AIAA 99–0167

Aerodynamic Computations Using a Higher-Order Algorithm

S. De Rango and D. W. Zingg
Institute for Aerospace Studies
University of Toronto
Toronto, Ontario, Canada

AIAA 37th Aerospace Sciences Meeting
Jan. 11-14, 1999/Reno, NV

Aerodynamic Computations Using a Higher-Order Algorithm

S. De Rango* and D. W. Zingg†
Institute for Aerospace Studies
University of Toronto
Toronto, Ontario, Canada

A globally third-order accurate algorithm is presented for computing steady turbulent flow over two-dimensional airfoils. The algorithm uses finite-differences applied through a generalized curvilinear coordinate transformation. Numerical dissipation is added using the matrix dissipation scheme. All components of the spatial discretization have been raised to a suitable level of accuracy. This includes the convective, viscous, and turbulence terms, the numerical boundary schemes, the numerical dissipation, and the integration technique used to calculate forces and moments. Results for several grid convergence studies show that this globally higher-order approach produces a dramatic reduction in the numerical error in drag. Very low errors are obtained on grids with only 13,000 nodes. The results provide a convincing demonstration of the benefits of higher-order methods for practical flows.

Introduction

DESPITE impressive progress in both algorithms and computers, the time required to compute the solution of the steady compressible Navier-Stokes equations for the flow about aerodynamic configurations remains excessive for routine use in aircraft design. Consequently, numerical solution techniques applicable to simpler physical models, such as panel methods or solvers incorporating the boundary-layer equations, are heavily used in the design process, despite their limitations. Although a substantial portion of the time required in a simulation based on the Navier-Stokes equations is involved in problem setup, including geometry definition and grid generation, and this needs to be addressed, the solution time is also excessive. One way to reduce the solution time is to improve the iterative method used to achieve a steady state.[1] Alternatively, the accuracy of the spatial discretization can be increased. 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 used. The promise of higher-order methods has been recognized for some time, beginning with the papers by Kreiss and Oliger[2] and Swartz and Wendroff.[3] As long as the increase in the computing expense per grid node associated with the use of the higher-order method is outweighed by the reduction in the number of grid nodes needed, the overall computing expense is reduced. Thus higher-order methods have received considerable use in the numerical solution of partial differential equations. Within the domain of aeronautics, they have primarily been used for time-dependent problems,[4] such as electromagnetics,[5] aeroacoustics,[6] and large-eddy and direct simulations of turbulent flow.[7,8]

Although the discretization of the inviscid fluxes is often raised to higher order, application of globally higher-order methods to steady aerodynamic flows has been more limited.[9,10] In fact, there exists no clear demonstration that higher-order methods are more efficient than conventional second-order methods in the computation of practical turbulent aerodynamic flows. The purpose of the present paper is to develop a third-order algorithm and to provide such a demonstration. A key point is that *every* component of the spatial discretization is raised to a suitable level of accuracy. This includes

1. the convective, viscous, and turbulence terms,
2. the terms in the turbulence model,
3. the numerical boundary schemes,
4. the numerical dissipation, and
5. the integration technique used to calculate forces and moments.

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.[11] Transonic flows with shocks introduce an additional degree of difficulty in that first-order numerical dissipation is typically added near

*Graduate Student, stan@oddjob.utias.utoronto.ca.
†Professor,Member AIAA.

shocks, which can potentially undermine the benefits of a higher-order method.[12]

Governing Equations and Numerical Method

The high-order spatial scheme has been implemented in the thin-layer Navier-Stokes solver ARC2D.[13] This solver uses a generalized curvilinear coordinate transformation and is thus applicable to structured grids. Convergence to steady state is achieved using the diagonal form of the Beam and Warming approximate-factorization algorithm with local time stepping. The effects of turbulence are modeled by the algebraic Baldwin-Lomax[14] model and the one-equation Spalart-Allmaras[15] model. Subiterations are used within the latter model. A circulation correction is used at the outer boundary.

In two-dimensional generalized coordinates, the thin-layer Navier-Stokes equations are given by

$$\partial_t \hat{Q} + \partial_\xi \hat{E} + \partial_\eta \hat{F} = Re^{-1} \partial_\eta \hat{S} \tag{1}$$

where $\hat{Q}$ contains the conservative flow variables and $\hat{E}$ and $\hat{F}$ are the flux vectors in the ξ and η directions respectively. $\hat{S}$ contains the viscous terms.

Applying implicit Euler time-marching to the discretized form of Eq. 1 and linearizing, one obtains:

$$[I + h\delta_\xi \hat{A}^n + h\delta_\eta \hat{B}^n - h\mathcal{R}e^{-1}\delta_\eta \hat{M}^n]\Delta\hat{Q}^n = \hat{R} \tag{2}$$

where

$$\hat{R}^n = -h[\delta_\xi \hat{E}(\hat{Q}^n) + \delta_\eta \hat{F}(\hat{Q}^n) - \mathcal{R}e^{-1}\delta_\eta \hat{S}(\hat{Q}^n)] \tag{3}$$

The time-step is denoted by h, $\hat{Q}^n = \hat{Q}(nh)$, $\Delta\hat{Q}^n = \hat{Q}^{n+1} - \hat{Q}^n$, and δ denotes a generic spatial operator. The matrices $\hat{A}$, $\hat{B}$, and $\hat{M}$ are the flux Jacobians, defined by

$$\hat{A} = \frac{\partial \hat{E}}{\partial \hat{Q}}, \quad \hat{B} = \frac{\partial \hat{F}}{\partial \hat{Q}}, \quad \hat{M} = \frac{\partial \hat{S}}{\partial \hat{Q}} \tag{4}$$

The diagonalized, approximately-factored form of Eq. 2 is:

$$T_\xi[I + \delta_\xi \Lambda_\xi]\hat{N}[I + \delta_\eta \Lambda_\eta]T_\eta^{-1}\Delta\hat{Q}^n = \hat{R}^n \tag{5}$$

where

$$\hat{N} = T_\xi^{-1} T_\eta \tag{6}$$

The matrices Λ_ξ and Λ_η are diagonal matrices whose elements are the eigenvalues of the flux Jacobians, including a contribution from the viscous flux Jacobian.[13] The matrix T_ξ has the eigenvectors of $\hat{A}$ as columns and T_η has the eigenvectors of $\hat{B}$ as columns.

Higher-Order Spatial Discretization

Convective fluxes

Centered differences with added numerical dissipation are used for the convective fluxes. The numerical dissipation is added using the matrix dissipation scheme of Swanson and Turkel.[16] As demonstrated by Frew et al.,[17] the scalar dissipation scheme of Jameson et al.[18] introduces considerable error. Hence there is no point in using a higher-order discretization as long as the scalar dissipation scheme is used. In contrast, the numerical error introduced using matrix dissipation is generally less than the truncation error from a second-order centered difference operator.[19] For subsonic flows, we do not require any first-order dissipation, and hence the numerical dissipation is third order. For transonic flows, we use the simple pressure switch described by Pulliam[13] to introduce a limited amount of first-order dissipation near shocks.

The numerical dissipation is added in the following manner:

$$\frac{\partial \hat{E}}{\partial \xi} \approx \delta_\xi \hat{E} - \nabla_\xi D_1 D_2 \tag{7}$$

with

$$\begin{aligned}
D_1 &= |\widehat{A}|_{j+\frac{1}{2},k} J^{-1}_{j+\frac{1}{2},k} \\
D_2 &= \epsilon^{(2)}_{j+\frac{1}{2},k}\Delta_\xi Q_{j,k} - \epsilon^{(4)}_{j+\frac{1}{2},k}\nabla_\xi\Delta_\xi\nabla_\xi Q_{j,k} \\
\epsilon^{(2)}_{j,k} &= \kappa_2 \Delta t \max(\Upsilon_{j+1,k}, \Upsilon_{j,k}, \Upsilon_{j-1,k}) \\
\epsilon^{(4)}_{j,k} &= \max(0, \kappa_4\Delta t - \epsilon^{(2)}_{j,k}) \\
\Upsilon_{j,k} &= \frac{|p_{j+1,k} - 2p_{j,k} + p_{j-1,k}|}{|p_{j+1,k} + 2p_{j,k} + p_{j-1,k}|}
\end{aligned}$$

where J^{-1} is the metric Jacobian of the coordinate transformation, δ_ξ is a centered difference operator, Δ_ξ and ∇_ξ are first-order forward and backward difference operators, U is a contravariant velocity component, and κ_2 and κ_4 are constants equal to 0.5 and 0.01 respectively. An analogous term appears in the η direction. The matrix $|\hat{A}|$ is given by

$$|\hat{A}| = T_\xi |\Lambda|_\xi T_\xi^{-1} \tag{8}$$

where the columns of T_ξ are the right eigenvectors of $\widehat{A}$ and

$$|\Lambda|_\xi = \begin{bmatrix} |U| & 0 & 0 & 0 \\ 0 & |U| & 0 & 0 \\ 0 & 0 & |U + a\sqrt{\xi_x^2 + \xi_y^2}| & 0 \\ 0 & 0 & 0 & |U - a\sqrt{\xi_x^2 + \xi_y^2}| \end{bmatrix} \tag{9}$$

In evaluating $|\hat{A}|_{j+\frac{1}{2},k}$ we have used a simple average, although the Roe average can also be used.

Near stagnation points and sonic points, some of the eigenvalues of $\hat{A}$ approach zero. In order to avoid the problems this may introduce, the elements of $|\Lambda|_\xi$ can be modified as follows:

$$\begin{aligned}\tilde{\lambda}_1, \tilde{\lambda}_2 &= \max(\lambda_{1,2}, V_l\sigma) \\ \tilde{\lambda}_3 &= \max(\lambda_3, V_n\sigma) \\ \tilde{\lambda}_4 &= \max(\lambda_4, V_n\sigma) \qquad (10)\end{aligned}$$

We use $V_l = V_n = 0$ for subsonic flows and $V_l = 0.025$, $V_n = 0.25$ for transonic flows.

Gustafsson[20] has shown that the 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. 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.

The following operators are used to approximate first derivatives:

Interior (4th-order)

$$\delta_x q_j = \frac{1}{12\Delta x}(-q_{j+2} + 8q_{j+1} - 8q_{j-1} + q_{j-2}) \quad (11)$$

First Interior Node (3rd-order)

$$\delta_x q_j = \frac{1}{6\Delta x}(-2q_{j-1} - 3q_j + 6q_{j+1} - q_{j+2}) \quad (12)$$

Boundary (3rd-order)

$$\delta_x q_j = \frac{1}{24\Delta x}(-11q_j + 18q_{j+1} - 9q_{j+2} + 2q_{j+3}) \quad (13)$$

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

The following operator is used for the numerical dissipation:

$$\frac{1}{\Delta x}(q_{j+2} - 4q_{j+1} + 6q_j - 4q_{j-1} + q_{j-2}) \quad (14)$$

Since

$$(q_{j+2} - 4q_{j+1} + 6q_j - 4q_{j-1} + q_{j-2}) = \Delta x^4 \frac{\partial^4 q}{\partial x^4} \quad (15)$$

the dissipation is a third-order term. This is the same operator as is commonly used in combination with second-order centered differences, i.e., we have not improved the accuracy of the numerical dissipation operator.

The finite-difference stencil in Eq. 14 cannot be used at the first and last interior nodes. The following operator should be used to maintain third-order scaling for the dissipation at near-boundary nodes:

$$\frac{1}{\Delta x}(q_{j+1} - 4q_j + 6q_{j-1} - 4q_{j-2} + q_{j-3}) \quad (16)$$

Eq. 16 can only be applied to the explicit or right-hand side of the equations if one wishes to maintain the pentadiagonal structure of the implicit side.

Viscous Fluxes

The viscous terms are in the following general form:

$$\partial_x(\alpha_j \partial_x \beta_j) \quad (17)$$

The following fourth-order expression is used to calculate the $\partial_x \beta_j$ term at half nodes:

$$(\delta_x\beta)_{j+\frac{1}{2}} = \frac{1}{24\Delta x}(\beta_{j-1} - 27\beta_j + 27\beta_{j+1} - \beta_{j+2}) \quad (18)$$

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

$$(\delta_x\beta)_{j+\frac{1}{2}} = \frac{1}{24\Delta x}(-23\beta_j + 21\beta_{j+1} + 3\beta_{j+2} - \beta_{j+3}) \quad (19)$$

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

$$\alpha_{j+\frac{1}{2}} = \frac{1}{16}(-\alpha_{j-1} + 9\alpha_j + 9\alpha_{j+1} - \alpha_{j+2}) \quad (20)$$

Near boundaries, a third-order formula is used:

$$\alpha_{j+\frac{1}{2}} = \frac{1}{8}(3\alpha_j + 6\alpha_{j+1} - \alpha_{j+2}) \quad (21)$$

The complete operator is then found from:

$$\begin{aligned}\delta_x(\alpha_j\delta_x\beta_j) = \; & \frac{1}{24\Delta x}(\alpha_{j-3/2}(\delta_x\beta)_{j-3/2} \qquad (22) \\ & -27\alpha_{j-1/2}(\delta_x\beta)_{j-1/2} \\ & +27\alpha_{j+1/2}(\delta_x\beta)_{j+1/2} \\ & -\alpha_{j+3/2}(\delta_x\beta)_{j+3/2})\end{aligned}$$

in the interior, and

$$\begin{aligned}\delta_x(\alpha_j\delta_x\beta_j) = \; & \frac{1}{24\Delta x}(-23\alpha_{j-1/2}(\delta_x\beta)_{j-1/2} \qquad (23) \\ & +21\alpha_{j+1/2}(\delta_x\beta)_{j+1/2} \\ & +3\alpha_{j+3/2}(\delta_x\beta)_{j+3/2} \\ & -\alpha_{j+5/2}(\delta_x\beta)_{j+5/2})\end{aligned}$$

near boundaries. This approach leads to a seven-point stencil. Although fourth-order accuracy can also be achieved using a five-point stencil, the present approach is simple to program and only marginally less accurate. On the left-hand-side of the approximate factorization algorithm, a second-order operator is used, with no apparent degradation in convergence rate.

Turbulence Models

The implementation of the Baldwin-Lomax turbulence model requires the calculation of the vorticity at half nodes. This is accomplished using the operators given in Eqs. 18 and 19.

For the Spalart-Allmaras model, the diffusive terms are handled in the same manner as the viscous terms described in the preceding subsection. A third-order upwind-biased scheme is used for the convective terms. Note that the original lower-order solver uses a first-order upwind approximation for the convective terms in the turbulence model in order to maintain positivity of the eddy viscosity.

Boundary Conditions

Far Field Boundary

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

$$\begin{aligned} \hat{Q}_{bc} &= \frac{1}{2}(\hat{Q}_\infty + \hat{Q}_{ext}) \\ &\quad -\frac{1}{2}\mathrm{sign}(\hat{A})(\hat{Q}_\infty - \hat{Q}_{ext}) \end{aligned} \tag{24}$$

$$\mathrm{sign}(\hat{A}) = T_\kappa \ \mathrm{sign}(\Lambda_\kappa) \ T_\kappa^{-1} \tag{25}$$

where κ is chosen in the direction normal to the boundary, bc indicates the boundary value, ∞ indicates values obtained from free stream conditions, and ext indicates values extrapolated from the interior nodes of the mesh. The eigenvalues and eigenvectors are calculated from the mean state, $\hat{Q}_{avg} = \frac{1}{2}(\hat{Q}_\infty + \hat{Q}_{ext})$. The following second-order extrapolation operator is used at the farfield boundary:

$$q_j = 3q_{j+1} - 3q_{j+2} + q_{j+3} \tag{26}$$

Note that the use of a finite domain does introduce error, even when a circulation correction is used.[21] The error varies with the inverse of the distance to the outer boundary.[22] 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 $\partial p/\partial n = 0$, which gives

$$p_1 = \frac{1}{11}(18p_2 - 9p_3 + 2p_4) \tag{27}$$

For an adiabatic wall, $\partial T/\partial n = 0$, and, when coupled with the assumption of zero pressure gradient and the perfect gas law, this implies $\partial \rho/\partial n = 0$. Hence the density at the airfoil surface is determined from an expression analogous to Eq. 27. Note that the assumption that $\partial p/\partial n = 0$ is not strictly correct. However, for aerodynamic flows at high Reynolds numbers, the error introduced is very small. We have experimented with extrapolation of pressure with no significant change in the solution. Furthermore, the error introduced is non-ordered,[22] and hence does not affect the conclusions from grid convergence studies. The velocity is zero on the body surface.

Wake-Cut

The C-grid wake-cut is treated explicitly. The interpolation at the wake-cut (wc) is computed to fourth-order using the data above and below the wake-cut as follows:

$$q_{k_{wc}} = \frac{1}{6}(-q_{k_{wc}+2} + 4q_{k_{wc}+1} + 4q_{k_{wc}-1} - q_{k_{wc}-2}) \tag{28}$$

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. 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 force, C_N, and axial force, C_A, coefficients with respect to the chord line. (For ease of presentation, we consider the pressure contribution only.)

$$C_N = \frac{1}{c}\oint -C_p(\hat{n}\cdot\hat{y})ds \tag{29}$$

$$C_A = \frac{1}{c}\oint -C_p(\hat{n}\cdot\hat{x})ds \tag{30}$$

where c is the chord length, s is the arclength along the airfoil surface, 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

$$\hat{n} = \frac{-\frac{dy}{ds}\hat{x} + \frac{dx}{ds}\hat{y}}{\sqrt{(\frac{dx}{ds})^2 + (\frac{dy}{ds})^2}} \tag{31}$$

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 Eqs. 29 and 30. The quadrature routine

Grid	Dimensions	Points on Airfoil	Off-Wall Spacing ($\times 10^{-6}$)	Leading Edge Clustering ($\times 10^{-3}$)	Trailing Edge Clustering ($\times 10^{-3}$)
A	1057×193	801	0.23	0.1	0.5
B	529×97	401	0.53	0.2	1.0
C	265×49	201	1.2	0.4	2.0

Table 1 Summary of grids.

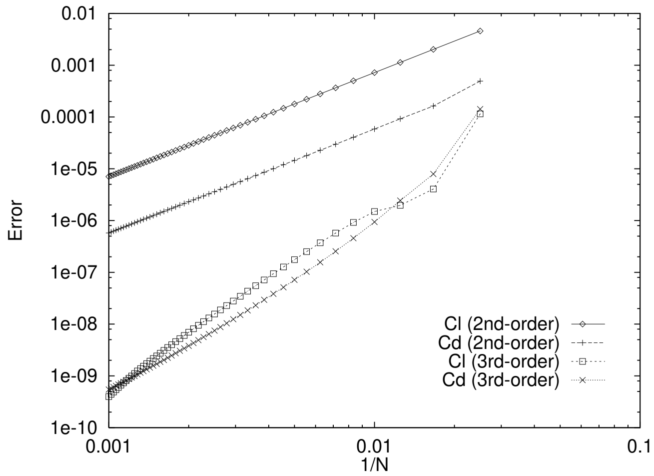


Fig. 1 Error in force integration algorithm.

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.[23]

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. For the case studied, the analytical lift and drag coefficients are 1.2181 and zero respectively. Figure 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 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.

Results and Discussion

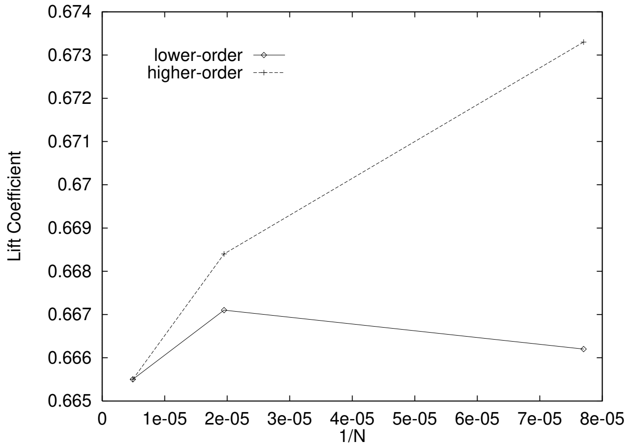
In order to evaluate the higher-order algorithm, the flow over the NACA 0012 airfoil is computed under the following conditions:

1. M_∞=0.16, α=6°, Re=2.88$\times 10^6$, transition at 0.05 and 0.8 chords on the upper and lower surfaces, respectively.
2. $M_\infty = 0.16$, $\alpha = 12°$, $Re = 2.88 \times 10^6$, transition at 0.01 and 0.95 chords on the upper and lower surfaces, respectively.
3. M_∞=0.7, α=3°, Re=9 $\times 10^6$, transition at 0.05 chords on both surfaces.

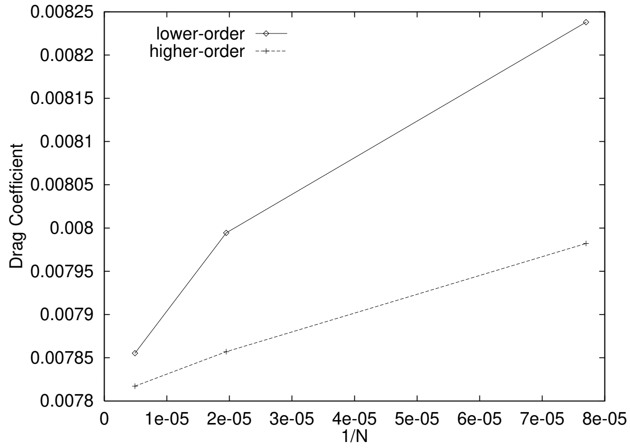
Table 1 provides a description of the grids used. All of the grids have a "C" topology, a distance to the outer boundary of 12 chords and were generated using an elliptic grid generator. Grid A has farfield cells which are one chord in length. Grid B was generated by removing every second node in both coordinate directions from grid A, and grid C was similarly generated from grid B. In the absence of grids generated according to prescribed analytical functions, this produces a sequence of grids which are suitable for grid convergence studies.[21] Grids A, B, and C have roughly 204,000, 51,000, and 13,000 nodes, respectively. 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.

Figs. 2 to 4 show the grid convergence of the lift coefficient, C_l, drag coefficient, C_d, the pressure drag component, C_{d_p}, and the friction drag component, C_{d_f}, for the three test cases computed using the Baldwin-Lomax turbulence model. The results are plotted vs $1/N$ where N is the total number of grid nodes. Results are shown for the present higher-order algorithm ("higher-order") and the original second-order algorithm with matrix dissipation ("lower-order").[24] Figs. 5 to 7 show the corresponding results computed using the Spalart-Allmaras turbulence model.

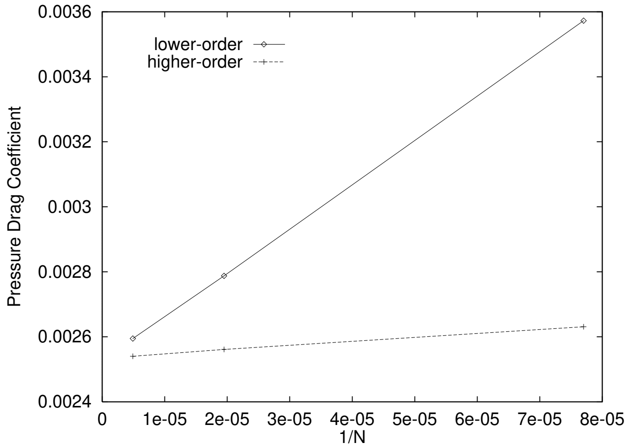
The figures suggest that the solutions obtained on grid A are almost grid independent, i.e., discretization errors are very small on this grid. Hence a superior discretization is one which produces a solution sufficiently close to the grid-independent solution on as coarse a



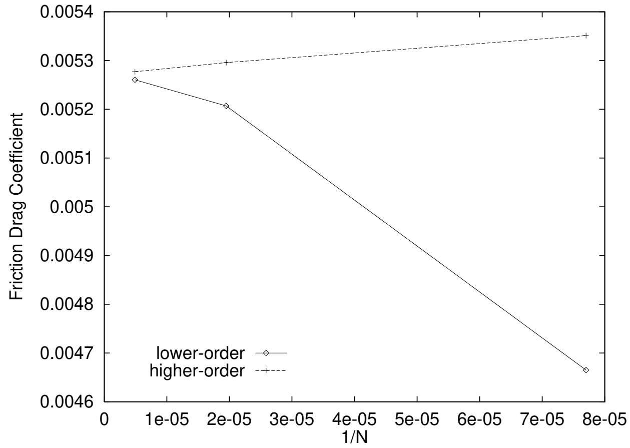
a) C_l



b) C_d



c) C_{d_p}



d) C_{d_f}

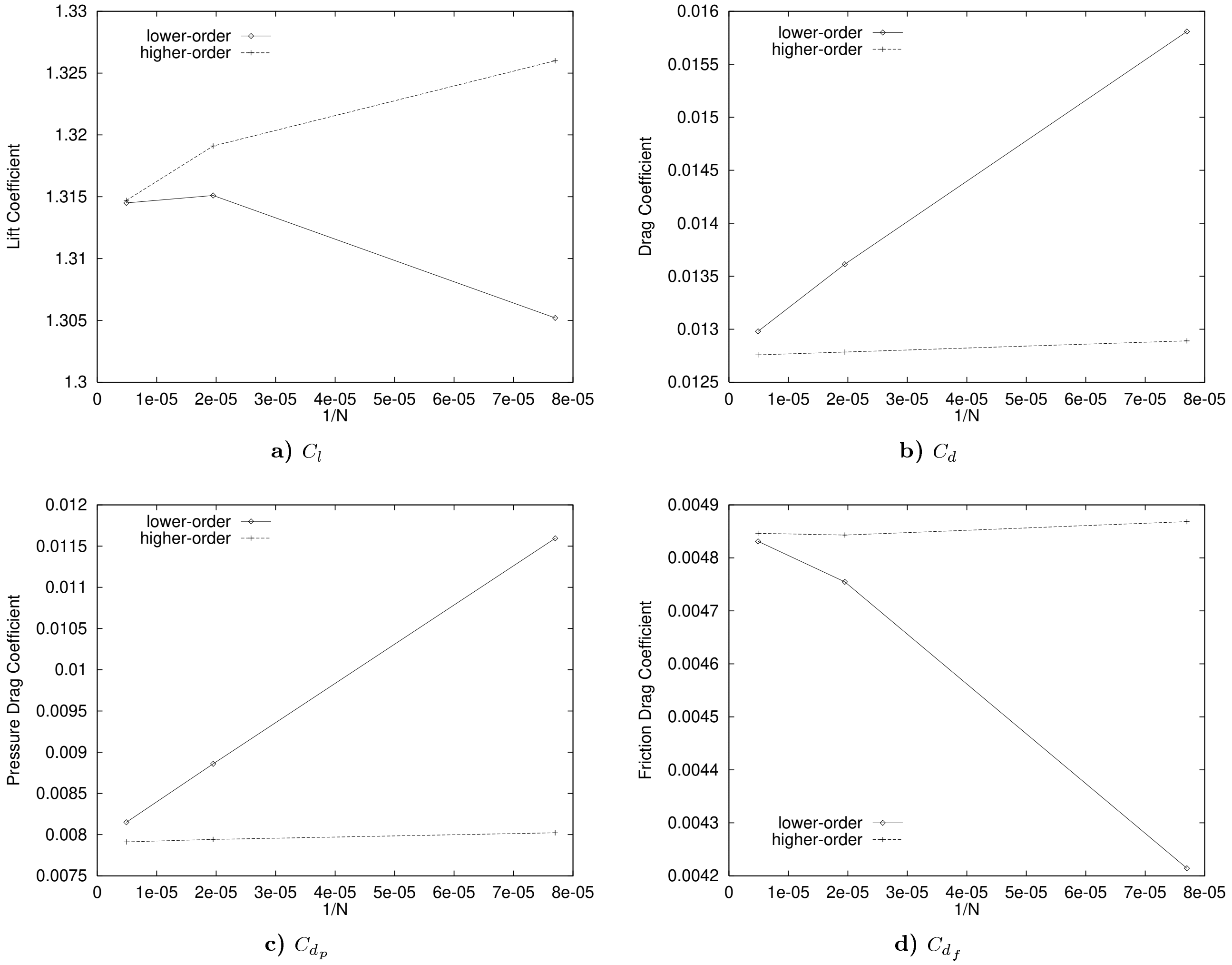
Fig. 2 Grid convergence study for case 1 (Baldwin-Lomax model).

grid as possible. In particular, we would like to see discretization errors below one percent on grid C.

Examination of the behavior of the lift coefficient with the Baldwin-Lomax model shows that an error less than one percent is generally achieved, even on grid C. Furthermore, the higher-order algorithm generally produces no improvement in the accuracy of the lift coefficient. In contrast, for the cases computed with the Spalart-Allmaras model, the error on grid C computed using the lower-order algorithm approaches two percent, while the error from the higher-order algorithm is much lower. The difference in the behavior with the two turbulence models is probably associated with the method used to handle laminar-turbulent transition. We use a simple ramp over two grid points for the Baldwin-Lomax turbulence model. This is inherently grid dependent. With the Spalart-Allmaras model, we use the trip term described by Spalart and Allmaras,[15] which is not grid dependent. The errors in lift obtained using the higher-order algorithm are reduced with the Spalart-Allmaras model, and hence the grid dependence of the transition treatment is the most likely cause of the lift errors with the Baldwin-Lomax model.

The errors in the drag coefficient are much larger, with the lower-order algorithm consistently producing errors greater than one percent even on grid B, and up to twenty percent on grid C. In general, the higher-order algorithm produces much lower errors in drag. In the cases for which the lower-order algorithm produces a smaller error in drag, inspection of the individual drag components shows that this is a consequence of error cancellation. In terms of pressure and friction drag, the higher-order algorithm consistently produces a large reduction in error relative to the lower-order algorithm. It is noteworthy that the higher-order algorithm often produces a substantially smaller error on grid C than that produced by the lower-order algorithm on grid B, which has four times as many nodes.

The errors in pressure drag obtained using the higher-order algorithm are higher for the transonic case (case 3) than for the other two cases. This is primarily associated with the first-order numerical dissipation added near the shock wave. With the value of the second-difference dissipation coefficient used, $\kappa_2 = 0.5$, the surface pressure distribution is slightly



a) C_l **b)** C_d **c)** C_{d_p} **d)** C_{d_f}

Fig. 3 Grid convergence study for case 2 (Baldwin-Lomax model).

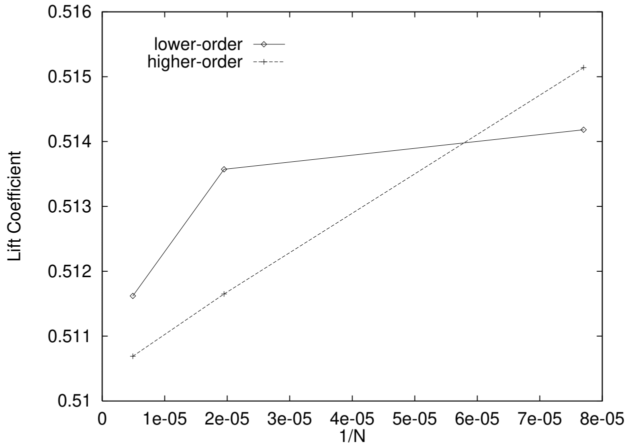
oscillatory, but the pressure drag error is reasonably low. Increasing this parameter to $\kappa_2 = 2$ eliminates the oscillations, but increases the pressure drag error.

Fig. 8 compares boundary-layer velocity profiles at the 85% chord station on the upper surface for case 3 computed using the Baldwin-Lomax turbulence model. The improvement in accuracy associated with the higher-order algorithm is clearly evident. The higher-order solution computed on grid C agrees very well with the lower-order solution computed on grid A, which has sixteen times as many nodes.

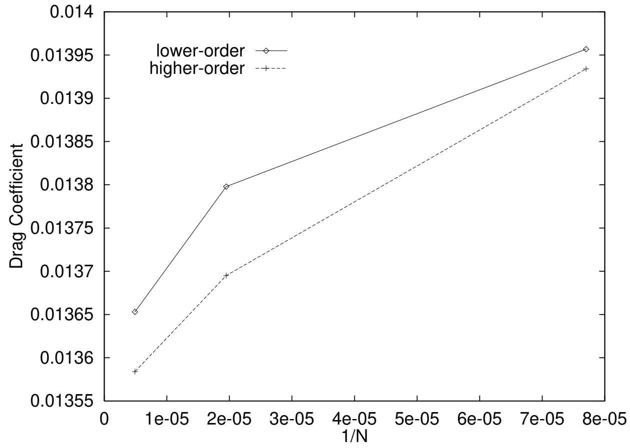
In the context under consideration here, namely the solution of the thin-layer Navier-Stokes equations using a generalized curvilinear coordinate transformation, the extension of all terms to higher order can be accomplished very efficiently. The cost per grid node per iteration is increased by about 5%. Since the convergence rates are not significantly affected, the overall cost increase on a given grid is about 5% relative to the lower-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 lower-order algorithm, which requires a much finer grid.

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.[25] Thus it is instructive to examine the relative importance of raising various terms to higher order. This issue is addressed in Figs. 9 and 10, which show results for cases 1 and 3 computed using the Baldwin-Lomax turbulence model. Results are shown for the following five algorithms:

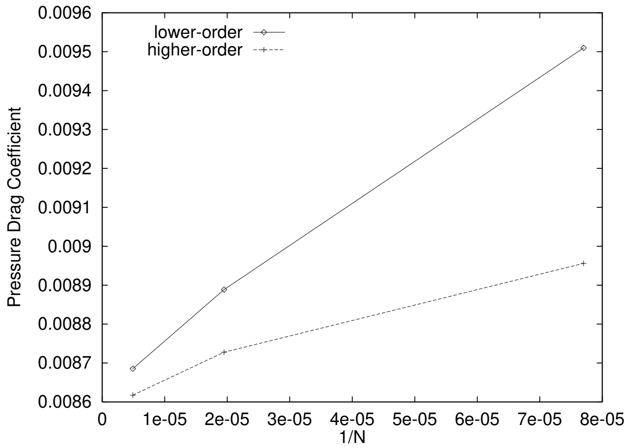
1. the lower-order algorithm (indicated on the figures by "lower-order"),
2. the higher-order algorithm ("higher-order"),
3. the lower-order algorithm with higher-order integration only ("higher-order integration"),
4. the higher-order algorithm with lower-order integration ("lower-order integration"),
5. the higher-order algorithm with lower-order viscous terms ("lower-order viscous terms").



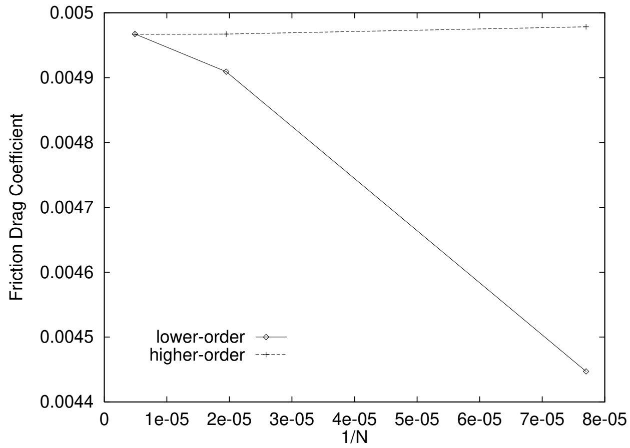
a) C_l



b) C_d



c) C_{d_p}



d) C_{d_f}

Fig. 4 Grid convergence study for case 3 (Baldwin-Lomax model).

Concentrating on the individual drag components, we can draw the following conclusions from these figures:

- Use of the higher-order integration routine with the lower-order algorithm produces a very small reduction in error.
- Use of the lower-order integration routine with the higher-order algorithm produces a large increase in friction drag error. Thus the integration routine is an important component of a higher-order algorithm.
- The use of lower-order viscous terms with the higher-order algorithm leads to a moderate increase in error in the friction drag for case 3. It appears that the treatment of the viscous terms is not quite as critical as the inviscid terms, but is nevertheless important.

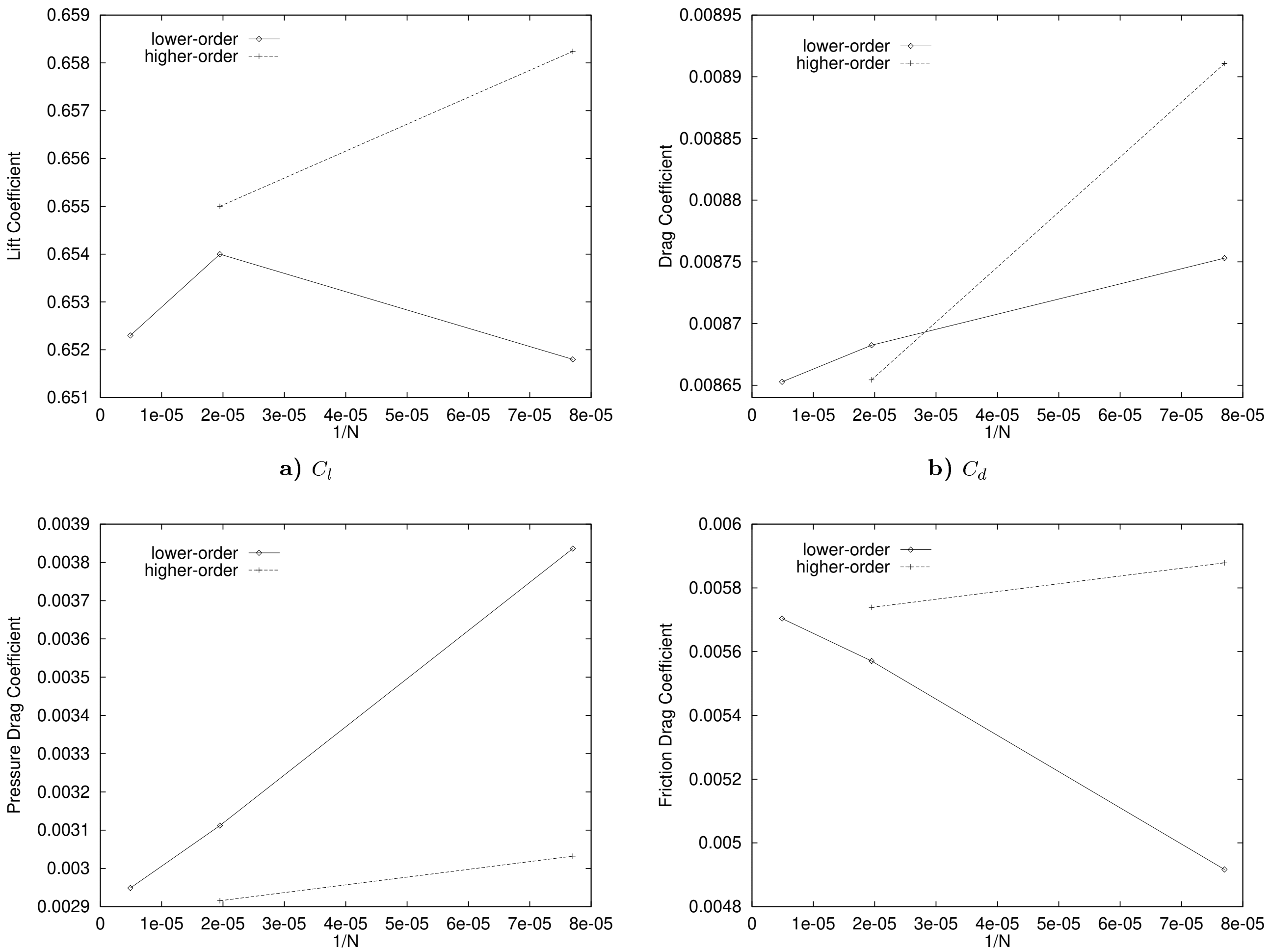
We have also found that the use of a first-order approximation for the convective terms in the Spalart-Allmaras model does not appear to be a significant source of error.

Conclusions

We have described a higher-order algorithm in which all components contributing to the accuracy of the spatial discretization have been raised to third order. Grid convergence studies demonstrate that the new algorithm produces a substantial reduction in the numerical error in drag in comparison with a second-order algorithm. 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.

Acknowledgments

This research was supported by the Natural Sciences and Engineering Research Council of Canada and de-Havilland Inc.



a) C_l **b)** C_d

c) C_{d_p} **d)** C_{d_f}

Fig. 5 Grid convergence study for case 1 (Spalart-Allmaras model).

References

[1]Pueyo, A., and Zingg, D.W., "An Efficient Newton-GMRES Solver for Aerodynamic Computations." AIAA Paper 97-1955, June 1997.

[2]Kreiss, H.-O., and Oliger, J., "Comparison of Accurate Methods for the Integration of Hyperbolic Equations," *Tellus*, vol. 24, pp. 199–215, 1972.

[3]Swartz, B., and Wendroff, B., "The Relative Efficiency of Finite Difference and Finite Element Methods," *Siam J. Num. Anal.*, vol. 11, no. 5, pp. 979–993, 1974.

[4]Zingg, D.W., "A Review of High-Order and Optimized Finite-Difference Methods for Simulating Linear Wave Phenomena." AIAA Paper 97-2088, June 1997.

[5]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.

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

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

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

[9]Rai, M.M., and Chakravarthy, S., "Conservative High-Order-Accurate Finite-Difference Methods For Curvilinear Grids." AIAA Paper 93-3380-CP, July 1993.

[10]Rigby, D.L., "Compact Spatial Differencing and Subiteration Time Marching in the PARC Code." AIAA Paper 96-0385, January 1996.

[11]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.

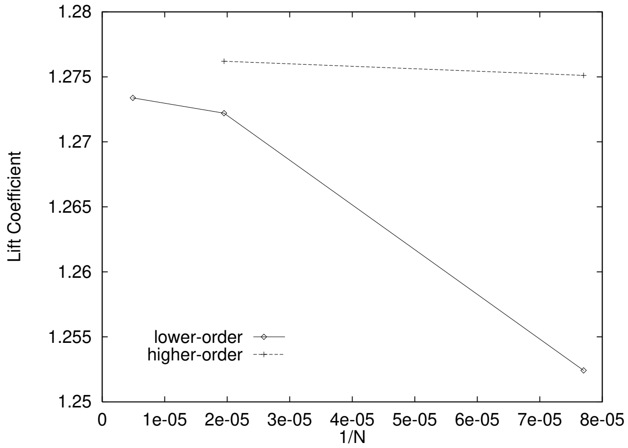
[12]Carpenter, M.H., and Casper, J.H., "The Accuracy of Shock Capturing In Two Spatial Dimensions." AIAA Paper 97-2107, June 1997.

[13]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.

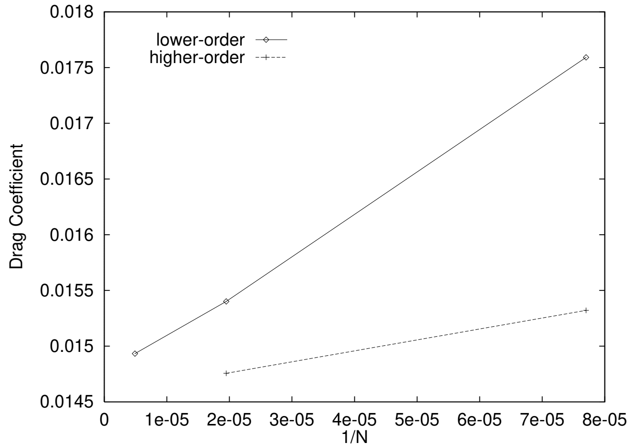
[14]Baldwin, B., and Lomax, H., "Thin-Layer Approximation and Algebraic Model for Separated Turbulent Flows." AIAA Paper 78-257, January 1978.

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

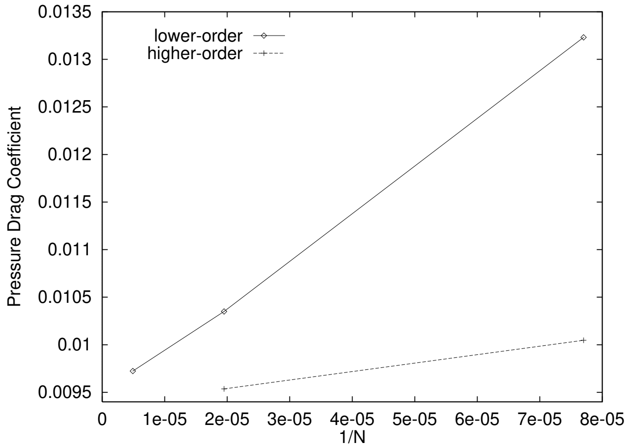
[16]Swanson, R.C., and Turkel, E., "On Central-Difference and Upwind Schemes," *J. Comp. Phys.*, vol. 101, pp. 292–306, 1992.



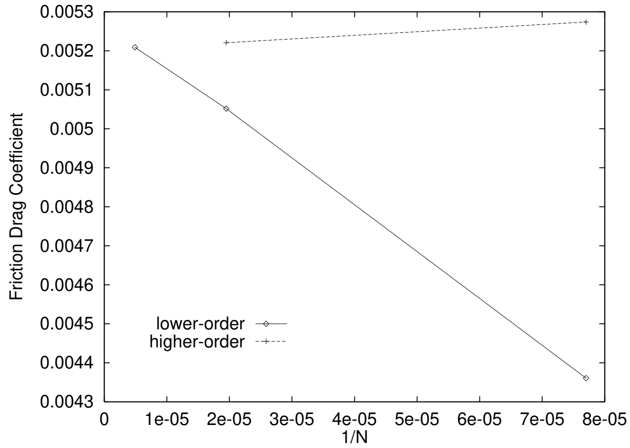
a) C_l



b) C_d



c) C_{d_p}



d) C_{d_f}

Fig. 6 Grid convergence study for case 2 (Spalart-Allmaras model).

[17]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.

[18]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.

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

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

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

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

[23]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.

[24]Frew, K., and Zingg, D.W., "On Artificial Dissipation Models for Viscous Airfoil Computations." AIAA Paper 96-1970, June 1996.

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

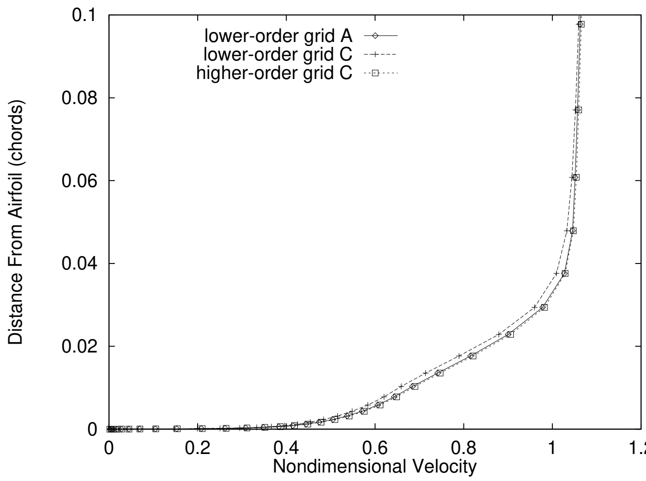
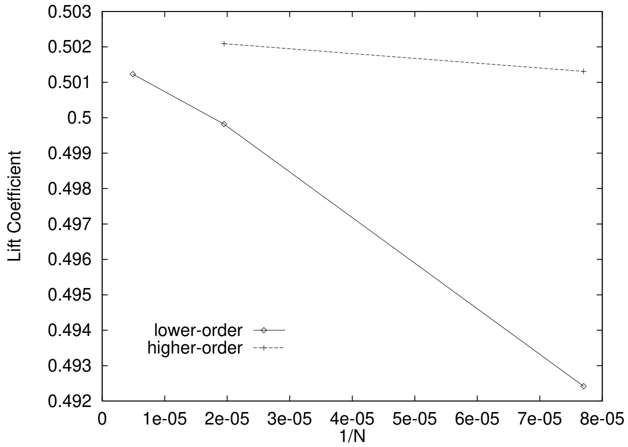
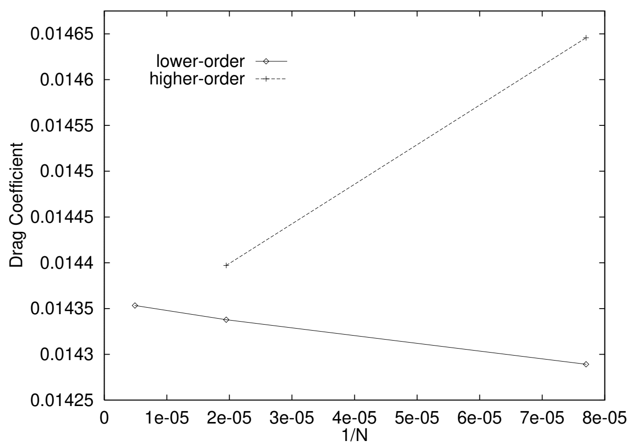


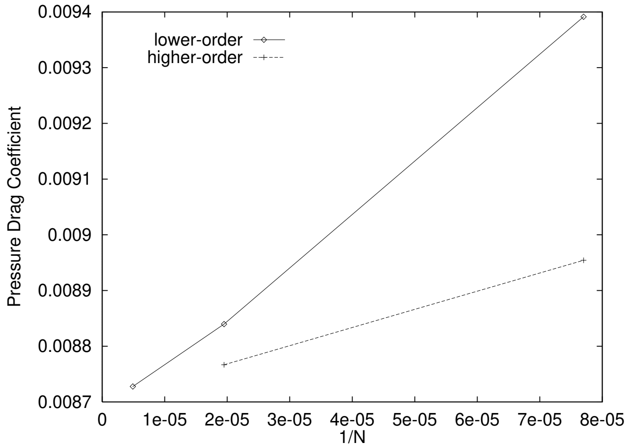
Fig. 8 Boundary layer velocity profiles.



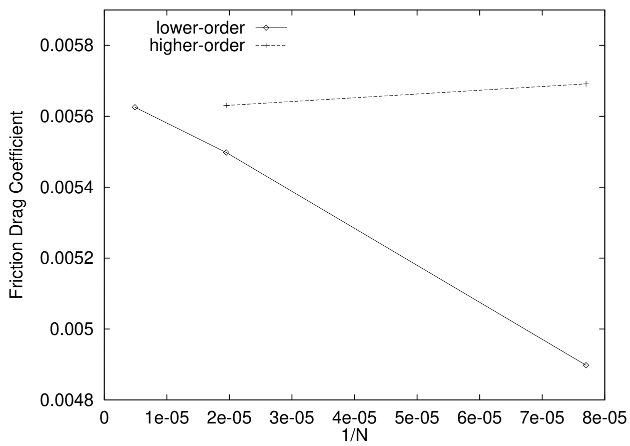
a) C_l



b) C_d

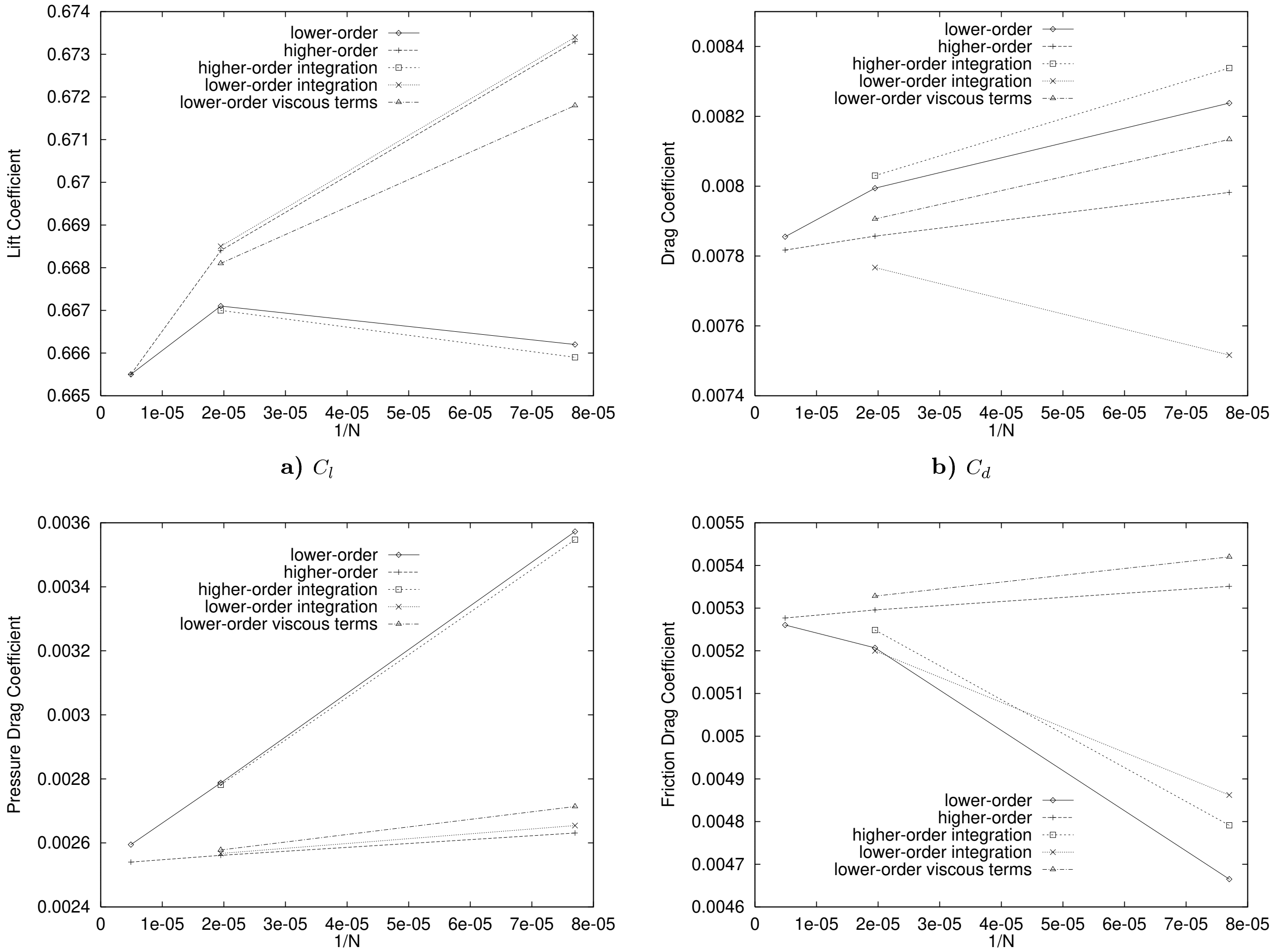


c) C_{d_p}



d) C_{d_f}

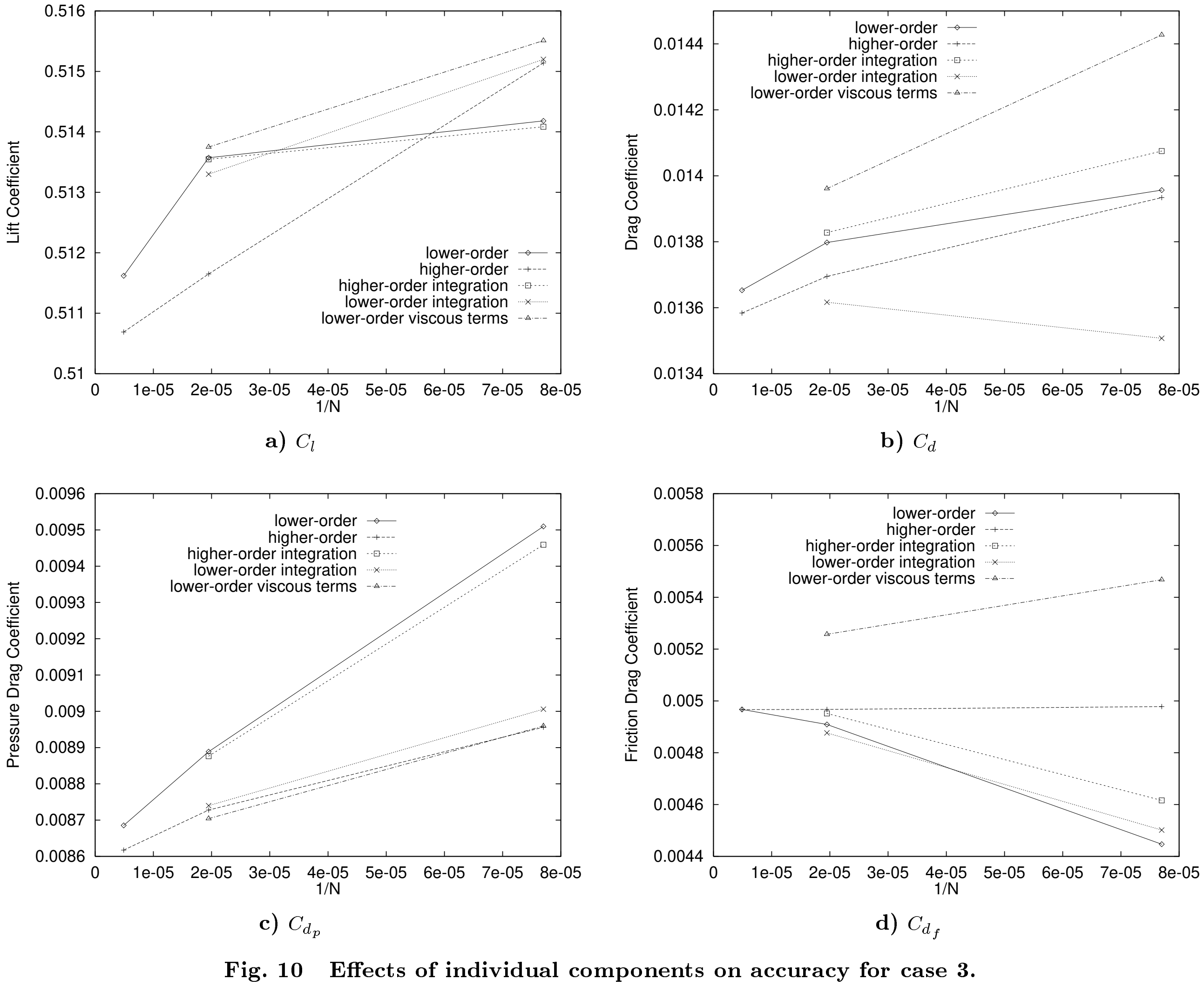
Fig. 7 Grid convergence study for case 3 (Spalart-Allmaras model).



a) C_l **b)** C_d

c) C_{d_p} **d)** C_{d_f}

Fig. 9 Effects of individual components on accuracy for case 1.



a) C_l **b)** C_d

c) C_{d_p} **d)** C_{d_f}

Fig. 10 Effects of individual components on accuracy for case 3.