Higher-Order Spatial Discretization for Turbulent Aerodynamic Flows

by

Stan De Rango

A thesis submitted in conformity with the requirements
for the degree of Doctor of Philosophy
Graduate Department of Aerospace Science and Engineering
University of Toronto

0-612-63580-5

Abstract

Higher-Order Spatial Discretization for
Turbulent Aerodynamic Flows

Stan De Rango
Doctor of Philosophy
Graduate Department of Aerospace Science and Engineering
University of Toronto
2001

A higher-order algorithm has been developed for computing steady turbulent flow over two-dimensional airfoils. The algorithm uses finite-differences applied through a generalized curvilinear coordinate transformation, applicable to single- and multi-block grids. Numerical dissipation is added using the matrix dissipation scheme. Turbulence is modeled using the Baldwin-Lomax and Spalart-Allmaras models. The various components of the spatial discretization, including the convective and viscous terms, the numerical boundary schemes, the numerical dissipation, and the integration technique used to calculate forces and moments, have all been raised to a level of accuracy consistent with third-order global accuracy. The two exceptions, both of which proved not to introduce significant numerical error, are the first-order numerical dissipation added near shocks and the first-order convective terms in the Spalart-Allmaras turbulence model. Results for several grid convergence studies show that this globally higher-order approach produces a dramatic reduction in the numerical error in drag. It can provide equivalent accuracy to a second-order algorithm on a grid with several times fewer nodes. For subsonic and transonic single-element cases, errors of less than two percent are obtained on grids with only 15,000 nodes while 4 times as many nodes are required for the second-order algorithm. Similar accuracy is obtained for a three-element case on grids with only 73,000 nodes, a third of that required by the second-order algorithm. The results provide a convincing demonstration of the benefits of higher-order methods for practical flows.

Acknowledgements

I have learned a great deal during my stay at UTIAS and have made many friends along the way. I extend my gratitude to each and everyone for their contributions to this work:

- My supervisor, Professor D.W. Zingg, whose guidance and tutelage greatly shaped my knowledge and understanding of my research and studies.

- To the members of the CFD Group, who made my stay at UTIAS both interesting and enjoyable. To Marian Nemec, who worked with me on many projects, and from whom I learned a great deal. Our lunch-time jogs will be missed. To the system administrators, Jason Lassaline and Mike Sullivan, who donated much of their time to helping the group members with software and network problems. To Gary Zuliani, Luis Manzano and Todd Chisholm, all of whom could always be counted on for a laugh or two to throw a little colour into my day.

- Tom Nelson's help with issues relating to TORNADO is greatly appreciated.

- My first few years at UTIAS were shared with a dear friend, Alberto Pueyo. Our conversations spanned many topics and influenced my life on many levels. I enjoyed immensely the many soccer and ball-hockey games that we played together.

- To all my family members, I would like to express my gratitude for their ongoing support throughout my educational endeavors and I share my achievements with them.

- Erika deserves a special place in my gratitude. She has been by my side for the better part of a decade as I completed the various stages of my post-secondary and graduate education. I drew upon her strength and support to remain determined and focussed on my goals. I owe her more than I could possibly convey in words.

Finally, I am very grateful to those who supported me financially: Bombardier Aerospace, the Government of Ontario (OGS), the University of Toronto, the Natural Sciences and Engineering Research Council of Canada, my supervisor and my parents.

STAN DE RANGO

University of Toronto
June 14, 2001

Contents

List of Tables

List of Figures

List of Symbols

A	inviscid flux Jacobian matrix in x-direction
$\hat{A}$	inviscid flux Jacobian matrix in ξ-direction
B	inviscid flux Jacobian matrix in y-direction
$\hat{B}$	inviscid flux Jacobian matrix in η-direction
C_N	normal force coefficient
C_A	axial force coefficient
C_l	two-dimensional lift coefficient
C_d	two-dimensional drag coefficient
C_{d_p}	two-dimensional pressure drag coefficient
C_{d_f}	two-dimensional friction drag coefficient
$\hat{K}$	thin-layer viscous flux Jacobian matrix
E	inviscid flux vector in x-direction
$\hat{E}$	inviscid flux vector in ξ-direction
E_v	viscous flux vector in x-direction
F	inviscid flux vector in y-direction
$\hat{F}$	inviscid flux vector in η-direction
F_v	viscous flux vector in y-direction
H	enthalpy
J	metric Jacobian
M	Mach number
Q	vector of conservative variables
$\hat{Q}$	vector of conservative variables in computational domain
$\hat{R}$	residual vector in computational domain
R_1, R_2	Riemann invariants
$\hat{S}$	viscous flux vector in thin-layer approximation
T	eigenvector matrix
U	contravariant velocity
V	contravariant velocity
V_n	normal velocity
V_t	tangential velocity

a speed of sound
c airfoil chord
c_p specific heat at constant pressure
e total energy
p pressure
t time
u x-component of velocity
v y-component of velocity
u^+ x-component of velocity divided by friction velocity
u_τ friction velocity
x physical (Cartesian) coordinate
y physical (Cartesian) coordinate

$\mathcal{P}r$ Prandtl number
$\mathcal{P}r_t$ turbulent Prandtl number
$\mathcal{R}e$ Reynolds number

Λ eigenvalue matrix
Γ circulation around the airfoil
α angle of attack
γ specific heat ratio
λ eigenvalue
κ_2 second-difference dissipation coefficient
κ_4 fourth-difference dissipation coefficient
κ_t thermal conductivity
μ dynamic viscosity
ν kinematic viscosity
ν_w kinematic viscosity on airfoil surface
μ_t eddy viscosity
τ_w shear stress on airfoil surface
ξ curvilinear coordinate
η curvilinear coordinate
Δt time step
Δx grid spacing in x-direction
ρ density
ρ_w density on airfoil surface

Chapter 1

Introduction

In any free-market economy, businesses work diligently to obtain some sort of advantage over their competition. Market dominance is often achieved by bringing a product to market quickly and cheaply. To do so involves efficient product design and development. This is certainly true in the commercial aircraft industry where the recent decade has seen Bombardier, the first company to introduce the regional jet, go on to play a dominant role in the regional jet industry.

The process of designing an aircraft has matured greatly since the Wright Flyer took its first powered flight in 1903. Early on, engineers employed empirical approaches to solve aerodynamic problems. In the 1950s computational aerodynamics, as a subset of computational fluid dynamics (CFD), was in its infancy. The computation of simple academic flows using linear equations and a few hundred unknowns was considered state of the art. But aircraft designs grew increasingly complex and so too did the types of flow conditions that needed to be examined. Designers began relying heavily on experiments in wind tunnels. The process was slow and expensive. It took almost 20,000 hours of wind tunnel testing to develop the General Dynamics F111 and the Boeing 747 [27]. Advancing computer capabilities have since enabled exceptional growth in CFD. The state of the art has now evolved to the ability to evaluate the flow about complete aircraft configurations using non-linear equations and several million unknowns. Dependence on wind tunnel testing has been significantly reduced with the increased use of CFD.

Today's CFD algorithms, however, do have their limits. Designers and engineers require computational methods that are robust, accurate, computationally inexpensive, and provide a fast turn-around. A substantial portion of the time required in a simulation based on the Navier-Stokes (NS) equations is involved in problem setup, including geometry definition and grid generation. Solutions to this obstacle are unable to keep pace with advances in computer technology and thus, should be the focus of intense research by the CFD community. Equally challenging, however, is the accurate and cost-effective simulation of viscous flow at high Reynolds numbers associated with full scale flight. The time required to compute the solution of the steady compressible Navier-Stokes equations for the flow about a complete aircraft remains excessive for routine use in aircraft design. Consequently, numerical solution techniques applicable to simpler physical models, such as panel methods or inviscid solvers incorporating the boundary-layer equations, are still heavily used in the design process, despite their limitations.

There are a number of ways to reduce the solution time for solving the Navier-Stokes equations. The first is to exploit computer technology to the fullest. Many CFD developers have turned to running existing algorithms on massively parallel computers. Another solution is to improve the iterative method used to achieve steady state such as modern Newton-Krylov type solvers [40]. An alternative approach would be to reduce the mesh requirements by improving the accuracy of the spatial discretization. This allows a reduction in the number of grid nodes required to achieve a given level of accuracy, resulting in savings in both computing time and memory. The accuracy of the spatial discretization can be improved by increasing the order of accuracy of the discrete operator. The purpose of this work is to use a higher-order spatial discretization to improve the efficiency of solving the Navier-Stokes equations for steady aerodynamic flows. In this work, the term "higher-order" is used to indicate orders of accuracy higher than second.

1.1 Background

Higher-order spatial discretizations are typically more computationally expensive per grid node than first- or second-order methods. They require smaller grid densities, however, for a given level of accuracy. The increase in the computing expense per node is generally outweighed by the reduction in the number of grid nodes needed, reducing the overall computing expense. The promise of higher-order methods has been recognized for some time, beginning with papers by Kreiss and Oliger [32] and Swartz and Wendroff [55]. These authors examined the application of various finite-difference spatial schemes to linear first-order hyperbolic equations with periodic boundary conditions. They demonstrated the ability of higher-order centered schemes to significantly reduce the number of nodes required to minimize phase speed errors when numerically simulating the propagation of linear waves. They showed that, for the cases studied, there was no significant advantage to using accuracies higher than sixth-order. In general, as the algorithm accuracy increased, the benefits, in the form of reduced grid density requirements, did not offset the added computational effort of the algorithm. Fornberg [21] performed a similar study of spatial difference schemes, including the use of fast Fourier transforms (FFT). The spatial derivatives of the governing equations were replaced with the FFT scheme, producing very accurate difference equations.

Higher-order methods have received considerable use in the numerical solution of partial differential equations. In particular, they have been applied to problems involving wave propagation over long distances. Within the domain of aeronautics, they have primarily been used for time-dependent problems such as electromagnetics [30] and aeroacoustics [62]. In these disciplines, the grid resolution requirements of second-order methods can become excessive, leading to impractical CPU and memory requirements. Zingg [67] reviews a number of higher-order and optimized finite-difference methods for numerically simulating the propagation and scattering of linear waves.

Another area where higher-order schemes are necessary to make the computation feasible is the simulation of transition and turbulence [44, 45, 65]. Direct Numerical

Simulation (DNS) of turbulent flow is often not practical in terms of computational effort and hardware requirements. An alternate approach that is less computationally intensive is Large Eddy Simulation (LES). Ghosal [22], and Kravchenko and Moin [31] provide a detailed analysis on the effect of numerical error on the accuracy and robustness of LES of turbulent flows. The results illustrate the necessity of using higher-order methods for these types of simulations. In LES, the full turbulent field is divided into a set of large-scale or "resolved" eddies and the small-scale or "subgrid" eddies. Only the resolved eddies are computed directly while the net effect of the large number of subgrid eddies are approximated by a single subgrid model. For accurate simulations, the numerical errors associated with the large-scale model should be small compared to the subgrid model. Both references demonstrate that, for the cases examined, the truncation errors associated with a second-order scheme in the large-scale model, are significantly larger than the subgrid term over a wide range of wavenumbers. Higher-order schemes are therefore necessary to accurately resolve the wide range of length scales of turbulence on practical grids.

In the disciplines discussed thus far, the use of higher-order methods remains an active area of research. The application of higher-order methods to steady aerodynamic flows, specifically the solution of the steady Reynolds-averaged Navier-Stokes (RANS) equations, has been more limited. Initially, many RANS solvers used the scalar artificial dissipation scheme presented in [28] to provide the numerical dissipation needed for stability. The scalar dissipation scheme, however, has been shown by numerous authors [3, 20] to be excessively dissipative in slow moving regions of flow for high Reynolds numbers. This results in contaminated boundary layers and overprediction of drag. There is no point in using a higher-order discretization as long as the scalar dissipation scheme is used. The development of upwind schemes [49] and matrix artificial dissipation [54] was thus critical to the successful implementation of higher-order methods. It was shown in [68] that the numerical error introduced using matrix dissipation is generally less than the truncation error from a second-order centered difference operator. With the implementation of these sophisticated numerical dissipation schemes, the leading source of numerical error became the discretization of the convective and diffusive fluxes. To reduce this leading source of error, researchers began implementing higher-order spatial schemes.

The discretization of the inviscid or convective terms for the RANS equations has received considerable attention over the last decade. It is common to combine a high-order treatment for the inviscid flux terms with a second-order approximation for the viscous fluxes. The higher-order treatment often consists of a third-order upwind-biased scheme. Examples of higher-order upwind schemes used in conjunction with second-order viscous approximations on structured grids can found in [50, 23, 18, 61, 29].

Compared to standard explicit finite differences, compact schemes offer the advantage of using smaller stencil sizes to obtain a comparable order of accuracy [33]. Tolstykh and Lipavski [58] use third- and fifth-order compact upwind differencing for the solution of Burgers equation and the 2D compressible NS equations. They combine the high-order spatial scheme with GMRES [40] to aid convergence to steady-state. Although Tolstykh and Lipavski use third-order approximations of the viscous terms in the solution of Burgers equation and demonstrate the benefits of doing so, they elect to use only second-order differencing when solving the Navier-Stokes equations. Mahesh [34] introduces a compact scheme applicable to the solution of the Navier-Stokes equations in which both first- and second-order derivatives are solved simultaneously. The intent is to solve the inviscid terms, consisting of first-order derivatives, and the viscous terms, consisting of second-order derivatives, simultaneously. This new scheme was compared to the standard Padé scheme for efficiency and accuracy using Fourier analysis. Unfortunately, Mahesh only illustrates how the spatial scheme can be applied to the NS equations but does not show any solutions.

Yee [63] formulates a fourth- and sixth-order compact scheme based on the work of Abarbanel and Kumar [1]. Compact schemes tend to exhibit better spectral resolution compared to their non-compact cousins. They involve, however, a tridiagonal matrix inversion which increases the operational count per node. Abarbanel and Kumar [1] proposed a spatially fourth-order compact scheme without the associated matrix inversion. Numerical experiments showed that their scheme exhibits poor shock resolution even with added linear numerical dissipation. In [63], Yee modifies the Abarbanel-Kumar compact scheme to be high-resolution at discontinuities and extends the scheme to a larger class of explicit and implicit high-resolution schemes. Yee [63] also states that formal extension of the new schemes to include viscous terms while maintaining the same order of accuracy is quite involved and computationally expensive. Yee suggests the option of using standard non-compact second- or fourth-order central differencing. Doing so, Yee [63] adds, raises the question as to the effect of the inconsistent discretization of the equations on the overall performance and accuracy of the final scheme. More recent work where Yee and colleagues examine higher-order compact spatial algorithms in the context of TVD and ENO type schemes, including higher-order approximations for the viscous terms, can be found in [64]. The application of the higher-order viscous terms are, however, applied to DNS simulations and not the type of aerodynamic flows examined in this thesis.

In [19], Ekaterinaris presents a fourth-order accurate compact spatial discretization for the Euler equations. Although we have restricted ourselves to the NS or RANS equations thus far, this reference is included here because the higher-order scheme is applied to the diagonal form [43] of the implicit ADI method of Beam-Warming [8]. The work presented in this thesis also employs the diagonalized Beam-Warming factorization, and a comparison between the two schemes in future work might prove useful. Also, unlike many researchers using implicit time-marching methods, Ekaterinaris shows how to obtain fourth-order accuracy for the implicit operators. It is common to retain low-order accuracy for the implicit operators for simplicity and computational efficiency. But this practice of improper linearization of the discretized equations can negatively affect convergence rates to steady-state.

It is apparent that much of the research on higher-order schemes in the literature concentrates on the application of such schemes to the convective terms of the NS equations. Very little attention is given to the viscous or turbulence terms. Researchers seem content to improve the spatial accuracy of the inviscid terms of a solver while using a low-ordered approximation to the viscous terms. The assumption made is that the error introduced in the differencing of the viscous fluxes is small. All the references presented thus far demonstrate the significant benefits offered by higher-order methods in terms of accuracy and efficiency. However, they do not address or attempt to quantify the penalty of not improving the viscous flux approximation. An error in the viscous flux approximation within the boundary layer, where the flow is dominated by a balance between the viscous and inviscid fluxes in the streamwise momentum equation, can lead to large errors in the prediction of drag.

Early evidence of an attempt to use a higher-order treatment of the viscous terms for steady flow can be found in [7], where the case of a supersonic boundary layer over a flat plate was examined. Hayder et al. [26] apply the same spatial algorithm for subsonic flow over a flat plate. They compare two spatial schemes. The first consists of fourth-order accurate approximations for the inviscid terms and second-order for the viscous terms. The second scheme was uniformly fourth-order accurate for both the inviscid and viscous terms. Boundary layer profiles illustrate the marked improvement that comes from using a fourth-order accurate treatment of the viscous terms.

Sjogreen [52] and Treidler and Childs [59] use a higher-order treatment for both inviscid and viscous terms on structured grids for the solution of laminar flow past a cylinder. Sjogreen [52] examines supersonic flow at low Reynolds numbers. Second- and fourth-order schemes are evaluated by performing grid convergence studies comparing surface skin-friction distributions. Accurate solutions were obtained on relatively coarse grids (65x33) using the fourth-order scheme. In areas where the boundary layer remained fully attached, the second-order scheme required 4 times as many nodes to obtain similar accuracy. There were some limitations, however, to the higher-order centered difference schemes used in [52]. They could not be used for discontinuous solutions. Hence the outer boundary of the grid was fit to the bow shock by using the Rankine-Hugoniot condition. Furthermore, scalar dissipation was used, which may have limited the potential of the higher-order scheme.

Treidler and Childs [59] examine subsonic flow about a cylinder and also perform grid convergence studies comparing various ordered spatial schemes implemented in OVERFLOW [29], a well known compressible Navier-Stokes flow solver developed at NASA Ames research center. The results indicate that examining total drag on each grid can be misleading. Skin-friction and drag due to pressure can asymptote to the grid independent value from opposite sides. Hence, numerical error in each of these terms can cancel each other out presenting a far more positive picture than may actually exist. The friction drag results computed using the higher-order discretization described in [59] did not show significant improvement over the second-order solver. The authors indicated that further study is necessary to isolate the cause of this behaviour. The authors added that there was some concern as to the accuracy of force and moment integration and that more accurate post-processing might be necessary.

Visbal and Gaitonde [60] present both qualitative and quantitative analysis for laminar, incompressible, subsonic flow past a flat plate and cylinder. Centered compact schemes of up to sixth-order order are developed with filtering schemes of up to tenth-order. As with previous authors, Visbal and Gaitonde [60] demonstrate the reduced grid density requirements when using high-order spatial schemes. The second-order scheme used in most of their comparisons with the higher-order methods, however, used scalar artificial dissipation which is likely the major source of error presented in the results. When the second-order method was combined with a fourth-order filter (instead of damping) in one experiment for the flat-plate solution, its accuracy improved substantially. For the unsteady laminar flow past a cylinder, Visbal and Gaitonde [60] compare results on two grids for the various spatial schemes and compare the Strouhal number and maximum lift and drag coefficients. Individual drag components were not provided, making it difficult to determine the true benefit of the higher-order viscous terms. Further qualitative analysis was performed in 3D for the unsteady laminar simulation of spiral vortex breakdown above a slender delta wing. Compared to the second-order scheme, the sixth-order scheme was better able to resolve the complex flow structures inherent in this flow. The sixth-order scheme was 1.9 - 2.4 times more computationally expensive than the second-order scheme depending on the details of the iterative solver. The authors make a conservative estimate that the number of mesh points required in each coordinate direction can be reduced by a factor of two. Consequently, for the type of 3D flow examined in [60], the required memory and CPU resources can be reduced by factors of at least eight and four, respectively.

Much of the analysis presented thus far demonstrating the efficiency of higher-order methods for solving the NS equations is based on laminar flow conditions with small free-stream Mach numbers. In the solution of the RANS equations in conjunction with turbulence modeling for compressible subsonic or transonic flow conditions, the case is not as clear. Given the complexities of high-Reynolds number external flows about single- or multi-element airfoils, the use of higher-order methods may provide significant benefits. The flow around multi-element airfoils is generally more complex than for single-element geometries. Adequately resolving separated regions and confluent boundary layers often requires large grid densities. Nelson et al. [38] demonstrated that second-order multi-block solutions for a 3-element airfoil are grid-dependent, even with grid densities of over 100,000 points. New algorithms are needed to reduce the number of nodes required to achieve sufficiently grid independent results, which would in turn reduce the CPU and memory requirements. In combination with modern convergence acceleration techniques, higher-order methods appear to be the next step towards achieving this goal.

Published research using higher-order methods with turbulence models for solving external turbulent flow over single- or multi-element airfoils is limited. One example can be found in Rangwalla and Rai [46]. They present a fourth-order finite-difference scheme for solving the compressible thin-layer Navier-Stokes equations on grids having multiple zones. They examine subsonic flow through an experimental turbine stage and compare the fourth-order results with a standard third-order accurate upwind-biased method. The third-order method was combined with second-order approximations for the viscous terms. The Baldwin-Lomax [4] turbulence model was used with both spatial schemes. As in [60], pressure and entropy contours illustrate the capability of the fourth-order treatment to resolve complex flow physics between elements and the vortices shed from the trailing edges. The third-order method with second-order viscous approximations was unable to resolve and track the small scale flow features given the grid density used.

Although the benefits of the fourth-order method in [46] were significant, a number of issues were not quantified. The authors present a fourth-order scheme for the viscous approximations but indicate that second-order approximations are used at the surface and zonal boundary points. It is unclear as to how this affects the accuracy of the solution in those areas. There is no indication as to the spatial accuracy of the turbulence model or computation of vorticity. These quantities can also affect the accuracy of the viscous approximations. Much of the data presented in [46] are related to pressure. There are no data presented with respect to viscous related items like skin-friction or boundary-layer profiles. Such data are vital to aid in determining and quantifying the benefits of using higher-order approximations for the viscous terms.

This literature review briefly summarizes the development of higher-order spatial schemes for the NS or RANS equations. Although the application of higher-order methods to DNS or laminar external flows has received considerable attention in recent years, there still exists no clear demonstration that higher-order methods are more efficient than conventional second-order methods in the computation of practical turbulent aerodynamic flows.

1.2 Objectives

The objective of this thesis is to develop a spatial discretization consistent with third-order accuracy, and to provide a clear demonstration as to the efficiency of this algorithm compared to a standard second-order scheme. We wish to demonstrate the improved accuracy of the new higher-order algorithm on grids of practical density. In order realize the full potential of the higher-order method, *every* aspect of the spatial discretization must be addressed and raised to a suitable level of accuracy. This includes:

- inviscid fluxes, including artificial dissipation or filtering,
- metrics of the curvilinear coordinate transformation,
- viscous fluxes,
- convective and diffusive fluxes in the turbulence model,
- near-boundary operators,
- extrapolation at boundaries,
- interpolation at zonal interfaces,

- integration for force and moment calculations.

Although it is generally fairly straightforward to increase the accuracy of the basic flux derivatives, some of the other components can be more problematic. In particular, high-order numerical boundary schemes can cause instabilities [70]. Transonic flows with shocks introduce an additional degree of difficulty in that first-order numerical dissipation is typically added near shocks, which can potentially undermine the benefits of a higher-order method [9]. Grid convergence studies [66, 47] are used to compare the accuracy of the higher-order discretization with a well-established second-order discretization.

Chapter 2

Governing Equations

In this chapter the solution method for solving the Navier-Stokes equations is presented. The higher-order spatial scheme is developed and implemented in CYCLONE [14], which is based on the thin-layer Navier-Stokes solver ARC2D [41]. This solver uses a generalized curvilinear coordinate transformation and is thus applicable to structured grids. The new spatial algorithm is also implemented in TORNADO [17], an extension of CYCLONE to multi-block grids. TORNADO is used to model complex flow around multi-element airfoils. The governing equations are presented in Section 2.1. The thin-layer approximation is described in Section 2.2, and the Baldwin-Lomax and Spalart-Allmaras turbulence models are outlined in Section 2.3. A description of the boundary conditions follows in Section 2.4.

2.1 Navier-Stokes Equations

The governing equations for aerodynamic flows are the Navier-Stokes equations. In two-dimensional form for Cartesian coordinates (x, y), the equations can be written as

$$\frac{\partial Q}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} = \mathcal{R}e^{-1}\left(\frac{\partial E_v}{\partial x} + \frac{\partial F_v}{\partial y}\right) \tag{2.1}$$

where

$$Q = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{bmatrix} \tag{2.2}$$

contains the conservative variables. Here we scale the dimensional variables, Cartesian coordinates $(\tilde{x}, \tilde{y})$, density $(\tilde{\rho})$, velocity $(\tilde{u}, \tilde{v})$, total energy $(\tilde{e})$, and time $(\tilde{t})$, as

$$x = \frac{\tilde{x}}{c}, \quad y = \frac{\tilde{y}}{c}, \quad \rho = \frac{\tilde{\rho}}{\tilde{\rho}_\infty}, \quad u = \frac{\tilde{u}}{\tilde{a}_\infty}, \quad v = \frac{\tilde{v}}{\tilde{a}_\infty}, \quad e = \frac{\tilde{e}}{\tilde{\rho}_\infty \tilde{a}_\infty^2}, \quad t = \frac{\tilde{t}\, a_\infty}{c} \tag{2.3}$$

where ∞ refers to free-stream quantities, c is the chord length, and a is the speed of sound, which for ideal fluids is $a = \sqrt{\gamma p/\rho}$. The ratio of specific heats, γ, is taken as 1.4 for air. Pressure, p, is related to the conservative flow variables, Q, by the equation of state for a perfect gas, as follows

$$p = (\gamma - 1)\left(e - \frac{1}{2}\rho(u^2 + v^2)\right) \tag{2.4}$$

Referring to Equation 2.1, the convective and viscous flux vectors are

$$E = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(e+p) \end{bmatrix}, \quad F = \begin{bmatrix} \rho v \\ \rho vu \\ \rho v^2 + p \\ v(e+p) \end{bmatrix} \tag{2.5}$$

and

$$E_v = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \varphi_1 \end{bmatrix}, \quad F_v = \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \varphi_2 \end{bmatrix} \tag{2.6}$$

respectively, with

$$\begin{aligned}
\tau_{xx} &= (\mu+\mu_t)(4u_x-2v_y)/3 \\
\tau_{xy} &= (\mu+\mu_t)(u_y-v_x) \\
\tau_{yy} &= (\mu+\mu_t)(-2u_x+4v_y)/3 \\
\varphi_1 &= u\tau_{xx}+v\tau_{xy}+(\mu\mathcal{P}r^{-1}+\mu_t\mathcal{P}r_t^{-1})(\gamma-1)^{-1}\partial_x(a^2) \\
\varphi_2 &= u\tau_{xy}+v\tau_{yy}+(\mu\mathcal{P}r^{-1}+\mu_t\mathcal{P}r_t^{-1})(\gamma-1)^{-1}\partial_y(a^2)
\end{aligned} \tag{2.7}$$

where $\mu = \tilde{\mu}/\tilde{\mu}_\infty$ is the non-dimensional dynamic viscosity, μ_t is the non-dimensional turbulent eddy viscosity, $\mathcal{R}e$ is the Reynolds number, $\mathcal{P}r$ is the Prandtl number and $\mathcal{P}r_t$ is the turbulent Prandtl number. The Prandtl number is defined by

$$\mathcal{P}r = \frac{c_p\mu}{\kappa_t} \tag{2.8}$$

where κ_t is the thermal conductivity and c_p the specific heat at constant pressure. The Prandtl number is considered constant in this study and is set to $\mathcal{P}r = 0.72$ and $\mathcal{P}r_t = 0.90$. Using the chord of the airfoil, c, as the reference length, we define the Reynolds number as

$$\mathcal{R}e = \frac{\rho_\infty\, c\, a_\infty}{\mu_\infty}. \tag{2.9}$$

2.2 Thin-Layer Navier-Stokes Equations

For the aerodynamic flows studied in this work, namely high Reynolds number viscous flows, the effects of viscosity are concentrated near the airfoil surface and in wake regions. Typically, the viscous derivatives in the streamwise direction are neglected. This leads to the thin-layer approximation of the NS equations. The rationale behind this approximation is that for attached and mildly separated flows, the gradients of the streamwise diffusion terms are small compared to the normal gradients.

Converting to curvilinear coordinates (ξ, η) [41], and dropping all the viscous derivatives in the ξ direction, we arrive at the thin-layer Navier-Stokes equations for a curvilinear coordinate system as follows (see Figure 2.1):

$$\frac{\partial \hat{Q}}{\partial t} + \frac{\partial \hat{E}}{\partial \xi} + \frac{\partial \hat{F}}{\partial \eta} = \mathcal{R}e^{-1}\frac{\partial \hat{S}}{\partial \eta} \tag{2.10}$$

where,

$$\hat{Q} = J^{-1}\begin{bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{bmatrix} \tag{2.11}$$

The convective flux vectors are

$$\hat{E} = J^{-1}\begin{bmatrix} \rho U \\ \rho U u + \xi_x p \\ \rho U v + \xi_y p \\ (e+p)U - \xi_t p \end{bmatrix}, \quad \hat{F} = J^{-1}\begin{bmatrix} \rho V \\ \rho V u + \eta_x p \\ \rho V v + \eta_y p \\ (e+p)V - \eta_t p \end{bmatrix} \tag{2.12}$$

with

$$U = \xi_t + \xi_x u + \xi_y v, \quad V = \eta_t + \eta_x u + \eta_y v \tag{2.13}$$

the contravariant velocities. The variable J represents the metric Jacobian of the transformation:

$$J^{-1} = (x_\xi y_\eta - x_\eta y_\xi) \tag{2.14}$$

The viscous flux vector is

$$\hat{S} = J^{-1}\begin{bmatrix} 0 \\ \eta_x m_1 + \eta_y m_2 \\ \eta_x m_2 + \eta_y m_3 \\ \eta_x(u m_1 + v m_3 + m_4) + \eta_y(u m_2 + v m_3 + m_5) \end{bmatrix} \tag{2.15}$$

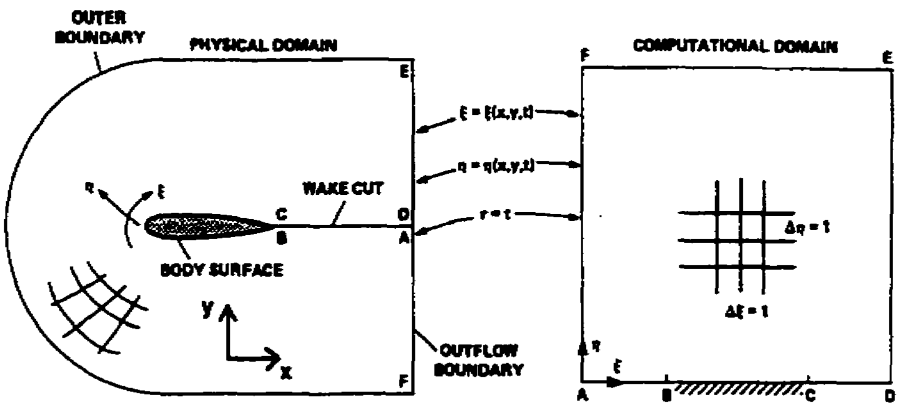


Figure 2.1: Curvilinear coordinate transformation (used with permission from T. H. Pulliam)

with

$$
\begin{aligned}
m_1 &= (\mu + \mu_t)(4\eta_x u_\eta - 2\eta_y v_\eta)/3 \\
m_2 &= (\mu + \mu_t)(\eta_y u_\eta + \eta_x v_\eta) \\
m_3 &= (\mu + \mu_t)(-2\eta_x u_\eta + 4\eta_y v_\eta)/3 \\
m_4 &= (\mu Pr^{-1} + \mu_t Pr_t^{-1})(\gamma - 1)^{-1}\eta_x \partial_\eta(a^2) \\
m_5 &= (\mu Pr^{-1} + \mu_t Pr_t^{-1})(\gamma - 1)^{-1}\eta_y \partial_\eta(a^2)
\end{aligned}
\tag{2.16}
$$

2.3 Turbulence Models

The effects of turbulence can be approximated by adding an eddy viscosity term, μ_t, to the dynamic viscosity μ in the fashion shown in Eqs. 2.7 and 2.16. In our study, we use the algebraic Baldwin-Lomax [4] model and the one-equation Spalart-Allmaras [53] model to compute μ_t.

2.3.1 Baldwin-Lomax Turbulence Model

Currently, the Baldwin-Lomax turbulence model is only implemented in CYCLONE. It is inadequate for high-lift multi-element flow computations. The flow may contain confluent boundary layers, large separated regions, and separated wakes, none of which can be treated properly with algebraic models. Hence, it is not used in TORNADO. This model is, however, quick and robust for single-element computations and provides sufficiently accurate results for attached and mildly separated flows. In the Baldwin-Lomax model, the boundary layer is divided into two layers, an outer and inner layer. The eddy viscosity in the two layers is given by

$$\mu_t = \begin{cases} (\mu_t)_{inner} & y \leq y_{crossover} \\ (\mu_t)_{outer} & y > y_{crossover} \end{cases} \tag{2.17}$$

where y is the normal distance from the wall and $y_{crossover}$ is the smallest value of y at which values from the inner and outer formulas are equal.

For the inner region, the Prandtl-Van Driest formulation is used:

$$(\mu_t)_{inner} = \rho\, l^2\, |\omega| \tag{2.18}$$

$$l = k\, y\, [1 - e^{-(y^+/A^+)}] \tag{2.19}$$

where k and A^+ are constants, $|\omega|$ is the magnitude of the vorticity:

$$|\omega| = \frac{\partial u}{\partial y} - \frac{\partial v}{\partial x} \tag{2.20}$$

and the law-of-the-wall coordinate y^+ is given by

$$y^+ = \frac{\rho_w\, u_\tau\, y}{\mu_w} = \frac{\sqrt{\rho_w\, \tau_w}\, y}{\mu_w} \tag{2.21}$$

The subscript w denotes values at the wall, $u_\tau = \sqrt{\tau_w/\rho_w}$ is the friction velocity, and τ_w is the shear stress at the wall.

In the outer region, μ_t takes the following form:

$$(\mu_t)_{outer} = K\, C_{cp}\, \rho\, F_{wake}\, F_{kleb}(y) \tag{2.22}$$

where K is the Clauser constant, C_{cp} is an additional constant, and

$$F_{wake} = min \begin{cases} y_{max}\, F_{max} \\ C_{wk}\, y_{max}\, u_{diff}^2 / F_{max} \end{cases} \tag{2.23}$$

The quantities F_{max} and y_{max} are determined from the function

$$F(y) = y\,|\omega|\,[1 - e^{-(y^+/A^+)}] \tag{2.24}$$

In wakes, the exponential term of Equation 2.24 is set equal to zero. F_{max} is the maximum value of $F(y)$ in a profile and y_{max} is the value of y that satisfies $F(y) = F_{max}$. The function $F_{kleb}(y)$ is the Klebanoff intermittency factor given by

$$F_{kleb}(y) = \left[1 + 5.5\left(\frac{y\,C_{kleb}}{y_{max}}\right)^6\right]^{-1} \tag{2.25}$$

The quantity u_{diff} is the difference between maximum and minimum total velocity in the profile and is given by

$$u_{diff} = \begin{cases} (\sqrt{u^2+v^2})_{max} & \text{in boundary layers} \\ (\sqrt{u^2+v^2})_{max} - (\sqrt{u^2+v^2})_{min} & \text{in wakes} \end{cases} \tag{2.26}$$

The Baldwin-Lomax turbulence model is patterned after that of Cebeci [10]. Requiring agreement with the Cebeci formulation for constant pressure boundary layers at transonic speeds leads to the following values for the constants used in the above equations:

$$A^+ = 26, \quad C_{cp} = 1.6, \quad C_{kleb} = 0.3$$

$$C_{wk} = 0.25, \quad k = 0.4, \quad K = 0.0168$$

2.3.2 Spalart-Allmaras Turbulence Model

The Spalart-Allmaras turbulence model is a one-equation transport model written in terms of the eddy-viscosity-like term $\tilde{\nu}$. The equation is

$$\begin{aligned} \frac{D\tilde{\nu}}{Dt} = c_{b1}\,[1 - f_{t2}]\,\tilde{S}\tilde{\nu} + \frac{1}{\sigma}[\nabla\cdot((\nu+\tilde{\nu})\,\nabla\tilde{\nu})] + c_{b2}\,(\nabla\tilde{\nu})^2 \\ - \left[c_{w1}f_w - \frac{c_{b1}}{\kappa^2}f_{t2}\right]\left(\frac{\tilde{\nu}}{d}\right)^2 + f_{t1}\Delta U^2 \end{aligned} \tag{2.27}$$

The kinematic eddy viscosity, ν_t is related to the eddy viscosity term $\tilde{\nu}$ through the equation

$$\nu_t = \tilde{\nu} f_{v1} \tag{2.28}$$

where

$$f_{v1} = \frac{\chi^3}{\chi^3 + c_{v1}^3} \tag{2.29}$$

and

$$\chi = \frac{\tilde{\nu}}{\nu} \tag{2.30}$$

The production term $\tilde{S}$ in the differential equation is given by

$$\tilde{S} = S + \frac{\tilde{\nu}}{\kappa^2 d^2} f_{v2} \tag{2.31}$$

where S is the magnitude of the vorticity, d is the distance to the wall and

$$f_{v2} = 1 - \frac{\chi}{1 + \chi f_{v1}} \tag{2.32}$$

The destruction function f_w is given by

$$f_w = g \left[\frac{1 + c_{w3}^3}{g^6 + c_{w3}^6} \right]^{\frac{1}{6}} \tag{2.33}$$

where

$$g = r + c_{w2}(r^6 - r) \tag{2.34}$$

and

$$r = \frac{\tilde{\nu}}{\tilde{S}\kappa^2 d^2} \tag{2.35}$$

Transition is included using a trip function. The transition functions are

$$f_{t1} = c_{t1} g_t \exp(-c_{t2} \frac{\omega_t^2}{\Delta U^2} [d^2 + g_t^2 d_t^2]) \tag{2.36}$$

$$f_{t2} = c_{t3} \exp(-c_{t4} \chi^2) \tag{2.37}$$

where

$$g_t = \min(0.1, \frac{\Delta U}{\omega_t \Delta x_t}) \tag{2.38}$$

In the transition functions, d_t is the distance to the trip, ω_t is the vorticity at the trip, and Δx_t is the grid spacing at the trip. The velocity difference between a field point and the trip is ΔU. The constants used for the Spalart-Allmaras model are

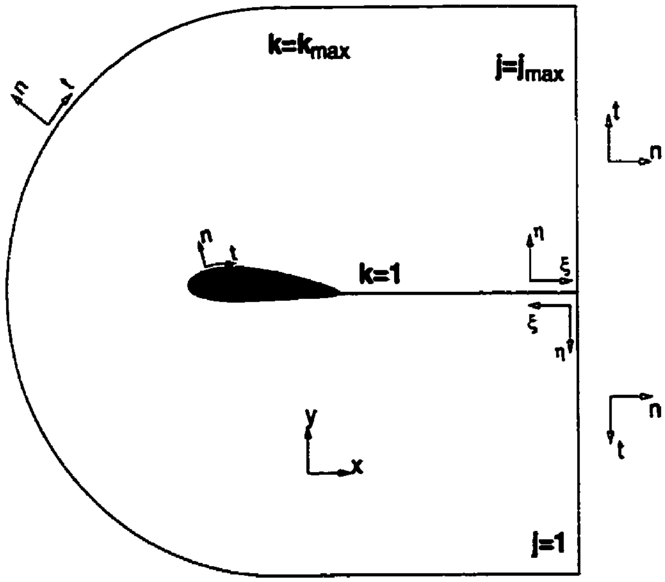
$$
\begin{array}{ll}
c_{b1} = 0.1355 & c_{b2} = 0.622 \\
c_{t1} = 5.0 & c_{t2} = 2.0 \\
c_{t3} = 1.2 & c_{t4} = 0.5 \\
c_{w1} = c_{b1}/\kappa^2 + (1 + c_{b2})/\sigma & \\
c_{w2} = 0.3 & c_{w3} = 2.0 \\
c_{v1} = 7.1 & \\
\sigma = \frac{2}{3} & \kappa = 0.41
\end{array}
$$

For multi-element airfoils there is a trip on the upper and lower surface of each element so a point in the field could refer to more than one trip. In this instance the closest trip of those on the correct surface of each airfoil is used. Since the effect of the trip is very localized, two trips are never close enough to cause a significant effect on the same field point. There are many subtleties regarding the implementation of this model in a multi-block/multi-element context. Nelson [37] describes these implementation issues for the Baldwin-Barth [5] turbulence model. The Baldwin-Barth model is also a one-equation transport model and is implemented in much the same manner as the Spalart-Allmaras model.

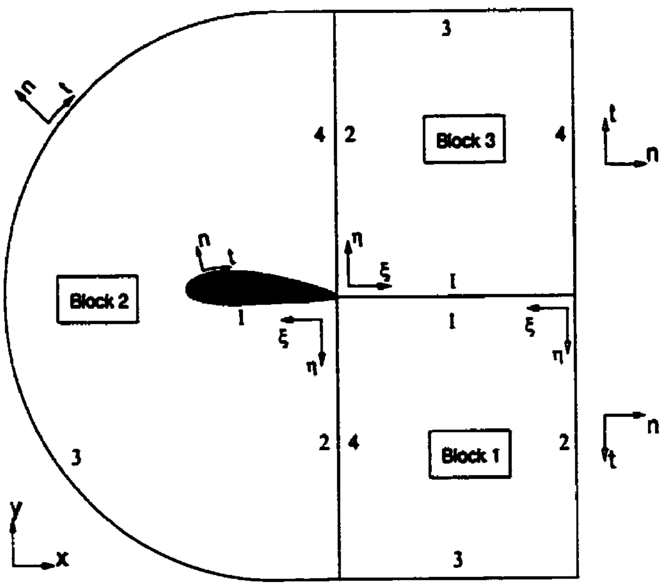
2.4 Boundary Conditions

The interior numerical scheme used in CYCLONE and TORNADO requires four boundary conditions to be specified at each domain boundary. Physical boundary conditions provide some of the necessary equations. Numerical boundary conditions, normally obtained through extrapolation from the interior, are employed to make up the balance. In this section, the boundary conditions are outlined, while the corresponding spatial schemes are discussed in Section 3.2.

Figure 2.1 illustrates the computational domain for an external flow around a single-element airfoil. There are three boundaries to address: body surface boundaries, far-field boundaries, and the wake-cut. All boundaries are treated explicitly in both CYCLONE and TORNADO, although there is an option in CYCLONE to



(a) CYCLONE.



(b) TORNADO.

Figure 2.2: Normal and tangential directions at the boundaries.

treat the wake-cut implicitly if the turbulence model is set up to do the same.

The normal and tangent directions at the boundaries are defined in a similar manner in both CYCLONE and TORNADO for single-element C-grid topologies as shown in Figure 2.2. In Figure 2.2(b) the solution domain is divided into three blocks and each block side is numbered 1 - 4. The normal at the farfield boundaries is defined to point out of the domain for both solvers.

2.4.1 Body Surface

Inviscid Flow

At the body surface (between points B and C in Figure 2.1), flow tangency must be satisfied for inviscid flows. The normal component of velocity is set to zero and the tangential components are linearly extrapolated. The normal and tangential velocities at the body are given by

$$V_n = \frac{\eta_x u + \eta_y v}{\sqrt{\eta_x^2 + \eta_y^2}} \tag{2.39}$$

$$V_t = \frac{\eta_y u - \eta_x v}{\sqrt{\eta_x^2 + \eta_y^2}} \tag{2.40}$$

The pressure at the surface is extrapolated from the interior. The density is determined from the equation for free-stream stagnation enthalpy, H_∞, which is held constant at the surface:

$$H_\infty = \frac{e_\infty + p_\infty}{\rho_\infty} = \frac{e_{surf} + p_{surf}}{\rho_{surf}} \tag{2.41}$$

Using Equation 2.4 and substituting for the energy variable in Equation 2.41, one obtains the expression for density on the body surface:

$$\rho_{surf} = \frac{\gamma\, p_{surf}}{(\gamma - 1)(H_\infty - \frac{1}{2}(u_{surf}^2 + v_{surf}^2))} \tag{2.42}$$

Viscous Flow

For viscous flow, no-slip conditions are applied on the body surface. This condition leads to the first two equations, velocities $u = v = 0$. Pressure can be obtained by extrapolation from the interior or a Neumann condition can be used where the normal gradient of pressure at the wall is set to zero:

$$\frac{\partial p}{\partial \eta} = 0 \tag{2.43}$$

We prefer to use the Neumann condition as it is found to be more robust for higher-order spatial stencils. Note that the assumption that $\partial p/\partial \eta = 0$ is not strictly correct. However, for aerodynamic flows at high Reynolds numbers, the error introduced is very small. We have experimented with both extrapolation of pressure and the Neumann condition with no significant change in the solution. Furthermore, the error introduced is a physical error, not a discretization error [48], and hence does not affect the conclusions from grid convergence studies. Further discussion on this issue can be found in Section 3.2.5. Density can be determined from either adiabatic or isothermal conditions at the surface. We use adiabatic conditions in all calculations. For an adiabatic wall, when coupled with the assumption of zero pressure gradient and the perfect gas law, density also satisfies a zero normal gradient at the wall:

$$\frac{\partial \rho}{\partial \eta} = 0 \tag{2.44}$$

2.4.2 Far-field Boundaries

Inviscid - CYCLONE

The value of $\hat{Q}$ at the boundary node is calculated as follows:

$$\begin{aligned} \hat{Q}_{bc} &= \frac{1}{2}(\hat{Q}_{\infty} + \hat{Q}_{ext}) - \frac{1}{2}\mathrm{sign}(\Psi)(\hat{Q}_{\infty} - \hat{Q}_{ext}) \\ \mathrm{sign}(\Psi) &= T_{\kappa}\ \mathrm{sign}(\Lambda_{\kappa})\ T_{\kappa}^{-1} \end{aligned} \tag{2.45}$$

where κ is chosen in the direction normal to the boundary, and Ψ represents the corresponding flux Jacobian as follows:

$$\text{for } k = k_{max} \qquad \left\{ \begin{array}{l} \Psi = \hat{B} \\ \kappa = \eta \end{array} \right. ,$$

$$\text{for } j = 1 \text{ or } j = j_{max} \quad \left\{ \begin{array}{l} \Psi = \hat{A} \\ \kappa = \xi \end{array} \right. ,$$

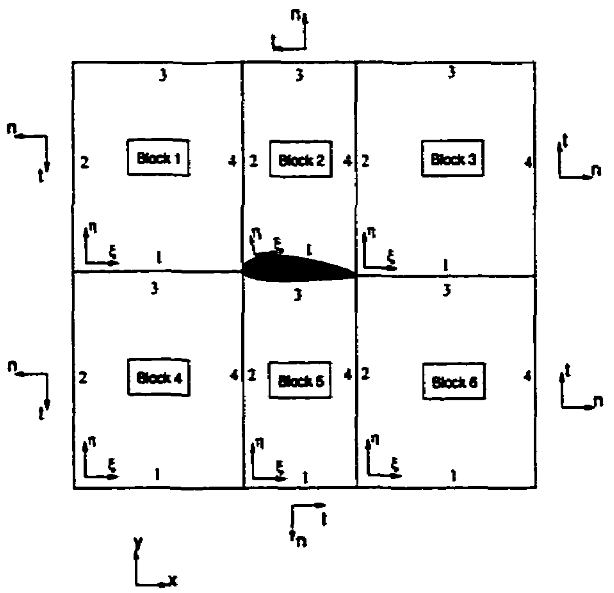


Figure 2.3: Normal and tangential directions at the boundaries for H-grid topologies.

In Equation 2.45, *bc* indicates the boundary value, ∞ indicates values obtained from free-stream conditions, and *ext* indicates values extrapolated from the interior nodes of the mesh. Λ is a vector containing the eigenvalues of the flux Jacobian matrix $\hat{A} = \partial\hat{E}/\partial\hat{Q}$ or $\hat{B} = \partial\hat{F}/\partial\hat{Q}$. The matrix T contains the eigenvectors of Ψ. These variables will be derived in detail in Chapter 3. The eigenvalues and eigenvectors are calculated from the mean state, $\hat{Q}_{avg} = \frac{1}{2}(\hat{Q}_\infty + \hat{Q}_{ext})$.

Inviscid - TORNADO

Before proceeding, it is prudent to expand on the normal and tangent definitions outlined in Figure 2.2(b). For the majority of cases, TORNADO is used to solve flows on H-grid topologies. Consistent with the normal and tangent directions defined in Figure 2.2(b) for a multi-block C-grid, Figure 2.3 illustrates the directions defined for an H-grid topology. Here, the normal and tangential velocities, depending on the side forming the far-field boundary, are as follows:

$$V_n = \phi(\tilde{\kappa_x}u + \tilde{\kappa_y}v) \begin{cases} \phi = 1 & \text{sides 3 and 4} \\ \phi = -1 & \text{sides 1 and 2} \end{cases} \tag{2.47}$$

$$V_t = \phi(\tilde{\kappa_y} u - \tilde{\kappa_x} v) \begin{cases} \phi = 1 & \text{sides 1 and 4} \\ \phi = -1 & \text{sides 2 and 3} \end{cases} \tag{2.48}$$

The metric terms are defined by

$$\tilde{\kappa_x} = \begin{cases} \dfrac{\xi_x}{\sqrt{\xi_x^2 + \xi_y^2}} & \text{sides 2 and 4} \\ \dfrac{\eta_x}{\sqrt{\eta_x^2 + \eta_y^2}} & \text{sides 1 and 3} \end{cases} \quad \text{and} \quad \tilde{\kappa_y} = \begin{cases} \dfrac{\xi_y}{\sqrt{\xi_x^2 + \xi_y^2}} & \text{sides 2 and 4} \\ \dfrac{\eta_y}{\sqrt{\eta_x^2 + \eta_y^2}} & \text{sides 1 and 3} \end{cases} \tag{2.49}$$

The boundary conditions of Equation 2.45 have also been implemented in TORNADO but were not used in this study. Instead, characteristic conditions are used to apply the explicit far-field boundaries. The four variables used are the locally one-dimensional Riemann invariants:

$$R_1 = V_n - \frac{2a}{\gamma - 1} \tag{2.50}$$

$$R_2 = V_n + \frac{2a}{\gamma - 1} \tag{2.51}$$

as well as V_t and a function of entropy:

$$R_3 = V_t \tag{2.52}$$

$$R_4 = \frac{\rho^\gamma}{p} \tag{2.53}$$

These four values are set to free-stream values or they are extrapolated from the interior flow variables depending on the sign of the corresponding characteristic speed. For the Riemann invariants R_1 and R_2, the corresponding characteristic speeds are $\lambda_1 = V_n - a$ and $\lambda_2 = V_n + a$ respectively.

For subsonic inflow, $V_n < 0$, $\lambda_1 < 0$, and $\lambda_2 > 0$ so the Riemann invariant R_1 is determined from free-stream conditions $(u_\infty, v_\infty, a_\infty)$, and R_2 is determined by extrapolation from the interior $(u_{ext}, v_{ext}, a_{ext})$. For subsonic outflow, $V_n > 0$, $\lambda_1 < 0$, and $\lambda_2 > 0$. As in the case for subsonic inflow, R_1 is determined from free-stream conditions and R_2 is extrapolated. For inflow, R_3 and R_4 are set to free-stream conditions. For outflow, they are extrapolated from the interior. Once these four variables on the boundary are calculated, a number of quantities are recalculated using the following relations:

$$V_n = \frac{1}{2}(R_1 + R_2) \tag{2.54}$$

$$a = \frac{1}{4}(\gamma - 1)(R_2 - R_1) \tag{2.55}$$

$$\rho = (\frac{1}{\gamma} a^2 R_4)^{\frac{1}{\gamma-1}} \tag{2.56}$$

$$p = \frac{1}{\gamma} \rho a^2 \tag{2.57}$$

and the velocities are obtained from

$$u = \begin{cases} -\tilde{\kappa_x} V_n + \tilde{\kappa_y} V_t & \text{side 1} \\ -\tilde{\kappa_x} V_n - \tilde{\kappa_y} V_t & \text{side 2} \\ \tilde{\kappa_x} V_n - \tilde{\kappa_y} V_t & \text{side 3} \\ \tilde{\kappa_x} V_n + \tilde{\kappa_y} V_t & \text{side 4} \end{cases} \quad \text{and} \quad v = \begin{cases} -\tilde{\kappa_y} V_n - \tilde{\kappa_x} V_t & \text{side 1} \\ -\tilde{\kappa_y} V_n + \tilde{\kappa_x} V_t & \text{side 2} \\ \tilde{\kappa_y} V_n + \tilde{\kappa_x} V_t & \text{side 3} \\ \tilde{\kappa_y} V_n - \tilde{\kappa_x} V_t & \text{side 4} \end{cases} \tag{2.58}$$

For supersonic inflow and outflow conditions, the reader is referred to [41].

Viscous

A common practice in viscous flow computations is to use simple extrapolation of all variables from the interior at outflow boundaries ($j = 1$ and $j = j_{max}$ in Figure 2.2(a)). The entropy gradients associated with convection of the wake make the characteristic analysis used for inviscid flows inappropriate. Zeroth-order extrapolation of ρ, ρu, ρv and p is often used. This process is equivalent to a first-order approximation to $\partial q/\partial \xi = 0$ where q may represent any of the variables mentioned above and ξ is in the direction normal to the outflow boundary. This approach is used for both CYCLONE and TORNADO. Second-order approximations to the zero normal gradient are used for the higher-order scheme.

2.4.3 Circulation Correction

For lifting bodies, the far-field boundary may affect the solution. To correct for this effect, a far-field circulation correction is applied to the free-stream variables, as described in Appendix A.

Chapter 3

Numerical Method

CYCLONE [14] and TORNADO [17] use implicit time-marching techniques to iterate to steady-state. The time-marching method and other details of these two solvers are described in Section 3.1. Section 3.2 describes the spatial discretization of the new higher-order algorithm, which is consistent with global third-order accuracy.

3.1 Time-Marching Method

Although we are only interested in steady-state solutions for this work, the two solvers are capable of efficiently solving unsteady external flow about an airfoil [13, 15]. Since we are not interested in time accuracy, it is sufficient to use a first-order time-marching method to advance the solution to steady state. The first-order implicit Euler method is used since it has a broad stability region. When the implicit Euler time-marching scheme is applied to Equation 2.10 one obtains

$$\Delta\hat{Q}^n + \Delta t\left(\partial_\xi \hat{E}^{n+1} + \partial_\eta \hat{F}^{n+1} - \mathcal{R}e^{-1}\partial_\eta \hat{S}^{n+1}\right) = 0 \tag{3.1}$$

where Δt is the time step and $\Delta\hat{Q} = \hat{Q}^{n+1} - \hat{Q}^n$ with $\hat{Q}^n = \hat{Q}(n\Delta t)$. The vectors $\hat{E}$, $\hat{F}$, and $\hat{S}$ are locally linearized:

$$\begin{aligned} \hat{E}^{n+1} &= \hat{E}^n + \hat{A}^n\Delta\hat{Q}^n + O(\Delta t^2) \\ \hat{F}^{n+1} &= \hat{F}^n + \hat{B}^n\Delta\hat{Q}^n + O(\Delta t^2) \\ \hat{S}^{n+1} &= \hat{S}^n + \hat{K}^n\Delta\hat{Q}^n + O(\Delta t^2) \end{aligned} \tag{3.2}$$

where the matrices $\hat{A}$, $\hat{B}$, and $\hat{K}$ are the flux Jacobians, defined by

$$\hat{A} = \frac{\partial \hat{E}}{\partial \hat{Q}}, \quad \hat{B} = \frac{\partial \hat{F}}{\partial \hat{Q}}, \quad \text{and} \quad \hat{K} = \frac{\partial \hat{S}}{\partial \hat{Q}}$$

Combining Equations 3.1 and 3.2 one arrives at the following:

$$[I + \Delta t \partial_\xi \hat{A}^n + \Delta t \partial_\eta \hat{B}^n - \Delta t \mathcal{R}e^{-1} \partial_\eta \hat{K}^n] \Delta \hat{Q}^n = \hat{R}^n \tag{3.3}$$

where

$$\hat{R}^n = -\Delta t [\partial_\xi \hat{E}(\hat{Q}^n) + \partial_\eta \hat{F}(\hat{Q}^n) - \mathcal{R}e^{-1} \partial_\eta \hat{S}(\hat{Q}^n)]$$

In general, Equation 3.3 is prohibitively time consuming to solve directly. Various approximations can be made to the implicit operator (left-hand side) in order to reduce the required computational time. The approximately-factored method of Beam and Warming [8] can be applied to Equation 3.3. In combination with spatial differences, the equations take on the following form:

$$[I + \Delta t \delta_\xi \hat{A}^n][I + \Delta t \delta_\eta \hat{B}^n - \Delta t \mathcal{R}e^{-1} \delta_\eta \hat{K}^n] \Delta \hat{Q}^n = \hat{R}^n \tag{3.5}$$

where

$$\hat{R}^n = -\Delta t [\delta_\xi \hat{E}(\hat{Q}^n) + \delta_\eta \hat{F}(\hat{Q}^n) - \mathcal{R}e^{-1} \delta_\eta \hat{S}(\hat{Q}^n)]$$

The symbol δ in Equation 3.5 denotes a spatial operator. Central differences are used for the spatial discretization. Note that central differences require the explicit addition of numerical dissipation as described in Section 3.2.1.

To further reduce the complexity of the left-hand-side, the diagonal form of Pulliam and Chaussee [43] is implemented. The Jacobian matrices are diagonalized as follows:

$$\Lambda_\xi = T_\xi^{-1} \hat{A} T_\xi \tag{3.7}$$

$$\Lambda_\eta = T_\eta^{-1} \hat{B} T_\eta \tag{3.8}$$

where the matrices Λ_ξ and Λ_η are diagonal matrices whose elements are the eigenvalues of the flux Jacobians. The viscous flux Jacobian $\hat{K}$ cannot be simultaneously diagonalized with the flux Jacobian $\hat{B}$, so it has been dropped from the left-hand side.

However, a term approximating the viscous eigenvalues is added to the diagonal of $\hat{B}$, as described by Pulliam [41]. The matrix T_ξ has the eigenvectors of $\hat{A}$ as columns, and T_η has the eigenvectors of $\hat{B}$ as columns. The eigenvector matrices are factored out, giving

$$T_\xi[I + \Delta t\delta_\xi\Lambda_\xi]T_\xi^{-1}T_\eta[I + \Delta t\delta_\eta\Lambda_\eta - \Delta tI\delta_\eta(\lambda_\nu)]T_\eta^{-1}\Delta\hat{Q}^n = \hat{R}^n, \tag{3.9}$$

where λ_ν is the term approximating the viscous eigenvalues and is defined as

$$\lambda_\nu = \frac{\mu}{JRe}(\eta_x^2 + \eta_y^2)\delta_\eta\left(\frac{J}{\rho}\right) \tag{3.10}$$

Variable time stepping is used to accelerate the convergence rate by roughly equalizing the Courant numbers of each cell. Using a spatially varying time step can be effective for grids with widely varying cell dimensions. Such grids are typical in aerodynamic simulations. The Courant number variation can be made more uniform by scaling with the Jacobian:

$$\Delta t = \frac{\Delta t_{ref}}{1 + \sqrt{J}} \tag{3.11}$$

3.2 Spatial Discretization and Force Integration

As stated in Section 1.2, one of the objectives of this thesis is to develop a consistent third-order algorithm. To accomplish this, with the exception of first-order dissipation used near shocks and some terms within the turbulence models, only finite-difference stencils of at least third-order accuracy are used. The only other exceptions are the stencils used near some boundaries. Numerical boundary schemes must be chosen such that they, when combined with the interior scheme, remain stable for a wide variety of flow conditions and preserve the global spatial accuracy of the interior scheme. Gustafsson [25] has shown that numerical boundary schemes can be one order lower than the interior scheme without reducing the global order of accuracy.

Hence we can use second-order numerical boundary schemes while preserving third-order global accuracy. Nevertheless, we use third-order boundary schemes wherever possible. The details of the following items will be addressed in this section:

- numerical dissipation,
- inviscid fluxes,
- metrics of the curvilinear coordinate transformation,
- viscous fluxes,
- convective and diffusive fluxes in the turbulence model,
- near-boundary operators,
- extrapolation at boundaries,
- interpolation at zonal interfaces,
- integration for force and moment calculations.

The following sections describe the new higher-order algorithm.

3.2.1 Numerical Dissipation

In order to maintain stability, numerical dissipation, often referred to as artificial dissipation, must be added to the centered difference scheme used for the convective fluxes. The numerical dissipation is added using the matrix dissipation scheme of Swanson and Turkel [54]. It is implemented in the following manner:*

$$\left(\frac{\partial \hat{E}}{\partial \xi}\right)_{j,k} = \delta_\xi \hat{E}_{j,k} - \Delta_\xi d_{j+\frac{1}{2},k} \tag{3.12}$$

*An analogous term appears in the η direction with

$$
\begin{aligned}
d_{j+\frac{1}{2},k} &= |\hat{A}|_{j+\frac{1}{2},k}\, J^{-1}_{j+\frac{1}{2},k} \left(\epsilon^{(2)}_{j+\frac{1}{2},k} \Delta_\xi\, J_{j,k}\, \hat{Q}_{j,k} - \epsilon^{(4)}_{j+\frac{1}{2},k} \Delta_\xi \nabla_\xi \Delta_\xi\, J_{j,k}\, \hat{Q}_{j,k} \right) \\
\epsilon^{(2)}_{j,k} &= \kappa_2 \max(\Upsilon_{j+1,k}, \Upsilon_{j,k}, \Upsilon_{j-1,k}) \\
\epsilon^{(4)}_{j,k} &= \max(0, \kappa_4 - \epsilon^{(2)}_{j,k}) \\
\Upsilon_{j,k} &= \frac{|p_{j+1,k} - 2p_{j,k} + p_{j-1,k}|}{|p_{j+1,k} + 2p_{j,k} + p_{j-1,k}|}
\end{aligned}
$$

where δ_ξ is a centered difference operator, Δ_ξ and ∇_ξ are first-order forward and backward difference operators, and $\kappa_4 = 0.02$. We use $\kappa_2 = 0$ for subsonic flows, and $\kappa_2 = 1.0$ for transonic flows. The term $\Upsilon_{j,k}$ is a pressure switch to control the use of first-order dissipation near shock waves. The matrix $|\hat{A}|$ is given by

$$|\hat{A}| = T_\xi\, |\Lambda_\xi|\, T_\xi^{-1} \tag{3.13}$$

Here $|\Lambda_\xi|$ contains the eigenvalues of the flux Jacobian matrix $\hat{A} = \frac{\partial \hat{E}}{\partial \xi}$, as follows:

$$|\Lambda_\xi| = \begin{bmatrix} |\lambda_1| & 0 & 0 & 0 \\ 0 & |\lambda_2| & 0 & 0 \\ 0 & 0 & |\lambda_3| & 0 \\ 0 & 0 & 0 & |\lambda_4| \end{bmatrix} = \begin{bmatrix} |U| & 0 & 0 & 0 \\ 0 & |U| & 0 & 0 \\ 0 & 0 & |U + a\theta| & 0 \\ 0 & 0 & 0 & |U - a\theta| \end{bmatrix} \tag{3.14}$$

where U is the contravariant velocity component in the ξ direction, a is the speed of sound, $\theta = \sqrt{\xi_x^2 + \xi_y^2}$, and ξ_x and ξ_y are the metrics of the curvilinear coordinate transformation. The matrix T_ξ contains the right eigenvectors of $\hat{A}$. In evaluating $|\hat{A}|_{j+\frac{1}{2},k}$ we have used the simple average $(\frac{1}{2}[\,|\hat{A}|_{j,k} + |\hat{A}|_{j+1,k}])$; the Roe average is recommended for flows containing very strong shock waves. To avoid zero eigenvalues, the elements of $|\Lambda|_\xi$ are modified as follows:

$$
\begin{aligned}
\tilde{\lambda}_1, \tilde{\lambda}_2 &= \max(\lambda_{1,2}, V_l\, \sigma) \\
\tilde{\lambda}_3 &= \max(\lambda_3, V_n\, \sigma) \\
\tilde{\lambda}_4 &= \max(\lambda_4, V_n\, \sigma)
\end{aligned} \tag{3.15}
$$

where σ is the spectral radius of the flux Jacobian. We use $V_l = V_n = 0$ for subsonic flow, and $V_l = 0.025$, $V_n = 0.25$ for transonic flows. Note that the value of V_l has a much greater effect on stability and total drag than V_n.

Pulliam [42] showed that the best rate of convergence for the Euler equations is achieved when matched artificial dissipation operators are included both implicitly and explicitly. A contribution from the dissipation, analogous to Equation 3.12, is therefore added to the left-hand-side of the implicit algorithm of Equation 3.9.

The variable $d_{j+\frac{1}{2},k}$ in Equation 3.12 contains second- and fourth-difference terms which scale as first- and third-order terms respectively. The fourth-difference term uses a symmetric five-point stencil:

$$\frac{1}{\Delta\xi}(q_{j+2} - 4q_{j+1} + 6q_j - 4q_{j-1} + q_{j-2}) \tag{3.16}$$

At near-boundary nodes, the following operator is commonly used for the dissipation:

$$\frac{1}{\Delta\xi}(-2q_{j+1} + 5q_j - 4q_{j-1} + q_{j-2}) \tag{3.17}$$

Since this term is only first-order accurate, it is replaced by the following second-order operator for use with the higher-order scheme:

$$\frac{1}{\Delta\xi}(q_{j+2} - 3q_{j+1} + 3q_j - q_{j-1}) \tag{3.18}$$

Oscillations in the vicinity of shocks in transonic flow can arise when using third-order dissipation. To provide better shock resolution, first-order dissipation is added near shocks through the use of the pressure switch, Υ, described above. The effect of first-order dissipation on the global accuracy of transonic solutions is investigated in the Chapter 4.

3.2.2 Inviscid Fluxes

Centered differences are used for the convective fluxes. Note that the use of fourth-difference (third-order) dissipation necessitates the use of a five-point stencil and thus the solution of pentadiagonal systems. Increasing the accuracy of the centered difference operator to fourth order does not increase the stencil size, and the overall increase in computing expense per grid node is small. Finally, note that the grid metrics are evaluated using the same operators as the convective fluxes without any numerical dissipation. It must be stressed that matching the spatial operators of the metrics and convective terms is critical. Not doing so generates large truncation errors. The resulting source term precludes the ability to obtain a zero residual for initial uniform free-stream conditions.

The following operators are used to approximate first derivatives:

Higher-order Algorithm

Interior (4th-order)

$$\delta_\xi q_j = \frac{1}{12\Delta\xi}(-q_{j+2} + 8q_{j+1} - 8q_{j-1} + q_{j-2}) \tag{3.19}$$

First Interior Node (3rd-order)

$$\delta_\xi q_j = \frac{1}{6\Delta\xi}(-2q_{j-1} - 3q_j + 6q_{j+1} - q_{j+2}) \tag{3.20}$$

Boundary (3rd-order)

$$\delta_\xi q_j = \frac{1}{24\Delta\xi}(-11q_j + 18q_{j+1} - 9q_{j+2} + 2q_{j+3}) \tag{3.21}$$

The last equation is required only for the calculation of grid metrics.

3.2.3 Viscous Fluxes

The viscous terms are in the following general form:

$$\partial_\eta(\alpha_j \partial_\eta \beta_j) \tag{3.22}$$

There are a number of ways to deal with Equation 3.22. Some researchers elect to expand the expression, through chain-rule differentiation, into its non-conservative counterpart [33, 44, 45, 34] consisting of first and second derivatives as follows:

$$(\alpha\beta_\eta)_\eta = \alpha_\eta\beta_\eta + \alpha\beta_{\eta\eta} \tag{3.23}$$

The reason is that direct evaluation of the second derivative is significantly more accurate at the small scales than two applications of a first-derivative operator. This would make the non-conservative form more attractive to those using DNS. Another reason could be that successive applications of standard centered-difference operators for first derivatives may not provide sufficient damping to odd-even modes. In the present work, the conservative form of the viscous terms is computed. We apply successive differentiation using quantities at mid-points of the mesh to obtain a conservative operator. The differentiation is first biased in one direction and then biased in the opposite direction to complete the second derivative. We have used this approach without encountering any difficulty.

The following fourth-order expression is used to calculate the $\partial_\eta \beta_j$ term, from Equation 3.22, at half nodes:

$$(\delta_\eta \beta)_{j+\frac{1}{2}} = \frac{1}{24\Delta\eta}(\beta_{j-1} - 27\beta_j + 27\beta_{j+1} - \beta_{j+2}) \tag{3.24}$$

Near boundaries, the following third-order expression is used:

$$(\delta_\eta \beta)_{j+\frac{1}{2}} = \frac{1}{24\Delta\eta}(-23\beta_j + 21\beta_{j+1} + 3\beta_{j+2} - \beta_{j+3}) \tag{3.25}$$

The value of $\alpha_{j+\frac{1}{2}}$ is determined using the following fourth-order interpolation formula:

$$\alpha_{j+\frac{1}{2}} = \frac{1}{16}(-\alpha_{j-1} + 9\alpha_j + 9\alpha_{j+1} - \alpha_{j+2}) \tag{3.26}$$

Near boundaries, a third-order formula is used:

$$\alpha_{j+\frac{1}{2}} = \frac{1}{8}(3\alpha_j + 6\alpha_{j+1} - \alpha_{j+2}) \tag{3.27}$$

Using the following similar expressions,

$$\begin{aligned}
\phi_j &= \alpha_{j+\frac{1}{2}} \delta_\eta \beta_{j+\frac{1}{2}} \\
(\delta_\eta \phi)_{j-\frac{1}{2}} &= \frac{1}{24\Delta\eta}(-\phi_{j+1} + 27\phi_j - 27\phi_{j-1} + \phi_{j-2}) \qquad (3.28) \\
(\delta_\eta \phi)_{j-\frac{1}{2}} &= \frac{1}{24\Delta\eta}(-23\phi_{j-1} + 21\phi_j + 3\phi_{j+1} - \phi_{j+2})
\end{aligned}$$

the complete operator becomes:

$$\begin{aligned}
\delta_\eta(\alpha_j \delta_\eta \beta_j) &= \frac{1}{24\Delta\eta}(\alpha_{j-3/2}(\delta_\eta\beta)_{j-3/2} \qquad (3.29) \\
&\quad -27\alpha_{j-1/2}(\delta_\eta\beta)_{j-1/2} \\
&\quad +27\alpha_{j+1/2}(\delta_\eta\beta)_{j+1/2} \\
&\quad -\alpha_{j+3/2}(\delta_\eta\beta)_{j+3/2})
\end{aligned}$$

in the interior, and

$$\begin{aligned}\delta_\eta(\alpha_j\delta_\eta\beta_j) &= \frac{1}{24\Delta\eta}(-23\alpha_{j-1/2}(\delta_\eta\beta)_{j-1/2} \\ &+21\alpha_{j+1/2}(\delta_\eta\beta)_{j+1/2} \\ &+3\alpha_{j+3/2}(\delta_\eta\beta)_{j+3/2} \\ &-\alpha_{j+5/2}(\delta_\eta\beta)_{j+5/2})\end{aligned} \tag{3.30}$$

near boundaries. This approach leads to a seven-point stencil. On the left-hand-side of the approximate factorization algorithm, we use a second-order operator which is identical to the one used in the original second-order algorithm.

3.2.4 Turbulence Models

The implementation of the Baldwin-Lomax and Spalart-Allmaras turbulence models requires the calculation of the vorticity. The procedure is slightly different for the two turbulence models due to implementation issues.

For the Baldwin-Lomax turbulence model, vorticity is computed at the half-nodes using the operators given in Equations 3.24 and 3.25. The grid metrics are interpolated to the half nodes using Equations 3.26 and 3.27. Since the computation of the eddy-viscosity takes place at the half nodes, all other relevant information is also interpolated using the higher-order interpolants.

For the Spalart-Allmaras model, the eddy-viscosity is first computed at each node and then interpolated to the half node position. Hence, vorticity is computed using Equations 3.19 - 3.21. Equation 3.21 is used to compute vorticity on the airfoil surface. Since the grid metrics are computed at the same nodal positions, there is no need to interpolate. Once the eddy-viscosity is computed at each node, it is interpolated to the half nodes using Equations 3.26 and 3.27.

The diffusive terms in the Spalart-Allmaras turbulence model are handled in the same manner as the viscous terms described in the preceding subsection. A first-order upwind scheme is used for the convective terms in order to maintain positivity of the eddy viscosity. We have experimented with a third-order upwind-biased treatment of the convective terms and seen no degradation in accuracy associated with the use of the first-order operator.

3.2.5 Boundary Conditions and Zonal Interfaces

Far-Field Boundary

The far-field boundary conditions are described in Section 2.4.2 for both CYCLONE and TORNADO. The following second-order extrapolation operator is used at the far-field boundary:

$$q_1 = 3q_2 - 3q_3 + q_4 \tag{3.31}$$

Extrapolation formulas of third order and higher in combination with the fourth-order interior scheme (i.e. Equation 3.19) proved unstable for both far-field and airfoil-body boundary conditions. We expand on this topic in the following subsection.

As described in Section 2.4.2, the use of a finite domain does introduce error, even when a circulation correction is used [66]. The error varies with the inverse of the distance to the outer boundary [48]. However, this error does not depend on the grid density and thus does not affect the error estimates from the grid convergence studies.

Airfoil Body

The pressure at the airfoil surface is determined from a third-order approximation to $\partial p/\partial n = 0$ (see Section 2.4.1), which gives

$$p_1 = \frac{1}{11}(18p_2 - 9p_3 + 2p_4) \tag{3.32}$$

Note that third-order boundary schemes are sufficent to maintain fourth-order global accuracy. Density at the airfoil surface is determined from an expression analogous to Equation 3.32. We have experimented with extrapolation of pressure and density using Equation 3.31 and the following third-order operator:

$$q_1 = 4q_2 - 6q_3 + 4q_4 - q_5 \tag{3.33}$$

Both one- and two-dimensional experiments have shown that for extrapolation, the highest order of accuracy that can be used while maintaining stability appears to be

2 less than that of the interior scheme. In our experience, Equation 3.33 has proven to be unconditionally unstable in conjunction with Equation 3.19 used in the interior. Equation 3.31 was mildly stable for small time-steps. First-order extrapolation proved to be very robust but it would undermine the global accuracy of the higher-order scheme. Experiments with stencils of up to third order (i.e. Equation 3.32) used to approximate Equation 2.43 proved stable for all the cases examined in this work.

Wake-Cut

Although the Baldwin-Lomax model is implemented with either an implicit or an explicit wake-cut, the Spalart-Allmaras model is not implemented to handle wake-cuts implicitly. For consistency, all results presented in this thesis are computed while treating the wake-cut explicitly. The interpolation at the wake-cut (wc) is computed to fourth-order using the data above and below the wake-cut as follows:

$$q_{k_{wc}} = \frac{1}{6}(-q_{k_{wc}+2} + 4q_{k_{wc}+1} + 4q_{k_{wc}-1} - q_{k_{wc}-2}) \tag{3.34}$$

Treatment of Block Interfaces

Neighbouring block boundaries, in the streamwise direction, are overlapped at the interfaces. A specified number of columns of points are taken from the neighbouring block (known as the *halo* column). Consider the rectangular 2-block grid in Figure 3.1. For simplicity, only one *halo* column will be considered here. The first interior column of block 2 is stored in the *halo* column of block 1, and the last interior column of block 1 is stored in the *halo* column of block 2. Blocks 1 and 2 are then updated independently, resulting in two solutions at the block interface. The two interface solutions are subsequently averaged. At steady state, the streamwise interface is completely transparent. Common block interfaces in the cross-stream direction (i.e., sides 1 and 3 in Figures 2.2(b) and 2.3) are treated like wake-cuts and employ Equation 3.34.

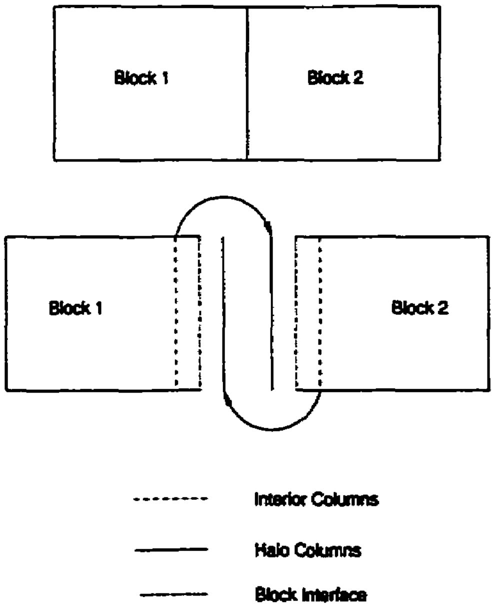


Figure 3.1: 2-block grid with halo data.

3.2.6 Force Integration

A popular second-order approach to the integration of the pressure field is to take the average C_p value between two neighbouring nodes on the airfoil surface and have the vector act normal to the line joining the two points. This is illustrated in Figure 3.2. Proceeding around the airfoil, the appropriate contributions in both the normal and axial direction with respect to the chord line are summed. Once the shear stress is computed at every node, it too is averaged and summed to give the viscous contribution to the normal and axial forces. A more accurate procedure is necessary to maintain high-order global accuracy.

The following expressions are used to evaluate the normal and axial force coefficients, C_N and C_A respectively, with respect to the chord line. (For ease of presentation, we consider the pressure contribution only.)

$$C_N = \frac{1}{c}\oint -C_p(\hat{n}\cdot\hat{y})ds \tag{3.35}$$

$$C_A = \frac{1}{c}\oint -C_p(\hat{n}\cdot\hat{x})ds \tag{3.36}$$

where c is the chord length, s is the arclength along the airfoil surface (see Figure 3.2),

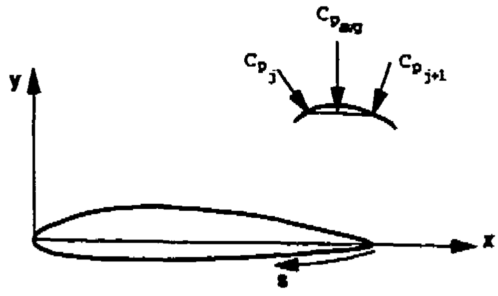


Figure 3.2: Average C_p values for integration of surface pressure.

x and y are the Cartesian coordinates, and $\hat{x}$ and $\hat{y}$ are unit vectors in the coordinate directions. The unit normal with respect to the surface, $\hat{n}$, is given by

$$\hat{n} = \frac{-\frac{dy}{ds}\hat{x} + \frac{dx}{ds}\hat{y}}{\sqrt{(\frac{dx}{ds})^2 + (\frac{dy}{ds})^2}} \tag{3.37}$$

We integrate the pressure and shear stress distributions with respect to the arclength around the airfoil. This avoids any possible singularities near the leading or trailing edges.

A cubic spline is used to fit a curve through the nodes making up the airfoil surface. The spline allows for the 3rd-order interpolation of $\frac{dx}{ds}$ and $\frac{dy}{ds}$ at any point on the airfoil surface. The pressure distribution is also splined. An adaptive quadrature routine is used to integrate Equations 3.35 and 3.36. The quadrature routine uses the two-point Gauss-Legendre rule as the basic integration formula with a global error-control strategy. Details regarding the mechanics of the global strategy can be found in Malcolm and Simpson[36].

The calculation of C_p does not explicitly involve any differencing. The skin-friction coefficient, C_f, however, is computed as follows:

$$C_f = \frac{\tau_w}{q_\infty} \tag{3.38}$$

where τ_w is the shear stress along the airfoil surface, $q_\infty = \frac{1}{2}\rho_\infty M_\infty^2$, is the dynamic pressure, and M_∞ is the free-stream Mach number. The shear stress is computed as follows:

$$\begin{aligned}\tau_w &= \mu\left(\frac{\partial u}{\partial y} - \frac{\partial v}{\partial x}\right) \\ &= \mu\Big[(u_\xi \xi_y + u_\eta \eta_y) - (v_\xi \xi_x + v_\eta \eta_x)\Big]\end{aligned} \tag{3.39}$$

For viscous flow, $u = v = 0$ along the airfoil surface, hence $u_\xi = v_\xi = 0$. The following fourth-order spatial operator is used for the derivatives computed normal to the surface:

$$\delta_\eta q_1 = \frac{1}{12\Delta\eta}(-25q_1 + 48q_2 - 36q_3 + 16q_4 - 3q_5) \tag{3.40}$$

The grid metric terms use Equation 3.21. Note that explicit use of the metric terms in Equation 3.39 underlines the importance of the accuracy of the metric terms if one is to compute skin friction accurately. The commonly used second-order counterpart to Equation 3.40 is as follows:

$$\delta_\eta q_1 = \frac{1}{2\Delta\eta}(-3q_1 + 4q_2 - q_3) \tag{3.41}$$

It has been our experience that this stencil is inaccurate for grids that are highly stretched in the direction normal to the airfoil surface. The truncation errors are large and contribute considerable error to the integrated drag force. For a thorough study of the effect of grid density and distribution on skin friction using various turbulence models, the reader is referred to reference [57].

Chapter 4

Results and Discussion

In this section, we compare results computed using the higher-order algorithm with those computed using a second-order discretization. Both schemes use matrix dissipation. The second-order scheme uses a second-order three-point centered stencil for the grid-metric approximations and the inviscid and viscous fluxes. Zeroth-order extrapolation is used at the body surface and the far-field boundaries. The shear-stress distribution on the airfoil surface and integrated body forces are computed to second-order accuracy.

Zingg at al. [69] showed that this second-order discretization produces numerical accuracy which is very similar to that obtained using either a third-order upwind-biased flux-difference-split scheme or the convective upstream split pressure scheme with second-order approximations for the viscous fluxes. Hence this lower-order discretization is representative of the most popular current algorithms and provides a suitable benchmark for assessing the higher-order discretization. The goal of many researchers in CFD today is to be able to predict total vehicle drag to within 1 or 2 percent [2, 12]. Physical-model errors such as those associated with laminar-turbulent transition, turbulence, and the thin-layer approximation generally exceed those error levels. Limiting the numerical error to two percent, however, helps to avoid compounding these errors and allows for a more accurate assessment of the physical model.

The accuracy of the integration routines is investigated in Section 4.1. Results obtained using CYCLONE are presented in Section 4.2 and those obtained using TORNADO are presented in Section 4.3. The figures of this chapter are in Appendix B.

4.1 Force Integration

To investigate the accuracy of the integration routines we examine a flow for which there is an analytical solution. Using a conformal mapping, we can obtain the pressure distribution and the lift coefficient for the steady incompressible potential flow over a Joukowsky airfoil. Thus we can evaluate integration techniques by applying them to a finite number of pressure values from the analytical pressure distribution and comparing with the analytical lift and drag coefficients. Figure B.1 depicts the error incurred in using the two different integration algorithms (designated "2nd-order" and "third-order") in computing C_l and C_d as a function of the number of points used in the integration procedure, N. The figure shows the expected slopes corresponding to second- and third-order accuracy. The higher-order integration scheme reduces the error significantly. With, say, 200 points distributed around the airfoil, the second-order integration procedure produces an error in the lift coefficient well below 0.001, which should suffice for virtually any aerodynamic application. However, the error in the drag coefficient is about 1×10^{-5}, which is likely to affect the third significant figure in a practical context.

4.2 CYCLONE Results

4.2.1 Overview of Test Cases and Grid Details

Computational results, obtained using CYCLONE, are presented for the following test cases:

1. NACA 0012 airfoil, M_∞=0.16, α=6°, $\mathcal{R}e$=2.88 × 10^6, laminar-turbulent transition at 0.05 and 0.8 chords on the upper and lower surfaces, respectively.

2. NACA 0012 airfoil, $M_\infty = 0.16$, $\alpha = 12°$, $\mathcal{R}e = 2.88 \times 10^6$, laminar-turbulent transition at 0.01 and 0.95 chords on the upper and lower surfaces, respectively.

3. NACA 0012 airfoil, $M_\infty = 0.7$, $\alpha = 3°$, $\mathcal{R}e = 9.0 \times 10^6$, laminar-turbulent transition at 0.05 chords on both surfaces.

4. RAE 2822 airfoil, $M_\infty = 0.729$, $\alpha = 2.31°$, $\mathcal{R}e = 6.5 \times 10^6$, laminar-turbulent transition at 0.03 chords on both surfaces.

5. RAE 2822 airfoil, $M_\infty = 0.754$, $\alpha = 2.57°$, $\mathcal{R}e = 6.2 \times 10^6$, laminar-turbulent transition at 0.03 chords on both surfaces.

These cases span a range of typical aerodynamic flows. Cases 1 and 2 are both subsonic flows, the former fully attached, the latter mildly separated. Experimental data can be found in Gregory and O'Reilly [24]. Cases 3 and 4 are transonic flows with moderate-strength shock waves. Case 5 is characterized by a much stronger shock wave on the upper surface than Case 4. There is also a much larger region of shock-induced boundary-layer separation. Experimental data for Cases 4 and 5 can be found in Cook et al. [11] The measured coordinates for the RAE 2822 airfoil are used, as in Maksymiuk et al. [35], rather than the standard coordinates.

Tables 4.1 and 4.2 summarize the grids used for CYCLONE. The family of grids outlined in Table 4.1 is primarily used for subsonic cases 1 and 2, while the grids described in Table 4.2 are used for transonic cases 4 and 5 only. Both families of grids are used for case 3. All of the grids have a "C" topology. The distance to the far-field boundary is 12 chords for all grids. While this causes some numerical error, the error does not scale with grid density. The error is proportional to the inverse of the distance to the outer boundary [48]. Since the distance to the outer boundary is common to all grids, this error will not affect our conclusions. Grid A† was generated using an elliptic grid generator. Grid B was generated by removing every second node in both coordinate directions from grid A, and grid C was similarly generated from grid B. This technique produces a sequence of grids suitable for a grid convergence

†When a grid is referred to without a numeric reference, i.e. grid A instead of grid A1 or A2, we are referring to grid A from both sets of families.

Grid	Dimensions	Points on Upper Surface	Points on Lower Surface	Off-Wall Spacing ($\times 10^{-6}$)	Leading Edge Clustering ($\times 10^{-3}$)	Trailing Edge Clustering ($\times 10^{-3}$)
A1	1057×193	401	400	0.23	0.1	0.5
B1	529×97	201	200	0.53	0.2	1.0
C1	265×49	101	100	1.2	0.4	2.0
C1a	277×49	113	100	1.2	0.4	2.0

Table 4.1: Grid family 1 (CYCLONE).

Grid	Dimensions	Points on Upper Surface	Points on Lower Surface	Off-Wall Spacing ($\times 10^{-6}$)	Leading Edge Clustering ($\times 10^{-3}$)	Trailing Edge Clustering ($\times 10^{-3}$)
A2	1025×225	501	300	0.23	0.1	0.25
B2	513×113	251	150	0.53	0.2	0.5
C2	257×57	126	75	1.2	0.4	1.0

Table 4.2: Grid family 2 (CYCLONE).

study. Two examples of the grids used are shown in Figure B.2. Where transonic cases are examined using the first family of grids, we also show results for grid C1a, which has additional grid nodes clustered near the upper-surface shock wave. The second family of grids has increased node density in the normal direction and more nodes on the upper surface than the lower surface (but no clustering at the shock). Grid C, from both families, is relatively coarse (under 15,000 nodes), with a node density suitable for extension to three-dimensional computations, while grids A and B are primarily for estimation of solution error. For all grids and cases, the y^+ value at the first point from the surface is less than one, where y^+ is the standard law-of-the-wall coordinate, and therefore there are a few grid points in the linear sublayer of the turbulent boundary layers.

4.2.2 Test Case 1 - NACA 0012 Subsonic Flow

Figure B.3 shows the lift, pressure drag, and skin-friction drag computed using the Baldwin-Lomax turbulence model for case 1 on grids A1, B1, and C1. The corresponding results computed using the Spalart-Allmaras turbulence model can be found in Figure B.4. They are plotted versus $1/N$, where N is the number of grid nodes. Agreement between the two algorithms on grid A1 is good, indicating that numerical errors are very small on this grid. Thus grid A1 provides a reference for estimating numerical errors on grids B1 and C1. Individual drag components, pressure and friction drag, are shown instead of total drag in order to get a better picture of solution accuracy. The errors in these components are often of opposite sign.

Examining the results using the Baldwin-Lomax model for case 1, one finds that both discretization schemes produce errors in lift of less than one percent on grid C1. The errors in pressure drag on grid C1 are larger, with the higher-order algorithm producing about 3 percent error and the second-order algorithm producing an error of approximately 40 percent. Similarly, the higher-order algorithm produces an error in friction drag below 2 percent on grid C1, while the error from the second-order algorithm approaches 12 percent.

The results obtained using the Spalart-Allmaras model, displayed in Figure B.4, are similar to those obtained using the Baldwin-Lomax turbulence model. The similar error levels indicate that the first-order convective terms in the Spalart-Allmaras model are not a significant source of numerical error. The second-order algorithm produces an error in pressure and friction drag, on grid C1, of almost 30 percent and 15 percent respectively. Similar to the Baldwin-Lomax results, the higher-order algorithm produces error levels of approximately three percent in pressure drag and one percent in friction drag.

The results presented in this work are not intended to demonstrate the formal order of accuracy of the higher-order algorithm. In fact, the data in Figures B.3 and B.4 provide no indication that the order of accuracy of the higher-order algorithm is greater than second-order. Without proper treatment of flow or grid singularities, it is unlikely that third-order behaviour can be shown. Instead, we wish to emphasize the improved accuracy of the higher-order algorithm on grid C1, which is typical of grids used in practice. Based on Figures B.3 and B.4, the higher-order solution computed on grid C1 is more accurate than the second-order solution computed on grid B1, which has four times as many nodes.

4.2.3 Test Case 2 - NACA 0012 Subsonic Flow

The flow in this test case is characterized by a small region of separated flow on the upper surface near the trailing edge. To provide a broader perspective on the potential of the higher-order scheme, the accuracy of four different spatial discretizations are compared. The family of grids from Table 4.1 are used and the results, using the Baldwin-Lomax turbulence model, are plotted in Figure B.5. The methods are labelled as follows:

1. **Second-order** - matrix artificial dissipation with second-order centered differences for both inviscid and viscous fluxes, second-order metric approximations and force integration;

2. **CUSP** - the convective upstream split pressure (CUSP [56]) scheme as implemented by Nemec and Zingg [39], with second-order centered differences for both inviscid and viscous fluxes, second-order metric approximations and force integration;

3. **Third-order upwind** - third-order upwind-biased scheme [49] for the inviscid terms, as implemented by Jespersen et al. [29], with second-order viscous terms, grid metrics, and integration;

4. **Higher-order** - matrix artificial dissipation with fourth-order centered differences for the inviscid and viscous terms, fourth-order metric approximations and third-order force integration;

The errors in the lift coefficient for all four methods are below two percent on grid C1. All of the methods appear to be equally accurate. Closer inspection of the flowfield within the boundary layer shows that this is not the case, as we shall see later. The errors in the drag coefficients computed on grid C1 are much larger. For the two subsonic cases, examined thus far, the benefits of the higher-order scheme are significant. The errors in pressure and friction drag produced by the higher-order scheme on grid C1 are generally less than two percent. Furthermore, these errors are smaller than those produced by the other three schemes on grid B1, which has four times as many nodes. The errors in the drag components from the three remaining schemes are greater than two percent, even on grid B1.

Closer inspection of the solutions reveals that the lower-order schemes lead to an overprediction of the boundary-layer thickness on the upper surface, consistent with overprediction of pressure drag and underprediction of friction drag. Figure B.6 shows the skin-friction distribution near the leading edge. The grid A1 results, which provide an accurate reference solution, show every fourth grid node. The higher-order results on grid C1 are in good agreement with the reference solution, while the second-order result significantly underpredicts the maximum near the leading edge. The higher-order solution is also more accurate along the upper surface where the error in the second-order results persists all the way to the trailing edge. Figures B.7 and B.8 show the boundary-layer profiles at 85% chord on the upper surface computed on grid C1. The higher-order scheme is superior to the other schemes, illustrating the importance of raising the accuracy of the discretization of the viscous terms. The higher-order results are virtually grid independent, even on grid C1. Note that the second- and higher-order algorithms both use the same numerical dissipation scheme. Given the accurate results of the higher-order scheme on grid C1, the third-order matrix dissipation does not appear to contaminate the solution. It is achieving its goal of producing stability and damping under-resolved modes without introducing significant error. The error in the second-order results must, therefore, be dominated by discretization errors of the inviscid and viscous terms. The third-order upwind results, however, are not much better than the second-order results suggesting that the discretization error of the metrics and the viscous terms dominate the third-order upwind results.

4.2.4 Test Case 3 - NACA 0012 Transonic Flow

The results computed using the second- and higher-order scheme with the Baldwin-Lomax model, on grid family 1, are displayed in Figure B.9. The higher-order discretization produces the smallest pressure drag error on grid C1, but is nevertheless well in excess of two percent. One source of this error is the first-order dissipation introduced near the shock wave. Both schemes produce lower pressure drag errors when run without any limiting, i.e. without any first-order dissipation, but visible oscillations result. Another source of error with the matrix dissipation scheme (thus affecting the higher-order algorithm as well) is the requirement of nonzero values of V_l and V_n in Equation 3.15 for transonic flows. This leads to some of the overdissipation characteristic of the scalar artificial dissipation scheme. For example, the results in Figure B.9 were obtained using $V_l = 0.025$. Reducing V_l to a value of 0.015 reduces the pressure drag result for the higher-order algorithm on grid C1 from 0.00896 to 0.00889. The corresponding error in those values, compared to the solution on grid A1, is 3.9% and 3.1% respectively, a 20% reduction in error. Reducing V_l even further to 0.005 does not improve the result. Although the relative reduction in error is substantial, the goal is to produce results, on grids with similar densities as grid C1, with errors no greater than two percent.

Adding nodes near the shock, as in grid C1a, does not reduce the pressure drag error significantly. Figure B.10 shows details of the pressure coefficient on the upper surface of the airfoil computed on grid C1a. For the grid A1 solution, every second grid point is plotted. The higher-order algorithm produces an improvement in the shock location and a significant reduction in error in the low pressure region forward of the shock, with the grid A1 solution taken as a reference. The grid A1 results show a spike at the laminar-turbulent transition point which is not seen on the coarser grid C1a. The present treatment of transition in the Baldwin-Lomax turbulence model is slightly grid-dependent and may explain some of the error seen.

The added nodes near the shock wave do little to reduce the pressure drag error. The numerical error associated with the added first-order dissipation in that region is, therefore, not the largest source of error. Results using the Spalart-Allmaras model produce similar error levels, indicating that the error is not likely to be related to discretization errors within the turbulence models. It turns out that a small recirculation region exists just aft of the shock location. More nodes are needed in the direction normal to the airfoil surface to adequately resolve the boundary layer in that region. The second family of grids, outlined in Table 4.2, addresses this issue. Using grid family 2 reduces the pressure drag error significantly. Results for this test case using the Spalart-Allmaras model on grids A2, B2, and C2 are found in Figure B.11. The numerical errors, for the higher-order algorithm, have all been reduced to less than 2% on grid C2, which has less than 15,000 nodes.

4.2.5 Test Cases 4 and 5 - RAE 2822 Transonic Flow

Figures B.12 and B.13 show the lift and drag components computed using the Spalart-Allmaras model on grid family 2. The results indicate that the higher-order discretization leads to a significant reduction in the error relative to the second-order scheme, generally producing solutions on grid C2 which are accurate to within 2 percent. The exception is the pressure drag for case 5, for which the higher-order solution has an error of nearly 4 percent, and the solution computed using the second-order scheme has an error just over 5 percent. Using third-order dissipation alone only marginally improves the pressure drag results indicating that using first-order dissipation near the shock is not the source of this error. It appears that there is insufficient grid resolution in the vicinity of the separation bubble at the shock even for the higher-order scheme. Despite the small improvement in pressure drag error, closer examination of local flow characteristics indicates that the higher-order solution is significantly more accurate than the second-order solution. Figure B.14 shows a portion of the computed pressure coefficient distribution on the upper surface for case 5. The solution using the higher-order discretization on grid C2 lies much closer to the grid A2 solution than that computed using the second-order scheme on grid C2. Boundary-layer profiles confirm the improved accuracy of the higher-order scheme. Figure B.15 shows the computed profiles on the upper surface at 95% chord. The error in the velocity profile computed on grid C2 using the second-order scheme is quite large, while the error in the higher-order results, though visible, is small.

4.2.6 Accuracy of viscous terms, grid metrics and force integration

In the context under consideration here, namely the solution of the thin-layer Navier-Stokes equations using a generalized curvilinear coordinate transformation, the extension of all terms to higher order can be accomplished very efficiently (see Section 4.2.7). In other contexts, such as the full Navier-Stokes equations or finite-volume algorithms on unstructured grids, the costs associated with higher-order approximations can be substantial [6]. Thus it is instructive to examine the relative importance of raising various terms to higher order.

The spatial discretization of the entire code can be broken down into various components, as outlined in Section 3.2. In this section, all terms relating to viscosity and turbulence are referred to as viscous terms. It is our experience that the most accurate results are obtained when all the components are treated in a similar fashion, that is, the order of accuracy of all the components of the discretization is consistent. Mixing higher-order inviscid terms with low-order[‡] metric terms can lead to large truncation errors. Similarly, mixing higher-order metrics with low-order viscous terms can also lead to erroneous results. The problem is magnified when dealing with cases involving flow discontinuities such as shocks. Hence, it can be difficult to determine exactly how effective any one component of the discretization is at reducing numerical error since the result can behave in a very nonlinear fashion. Nonetheless, we attempt to address some of these issues here.

In Section 4.2.3, we compared various discretization schemes for a subsonic case, including a third-order upwind scheme. It can be shown that the third-order upwind treatment of the inviscid terms is equivalent to a fourth-order central scheme with a third-order dissipative component, much like the higher-order algorithm described here [69]. The third-order upwind scheme is combined with second-order grid metrics, viscous terms, and force integration. Figure B.16 shows a portion of the upper surface pressure distribution, for case 4, obtained using the Baldwin-Lomax turbulence model on grid C1a. For the grid A1 solution, every second point is plotted. Compared to the

[‡] "Low-order" referes to orders of accuracy of second-order or lower

third-order scheme, the higher-order discretization produces a significant reduction in error over the first 20% chord. The result provides further evidence of the importance of raising all components of the discretization, including the grid metrics, to a higher-order of accuracy.

In [16], De Rango and Zingg, carried out a study of the effect on accuracy of raising the viscous terms and integration algorithm to higher order. We now summarize and expand upon those results. First we address the relative importance of the accuracy of the viscous terms. Plotted in Figure B.17 are results for case 1 using the higher-order scheme with second- and fourth-order centered treatments of the viscous terms. Grid family 1 is used. Note that higher-order metrics and force integration are used for both sets of results. The lower-order treatment of the viscous terms surprisingly improves the lift results. The error in the original higher-order lift result, however, is less than one percent on grid C1 and is considered sufficiently accurate. The second-order viscous terms have the opposite effect on the accuracy of the individual drag components. In fact, the error on grid C1 doubles when the lower-order viscous terms are used.

Although the results vary from case to case, the higher-order viscous terms generally account for roughly 10% of the error reduction associated with the higher-order discretization relative to the second-order scheme. Using case 2 as an example, the second-order algorithm produces an error in pressure drag on grid C1 of roughly 47% in comparison with the grid A1 solution. Using the higher-order algorithm, this error is reduced to 1.3%. If lower-order approximations are used for the viscous terms, the error increases to 4.6% percent. Although the higher-order viscous terms account for a relatively small fraction of the overall error reduction, they reduce the error by a factor greater than three in this example.

Velocity boundary-layer profiles for case 5 were shown in Figure B.15. In Figure B.18, we add to those results the velocity profile computed using the higher-order discretization with the viscous terms discretized using the second-order scheme. Consistent with the previous example, raising the viscous terms to higher order accounts for roughly 10 percent of the overall error reduction.

For some transonic cases, using a lower-order approximation for the viscous terms

Shear-Stress Approximation	Grid B1 % Error	Grid C1 % Error
higher-order	0.4	1.4
2nd-order	-1.5	-7.9

Table 4.3: Effect of shear-stress approximation on accuracy of C_{d_f} using higher-order solution for case 1 (Baldwin-Lomax model). Note: % error is relative to the value computed using the higher-order algorithm on grid A1, which is $C_{d_f} = 0.005277$.

Shear-Stress Approximation	Grid B1 % Error	Grid C1 % Error
higher-order	0.0	0.2
2nd-order	-1.8	-9.4

Table 4.4: Effect of shear-stress approximation on accuracy of C_{d_f} using higher-order solution for case 3 (Baldwin-Lomax model). Note: % error is relative to the value computed using the higher-order algorithm on grid A1, which is $C_{d_f} = 0.004967$.

can have a more adverse effect on the accuracy of the higher-order algorithm. Friction drag results for case 3 are shown in Figure B.19. Compared to the grid A1 solution, the skin friction obtained on grid C1, using second-order approximations for the viscous terms, is in error by approximately 10%. The error is not reduced significantly even using grid B1 which has four times as many nodes. This is a good example of the nonlinear effects discussed earlier when mixing components of different spatial accuracies.

The computation of friction drag is a two-step process, the first being the computation of shear stress given by Equation 3.39. The second involves the actual integration of the surface shear-stress distribution. The accuracy of the shear-stress computation has a much larger effect on the accuracy of the computed friction drag than the integration of the shear-stress distribution. Friction drag results for cases 1 and 3, for the higher-order solution using the higher-order integration routine, are shown in Tables 4.3 and 4.4. The effect on accuracy using second- and fourth-order approximations (Equations 3.40 and 3.41) for the normal velocity derivatives in Equation 3.39 is examined. Third-order approximations are used for the grid metrics on the airfoil surface. The results clearly indicate the importance of treating most terms in a consistent manner. Similar results to the second-order results are obtained if fourth-order approximations are used for the velocity derivatives and second-order approximations are used for the surface grid metrics. The second-order three-point one-sided difference operator (Equation 3.41) typically used for grid metric terms and differencing on the surface is found to be particularly susceptible to error from grid stretching.

For the cases studied, the difference in integrated lift and friction drag values between the second- and third-order integration routines is small. The third-order integration routine improves the integrated pressure drag values by 0.5 - 1% of the reference C_{d_p} obtained on grid A1.

4.2.7 Convergence rate and computational efficiency

Density residual convergence histories for grid C, using the Spalart-Allmaras turbulence model, are displayed in Figure B.20. Convergence using the Baldwin-Lomax model is similar. In all cases, the two algorithms converge similarly for the first three to four orders of magnitude reduction in residual, and the higher-order algorithm converges somewhat more slowly after that. Figure B.21 shows the drag convergence histories for the four cases. Convergence of lift and drag is typically achieved after about four orders of residual reduction on grid C, so the extra cost associated with the higher-order algorithm is quite small.

For the solution of the thin-layer Navier-Stokes equations using a generalized curvilinear coordinate transformation, the extension of all terms to higher order can be accomplished very efficiently. The cost per grid node per iteration is increased by about 6%. Since the lift and drag convergence rates are not significantly affected, the overall cost increase on a given grid is about 6% relative to the second-order algorithm. Hence the computational effort required to achieve a given level of accuracy is greatly reduced using the higher-order algorithm. In some cases, equivalent accuracy is achieved in less than 1/16 the expense of the second-order algorithm, which requires a much finer grid.

Element	Upper Surface	Lower Surface
Slat	5.00	-
Main	0.97	11.0
Flap	2.70	99.3

Table 4.5: Transition locations for case 6 given as percentage of elemental chord.

4.3 TORNADO Results

4.3.1 Overview of Test Case and Grid Details

The test case examined is Case A-2 from AGARD Advisory Report No. 303. Wind tunnel data were measured for a two-dimensional supercritical airfoil with high-lift devices and the model designation is NHLP 2D. These data were obtained during the 1970's as part of the National High Lift Programme in the United Kingdom. The case selected for examination here is L1T2 which includes a 12.5%c leading-edge slat and a 33%c single slotted flap, where c is the chord length of the nested configuration. The slat is located in the optimum position at an angle of 25 degrees and the flap angle is 20 degrees. This geometry, which is typical of a take-off configuration, is shown in Figure B.22. TORNADO results for this case were first presented by Nelson et al. [38]. It should be noted that in reference [38], the blunt trailing edge of the flap is closed by rotating the upper and lower surfaces through equal angles. The sharp points on the lower surface of the slat and main element are actually very small blunt edges. The same coordinates are used in this study with the exception that the lower-surface blunt edges of the slat and main element (not referring to the trailing edges) are also closed to a single point. A full set of coordinates for this case can be found in Appendix D.

The flow conditions for this case, test case 6, are set at $M_\infty = 0.197$, a Reynolds number of 3.52×10^6, and an angle of attack of 20.18°. The transition points are tabulated in Table 4.5. The transition for the lower surface of the slat is fixed to the third node from the sharp point between the slat leading and trailing edges as

Grid	Number of Nodes
A	255,295
B	183,721
C	126,185
D	72,837
E	51,749

Table 4.6: Multi-block grid densities.

illustrated in Figure B.23.

Five grids are used in this study, each of them generated independently with an H-mesh topology. Generating a sequence of grids, suitable for multi-element cases, in the fashion described in Section 4.2.1, would make the grid density of a grid A far too impractical. The solution domain for each grid is divided into 27 blocks. The same domain decomposition is shared amongst all five grids and is shown in Figure B.24. The grids are labeled as grids A through E. The grid densities are given in Table 4.6. Grid densities of individual blocks are outlined in Tables C.1 - C.5 in Appendix C. Given that this is a high-lift case, the far-field boundary is placed at a distance of 24 chords from the airfoil surface, twice the grid extent used for the single-element cases. The grid cells at the far-field boundaries are approximately 1 chord in length. Individual block detail is provided for two reasons: 1) to allow for precise reproduction of the grids by other researchers in the future, and 2) to give the reader a detailed account of the distribution of the grid points amongst the three elements, noting that the grid density required for accurate results is different for each element.

The slat requires special attention. At an angle of attack of 20.18°, the pressure gradients near the leading edge are quite large. In fact, in Nelson et. al. [38], it was shown that at high angles of attack, very high flow gradients exist outside the boundary layer near the leading edge of the slat. Hence, given the finite number of grid points available, it is felt that grids B through E require a cluster point in that region. The cluster point is located at approximately 14.7% along the arclength from the begining of side 1, block 2, to the trailing edge on the upper surface. It is

Drag Component	Higher-order			Second-order		
	Slat	Main Element	Flap	Slat	Main Element	Flap
C_{d_p}	-0.71975	0.54096	0.22272	-0.71929	0.54082	0.22311
C_{d_f}	0.00196	0.00683	0.00189	0.00193	0.00672	0.00184

Table 4.7: Elemental drag components on grid A for case 6.

located just after the leading edge, with a spacing of $2 \times 10^{-4}c$. The cluster point was not used in grid A both to avoid any influence on the reference solution from cluster points, and since the large number of points placed on the slat makes cluster points unneccesary for this grid.

Trailing-edge clustering is set at $5 \times 10^{-4}c$ for all three elements. The trailing-edge clustering is kept constant for all of the grids. Given that this spacing is the smallest of those used in the single-element cases, it is unlikely to introduce any significant numerical error. The off-wall spacing is also kept constant for all five grids at 10^{-6} chords.

4.3.2 Test Case 6 - High-Lift Subsonic Flow

The higher-order discretization schemes implemented in CYCLONE and TORNADO are identical with one exception. The original second-order treatment of the diffusive terms within the Spalart-Allmaras turbulence model is used. The higher-order treatment of those terms in TORNADO presented stability problems which are likely related to the handling of block interfaces. The Spalart-Allmaras turbulence model was exclusively used for this test case. Values for the limiters, V_l and V_n, used in Equation 3.15 in the matrix dissipation scheme were set to 0.01.

The results for this test case are plotted in Figure B.25. The errors in lift coefficient are small. Nevertheless, similar to the results for test cases 4 and 5, the higher-order scheme, coupled with the Spalart-Allmaras turbulence model, predicts lift more accurately than the second-order scheme on grids D and E. The errors in the drag components are much larger. Compared to the solution on grid A, the errors in the drag components for the higher-order scheme on grid D are less than 3% while the

Algorithm	Slat	Main Element	Flap	Complete Airfoil
higher-order	2.9	5.1	2.3	10.3
second-order	98.9	-34.7	-8.8	55.4

Table 4.8: Elemental pressure drag error in counts on grid D for case 6. (Note: One count is equivalent to 0.0001 units in drag. The count errors are relative to the solution of each respective algorithm on grid A - see Table 4.7)

Algorithm	Slat	Main Element	Flap	Complete Airfoil
higher-order	0.2	-0.3	-0.5	-0.6
second-order	-1.0	-8.6	-3.9	-13.5

Table 4.9: Elemental friction drag error in counts on grid D for case 6. (Note: One count is equivalent to 0.0001 units in drag. The count errors are relative to the solution of each respective algorithm on grid A - see Table 4.7)

error in the second-order results for both pressure and friction drag coefficents on grid D exceeds 12%. The error in the drag components for the second-order scheme approaches 22% on grid E while the error in the higher-order result is still well behaved at approximately 5%.

Individual drag components were plotted for single element cases in Section 4.2 to avoid cancelation of errors between pressure and friction drag. When analyzing solutions about multi-element airfoils, the same care must be taken to avoid cancelation error between elemental pressure and friction drag. For example, the pressure drag is negative on the slat and positive on the remaining two elements. Elemental pressure and friction drag values for solutions on grid A are presented in Table 4.7. The results in Table 4.7 provide a reference for the elemental-drag errors presented in Tables 4.8 and 4.9 for solutions on grid D. A benefit of analyzing the data in this fashion is that it provides the reader with some insight into which areas of the grid need to be refined if further improvement in accuracy is desired. The pressure-drag error on the slat and main element for the second-order solution is quite large. The opposite sign of the errors lead to significant cancellation error as well. The elemental pressure- and friction-drag error for the higher-order solution is several times smaller than those produced by the second-order algorithm.

Grid	C_l	C_{d_p}	C_{d_f}
AA	4.074	0.04398	0.01070
A	4.074	0.04393	0.01068

Table 4.10: Higher-order results for case 6.

To ensure that the off-wall spacing of 10^{-6} chords is sufficiently small enough not to introduce significant numerical error, we introduce grid AA. Grid AA is identical to grid A except that the off-wall spacing is reduced to 5×10^{-7} chords. The higher-order results on grid A and AA are shown in Table 4.10. The difference between the two solutions is negligible indicating that the original choice for the off-wall spacing is adequate. For all grids, the y^+ value at the first point from the surface is less than one, having a maximum of 0.8 on the main element for grid E.

Figure B.26 shows the experimental and computed surface pressure distributions for the NHLP airfoil. The computed result, using the second-order scheme on grid A, is very accurate, comparable to those presented by Nelson et. al. [38], using a somewhat different grid configuration. A portion of the upper surface pressure distribution of the slat is shown in Figure B.27. The second-order scheme, on grid D, does poorly at computing the minimum pressure. Similar results are found on the main element as well.

Figure B.28 shows boundary-layer velocity profiles at the trailing edge of the flap. For the second-order solution on grid A, every third grid point is plotted. The profiles can be divided into four regions:

1. the first 3% of chord above the flap surface corresponds to the flap boundary layer;

2. the region between 3% and 10% of chord corresponds to the wake from the main element;

3. the region between 10% and 18% of chord corresponds to the wake from the slat;

4. beyond 18% of chord above the flap surface, the flow slowly returns to free-stream conditions

Given the superior results presented thus far for the higher-order scheme, the higher-order result on grid A is taken as the reference solution. Region 1 appears to be adequately resolved for both discretization schemes, even on grid D. In region 2, the higher-order result on grid D is more accurate than the second-order result on grid A, a grid with more than 3 times as many nodes. In region 3, the error in the second-order result on Grid D is quite large and increases in region 4. These slower velocities in the wake explain the larger drag values reported earlier. The second-order result on grid A and the higher-order result on grid D provide similar accuracy in regions 3 and 4. The second-order grid A result is slightly better in region 3 while the higher-order grid D result is slightly better in region 4. Nonetheless, the higher-order scheme is in excellent agreement with the reference solution using a grid with only 73,000 nodes, one third of the grid density of grid A.

4.3.3 Convergence rate and computational efficiency

Density residual and drag convergence histories for grid D, are displayed in Figure B.29. As in the single-element cases, the two algorithms converge similarly for the first three to four orders of magnitude reduction in residual, and the higher-order algorithm converges somewhat more slowly after that. In this case, the higher-order scheme takes approximately 33% more iterations than the second-order scheme to converge to within 0.2% of the converged drag. The higher-order algorithm, however, produces a solution that is far more accurate on grid D with the second-order scheme requiring at least 3 times as many nodes to produce similar accuracy.

The added computational cost of the higher-order scheme in TORNADO is similar to the added cost in CYCLONE. Compared to the second-order algorithm, the cost per grid node per iteration is increased by about 7%, with no increase in memory usage. This is an important factor when considering extending TORNADO to 3D applications. The higher-order scheme produces accurate results on relatively coarse grids, thereby reducing memory requirements and computational costs. Table 4.11

Grid	Number of Nodes	Memory Used (megabytes)
A	255,295	102
B	183,721	75
C	126,125	54
D	72,837	34
E	51,749	26

Table 4.11: Memory requirements for TORNADO.

summarizes the memory requirements of TORNADO for the grid densities used in this study. The higher-order scheme produces results on grid D within three percent of the solutions on grid A while only using 34 megabytes of memory.

Chapter 5

Contributions and Conclusions

We have presented a stable, accurate, and robust higher-order algorithm for aerodynamic flows, and, furthermore, we have compared its efficiency with that of a well-established second-order algorithm. The higher-order algorithm was implemented in both a single- and multi-block solver. With a few exceptions, all components of the spatial discretization, including the convective and viscous terms, the numerical boundary schemes, the numerical dissipation, and the integration technique used to calculate forces and moments, have been raised to a level of accuracy consistent with third-order global accuracy. The turbulence models were also addressed, with most of the terms raised to a higher order of accuracy. A detailed quantitative evaluation of the higher-order algorithm was performed with emphasis on accuracy, robustness, and computational cost.

Grid convergence studies demonstrate that the new algorithm produces a substantial reduction in the numerical error in drag in comparison with the second-order algorithm for both subsonic and transonic flows. The results show that the higher-order algorithm produces a smaller error on a given grid than the second-order algorithm produces on a grid with several times as many nodes. Hence the higher-order algorithm can provide equivalent accuracy with a large reduction in computing expense. For example, using the higher-order discretization, numerical errors of less than 2-3% can be obtained in the computation of lift and drag components for grids with less than 15,000 nodes for single-element cases and less than 73,000 nodes for a three-element airfoil. The second-order algorithm required 3-4 times as many nodes to achieve similar accuracy.

Compared to the second-order algorithm, the increased cost per grid node per iteration, when using the higher-order algorithm, is approximately 6 - 7%. There is no penalty in memory usage. For single-element cases the lift and drag convergence rates were very similar for both discretization schemes, while the higher-order algorithm converges approximately 33% slower for the multi-element case. The second-order algorithm, however, requires 3 - 4 times as many nodes as the higher-order algorithm to produce similar accuracy. Both schemes prove to be equally robust.

A key aspect of the higher-order algorithm is the consistency of the discretization with respect to accuracy. Accuracy was significantly compromised when low-order and higher-order terms were mixed in some areas of the discretization. In this work, almost all approximations are consistent with third-order global accuracy. The exceptions are the first-order treatment of the convective terms in the Spalart-Allmaras turbulence model, the second-order differences used for the diffusive terms in the Spalart-Allmaras model in TORNADO, and the first-order numerical dissipation added near shocks. The grid convergence studies provide an accurate means to compare the discretization schemes. Comparison of surface pressure and velocity boundary-layer profiles on several grids revealed a number of items:

- It was shown that accuracy was not adversely affected by the first-order terms. In fact, very accurate results were obtained for transonic cases without clustering the grid near the shocks despite using first-order dissipation to capture shocks.

- The grid metrics play a critical role in achieving accurate results. The poor results from the third-order upwind scheme indicate that the metric terms should be raised to the same level of accuracy as the convective terms.

- The higher-order discretization of the viscous terms accounted for approximately 10% of the overall error reduction achieved with the higher-order algorithm relative to to the second-order scheme.

- Prior to this work, whether or not more accurate post-processing was necessary to obtain accurate force and moment coefficients needed to be addressed. The higher-order computation of shear stress proved critical to the accurate prediction of friction drag. The higher-order force integration provided only a small benefit for the cases examined.

5.1 Recommendations for Future Work

This investigation suggests a number of avenues for future work, including the following:

- Solutions to various flows were presented with numerical errors of less than 3% on relatively coarse grids. It is not clear whether it is necessary to progress to even higher orders of accuracy. The next step should be to determine, and address if practical, the largest remaining source of numerical error. For example, the effect of grid singularities on solution accuracy should be investigated.

- Grid convergence studies were used to compare the higher-order algorithm to a number of discretization schemes, all of which are applicable to solving the Navier-Stokes equations on structured grids. It would be informative to see similar studies to assess the relative accuracy of various discretizations on unstructured grids.

- The ability of the higher-order algorithm to obtain accurate results on relatively coarse grids has been demonstrated. Extension to three dimensions should be carried out.

- The higher-order algorithm should be combined with modern convergence acceleration techniques such as multigrid or GMRES.

- There remains a need for efficient error estimation techniques.

References

[1] Abarbanel, S., and Kumar, A., "Compact High-Order Schemes for the Euler Equations," *SIAM J. Sci. Comput.*, vol. 3, no. 3, pp. 275–288, 1988.

[2] Agrawal, S., and Narducci, R., "Drag Prediction using Nonlinear Computational Methods." AIAA Paper 2000-0382, January 2000.

[3] Allmaras, S.R., "Contamination of Laminar Boundary Layers by Artificial Dissipation in Navier-Stokes Solutions," in *Numerical Methods for Fluid Dynamics* (M.J. Baines and K.W. Morton, ed.), pp. 443–449, UK: Oxford, 1993.

[4] Baldwin, B., and Lomax, H., "Thin-Layer Approximation and Algebraic Model for Separated Turbulent Flows." AIAA Paper 78-257, January 1978.

[5] Baldwin, B. S., and Barth, T. J., "A One-Equation Turbulence Transport Model for High Reynolds Number Wall-Bounded Flows." NASA TM 102847, August 1990.

[6] Barth, T.J., and Fredrickson, P.O., "Higher Order Solution of the Euler Equations on Unstructured Grids Using Quadratic Reconstruction." AIAA Paper 90-0013, January 1990.

[7] Bayliss, A., Parikh, P., Maestrello, L., Turkel, E., "A Fourth-Order Scheme for the Unsteady Compressible Navier-Stokes Equations." AIAA Paper 85-1694, July 1985.

[8] Beam, R.M., and Warming, R.F., "An Implicit Factored Scheme for the Compressible Navier-Stokes Equations," *AIAA Journal*, vol. 26, no. 4, pp. 393–402, 1978.

[9] Carpenter, M.H., and Casper, J.H., "The Accuracy of Shock Capturing In Two Spatial Dimensions." AIAA Paper 97-2107, June 1997.

[10] Cebeci, T., "Calculation of Compressible Turbulent Boundary Layers with Heat and Mass Transfer." AIAA Paper 70-741, June 1970.

[11] Cook, P.H., and MacDonald, M.A., and Firmin, M.C.P., "Aerofoil RAE 2822 - Pressure Distributions, and Boundary-Layer and Wake Measurements." AGARD-AR-138, May 1979.

[12] Cosner, R.R., "Assessment of Vehicle Performance Prediction Using CFD." AIAA Paper 2000-0384, January 2000.

[13] De Rango, S., "Implicit Navier-Stokes Computations of Unsteady Flows Using Subiteration Methods.," Master's thesis, University of Toronto, January 1996.

[14] De Rango, S., and Zingg, D.W., "CYCLONE User's Manual." University of Toronto, Institute for Aerospace Studies, 1997.

[15] De Rango, S., and Zingg, D.W., "Improvements to a Dual-Time-Stepping Method for Computing Unsteady Flows," *AIAA Journal*, vol. 35, pp. 1548–1550, September 1997.

[16] De Rango, S., and Zingg, D.W., "Aerodynamic Computations Using a Higher-Order Algorithm." AIAA Paper 99-0167, January 1999.

[17] De Rango, S., and Zingg, D.W., "TORNADO User's Manual." University of Toronto, Institute for Aerospace Studies, 1999.

[18] Ekaterinaris, J.A., "Effects of Spatial Order of Accuracy on the Computation of Vortical Flowfields," *AIAA Journal*, vol. 32, no. 12, pp. 2471–2474, 1994.

[19] Ekaterinaris, J.A., "Implicit, High-Resolution, Compact Schemes for Gas Dynamics and Aeroacoustics," *J. Comp. Phys.*, vol. 156, no. 2, pp. 272–299, 1999.

[20] Frew, K., Zingg, D.W., and De Rango, S., "On Artificial Dissipation Models for Viscous Airfoil Computations," *AIAA Journal*, vol. 36, pp. 1732–1734, September 1998.

[21] Fornberg, B., "On a Fourier Method for the Integration of Hyperbolic Equations," *SIAM J. on Numerical Analysis*, vol. 12, no. 4, pp. 509–528, 1975.

[22] Ghosal, S., "An Analysis of Numerical Errors in Large-Eddy Simulations of Turbulence," *J. Comp. Phys.*, vol. 125, pp. 187–206, 1996.

[23] Goorjian, P.M. and Obayashi, S., "High-Order Accuracy for Upwind Methods by Using the Compatibility Equations," *AIAA Journal*, vol. 31, no. 2, pp. 251–256, 1993.

[24] Gregory, N., and O'Reilly, C.L., "Low-Speed Aerodynamic Characteristics of NACA 0012 Airfoil Section, Including the Effects of Upper-Surface Roughness Simulating Hoar Frost." Aeronautical Research Council, Reports and Memoranda No. 3726, U.K., Jan. 1970.

[25] Gustafsson, B., "The Convergence Rate for Difference Approximations to Mixed Initial Boundary Value Problems," *Math. Comp.*, vol. 29, no. 130, pp. 396–406, 1975.

[26] Hayder, M.E., Turkel, E., "High Order Accurate Solutions of Viscous Problems." AIAA Paper 93-3074, July 1993.

[27] Jameson A., "Computational Aerodynamics for Aircraft Design," *Science*, vol. 245, pp. 361–371, 1989.

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

[29] Jespersen, D., Pulliam, T., and Buning, P., "Recent Enhancements to OVERFLOW." AIAA Paper 97-0644, January 1997.

[30] Jurgens, H., and Zingg, D.W., "Implementation of a High-Accuracy Finite-Difference Scheme for Linear Wave Phenomena." Procedings of the International Conference on Spectral and High Order Methods, June 1995.

[31] Kravchenko, A.G., and Moin, P., "On the Effect of Numerical Errors in Large Eddy Simulations of Turbulent Flows," *J. Comp. Phys.*, vol. 131, pp. 310–322, 1997.

[32] Kreiss, H.-O., and Oliger, J., "Comparison of Accurate Methods for the Integration of Hyperbolic Equations," *Tellus*, vol. 24, no. 3, pp. 199–215, 1972.

[33] Lele, S.K., "Compact Finite Difference Schemes with Spectral-Like Resolution," *J. Comp. Phys.*, vol. 103, pp. 16–42, 1992.

[34] Mahesh, K., "A Family of High Order Finite Difference Schemes with Good Spectral Resolution," *J. Comp. Phys.*, vol. 145, pp. 332–358, 1998.

[35] Maksymiuk, C.M., and Swanson, R.C., and Pulliam, T.H., "A Comparison of Two Central Difference Schemes for for Solving the Navier-Stokes Equations." NASA TM-102815, July 1990.

[36] Malcolm, M.A., and Simpson, R.B., "Local Versus Global Strategies for Adaptive Quadrature." ACM Transactions on Mathematical Software, Vol. 1, No. 2, June 1975.

[37] Nelson, T. E., *Numerical Solution of the Navier-Stokes Equations for High-Lift Airfoil Configurations.* PhD thesis, University of Toronto, May 1994.

[38] Nelson, T.E., and Godin, P., and De Rango, S., and Zingg, D.W., "Flow Computations For A Three-element Airfoil System," *CASI J.*, vol. 45, no. 2, pp. 132–139, June 1999.

[39] Nemec, M., and Zingg, D.W., "Aerodynamic Computations Using the Convective-Upstream Split Pressure Scheme with Local Preconditioning," *AIAA Journal*, vol. 38, no. 3, pp. 402–410, 2000.

[40] Pueyo, A., and Zingg, D.W., "Efficient Newton-Krylov Solver for Aerodynamic Computations," *AIAA Journal*, vol. 36, no. 11, pp. 1991–1997, 1998.

[41] Pulliam, T. H., "Efficient Solution Methods for the Navier-Stokes Equations." Lecture Notes For The von Karman Institute, for Fluid Dynamics Lecture Series: *Numerical Techniques For Viscous Flow Computation In Turbomachinery Bladings*, von Karman Institute, Brussels, Belgium, Jan. 1986.

[42] Pulliam, T.H., "Artificial Dissipation Models for the Euler Equations," *AIAA Journal*, vol. 24, no. 12, pp. 1931–1940, 1986.

[43] Pulliam, T.H., and Chaussee, D.S, "A Diagonal Form of an Implicit Approximate Factorization Algorithm," *J. Comp. Phys.*, vol. 39, pp. 347–363, 1981.

[44] Rai, M.M., and Moin, P., "Direct Numerical Simulation of Transition and Trubulence in a Spatially Evolving Boundary Layer," *J. Comp. Phys.*, vol. 109, no. 2, pp. 169–192, 1993.

[45] Rai, M.M., Gatski, T.B., and Erlebacher, G., "Direct Simulation of Spatially Evolving Compressible Turbulent Boundary Layers." AIAA Paper 95-0583, January 1995.

[46] Rangwalla, A.A., and Rai, M.M., "A Multi-Zone High-Order Finite-Difference Method for the Navier-Stokes Equations." AIAA Paper 95-1706, June 1995.

[47] Roache, P.J., *Verification And Validation In Computational Science And Engineering*. Hermosa Publishing, New Mexico, 1998.

[48] Roache, P.J., "Verification of Codes and Calculations," *AIAA Journal*, vol. 36, pp. 696–702, May 1998.

[49] Roe, P.L., "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," *J. Comp. Phys.*, vol. 43, pp. 357–372, 1981.

[50] Rogers, S.E., Kwak, D., and Kiris, C., "Steady and Unsteady Solutions of the Incompressible Navier-Stokes Equations," *AIAA Journal*, vol. 29, pp. 603–610, April 1991.

[51] Salas, M., Jameson, A., and Melnik, R., "A Comparative Study of the Nonuniqueness Problem of the Potential Equation." AIAA Paper 83-1888, June 1983.

[52] Sjogreen, B., "High Order Centered Difference Methods for the Compressible Navier-Stokes Equations," *J. Comp. Phys.*, vol. 117, pp. 67–78, 1995.

[53] Spalart, P. R., and Allmaras, S. R., "A One-Equation Turbulence Model for Aerodynamic Flows." AIAA Paper 92-0439, January 1992.

[54] Swanson, R.C., and Turkel, E., "On Central-Difference and Upwind Schemes," *J. Comp. Phys.*, vol. 101, pp. 292–306, 1992.

[55] Swartz, B., and Wendroff, B., "The Relative Efficiency of Finite Difference and Finite Element Methods," *SIAM J. on Numerical Analysis*, vol. 11, no. 5, pp. 979–993, 1974.

[56] Tatsumi, S., and Martineeli, L., and Jameson, A., "A New High Resolution Scheme for Compressible Viscous Flow with Shocks." AIAA Paper 95-0466, Jan. 1995.

[57] Thivet, F., Besbes, O., and Knight, D.D., "Effect of Grid Resolution on Accuracy of Skin Friction and Heat Transfer in Turbulent Boundary Layers." AIAA Paper 2000-0820, January 2000.

[58] Tolstykh, A.I., and Lipavskii, V., "On Performance of Methods with Third- and Fifth-Order Compact Upwind Differencing," *J. Comp. Phys.*, vol. 140, pp. 205–232, 1998.

[59] Treidler, E.B., and Childs, R.E., "High-Accuracy Spatial Discretization Schemes for CFD." AIAA Paper 97-0541, January 1997.

[60] Visbal, M.R., and Gaitonde, D.V, "High-Order Accurate Methods for Complex Unsteady Subsonic Flows," *AIAA Journal*, vol. 37, no. 10, pp. 1231–1239, 1999.

[61] Wake, B.E., and Choi, D., "Investigation of High-Order Upwinded Differencing for Vortex Convection," *AIAA Journal*, vol. 34, no. 2, pp. 332–337, 1996.

[62] Wells, V.L., and Renaut, R.A., "Computing Aerodynamically Generated Noise," in *Annu. Rev. Fluid Mech.*, vol. 29, pp. 161–199, 1997.

[63] Yee, H.C., "Explicit and Implicit Multidimensional Compact High-Resolution Shock-Capturing Methods: Formulation," *J. Comp. Phys.*, vol. 131, pp. 216–232, 1997.

[64] Yee, H.C., Snadham, N.D., and Djomehri, M.J., "Low-Dissipative High-Order Shock-Capturing Methods Using Characteristic-Based Filters," *J. Comp. Phys.*, vol. 150, pp. 199–238, 1999.

[65] Zhong, X., "High-Order Finite-Difference Schemes for Numerical Simulation of Hypersonic Boundary-Layer Transition," *J. Comp. Phys.*, vol. 144, no. 2, pp. 662–709, 1998.

[66] Zingg, D. W., "Grid Studies for Thin-Layer Navier-Stokes Computations of Airfoil Flowfields," *AIAA Journal*, vol. 30, pp. 2561–2564, Oct. 1992.

[67] Zingg, D.W., "Comparison of High-Accuracy Finite-Difference Methods for Linear Wave Propagation," *SIAM J. Sci. Comput.*, vol. 22, no. 2, pp. 476–502, 2000.

[68] Zingg, D.W., and De Rango, S., and Nemec, M., "An Investigation of Numerical Errors in Aerodynamic Flow Computations." 6th Annual Meeting of the CFD Society of Canada, Quebec, 1998.

[69] Zingg, D.W., and De Rango, S., Nemec, M., and Pulliam, T.H., "Comparison of Several Discretizations for the Navier-Stokes Equations," *J. Comp. Phys.*, vol. 160, no. 2, pp. 683–704, 2000.

[70] Zingg, D.W., and Lomax, H., "On the Eigensystems Associated with Numerical Boundary Schemes for Hyperbolic Equations." in *Numerical Methods for Fluid Dynamics* (M. Baines and K. Morton, eds.), vol. 4, p. 473, Oxford University Press, 1993.

Appendices

Appendix A

Circulation Correction

One generally assumes that the flow at the far-field boundaries is uniform and meets free-stream conditions. If those boundaries are placed close to a lifting airfoil, however, disturbances from the airfoil surface may not have settled down to free-stream conditions by the time they reach the far-field boundaries. Hence, imposing free-stream conditions under those circumstances would undoubtedly affect the physics of the flow. One way to alleviate this problem is to place the far-field boundaries very far away from any lifting bodies. That solution is impractical since it would require many more grid nodes. Pulliam [41] shows that boundaries as much as 96 chords away are needed to minimize the effect of the boundaries on the accuracy of the solution. Following the work of Salas et al. [51], Pulliam added a compressible potential vortex solution as a perturbation to the free-stream velocity giving

$$u_f = u_\infty + \frac{\beta\Gamma\sin(\theta)}{2\pi r[1 - M_\infty^2\sin^2(\theta-\alpha)]} \tag{A.1}$$

$$v_f = v_\infty - \frac{\beta\Gamma\cos(\theta)}{2\pi r[1 - M_\infty^2\sin^2(\theta-\alpha)]} \tag{A.2}$$

where $\Gamma = \frac{1}{2}M_\infty c C_l$, c is the chord of the airfoil, C_l is the coefficient of lift, M_∞ the free-stream Mach number, α the angle of attack, $\beta = \sqrt{1-M_\infty^2}$ and r and θ are the polar coordinates to the point of application on the far-field boundary relative to the quarter-chord point on the airfoil chord line. The speed of sound is also corrected to enforce constant free-stream enthalpy at the boundary as follows:

$$a_f^2 = (\gamma - 1)\left(H_\infty - \frac{1}{2}(u_f^2 + v_f^2)\right) \tag{A.3}$$

Using this far-field vortex correction, Zingg [66] was able to produce very good results in drag on grids with a grid extent of 12 chords. The results were compared to solutions obtained on grids with far-field boundaries set at 96 chords.

Appendix B

Figures

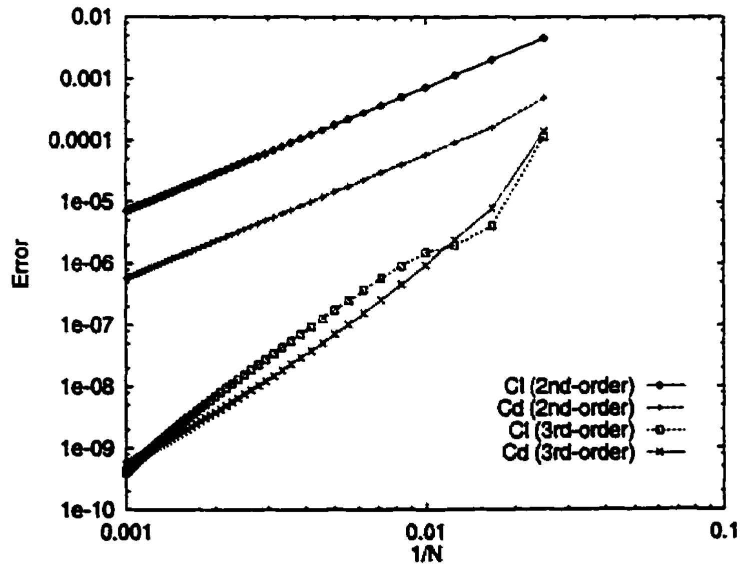
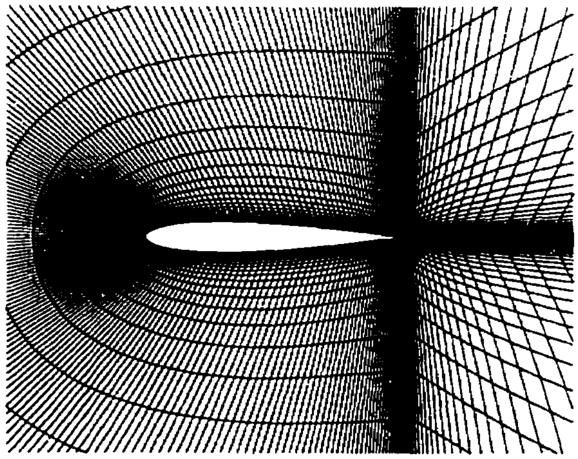
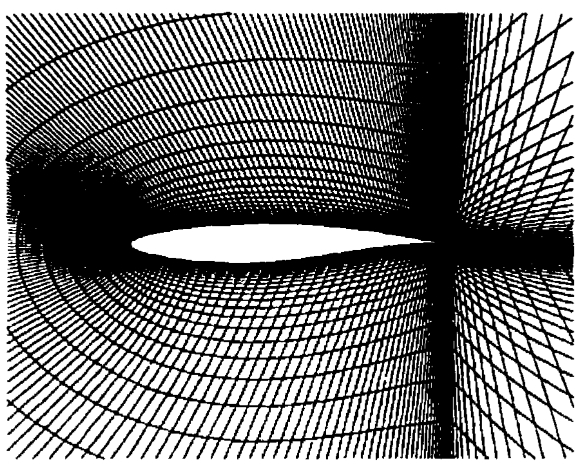


Figure B.1: Error in force integration algorithm. Note: error is realtive to analytical lift and drag coefficients which are $C_l = 1.218153560$ and $C_d = 0.0$.

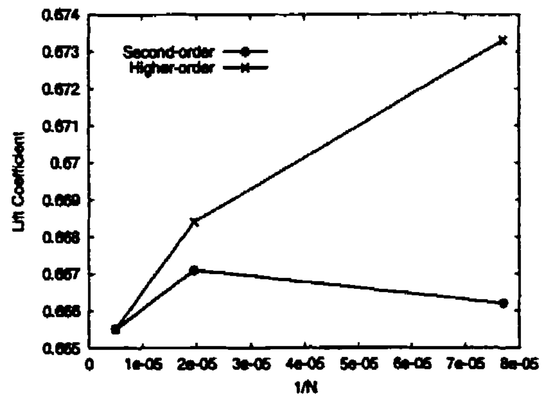


(a) Close-up of NACA 0012 airfoil (Grid C1).

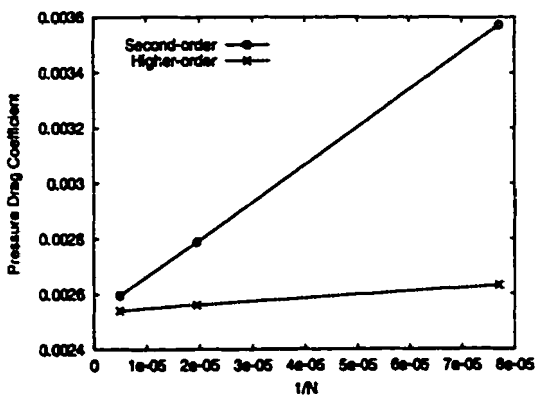


(b) Close-up of RAE 2822 airfoil (Grid C2).

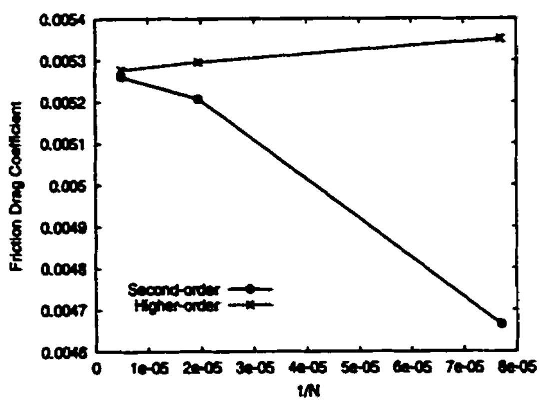
Figure B.2: Airfoil geometries and sample grids used with CYCLONE.



(a) C_l

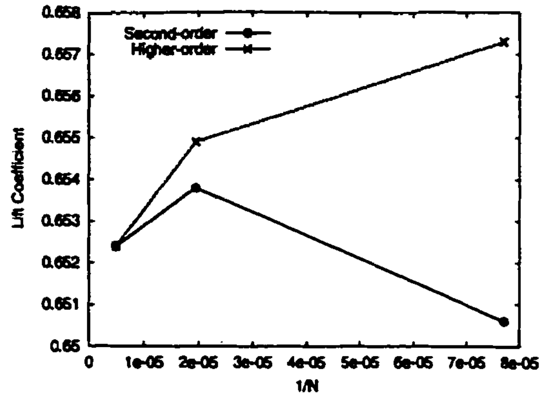


(b) C_{d_p}

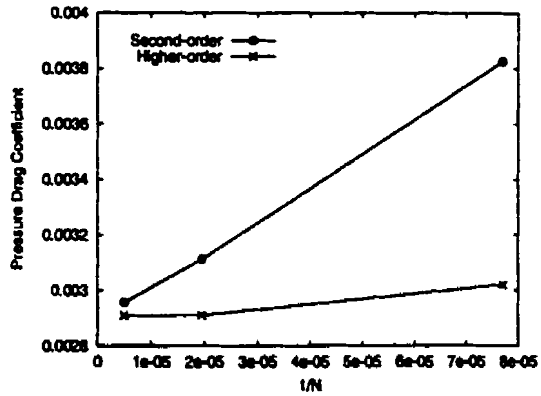


(c) C_{d_f}

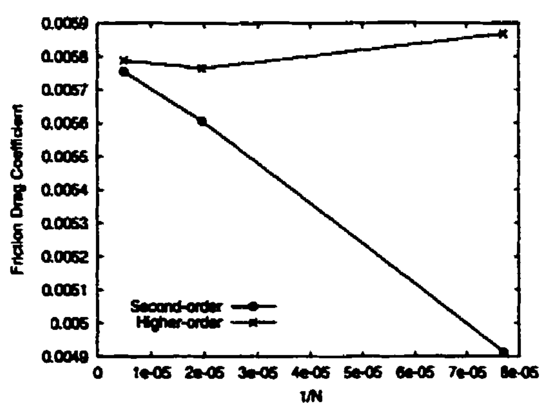
Figure B.3: Grid convergence study for case 1 using grid family 1 (Baldwin-Lomax model).



(a) C_l

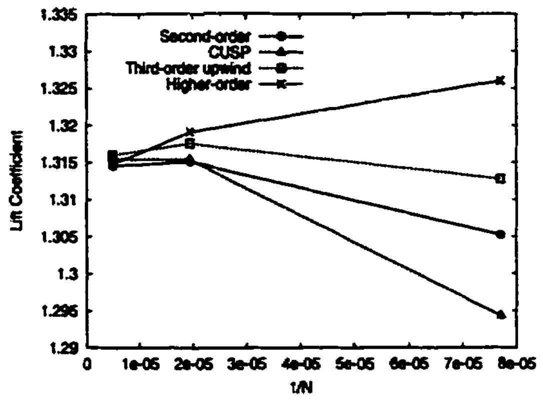


(b) C_{d_p}

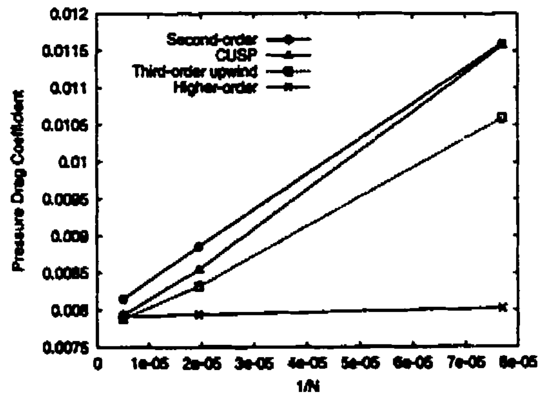


(c) C_{d_f}

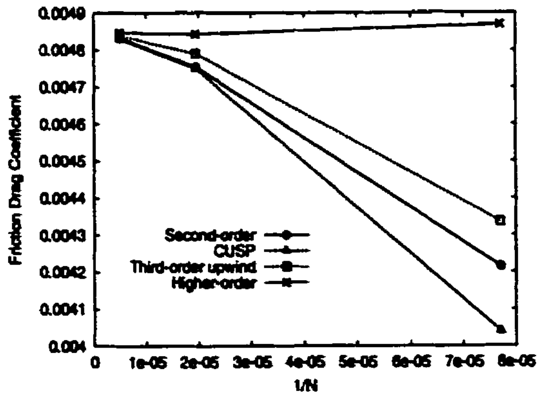
Figure B.4: Grid convergence study for case 1 using grid family 1 (Spalart-Allmaras model).



(a) C_l



(b) C_{d_p}



(c) C_{d_f}

Figure B.5: Grid convergence study for case 2 using grid family 1 (Baldwin-Lomax model).

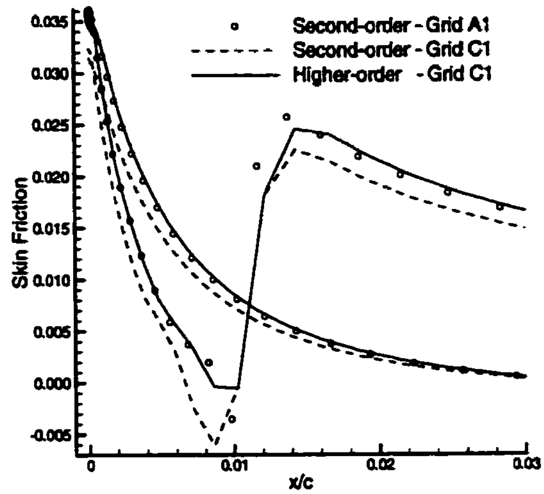


Figure B.6: Skin-friction distribution near leading-edge, case 2 (Baldwin-Lomax).

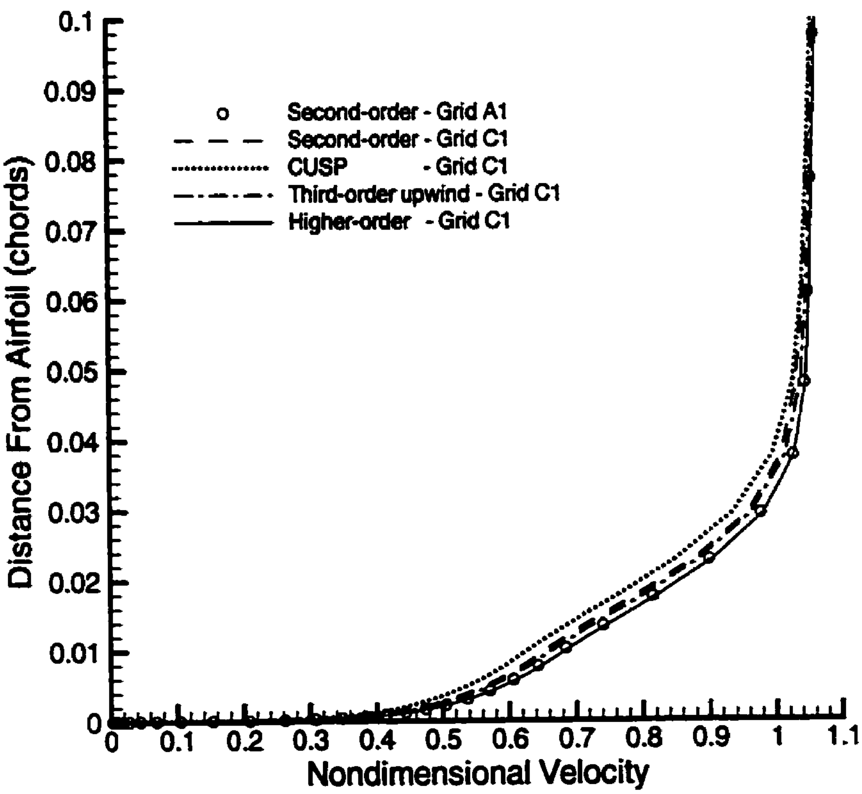


Figure B.7: Boundary-layer velocity profiles on the upper surface at 85% chord, case 2 (Baldwin-Lomax).

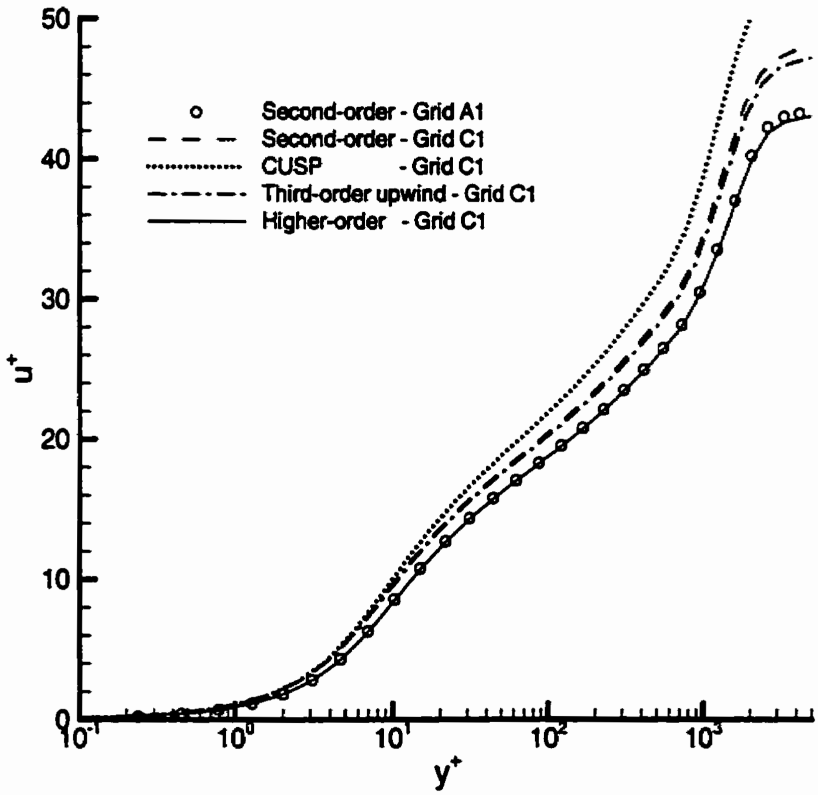
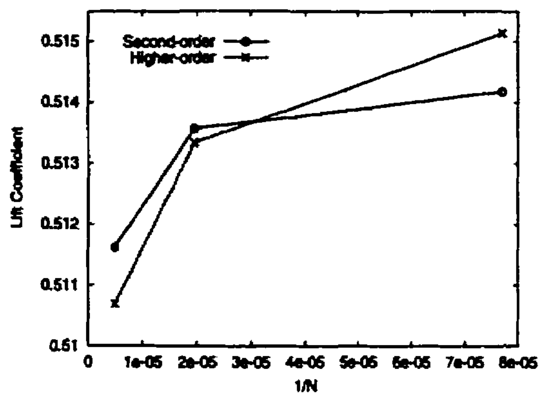
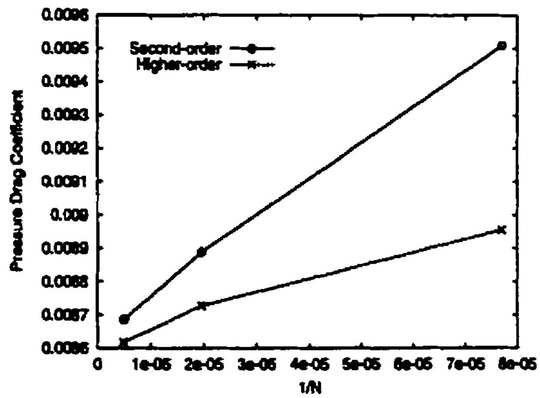


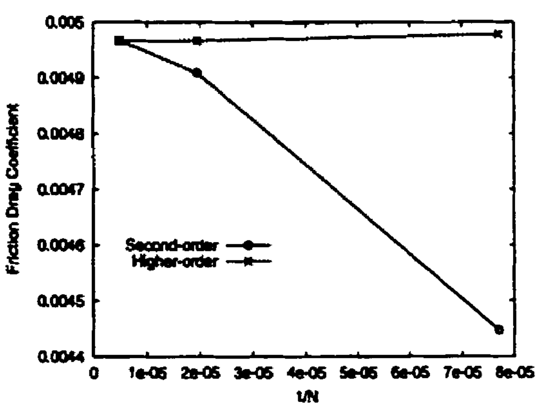
Figure B.8: Boundary-layer profiles of $u^{+}=\frac{u}{u_\tau}$ vs. $y^{+}=\frac{u_\tau y}{\nu_w}$ on the upper surface at 85% chord, case 2 (Baldwin-Lomax).



(a) C_l



(b) C_{d_p}



(c) C_{d_f}

Figure B.9: Grid convergence study for case 3 using grid family 1 (Baldwin-Lomax model).

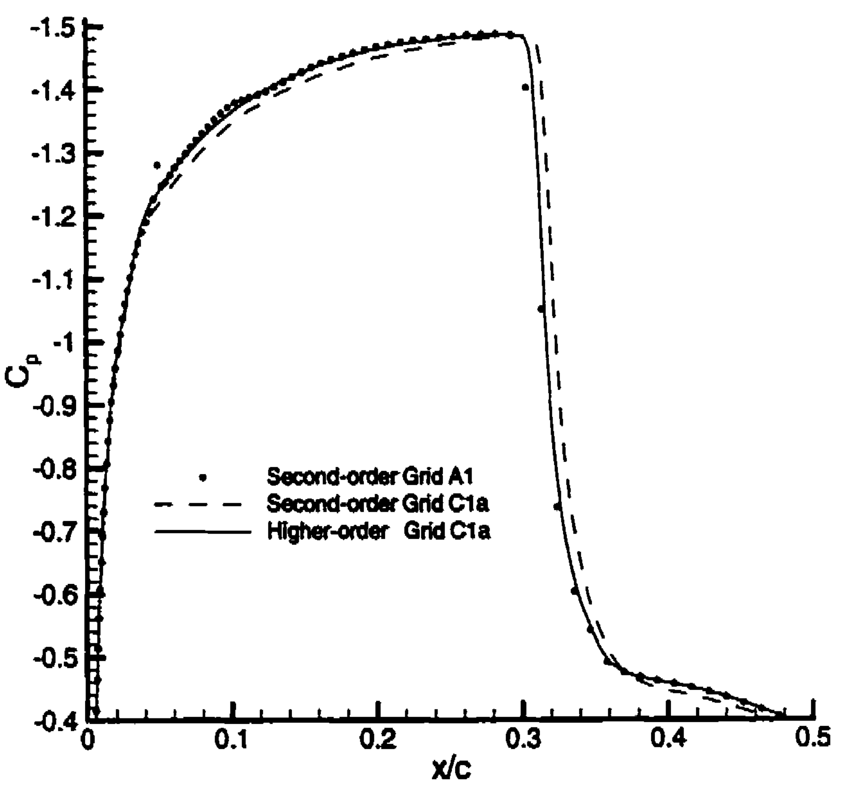
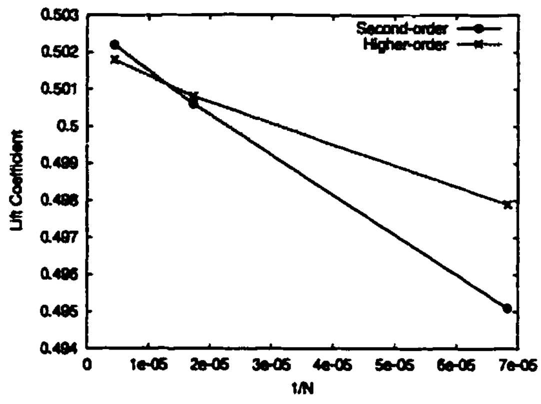
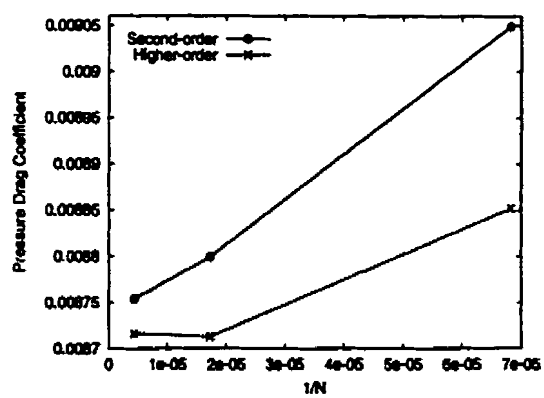


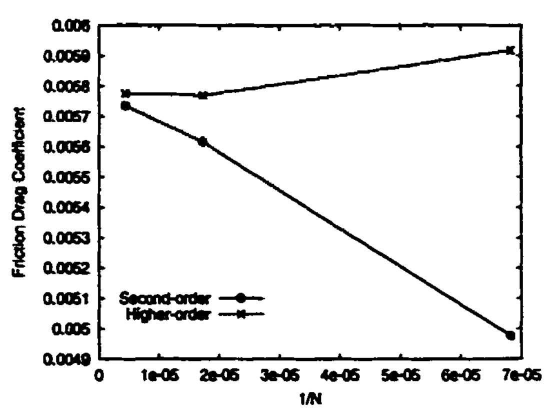
Figure B.10: Surface pressure distribution for case 3, computed on grid C1a (Baldwin-Lomax).



(a) C_l

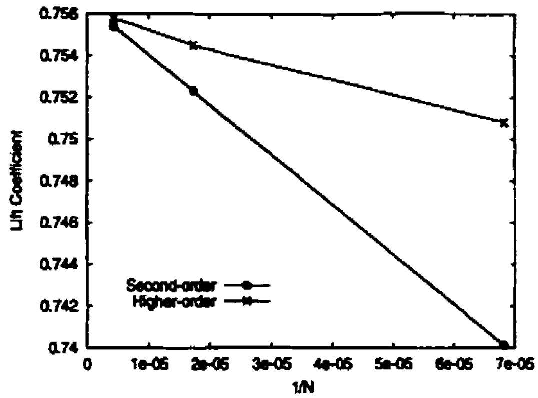


(b) C_{d_p}

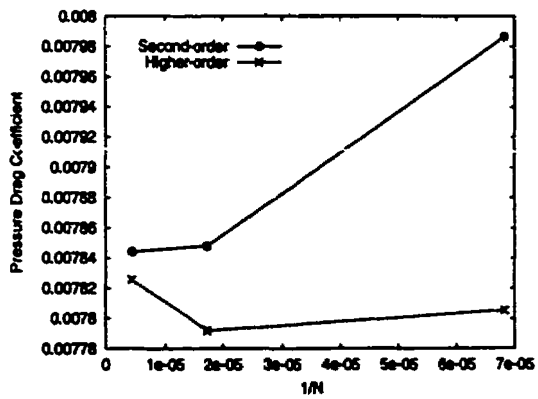


(c) C_{d_f}

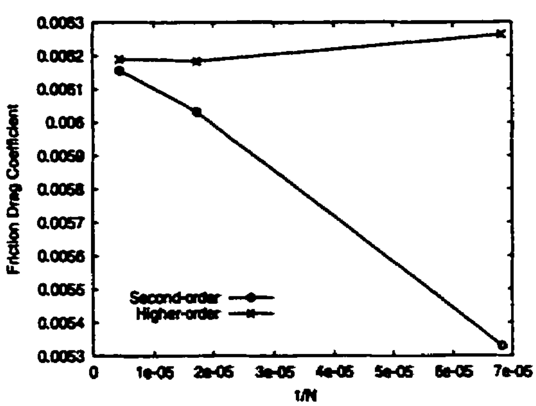
Figure B.11: Grid convergence study for case 3 using grid family 2 (Spalart-Allmaras model).



(a) C_l

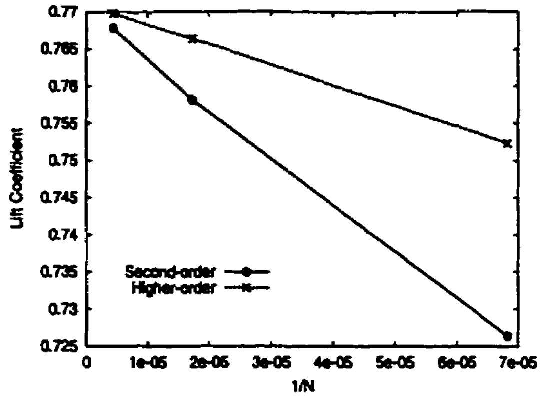


(b) C_{d_p}

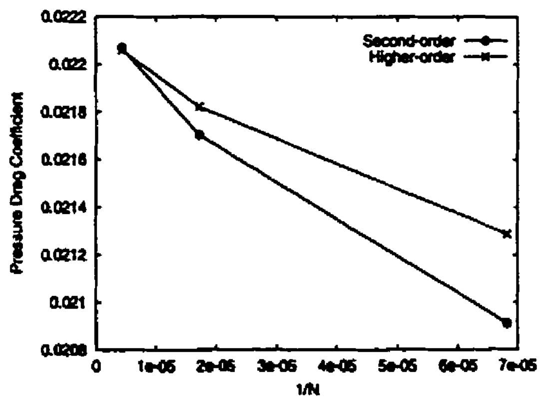


(c) C_{d_f}

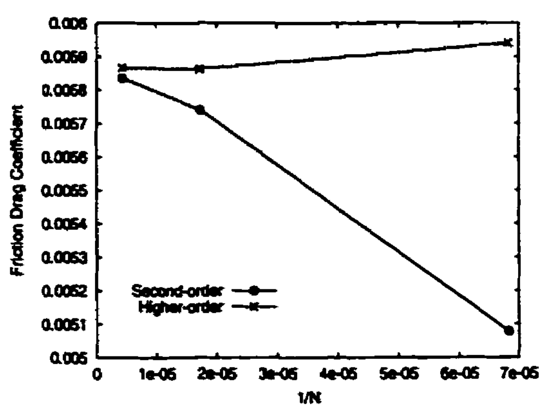
Figure B.12: Grid convergence study for case 4 using grid family 2 (Spalart-Allmaras model).



(a) C_l



(b) C_{d_p}



(c) C_{d_f}

Figure B.13: Grid convergence study for case 5 using grid family 2 (Spalart-Allmaras model).

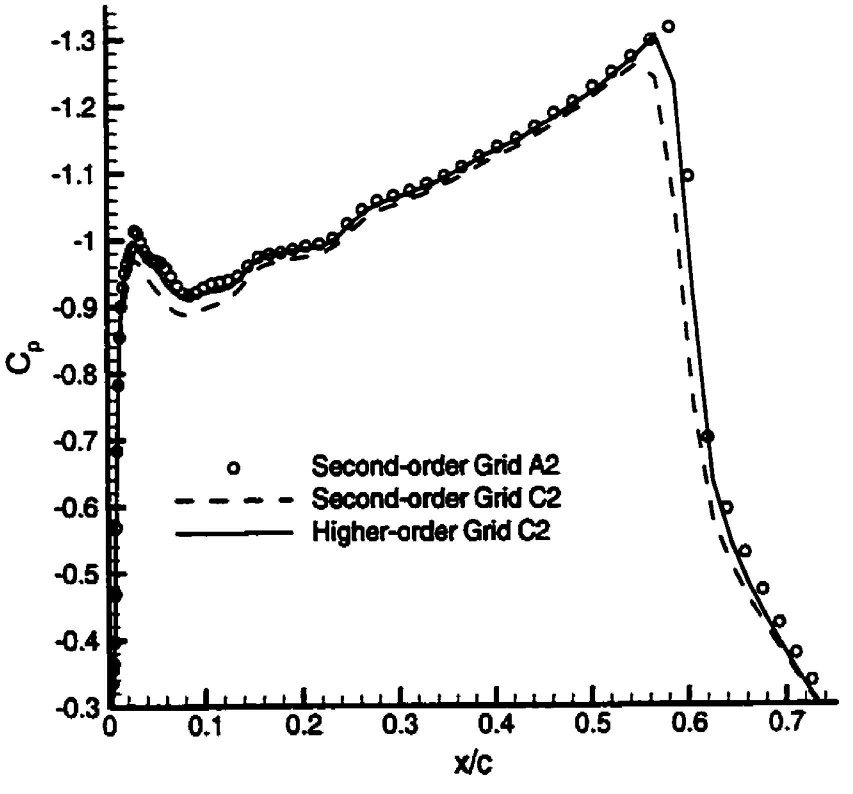


Figure B.14: Surface pressure distribution for case 5, computed on grid C2 (Spalart-Allmaras model).

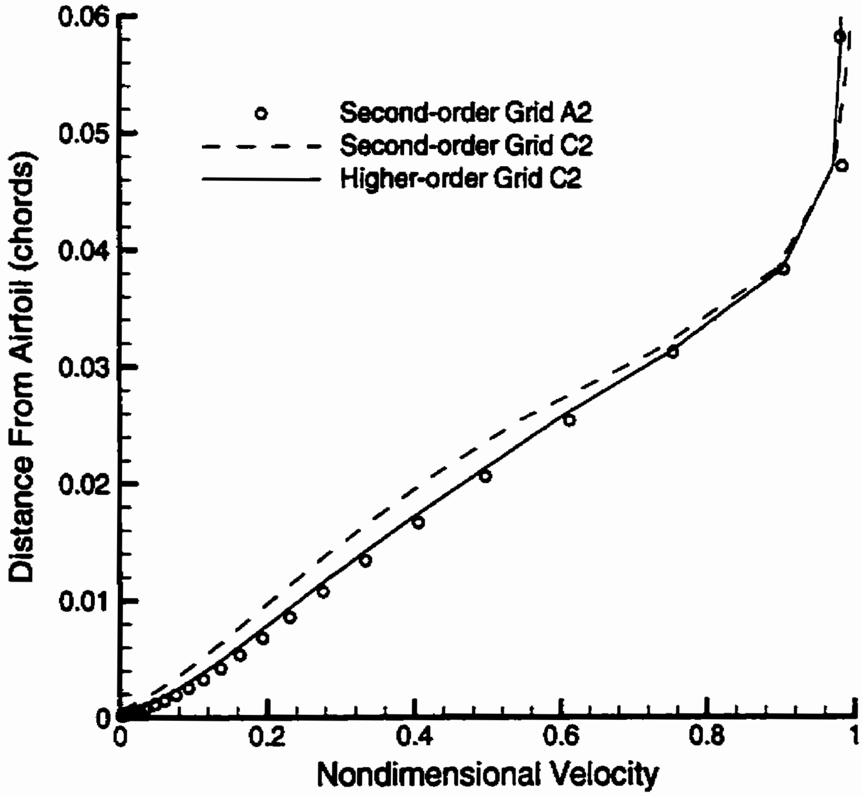


Figure B.15: Boundary-layer profile for case 5, upper surface at 95% chord (Spalart-Allmaras model).

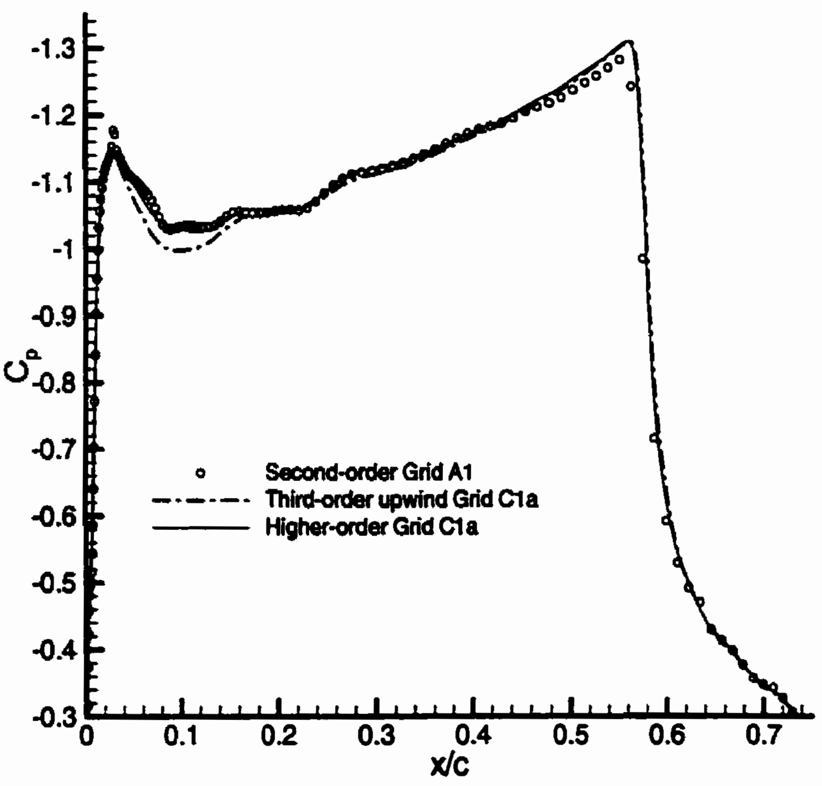
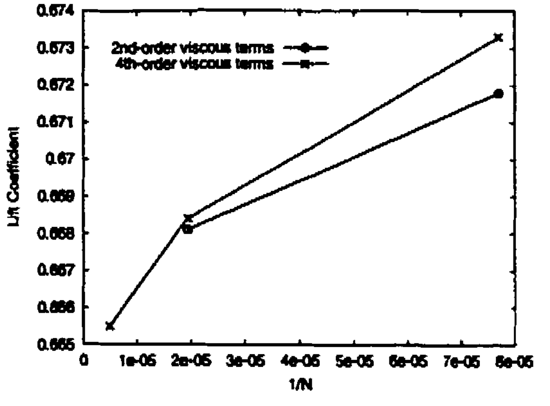
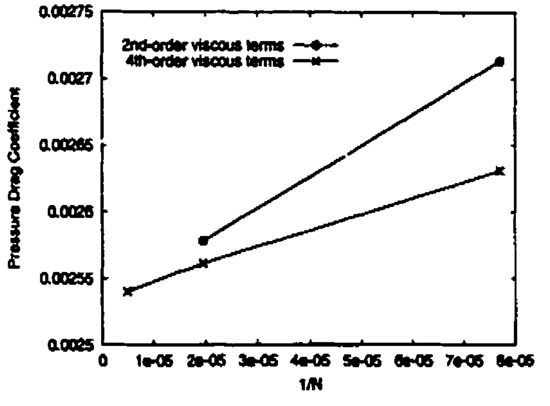


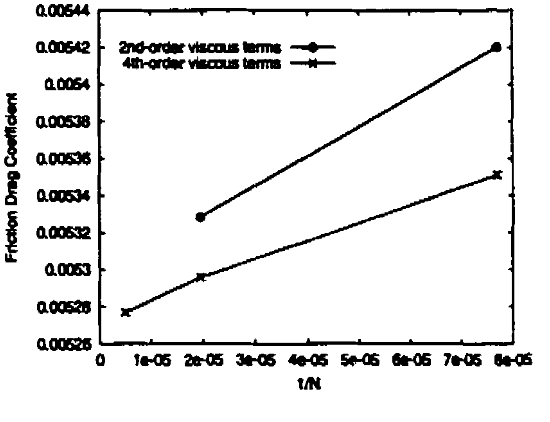
Figure B.16: Surface pressure distribution for case 4, computed on grid C1a (Baldwin-Lomax model).



(a) C_l



(b) C_{d_p}



(c) C_{d_f}

Figure B.17: Effect of viscous terms on accuracy of higher-order algorithm for case 1 (Baldwin-Lomax model).

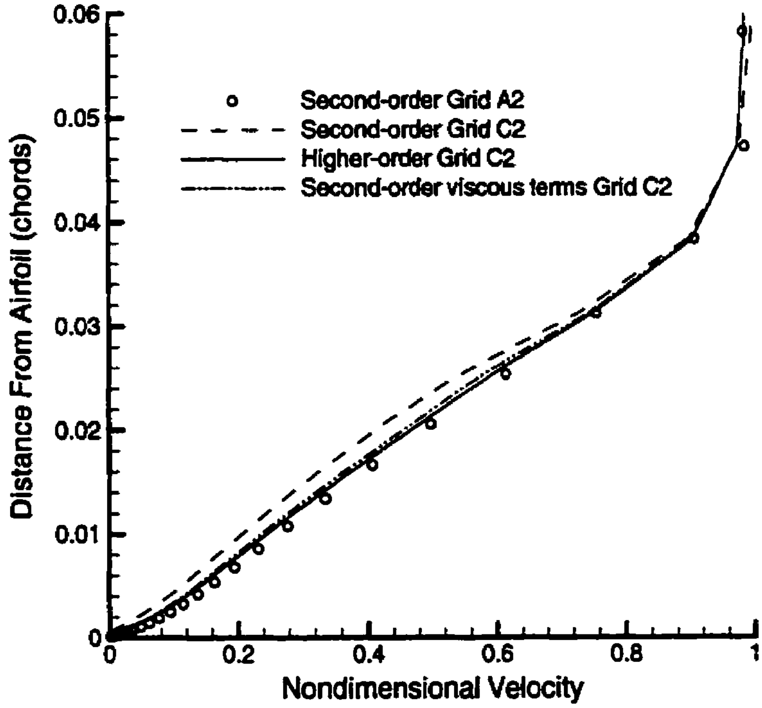


Figure B.18: Boundary-layer profile for case 5, upper surface at 95% chord (Spalart-Allmaras).

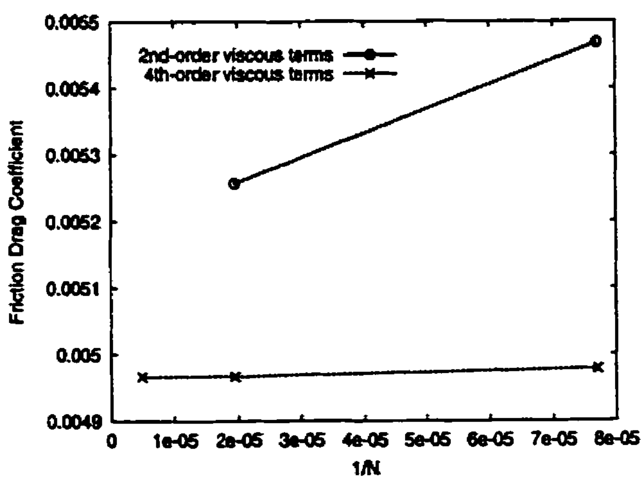
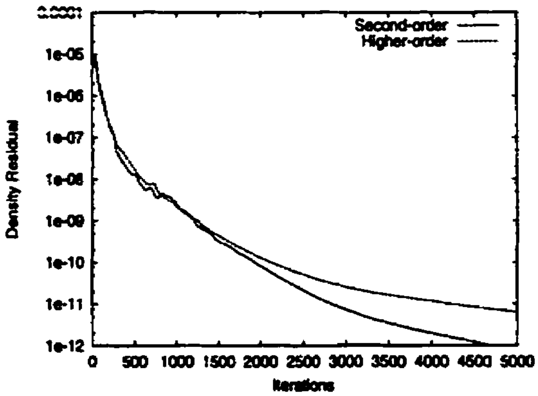
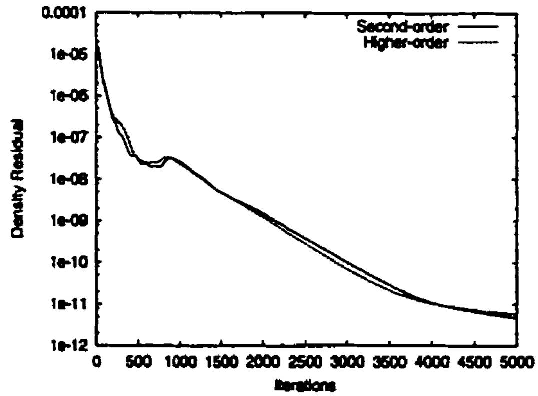


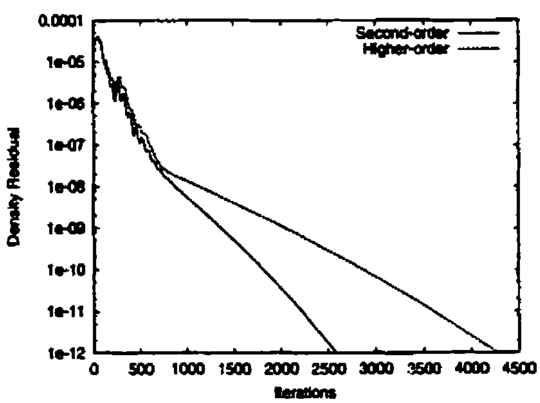
Figure B.19: Effect of viscous terms on accuracy of higher-order algorithm for case 3 (Baldwin-Lomax model).



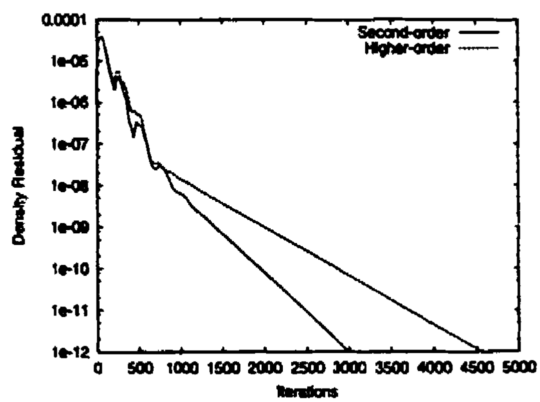
(a) Case 1 - Grid C1



(b) Case 2 - Grid C1

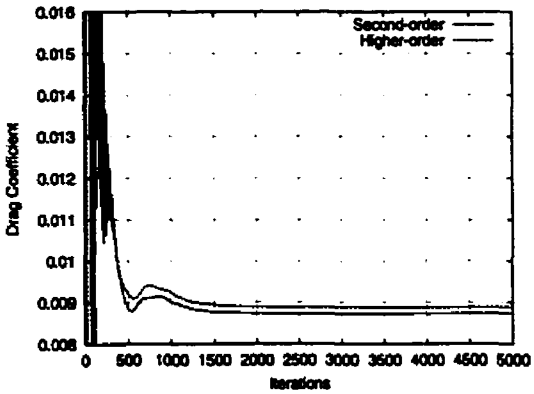


(c) Case 3 - Grid C2

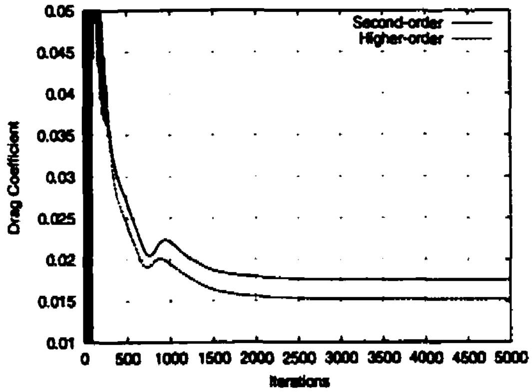


(d) Case 4 - Grid C2

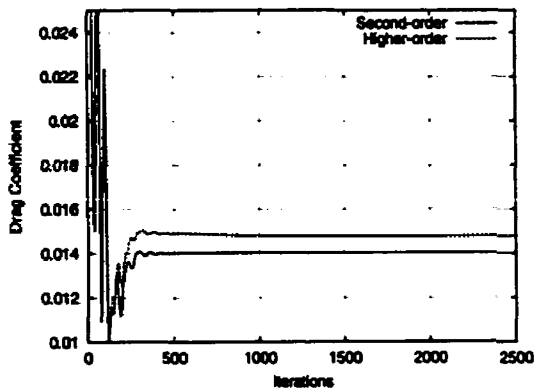
Figure B.20: Residual convergence histories (Spalart-Allmaras model).



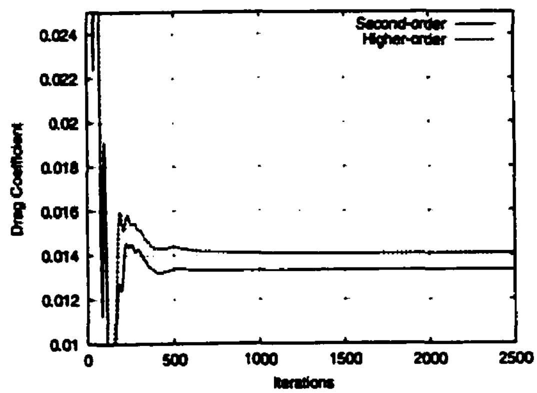
(a) Case 1 - Grid C1



(b) Case 2 - Grid C1



(c) Case 3 - Grid C2



(d) Case 4 - Grid C2

Figure B.21: Drag convergence histories (Spalart-Allmaras model).

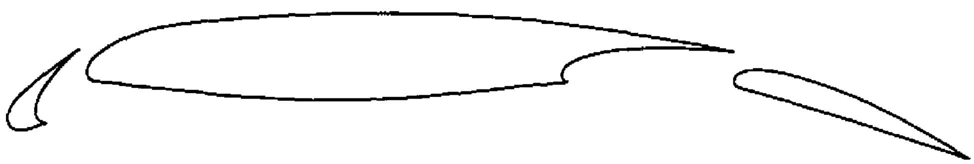


Figure B.22: High-lift test case A2.

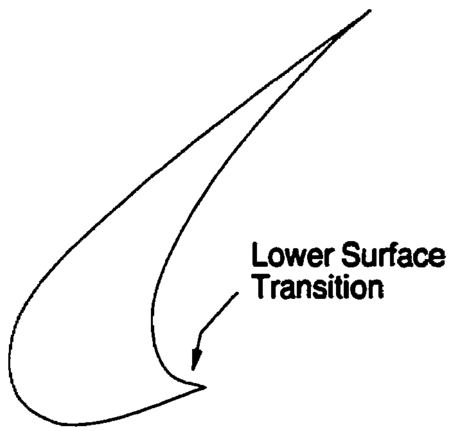


Figure B.23: Location of fixed transition point on lower surface of slat for case 6.

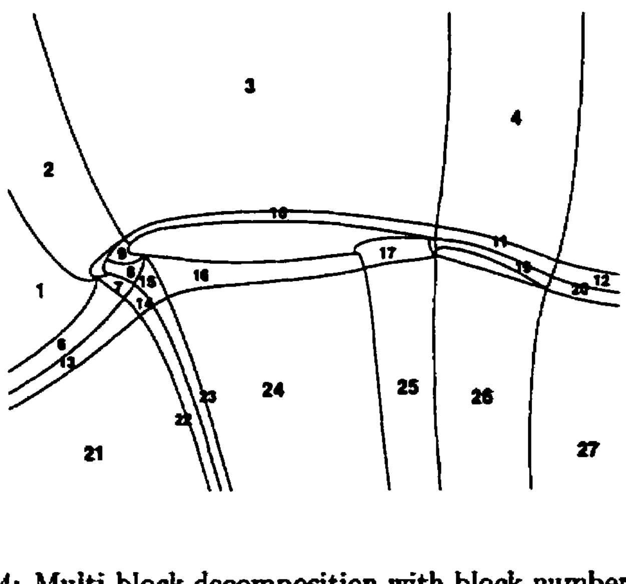
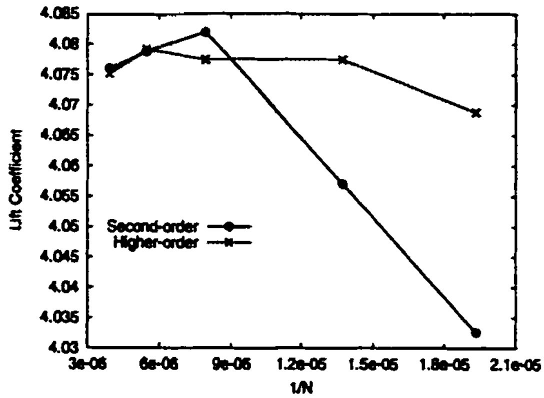
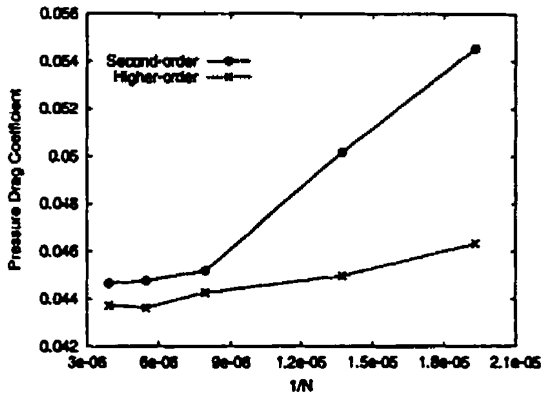


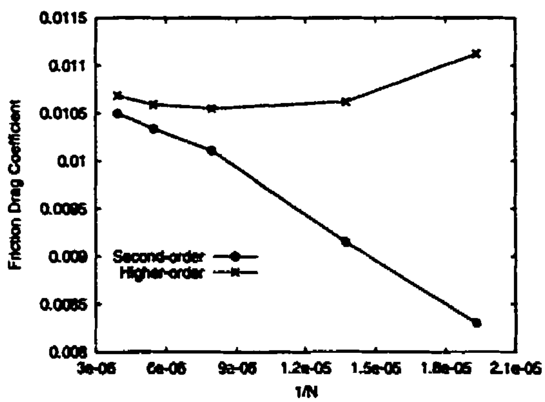
Figure B.24: Multi-block decomposition with block number for case 6.



(a) C_l



(b) C_{d_p}



(c) C_{d_f}

Figure B.25: Grid convergence study for case 6.

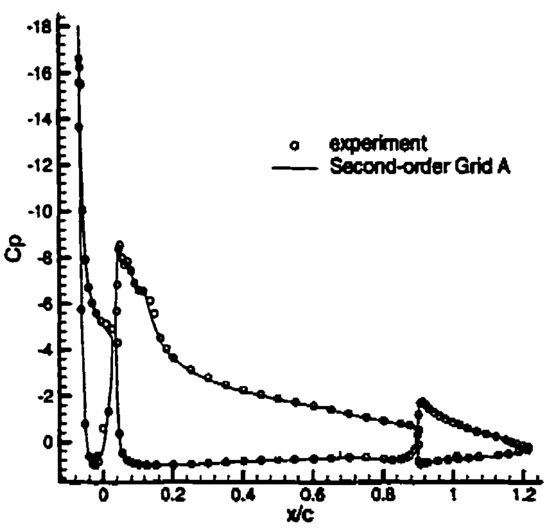


Figure B.26: Pressure distribution for the NHLP 2D configuration L1T2 (case 6).

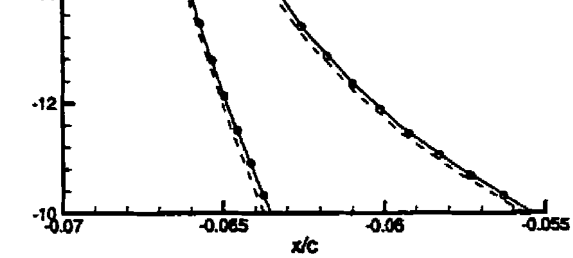


Figure B.27: Pressure distribution on upper surface of slat for case 6.

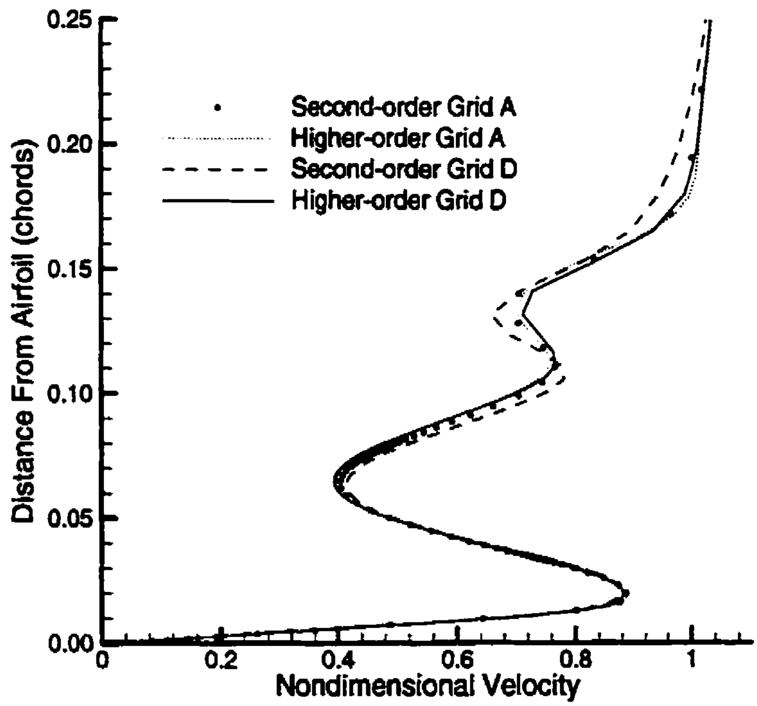
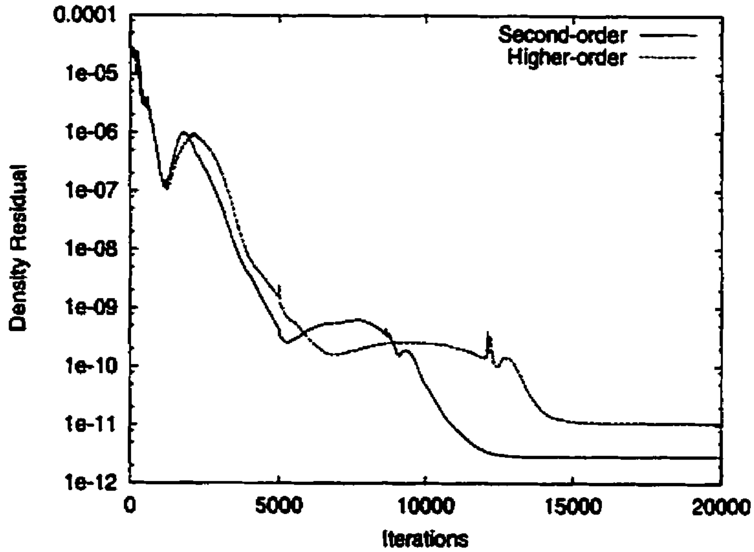
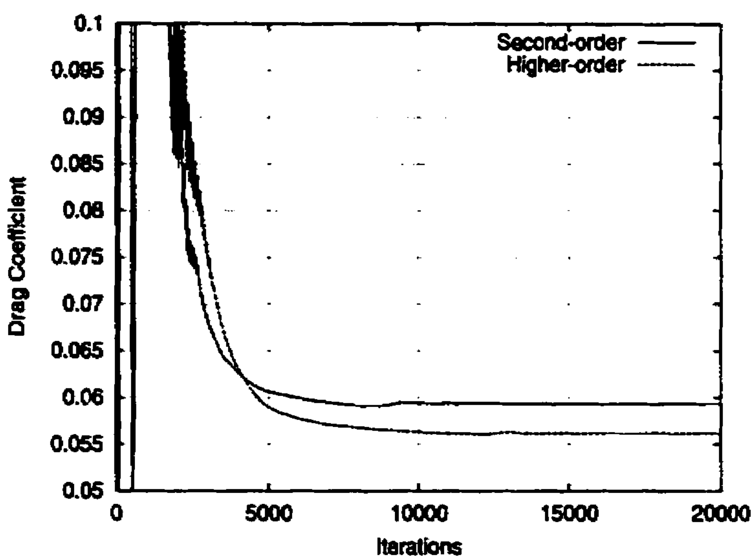


Figure B.28: Boundary-layer profile for case 6, upper surface of flap at trailing edge.



(a)



(b)

Figure B.29: Density residual and drag convergence history on grid D for case 6.

Appendix C

Multi-block Grids

Block	Block Dimensions ($\xi \times \eta$)
1	65×161
2	222×161
3	189×161
4	81×161
5	65×161
6	65×97
7	65×97
8	57×97
9	85×97
10	189×97
11	81×97
12	65×97
13	65×97
14	65×97
15	57×97
16	121×97
17	77×97
18	40×97
19	81×97
20	65×97
21	65×77
22	65×77
23	57×77
24	121×77
25	77×77
26	81×77
27	65×77
total nodes	255,295

Table C.1: Block dimensions for grid A.

Block	Block Dimensions ($\xi \times \eta$)
1	61×129
2	177×129
3	169×129
4	65×129
5	49×129
6	61×93
7	53×93
8	53×93
9	73×93
10	169×93
11	65×93
12	49×93
13	61×77
14	53×77
15	53×77
16	109×77
17	69×77
18	35×77
19	65×77
20	49×77
21	61×65
22	53×65
23	53×65
24	109×65
25	69×65
26	65×65
27	49×65
total nodes	183,721

Table C.2: Block dimensions for grid B.

Block	Block Dimensions ($\xi \times \eta$)
1	57×109
2	149×109
3	141×109
4	41×109
5	41×109
6	57×77
7	45×77
8	45×77
9	49×77
10	141×77
11	41×77
12	41×77
13	57×65
14	45×65
15	45×65
16	89×65
17	57×65
18	35×65
19	41×65
20	41×65
21	57×53
22	45×53
23	45×53
24	89×53
25	57×53
26	53×53
27	41×53
total nodes	126,185

Table C.3: Block dimensions for grid C.

Block	Block Dimensions ($\xi \times \eta$)
1	49×81
2	111×81
3	97×81
4	41×81
5	33×81
6	49×57
7	33×57
8	33×57
9	49×57
10	97×57
11	41×57
12	33×57
13	49×49
14	33×49
15	33×49
16	65×49
17	39×49
18	25×49
19	41×49
20	33×49
21	49×39
22	33×39
23	33×39
24	65×39
25	39×39
26	39×39
27	33×39
total nodes	72,837

Table C.4: Block dimensions for grid D.

Block	Block Dimensions ($\xi \times \eta$)
1	42×69
2	95×69
3	83×69
4	33×69
5	29×69
6	42×49
7	29×49
8	29×49
9	41×49
10	83×49
11	33×49
12	29×49
13	42×37
14	29×37
15	29×37
16	55×37
17	33×37
18	21×37
19	33×37
20	29×37
21	42×33
22	29×33
23	29×33
24	55×33
25	33×33
26	33×33
27	29×33
total nodes	51,749

Table C.5: Block dimensions for grid E.

Appendix D

NHLP Multi-element Airfoil Coordinates

X	Y
0.027257	0.020498
0.020872	0.015563
0.011227	0.006611
-0.004398	-0.009567
-0.012387	-0.019220
-0.019165	-0.028956
-0.023010	-0.035321
-0.027407	-0.044011
-0.029561	-0.050210
-0.030276	-0.059158
-0.028458	-0.067011
-0.024626	-0.071463
-0.018779	-0.073341
-0.016801	-0.073880
-0.025502	-0.076949
-0.035312	-0.080319
-0.042159	-0.082184
-0.049070	-0.083118
-0.055921	-0.082630
-0.059094	-0.081361
-0.062121	-0.079626
-0.066375	-0.074374
-0.067787	-0.068143
-0.067082	-0.061726
-0.066222	-0.058840
-0.063378	-0.053099
-0.059606	-0.047766
-0.053544	-0.041046
-0.046321	-0.034484
-0.034245	-0.024138
-0.019484	-0.012454
-0.066512	-0.002938
0.008815	0.007981
0.014342	0.011734
0.020072	0.015591
0.027257	0.020498

Table D.1: NHLP slat coordinates.

X	Y
0.899869	0.017549
0.871248	0.019768
0.835977	0.021513
0.802387	0.022130
0.766457	0.021215
0.753478	0.020250
0.735188	0.018182
0.729529	0.017310
0.718889	0.015446
0.709570	0.013482
0.701591	0.011279
0.690283	0.007384
0.682294	0.003691
0.676976	0.000089
0.672987	-0.003902
0.669990	-0.011123
0.673323	-0.018802
0.677314	-0.020700
0.668664	-0.021464
0.634736	-0.024607
0.602137	-0.027781
0.567868	-0.031354
0.534940	-0.034498
0.500681	-0.037632
0.466742	-0.040175
0.432483	-0.042089
0.400213	-0.043302
0.368273	-0.043775

Table D.2: NHLP Main element coordinates.

... continued on next page

X	Y
0.333683	-0.043629
0.300083	-0.042773
0.267812	-0.041326
0.232882	-0.039080
0.202600	-0.036463
0.167009	-0.032877
0.136738	-0.029729
0.101466	-0.026254
0.084176	-0.024531
0.071125	-0.023236
0.067865	-0.022807
0.061245	-0.022280
0.054724	-0.021653
0.052894	-0.021413
0.048644	-0.021055
0.043553	-0.019327
0.039062	-0.015139
0.037490	-0.009480
0.039193	-0.001600
0.043656	0.006187
0.048248	0.010875
0.055930	0.016862
0.062122	0.020980
0.072404	0.026895
0.083746	0.032221
0.104780	0.040192
0.117321	0.044177
0.134553	0.047860

Table D.2: NHLP Main element coordinates.

... continued on next page

X	Y
0.153484	0.050622
0.168355	0.052406
0.172915	0.052905
0.182465	0.053971
0.203186	0.056082
0.234227	0.058840
0.266158	0.061187
0.301089	0.063263
0.335030	0.064849
0.366300	0.065896
0.401560	0.066622
0.434490	0.066839
0.468430	0.066555
0.499030	0.065842
0.533300	0.064538
0.567889	0.062624
0.599828	0.060252
0.635427	0.057277
0.668685	0.053304
0.701283	0.049230
0.734552	0.044657
0.765160	0.040154
0.799088	0.034791
0.833345	0.029117
0.867943	0.023183
0.899869	0.017549

Table D.2: NHLP main element coordinates.

X	Y
1.214680	-0.118471
1.193338	-0.111918
1.151011	-0.098866
1.123652	-0.090382
1.097227	-0.082259
1.058956	-0.070693
1.034640	-0.063559
0.996144	-0.052613
0.967852	-0.044769
0.953672	-0.040925
0.940453	-0.037366
0.926249	-0.033587
0.917217	-0.031188
0.909172	-0.028967
0.905110	-0.027491
0.901393	-0.025044
0.900337	-0.017233
0.907724	-0.009613
0.915381	-0.007061
0.920258	-0.006296
0.926455	-0.005819
0.930336	-0.005861
0.937716	-0.005997
0.941851	-0.006334
0.950354	-0.007272
0.960401	-0.008849
0.971630	-0.011069
0.977521	-0.012397
0.989219	-0.015288
0.995342	-0.016946
1.001426	-0.018707
1.020144	-0.024678
1.034262	-0.029655
1.063794	-0.040911
1.097863	-0.055278
1.125609	-0.068513
1.155571	-0.084427
1.183471	-0.100154
1.214680	-0.118471

Table D.3: NHLP flap coordinates.