration, current knowledge is not sufficient to predict the onset of vortex asymmetry.

Another phenomenon associated with leading edge vortices (and vortices in general) is vortex bursting, which involves a sudden stagnation of the vortex core and subsequent breakdown of the vortex itself. Vortex bursting is usually a highly unsteady phenomenon, which can lead to large unsteady pressure transients in the region of the burst. Vortex bursting is a very poorly understood phenomenon. A couple of excellent reviews of progress made to date in vortex bursting are the papers by Liebovich [16] and Hall [9].

At lower angles of attack, the vortex usually bursts downstream of the trailing edge of the wing, having little effect on the wing itself. As the angle is increased, however, the breakdown location moves upstream until it does occur over the wing. When this happens, there is a sudden loss of lift due to the increased pressure under the bursting vortex. Associated with this is a nose-up pitching moment due to higher pressures on the aft part of the wing.

The large unsteady pressure transients associated with vortex bursting can cause structural fatigue in complex configurations where the vortex shed from one part of the vehicle bursts near another component. An example of this is the F-18 fighter, where a vortex being shed from the forebody strakes was bursting near the vertical tail, causing fatigue at the tail's base.

1.2 Previous Delta Wing Studies

Previous studies of vortex flows over delta wings may be broken down into two groups. In the first group are methods in which the topology of the separation is known *a priori* and the vortex sheet arising from the separation is modeled in an approximate way. The second group consists of methods in which the leading edge vortex is captured as part of the solution of the governing equations of the flow. Hoeijmakers [10] presents a good review of methods from the first group. Only studies which fall into the second group are of interest here.

Studies which involve solution of the governing equations of the flow may be further divided into two types; Euler studies and Navier-Stokes studies. For an excellent and thorough review of work done so far in these two areas, consult the paper by Newsome and Kandil [23]. Brief discussion of some representative studies in each category is given below.

The Euler equations have been used for numerous studies, such as those by Goodsell [8], Murman and Rizzi [22], Rizzi et al [27], Powell [25] and Powell et al [26]. Most of the studies done so far have captured the primary vortex and correlated fairly well with experiments in the subsonic and supersonic flow regimes. The correlation is poorest in the transonic regime [8]. None of the Euler studies have captured the secondary vortex, since this is a viscous phenomenon which is not modeled in the equations. The Euler equations have been applied almost exclusively to sharp edged delta wings, since the separation point can be captured correctly due to the Kutta condition.

A controversy surrounding Euler studies, however, is how the equations manage to capture the separation of the vortex when there is no viscosity present and thus no mechanism to induce the separation. Powell [25], who solved the conical Euler equations for flow over a flat delta wing, explains that the separation at the leading edges is due to the numerical dissipation in the algorithm. The numerical dissipation mimics the physical viscosity and causes a Kutta condition to exist at the sharp edge.

Not many Euler studies have been done for blunt-edged delta wings, as the absence of

a fixed separation point such as a sharp leading edge would render the equations incapable of predicting the separation point correctly. In fact, for flow over blunt edged delta wings, Kandil and Chuang [13] showed conclusively that the numerical dissipation is the cause of the inviscid separation. By varying the amount of dissipation, they could cause separation not to occur.

Numerous investigators have used the full Navier-Stokes equations or simplifications thereof to compute delta wing flows. Since they include the effects of physical viscosity, these equations should capture all phenomena such as the secondary separation and subsequent secondary vortex formation. They should also correctly capture the separation point of the primary vortex on a blunt leading edge. As with any viscous simulation, however, the limitations of present turbulence models limits the quality of solutions.

A few representative studies using Navier-Stokes equations are those by Fujii and Schiff [7], Loyd [17], and Rizzi et al [27]. All of these captured the viscous flow physics quite well. Fujii and Schiff [7] solved the thin layer Navier Stokes equations over a strake-delta wing, using an extremely fine grid with 850,000 points. Both primary and secondary vortices were captured and, surprisingly, vortex bursting was captured at high angles of attack.

Loyd [17] uses a *semi-implicit* algorithm to solve for the laminar flow over a blunt edged, elliptical delta wing. The primary and secondary vortices were captured, and two different mechanisms for separation at blunt leading edges were identified, namely separation due to shockless recompression and separation involving a leading edge shock. Loyd found the semi-implicit algorithm to be very efficient, as it eliminates the stiffness associated with N-S solutions.

1.3 Present Study

1.3.1 Approach to Problem

The approach selected for this investigation consists of splitting the flow domain into two regions, and solving a different set of governing equations on each. The thin layer Navier-Stokes (TLNS) equations are solved on an inner region near the body and in the wake where the viscous effects are significant. The Euler equations are solved in the outer part of the domain, where the flow is effectively inviscid.

The first step is to generate a grid consisting of two sections; an *inner* grid and an *outer* grid. The *inner* grid surrounds the wing and wake, and it is on this grid that the TLNS equations are solved. The *outer* grid surrounds the inner grid over most of the domain. This is the grid that the Euler equations are solved on.

After obtaining the grid, an Euler solver and a thin layer Navier-Stokes solver are combined into a hybrid solver; each flow solver is run on its respective portion of the grid, with the solvers being coupled at the interface between the two regions of the grid. Both solvers are run until a steady state solution has been obtained in the domain.

1.3.2 Research Objectives

There are three main objectives to this investigation. The first objective is to capture all the relevant flow physics for this type of problem. This includes capture of the secondary (and possibly tertiary) vortex, and correct prediction of the separation point at the leading edge. Capture of the secondary vortex is especially important in transonic flows, as the secondary vortex has its greatest influence on the flow in this regime. This objective is accomplished by applying a viscous solver to the area where all these viscous-dominated phenomena are taking place.

The second objective is to acheive computational efficiency while still capturing all flow phenomena. This is accomplished by running the Euler solver in the region away from the

wing where the flow is essentially inviscid, thus avoiding the computation of viscous terms from the Navier-Stokes equations which would be of negligible magnitude anyway.

The third objective is to investigate the use of these multi-zone, multi-solver approaches on this type of flows, assessing the degree to which they succeed in correctly computing the flow and the computational savings which they offer.

1.3.3 Thesis Summary

The first chapter of this thesis gives a brief qualitative description of the phenomenon of leading edge vortices, along with a discussion of a few of the computational studies which have been done to date on this topic. Also covered are the general approach to the problem which is taken here and the objectives of the research.

Chapter 2 presents the two sets of governing equations whose solutions are the topic of this thesis, along with the boundary conditions which are applied to each. Some comments are made on the assumptions which lead to each equation set and on their applicability to different problems.

Chapter 3 provides the details of the grid generation done for this research. Both the inner grid and outer grid are described, along with the combination grid made up of the two. Examples of both grids and the combination are given for the grid size which was used in the final computations.

Chapter 4 gives the details of both solution algorithms which were used and describes how the two were coupled together to give the hybrid solver. Algorithm characteristics which are common to both algorithms are discussed, followed by discussion of those items which are particular to each of the solvers.

Chapter 5 presents the computational results obtained using the hybrid code, giving qualitative data in the form of color graphics and some limited quantitative data. Discussion of the results is given.

Finally, Chapter 6 discusses some of the conclusions that may be drawn from the results of the investigation.

Chapter 2

Governing Equations

The full, unsteady 3-D Navier-Stokes (N-S) equations govern the flow of any Newtonian continuum fluid. They are a system of five coupled, nonlinear partial differential equations. The fact that they are coupled and non-linear, in addition to the fact that the classification of the equations changes depending on the flow regime (for unsteady flows they exhibit both hyperbolic and elliptic behavior), makes it almost impossible to obtain an analytical solution for a general case. A few exact solutions for these equations have been obtained, although they are for very simplified cases. These analytic solutions provide a great deal of insight about the N-S equations, however they are not general enough to be directly applicable to practical problems.

Given the difficulty in obtaining analytical solutions, one must resort to numerical solutions of the equations using a computer. While algorithms are available which solve the full N-S equations, they are currently very expensive in terms of computing resource time. Consequently, people have resorted to solving simplifications of the equations which still give physically meaningful results but whose numerical solution is much cheaper.

This chapter describes the simplifications of the N-S equations whose numerical solution is the topic of this thesis. These are the Euler equations and the thin-layer Navier-Stokes (TLNS) equations. No detailed derivation is given, since this is found in many good texts on fluid mechanics. Instead, the equations will be presented along with their respective boundary conditions, some explanation will be given about the assumptions inherent in these simplifications, and the range of applicability of the simplified equation sets will be discussed briefly.

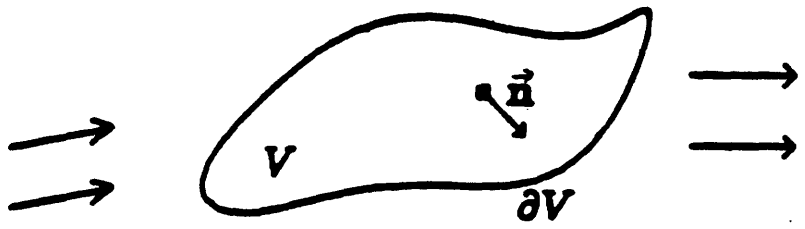

Figure 2.1: Control volume for Navier-Stokes derivation.

2.1 Basic Considerations

2.1.1 Conservation Laws

The full Navier-Stokes equations are the mathematical model of a fluid flow which comes about by application of conservation laws to the flow. Usually, they are derived by considering the flow through a control volume fixed in space, as shown in Fig 2.1.

Application of the conservation laws to this control volume simply states that the rate of change of a physical property inside the control volume must be equal to the flux of this property through the surface of the control volume plus the rate of change due to causes internal to the control volume. In the case of a fluid flow, there are three physical quantities which are preserved: mass, momentum and energy. Mass and energy, being scalar quantities, yield one equation each. Momentum, being a vector quantity, yields three equations since conservation must occur in each of its three component directions. Usually one more equation, the equation of state, is needed to obtain closure for the system of equations.

2.1.2 Turbulence Issues

Turbulence has long been considered the greatest challenge of fluid mechanics, due to the extreme complexity and poor understanding of the phenomenon. Mathematicians and scientists have been working on turbulence for years, to little avail.

Turbulence has both positive and negative effects. Negative effects include higher friction drag, higher pressure loss and more noise. Positive effects are that the turbulent boundary layer is more energetic and can negotiate larger regions of adverse pressure gradient, delaying separation of the flow over bodies. The overall effect of turbulence is to cause the flow to behave as if the viscosity had increased by a large amount. Whatever the effect of turbulence, it is significant in most flows where it occurs, so that some attempt has to be made to predict its effects.

It is generally accepted that turbulent flows are governed by the unsteady Navier-Stokes equations, so that, in principle, all the effects of turbulence could be captured simply by solving the equations. Unfortunately this is not possible, due to the extremely small length and time scales involved in turbulent fluid motion. To resolve such scales numerically for a problem of practical significance would require a tremendous concentration of grid points. No computer available today, and no computer in the forseeable future, could handle the amount of computation, memory and storage involved.

This being the case, the approach taken by the fluid dynamics community has been to come up with ways to model the turbulence without calculating it from fundamental considerations. Numerous models exist in varying degrees of complexity, however none has been able to correctly predict the flow over a wide range of conditions. Most of the models contain so many free parameters that the flow solution can be "tuned" to match experimental data, giving the impression that the model works well. While they may do a good job in this *post-dictive* mode, use of present turbulence models as *predictive* tools is cast in doubt by this.

Although the NTF delta wing is designed to be run at very high Reynolds numbers where the flow will certainly be turbulent, all the cases run in this study are assumed

laminar in nature. It has been found that use of a turbulence model is not that important to capture the relevant physics of the primary vortex. In fact, Powell [25] has shown that the artificial dissipation added to stabilize the time marching algorithm becomes large enough to overwhelm the physical viscosity in the region of the primary vortex. Turbulence has a more profound effect on the wing boundary layer and the secondary vortex. The usual effect of turbulence is to decrease the size and strength of the secondary vortex and to increase the magnitude of the suction pressure peak under the primary vortex.

Since the use of turbulence models as predictive tools is doubtful, and this study is essentially a predictive study since experimental data is not yet available for comparison, it was felt that laminar calculations would probably compare with experimental data just as well as turbulent ones.

2.1.3 General Form

The 3-D Navier-Stokes equations and all of its simplified forms may be written in a general form. This form is an integral equation in Cartesian coordinates over a control volume V , surrounded by a control surface ∂V, as follows (see Fig 2.1):

$$\frac{\partial}{\partial t}\iiint_V \mathbf{U}\, dV + \iint_{\partial V} \vec{\mathbf{F}}(\mathbf{U}) \cdot \vec{\mathbf{n}} dS = 0. \tag{2.1}$$

Equation 2.1 is written in conservative vector form. Writing the equation in this form allows correct capturing of flow features such as shocks and, in unsteady flows, the shock speed. It is written with the assumption that body forces such as gravity or electromagnetic forces are negligible and that there are no internal heat or mass sources. In other words, changes in the control volume are due only to fluxes through its surface.

For both simplifications of the Navier-Stokes equations considered in this thesis, the vector of state variables, $\mathbf{U}$, is the same and is given by $\mathbf{U} = (\rho \;\; \rho u \;\; \rho v \;\; \rho w \;\; \rho E)^T$, where ρ is the density, u, v, w are components of velocity in the x, y, and z directions repectively, and E is energy per unit mass. The only aspect in which the simplifications of the N-S

equations differ is in the contents of the flux vector $\vec{F}$.

2.2 Euler Equations

2.2.1 Assumptions and Applications

The Euler equations are the governing equations for a flow which is totally inviscid. These equations are obtained by elimination of the all of the viscous terms and all of the heat transfer terms from the full Navier-Stokes equations.

The Euler equations are more useful than approaches such as the potential equations because they can convect and capture vorticity in the field. They are also capable of capturing vortex stretching and tilting in three-dimensional flows. Using the potential approach, these phenomena must be modeled by placing singularities in the flow field. The Euler equations can also capture flow discontinuities such as shocks as part of their solution (a shock wave is a weak solution of the hyberbolic equations), and the solutions satisfy the Rankine-Hugoniot equations relating the end states across a shock.

While treating the flow as totally inviscid may seem like quite a severe restriction on the applicability of the equations, they in fact model a large range of conditions. The Euler equations are most applicable for flows where the Reynolds number is large enough that the effects of viscosity are confined to a thin layer near a solid boundary. In cases where the viscous layer is very thin compared with the characteristic dimension of the flowfield, the inviscid portion of the flow field may be solved for independently and the Euler equations may be used.

One of the best applications of the Euler equations is for preliminary design purposes, as they usually do a good job of predicting the pressure field, and thus the lift and pressure drag, on a configuration. The cost of using Euler equations is substantially lower than the cost of Navier-Stokes solutions, and the Euler solutions are obtained faster.

2.2.2 Equations

The Euler equations are given by the general form shown in Equation 2.1, with the flux vector $\vec{\mathbf{F}}$ being divided into components in each of the three coordinate directions

$$\vec{\mathbf{F}} = \mathbf{F}\vec{i} + \mathbf{G}\vec{j} + \mathbf{H}\vec{k}\,, \tag{2.2}$$

where the unit vectors $\vec{i}$, $\vec{j}$, and $\vec{k}$ denote the x, y, and z coordinate directions , respectively.

The flux vectors $\mathbf{F}$, $\mathbf{G}$, and $\mathbf{H}$ are given by

$$\mathbf{F} = \begin{pmatrix} \rho u \\ \rho u^2 + P \\ \rho uv \\ \rho uw \\ \rho uH \end{pmatrix}, \quad \mathbf{G} = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + P \\ \rho vw \\ \rho vH \end{pmatrix}, \quad \mathbf{H} = \begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + P \\ \rho wH \end{pmatrix}, \tag{2.3}$$

where H is the total enthalpy, $H = E + P/\rho$, and P is the static pressure.

The equation of state which is used to close the system of equations is the perfect gas equation, $P = \rho RT$, which is combined with the definition of internal energy per unit mass to give the closing relation

$$E = \frac{1}{\gamma - 1}\frac{P}{\rho} + \frac{1}{2}(u^2 + v^2 + w^2). \tag{2.4}$$

2.2.3 Boundary Conditions

As with any PDE or system of PDE's, it is important to specify boundary conditions to have a well-posed mathematical problem. For the Euler equations, there are two types of boundary conditions which are usually specified; conditions in the far field, away from any solid boundaries, and conditions at a solid boundary. In addition, some sort of Kutta condition must be applied for flow over a lifting body.

The boundary condition at the upstream and lateral far field is simply that the flow conditions there should be the same as freestream, i.e. that no influence of the body on the

flow is felt there. This is expressed mathematically as

$$\lim_{\substack{x\to-\infty \\ y,z\to\pm\infty}} \mathbf{U} = \mathbf{U}_\infty. \tag{2.5}$$

At the downstream far field boundary ($x \to +\infty$) the flow does not return to free stream conditions, due to the wake created by the presence of the body and due to changes in flow properties which occur across any shocks which are present. While the pressure at the downstream far field is usually assumed to have returned to the free stream value, other properties, such as the entropy, will be at values different from the free stream. For flow over a lifting body, the condition that flow perturbations in the streamwise direction vanish in the Trefftz plane applies. This is stated mathematically as

$$\lim_{x\to+\infty} \frac{\partial \mathbf{U}}{\partial x} = 0. \tag{2.6}$$

The boundary condition at a solid boundary states that there can be no flow through the boundary. In other words, the velocity normal to the boundary must vanish there. This is expressed mathematically as

$$\vec{V} \cdot \hat{n} = 0, \tag{2.7}$$

where $\vec{V}$ is the total velocity vector and $\hat{n}$ is an outward pointing unit vector normal to the body surface.

The final boundary condition which must be applied is the Kutta condition. This condition makes the flow solution unique by fixing the circulation at that value which causes the flow to separate from the body at sharp edges. This condition is not usually enforced explicitly, in Euler or Navier-Stokes calculations, as it evolves as part of the solution. With other schemes, such as vortex panel methods and potential methods, it is necessary to enforce the Kutta condition explicitly.

2.3 Thin Layer Navier-Stokes Equations

2.3.1 Assumptions and Applications

The thin layer Navier-Stokes equations come about in an *ad hoc* manner in a way similar to the boundary layer equations (see Anderson et al, [1]). As with the boundary layer equations, an order-of-magnitude analysis is performed on the full Navier-Stokes equations and terms of order $1/\sqrt{Re_L}$ and smaller are neglected. This means that, in the viscous terms, all derivatives in the directions parallel to the body surface are dropped since they are substantially smaller than the derivatives normal to the surface.

Unlike the boundary layer equations, however, in the thin layer Navier-Stokes equations the normal momentum equation is not simplified to $\partial p/\partial y = 0$. Instead, all the terms in the normal momentum equation are retained, with the exception of the viscous derivatives parallel to the body, as described above.

Experience by numerous investigators has shown that the TLNS equations may be applied to a wide range of practical problems. Since most of the terms in the normal momentum equation are left intact, the equations may be used to calculate separated and reverse regions of flow in a straightforward manner. Also, the equations can easily model flows with large pressure gradients. With sufficient resolution, the phenomenon of vortex bursting may even be captured with this equation set, as shown by Fujii and Schiff [7].

2.3.2 Equations

The TLNS equations again have the same general form as Equation 2.1. The flux vector $\vec{\mathbf{F}}$ is again split into components in three directions, however each component now consists of viscous and inviscid contributions

$$\vec{\mathbf{F}} = (\mathbf{F}_I - \mathbf{F}_V)\vec{i} + (\mathbf{G}_I - \mathbf{G}_V)\vec{j} + (\mathbf{H}_I - \mathbf{H}_V)\vec{k}, \tag{2.8}$$

where the unit vectors $\vec{i}$, $\vec{j}$, and $\vec{k}$ represent the x, y and z directions, respectively. The subscripts I and V refer to the inviscid and viscous contributions, respectively.

The inviscid parts of the flux vectors are the same as those previously given for the Euler equations:

$$\mathbf{F}_I = \begin{pmatrix} \rho u \\ \rho u^2 + P \\ \rho uv \\ \rho uw \\ \rho uH \end{pmatrix}, \quad \mathbf{G}_I = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + P \\ \rho vw \\ \rho vH \end{pmatrix}, \quad \mathbf{H}_I = \begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + P \\ \rho wH \end{pmatrix}, \tag{2.9}$$

where H is the total enthalpy, $H = E + P/\rho$, and P is the static pressure.

The viscous fluxes, which represent the effects of shearing stresses and heat conduction across the surface of the control volume, are given by

$$\mathbf{F}_V = \begin{pmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{xz} \\ u\tau_{xx} + v\tau_{xy} \\ +w\tau_{xz} - q_x \end{pmatrix}, \quad \mathbf{G}_V = \begin{pmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \tau_{yz} \\ u\tau_{xy} + v\tau_{yy} \\ +w\tau_{yz} - q_y \end{pmatrix}, \quad \mathbf{H}_V = \begin{pmatrix} 0 \\ \tau_{xz} \\ \tau_{yz} \\ \tau_{zz} \\ u\tau_{xz} + v\tau_{yz} \\ +w\tau_{zz} - q_z \end{pmatrix}. \tag{2.10}$$

Before application of the thin layer approximation, the components of the stress tensor, τ, are given by

$$\begin{aligned} \tau_{xx} &= \frac{2}{3}\mu\left(2\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} - \frac{\partial w}{\partial z}\right) & \tau_{xy} &= \tau_{yx} = \mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) \\ \tau_{yy} &= \frac{2}{3}\mu\left(2\frac{\partial v}{\partial y} - \frac{\partial u}{\partial x} - \frac{\partial w}{\partial z}\right) & \tau_{xz} &= \tau_{zx} = \mu\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) \\ \tau_{zz} &= \frac{2}{3}\mu\left(2\frac{\partial w}{\partial z} - \frac{\partial u}{\partial x} - \frac{\partial v}{\partial y}\right) & \tau_{yz} &= \tau_{zy} = \mu\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right), \end{aligned} \tag{2.11}$$

and the heat flux terms are given by

$$q_x = -k\frac{\partial T}{\partial x} \qquad q_y = -k\frac{\partial T}{\partial y} \qquad q_z = -k\frac{\partial T}{\partial z}. \tag{2.12}$$

By introducing the equation of state for a perfect gas, given by

$$H = C_p T + .5(u^2 + v^2 + w^2) \tag{2.13}$$

and introducing the Prandtl number $Pr = \frac{\mu C_p}{k}$, we may eliminate the temperature T as a variable and re-write the heat flux terms as

$$\begin{aligned} q_x &= -\frac{\mu}{Pr}\left(\frac{\partial H}{\partial x} - u\frac{\partial u}{\partial x} - v\frac{\partial v}{\partial x} - w\frac{\partial w}{\partial x}\right) , \\ q_y &= -\frac{\mu}{Pr}\left(\frac{\partial H}{\partial y} - u\frac{\partial u}{\partial y} - v\frac{\partial v}{\partial y} - w\frac{\partial w}{\partial y}\right) , \\ q_z &= -\frac{\mu}{Pr}\left(\frac{\partial H}{\partial z} - u\frac{\partial u}{\partial z} - v\frac{\partial v}{\partial z} - w\frac{\partial w}{\partial z}\right) . \end{aligned} \tag{2.14}$$

The value of the Prandtl number Pr was taken as 0.72 for the laminar flow solutions presented in this thesis. The Stokes hypothesis is assumed to hold for the viscous terms, and the coefficient of viscosity is related to the local enthalpy by the Sutherland formula

$$\frac{\mu}{\mu_\infty} = \left(\frac{h}{h_\infty}\right)^{3/2} \frac{h_\infty + h_1}{h + h_1} , \tag{2.15}$$

where h_1 is a dimensional constant (= 111421 in *SI* units), and ∞ refers to free stream conditions. This relation closes the system of equations.

2.3.3 The Thin Layer Approximation

When dealing with a Cartesian coordinate system, the Cartesian directions will not, in general, correspond exactly with the directions parallel and normal to the body. The body-fitted coordinate system, however, does correspond to the directions parallel and normal to the surface. Because the thin layer form of the Navier Stokes equations only retains the viscous terms in the body-normal direction, the viscous stresses must be transformed from the Cartesian x, y, z system to body-fitted ξ, η, and ς coordinates.

By applying the chain rule of differentiation, derivative terms with respect to x, y, and z may be transformed to derivatives with respect to ξ, η, and ς coordinates:

$$\begin{aligned} \frac{\partial}{\partial x} &= \frac{\partial \xi}{\partial x}\frac{\partial}{\partial \xi} + \frac{\partial \eta}{\partial x}\frac{\partial}{\partial \eta} + \frac{\partial \varsigma}{\partial x}\frac{\partial}{\partial \varsigma} = \xi_x\frac{\partial}{\partial \xi} + \eta_x\frac{\partial}{\partial \eta} + \varsigma_x\frac{\partial}{\partial \varsigma} \\ \frac{\partial}{\partial y} &= \frac{\partial \xi}{\partial y}\frac{\partial}{\partial \xi} + \frac{\partial \eta}{\partial y}\frac{\partial}{\partial \eta} + \frac{\partial \varsigma}{\partial y}\frac{\partial}{\partial \varsigma} = \xi_y\frac{\partial}{\partial \xi} + \eta_y\frac{\partial}{\partial \eta} + \varsigma_y\frac{\partial}{\partial \varsigma} \\ \frac{\partial}{\partial z} &= \frac{\partial \xi}{\partial z}\frac{\partial}{\partial \xi} + \frac{\partial \eta}{\partial z}\frac{\partial}{\partial \eta} + \frac{\partial \varsigma}{\partial z}\frac{\partial}{\partial \varsigma} = \xi_z\frac{\partial}{\partial \xi} + \eta_z\frac{\partial}{\partial \eta} + \varsigma_z\frac{\partial}{\partial \varsigma} . \end{aligned} \tag{2.16}$$

Applying the thin layer assumption that derivatives in the downstream ξ and cross stream ς directions are small compared to derivatives in the body-normal η direction,

$$\frac{\partial}{\partial \eta} >> \frac{\partial}{\partial \xi}\,,\ \frac{\partial}{\partial \varsigma}\,, \tag{2.17}$$

all ξ and ς derivative terms may be dropped from (2.16) to obtain

$$\frac{\partial}{\partial x} \cong \eta_x \frac{\partial}{\partial \eta}\,, \qquad \frac{\partial}{\partial y} \cong \eta_y \frac{\partial}{\partial \eta}\,, \quad \text{and} \qquad \frac{\partial}{\partial z} \cong \eta_z \frac{\partial}{\partial \eta}\,. \tag{2.18}$$

Inserting (2.18) into (2.11) gives the components of the thin layer stress tensor,

$$\begin{aligned}
\tau_{xx} &\cong \frac{2}{3}\mu\left(2\eta_x\frac{\partial u}{\partial \eta} - \eta_y\frac{\partial v}{\partial \eta} - \eta_z\frac{\partial w}{\partial \eta}\right) & \tau_{xy} = \tau_{yx} &\cong \mu\left(\eta_y\frac{\partial u}{\partial \eta} + \eta_x\frac{\partial v}{\partial \eta}\right)\\
\tau_{yy} &\cong \frac{2}{3}\mu\left(2\eta_y\frac{\partial v}{\partial \eta} - \eta_x\frac{\partial u}{\partial \eta} - \eta_z\frac{\partial w}{\partial \eta}\right) & \tau_{xz} = \tau_{zx} &\cong \mu\left(\eta_z\frac{\partial u}{\partial \eta} + \eta_x\frac{\partial w}{\partial \eta}\right)\\
\tau_{zz} &\cong \frac{2}{3}\mu\left(2\eta_z\frac{\partial w}{\partial \eta} - \eta_x\frac{\partial u}{\partial \eta} - \eta_y\frac{\partial v}{\partial \eta}\right) & \tau_{yz} = \tau_{zy} &\cong \mu\left(\eta_z\frac{\partial v}{\partial \eta} + \eta_y\frac{\partial w}{\partial \eta}\right)\,,
\end{aligned} \tag{2.19}$$

Likewise, applying (2.18) to (2.14), the thin layer heat fluxes are given by

$$\begin{aligned}
q_x &\cong -\frac{\mu}{Pr}\left(\eta_x\frac{\partial H}{\partial \eta} - u\eta_x\frac{\partial u}{\partial \eta} - v\eta_x\frac{\partial v}{\partial \eta} - w\eta_x\frac{\partial w}{\partial \eta}\right)\\
q_y &\cong -\frac{\mu}{Pr}\left(\eta_y\frac{\partial H}{\partial \eta} - u\eta_y\frac{\partial u}{\partial \eta} - v\eta_y\frac{\partial v}{\partial \eta} - w\eta_y\frac{\partial w}{\partial \eta}\right)\\
q_z &\cong -\frac{\mu}{Pr}\left(\eta_z\frac{\partial H}{\partial \eta} - u\eta_z\frac{\partial u}{\partial \eta} - v\eta_z\frac{\partial v}{\partial \eta} - w\eta_z\frac{\partial w}{\partial \eta}\right)\,.
\end{aligned} \tag{2.20}$$

While these expressions seem no simpler than the original stress tensor and heat flux terms, they in fact allow considerable computational savings, since derivatives in this approximation need be evaluated only on the two faces of each volume that are parallel to the body. Evaluating the thin layer terms on the other four faces is unnecessary since, by assumption, face-to-face changes in the streamwise and cross stream viscous terms are negligible.

2.3.4 Boundary Conditions

As with the Euler equations, the TLNS equations also require boundary conditions to form a well posed problem. The boundary conditions for the TLNS equations are the same as those for the Euler equations with the exception of the condition at a solid boundary.

Instead of only having the velocity *normal* to the flow go to zero, the no-slip condition is applied so that the *total* velocity at the surface is zero:

$$|\vec{V}_{surface}| = 0. \tag{2.21}$$

2.4 Non-dimensionalization

Non-dimensionalization of variables in the governing equations allows characteristic flow parameters such as the Mach number and Reynolds number to be identified and varied independently. The free stream values of density, speed of sound, and viscosity ρ_∞, a_∞, μ_∞, and a characteristic length c were chosen as reference values. Introducing these quantities gives the nondimensional variables:

$$\begin{array}{lll} x' = \dfrac{x}{c} & y' = \dfrac{y}{c} & z' = \dfrac{z}{c} \\ u' = \dfrac{u}{a_\infty} & v' = \dfrac{v}{a_\infty} & w' = \dfrac{w}{a_\infty} \\ \rho' = \dfrac{\rho}{\rho_\infty} & p' = \dfrac{p}{\rho_\infty a_\infty^2} & t' = \dfrac{t}{c/a_\infty} \\ \mu' = \dfrac{\mu}{\mu_\infty}\dfrac{M_\infty}{Re_\infty} & E' = \dfrac{E}{a_\infty^2} & H' = \dfrac{H}{a_\infty^2} \, . \end{array} \tag{2.22}$$

The forms of the nondimensional governing equations, equation of state, and viscosity model are identical to that of the dimensional equations. The free stream vector in terms of nondimensional variables is

$$\mathbf{U}_\infty = \begin{pmatrix} 1 \\ M_\infty \cos\alpha \\ 0 \\ M_\infty \sin\alpha \\ \dfrac{1}{\gamma(\gamma-1)} + \dfrac{M_\infty^2}{2} \end{pmatrix}, \tag{2.23}$$

where α is the angle of attack and the yaw angle, β, is assumed zero. The non-dimensional free stream pressure is $p_\infty = 1/\gamma$.

To avoid carrying around many symbols, the primes will be dropped from all equations that follow in this document. Henceforth, all variable references are to be interpreted as applying to non-dimensional variables.

Chapter 3
Grid Generation

The first step in any computational study of fluid flow is the generation of a suitable grid about the body of interest to provide the points at which the discretized governing equations will be solved. Exactly what constitutes a "suitable" grid is not clear. In general, this is a qualitative measure obtained by visual inspection of the grids. The qualities most sought are smoothness of the grid, lack of skewness of cells and sufficient number of points in the areas of the flow where interesting phenomena are occurring, so that these phenomena may be adequately resolved.

This chapter describes the generation of the grids for the NTF delta wing which were used to perform calculations. First, the geometry of the NTF wing itself is described, along with some of its geometric characteristics. Next, in keeping with the philosophy of describing the whole before the parts, the combined grid system is described. This is followed by detailed descriptions of the generation of the two zones of the grid. Finally, some graphics results are presented showing the grids obtained using the generation methods described.

3.1 The NTF Delta Wing

3.1.1 Description

The NTF delta wing [19] is a blunt edged delta wing which is to be tested at NASA's Langley Research Center in the National Transonic Facility (thus the name "NTF Delta Wing"), a cryogenic nitrogen tunnel used for testing at very high Reynolds numbers.

Table 3.1: Characteristics of the NTF Delta Wing.

Parameter	Value
Aspect Ratio	1.865
Root Chord(ft)	2.140
Span(ft)	1.996
t/c	0.034
Sweep (deg)	65.00

The wing has a set of three interchangeable leading edges, having radii of curvature $(r/\bar{c})$ of 0.05, 0.15 and 0.30. These are designated leading edges 1, 2, and 3, respectively. All the computations in this thesis were done using leading edge # 2. For data collection, the NTF wing is instrumented with a large number of static pressure taps on the bottom and top surfaces. The taps are located in spanwise rows at several chordwise stations. In addition, there are pressure taps located right on the leading edges.

The geometric characteristics of the NTF wing are shown in Table 3.1. The wing has no twist or camber.

Note that, in the description which follows, all lengths have been normalzed by the root chord of the wing.

3.1.2 Physical Coordinate System

Figure 3.1 shows the physical coordinate system which is used for this study. Note that this figure is merely a schematic, as it depicts a wing with sharp leading edges while the actual NTF wing has blunt leading edges. The coordinate system is right handed, Cartesian and body-fixed. The coordinate origin, $(x, y, z) = (0, 0, 0)$, is located at the apex of the wing, with the x coordinate being in the chordwise direction, the y coordinate in the spanwise direction and the z coordinate in the direction normal to the wing at its centerline.

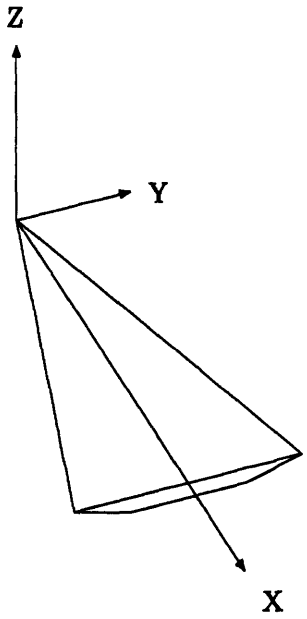

Figure 3.1: Schematic of body-fixed physical coordinate system.

3.1.3 Wing Geometry

The NTF delta wing consists of four distinct sections; a flat plate region which makes up most of the wing, analytically defined leading edges and trailing edge closure regions, and an analytically defined sting fairing where the sting fits in to the back of the wing. This layout is shown in Figure 3.2. Section views of the wing in streamwise and spanwise directions are shown in Figures 3.3 and 3.4, respectively.

In the present study, all of these regions are modeled except for the sting fairing, which was not modeled due to ambiguity in the model description by Luckring [19]. Thus, for computational purposes, the wing is considered to be an isolated body with no solid structure (such as a sting) supporting it.

The semi-thicknesses ϕ of the wing in the leading-edge and trailing edge closure regions are defined by the equation

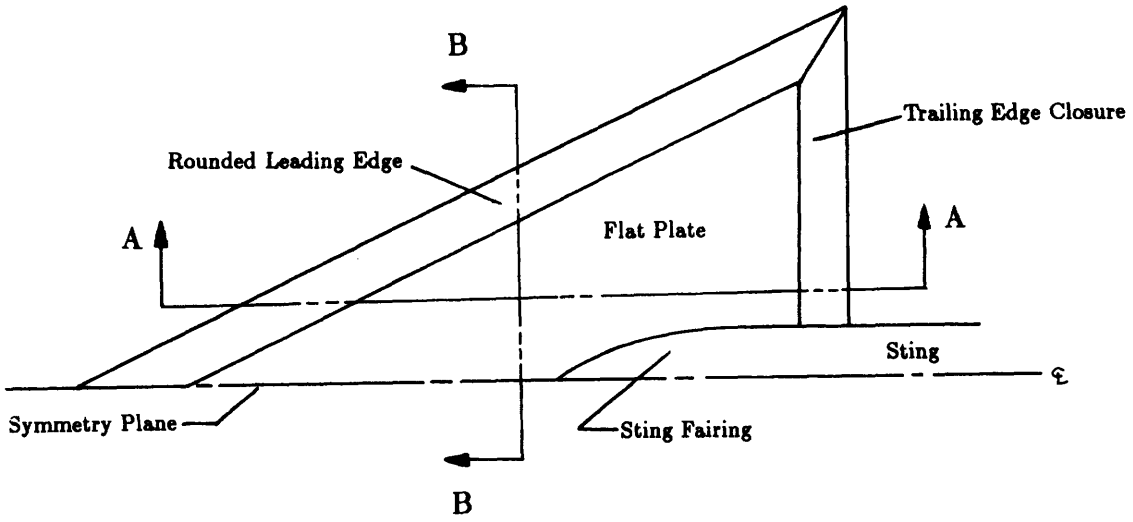

Figure 3.2: Schematic Layout of the NTF Delta Wing.

Section A – A

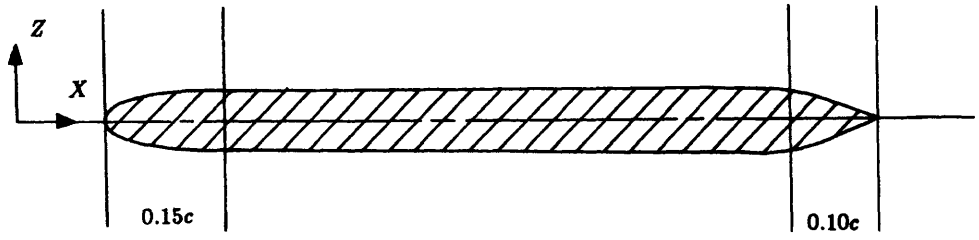

Figure 3.3: Streamwise section of NTF wing (section A-A in Figure 3.2).

Section B – B

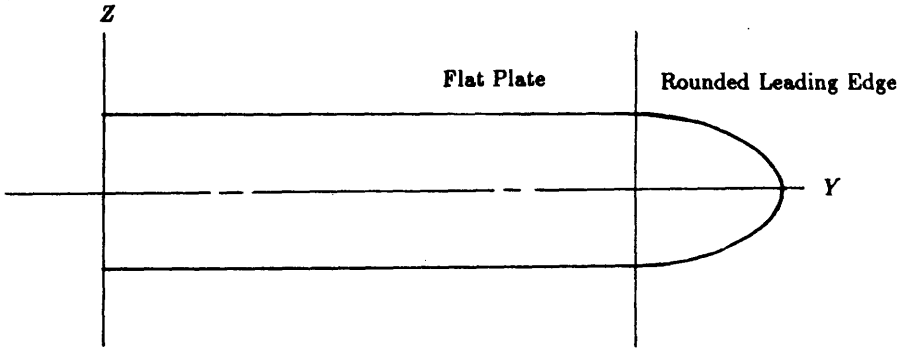

Figure 3.4: Spanwise section of NTF wing (section B-B in Figure 3.2).

$$\phi(\theta) = \pm\, K\, (a\sqrt{\theta} + b\theta + c\theta^2 + d\theta^3) \quad ; \quad 0 \leq \theta \leq 1 \tag{3.1}$$

where the definition of θ and the constants a, b, c, d and K varies depending on whether one is describing the leading or trailing edge region.

The leading edge geometry is constant in the streamwise direction at each spanwise station (such as Section A-A, shown in Figure 3.4) except for a region near the wingtip where the geometries of the leading edge and the trailing edge closure region intersect, resulting in a discontinuity in the surface. The rounded leading edges are designed to meet the flat plate portion of the wing at a streamwise distance of 15 percent root chord aft of the leading edge, with continuity through the second derivative. In the leading edge region, θ is given by

$$\theta = \frac{(x - x_0)}{K} \quad \text{where} \quad \begin{cases} x_0 = x_{le} \\ K = 0.15 \end{cases} . \tag{3.2}$$

Here, x_{le} is the x location of the leading edge for each y location, i.e. at any station along the span. For easier stacking of 2-D grid planes in the chordwise direction, Equation 3.2 was transformed so it would be in terms of y by replacing $(x - x_0)$ with $(y - y_0)\tan\Lambda$.

The trailing edge closure region is designed to exhibit a sharp trailing edge and to match the flat plate portion of the wing at 90 percent root chord, also with continuity through the second derivative. In the trailing edge region, since it is constant in the spanwise direction at any given x station, θ is defined in terms of x as

$$\theta = \frac{(x_0 - x)}{K} \quad \text{where} \quad \begin{cases} x_0 = 1 \\ K = 0.10 \end{cases} . \tag{3.3}$$

In the flat plate portion of the wing, the wing semi thickness is described simply by

$$\phi(\theta) = \pm\ 0.017000800036901 \quad ; \quad 0 \leq \theta \leq 1 \tag{3.4}$$

where θ in this case is given by

$$\theta = \frac{(x - x_0)}{K} \quad \text{where} \quad \begin{cases} x_0 = x_{le} + 0.15 \\ K = 0.90 - x_0 \end{cases} . \tag{3.5}$$

In the region near the wingtips where the leading and trailing edge geometries intersect, one uses either definition 3.2 or 3.3, depending on which side of the discontinuity between the two regions one is interested in calculating.

The values of the constants a, b, c, and d for the three interchangeable leading edges and the trailing edge closure region are given in Appendix A.

3.2 Features of the Combined Grid

3.2.1 Computational Coordinate Systems

The computational coordinate system used to generate both grids is shown in Figure 3.5. The figure shows that the computational coordinates, like the physical ones, form a right-handed system with the i index denoting the streamwise direction, the j index denoting the body-normal direction and the k index wrapping around the wing from bottom to top. The i, j, k indices represent the body-fitted coordinates ξ, η, ς respectively.

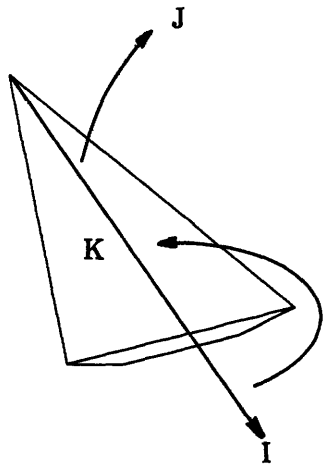

Figure 3.5: Schematic of right-handed computational coordinate system.

An important point is that each grid has its own separate indexing system. For example, the i index of the apex in the inner grid is different than the i index for the same location in the outer grid (the apex is at $i = 1$ in the inner grid and $i = 11$ in the outer grid). For each grid, the ranges of indices in the i, j and k directions are $1 \rightarrow i_{max}$, $1 \rightarrow j_{max}$ and $1 \rightarrow k_{max}$, respectively. In addition, each direction i, j and k has two "ghost cells" at each extreme. For example, in the i direction there is two ghost cells having indices 0 and -1 at the lower extreme and two ghost cells having indices $i_{max} + 1$ and $i_{max} + 2$ at the upper extreme.

3.2.2 Description

The combination grid system on which all solutions are computed consists of two parts; an *inner* grid and an *outer* grid. The two grids are constructed to fit together, with grid lines having C^0 continuity across the interface between them. This is shown schematically in Figure 3.6. Both grids are assembled by stacking of two-dimensional grids in the chordwise (x) direction, giving them both an O-H topology.

The inner grid begins at the apex of the wing at $x = 0$ and runs back to the outflow boundary of the domain, located one chord length downstream of the trailing edge at $x = 2$. This grid is generated algebraically at each chordwise station along the wing by computing a family of wing shapes in each $y - z$ plane. The 2-D planes at each chordwise station are then stacked to give the 3-D grid. At the apex, the wing comes to a point, creating a singularity in the grid where all the grid points at that chordwise station collapse onto the apex. The outer boundary of the inner grid is not fixed in advance, but is a result of the generation. The TLNS equations are solved on this grid, since it lies closest to the wing.

The outer grid begins at the upstream boundary of the domain, located at $x = -1$, and extends all the way to the downstream boundary of the domain. In the area of the wing and wake, the outer grid surrounds the inner grid, with the outer boundary of the *inner* grid being the inner boundary of the *outer* grid. This grid is generated using an elliptic method to compute two-dimensional grids in the $y - z$ plane at each chordwise (x) station on the wing. The 2-D grids at each x location are then stacked to give the 3-D grid. The Euler equations are solved on this grid.

Both the inner and outer grids are generated only over half of the wing. This is because only symmetrical cases are being computed (that is, cases with zero yaw angle, β) so that it is more efficient to compute on half the wing and apply symmetry conditions at the symmetry boundary. Better resolution at places of interest on the wing is thus acheived by placing the same number of points over half the wing that one would use in computing the whole wing.

3.3 Inner Grid

This section describes in detail how the inner grid is generated. Note that this grid is generated only between the apex and the trailing edge of the wing. The portion of the inner grid in the wake is computed simply by extending the grid from the trailing edge station to the downstream boundary of the domain. In other words, at a streamwise location in the wake, the inner grid looks exactly like the inner grid at the trailing edge station ($x = 1$).

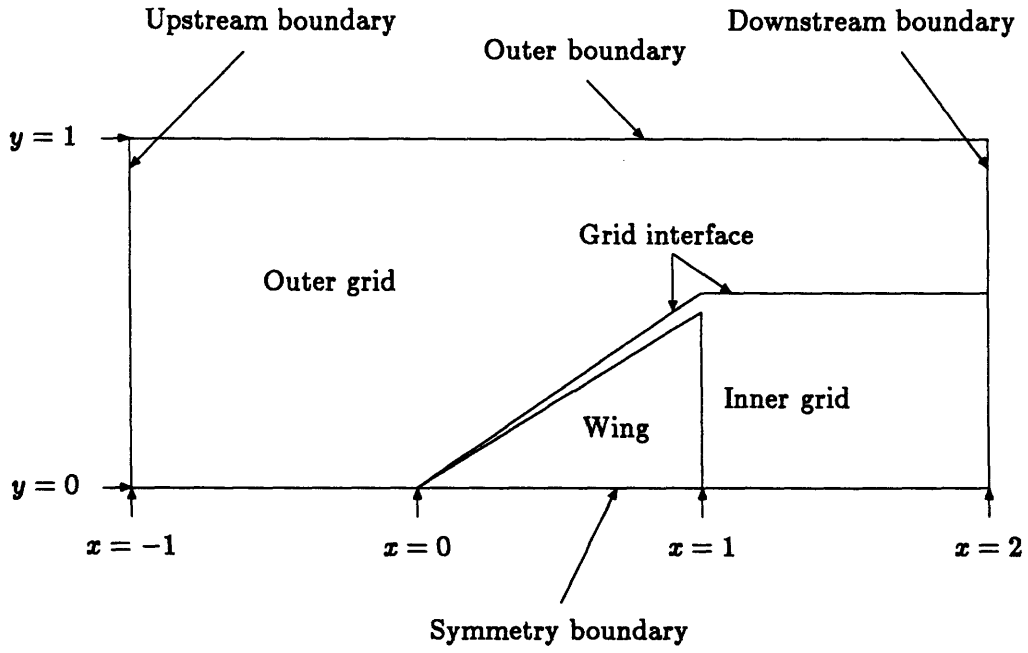

Figure 3.6: Planform schematic view of the combined computational grid.

3.3.1 Generation

The inner grid is generated algebraically. This method was chosen for generation because the equations for the wing surface are known analytically, making it easy to generate a grid which is orthogonal to the wing, as required by the TLNS equations. The resulting grid is orthogonal over most of the wing, except at the leading edges due to the sweep of the wing and at the trailing edge due to the slope of the closure region in the streamwise direction.

The grid is generated by calculating a series of wing families in the $y-z$ plane. In other words, if the wing surface is defined by $z = \phi(\theta)$, then each grid surface in the body-normal direction is given by $z_j = f(j)\phi(\theta)$, where $f(j)$ is a factor depending on what body-normal (j) location is being considered.

Since the equation of the wing is known analytically, it is straightforward to calculate the slope of a constant j grid line at any point:

$$\frac{dz}{dy}_j = f(j)\left(\frac{d\phi}{dy}\right)_{j=0} = f(j)\left(\frac{d\phi}{d\theta}\frac{d\theta}{dy}\right)_{j=0}. \tag{3.6}$$

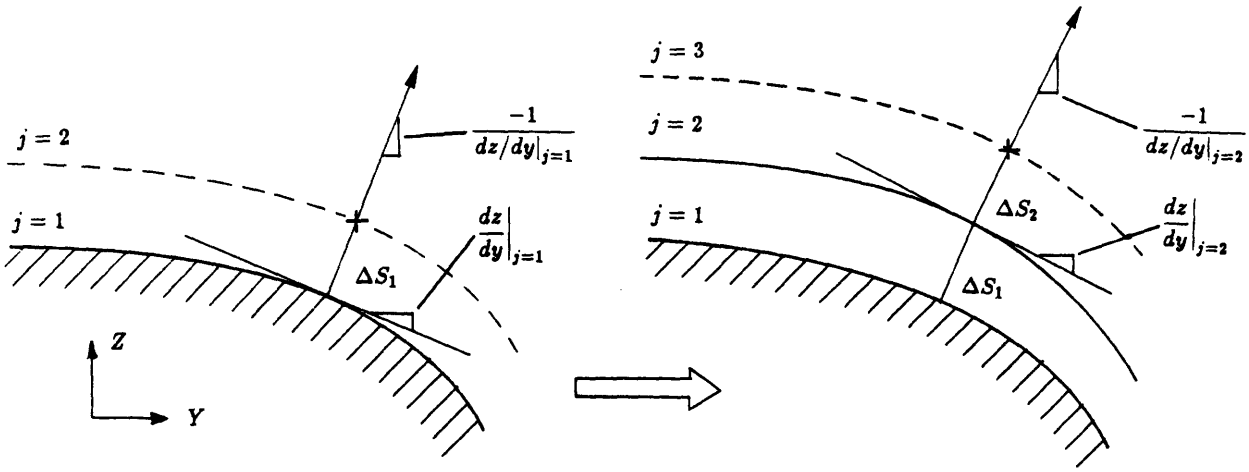

Figure 3.7: Generation of the Inner Grid.

Knowing the slope of the grid line at any point, the slope of the line which is normal to the grid line at that point is determined using the fact that the product of the slope of a line and the slope of its normal should equal -1. Thus, the slope of the normal line is given by

$$m_j = \frac{-1}{dz/dy_j}. \tag{3.7}$$

Once the slope of the normal line is known, the next grid point in the body-normal (j) direction is placed along the normal line at a distance ΔS_j from the current position. This is illustrated in Figure 3.7. The distribution of ΔS_j is specified beforehand and determines what the clustering of points along the normal lines will be.

There are two places where the above formulation will run into trouble, namely the flat plate portion of the wing, where the slope of the normal line is infinite, and the tip of the wing where the slope of the grid line will be infinite. This is easily handled, however, simply by placing the next point out along a vertical or horizontal line, respectively.

3.3.2 Grid Stretching and Scaling

With any Navier-Stokes calculation, it is necessary to have a great deal of resolution in the lower part of the boundary layer near a solid wall due to the small physical scales which govern viscous diffusion in the direction normal to the wall. Many grid points can be placed in the lower part of the boundary layer by "stretching" the grid. In the formulation of the previous section, this is done by specifying a small ΔS adjacent to the boundary and then increasing the ΔS_j's as a function of the body-normal distance from the wing. This gives a continuous increase in cell size between the wing surface and the outer grid boundary.

In this case, the grid is stretched exponentially. To begin with, body-normal "sizes" for the innermost and outermost grid cells, $\Delta\sigma_1$ and $\Delta\sigma_{jmax}$, are specified. The difference between these two quantities will determine the amount of stretching. The "sizes" of the grid cells in between are then specified by

$$\Delta\sigma_j = \Delta\sigma_1 \, e^{j\Delta V} \tag{3.8}$$

where

$$\Delta V = \frac{\log(\Delta\sigma_{jmax}) - \log(\Delta\sigma_1)}{jmax - 1}. \tag{3.9}$$

To account for the growth of the boundary layer over the wing, the outer boundary of the inner grid is scaled by some measure of the local boundary layer thickness. Thus, the *actual* cell sizes ΔS_j are determined by multiplying all of the $\Delta\sigma_j$ by a scale factor which reflects the thickness of the boundary layer at that particular chordwise location:

$$\Delta S_j = \Delta\sigma_j \frac{\delta}{\sum_{j=1}^{jmax-1} \Delta\sigma_j}. \tag{3.10}$$

Here, δ is the boundary layer thickness at the chordwise location, calculated somewhat un-sophisticatedly as a flat plate boundary layer using

$$\delta = \frac{4.960x}{\sqrt{Re_x}} \quad ; \quad 0 \le x \le 1. \tag{3.11}$$

Note that the boundary layer scaling takes place only from the apex to the trailing edge of the wing. Aft of the trailing edge, the grid is simply extended backwards from the last station on the wing to the downstream boundary of the domain.

3.4 Outer Grid

This section describes how the outer (Euler) part of the grid is generated. Note that this grid is also generated only between the apex and the trailing edge of the wing. The portion of the outer grid in the wake is computed simply by extending the grid from the trailing edge station to the downstream boundary of the domain. Likewise, the portion of the grid upstream of the apex is computed by extending the apex station forward to the upstream domain boundary. In other words, at any streamwise location in the wake, the outer grid looks exactly like the outer grid station at the trailing edge ($x = 1$), and at any streamwise location upstream of the apex, the grids looks exactly like the grid at the apex station ($x = 0$).

3.4.1 Generation

The outer grid is generated elliptically. This method was chosen because of the smooth grids it generates, and to provide a smooth transition of the grid from the inner boundary (which is the outer boundary of the inner grid) to the outer boundary, which is a circle of radius equal to one root chord.

The elliptic system which is solved for the grid is given by

$$\begin{aligned} (k\xi_y)_y + (k\xi_z)_z &= 0 \\ (k\eta_y)_y + (k\eta_z)_z &= 0, \end{aligned} \tag{3.12}$$

where k is a factor analogous to the thermal conductivity in a heat transfer problem and will later be used to cluster the grid. This approach was first used by Giles and Drela in their theses. A further description is given by Murman and Giles [21].

To be useful, the above system must first be transformed so that y and z are the dependent variables. We do this by applying the standard transformation (see Anderson et al [1]) and we get

$$\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} = \frac{J}{k}(k_\xi z_\eta - k_\eta z_\xi)$$

$$\alpha z_{\xi\xi} - 2\beta z_{\xi\eta} + \gamma z_{\eta\eta} = \frac{-J}{k}(k_\xi y_\eta - k_\eta y_\xi) \tag{3.13}$$

where α, β, and γ are coefficients and J is the Jacobian of the transformation. These quantities are defined as follows:

$$\begin{aligned} \alpha &= {y_\eta}^2 + {z_\eta}^2 \\ \beta &= y_\xi y_\eta + z_\xi z_\eta \\ \gamma &= {y_\xi}^2 + {z_\xi}^2 \\ J &= y_\xi z_\eta - y_\eta z_\xi. \end{aligned} \tag{3.14}$$

This set of equations is then written in finite difference form , using central differences for all derivatives. The system is then solved using an SLOR iterative method for elliptic equations. The convergence criteria for the grid is that the RMS change in the y and z positions of the grid from one iteration to the next be less than a specified small amount:

$$\Delta y_{RMS}, \Delta z_{RMS} \leq 5 \times 10^{-4}. \tag{3.15}$$

3.4.2 Grid Clustering

Solution of the Equations 3.13 does not always produce a grid that places enough points in the places of interest in the flowfield. It is therefore necessary to find a way to alter the grid which is generated by these equations. Alteration is accomplished through variation of the factor k in the equations. This factor, if correctly varied, will cause the system to cluster more points in certain places in the domain. In this case, we want to cluster more points near the interface between the grids so that phenomena near the interface are captured correctly and so that there is not a large disparity in size between the outermost cells of the *inner* grid and the innermost cells of the *outer* grid.

A clustering function k which gives adequate results is given by

$$k = C_1 + C_2 \frac{(j - j_{min})}{(j_{max} - j_{min})} \times \frac{(i - i_{le})}{C_3} \tag{3.16}$$

where the factors C_1, C_2 and C_3 may be varied to change the clustering. The values $5, 15$ and 2 , respectively, were found to work well. The last factor containing i was inserted to

turn off the clustering at the apex and make it more powerful at each streamwise station. If this wasn't done, the clustering effect was too severe near the apex, causing grid points to be pulled inside the inner grid. If the clustering was turned off in the whole domain, however, the resolution obtained near the trailing edge was poor.

3.5 Grid Results

This section gives graphic results of the grids which were obtained using the generation methods described. The results are limited to two-dimensional views of the grids at different locations. This is because, at the time of writing, no program which produces 3-D hardcopy is available to the author.

Results of the inner grid generation are presented first, followed by results of the outer grid generation. The section concludes with some views of the combination grid.

3.5.1 Inner Grid

The results shown are the grids which were used in the final flow calculations in this thesis. The inner grid has dimensions of $40 \times 20 \times 120$ cells in the i, j and k directions respectively. This gives a total of 96,000 computational cells in the inner grid. The 40 streamwise cells are distributed with 10 cells upstream, 25 cells on the wing and 15 cells in the wake.

Figure 3.8 shows a planform view of the inner grid. The wing is located between $x = 0$ and $x = 1$. It can be seen that , on the wing, constant k grid lines are rays emanating from the apex. The portion of the grid between the trailing edge and the downstream boundary represents the computational wake boundary, which is just a slit of zero thickness in the $y - z$ plane. It is also seen from this figure that more points have been concentrated near the leading edges to capture the phenomena there.

Figure 3.9 shows a view of the inner grid at a typical chordwise station on the wing.

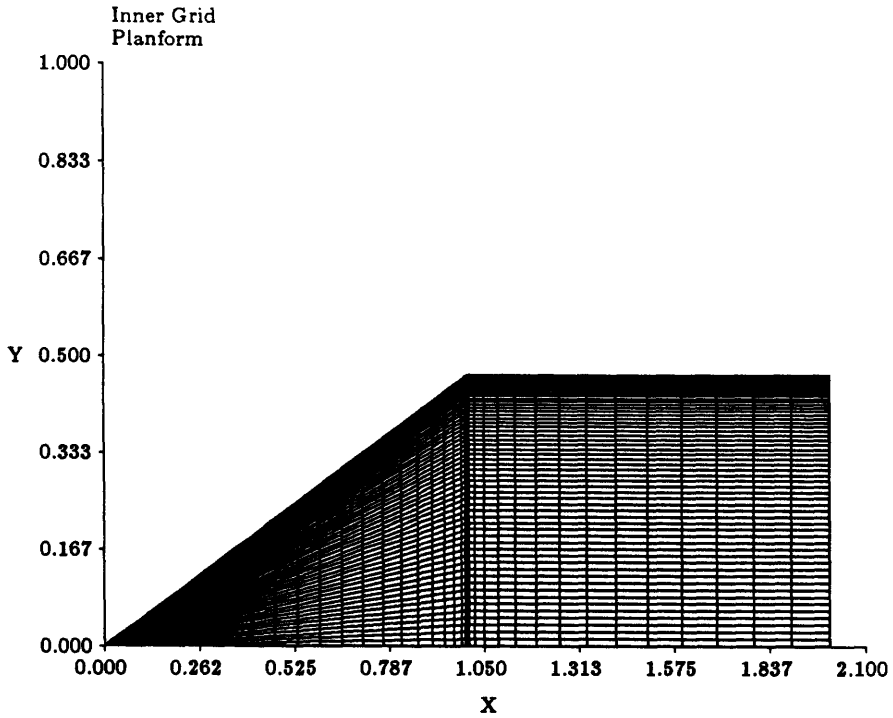

Figure 3.8: Planform view of the inner grid.

The shape of the wing at a chordwise section and the bluntness of the leading edge are evident. The orthogonality of the grid can be seen in this view, along with the effects of the grid stretching which was applied.

Figure 3.10 is a blowup of the leading edge region of Figure 3.9. This figure more clearly shows the orthogonality of the inner grid and the concentration of grid points close to the body.

Figure 3.11 is a view of the inner grid at a spanwise station, in this case the wing centerline. The boundary layer scaling of the grid is clearly seen and the extension of the grid from the trailing edge to the downstream boundary of the domain is evident. Note that the vertical scale of this figure is greatly exaggerated, so that the orthogonality of the grid in the trailing edge and apex regions is actually much better than it looks.

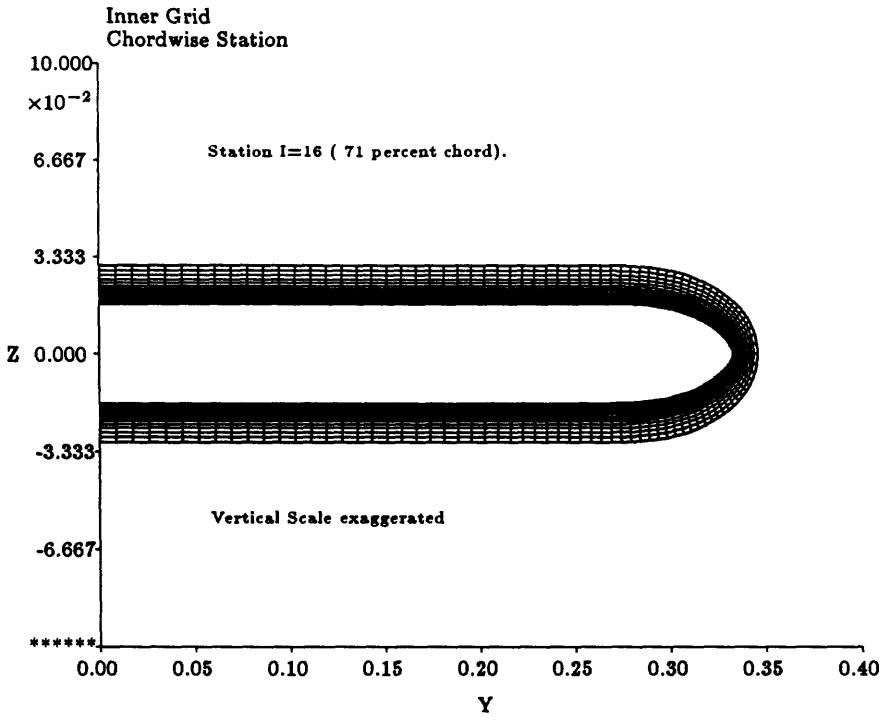

Figure 3.9: View of inner grid at a typical streamwise location on the wing.

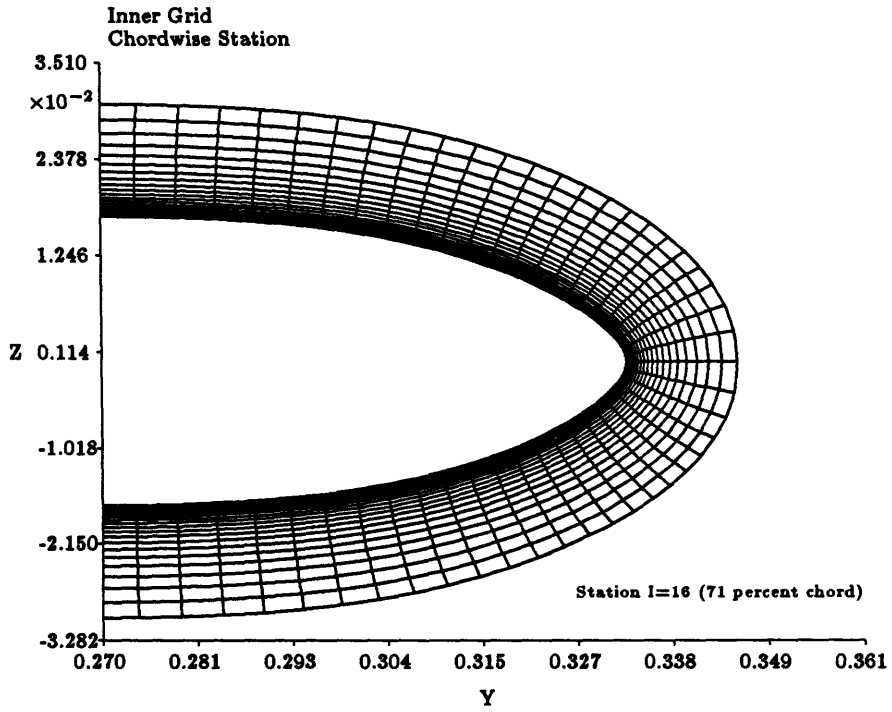

Figure 3.10: Blowup of inner grid at leading edge to show stretching.

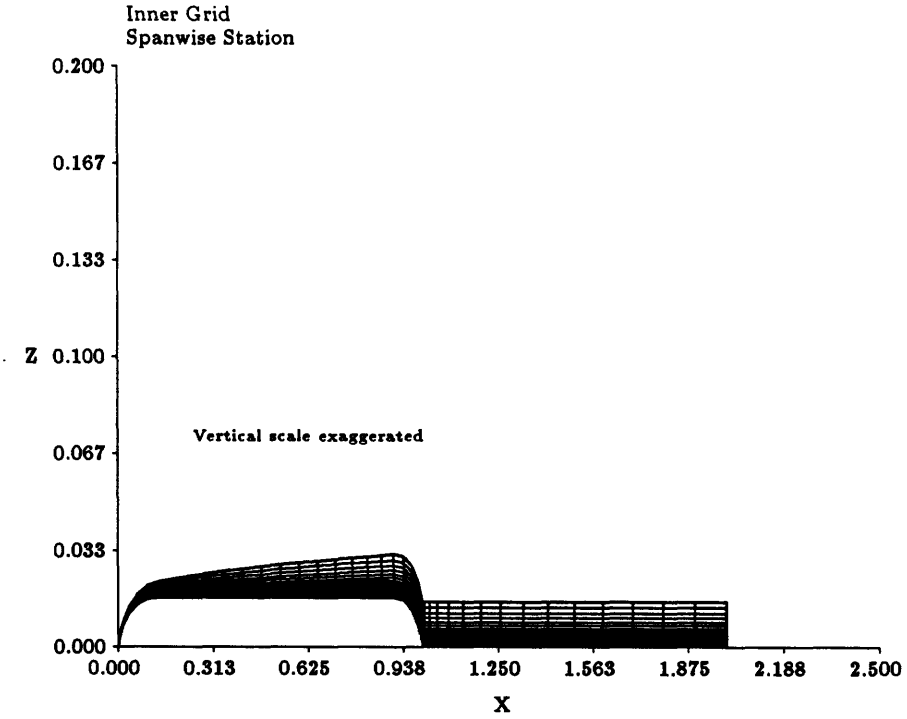

Figure 3.11: View of inner grid at a spanwise station (wing centerline).

3.5.2 Outer Grid

The dimensions of the outer grid used in the final calculations is $50 \times 30 \times 120$ cells in the i, j and k directions, respectively , for a total of 180,000 computational cells in the outer grid. The streamwise distribution of cells places 25 on the wing and 15 in the wake.

Figure 3.12 shows a view of the outer grid at a typical chordwise station at 71 percent chord. Some degree of grid clustering is evident in this figure. Figure 3.13 is an enlargement of the leading edge region of Figure 3.12. The grid clustering is shown better in this view. The boundary where the points are clustered is the outer boundary of the inner grid. The clustering is not as severe as one is used to seeing, since the main intent here is to try to match the cell sizes of the two grids at the interface. The secondary goal of the clustering is to increase the resolution in this area, which seems to be accomplished, especially at the leading edge.

Figure 3.14 is a view of the outer grid at a spanwise location, in this case the wing centerline (which is also the symmetry plane). Here, the wing is between $x = 0$ and $x = 1$,

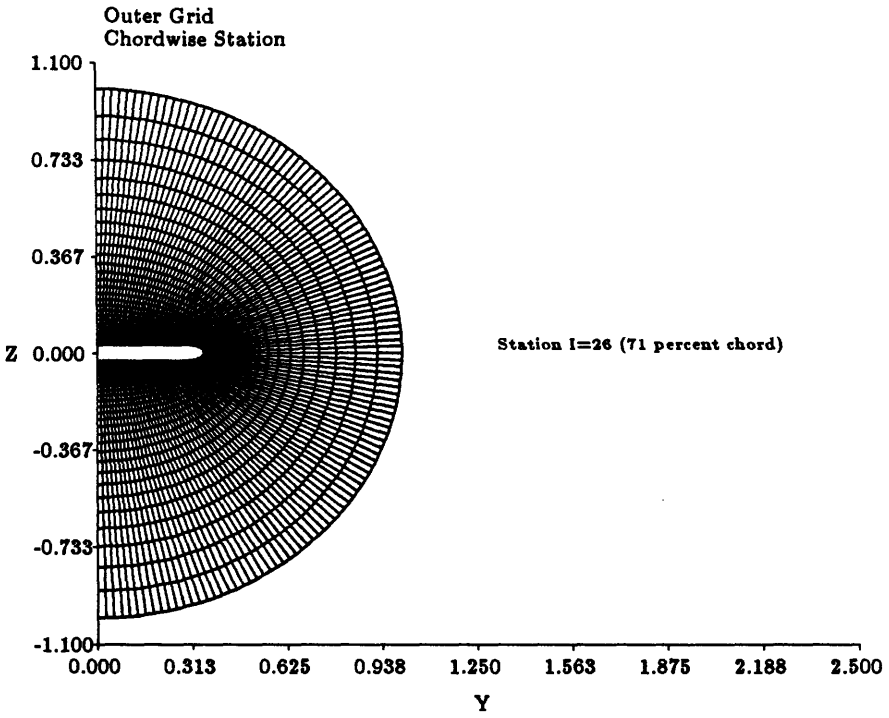

Figure 3.12: View of outer grid at a typical streamwise location on the wing.

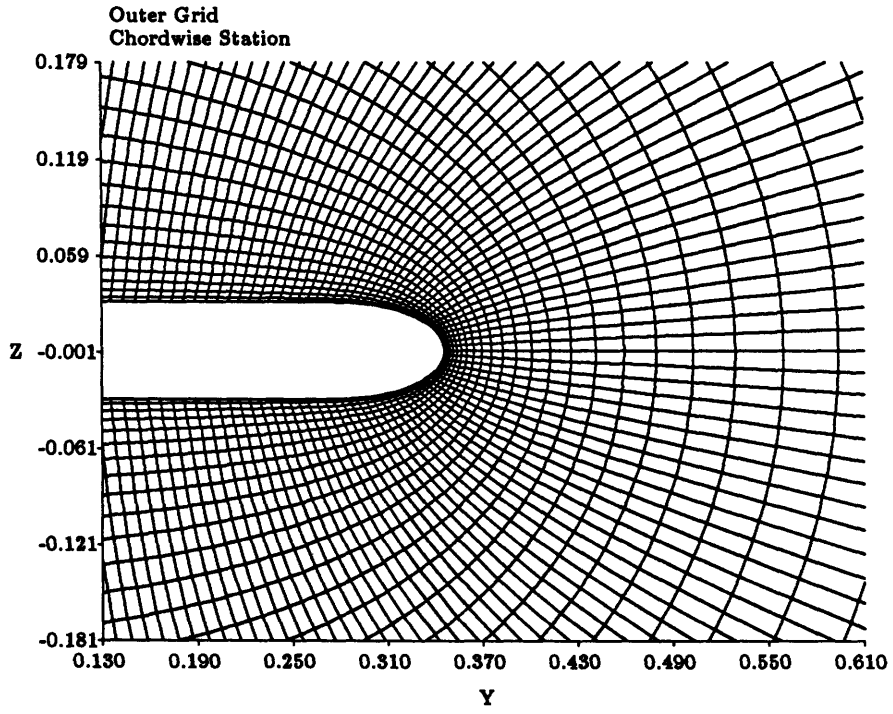

Figure 3.13: Blowup of outer grid at leading edge.

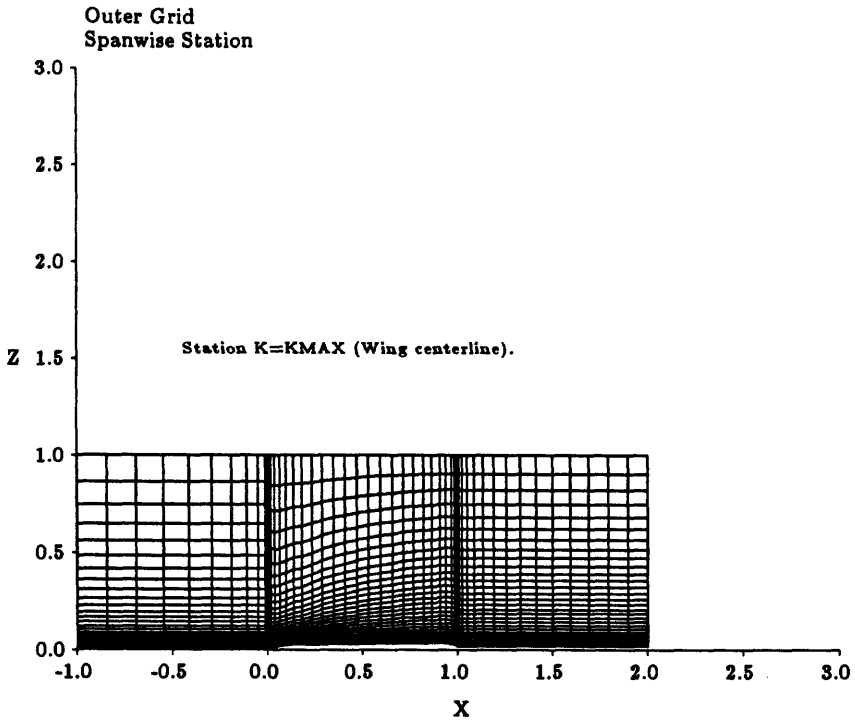

Figure 3.14: View of outer grid at a spanwise location (wing centerline).

and the wake runs between $x = 1$ and $x = 2$. It can be seen in this figure that an attempt was made to place more streamwise points near the apex and trailing edge regions, as more resolution is needed there. Careful inspection of the bottom portion of this figure in the wake region reveals where the inner grid fits inside this outer portion of the grid.

3.5.3 Combination Grid System

The combined grid system, as previously mentioned, is a composite of the inner and outer grids. Thus the number of computational cells in the grid system is equal to the sum of the two individual grids, giving a total of 276,000 cells used for calculation in the domain.

Figure 3.15 is an enlargement of the leading edge region of the combination grid at a streamwise station on the wing. It can be seen that the grid sizes at the interface seem to match adequately. The C^0 continuity of the grid lines across the interface is apparent, as is the lack of C^1 continuity.

Other views of the combination grid do not reveal much more than is shown in this

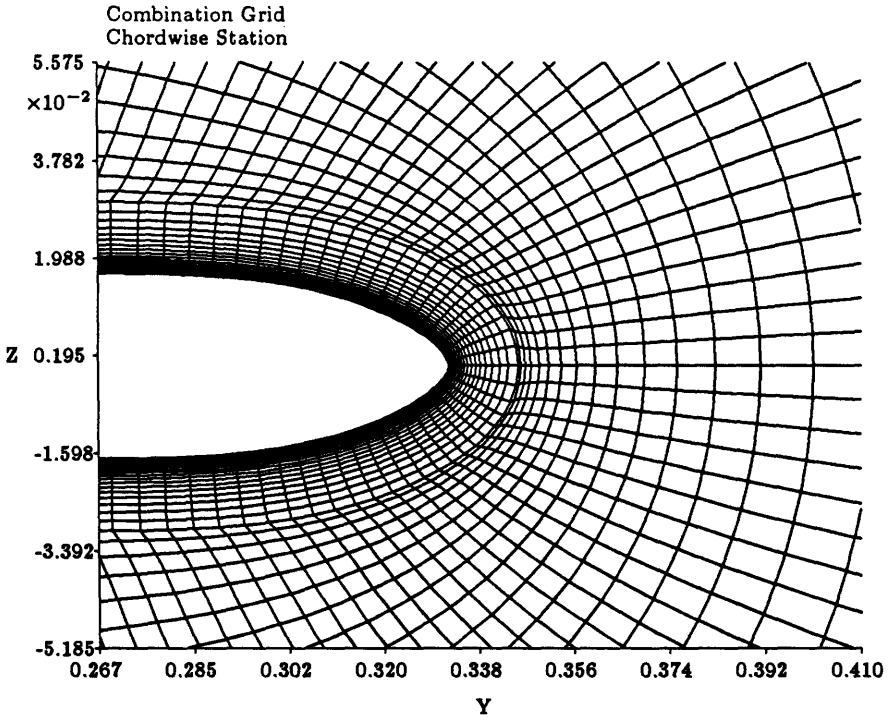

Figure 3.15: Blowup of combination grid at leading edge (71 percent chord).

view, and are thus omitted.

Chapter 4
Solution Algorithms

This chapter describes the numerical algorithms used in this thesis to solve the governing equations presented in Chapter 2. Both of the algorithms (for the Euler and TLNS equations) are of the cell-centered finite volume variety, the difference between them being in the temporal integration applied to march the solution forward in time to a steady state. The Euler solver used is that of Roberts [28] as modified by Goodsell [8]. The TLNS code used is the semi-implicit SINSS code of Loyd [17]. Each code was modified extensively to do the code coupling.

The chapter begins by describing items which are common to all finite-volume type algorithms (and are thus similar for both algorithms being dealt with here), such as the spatial discretization, artificial dissipation and convergence acceleration. This is followed by a description of the unique aspects of each of the solvers, such as the temporal integration and the boundary conditions. Finally, the treatment of the interface between the two grids and the coupling of the two flow solvers is described.

4.1 The Finite Volume Method

4.1.1 Spatial Discretization

In order to render Equation 2.1 solvable by a numerical method, it must first be discretized. The discretization consists of two different aspects; time discretization and space discretization. The first term of Equation 2.1, which is the time rate of change of the state vector in the control volume, is subject to discretization in time, while the second term

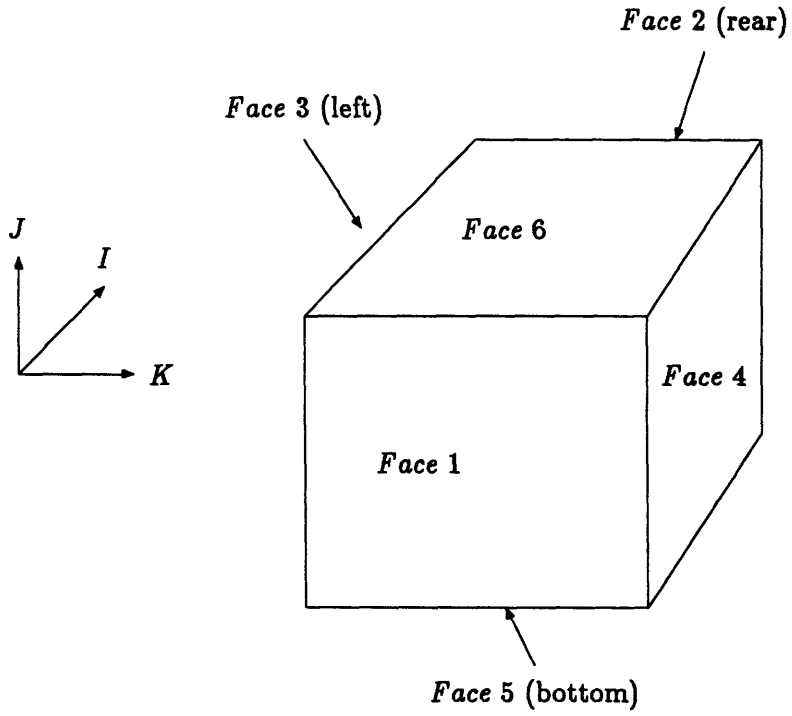

Figure 4.1: Hexahedral Cell and Cell Nomenclature.

of the equation, which represents the flux into or out of the control volume, is subject to discretization in space.

In the three dimensional finite volume method, the spatial discretization is done by dividing the domain of interest into a large number of hexahedral cells (i.e., "finite volumes"). A typical hexahedral cell is shown in Figure 4.1. In this figure, Faces 5 and 6 are in the body-normal direction (j), Faces 1 and 2 are in the streamwise direction (i) and Faces 3 and 4 are in the cross-stream direction (k).

The values of the state vector are assumed to be stored at the center of each of cell and the flux vector $\vec{\mathbf{F}}$ is calculated at each cell based on the state vector there. The flux vector is assumed to have a constant value over any cell face.

Applying this discretization leads to the semi-discrete form of the governing equation (2.1):

$$\begin{aligned} \frac{d}{dt}(\mathbf{U}_{i,j,k}V_{i,j,k}) &= -\sum_{f=1}^{6}\vec{\mathbf{F}}(\mathbf{U})_f \cdot \vec{S}_f \\ &= -\sum_{f=1}^{6}(\mathbf{F}S_x + \mathbf{G}S_y + \mathbf{H}S_z)_f \end{aligned} \tag{4.1}$$

Here, the flux integral over the surface of the cell has been converted into a discrete sum of fluxes through the faces of the cell. The flux through each face is calculated by multiplying the projected area of the face by the value of the flux vector in the corresponding direction. For example, the net flux through a cell in the x direction is found using

$$\sum_{f=1}^{6}(\mathbf{F}S_x)_f = (\mathbf{F}S_x)_1 + (\mathbf{F}S_x)_2 + (\mathbf{F}S_x)_3 + (\mathbf{F}S_x)_4 + (\mathbf{F}S_x)_5 + (\mathbf{F}S_x)_6 \ , \tag{4.2}$$

where S_x is the projected area of the cell face in the $y-z$ plane. Similarly, in the y and z directions the fluxes are obtained by replacing $(\mathbf{F}S_x)$ with $(\mathbf{G}S_y)$ and $(\mathbf{H}S_z)$, respectively.

The value of the inviscid flux at a cell face is computed by averaging the values of the flux vectors at adjacent cells which share that particular face. Using the component $\mathbf{G}$ as an example, the values of the flux on each of the six faces of a cell are given by

$$\begin{aligned} \mathbf{G}_1 &= .5(\mathbf{G}_{I_{i-1,j,k}} + \mathbf{G}_{I_{i,j,k}}) & \mathbf{G}_2 &= .5(\mathbf{G}_{I_{i+1,j,k}} + \mathbf{G}_{I_{i,j,k}}) \\ \mathbf{G}_3 &= .5(\mathbf{G}_{I_{i,j,k-1}} + \mathbf{G}_{I_{i,j,k}}) & \mathbf{G}_4 &= .5(\mathbf{G}_{I_{i,j,k+1}} + \mathbf{G}_{I_{i,j,k}}) \\ \mathbf{G}_5 &= .5(\mathbf{G}_{I_{i,j-1,k}} + \mathbf{G}_{I_{i,j,k}}) & \mathbf{G}_6 &= .5(\mathbf{G}_{I_{i,j+1,k}} + \mathbf{G}_{I_{i,j,k}}) \ . \end{aligned} \tag{4.3}$$

The values of the viscous terms, when they occur, are written in finite difference form using forward or backwards differences as

$$\frac{\partial(\ )}{\partial\eta} = \frac{(\ )_{j+1} - (\ )_j}{\Delta\eta} \quad \text{or} \quad \frac{\partial(\ )}{\partial\eta} = \frac{(\ )_j - (\ )_{j-1}}{\Delta\eta} \ , \tag{4.4}$$

depending whether the viscous flux across *Face* 5 or *Face* 6 is sought. $\Delta\eta$ is equal to 1 by definition. Notice that the viscous terms in the streamwise and cross-stream directions are not computed. This is to be consistent with the thin layer assumption that derivatives in these directions are small in comparison to the normal direction, and can thus be neglected.

4.1.2 Artificial Viscosity

Since the discretization described above uses central differencing, and central difference algorithms are dispersive rather than dissipative, it is necessary to add some form of artificial damping to the algorithm to stabilize the time integration. This is done following the approach of Jameson [11], which defines a dissipation operator **D** as a sum of non-linear second and fourth difference operators. In addition, the dissipation is weighted by "pressure switches" designed to regulate the relative magnitudes of the second and fourth order dissipations, depending on the pressure gradients in the flow. The actual equation which is solved is then Equation (4.1) with the dissipation operator **D** subtracted from the right hand side.

The purpose of the fourth-order dissipation is to provide a low level of dissipation throughout the domain to eliminate the odd-even oscillations which are characteristic of methods employing central differences. It also serves to stabilize the time integration of the equations, but may lead to de-stabilization of the scheme when there are discontinuities in the flow, such as shock waves. This is because the fourth order smoothing tends to increase the magnitude of the Gibbs phenomenon near discontinuities. In other words, it increases overshoots in physical quantities such as the pressure near the shocks.

Second-order dissipation is added to the scheme to make capturing of shocks and flow discontinuities possible. Without it, the fourth order dissipation is insufficient to stabilize the time integration in the presence of the discontinuity.

The dissipation operator **D** at each cell consists of contributions in each of the three computational coordinate directions:

$$\mathbf{D}(\mathbf{U}) = \mathbf{D}_\xi(\mathbf{U}) + \mathbf{D}_\eta(\mathbf{U}) + \mathbf{D}_\zeta(\mathbf{U}), \tag{4.5}$$

where $\mathbf{D}_\xi$, $\mathbf{D}_\eta$, and $\mathbf{D}_\zeta$ are differences across the cell in the ξ, η and ζ directions, respectively:

$$\begin{aligned} \mathbf{D}_\xi(\mathbf{U}) &= \mathbf{d}_{i+1/2,j,k} - \mathbf{d}_{i-1/2,j,k} \\ \mathbf{D}_\eta(\mathbf{U}) &= \mathbf{d}_{i,j+1/2,k} - \mathbf{d}_{i,j-1/2,k} \\ \mathbf{D}_\zeta(\mathbf{U}) &= \mathbf{d}_{i,j,k+1/2} - \mathbf{d}_{i,j,k-1/2}. \end{aligned} \tag{4.6}$$

The damping flux terms $\mathbf{d}_{i+1/2,j,k}, \mathbf{d}_{i-1/2,j,k}$, etc. are constructed such that $\mathbf{D}(\mathbf{U})$ is a sum of 2^{nd} and 4^{th} differences across the cell. Each $\mathbf{d}$ is the sum of a first and third difference, so that differencing of $\mathbf{d}$'s across the cells makes $\mathbf{D}$ a sum of second and fourth differences. For example, in the ξ computational direction,

$$\begin{aligned}\mathbf{d}_{i+1/2} = \left(\frac{V}{\Delta t}\right)_{i+1/2}\Big[\quad & \epsilon^{(2)}_{i+1/2}(\mathbf{U}_{i+1} - \mathbf{U}_i) - \\ & \epsilon^{(4)}_{i+1/2}(\mathbf{U}_{i+2} - 3\mathbf{U}_{i+1} + 3\mathbf{U}_i - \mathbf{U}_{i-1})\Big]\end{aligned} \tag{4.7}$$

and

$$\begin{aligned}\mathbf{d}_{i-1/2} = \left(\frac{V}{\Delta t}\right)_{i-1/2}\Big[\quad & \epsilon^{(2)}_{i-1/2}(\mathbf{U}_{i} - \mathbf{U}_{i-1}) - \\ & \epsilon^{(4)}_{i-1/2}(\mathbf{U}_{i+1} - 3\mathbf{U}_i + 3\mathbf{U}_{i-1} - \mathbf{U}_{i-2})\Big]\end{aligned} \tag{4.8}$$

where the indices j, k have been suppressed for clarity. Analogous differences are defined in the η and ς directions by replacing the index i with j or k, respectively. With this formulation and the factor of $V/\Delta t$, the artificial viscosity terms are conservative.

The coefficients $\epsilon^{(2)}$ and $\epsilon^{(4)}$ contain pressure "switches" designed to turn on the second order dissipation in regions near shocks and to turn it off in smooth regions of the flow. It does the opposite to the fourth order dissipation, turning it down near shocks and up in smooth regions of the flow. The coefficients are defined as:

$$\epsilon^{(2)}_{i+1/2} = \kappa^{(2)}(\nu_{\xi_{i+1}} + \nu_{\xi_i})/2 \,, \tag{4.9}$$

$$\epsilon^{(4)}_{i+1/2} = \max(0, \kappa^{(4)} - \epsilon^{(2)}_{i+1/2}) \,. \tag{4.10}$$

The factor ν is the actual pressure switch, calculated based on pressure gradients in the region:

$$\nu_{\xi_i} = \frac{|P_{i+1} - 2P_i + P_{i-1}|}{P_{i+1} + 2P_i + P_{i-1}} \,. \tag{4.11}$$

Some representative values of the damping coefficients $\kappa^{(2)}$ and $\kappa^{(4)}$, which are specified by the user, are 0.07 and 0.012, respectively.

Establishing adequate boundary conditions for the artificial viscosity is important for maintaining a well-posed numerical problem, however this is difficult because the artificial viscosity is a non-physical term added to stabilize the integration.

The approach taken here is due to Eriksson [5], which guarantees that the artificial viscosity terms are dissipative throughout the domain. To accomodate the five point difference stencil of the fourth order dissipation, two "ghost cells" are defined outside the boundary. This is illustrated in Figure 4.2. Values of the state vector at the ghost cells are found by linear extrapolation from the interior of the domain:

$$\begin{aligned} \mathbf{U}_0 &= 2\mathbf{U}_1 - \mathbf{U}_2 \\ \mathbf{U}_{-1} &= 3\mathbf{U}_1 - 2\mathbf{U}_2 \ , \end{aligned} \tag{4.12}$$

where the subscript $_0$ denotes the ghost cell adjacent to the boundary and $_{-1}$ denotes the next ghost cell.

The fourth order dissipation at the boundary is then calculated based on the last three cells interior to the boundary and the two ghost cells outside the boundary, while the second order is calculated based on the last two cells inside the domain and the first ghost cell.

This particular boundary treatment is applied at all boundaries, with the exception of the symmetry boundaries, wake boundary and the interface boundary. At the symmetry boundaries, a reflection condition is applied, as described in Section 4.2.2. At the wake boundary , a throughflow condition is applied as described in Section 4.3.3. The conditions at the interface between grids are described in Section 4.4.

4.1.3 Convergence Acceleration Techniques

It is often advantageous (and cheaper) to try to speed the convergence to steady state of the time integration. This can be accomplished through several techniques, such as multigrid, time inclining, residual smoothing, enthalpy damping and local time stepping. The ones chosen for this application are implicit residual smoothing, due to Jameson and Baker [12], enthalpy damping, due to Jameson [11], and local time stepping. Each of these is described briefly below.

Implicit residual smoothing functions by spreading out the change in the residuals over more of the domain. This effectively increases the size of the difference stencil which is

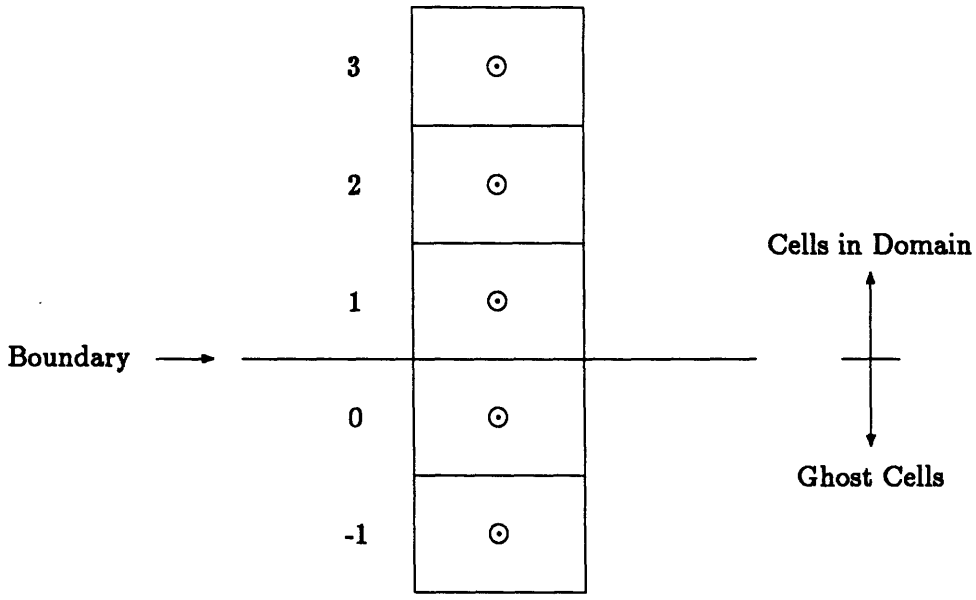

Figure 4.2: Schematic of Ghost Cells at a Boundary.

used , and allows a greater CFL number, and thus a greater time step, to be used, speeding convergence to the steady state.

In three dimensions, residual smoothing is implemented by applying a smoothing operator succesively in each of the three computational coordinate directions ξ, η, ς as follows:

$$\begin{aligned} (1 - \mu_R \delta_{\xi\xi})\mathbf{R}' &= \mathbf{R} \\ (1 - \mu_R \delta_{\eta\eta})\mathbf{R}'' &= \mathbf{R}' \\ (1 - \mu_R \delta_{\varsigma\varsigma})\mathbf{R}''' &= \mathbf{R}'' , \end{aligned} \tag{4.13}$$

The smoothing operator consists of the second difference operator δ applied to the vector of residuals $\mathbf{R}$ and a "residual smoothing coefficient" , μ_R, which weights the second difference operator. Note that setting μ_R to zero leaves the residuals unmodified after application of the smoothing operator. The operator δ is given in the ξ direction by $\delta_{\xi\xi}\mathbf{R} = \mathbf{R}_{i+1} - 2\mathbf{R}_i + \mathbf{R}_{i-1}$ and is thus an *undivided* second difference operator.

In the computation, the vector of residuals $\mathbf{R}$ is smoothed using the above sequence of smoothing operators and the "smoothed" residuals $\mathbf{R}'''$ are then placed back into the

computation, which proceeds as normal. The optimum value of μ_R was found for each case by numerical experimentation. Complete details of the implementation of the residual smoothing and solution of the resulting matrix systems may be found in Appendix C of Loyd [17].

In the TLNS solver, residual smoothing in a given direction is superfluous if the discrete fluxes are already treated implicitly as in a semi-implicit or fully implicit scheme. In a semi-implicit scheme, it is still profitably applied in the explicit directions.

Enthalpy damping is a convergence acceleration technique mostly applicable to subsonic inviscid flow. It has its greatest effect at very low Mach numbers and in supersonic flows it is de-stabilizing, however Roberts [28] and others have found that it is not necessary to turn the enthalpy damping off in transonic flows. It is implemented by solving a modified version of the Euler equations in which a term proportional to the difference between the local enthalpy and the free-stream enthalpy is added :

$$\frac{\partial}{\partial t}\iiint_V \mathbf{U}\,dV + \iint_{\partial V} \vec{\mathbf{F}}(\mathbf{U})\cdot\vec{\mathbf{n}}dS \;+\; \alpha_E(H - H_\infty)\overline{\mathbf{U}} = 0. \tag{4.14}$$

Here, α_E is an enthalpy damping coefficient and $\overline{\mathbf{U}}$ is a modification of the normal state vector $\mathbf{U}$ given by $\overline{\mathbf{U}} = (\rho\ \rho u\ \rho v\ \rho w\ \rho)^T$. Modification of the state vector by replacing the last element ρH with ρ is necessary to ensure stability (Turkel, [31]).

Enthalpy damping, as the name implies, speeds convergence by damping out flow transients. It does not, however, increase the time step which may be taken. At steady state, an inviscid flow should have constant enthalpy, so that the added term in the equation goes to zero and the original equation is recovered. If a time accurate calculation is of interest, enthalpy damping cannot be used because the addition of the enthalpy term to the Euler equations introduces non-physical transients in the solution.

The enthalpy damping is applied in the code by adding the damping terms to the state vector at the the end of each iteration. A good value of α_E was found by Roberts [28] to be 0.025.

Local time stepping is applied by allowing the size of the time step Δt to vary from cell to cell. At each cell, the time step is allowed to be the highest value which meets the

CFL conditions and produces a convergent iteration. Thus, since each cell is advanced by a different time step, each iteration is described more aptly as a step forward in "pseudo-time". Time accurate calculations may be run by setting the time step in all cells equal to the minimum value of the entire domain, however this usually results in very slow convergence.

In the TLNS solver, Loyd [17] found that convergence in semi-implicit mode could not be acheived for separated flow cases using local time stepping, possibly due to some non-linear numerical instability. His approach was to set all the time steps in a given j,k plane equal to the minimum in that plane. Lee [15] found that convergence could still be acheived, and at a greater rate, if for each row of cells at a given j the time step was set to the minimum in that row. For the Euler solver, there were no problems in using local time steps.

4.1.4 Convergence Criterion

As with any iterative scheme, it is necessary to establish some criterion to determine when the solution has converged to its steady state solution, if such a solution exists. In this case we take the mathematical approach and assume that the steady state has been reached when the RMS change in the state vector from one time step to another , labeled the "error", drops by a given number of orders of magnitude. The RMS change in the state vector is defined by

$$\Delta \mathbf{U} = \sum_{N_{cells}} \left[(\Delta\rho)^2 + (\frac{\Delta\rho u}{u_\infty})^2 + (\frac{\Delta\rho v}{u_\infty})^2 + (\frac{\Delta\rho w}{u_\infty})^2 + (\frac{\Delta\rho E}{E_\infty})^2 \right] \tag{4.15}$$

and

$$\Delta \mathbf{U}_{RMS} = \frac{\sqrt{\Delta \mathbf{U}}}{5 N_{cells} \lambda} \tag{4.16}$$

where λ is the CFL number and N_{cells} is the number of cells in the particular domain. Note that the convergence criterion is the same for both the Euler and Navier-Stokes domains. The error $\Delta \mathbf{U}_{RMS}$ is computed separately in each domain and the error value must drop by a certain amount in both domains in order for convergence to be considered acheived. The exact number of orders of magnitude that the error must drop for convergence varies from case to case, at the discretion of the researcher.

4.2 Euler Solution Procedure

Having applied the finite-volume discretization to Equation 2.1 and arrived at the semi-discrete equation(4.1), it is now necessary to find a suitable time discretization and integration to march the solution of the equation forward in time to the steady state solution which is sought. This is done following the approach of Jameson [11].

4.2.1 Explicit Time Integration

The method chosen to integrate the semi-discrete equation forward in time for the Euler solver is the well-known, tried-and-true explicit 4-stage Runge-Kutta integration. Beginning with an initial guess $\mathbf{U}^n$, where n denotes the pseudo-time level, the value of the state vector at each cell at the next time level, $\mathbf{U}^{n+1}$, is calulated using

$$\begin{aligned}
\mathbf{U}^0 &= \mathbf{U}^n \\
\mathbf{U}^1 &= \mathbf{U}^0 - \alpha_1 \tfrac{\Delta t}{V} \left[\textstyle\sum_{f=1}^{6} \left(\mathbf{F}^0 S_x + \mathbf{G}^0 S_y + \mathbf{H}^0 S_z \right)_f - \mathbf{D}^0 \right] \\
\mathbf{U}^2 &= \mathbf{U}^0 - \alpha_2 \tfrac{\Delta t}{V} \left[\textstyle\sum_{f=1}^{6} \left(\mathbf{F}^1 S_x + \mathbf{G}^1 S_y + \mathbf{H}^1 S_z \right)_f - \mathbf{D}^0 \right] \\
\mathbf{U}^3 &= \mathbf{U}^0 - \alpha_3 \tfrac{\Delta t}{V} \left[\textstyle\sum_{f=1}^{6} \left(\mathbf{F}^2 S_x + \mathbf{G}^2 S_y + \mathbf{H}^2 S_z \right)_f - \mathbf{D}^0 \right] \\
\mathbf{U}^4 &= \mathbf{U}^0 - \alpha_4 \tfrac{\Delta t}{V} \left[\textstyle\sum_{f=1}^{6} \left(\mathbf{F}^3 S_x + \mathbf{G}^3 S_y + \mathbf{H}^3 S_z \right)_f - \mathbf{D}^0 \right] \\
\mathbf{U}^{n+1} &= \mathbf{U}^4 \, ,
\end{aligned} \tag{4.17}$$

where the pseudo-time level is identified by superscripts and the multistage coefficients α are given by

$$\alpha_1 = \frac{1}{4}, \quad \alpha_2 = \frac{1}{3}, \quad \alpha_3 = \frac{1}{2}, \quad \alpha_4 = 1 \, . \tag{4.18}$$

The terms $\mathbf{F}$, $\mathbf{G}$, and $\mathbf{H}$ are flux vectors in the x, y, and z directions, respectively. One iteration consists of an entire pass through the above sequence of time stepping stages. The superscript ^{0} on the dissipation operator indicates that the dissipation is only calculated at the beginning of each iteration and not at each stage of the integration scheme. The purpose of this is simply to reduce computational expense, as Jameson [11] has found it unnecessary to compute the dissipation after each stage.

As with most numerical schemes, the Runge-Kutta integration has limits on its numerical stability. By performing a Von Neumann stability analysis on the above scheme applied to the model linear equation for this problem (the wave equation), the scheme is found to be stable provided certain conditions on the time step Δt are met. This condition is known as the CFL condition (standing for Courant-Friedrichs-Lewy, who first discovered it) and is given by

$$\Delta t \cong \lambda \frac{V}{|\mathbf{W} \cdot \mathbf{S}|_{max} + C|\mathbf{S}_{max}|} \tag{4.19}$$

where C is the local speed of sound, V is volume of the cell, and $\mathbf{W}$ and $\mathbf{S}$ are vector velocity and projected surface area, $\mathbf{W} = u\vec{i} + v\vec{j} + w\vec{k}$ and $\mathbf{S} = S_x\vec{i} + S_y\vec{j} + S_z\vec{k}$, respectively. The stability analysis tells us that the value of λ , known as the " CFL Number ", is restricted by $\lambda \leq 2\sqrt{2}$.

4.2.2 Boundary Conditions

As with the governing PDE, boundary conditions must be applied to the numerical scheme to have a well posed problem. For the Euler solver as used in this study, the boundaries may be arranged into three groups at which the boundary treatments are the same. These are:

1. **Inflow, Outflow and Outer Boundaries.** These are the "far field" boundaries of the domain, located at $i = 1$, $i = i_{max}$ and $j = j_{max}$, respectively. A special treatment is applied at these boundaries to approximate Equation 2.5.

2. **Symmetry Boundaries.** These boundaries are present because only symmetrical flow cases are of interest for this study, and computation takes place only over half of the wing. The boundaries are located at $k = 1$ and $k = k_{max}$.

3. **Interface Boundary.** This is the boundary at which the Euler solver is matched to the thin layer Navier-Stokes solver. It is located at $j = 1$ between the wing apex and the downstream boundary ($i = i_{le}$ to $i = i_{max}$).

The first two of the above categories are describe in this section. The third is described in the last section of the chapter on code coupling.

Inflow, Outflow and Outer Boundaries

Application of the far-field boundary condition , Equation 2.5 is difficult in the case of a numerical computation because the domain boundaries are not located at infinity. Usually the boundaries are placed at a distance from the body at which the effect of the body on the flow is small. Often, this is a judgement call on the part of the particular researcher. Numerically, the inflow, outflow and outer boundaries are handled by assuming that the flow is locally one dimensional in the direction normal to the boundaries. The theory of characteristics is applied, giving a series of boundary conditions which allow disturbances to go through the boundaries with minimum reflection. This treatment is due to Jameson and Baker [12] and involves the introduction of Riemann invariants at the boundaries.

For flows with subsonic free stream velocities as are being dealt with here, the theory of characteristics tells us that, for a subsonic inflow, there are four characteristics entering the domain and one leaving the domain. Thus, four characteristic variables are specified and one is extrapolated from the interior. At a subsonic outflow, there is one incoming characteristic and four outgoing ones, so that four values are extrapolated from the interior and one is specified.

The incoming and outgoing Riemann invariants are given respectively by

$$R_- = \vec{u}_\infty \cdot \hat{n} - 2\frac{a_\infty}{\gamma - 1} = u_n - 2\frac{a}{\gamma - 1} \tag{4.20}$$

and

$$R_+ = \vec{u}_{ex} \cdot \hat{n} + 2\frac{a_{ex}}{\gamma - 1} = u_n + 2\frac{a}{\gamma - 1} \tag{4.21}$$

where subscripts ∞ and ex denotes freestream values and extrapolated values , respectively, and $\hat{n}$ is an outward pointing unit vector. These characteristics may be combined to give the velocity normal to the boundary and the speed of sound as

$$u_n = \frac{1}{2}(R_+ + R_-) \tag{4.22}$$

and

$$a = \frac{\gamma - 1}{4}(R_+ - R_-). \tag{4.23}$$

At an inflow boundary, the entropy is specified as the free stream value $s_\infty = 1/\gamma$. The tangential components of velocity are also specified from the free stream and the total velocity is given by

$$\vec{u} = \vec{u}_\infty + (u_n - \vec{u}_\infty \cdot \hat{n})\hat{n}. \tag{4.24}$$

On an outflow boundary, the entropy is extrapolated to the boundary using

$$s = s_{ex} = \frac{p_{ex}}{\rho_{ex}^\gamma}. \tag{4.25}$$

The tangential components of velocity are also extrapolated and the total velocity is given in the following manner:

$$\vec{u} = \vec{u}_{ex} + (u_n - \vec{u}_{ex} \cdot \hat{n})\hat{n}. \tag{4.26}$$

Symmetry Boundaries

If the wing is symmetric and is not at an angle of yaw then the flow need only be solved on half the wing and symmetry conditions can be assumed. These are applied by defining ghost points on the symmetry plane and reflecting all the conservation variables from the interior cells adjacent to the boundary. For example, the state vectors $\mathbf{U}$ at ghost points $(i,j,0)$ and $(i,j,kmax+1)$ are given by

$$\mathbf{U}_{(0,k_{max}+1)} = \begin{pmatrix} \rho \\ \rho u \\ -\rho v \\ \rho w \\ \rho E \end{pmatrix}_{(1,k_{max})} . \tag{4.27}$$

Likewise, the state vector at the second ghost points away from the boundary, $(i,j,-1)$ and $(i,j,kmax+2)$, are defined by applying the above, using the state vector at the real cells $(i,j,2)$ and $(i,j,kmax-1)$, respectively. While only the first ghost cell value is necessary for the actual computation, the additional ghost cells at this boundary must be set so that the artificial dissipation can be calculated.

4.3 Thin-Layer Navier-Stokes Solver

In the case of the TLNS solver, it was coded such that it could run in one of two modes; *Explicit* and *Semi-Implicit.*

4.3.1 Explicit Time Integration

In explicit mode, the time integration of the TLNS equations is carried out in exactly the same manner as the integration of the Euler equations described in Section 4.2.1. The only difference is that the restriction on the time step is modified by a viscous contribution to the denominator

$$\Delta t \cong \lambda \frac{V}{|\mathbf{W} \cdot \mathbf{S}|_{max} + C|\mathbf{S}_{max}| + \frac{4\mu}{\Delta l_n} S_n}. \tag{4.28}$$

In this equation, the quantity Δl_n is the cell dimension in the direction normal to the body and S_n is the projected area of the cell in that direction. See appendix A of Loyd [17] for a detailed derivation of this relation.

Examination of this equation reveals the problem with using explicit schemes for viscous flows. In these flows, it is necessary to have many points concentrated in the area near the body, resulting in small cell sizes there. The above restriction on the time step tells us that, correspondingly, the size of the allowed time step in these regions will be very small, making convergence extremely slow. The advantage of low cost per iteration for the explicit scheme is thus overridden because so many more iterations are necessary.

4.3.2 Semi-Implicit Time Integration

The numerical stiffness inherent in Navier-Stokes calculations due to the small spacing in the body-normal direction causes serious difficulties with explicit schemes. This difficulty may be eliminated by using an implicit scheme, however these are computationally very expensive due to the large matrix inversions which must be done. A compromise between the low cost per iteration of the explicit scheme and the large time step possible with

implicit schemes is acheived by treating the problem *semi-implicitly*. The integration in the normal direction is implicit while the integration in direction tangential to the body remain explicit. Referring to Figure 4.1, consider the first stage of (4.17) with implicit treatment of the normal component of the flux vectors

$$\begin{aligned}\mathbf{U}^1 - \mathbf{U}^0 = -\alpha_1 \frac{\Delta t}{V} [\quad & (\mathbf{F}^0 S_x + \mathbf{G}^0 S_y + \mathbf{H}^0 S_z)_1 + (\mathbf{F}^0 S_x + \mathbf{G}^0 S_y + \mathbf{H}^0 S_z)_2 + \\ & (\mathbf{F}^0 S_x + \mathbf{G}^0 S_y + \mathbf{H}^0 S_z)_3 + (\mathbf{F}^0 S_x + \mathbf{G}^0 S_y + \mathbf{H}^0 S_z)_4 + \\ & (\mathbf{F}^1 S_x + \mathbf{G}^1 S_y + \mathbf{H}^1 S_z)_5 + (\mathbf{F}^1 S_x + \mathbf{G}^1 S_y + \mathbf{H}^1 S_z)_6 - \mathbf{D}^0)] \ .\end{aligned} \tag{4.29}$$

The semi-implicit treatment is evident in these equations, as the fluxes through the stream-wise and cross-stream faces 1, 2 and 3, 4 respectively are evaluated at the time step before the fluxes through the body-normal faces, 5 and 6. Numerical experimentation by Loyd [17] shows that it is not necessary to treat the body normal artificial viscosity terms implicitly.

Since the flux vectors $\mathbf{F}, \mathbf{G}$ and $\mathbf{H}$ are nonlinear with respect to the state vector $\mathbf{U}$, the fluxes at *Faces* 5 and 6 must be linearized in time to allow calculation of $\mathbf{U}^1$. Writing the Taylor series in time for $\mathbf{F}$, $\mathbf{G}$, and $\mathbf{H}$ gives

$$\begin{aligned}\mathbf{F}^1 = \ & \mathbf{F}^0 + \frac{\partial \mathbf{F}^0}{\partial t}\Delta t + O(\Delta t^2) = \ \mathbf{F}^0 + [A]^0 \Delta \mathbf{U}^1 + O(\Delta t^2) \\ \mathbf{G}^1 = \ & \mathbf{G}^0 + \frac{\partial \mathbf{G}^0}{\partial t}\Delta t + O(\Delta t^2) = \ \mathbf{G}^0 + [B]^0 \Delta \mathbf{U}^1 + O(\Delta t^2) \\ \mathbf{H}^1 = \ & \mathbf{H}^0 + \frac{\partial \mathbf{H}^0}{\partial t}\Delta t + O(\Delta t^2) = \ \mathbf{H}^0 + [C]^0 \Delta \mathbf{U}^1 + O(\Delta t^2)\end{aligned} \tag{4.30}$$

where $[A]$, $[B]$, and $[C]$ are the 5×5 Jacobian matrices $[\partial \mathbf{F}/\partial \mathbf{U}]$, $[\partial \mathbf{G}/\partial \mathbf{U}]$, and $[\partial \mathbf{H}/\partial \mathbf{U}]$, respectively, and the vector of changes $\Delta \mathbf{U}$ is defined as $\Delta \mathbf{U}^1 \equiv \mathbf{U}^1 - \mathbf{U}^0$.

Inserting the linearization into (4.29) gives

$$\begin{aligned}\Delta \mathbf{U} \ = -\alpha_1 \tfrac{\Delta t}{V} \Big\{ & \left(\textstyle\sum_{f=1}^{6} (\mathbf{F} S_x + \mathbf{G} S_y + \mathbf{H} S_z)_f - \mathbf{D} \right)^0 \\ & + [[A]_5 S_{x_5} + [A]_6 S_{x_6} + [B]_5 S_{y_5} + [B]_6 S_{y_6} + [C]_5 S_{z_5} + [C]_6 S_{z_6}] \, \Delta \mathbf{U} \Big\} \ .\end{aligned} \tag{4.31}$$

Re-arranging this a bit leads to

$$\begin{aligned}\Big[[I] + \alpha_1 \tfrac{\Delta t}{V} ([A]_5 S_{x_5} + [A]_6 S_{x_6} + [B]_5 S_{y_5} + [B]_6 S_{y_6} + [C]_5 S_{z_5} + [C]_6 S_{z_6}) \Big] \Delta \mathbf{U} = \\ -\alpha_1 \tfrac{\Delta t}{V} \left[\textstyle\sum_{f=1}^{6} (\mathbf{F} S_x + \mathbf{F} S_y + \mathbf{H} S_z)_f - \mathbf{D} \right]^0 ,\end{aligned} \tag{4.32}$$

where the quantity in the square brackets [ ] prefacing $\Delta\mathbf{U}$ is a matrix which operates on the $5 \times J_{max}$ vector of changes $\Delta\mathbf{U}$ at a given (i,k). Thus, (4.32) is an implicit matrix equation for the changes in the state vector $\mathbf{U}$ at a line of cells (i,k). The right-hand side is the usual semi-discrete form of the residual, while the left-hand side differs from the identity matrix $[I]$ due to introduction of the terms from the time linearization of the fluxes.

The remaining steps in the integration scheme are computed in the same fashion as

$$\begin{aligned}
\mathbf{U}^0 &= \mathbf{U}^n \\
[LHS]^0 \Delta\mathbf{U}^1 &= \mathbf{R}^0 \\
[LHS]^1 \Delta\mathbf{U}^2 &= \mathbf{R}^1 - \Delta\mathbf{U}^1 \\
[LHS]^2 \Delta\mathbf{U}^3 &= \mathbf{R}^2 - (\Delta\mathbf{U}^2 + \Delta\mathbf{U}^1) \\
[LHS]^3 \Delta\mathbf{U}^4 &= \mathbf{R}^3 - (\Delta\mathbf{U}^3 + \Delta\mathbf{U}^2 + \Delta\mathbf{U}^1) \\
\mathbf{U}^{n+1} &= \mathbf{U}^3 + \Delta\mathbf{U}^4
\end{aligned} \tag{4.33}$$

where R is given by

$$\mathbf{R}^n = -\alpha_n \frac{\Delta t}{V} \left[\sum_{f=1}^{6} (\mathbf{F}S_x + \mathbf{G}S_y + \mathbf{H}S_z)_f^n - \mathbf{D}^0 \right] , \tag{4.34}$$

and the matrix $[LHS]$ is given by

$$[LHS]^n = [I] + \alpha \frac{\Delta t}{V} \left([A]_5 S_{x_5} + [A]_6 S_{x_6} + [B]_5 S_{y_5} + [B]_6 S_{y_6} + [C]_5 S_{z_5} + [C]_6 S_{z_6} \right)^n . \tag{4.35}$$

The change in the state vector $\Delta\mathbf{U}$ is defined by

$$\Delta\mathbf{U}^S = \mathbf{U}^S - \mathbf{U}^{S-1} . \tag{4.36}$$

Note that the explicit scheme is recovered when $[LHS] = [I]$. The *LHS* only contains dependent variables at $(j-1, j, j+1)$ due to the compact three point stencil used in the spatial discretization of the viscous and inviscid terms. Thus, (4.32) is a block tridiagonal system of equations, which can be efficiently inverted. Details of the setup of the matrices and solution of the matrix system are discussed in Appendix B of Loyd [17].

The time step in the semi-implicit integration is limited by the tangential and cross

stream flux terms, due to their explicit treatment:

$$\Delta t \leq 2\sqrt{2}\frac{Vol}{(|\mathbf{V}\cdot\mathbf{S}|)_{max_{1,2}} + (|\mathbf{V}\cdot\mathbf{S}|)_{max_{3,4}} + C|\mathbf{S}_{max}|_{1,2,3,4}} \ . \tag{4.37}$$

As with the explicit scheme, the limitation on the CFL numer λ may be taken as $\lambda \leq 2\sqrt{2}$. Details of the derivation of this time step restriction are given in Appendix A of Loyd [17].

4.3.3 Boundary Conditions

As with the Euler code, it is necessary to provide boundary conditions for the TLNS code. The boundaries of the TLNS domain may be grouped into five categories at which the boundary treatment is the same:

1. **Upstream and Outer Boundaries.** These are the boundaries of the TLNS grid at which coupling with the Euler solver takes place, located at $i = 1$ and $j = j_{max}$, respectively. The conditions applied are described in the section on code coupling.

2. **Wing Surface.** This boundary is at $j = 1$ between the apex and the trailing edge of the wing (i_{le} to i_{te}). The condition of no slip at a solid boundary is applied here.

3. **Wake Boundary.** This boundary is at $j = 1$ between the trailing edge and the downstream boundary (i_{te} to i_{max}). Here, the inner boundary of the grid collapses to a "slit", and a special treatment must be applied so that flow may pass through it undisturbed.

4. **Downstream Boundary.** This boundary is at $i = i_{max}$. Values of the state vector must be specified here, although the procedure is different than in the Euler solver.

5. **Symmetry Boundaries.** These boundaries are located at $k = 1$ and $k = k_{max}$. They are treated exactly as in the Euler solver.

Descriptions of the boundary conditions at the wing surface, wake surface and downstream boundary follow. The symmetry boundaries are described in Section 4.2.2 and the interface treatment is described in Section 4.4.

Outflow Boundary

Since the characteristic treatment using Riemann invariants is not applicable to viscous flows due to their elliptic character, the boundary condition at the outflow is handled simply by extrapolating from the last streamwise cell in the domain to the outer boundary. This is written mathematically as

$$\mathbf{U}_{imax} = 2\mathbf{U}_{imax-1} - \mathbf{U}_{imax-2}. \tag{4.38}$$

Wing Boundary

On the wing surface, the no-slip condition associated with viscous flows at solid boundaries is applied. This is accomplished numerically by setting the values of the velocities at the first ghost cell inside the body equal to the negative of the velocity in the first real cell, so that the average of the velocities at the face between the cells, which is the body surface, will be zero. Mathematically, this is written as

$$\mathbf{U}_{i,j=0,k} = \begin{pmatrix} \rho \\ -\rho u \\ -\rho v \\ -\rho w \\ \rho E \end{pmatrix}_{(i,j=1,k)} . \tag{4.39}$$

Wake Boundary

On the wake boundary, the ghost cells are reflected across the wake so that there is essentially flow through the boundary as if it wasn't even there. This condition is implemented by setting the value of the first ghost cell at $j = 0$ equal to the first real cell in the domain directly on the other side of the wake. Likewise, the value at the second ghost cell from the boundary is computed using values from the corresponding cell on the other side of the wake. Figure 4.3 illustrates the procedure. This condition is expressed mathematically as

$$\mathbf{U}_{(i,j=0,k)} = \mathbf{U}_{(i,j=1,kmax-k)}$$

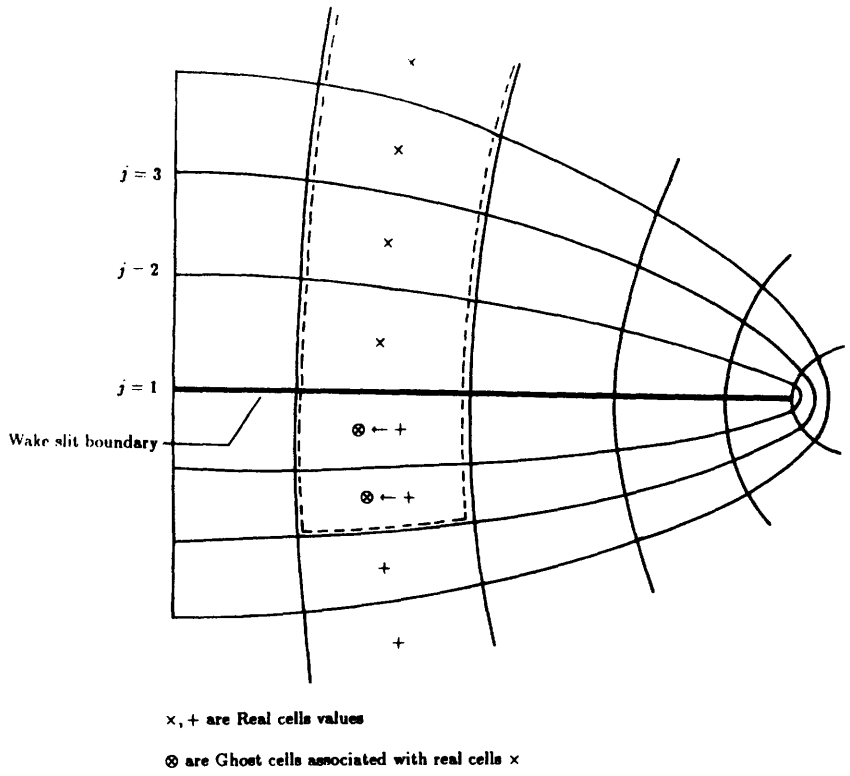

Figure 4.3: Treatment of Wake Boundary Condition.

$$\mathbf{U}_{(i,j=-1,k)} = \mathbf{U}_{(i,j=2,kmax-k)} \tag{4.40}$$

The flux and artificial viscosity across the wake boundary may then be calculated in the usual way.

4.4 Code Coupling

The main objective of the treatment of the interface between the Euler and TLNS solvers is to ensure that the interface is transparent to the flow, i.e., that the flow may pass through the interface undisturbed.

The coupling of the two codes is accomplished by matching values of the state vectors in the two domains at the interface. The two Euler ghost cells at the interface (which is is the $j = 1$ Euler boundary) are set equal to the two outermost real cells in the TLNS domain. Likewise, the two TLNS ghost cells at the interface (the $j = jmax$ TLNS boundary) are set equal to the first two real cells in the Euler domain. This is shown in Figure 4.4.

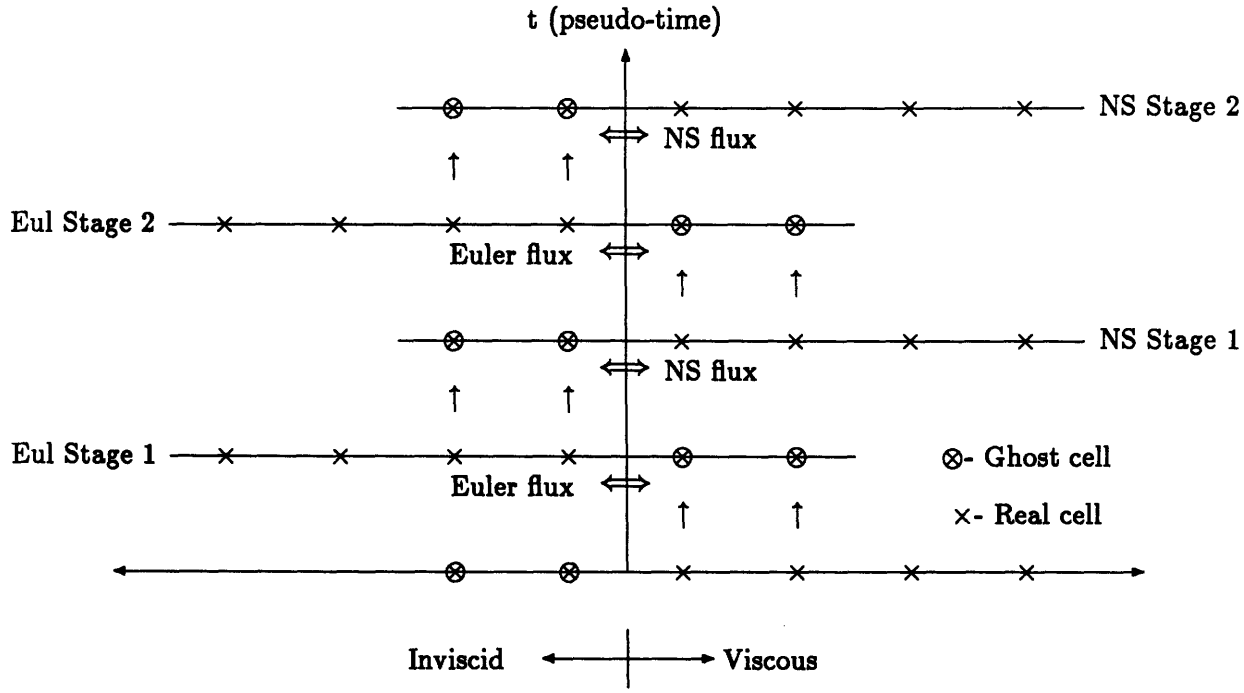

Figure 4.4: State vector matching at the interface between solvers.

Mathematically, this condition is written

$$\begin{aligned} [\mathbf{U}_{i,0,k}]_{Euler} &= [\mathbf{U}_{i,jmax,k}]_{NS} \\ [\mathbf{U}_{i,-1,k}]_{Euler} &= [\mathbf{U}_{i,jmax-1,k}]_{NS} \end{aligned} \tag{4.41}$$

and

$$\begin{aligned} [\mathbf{U}_{i,jmax+1,k}]_{NS} &= [\mathbf{U}_{i,1,k}]_{Euler} \\ [\mathbf{U}_{i,jmax+2,k}]_{NS} &= [\mathbf{U}_{i,2,k}]_{Euler} . \end{aligned} \tag{4.42}$$

Once the ghost cell values are set, the flux across the interface is calculated individually in each solver using its respective ghost cell values. Only the first ghost cell used for the flux calculation . The second ghost cell value for each domain is set so that the artificial viscosity, with its 5 point difference stencil, may be calculated at the interface. This treatment makes the interface transparent to the artificial viscosity.

To obtain a better coupling, the state vector matching takes place at each stage of the four-stage time integrations. This is illustrated for the first two stages in Figure 4.4. Remaining stages are computed similarly. To ensure the implicit treatment of the boundary for the semi-implicit solver, the ghost cell setting, flux computation and state vector updating had to be performed in a given sequence at each time step stage. This sequence is

1. Set Euler ghost cell values (Equation 4.41).
2. Calculate Euler fluxes.
3. Update Euler state vector.
4. Set N-S ghost cell values (Equation 4.42).
5. Calculate N-S fluxes.
6. Update Euler Navier-Stokes state vectors.
7. Go to step #1 for next time-step stage.

Using this particular sequence, the flux across the interface is not conservative until steady state is reached. An alternate sequence which made the flux conservative was attempted, but did not seem to make much of a difference to the functioning of the hybrid code.

Chapter 5

Results

This chapter presents the results obtained using the hybrid computational code on the NTF delta wing. Some preliminary results which were obtained on coarse grids are presented first, as 3-D color graphics were available for these cases. The 3-D graphics reveal some features of the flow which would not be evident in 2-D results such as contour plots. For the large final cases, the color graphics software was unable to handle grids and flowfields of that size, so no color graphics of the final results are shown. Instead, more standard black and white graphics, such as countour plots and line plots, are presented.

Table 5.1: Summary of NTF flow cases run.

Case #	M	α	Re_c	No. of cells	TLNS Mode
1	0.75	16°	1×10^5	11,400	Semi-Implicit
2	0.75	16°	1×10^5	36,000	Explicit
3	0.85	15°	2×10^6	276,000	Explicit
4	0.85	15°	7.5×10^6	276,000	Explicit

All cases run are summarized in Table 5.1. It is seen in this table that only one semi-implicit solution was obtained on a very coarse grid. This is because the semi-implicit mode of the TLNS solver turned out not to be robust. To obtain convergence, it was necessary to find the exact combination of smoothing coefficients and other parameters . This meant a lot of trial and error. Unfortunately, for the fine grid computations, the correct combination was never found.

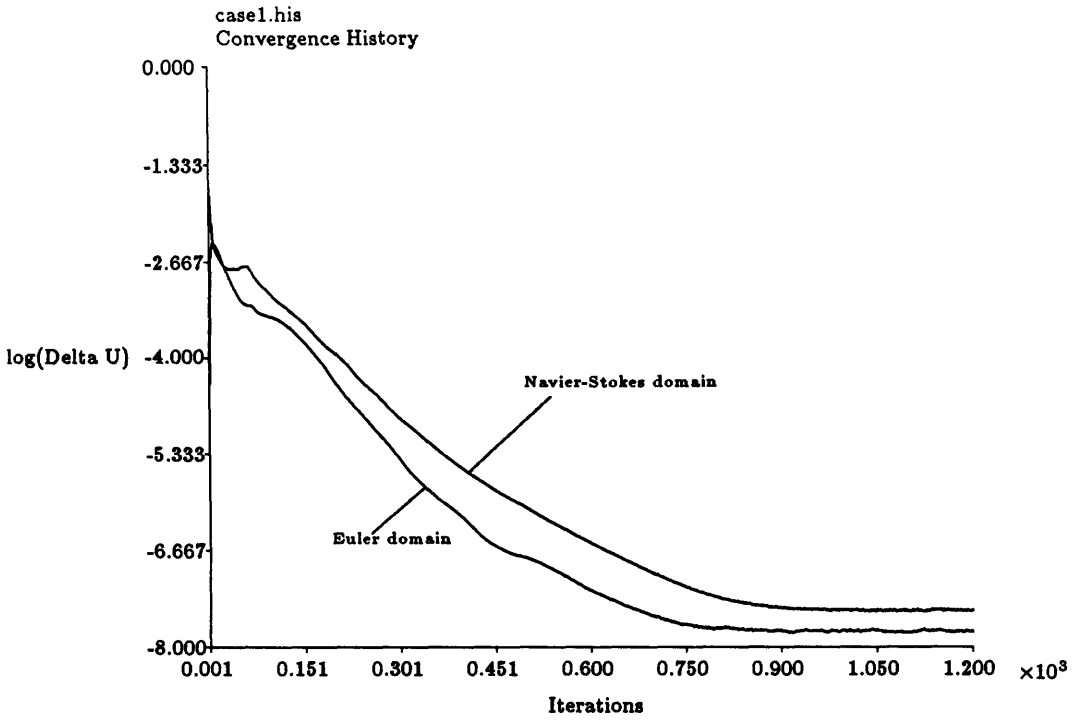

Figure 5.1: Convergence history for case 1.

5.1 Preliminary Flow Results

Since they were run on coarse grids that did not allow capture of all the relevant phenomena, cases 1 and 2 from Table 5.1 are presented as preliminary results.

5.1.1 Case 1 : Semi-Implicit TLNS – $M = 0.75$, $\alpha = 16°$, $Re_c = 1 \times 10^5$

The grid in this case contained 11,400 cells. The dimensions of the outer grid were $15 \times 20 \times 30$ in the i, j and k directions, respectively. Dimensions of the inner grid were $10 \times 8 \times 30$.

Results of these computations are shown in Figures 5.1 through 5.5. Figure 5.1 shows the convergence characteristics of the hybrid code with the TLNS solver running in semi-implicit mode. The error in both the Euler and Navier-Stokes domains drops about 5 orders of magnitude to machine zero in about 1200 iterations, with the error always being higher in the Navier-Stokes domain.

The distribution of total pressure in the domain is shown in Figure 5.2 The figure is divided into two regions; the left side shows color contours of total pressure (and thus static pressure, since it is a viscous flow) on the top wing surface. Also shown is a vortex cross section at a chordwise station near the trailing edge of the wing. This is generated using a thresholding algorithm where only values in a certain range are plotted and values falling outside the range are discarded. The surface which extends downstream of the trailing edge is the wake boundary (or wake "slit") of the grid. This is drawn by the graphics package, but does not represent a physical surface.

The right side of Figure 5.2 shows in detail the cross section of the vortex from the left part of the figure. The white line which wraps around the wing represents the interface between the two solvers, across which the solution is seen to be smooth and continuous. The rollup of the primary vortex from the leading edge and the total pressure loss in its core are evident, but the secondary vortex is not captured due to lack of grid resolution near the body. Inspection of the bottom and top of the wing also reveals the presence of the wing boundary layer.

Figure 5.3 again shows contours of total pressure generated using thresholding. The left side of the figure shows the wing surface with numerous vortex cross sections on it. This shows the tendency of the vortex core to turn towards the free-stream direction upon leaving the trailing edge of the wing, and its tendency to move inboard as well. The right side of this figure shows the cross section of the vortex at the last streamwise station. Due to the coarse grid, the vortex has diffused significantly and is much weaker at this location than in the wing region.

The left side of Figure 5.4 shows another vortex cross section at a chordwise station in the wake and numerous particle paths on top of the wing. The particle paths are seen to wrap around this cross section, showing the swirling nature of the flow and indicating that they are following the vortex. The right side of Figure 5.4 is a detailed cross section of the vortex at the wake location shown in the left side of the figure. The feeding sheet may be seen rolling up into the vortex and the total pressure loss in the core is evident.

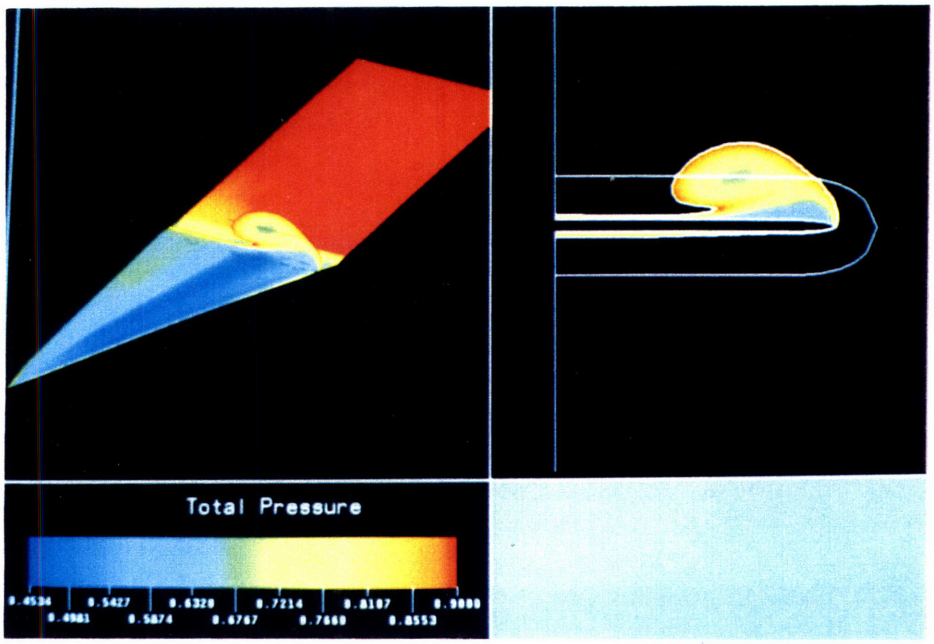

Figure 5.2: Total pressure contours on wing surface and vortex near trailing edge.

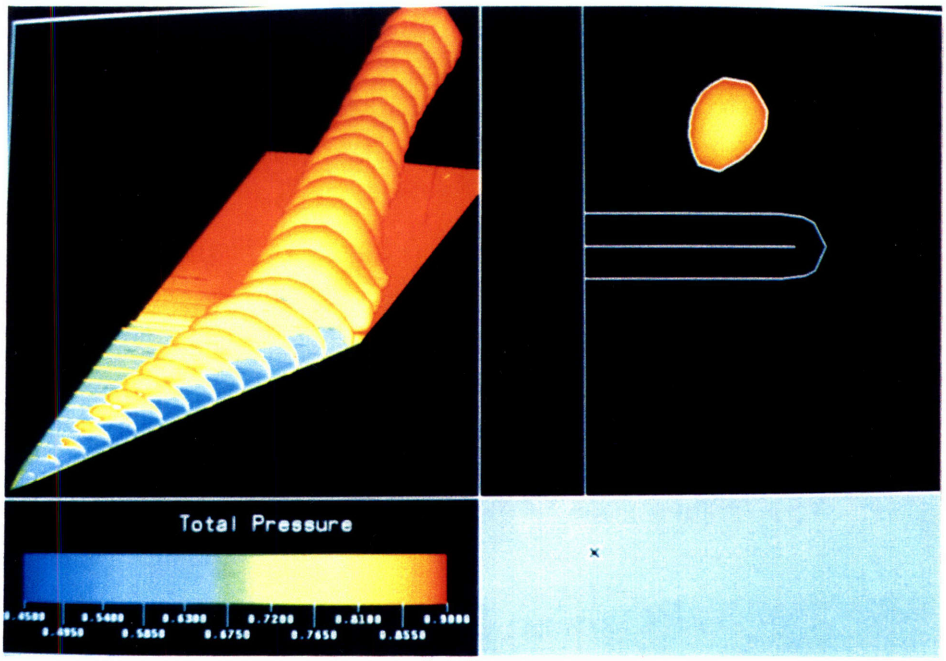

Figure 5.3: Thresholded total pressure contours showing vortex path.

5.1.2 Case 2 : Explicit TLNS – $M = 0.75$, $\alpha = 16°$, $Re_c = 1 \times 10^5$

The results obtained for this case are very similar to those from Case 1, so only limited data is presented. The increased resolution results in better capture of some flow features and less numerical diffusion of the vortex, but there is still not enough resolution in this computation to capture all the relevant viscous flow features, such as the secondary vortex.

The outer grid in this case has dimensions $25 \times 20 \times 40$ cells in the i, j and k directions, respectively. The inner grid dimensions are $20 \times 20 \times 40$ cells. This gives a total of 36,000 cells in the domain.

One intersting result for this case is shown in Figure 5.5. Compare this with Figure 5.6, which is a total pressure survey in the wake of a sharp edged delta wing, from an experiment by Fink [6]. They are remarkably similar qualitatively, although they vary quantitatively because Fink's experiment was run at a lower Mach number and angle of attack. Another difference is that and the experiment was run on a sharp edged wing, which will shed stronger vortices than a blunt edged one.

The convergence history for this case is shown in Figure 5.7. The error is small at the beginning because the history shown is for the period after a restart. The portion of the history before the restart was not available. The error is seen to drop two orders of magnitude in about 1500 iterations, never reaching machine zero. This is slower than the same flow case with the TLNS solver in semi-implicit mode, however an accurate comparison cannot be made because the grid sizes are different.

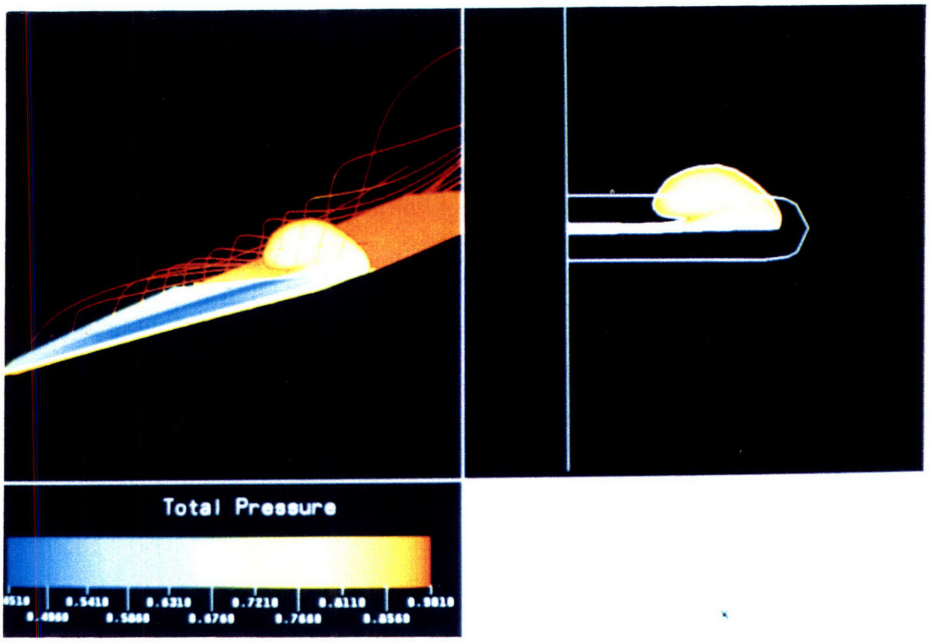

Figure 5.4: Total pressure contours showing particle paths and thresholded vortex in the wake.

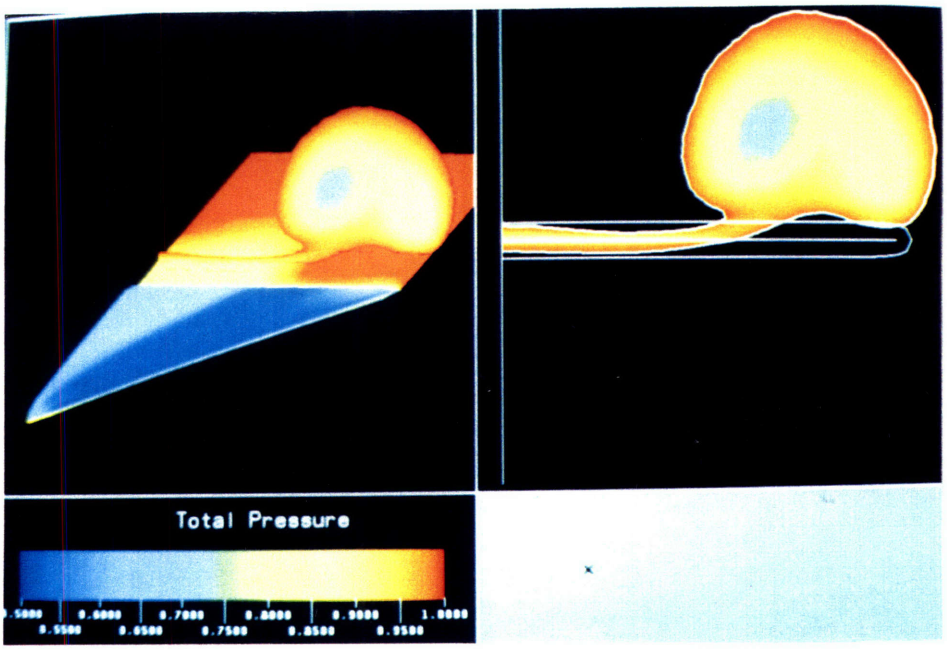

Figure 5.5: Thresholded Total pressure contours showing vortex in wake.

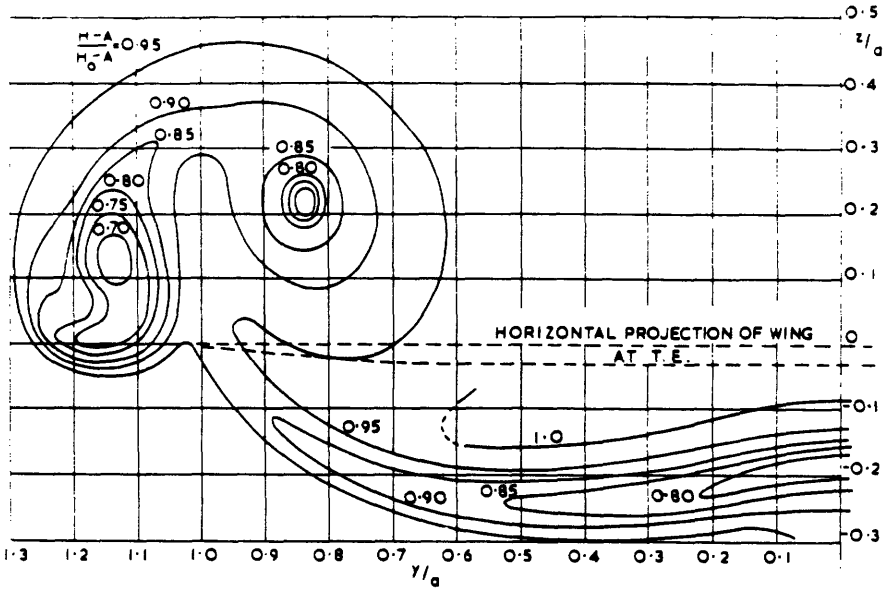

Figure 5.6: Total head survey in the wake of a sharp edged delta wing.

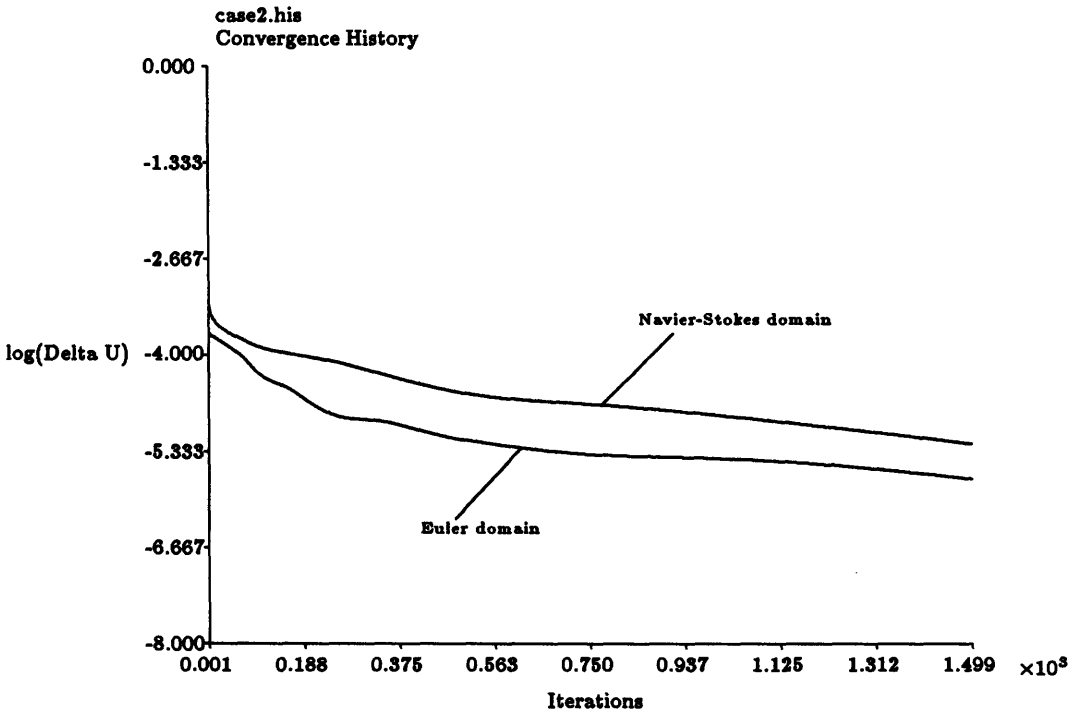

Figure 5.7: Convergence history for case 2.

5.2 Final Flow Results

Cases 3 and 4, which were run on the fine grid, are presented as "final" results, since there was sufficient resolution to capture all the flow phenomena in this case. The grid used in both of these cases has the dimensions described in Chapter 3.

For each case, data is presented for chordwise stations at 30 and 80 percent of chord. In all the figures presented, with the exception of planform views, the linear dimensions have been normalized by the local span of the wing. This allows comparisons to be made with greater ease.

5.2.1 Case 3: $M = 0.85$, $\alpha = 15°$, $Re = 2 \times 10^6$

Figure 5.8 shows the convergence history for this calculation. The error only drops about two orders of magnitude before the solution appears to stop converging. Ordinarily, one would expect the error to keep on dropping until machine zero, at which point the solution is considered converged. In the case of these vortex flows, however, it appears that the normal convergence criterion, derived from mathematical considerations, does not truly predict convergence. A vortical solution may in fact be converged in a global sense, however some localized unsteadiness may be present. Usually this appears as a "wobbling" of the vortex, which drives up the error and make it appear that the solution is unconverged. This should not come as a surprise, as a steady solution is not guaranteed anyway. Murman and Rizzi [22] also found this unsteadiness and lack of full convergence (in the mathematical sense) in their fine grid Euler solutions for flow over sharp edged delta wing. By examining the flowfield at a couple of iteration intervals, they found that all the flow features were predicted and the only change was a slight movement of the secondary vortex. Likewise, examination of the results that follow show that all the relevant flow features are captured, even though the solution is "unconverged" in the mathematical sense.

The presence of a vortex in this case is verified in Figure 5.9, which is a vector plot of the crossflow velocity at 30 percent chord . The primary vortex sitting above the wing at

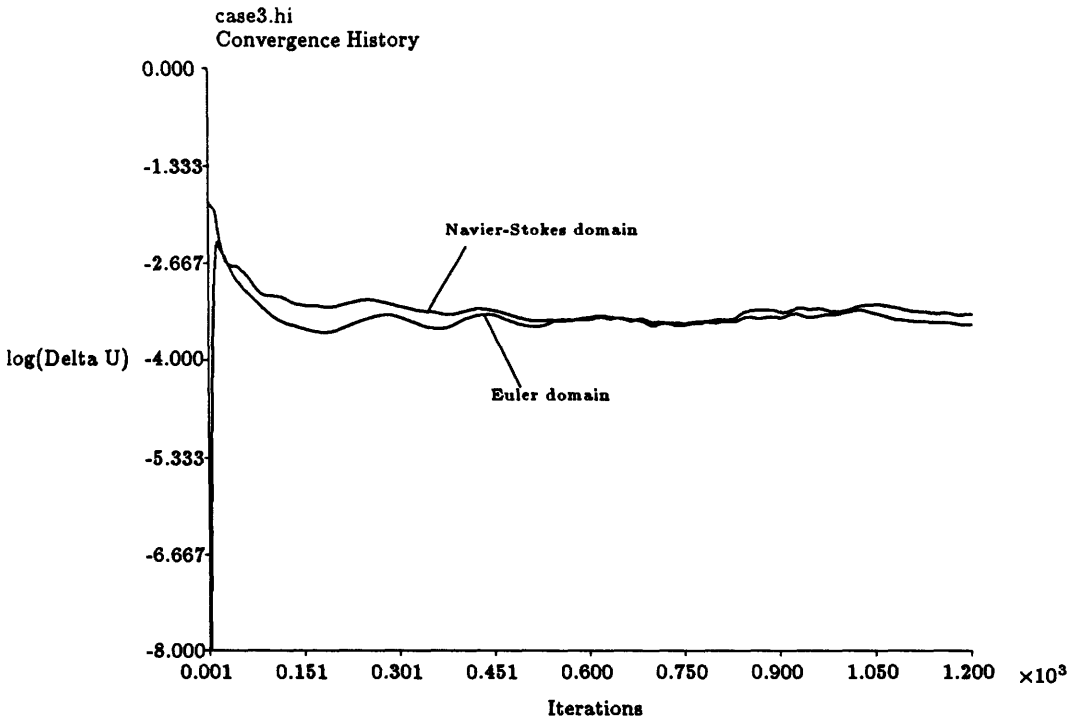

Figure 5.8: Convergence history for case 3.

about 65 percent of the local span is clearly visible. The secondary vortex has not yet fully developed this far forward on the wing, so it is not clearly visible in this plot. Inspection of the velocity vectors near the surface, which form a boundary layer profile, show that there is a region of reverse flow just outboard and underneath the primary vortex. There is significant displacement of the streamlimes over this reverse flow "bubble".

Mach number contours in the crossflow plane at 30 percent chord are shown in Figure 5.10. The primary vortex is visible in this plot, sitting above the wing. The Mach number in the core of the vortex is 1.387. This plot does not reveal much about the secondary vortex, however. The rapid expansion of the flow around the leading edge can be seen, as can the boundary layer forming on the wing, due to the Mach number going to zero on the wing surface as the no-slip condition is enforced.

Pressure coefficient information at 30 percent chord is given in Figures 5.11 and 5.12. The pressure coefficient is defined by the equation

$$C_P = \frac{P - P_\infty}{\frac{1}{2}\gamma P_\infty M_\infty^2}, \tag{5.1}$$

where P and P_∞ are the local and freestream pressures, respectively. Figure 5.11 shows

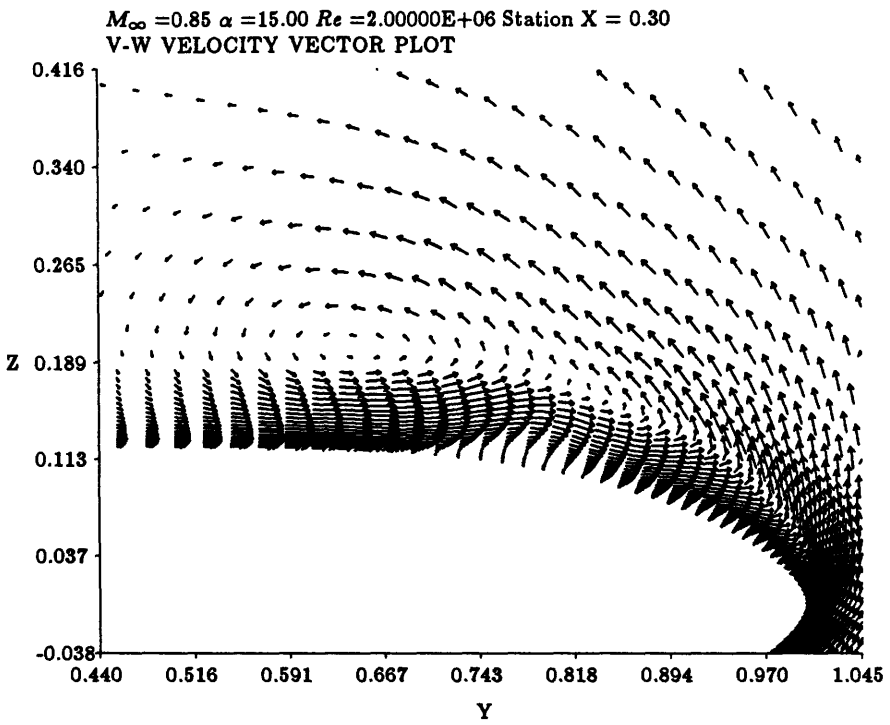

Figure 5.9: Crossflow velocity vector plot at 30 percent chord.

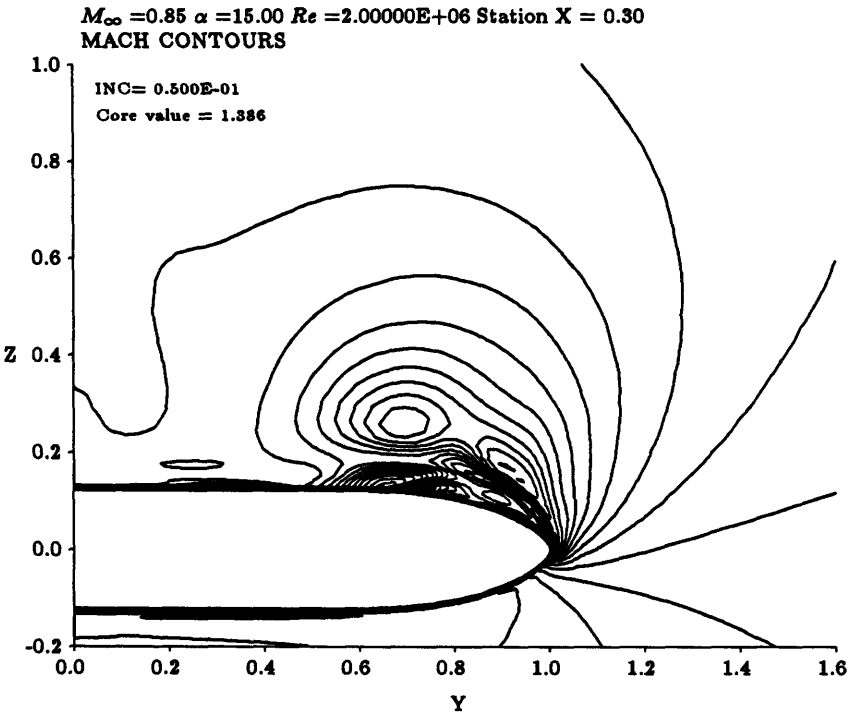

Figure 5.10: Mach number contours at 30 percent chord.

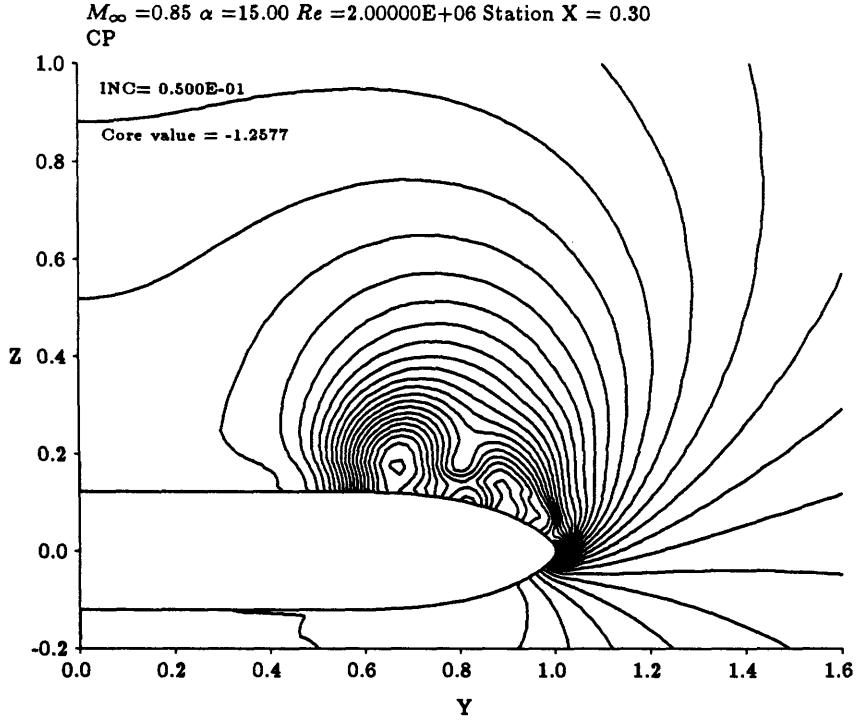

Figure 5.11: Pressure coefficient contours at 30 percent chord.

pressure coefficient contours in the crossflow plane. The core of the primary vortex is clearly seen in this figure, with a pressure coefficient of -1.2577 in the core. The secondary vortex does not show up in the contour plot. Figure 5.12, which is a plot of the spanwise C_P distribution on the wing surface, shows clearly the presence of both primary and secondary vortices. The suction peak on the left, with a C_P of about -1.25, is the region below the primary vortex, while the second suction peak to the right, having a C_P just slightly less than that of the primary, represents the secondary vortex.

Figure 5.13 is a contour plot of the total pressure loss C_{P_0} in the crossflow plane at 30 percent chord. The total pressure loss is defined by

$$C_{P_0} = \frac{P_{0_\infty} - P_0}{P_{0_\infty}}. \tag{5.2}$$

Again, the primary vortex is visible in this plot, having a C_{P_0} in the core of 0.491. The secondary vortex cannot be discerned. The wing boundary layer is clearly visible, as the total pressure loss will be high there due to the vorticity, which is related to the total pressure by Crocco's theorem.

The more interesting features of the flowfield show up in plots at 80 percent of the

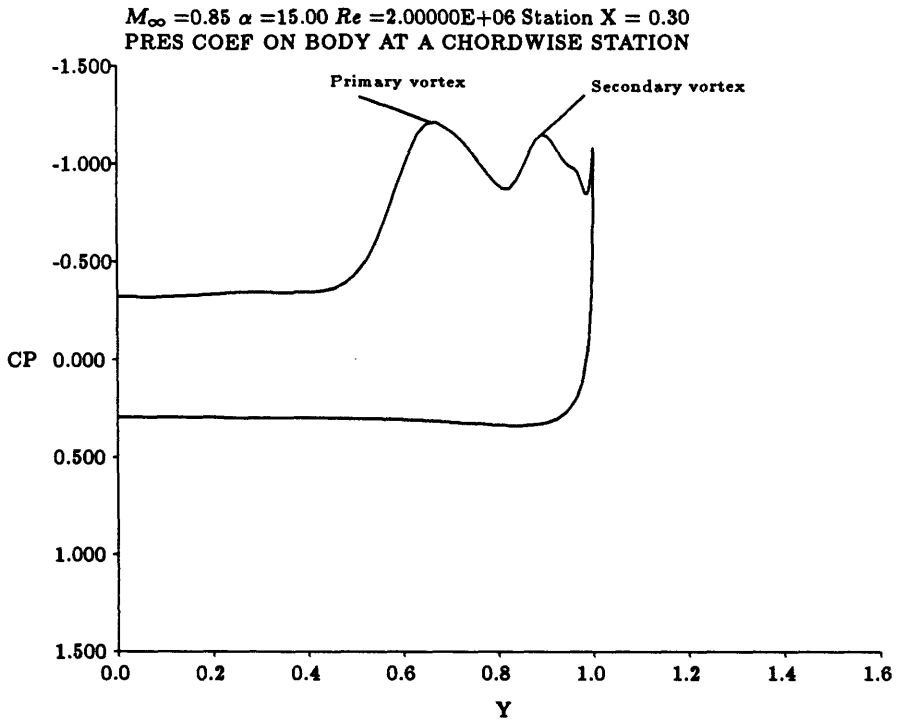

Figure 5.12: Spanwise distribution of pressure coefficient at 30 percent chord.

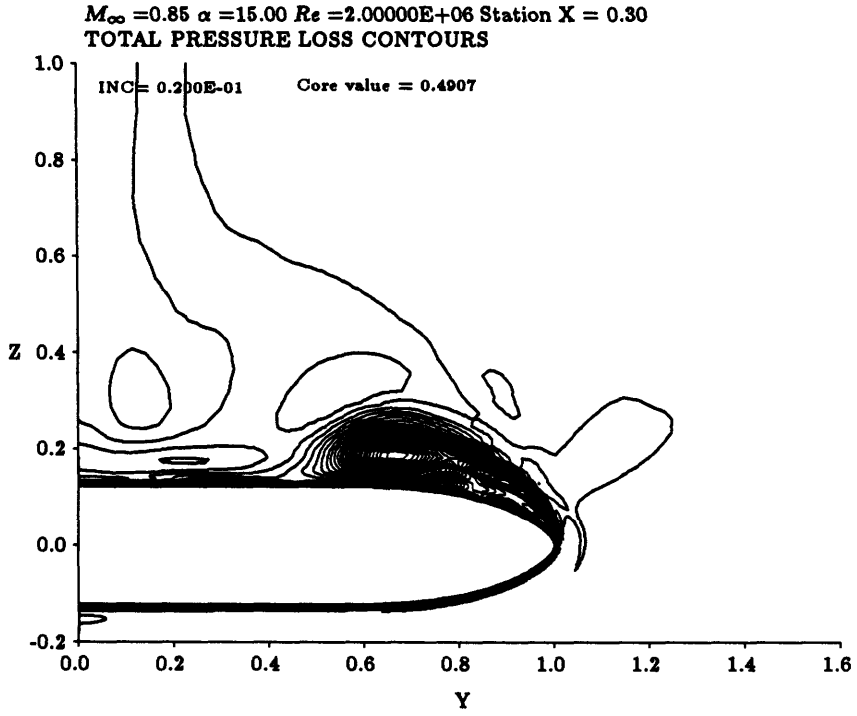

Figure 5.13: Total pressure loss contours at 30 percent chord.

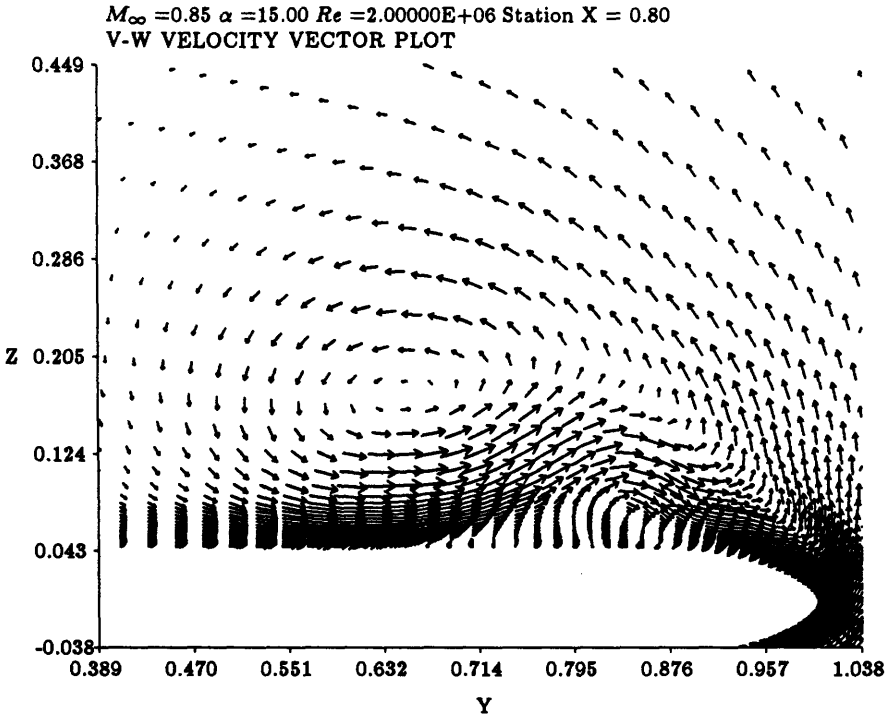

Figure 5.14: Crossflow velocity vector plot at 80 percent chord.

chord, as the primary vortex becomes stronger and the secondary vortex becomes better developed. Figure 5.14 is a plot of the crossflow velocity vectors at 80 percent chord. The primary vortex is clearly visible , as is the secondary vortex which is below and outboard of the primary vortex. Clear regions of reverse flow are seen by inspection of the boundary layer profiles outboard of the primary vortex. Comparison with the vector plot at 30 percent chord shows that the primary vortex has grown larger and stronger. The secondary vortex has also gained in strength and size, as shown by the increased deflection of the streamlines above it. The primary vortex is at about the same span fraction as at the 30 percent chord station. It is difficult to say from this plot whether the secondary vortex has moved, as it is hard to pinpoint its center based on the velocity vectors. A better measure of the secondary vortex position is found by looking at surface C_P distributions, as in Figure 5.17.

Mach number contours at 80 percent of chord are shown in Figure 5.15. The primary vortex and wing boundary layer are again clearly visible, while the secondary vortex is still not well defined in this plot. Inspection of this figure on the upper wing surface under the primary vortex reveals an area of high Mach number gradient. At first glance this appears to be a weak crossflow shock, however the pressure coefficient contours in Figure 5.16 do not

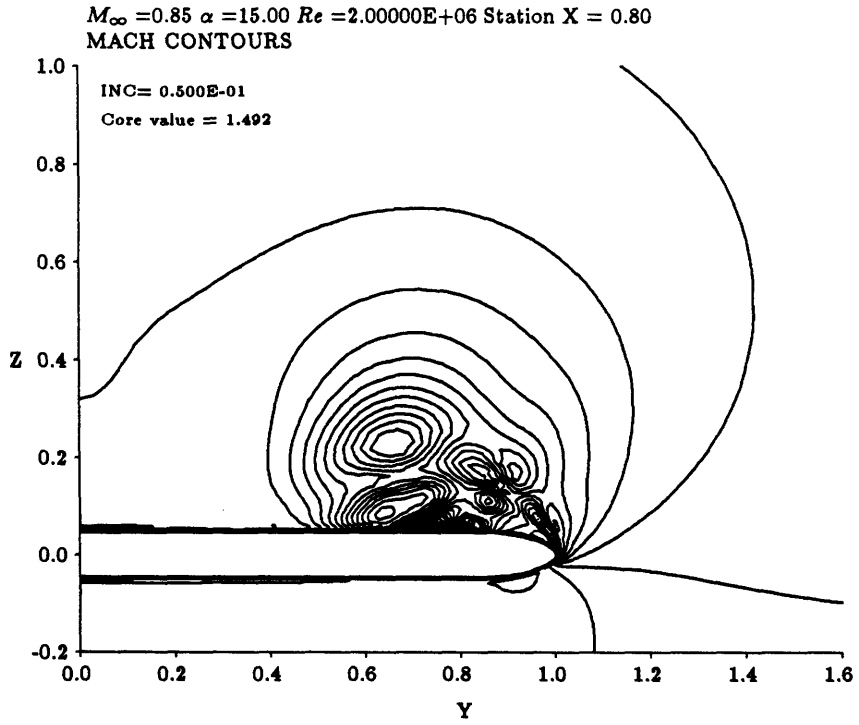

Figure 5.15: Mach number contours at 80 percent chord.

show the high pressure gradient which would correspond to a shock at that location. This region of high Mach number gradient is probably a shear layer, since the pressure doesn't increase abruptly. The possible instability of such a free shear layer may be responsible in part for the localized unsteadiness which is present.

The pressure coefficient contours at 80 percent chord are shown in Figure 5.16. The secondary and primary vortices are clearly visible, having pressure coefficients in their cores of -1.34604 and -1.4762 respectively. A region of high pressure gradient is seen just outboard of the secondary vortex. This could be interpreted as a shock, however the Mach contours of Figure 5.15 do not show a high Mach number gradient in this region. The spanwise pressure distribution of Figure 5.17 also does not reveal a sharp pressure rise there.

The spanwise surface distribution of C_P at 80 percent chord is shown in Figure 5.17. The suction peaks under the primary and secondary vortices are clearly seen. The reason for the small suction pressure peak outboard of the secondary vortex is unknown, but it is though to be related to phenomena occuring near the point of primary separation. Note that the high pressure gradient region shown in the C_P contours of Figure 5.16 just outboard

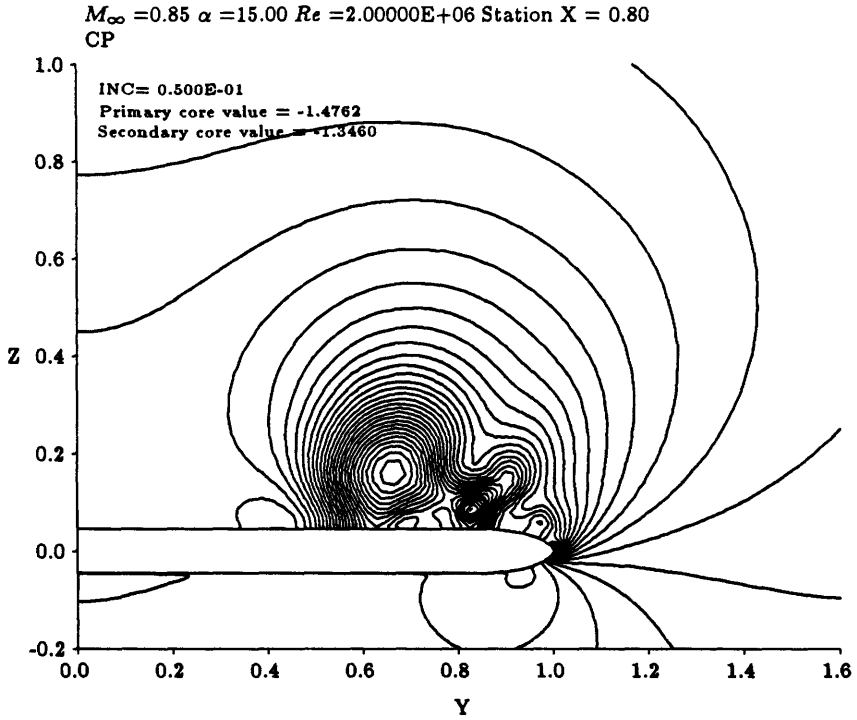

Figure 5.16: Pressure coefficient contours at 80 percent chord.

of the secondary vortex also shows up here. Comparison with the corresponding figure at 30 percent chord, Figure 5.12, shows that the primary vortex has remained at the same span fraction, while the secondary vortex has moved slightly inboard to about 82 percent of the local span, compared with 90 percent at the upstream station.

Contours of total pressure loss at 80 percent chord are shown in Figure 5.18. Again, the primary and secondary vortices are clearly visible, having total pressure losses in their cores of 0.5864 and 0.5234 respectively. The vortex sheet resulting from the secondary separation can be seen clearly rolling up into the secondary vortex.

Figure 5.19 is a contour plot of the total pressure loss at a station in the wake of the wing. The rollup of the feeding sheet into the vortex is evident. This again shows qualitative comparison with Figure 5.6, although in this case the contours are of total pressure *loss*, not total pressure itself.

The trace of the primary and secondary vortices on the wing planform is shown in Figure 5.20, which is a contour plot of the surface C_P on the lee-side of the wing. The primary and secondary vortices do not become distinguishable until about 30 percent of the chord.

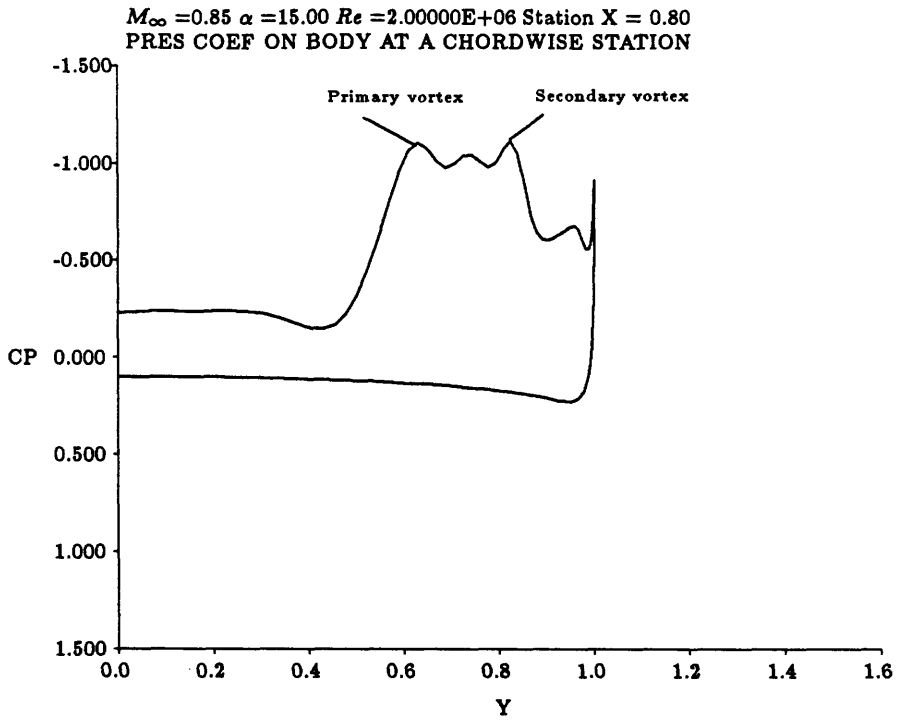

Figure 5.17: Spanwise distribution of pressure coefficient at 80 percent chord.

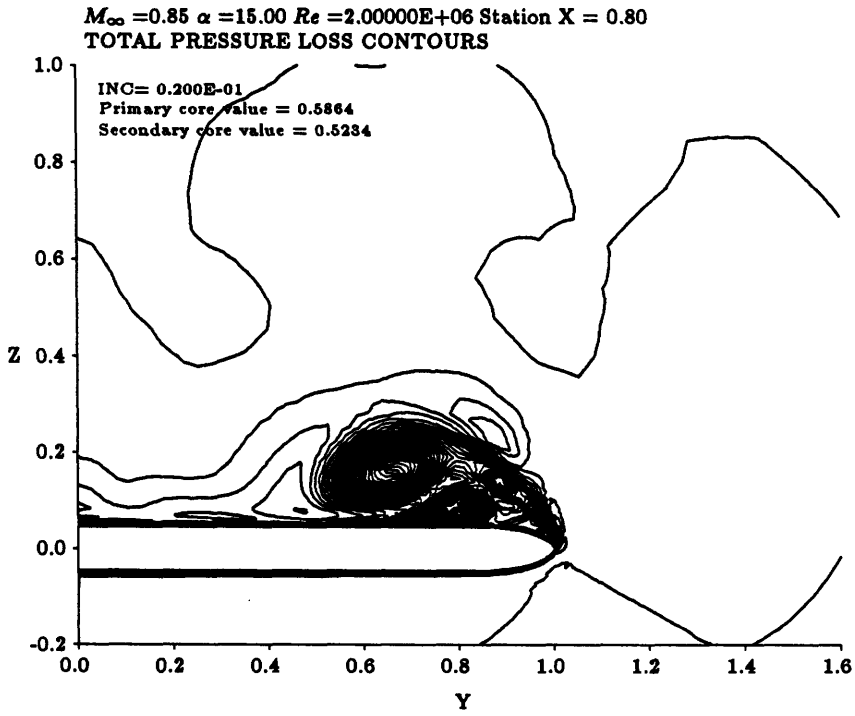

Figure 5.18: Total pressure loss contours at 80 percent chord.

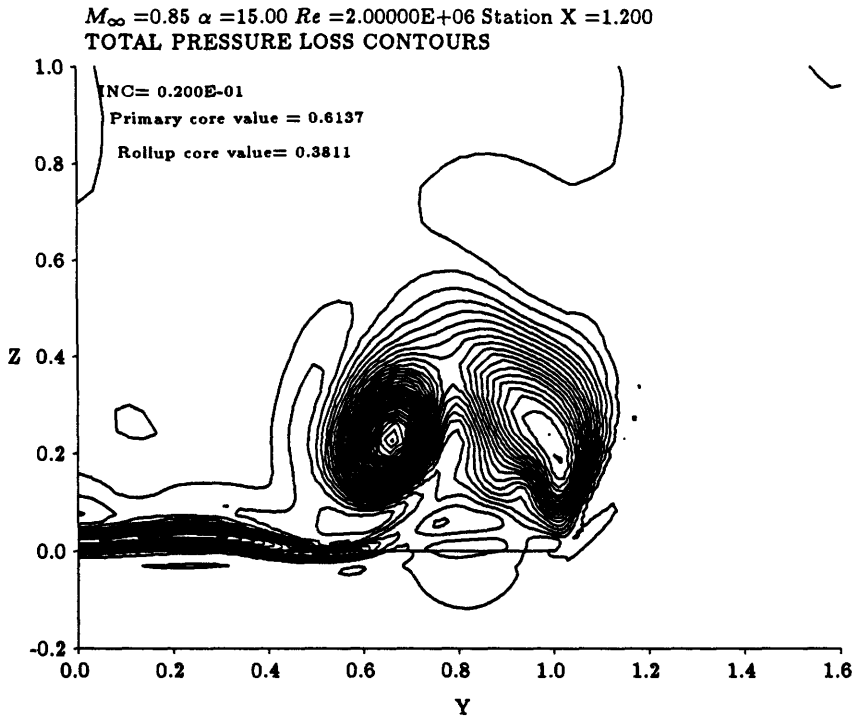

Figure 5.19: Total pressure loss contours in the wake (120 percent chord).

Thereafter, each vortex is visible and seems to follow essentially a straight line path as it travels down the wing.

5.2.2 Case 4: $M = 0.85$, $\alpha = 15°$, $Re_c = 7.5 \times 10^6$

This case presents a higher Reynolds number flow than that of Case 3. For this case, most of the data presented is at 80 percent of chord, as this is where the most significant differences with Case 3 show up. Most of the contour information at 30 percent chord does not reveal much difference between this case and Case 3. Plots of pressure coefficient at 30 percent chord are presented, as some difference is visible in these.

The convergence history for this case is shown in Figure 5.21. The large jump followed by a large drop in the error of the Euler domain is due to an initial start with high smoothing and a subsequent restart at about 500 iterations with a lower smoothing. Like Case 3, this case does not converge to machine zero but instead the error drops only a couple of orders of magnitude and then stops decreasing. This is for the same reasons cited in Case 3, i.e. that the vortex tends to wobble, driving the error up significantly, even though the solution

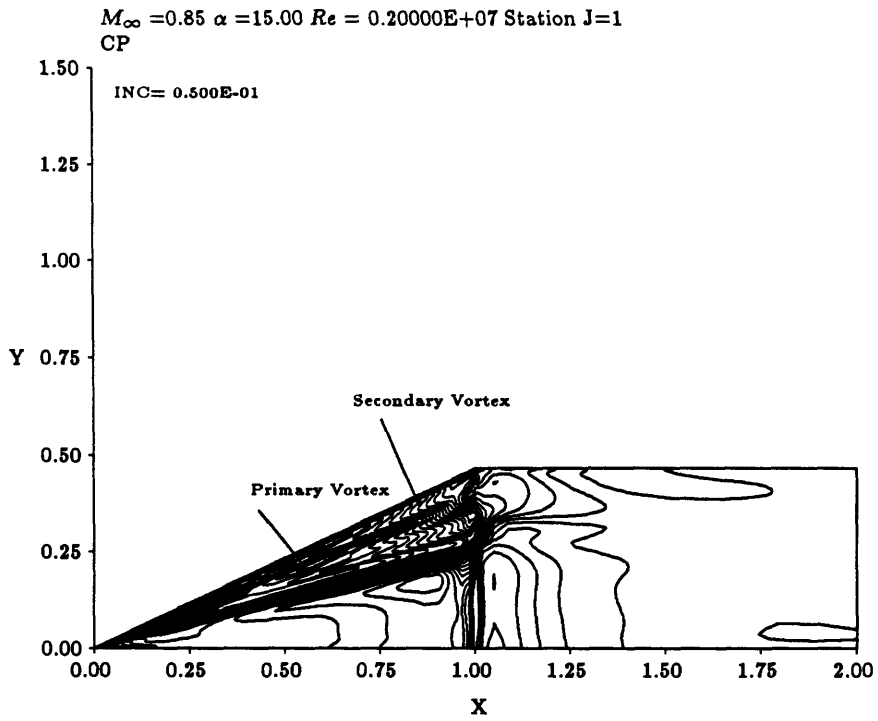

Figure 5.20: Pressure coefficient contours on the wing surface.

is converged in the remainder of the domain.

Cross flow velocity vectors at 80 percent chord are shown for this case in Figure 5.22. The primary vortex is clearly visible along with the secondary vortex and the displacement of streamlines about the secondary vortex. Examination of the velocity profiles on the wing in the area of the secondary vortex reveals some interesting velocity profiles and suggests the possibility that tertiary separation could be occurring.

Mach number contour plots at 80 percent chord are shown in Figure 5.23. The primary vortex is still clearly visible and the high Mach number gradient region, which is thought to be a shear layer, is still visible. The remainder of the flow structure is very complicated, making it hard to distinguish any other features. This may be related to the unsteadiness of the flow which led to the lack of full convergence.

Total pressure loss contours at 80 percent chord are shown in Figure 5.24. The primary and secondary vortices can be seen in this plot, having total pressure losses of 0.615 and 0.525 respectively. The vortex sheet from the secondary separation, and its subsequent rollup into the secondary vortex, can also be seen

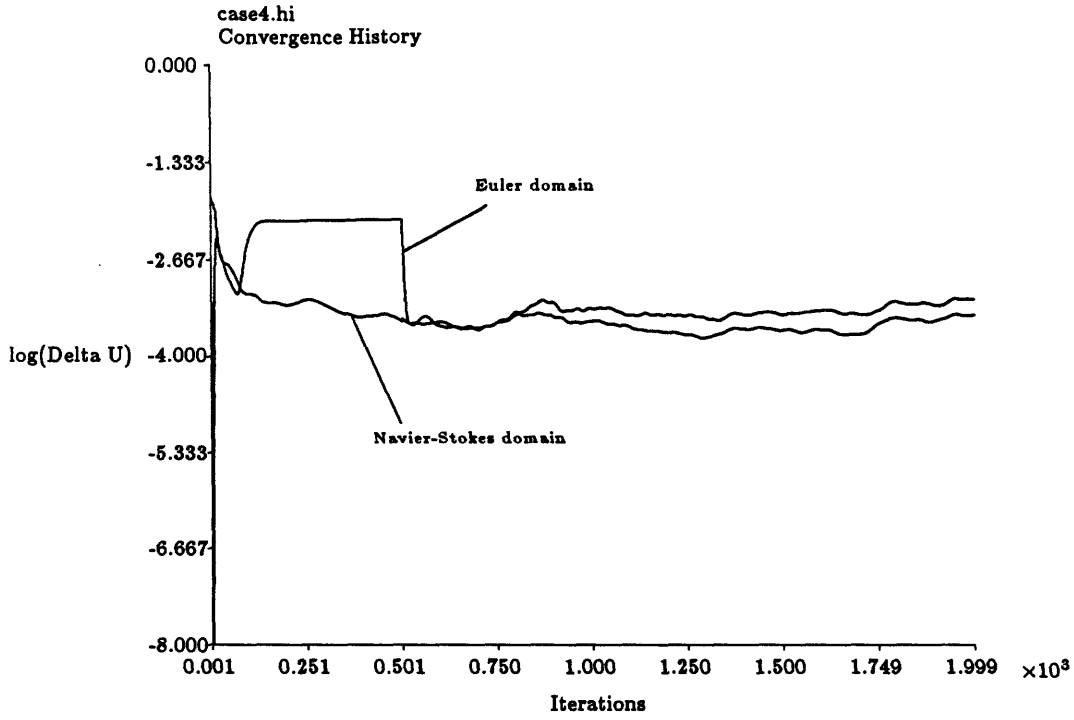

Figure 5.21: Convergence history for case 4.

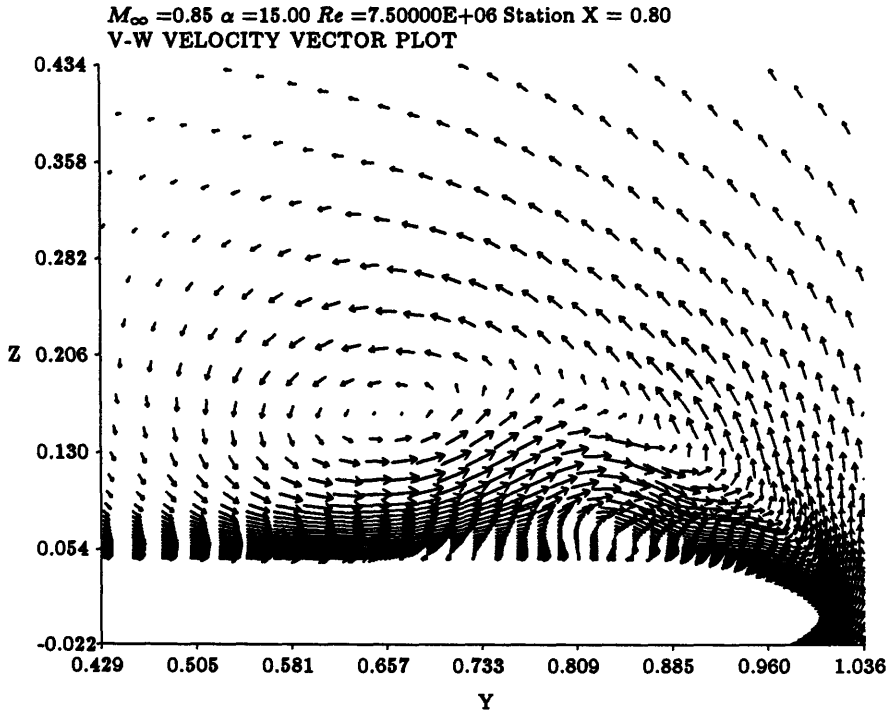

Figure 5.22: Crossflow velocity vector plot at 80 percent chord.

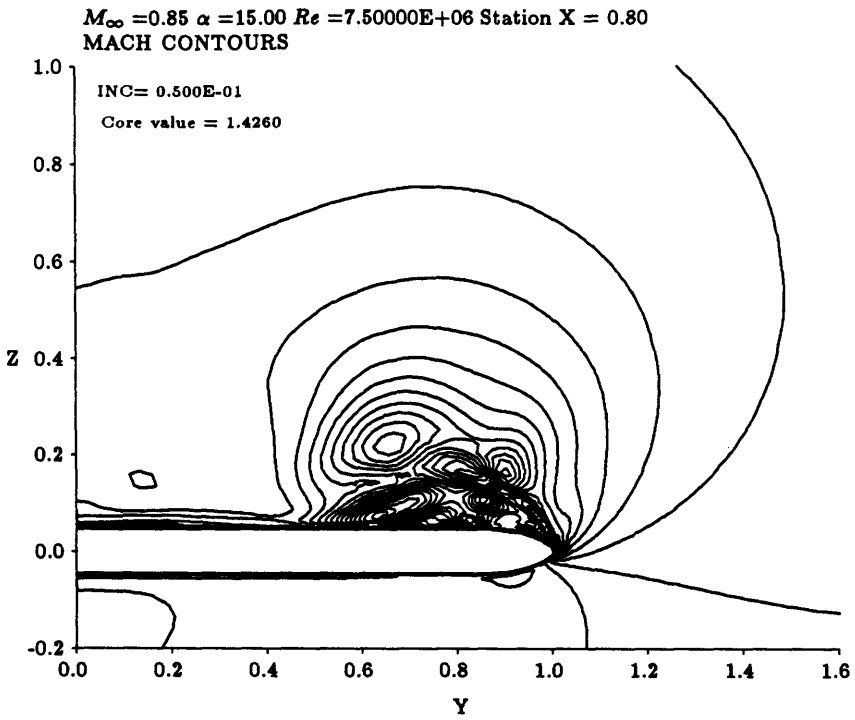

Figure 5.23: Mach number contours at 80 percent chord.

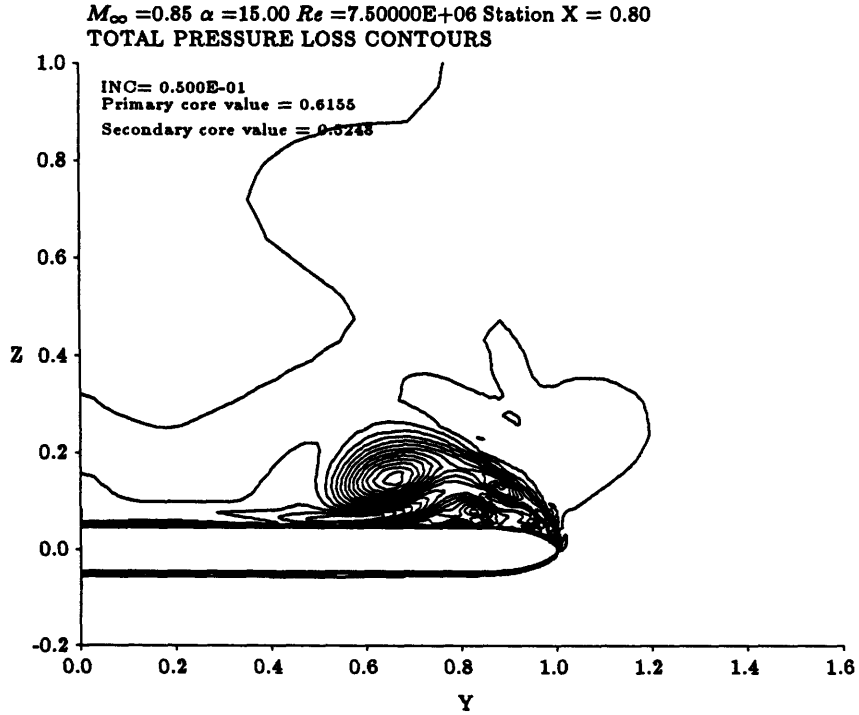

Figure 5.24: Total pressure loss contours at 80 percent chord.

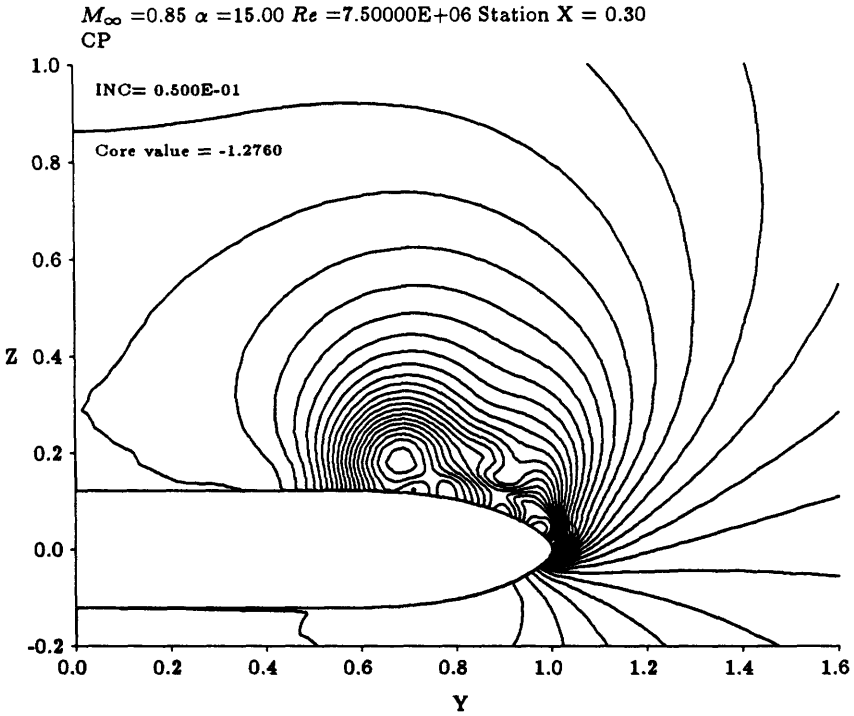

Figure 5.25: Pressure coefficient contours at 30 percent chord.

Pressure coefficient information at 30 percent chord is shown in Figures 5.25 and 5.26. Comparison of the contour plot in Figure 5.25 with the corresponding plot for Case 3, Figure 5.11, reveals that there is not much difference in contour plots at this location. The primary vortex is still seen in this case, but not much else. The spanwise pressure coefficient distribution at this station, Figure 5.26, again shows the primary and secondary vortices clearly, along with a suction peak outboard of the secondary vortex. There is some difference between this plot and the corresponding one for Case 3, however discussion of this is postponed until Section 5.3.

Information about the pressure coefficient at 80 percent chord is shown in Figures 5.27 and 5.28, respectively. The contour plot of Figure 5.27 shows the primary and secondary vortices having pressure coefficients in the core of -1.472 and -1.442, respectively. The spanwise surface C_P distribution of Figure 5.28 shows the primary and secondary vortices, although the structure of the vortex system is more complicated than in Case 3. The secondary vortex has grown considerably stronger than at 30 percent chord, and is stronger than the primary vortex at this station. The suction peak outboard of the secondary vortex is also seen in this plot.

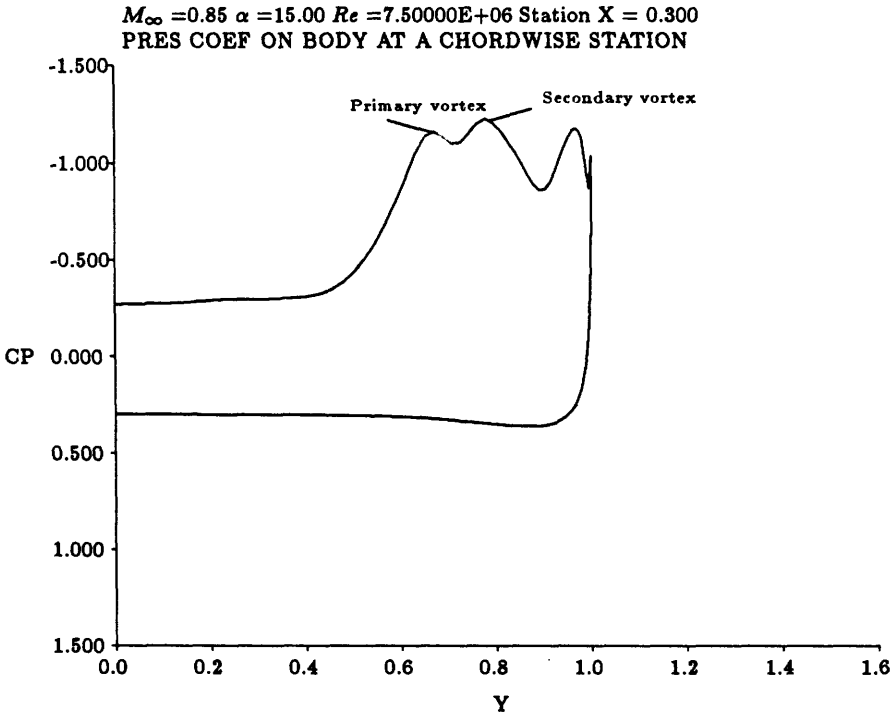

Figure 5.26: Spanwise distribution of pressure coefficient at 30 percent chord.

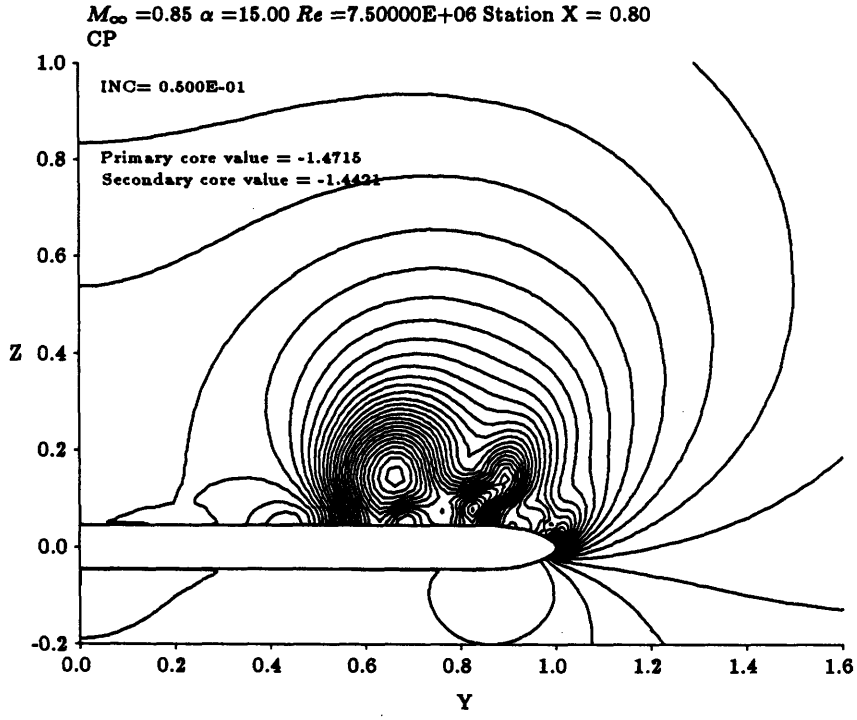

Figure 5.27: Pressure coefficient contours at 80 percent chord.

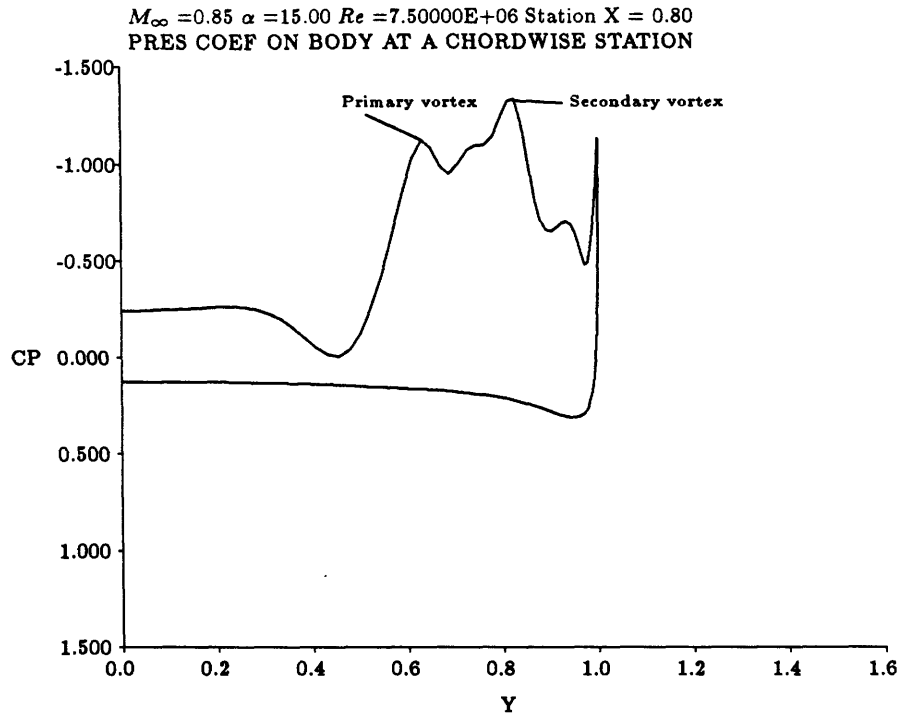

Figure 5.28: Spanwise distribution of pressure coefficient at 80 percent chord.

Pressure coefficient contours on the lee-side planform are shown in Figure 5.29. Again it shows the near straight line path of the vortex over the wing, but otherwise doesn't reveal much more than the similar figure of the previous case.

5.3 Reynolds Number Effect

An idea of the effect of the Reynolds number on the vortical flow can be obtained by comparison of some of the data from cases 3 and 4. Probably the best indicators of difference are the spanwise distributions of pressure coefficient, as they reveal strength and position of the vortices.

Comparison of the spanwise surface C_P distributions at 30 percent chord, Figures 5.12 and 5.26, show that the flow structures are similar, with the primary and secondary vortices being slightly stronger at the higher Re. The primary vortex is almost unaffected, whereas the C_P jumps from -1.2 for the low Re case to -1.3 for the high Re case. The largest effect is in the position of the secondary vortex, which has moved inboard for the higher Reynolds

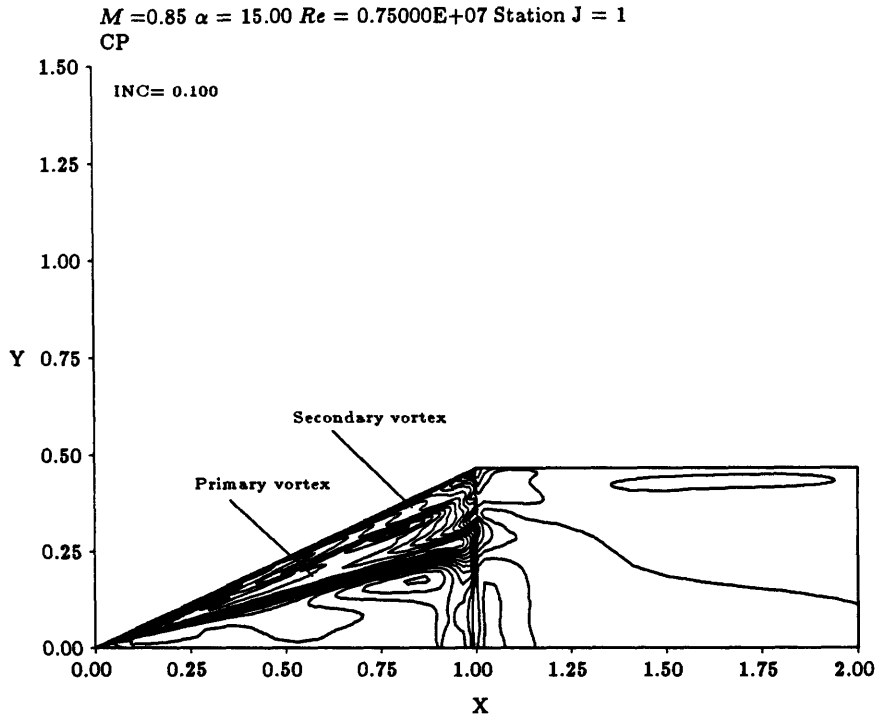

Figure 5.29: Pressure coefficient contours on the wing surface.

number. The primary vortex position has not changed significantly.

Comparison of spanwise C_P plots at 80 percent chord for Cases 3 and 4, Figures 5.17 and 5.28, show that at this station the primary vortex is only slihtly stronger at the higher Reynolds number. The secondary vortex has grown much stronger at the higher Reynolds number, with a peak suction pressure of about -1.4 for the $Re = 7.5 \times 10^6$ case and about -1.2 for the $Re = 2.0 \times 10^6$ case. The primary vortex stays in the same relative position for both cases, while the secondary vortex moves inboard for the higher Reynolds number. The inboard movement of the secondary vortex is not as pronounced at this chordwise station. The suction peak outboard of the secondary vortex also moves inboard for the higher Reynolds number.

Inspection of the Mach number contour plots at 80 percent chord, Figures 5.15 and 5.23 show that the flow structure is more complicated for the higher Reynolds number. The remaining contour plots do not reveal many differences in the flows at the two Reynolds numbers.

Some effects might also be introduced due to the action of the artificial viscosity. This

was not investigated here, but could be important. Further studies are needed.

5.4 Code Efficiency

Although one of the stated objectives of the research was to compute the flow over the NTF wing using a zonal method to acheive computational efficiency, the efficiency of the code was never measured against a full N-S code. This is left as a future exercise.

Chapter 6
Conclusions

The main goal of this research was to correctly capture the viscous dominated flow physics of a blunt edged wing, such as the separation point and secondary vortex, while attaining computational efficiency by not computing extra viscous terms in parts of the domain where they are negligible anyway. A secondary goal was simply to investigate the use of this multi-block, multi-solver approach for these types of flow.

To accomplish these goals, a hybrid Euler/Navier-Stokes method for solving the laminar vortical flows over the blunt-edged NTF delta wing is presented. An Euler solver is run in the area of the domain where the flow is essentially inviscid, and the Navier-Stokes solver is run in the area where viscous effects dominate. The two solvers are coupled at the interface between the two regions of the flow.

The semi-implicit mode of the thin layer Navier-Stokes (TLNS) solver was found to lack robustness. To acheive convergence, the exact combination of smoothing, time step and other numerical parameters had to be found. This amounted to trying many, many runs. Unfortunately, the correct combination was never found for the fine grid. Only one semi-implicit solution was found, using a coarse grid of 11,400 cells. With the TLNS solver running in explicit mode, the code was much more robust. However, due to the poor convergence characteristics of explicit schemes for high Reynolds number flows, a fully converged solution (i.e. the error goes to machine zero) was never obtained.

Even without full convergence in the mathematical sense, all the relevant features of the flow were captured. The treatment applied at the interface between solvers also worked well, providing smooth and continuous solutions across the interface. Timing studies were

not run on the code, so its efficiency compared to a full Navier-Stokes code is uncertain. This is left as future work.

The two final results at $M_\infty = 0.85$, $\alpha = 15°$ and Reynolds numbers of 2.0×10^6 and 7.5×10^6 were quite detailed and revealed a highly complex transonic vortical flow over the NTF wing. The effect of Reynolds number on the vortical flow was examined; the primary vortex was found to be insensitive to Reynolds number, while the strength and position of the secondary vortex were more dependent. Qualitative agreement with known flows over other delta configurations was good, however it is impossible to fully judge the quality of the results, as there is no experimental data for comparison.

Numerous possibilities exist for future work which may be done with this hybrid code. They are as follows:

- Change the algorithm from a central difference scheme to , for example, an upwinding scheme. This would improve the robustness of the code, since upwind codes are usually very robust.
- Investigate ways of enhancing the robustness of the current central difference scheme, such as using integration schemes with different numbers of stages and alternative strategies for evaluation of the dissipation terms at each stage.
- Convergence acceleration techniques, such as multigrid, could be added to improve the convergence characteristics of the code.
- The dual-solver approach could be tried using adaptation or grid embedding. This has the potential of generating some good, detailed results.
- Timing studies should be run on the hybrid code to evaluate its efficiency compared to a full Navier-Stokes simulation.
- Finally, a turbulence model should be added to assess the effects of turbulence on these types of flows.

Bibliography

[1] Anderson,D.A., Tannehill,J.C., Pletcher,R.H. : *Computational Fluid Mechanics and Heat Transfer* ,Hemisphere Publishing Co., 1984.

[2] Cebeci, T., Smith, A.M.O. : *Analysis of Turbulent Boundary Layers*, Academic Press, New York, 1974.

[3] Earnshaw,P.B. : " An Experimental Investigation of the Structure of a Leading-Edge Vortex", A.R.C. Reports and Memoranda, No. 3281, March 1961.

[4] Eriksson,L.E., et al : "Grid Generation and Inviscid Flow Computation About a Cranked-Winged Airplane Geometry", *Journal of Aircraft*, Vol. 25, No. 9, Sept 1988, pp.820-827.

[5] Eriksson,L.E., Rizzi,A. : "Analysis by Computer of the Convergence to Steady State of Discrete Approximations to the Euler Equations", AIAA Paper 83-1951, 1983.

[6] Fink, P.T. : "Some Early Experiments on Vortex Separation, Parts I, II and III", A.R.C. Reports and Memoranda, No. 3489, September 1966.

[7] Fujii,K., Schiff,L.B. : "Numerical Simulation of Vortical Flows over a Strake-Delta Wing", AIAA Paper 87-1229, June 1987.

[8] Goodsell,A.M. : "3-D Euler Calculations of Vortex Flows over Delta Wings", MIT Report CFDL-TR-87-6, July 1987.

[9] Hall,M.G. : "Vortex Breakdown", RAE Report, 1972.

[10] Hoeijmakers,H.W.M. : "Computational Vortex Flow Aerodynamics, Aerodynamics of Vortical Type Flows in Three Dimensions", AGARD CP-342, Paper No. 18, July 1983.

[11] Jameson,A., Schmidt,W., Turkel,E. : " Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time Stepping Schemes", AIAA Paper 81-1259, 1981.

[12] Jameson,A., Baker,T.J. : " Solution of the Euler Equations for Complex Configurations", ASME Symposium on Numerical Methods for Compressible Flows, Dec. 1986.

[13] Kandil,O.A., Chuang,A. : "Influence of Numerical Dissiaption in Computing Supersonic Vortex Dominated Flows", AIAA Paper 86-1073, May 1986.

[14] Krouthen,B. : "A Numerical Investigation of Hot Streaks in Turbines", MIT Report CFDL-TR-88-9, June 1988.

[15] Lee,K.M. : Verbal Communication, May 1989.

[16] Liebovich,S. : " The Sturcure of Vortex Breakdown", *Annual Review of Fluid Mechanics*, Vol. 10, pp. 221-246, 1978.

[17] Loyd, B. : "A Semi-Implicit Navier-Stokes Solver and Its Application to a Study of Separated Flow about Blunt Delta Wings", PhD Thesis, Massachusetts Institute of Technology, Feb 1989.

[18] Loyd,B., Murman,E.M., Abarbanel,S.S. : "A Semi-Implicit Scheme for the Navier-Stokes Equations", 7th GAMM Conference on Numerical Methods in Fluid Mechanics, Louvain-la-Neuve, Belgium, Sept. 1987.

[19] Luckring,J.M. : " NTF Delta Wing Geometry ", Model Description, NASA Langley, 1987.

[20] Miller,D.S., Wood,R.M. : "An Investigation of Wing Leading-Edge Vortices at Supersonic Speeds", AIAA Paper 83-1816, July 1983.

[21] Murman,E.M., Giles,M.B. : Class notes from MIT Course *16.026: Advanced Computational Fluid Dynamics*, Spring 1988.

[22] Murman,E.M., Rizzi,A. : " Application of Euler Equations to Sharp Edge Delta Wings With Leading Edge Vortices", *Applications of Compuational Fluid Dynamics in Aeronautics*, AGARD Conference Proceedings, No. 412, 1986.

[23] Newsome,R.W., Kandil,O.A. : "Vortical Flow Aerodynamics - Physical Aspects and Numerical Simulation", AIAA Paper 87-0205, January 1987.

[24] Payne,F.M., et al. : " Experimental Study of the Velocity Field on a Delta Wing", AIAA Paper 87-1231, June 1987.

[25] Powell, K.G. : "Vortical Solutions of the Conical Euler Equations", PhD thesis, Massachusetts Institute of Technology, June 1987.

[26] Powell,K.G., Murman,E.M., Wood,R.M., Miller,D.S. : "A Comparison of Experimental and Numerical Results for Delta Wings with Vortex Flaps", AIAA Paper 86-1840-CP, June 1986.

[27] Rizzi, A., et al. : "Navier Stokes and Euler Solutions for Vortex Flows over a Delta Wing", *Symposium Transsonicum III*, Springer-Verlag 1989.

[28] Roberts,T.W. : "Euler Equation Computations for the Flow over a Hovering Helicopter Rotor", PhD Thesis, Massachusetts Institute of Technology, November 1986.

[29] Schlichting, H. : *Boundary Layer Theory*, Sixth Edition, McGraw-Hill Book Co, New York, 1968.

[30] Stanbrook,A., Squire,L.C. : "Possible Types of Flows at Swept Leading Edges", *Aeronautical Quarterly*, Vol. 15, No. 1, pp. 72-82, February 1964.

[31] Turkel,E. : " Acceleration to a Steady State for the Euler Equations", ICASE Report No. 84-32, 1984.

Appendix A

Geometry Constants for the NTF Delta Wing

This appendix gives the values of the constants a, b, c, and d for computation of the geometry of the leading edges and the trailing edge closure region of the NTF Wing.

The constants for the three leading edges of the NTF wing are given in Table A.1 below:

Table A.1: Leading Edge Geometry Constants

L.E. No.	$r/\bar{c}$	a	b	c	d
1	0.05	0.06666666666667	0.21501600073802	-0.25668266740469	0.08833866691267
2	0.15	0.11547005383792	0.12350964979191	-0.19567843344062	0.07003739672345
3	0.30	0.16329931618554	0.03382978289013	-0.13589185550609	0.05210142334309

The constants for the the trailing edge closure region are given in Table A.2 below:

Table A.2: Trailing Edge Closure Constants

$r/\bar{c}$	a	b	c	d
0.00	0.00	0.51002400110703	-0.51002400110703	0.17000800036901