Math. Model. Nat. Phenom.
Vol. 6, No. 3, 2011, pp. 2-27
DOI: 10.1051/mmnp/20116301

Observations Regarding Algorithms Required for Robust CFD Codes

F. T. Johnson [1*], D. S. Kamenetskiy[2], R. G. Melvin[2], V. Venkatakrishnan[2], L. B. Wigton[2], D. P. Young[2], S. R. Allmaras[2], J. E. Bussoletti[2] and C. L. Hilmes[2]

[1] Contractor to The Boeing Company through YourEncore
[2] The Boeing Company, PO Box 3707, Seattle, Washington 98124-2207, USA

Abstract. Over the last three decades Computational Fluid Dynamics (CFD) has gradually joined the wind tunnel and flight test as a primary flow analysis tool for aerodynamic designers. CFD has had its most favorable impact on the aerodynamic design of the high-speed cruise configuration of a transport. This success has raised expectations among aerodynamicists that the applicability of CFD can be extended to the full flight envelope. However, the complex nature of the flows and geometries involved places substantially increased demands on the solution methodology and resources required. Currently most simulations involve Reynolds-Averaged Navier-Stokes (RANS) codes although Large Eddy Simulation (LES) and Detached Eddy Suimulation (DES) codes are occasionally used for component analysis or theoretical studies. Despite simplified underlying assumptions, current RANS turbulence models have been spectacularly successful for analyzing attached, transonic flows. Whether or not these same models are applicable to complex flows with smooth surface separation is an open question. A prerequisite for answering this question is absolute confidence that the CFD codes employed reliably solve the continuous equations involved. Too often, failure to agree with experiment is mistakenly ascribed to the turbulence model rather than inadequate numerics. Grid convergence in three dimensions is rarely achieved. Even residual convergence on a given grid is often inadequate. This paper discusses issues involved in residual and especially grid convergence.

Key words: CFD, grid refinement, residual convergence, adjoint, turbulence model
AMS subject classification: 65N12, 65N22, 65N30, 65N50

*Corresponding author. E-mail: forrester.johnson@boeing.com

Article published by EDP Sciences and available at http://www.mmnp-journal.org or http://dx.doi.org/10.1051/mmnp/20116301

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

1. Introduction

Over the last three decades Computational Fluid Dynamics (CFD) has gradually joined the wind tunnel and flight test as a primary flow analysis tool of aerodynamic designers, e.g., [6]. CFD is now acknowledged to provide substantial value and has even created a paradigm shift in vehicle design, analysis, and support processes. Experience to date has shown that CFD has had its most favorable impact on the aerodynamic design of the high-speed cruise configuration of a transport. This success has raised expectations among aerodynamicists that CFD can become a routine tool for loads analysis, stability and control analysis, and high-lift design processes. In fact, there is considerable thought that it is now feasible to populate the large databases involved with results from Navier-Stokes codes. However, the flight conditions of interest are frequently characterized by complex flows, particularly large regions of separated flow. For example, such flows are encountered on transports at low speed with deployed high-lift devices, at structural design load conditions, or when transports are subjected to in-flight upsets that expose them to speed and/or angle of attack conditions outside the envelope of normal flight. Solutions involving complex flows often involve complex geometries as well, e.g., flaps, vortex generators, chines, spoilers, landing gear, etc. Often 30 to 300 million grid points and enormous numbers of time steps or iterations are required to obtain reasonable accuracy. Moreover, flows at the corners of the flight envelope are sometimes highly unsteady, requiring the use of time accurate CFD codes (e.g. unsteady RANS codes) at perhaps an order of magnitude or two greater expense.

Fortunately, many flows of interest to high lift, loads, and stability and control engineers are approximately steady. Whether or not a particular case requires unsteady treatment is an unsolved issue. Conventional wisdom dictates that when steady RANS codes using artificial time stepping encounter convergence failure due to periodic variations in residual norms as well as lift, drag and moments, the limit of steady flow is reached. That this is not the case was shown in Reference [1]. The use of powerful solvers, accurate, stable discretizations and solution adaptive grids allowed a steady flow solution about a high lift airfoil at 90° angle of attack. The physical relevance of such a solution is highly questionable, and it is quite probable the flow steadiness was facilitated by a turbulence model that was being employed beyond its domain of validity. This does call into question unsteady RANS or other methods using turbulence models. It is not clear whether the 90° angle of attack steady solution is stable, but it was not that much harder to achieve than steady solutions at low angles of attack - which are likely stable.

A prerequisite for answering questions about the range of validity of steady flow solutions and the applicability of current turbulence models to flows at the corners of the flight envelope is absolute confidence that the CFD codes employed reliably solve the continuous equations involved. Most cases of practical interest tax the limits of current hardware, so that with many current algorithms residual convergence is suspect and grid convergence is rarely achieved. Too often, failure to agree with experiment is mistakenly ascribed to the turbulence model rather than inadequate numerics. When disagreement occurs it is reasonable that following issues should be addressed in order: residual convergence, grid convergence, discretization consistency and well-posedness of the continuous problem, turbulence and physical modeling applicability, experimental anomalies and bifurcations. It seems to the authors that a great deal of research is still needed regarding

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

residual and grid convergence before proceeding to the other topics. In Section 2 we discuss some issues involved in achieving residual convergence and in Section 3 we consider grid convergence. In Section 4 we briefly discuss bifurcations and multiple solutions, which seem to be common in some areas of the full flight envelope and pose an enormous challenge for CFD solution technology. Finally we draw conclusions in Section 5 regarding the need for algorithm breakthroughs and possible areas of research. This paper is considered to be a sequel to Reference [1].

2. Residual Convergence

Many issues involving residual convergence were discussed in [1] and to a lesser extent in [14]. We elaborate on one of these issues in this paper, namely the need for the ability to reduce residuals to machine error. Most Navier-Stokes codes deliver a solution when a norm of the residuals of the discrete equations is reduced by a specified number of digits beyond that of a starting guess. The number is usually fixed from experience at say three to six. For some codes it would be fruitless to proceed further as weak solvers result in severe slow-downs beyond this point. Unfortunately, there is little convergence experience with solutions at the corners of the flight envelope or even those involving complex geometries. The authors' experience has shown that quantities such as drag and pitching moment can occasionally require six or more digits of convergence to achieve acceptable engineering accuracy. This is especially the case when flows are on the ragged edge of separation or when users are comparing slightly different configurations. In some cases the use of adjoints for grid adaption also requires that the primal solution be correct to more than engineering accuracy.

To reduce confusion and rework, the authors have always believed that CFD codes should never deliver a solution until residual convergence to machine error has been achieved. Even here there is a remote possibility that the error could still be significant. However, for solvers powerful enough to achieve machine error the nature of the convergence generally indicates whether or not a solution has been achieved (as will be obvious from the plots below). For obtaining residual convergence, we favor the use of a globalized Newton's method to solve the nonlinear algebraic equations resulting from discretizing the Navier-Stokes equations [1]. Newton's method proceeds by employing a Taylor series expansion to define a linear system of equations, the solution of which represents an update of the unknowns that will decrease the nonlinear equation residual norms. The authors obtain the solution of this linear system using an iterative sparse solver with a drop tolerance to avoid fill-in. To make up for the dropped elements the linear system is solved via a GMRES algorithm with the sparse solver as a preconditioner. (A side benefit of using such a powerful linear solver is the relative ease with which one can generate quantities such as adjoints during post-processing.) The update is scaled at each step by a line search phase that computes an optimal step length. Such a step length is chosen to achieve the maximum reduction in the nonlinear residual norm subject to various constraints on the solution state. Even though our interest is in a steady solution, we find it is advantageous to include the time derivative. This time term, which is to be driven to zero at steady state, is used as a Levenberg-Marquart type parameter to augment Newton's method, enabling it to circumvent local extrema during the transient phase. For exterior

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

domain problems freestream flow with boundary conditions imposed is used as an initial guess. The existence of a Taylor series is predicated on the nonlinear equations being differentiable. This can be a problem for some turbulence models as well as upwinding schemes involving absolute values of eigenvalues. For safety's sake we employ rounding functions, although they rarely seem to be invoked.

The number of Newton steps required to solve a case from freestream initial conditions is usually on the order of 10 to 100, although occasionally we require over 300. In Figure 1 we show total and turbulence error norm reduction for a typical subsonic wing-body case along with Newton step size, lift, drag and moments. (The case has a freestream Mach number of 0.3 and employs the Spalart-Allamaras (SA) turbulence model [11] at a chord Reynolds number of 5 million.) In Figure 2 we show these quantities for the same case, but at transonic flight conditions (freestream Mach number of 0.78) with a relatively strong Mach 1.38 shock. The starting guess for each case is freestream. Convergence is well defined as residual norms "fall off a cliff" and the Newton step length approaches one. Lift, drag and moment are all within 0.1% at step 31 for Figure 1 and step 46 for Figure 2. One might be tempted to conclude that three digits of residual norm reduction is sufficient for engineering accuracy in such cases. Although widely used, specified digits of residual norm reduction is a rather imprecise indication of convergence, and it is important to be conservative here. In the case of Newton's method the lion's share of the work is generally done by the time such a criterion is satisfied and one may as well finish convergence to machine error in case there is any doubt.

In Figure 3 we show results for a multi- element high lift case with chord Reynolds number of 9 million, again using the SA turbulence model. Even at step 65 with a total residual norm reduction of over five orders of magnitude, drag and pitching moment are off by almost 3%. Worse yet, the lift coefficient would be off by over 25% if the stopping criteria had been 5 digits or a leveling off of forces, and convergence declared at around Newton step 19.We can see this better by plotting lift coefficient versus total residual norm as in Figure 4. The solution initially heads towards a higher lift answer, but is unable to continue reducing the residual norm at this point and must back out to a lower lift solution by temporarily increasing the size of the artificial time term. This may be because an important local phenomenon becomes visible at a residual norm level of around 1.0e-5. It also may be that there are two solutions. However, using several different codes and options we were unable to converge to a higher lift solution on this grid starting at the point where the residual norm began increasing. Of course, to determine whether or not multiple solutions exist one must be confident that any solution generated is a true solution of the continuous equations, and that requires unquestionable residual and grid convergence. The residual norm shelf of Figure 3 is typical of high lift cases and is probably caused by the highly non-linear turbulence equation and the slow build up of the turbulence unknown. For cases involving 300 or more Newton steps, the nature of the flow often changes from attached to separated or vice versa along the way. For adaptively refined grid cases many fewer Newton steps are required, as a good starting solution on each grid can be achieved using reconstruction from the previous grid - often a by-product of the grid refinement procedure. The initial grid still requires a solution from freestream of course, but the grid is typically small. When the flow suddenly attaches or separates on some grids, almost as many Newton steps are required as using a freestream initial guess.

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

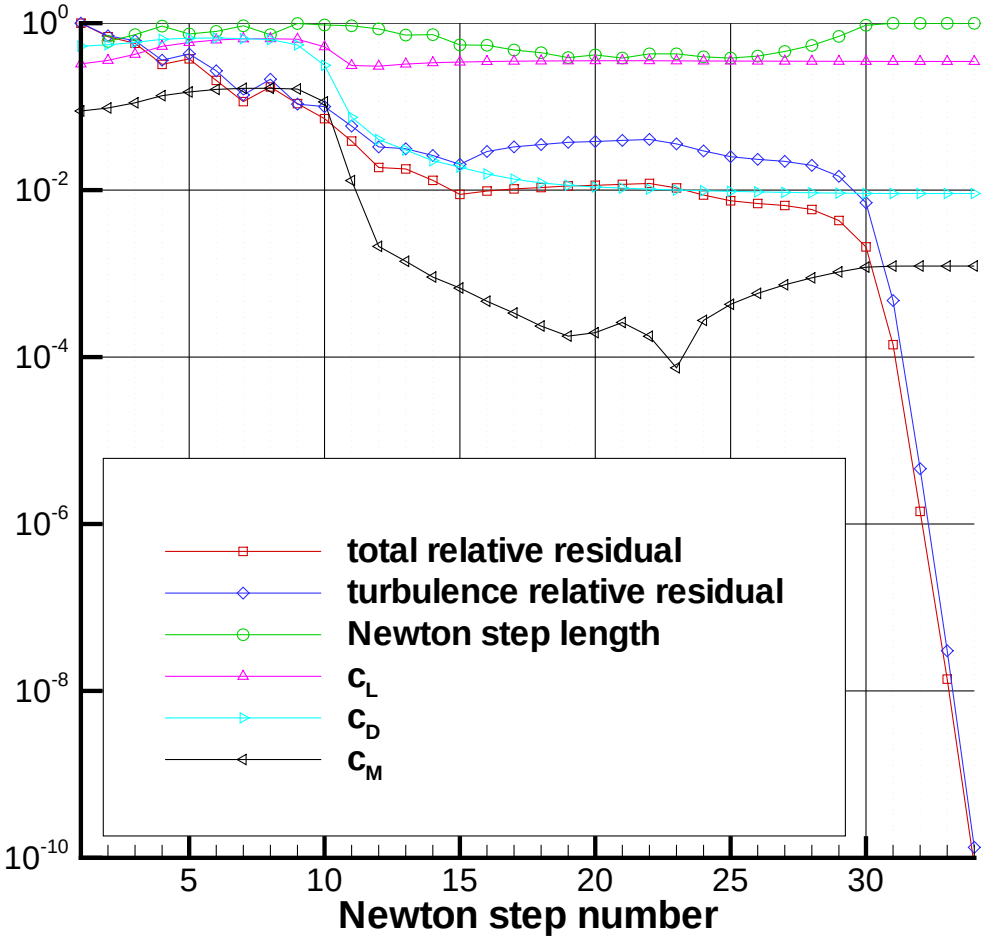


Figure 1: Convergence and Force History for a Subsonic Wing-body Configuration.

It is important to note that no solver, however powerful, can overcome a poor discretization. Many current discretization techniques implicitly assume grid regularity and sufficient density to resolve flow features. This places a tremendous burden on grid generation techniques. In all likelihood, these requirements cannot be ensured everywhere even for structured grids of 30-300 million nodes about complex configurations. Anticipating the discussion in the next section, most grid refinement algorithms lead to unstructured grids. The more flexible and versatile methods produce grids that are initially under-resolved and ultimately have high aspect ratio cells with a wide disparity in adjacent sizes and shapes. Even finite element discretization techniques can lead to near singular operators here if straightforwardly applied. Not only must a method survive such grids, but it must reasonably predict where and how to refine or redistribute the grids. A consid-

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

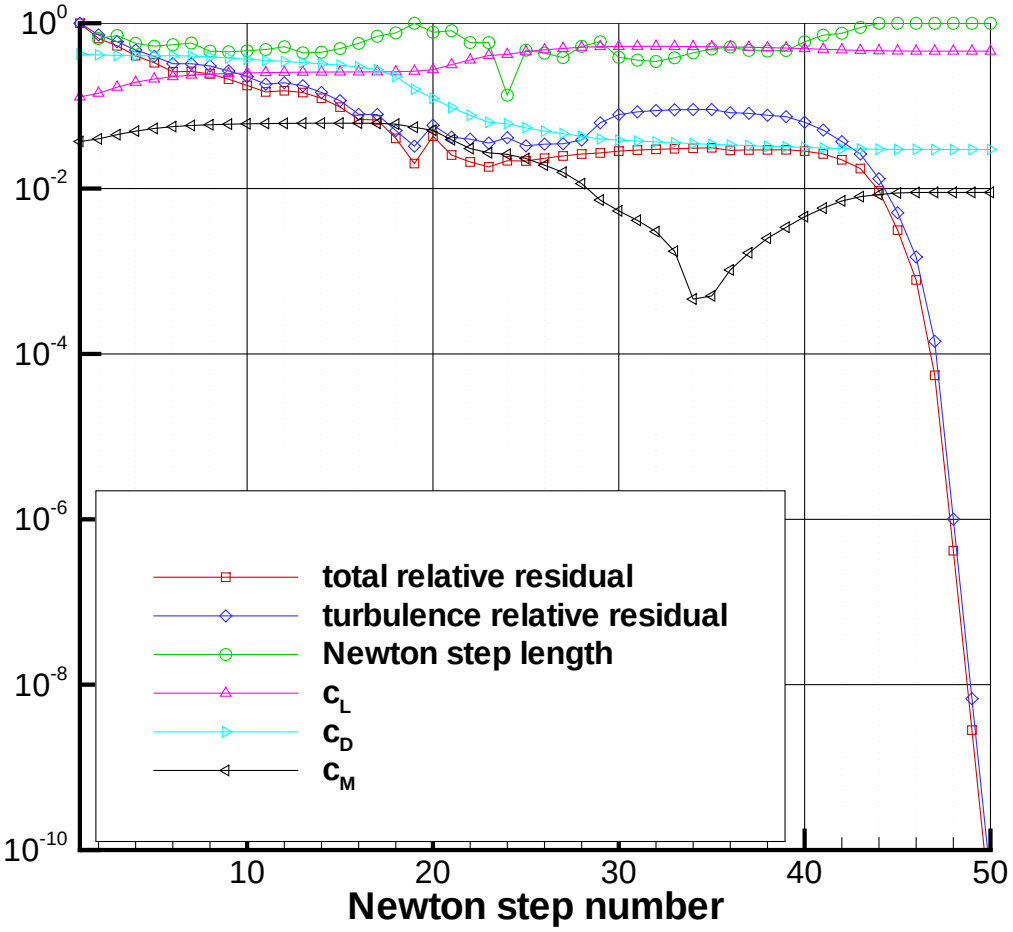


Figure 2: Convergence and Force History for a Transonic Wing-body Configuration.

erable effort was expended by the authors in Reference ([1]) to ensure second order, nonsingular discretizations, but much more needs to be done, especially with respect to higher order methods [14].

3. Grid Convergence

Currently, the vast majority of CFD runs are made with fixed grid codes where the initial grid is the final grid. Occasionally these codes employ grid sequencing, i.e., solutions are obtained on a sequence of three or four increasingly finer grids. In most instances this is more for facilitating residual convergence via a multigrid method than for obtaining grid convergence. On occasion,

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

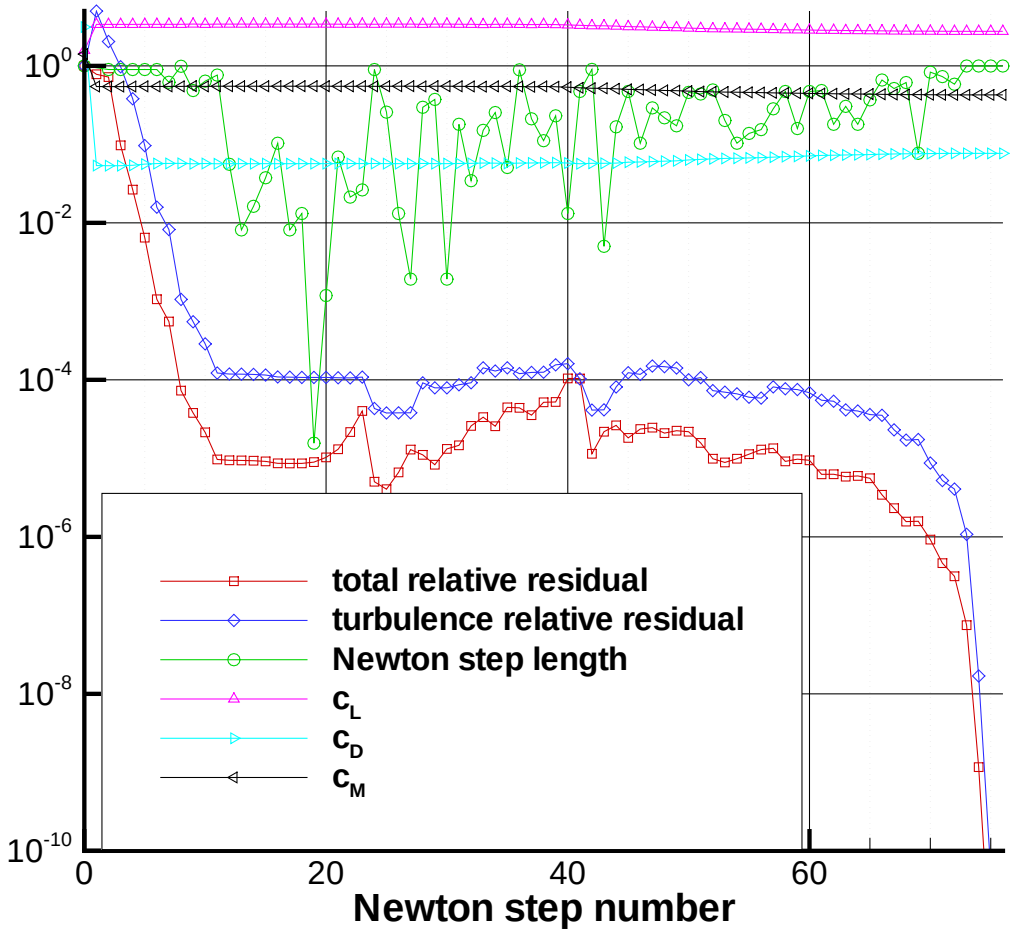


Figure 3: Convergence and Force History for a Multi-element High Lift Case.

the grids are really intended to assess grid convergence, e.g., [12]. While the fine grids can be a factor from 2 to 30 finer than the base grid, one has to realize that in three dimensions a factor of 8 would be necessary for just one uniform refinement. Usually the finer grids are all constructed using the same rules of thumb, so that seriously inadequate local gridding will remain inadequate. One notable exception to solving on fixed grids is the full potential/integral boundary layer code TRANAIR ([15], [2]), which from its inception in 1985 was intended to be a solution adaptive grid code. Its success (well over a million complex geometry cases run to date) has inspired the authors to seek a similar capability for solving the Reynolds Averaged Navier-Stokes equations about complex geometries in the corners of the flight envelope.

Current fixed grid Navier-Stokes codes rely on user experience to generate good hand crafted

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

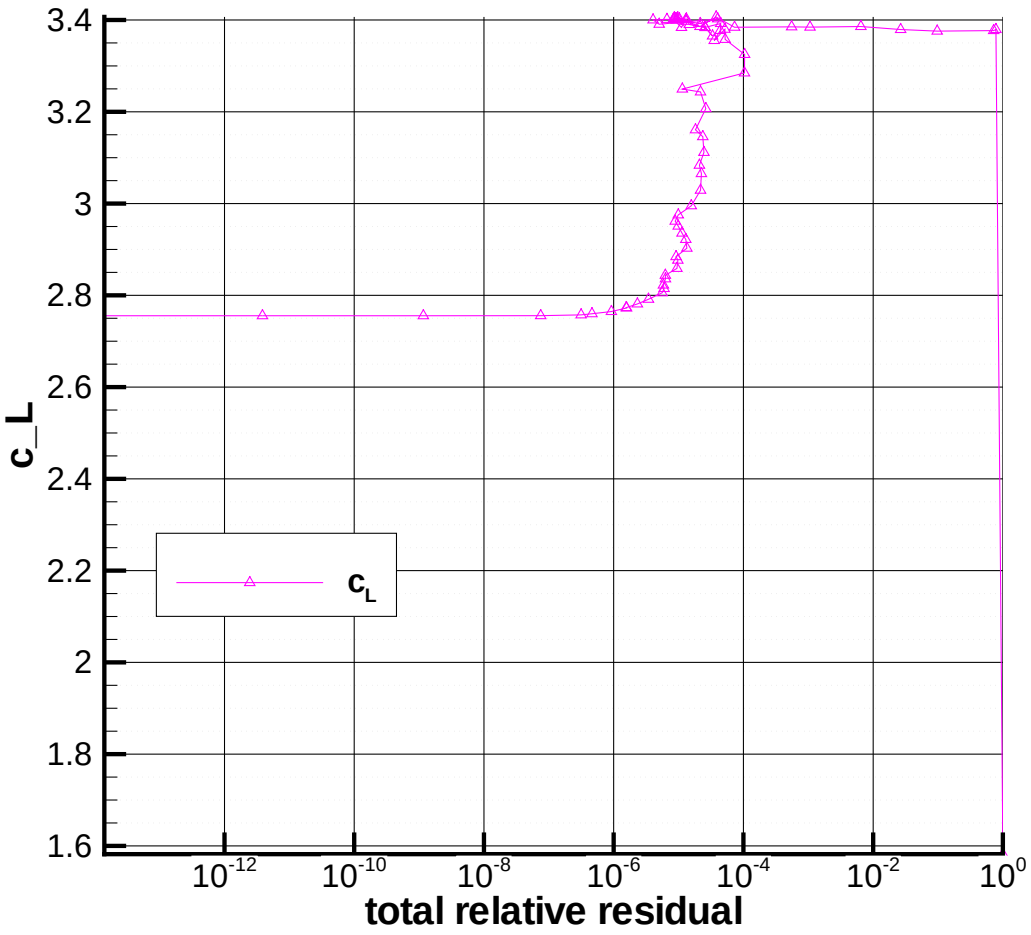


Figure 4: CL Versus Total Relative Residual for the High Lift Case.

grids. As pointed out in [1], the generation of a good grid is still somewhat of an art and often quite labor intensive. There is no precise definition as to what constitutes such a grid. In fact, the definition would probably vary from code to code and is certainly case dependent. More than likely the grid is compromised by what is possible on given hardware within a given turnaround period. Often boundary layer rules of thumb such as a y+ of 1 and a stretching ratio of $1.05 - -1.2$ are used even though current adjoint methods indicate that these rules are not universally applicable. Moreover, little attention is paid to the inviscid part of the flow field, where adjoint methods indicate that refinement is sometimes required even when flow gradients are almost zero. Still, current fixed grid methods have been very useful in the design of transport aircraft and well worth the effort required to gain the necessary user experience. It is questionable whether this is

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

the right route for the increasing variety flows and geometries that span the full flight envelope. We note here that solution adaptive gridding in the TRANAIR code was a paradigm shift that enabled transonic flow analysis and design of complex geometries in the late 1980's.

We are convinced that solution adaptive gridding is also the key to the routine use of CFD for the full flight envelope. Much work has already been done in this direction, e.g., [13, 8]. Both solution adaptive grid refinement and grid redistribution have been investigated. Currently, we prefer grid refinement over grid redistribution. The latter can be somewhat more efficient in theory, but perhaps more prone to cycling. The former only requires generation of an initial inviscid or Euler grid, which is much simpler task for complex geometries than periodically generating high aspect ratio viscous grids.

Solution adaptive grids require error indicators to establish density, orientation and aspect ratio of the corresponding cells. Some methods use local data such as the Hessian of a scalar quantity (e.g., pressure or Mach number). Others use global data via the calculation of adjoints. After some testing on our own, we believe that the use of multiple adjoint solutions to assess grid convergence and to aid refinement/derefinement decisions must be an important feature of grid adaption. There are a number of reasons for this. One is the failure of local indicators to deemphasize features which have no bearing on engineering quantities of interest. Early in the development of TRANAIR we were puzzled by the lack of improved accuracy when continually allowing refinement with up to a 100000 new nodes. It turned out that most of the nodes went into the tip vortex which trailed from the wing all the way to the wind tunnel diffuser. This led to the introduction of user specified field blocks inside which refinement was permitted only up to a certain level. Although users ultimately got used to the added burden it was always viewed quite unfavorably. Realistically, there may be no way around such a burden, but global error indicators using adjoints can probably mitigate the problem to a large extent.

A second reason is the ability to predict functional values on refined grids without having to solve on them. This allows increased accuracy with less work, and in addition contributes to the assessment of grid convergence. A third reason is the practical success described in [13, 8], where adjoint based adaption was applied to a high lift turbulent flow case - no small feat as we can attest. A fourth reason is our own 2D studies where, for example, the addition of 137 new nodes via an adjoint method to an inadequate 30557 node hand crafted grid yielded a lift coefficient more accurate than that of a hand crafted grid containing 64073 nodes [1]. A fifth reason is the sheer elegance and power of the theory, e.g., [10]. Being able to predict flow properties with grids orders of magnitude smaller than presently required might allow one to actually achieve grid converged solutions in three dimensions, something that rarely happens now. On the other hand, practical implementation difficulties seem to grow as enhancements to the theory provide more and more accuracy. At the current time we would probably be satisfied with modest improvements to the current state of the art in three dimensions.

3.1. Demonstration Problem

To illustrate some of the benefits and issues of adjoint based adaptation we present an explicit implementation of a fairly simple, straightforward example. It involves the solution of the linear

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

convection/diffusion equation in one dimension,

$$F(u, u_x, x)_x = 0, \tag{3.1}$$

where

$$F(u, u_x, x) \equiv cu - \nu(x)u_x; \quad u = u(x), 0 \leq x \leq 1, \quad u(0) = 1, u(1) = 0,$$

and

$$c = 1, \quad \nu(x) = c(1 + e^{-2\phi})/(2\lambda), \quad \phi = \lambda(1 - x), \quad c = 1, \quad \lambda = 10.$$

The subscript x here denotes total derivative with respect to x. The solution to Equation (3.1) is

$$u(x) = \tanh(\phi)/\tanh(\lambda), \tag{3.2}$$

and is shown in Figure 5. Increasing λ would result in steeper and steeper slopes at the right endpoint.

For a functional of interest we calculate,

$$D = F(u(1), u_x(1), 1)$$

The quantity D would correspond to skin friction in a laminar boundary layer although the profile of Equation (3.2) is not exactly a boundary layer profile. Integrating a quantity like D over a surface in higher dimensions would correspond to skin friction drag so we will often refer to D as drag.

To solve Equation (3.1) we approximate the continuous problem as follows. We define nodes:

$$0 = x_0 < x_1 ... < x_{i-1} < x_i < ... \, x_{n-1} < x_n = 1,$$

and solution unknowns at these nodes,

$$u_i, \quad i = 0, n.$$

We then define discrete equations and a functional using the finite element method. We assume $u(x)$ to be the linear interpolant of nodal values in each cell $x_{i-1} \leq x \leq x_i, i = 1, n$. We also define a continuous test function, w, which has arbitrary values defined at all nodes, w_i, and is linear in cells as well. A finite element or weak statement of the problem which generates discrete equations to be solved is then

$$0 = \int_0^1 -w_x F(u, u_x, x)dx + [w\underline{F}(u, u_x, x) + \sigma(u - U)]_0^1, \tag{3.3}$$

where $U = 1$ at $x = 0$, $U = 0$ at $x = 1$, and values of σ at $x = 0$ and 1 are arbitrary along with all w_i.The bar under F in the left most boundary term indicates that F is to be averaged over the center of the cell adjacent to the boundary. The solution to Equation (3.3) produces the Galerkin approximation to the true solution which is known to be unstable on under-resolved grids due to the

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

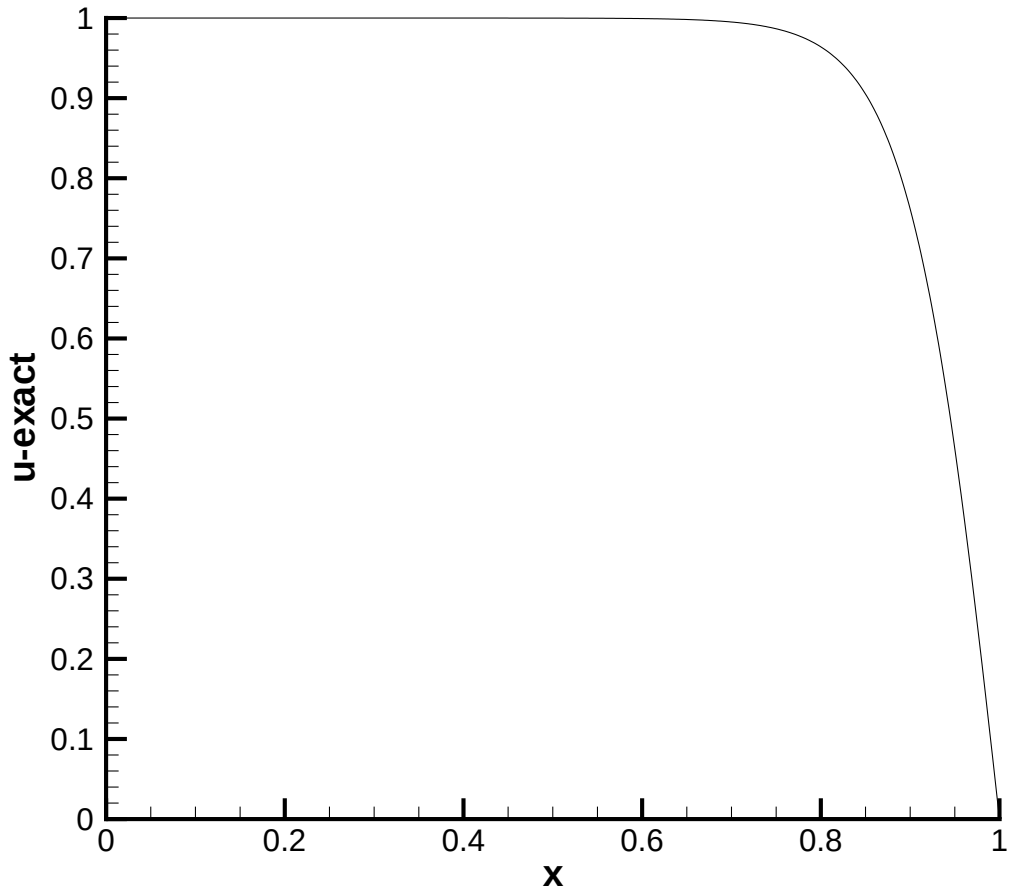


Figure 5: Exact Solution to Convection/Diffusion Equation with c = 1 and λ = 10.

advection term cu_x. There are a number of ways to remedy this problem. We favor the Streamwise Upwind Galerkin (SUPG) approach, which was originally designed for this type of problem and has a compact stencil [5]. It augments the flux F by in this case by replacing $\nu(x)$ with

$$\mu(x) = \nu(x) + \tau(x)c^2. \tag{3.4}$$

Here $\tau(x)$ is a positive quantity which is second order small in cell size h, but sufficiently large to eliminate the instability. There is a known value of τ that yields a nodally exact solution to this problem with constant ν on any grid. It will not generalize to other advection/diffusion problems and higher dimensions. For that reason we choose a definition similar to the one we use generally, i.e.,

$$\tau(x) = \frac{1}{2\frac{c}{h} + 4\frac{\nu(x)}{h^2}}.$$

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

After performing the integration, the discrete equations defined by Equation (3.3) (coefficients of σ and w_i) are then

$$F_{i+1/2} - F_{i-1/2} = 0; \quad i = 1, n-1; \quad u_0 = 1, u_n = 0, \tag{3.5}$$

where

$$F_{i-1/2} = c(u_i + u_{i-1})/2 - \mu_{i-1/2}(u_i - u_{i-1})/h_{i-1/2} \quad \text{with} \quad h_{i-1/2} = x_i - x_{i-1}, i = 1, n,$$

and

$$\mu_{i-1/2} = \nu_{i-1/2} + \tau_{i-1/2}c^2.$$

The quantities $\nu_{i-1/2}$ and $\tau_{i-1/2}$ represent averages over the cell with end nodes $i-1$ and i and are given by,

$$\nu_{i-1/2} = c\frac{1 + \frac{\sinh[\lambda h]}{[\lambda h]}e^{\lambda(x_i + x_{i-1} - 2)}}{2\lambda},$$

and

$$\tau_{i-1/2} = h\frac{2\lambda h - \log\left[\frac{(c+2h\nu_i)}{(c+2h\nu_{i-1})}\right]}{4(1+\lambda h)}.$$

We now define the functional,

$$J(u, w) \equiv [(\eta + w)\underline{F}(u, u_x, x, h) + \sigma(u - U)]_0^1 + \int_0^1 -w_x F(u, u_x, x, h)dx,$$

where

$$\eta = 1 \ \text{ at } \ x = 1 \ \text{ and } \ \eta = 0 \ \text{ at } \ x = 0.$$

If $u_i, i = 0, n$ is a solution to Equation (3.3), then J is equal to D for any values of w_i and σ . More precisely, J is equal to D evaluated at the midpoint of the right hand most cell. This slight modification corresponds to the practice of evaluating forces and moments via summed (discretely conserved) numerical fluxes rather than numerically integrated surface quantities. Again note that in the continuous limit F is constant so that there is little loss in accuracy from such an assumption. We also denote F as an explicit function of cell size now due to the fact we have added explicit artificial dissipation based on cell size. Then we have under the above assumptions

$$J = F_{n-1/2} = D. \tag{3.6}$$

Assume we have solved Equation (3.5) obtaining a value for D on a base grid, and want to obtain a more accurate value. We consider refining the base grid by dividing a subset of the cells in half with cell midpoints as new nodes. We would like to achieve the potential improvement in accuracy without the expense of solving on this finer grid. Note that in three dimensions a uniformly refined grid would be 8 times the size of the base grid, which may already tax the limits of the hardware. We define

$$\hat{J}(\hat{u}, \hat{v}) \equiv [(\eta + \hat{w})\underline{F}(\hat{u}, \hat{u_x}, x, \hat{h}) + \hat{\sigma}(\hat{u} - U)]_0^1 + \int_0^1 -\hat{w}_x F(\hat{u}, \hat{u}_x, x, \hat{h})dx$$

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

where the hat symbol denotes quantities on the finer grid. It is sufficient then to estimate ΔJ , where Δ denotes the difference between fine and base (now considered coarse) grid quantities. $w, \hat{w}, \sigma$ and $\hat{\sigma}$ are still arbitrary. There are many ways to proceed here, but we take a conservative approach by choosing $\hat{w} = w$ and $\hat{\sigma} = \sigma$. These choices are permissible as we can always take $\hat{w}$ equal to w at the coarse grid nodes and equal to the linear interpolant of the coarse grid nodal values at the new nodes. We have then,

$$\Delta J = [(\eta + w)\Delta \underline{F} + \sigma \Delta u]_0^1 + \int_0^1 -w_x \Delta F dx;$$

where

$$\Delta F = c\Delta u - \hat{\mu}\Delta u_x - \Delta \mu u_x.$$

Note that $\underline{F}$ for the fine and coarse grids may be defined at different locations in case the end cells are refined, but it won't make any difference anyway, since we now specify $w = -\eta$ at the endpoints.

Let $\Delta u = \Delta u^* + \Delta u'$, where on a given coarse grid cell Δu^* is the difference between the fine and coarse grid linear functions interpolating values at the two coarse grid nodes. Then $\Delta u'$ is the piecewise linear "hat" function which is zero at the two coarse grid nodes and equals $\hat{u} - u - \Delta u^*$ at the cell center. See Figure 6 for an illustration.

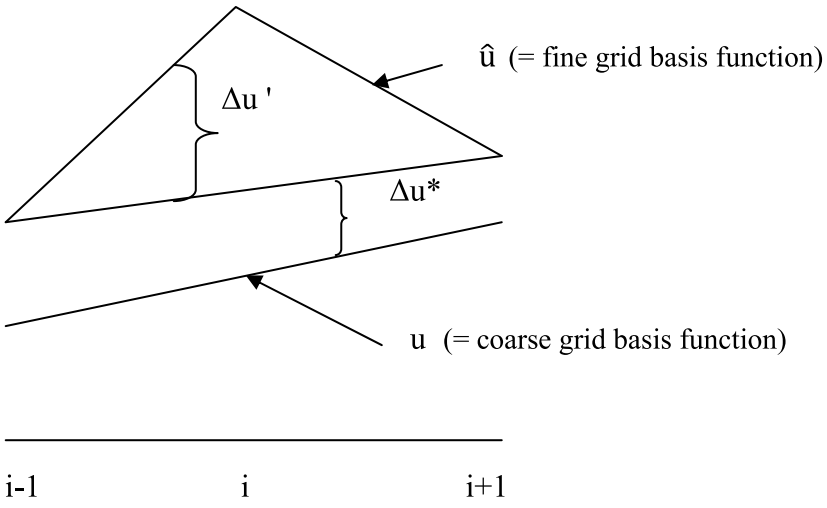


Figure 6: Fine Grid Basis Function Increments over that of Coarse Grid.

We have then that $\Delta J = \Delta J^* + \Delta J'$, where

$$\Delta J^* = [\sigma \Delta u^*]_0^1 + \int_0^1 -w_x[c\Delta u^* - \hat{\mu}\Delta u_x^*]dx$$

$$= [(\sigma - cw)\Delta u^*]_0^1 + \int_0^1 \Delta u_x^* G(w, w_x, x, h)dx;$$

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

with

$$G = cw + \hat{\mu} w_x,$$

and

$$\Delta J' = [(\sigma - cw)\Delta u']_0^1 + \int_0^1 [\Delta u'_x G(w, w_x, x, h) + \Delta\mu w_x u_x] dx$$

If we now choose $\sigma = cw - \underline{G}$ at $x = 0$ and $x = 1$, then ΔJ^* is zero if the coefficients of Δu_i^* are zero for $i = 1, n-1$, i.e.,

$$G_{i+1/2} - G_{i-1/2} = 0, \quad i = 1, n-1; \tag{3.7}$$

where

$$G_{i-1/2} = c\frac{(w_i + w_{i-1})}{2} + \hat{\mu}_{i-1/2}\frac{(w_i - w_{i-1})}{h_{i-1/2}}, \quad i = 1, n.$$

(The fine grid $\hat{\mu}_{i-1/2}$ at coarse grid cell centers is defined as the average of the values on both sub-cells, i.e., $\hat{\mu}_{i-1/2} = (\hat{\mu}_{i-3/4} + \hat{\mu}_{i-1/4})/2$, where $\hat{\mu}_{i-3/4}$ is the averaged value of $\mu(x)$ on the left sub-cell and $\hat{\mu}_{i-1/4}$ is the averaged value of $\mu(x)$ on the right sub-cell. Along with $w = -\eta$ at $x = 0$ and $x = 1$ (defining w_0 and w_n) these are the adjoint equations. Note that in this particular case the discrete adjoint Jacobian is not precisely the transpose of the discrete forward Jacobian due to the presence of the fine grid $\hat{\mu}$ as well as the Dirichlet conditions at at $x = 0$ and $x = 1$. The exact solution of the adjoint equations in the continuous limit is,

$$w(x) = -1 + \frac{(1 - e^{-2\phi})}{(1 - e^{-2\lambda})}.$$

A plot of the exact adjoint is shown in Figure 7. As expected, it seems to have the opposite character of the primal solution u, and the gradient of the adjoint is large where the curvature of u is small and vice versa.

So far no approximation has been made. Once we solve Equation (3.7) for w, we can compute $\Delta J'$ if we know $\Delta u'$. Here we have to assume that $\Delta u'$ can be estimated by using only coarse grid information. To do this we fit nodal values of u on the coarse grid with higher order polynomials to reconstruct quadratically accurate values of u at cell centers. Then, $\Delta u'$ is the piecewise linear "hat" function which is zero at the cell endpoints and equal to the increment of the reconstructed value of u over the actual value of u at cell centers. As $\Delta u'$ is zero at $x = 0$ and $x = 1$, the end point terms in $\Delta J'$ go away. Performing the remaining integration:

$$\Delta J = \Delta J' = \sum_i^n \left\{ [w_i - w_{i-1}] \left[\Delta u'_{i-1/2}(-\frac{c}{2} - \hat{\mu}_{x,i-1/2}) + \Delta\mu_{i-1/2} u_{x,i-1/2} \right] \right\}, \tag{3.8}$$

where

$$\hat{\mu}_{x,i-1/2} = \frac{(\hat{\mu}_{i-1/4} - \hat{\mu}_{i-3/4})}{(h_{i-1/2}/2)}$$

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

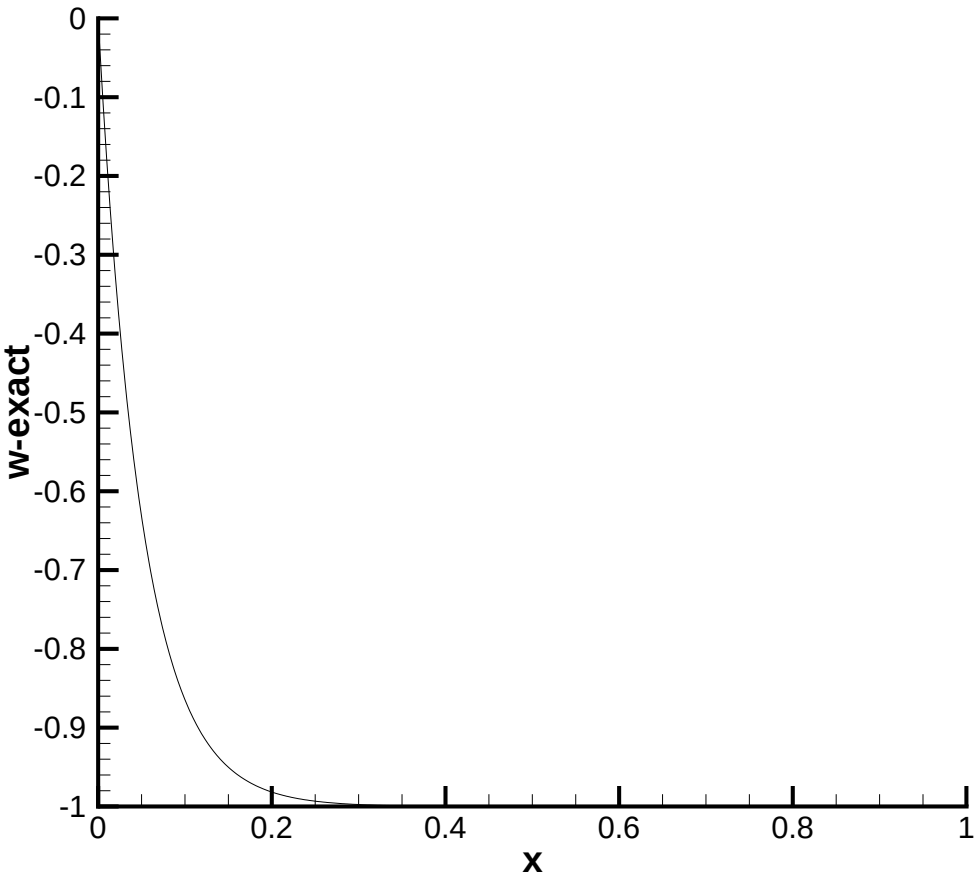


Figure 7: Exact Adjoint to Convection/Diffusion Equation with c = 1 and μ = 0.05.

and

$$u_{x,i-1/2} = \frac{u_i - u_{i-1}}{h_{i-1/2}}.$$

Here $\Delta u'_{i-1/2}$ is the previously described increment at the cell centers and $\Delta\mu_{i-1/2}$ is the difference between the fine and coarse grid $\mu_{i-1/2}$. Aside from size, the cell contribution to ΔJ depends on the gradient of the adjoint as well as the curvature (and to a lesser extent the gradient) of the primal. The first term in the cell contribution represents the improved accuracy of the fine grid due to lower truncation error and the second represents the improvement in accuracy due to lower artificial dissipation. The latter cannot be ignored as changes in μ are of the same order as μ itself when the addition of the artificial dissipation term of Equation (3.4) is important.

The absolute value of the term in curly brackets in Equation (3.8) is considered to be the local cell error indicator on the grounds that cells which are adequately small in size would not

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

significantly contribute to ΔJ. By refining cells with large error indicators $\Delta u'_{i-1/2}$, and $\Delta\mu_{i-1/2}$ will be reduced, thereby reducing their contributions. Of course this reasoning assumes the solution is smooth.

3.2. Computed Results

In Figures 8, 10, 11, 13 and 14, each symbol represents a grid with the number of points given by the abcissa, **nnode**. Considering Figure 5, one would intuitively begin a refinement process for an equally spaced grid by refining the right hand most cells where gradients and curvatures are extremely large and the skin friction is defined. However, Equation (3.5) is a discrete conservation equation so that what happens at the right endpoint is directly related to what happens at the left endpoint. Considering Figure 7 and the error indicator of Equation (3.8), it is unclear which cells should be refined first. For the purposes of experiment we start out with one cell and then add cell refinements one cell at a time. While this is not a good strategy from a practical point of view, it is a good test, as any method should be successful here or else have the difficult task of finding an empirical rule for the minimum number of cells required to be refined on each grid.

In Figure 8 we show D from Equation (3.6) as a function of the number of nodes (nnode) in each coarse grid using the error indicator of Equation (3.8). For this case we reconstruct $\Delta u'_{i-1/2}$ using infinite weights on the two cell endpoints and equal weights on its nearest two neighbors. (For the first and last cell we calculate $\Delta u'_{i-1/2}$ by linearly extrapolating its value from the adjacent two cells if they exist.) The computed value minus the exact value uses the finite element solution (Equations (3.5) and (3.6)) and the exact solution (Equation (3.2)) respectively, and is denoted by Dcomputed-Dexact. The predicted value minus exact value for a uniformly refined grid uses Equation (3.8) and is denoted by D2xpredicted-Dexact. The computed value minus the exact value on the actual uniform refinement of the current grid is denoted by D2xactual-Dexact. Successive grids are refined one cell at a time according to the maximum absolute value of the summands of Equation (3.8). The (signed) summands versus x coordinate for a typical grid are shown in Figure 9. The coordinates of the centers of the refined cells (denoted xref) are shown in Figure 10. Finally, as a check on reconstruction accuracy, the estimated values of $\Delta u'_{i-1/2}$ for the coarse cell of index i to be refined and the actual values of $\Delta u'_{i-1/2}$ on the corresponding uniformly refined grid are shown in Figure 11.

First we observe that the refinement process seems to be successful, although the predicted values for a uniformly refined grid do not head in the right direction until grid 13. The difference between the actual values and these predicted values is one measure of the adequacy of the grid, but obviously it cannot be the only one given the early coincidental agreement. Multiple functional predictions would help in this regard. The adjoint for grid 5 is plotted in Figure 12. Up to grid 7 the adjoint exhibits instabilities. In some sense, the grids up to that point are under-resolved as the fine grid μ is too small to prevent adjoint oscillations on the coarse grid. An instability in the adjoint is less of a severe problem than one in the primal, as the adjoint equations are linear. However, for more complex problems involving such instabilities any liberties taken with the discretization could result in a worse than useless error indicator. In Figure 11, for example, the grids from 7 to 13 are probably not fine enough for the computation of a sufficiently accurate prediction of

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

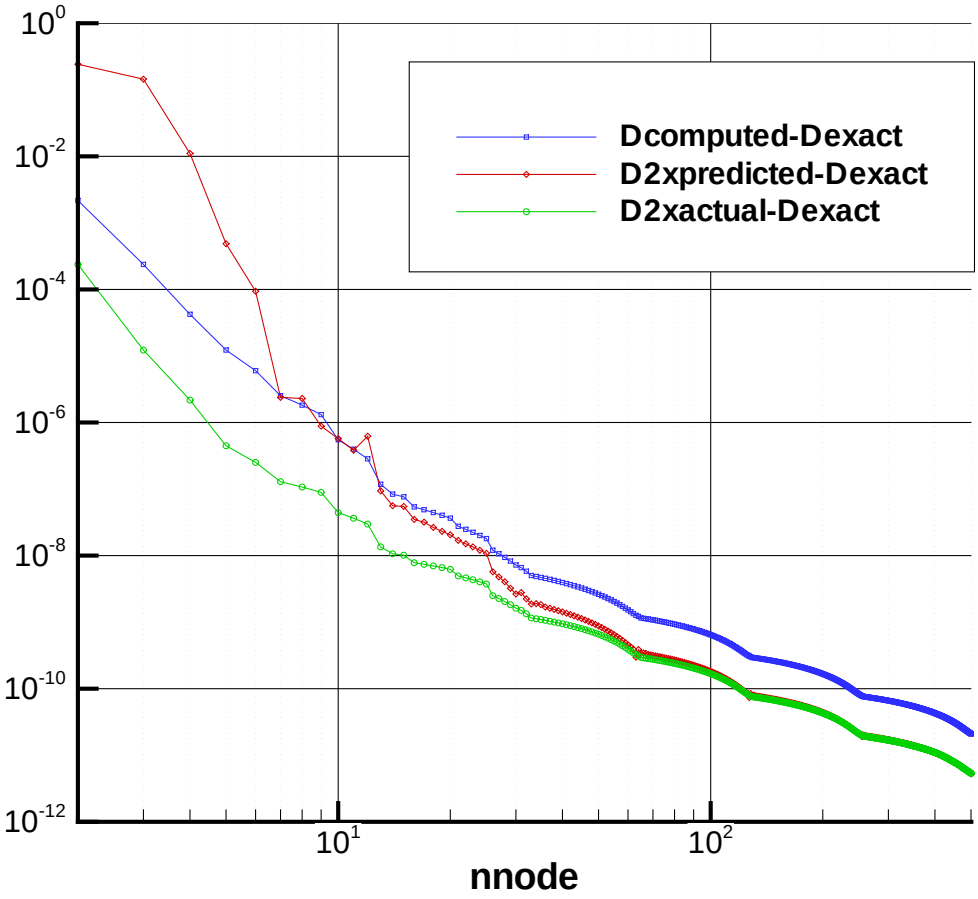


Figure 8: Computed, and Uniform Refinement Predicted and Actual Errors in D Using Adjoint Based Refinement.

$\Delta u'_{i-1/2}$. Contrary to intuition, our error indicator generally chooses to refine large cells whether or not the solution has large gradients and/or curvatures in the neighborhood of these cells. Figure 13 shows the results if we just use a local interpolation error as our refinement error indicator (i.e., absolute value of $\Delta u'_{i-1/2}$ in Equation (3.8)). Figure 14 shows the corresponding locations of the refined cells. In 500 refinement steps the method fails to refine any cells left of $x = 0.25$ and therefore never enters the asymptotic regime where predictions are valid. While this type of failure is undoubtedly exacerbated by the special one dimensional nature of the problem, it seems to occur all too often in higher dimensions as well.

Finally, we mention that a number of possibly more accurate options exist even when choosing $\hat{w} = w$. If the reconstruction of $\Delta u'$ is linear in coarse grid nodal values, then one can perform it using fine grid values on coarse grid nodes instead. These fine grid values would be eliminated by

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

an adjoint with an enlarged stencil. One can also reconstruct on the fine grid with the additional help of the primal differential equation, again resulting in a modified adjoint equation, but with better stability.

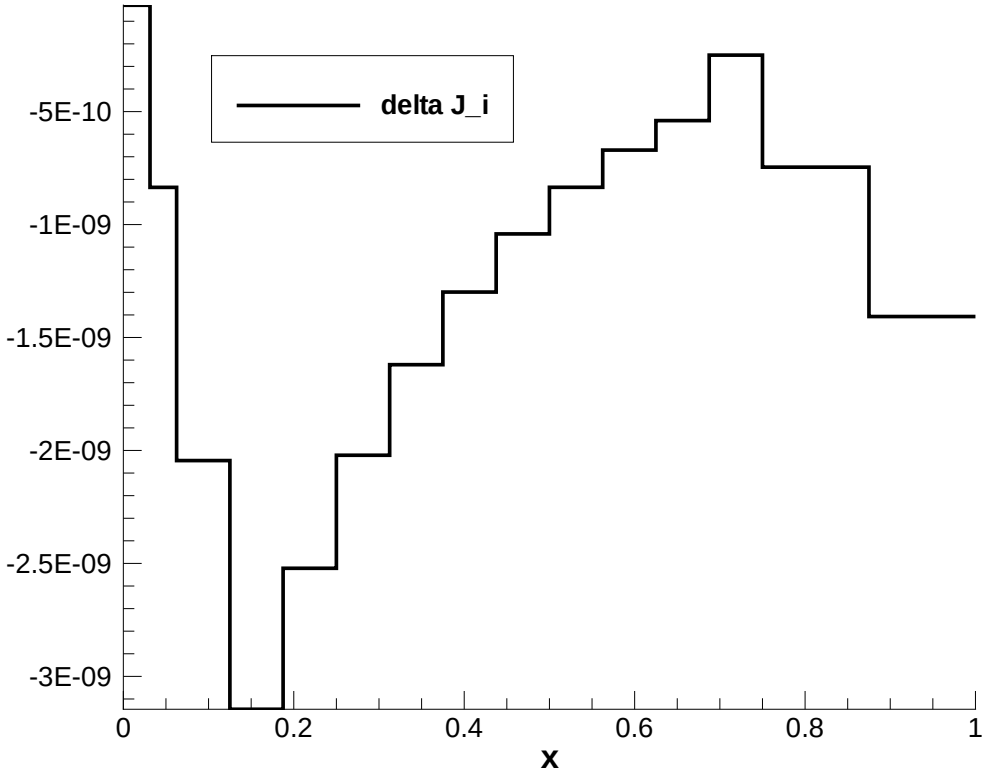


Figure 9: Signed Summands of Equation (3.8) for grid 15.

3.3. Issues

In the demonstration case of the previous sub-sections, the use of adjoint based refinement allowed one to get past the under-resolved and non-asymptotic refinement regimes in a reasonable number of refinement steps. Whether this holds for large, complex, non-linear problems is an open question, although our limited experience so far indicates that it does.

It is evident that the critical task associated with evaluating ΔJ (as well as with numerous other CFD related quantities) is the reconstruction process, i.e., the algorithm to evaluate $\Delta u'$. For flows involving unresolved features such as shocks and shear layers enormous problems exist. For smooth flows the situation is a little better, as one can use weighted least square fits or contour integrals. However, in two and three dimensions, near singular node locations can cause severe

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

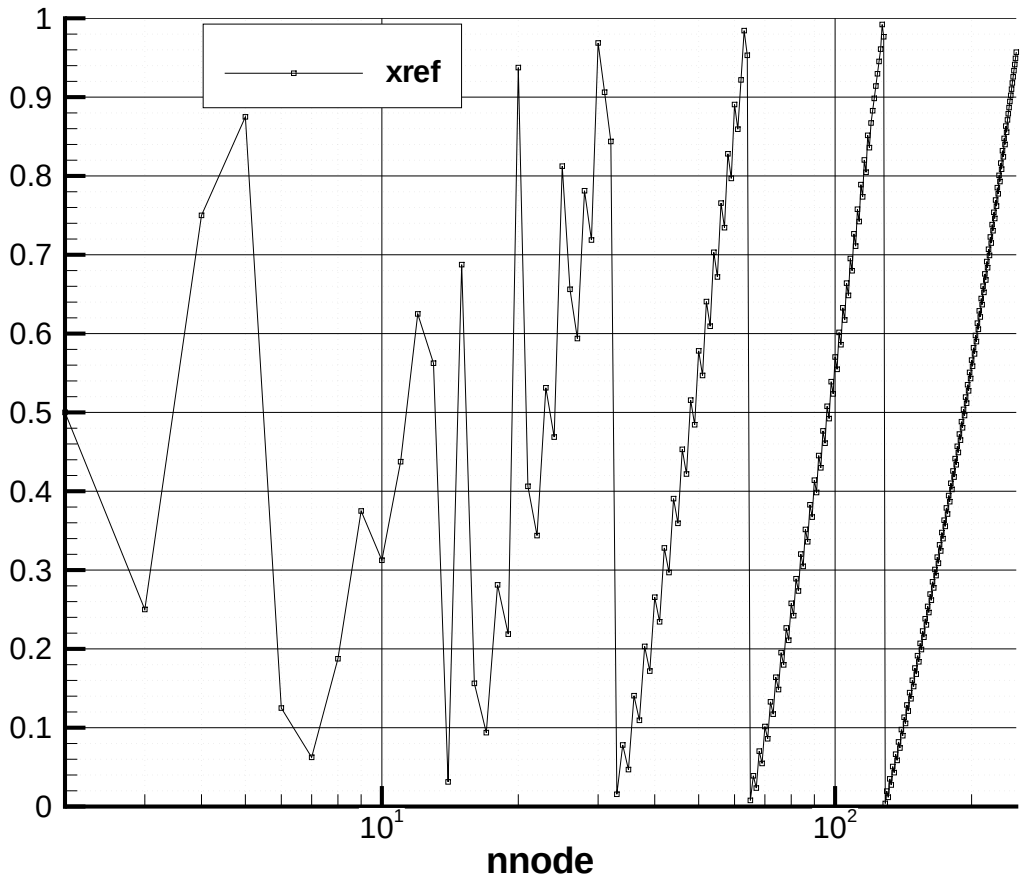


Figure 10: Coordinates of Centers of Refined Cells Using Adjoint Based Refinement.

problems. This is especially true for unstructured viscous grids with high aspect ratio cells and wide disparities in cell sizes and shapes, as well as for under-resolved curved boundaries. We have seen cases where a good least squares algorithm yielded reconstructed values two orders of magnitude larger than any values being interpolated. Of course, one could limit reconstructed values after the fact, but then the assumed quadratic accuracy of the result would be voided. This limiting does seem to allow useful but not optimal results in practice. For grids with 30-300 million nodes it is unlikely that anomalous reconstructions would not arise. It only takes one disastrous reconstruction to totally invalidate the computation of Equation (3.8). Even if one only uses Equation (3.8) for a cell error indicator, a disastrous reconstruction can feed on itself yielding worse and worse grids. Reconstruction is perhaps the most under-researched topic in CFD today.

A number of authors have made good progress in identifying reconstruction problems, e.g., references [7], [4] and [9]. Petrovskaya [9] has proposed modifications to the standard weighted

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

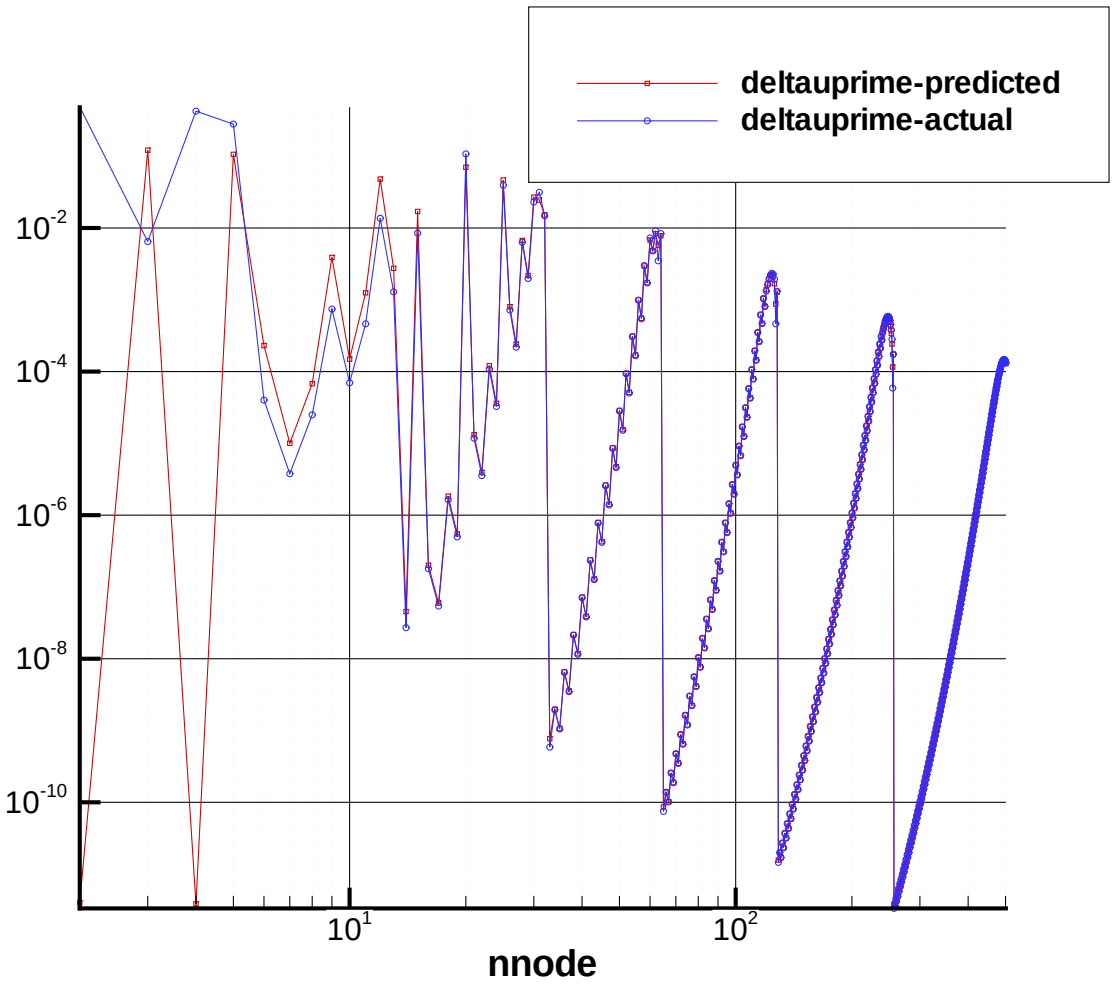


Figure 11: $\Delta u'_{i-1/2}$ Predicted and Actual Values on Each Grid.

least squares procedure to deal with numerically distant points, but is somewhat pessimistic about that chances of successful reconstruction without help from the refinement process. Mavriplis [7] believes that aligning the grid with the flow mitigates such problems. However, it is hard to imagine a 30-300 million node grid without some severe local flow anomalies. A method that eliminates the problem altogether is to solve on a fine grid which is always a uniform refinement of a coarse grid, see Demkovicz [3]. The fine grid would, of course, be an affordable grid. Adjoint analysis on the coarse grid would dictate which coarse grid (and hence fine grid) cells would be refined. Lagging the refinement in such a way would be suboptimal and one would not be able to use prediction techniques on the fine grid to obtain an accurate estimate on an even finer, unaffordable grid.

Clearly, a great deal more work needs to be done. One approach we use to circumvent the problem of under- resolved curved boundaries described in [1] is to use distance as a coordinate in

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

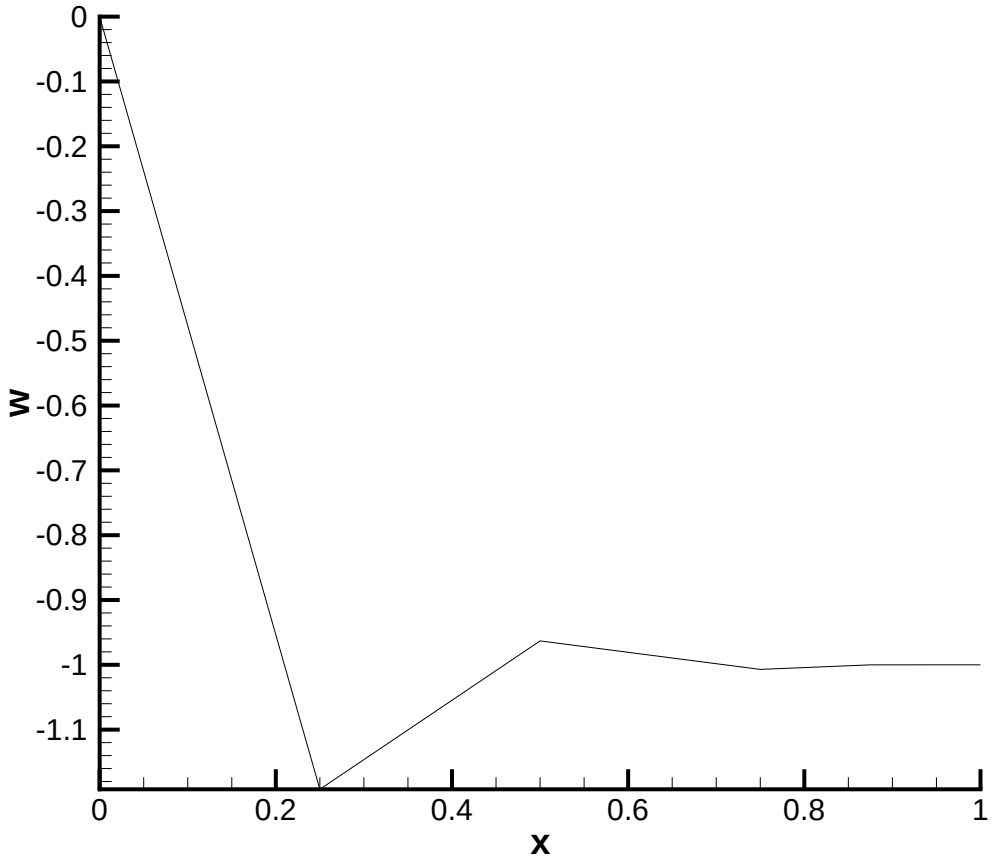


Figure 12: Adjoint Versus X-coordinate for Grid 5.

our least squares reconstruction procedure. Another approach to avoiding catastrophic fits when reconstructing values via least squares at edge midpoints is to define infinite weights at the end point nodes and then adjust the remaining weights to define a quadratically accurate fit which yields the least excursion from linear. For nearest neighbor fits in one dimension this corresponds to fitting a quadratic using either the left or right neighbor. Such an approach seems to work well for smooth solutions. We suspect that accurate functional prediction in cases involving shocks is problematic for almost any reconstruction method. However, reconstruction does appear to allow useful results for grid refinement purposes in such cases. Using the adjoint error indicator seems to naturally limit shock refinement, which certainly helps with residual convergence. On the other hand one would normally expect to refine grid much faster in the neighborhood of shocks in order to maintain accuracy in the rest of the field. At any rate, grid refinement in the presence of shocks seems to be an open subject. Pierce and Giles ([10]) accounted for shock movement in one dimension using shock fitting methods. We are looking into that and other means.

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

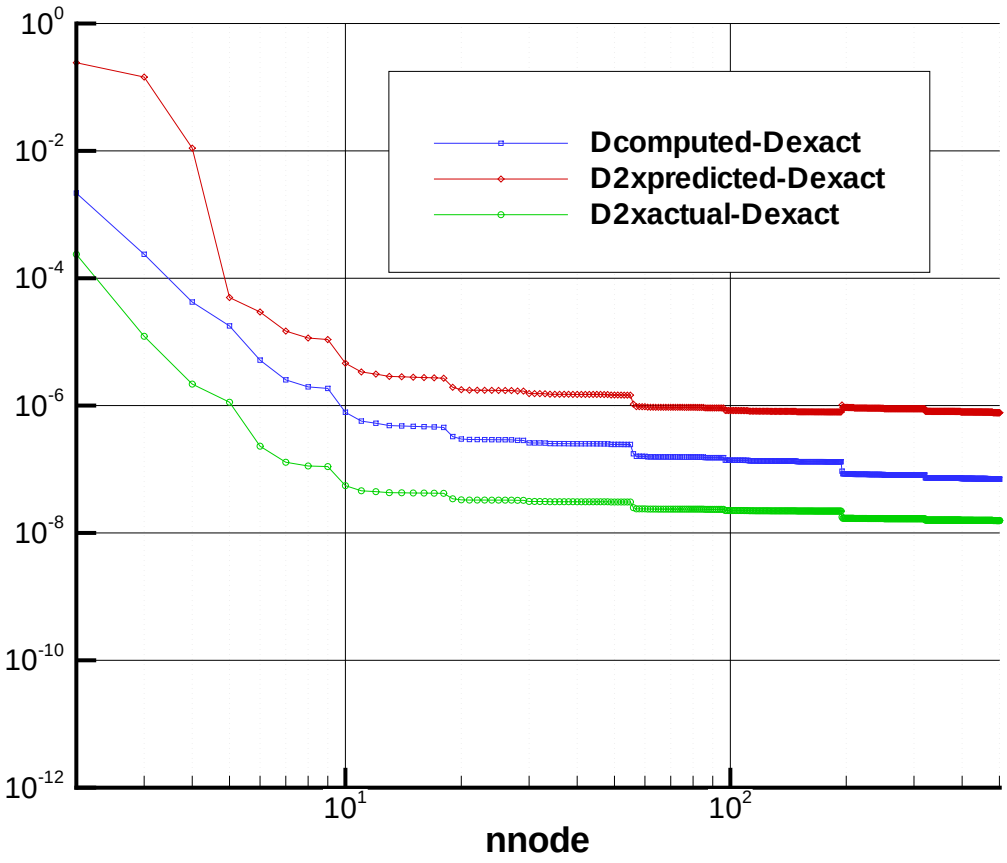


Figure 13: Computed, and Uniform Refinement Predicted and Actual Errors in D Using Truncation Based Refinement.

It is possible that refining cells by placing new nodes at other than the cell centers could lead to improved accuracy, but there is not enough information available to do that unless one uses even higher order reconstruction, which is more problematic. There are some interesting issues in generalizing Equation (3.8) to higher dimensions. For a node centered scheme, a uniform refinement of triangles in two dimensions or tetrahedrons in three involves placing new nodes at all edge midpoints. A metric analogous to Equation (3.8) can determine which cells to refine. At this point some methods use a Hessian of a scalar flow quantity metric to determine how to refine cells or redistribute nodes. Our AHA code does this, whereas our HIREF code relies on the adjoint metric alone, which is a little more consistent ([1]). The latter tries every possible binary refinement of the cell to see which refinement most reduces the error indicator in Equation (3.8). The corresponding edge midpoints are then required to be new nodes. For the purpose of adaptivity, quadrilaterals and hexahedrons or reduced hexahedrons are used, as these lead to much smaller adapted grids.

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

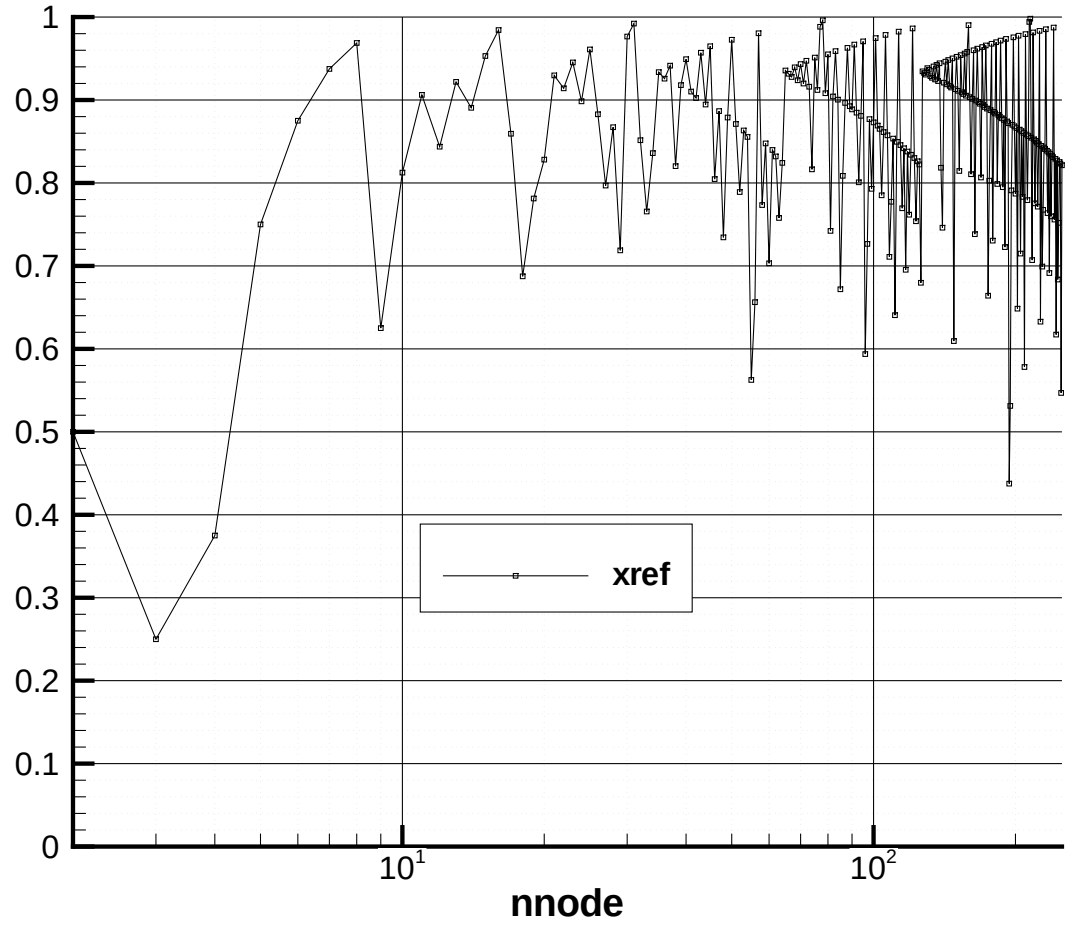


Figure 14: Coordinates of Centers of Refined Cells Using Truncation Based Refinement.

For solution, such cells are temporarily split into triangles and tetrahedrons again using the metric of Equation (3.8). Shear layers and shocks can eventually be resolved even though new nodes are only placed at edge midpoints. While not optimal the method is fairly robust.

Using only one adjoint or a single other criterion to refine grids could in theory result in grids that unacceptable from other points of view. This doesn't seem to happen for lift and drag functionals, but the authors do not have enough experience for a firm conclusion. For the pure adjoint method we often use multiple functionals, although there is a problem here. We must allow derefinement in case shocks, shear layers or other features including separation move or disappear in the course of solution process. Multiple criteria can counter each other leading to cycling. This must be resolved by letting each criteria veto derefining a specific cell. Finally, it seems advantageous to periodically revisit the original base grid and use the current solution to perform function

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

adaptation. This unwinds unnecessary and unaligned grid faster than derefinement, but is subject to cycling.

4. Bifurcations and Multiple Solutions

It is well known that hysteresis phenomena are observed in the lift characteristics of aircraft configurations. For example, these occur as flow conditions vary past stall and back through variation in angle of attack. What is not so well known is the uniqueness of the hysteresis loop as well as the degree to which computational modeling accurately captures such effects. In studies of canonical high lift systems we have observed the existence of two to three solutions at significantly different lift levels on a single grid, and many more (three to five) for the same configuration on different grids. All these solutions are residual converged, but may not be grid converged. Some solutions have been replicated using multiple codes. Some appear only with a subset of solvers. One or another solution may appear depending on either initial guess or some details of the non-linear solution process involving different strategies for limiting Newton step size or time stepping strategies. In all likelihood the non-linear systems of equations do not have unique solutions and continuation methods of one or another type may be essential in mapping out families of solutions. Such methods put an additional requisite for any adaptive scheme, i.e., the mechanics of resolution improvement or the error indicator driving the adaptive process must be controlled to ensure that solutions on increasingly fine grids remain on the same solution branch. Finally, current solution processes do not provide any way of knowing about the possible existence of additional solutions, even when they may be of significant relevance to the problem being studied. These issues, and the extent to which they are resolved, may ultimately be the determining factor in terms of how far into the more extreme flow conditions one can reliably use steady RANS modeling.

5. Conclusions

Despite considerable research, the routine application of CFD to the full flight envelope remains a distant target. Difficult algorithmic problems remain, although a number of promising developments over the last few years give rise to some optimism. It seems to us that a great deal of research is still needed regarding residual and grid convergence. The former is certainly attainable as evidenced by the fact that the authors have finally made a decision not to deliver a solution without it. We cannot say the same for the latter. In three dimensions robust adaptive grid refinement is still in the future. Clearly, local refinement based on interpolation or truncation error is inadequate and adjoint methods in one form or another must undoubtedly be part of the solution. For such methods to work properly accurate reconstruction is key. However, very little research is being devoted to this topic. Residual and grid convergence are necessary before other issues such as turbulence and physical modeling applicability can be reliably investigated. In one form or another, the situation is characteristic of the early days of cruise analysis and design CFD codes. The inadequacies of these codes were ultimately overcome by the development of robust discretizations with improved accuracy, powerful numerical solvers, adaptive grid refinement algorithms, non-equilibrium turbulence

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

models, commodity computing and years of user experience.

References

[1] S. R. Allmaras, J. E. Bussoletti, C. L. Hilmes, F. T. Johnson, R. G. Melvin, E. N. Tinoco, V. Venkatakrishnan, L. B. Wigton, D. P. Young. *Algorithm issues and challenges associated with the development of robust CFD codes*. Giuseppe Buttazzo, Aldo Frediani, Variational Analysis and Aerospace Engineering. New York, Springer, 33 (2009), 1–19.

[2] M. B. Bieterman, J. E. Bussoletti, C. L. Hilmes, F. T. Johnson, R. G. Melvin, D. P. Young. *An adaptive grid method for analysis of 3D aircraft configurations*. Computer Methods in Applied Mechanics and Engineering, 101 (1992), 225–249.

[3] L. Demkowicz. *Computing with hp-adaptive finite elements, Vol. 1: One and two dimensional elliptic and Maxwell problems*. Chapman and Hall/CRC Applied Mathematics, 2006.

[4] B. Diskin, J. L. Thomas. *Accuracy of gradient reconstruction on grids with high aspect ratio*. NIA Report No.2008-12, December, 2008.

[5] T. J. R. Hughes, A. Brooks. *A multi-dimensional upwind scheme with no crosswind diffusion*. Finite Element Methods for Convection-Dominated Flows (ed. T.J.R. Hughes) AMD 34, New York, ASME (1979), 19–35.

[6] F. T. Johnson, E. N. Tinoco, J. N. Yu. *Thirty years of development and application of CFD at Boeing Commercial Airplanes, Seattle*. Computers & Fluids, 34 (2005), 1115–1151.

[7] D. J. Mavriplis. *Revisiting the least-squares procedure for gradient reconstruction on unstructured meshes*. AIAA Paper 2003–3986.

[8] T. A. Oliver. *A high order, adaptive, discontinuous Galerkin finite element method for the Reynolds-averaged Navier-Stokes equations*. Ph. D. Thesis, M.I.T., (2008).

[9] N. B. Petrovskaya. *Discontinuous weighted least squares approximation on irregular grids*. CMES: Computer Modeling in Engineering & Sciences, 32 (2008), No. 2, 69–84 .

[10] N. A Pierce, M. B. Giles. *Adjoint and defect error bounding and correction for functional estimates*. J. Comp Phys., 200 (2004), 769–794.

[11] P. R. Spalart, S. R. Allmaras. *A One-equation turbulence model for aerodynamic flows*. La Recherche Ae'rospatiale, 1 (1994), 5–21 . Also AIAA paper 92-0439.

[12] J. C. Vassberg, E. N. Tinoco, M. Mani, B. Rider, T. Zickhur, D. W. Levy, O. P. Brodersen, B. Eisfeld, S. Crippa, R. A. Wahls, J. H. Morrison, D. J. Mavriplis, M. Murayama. *Summary of the fourth AIAA CFD Drag Prediction Workshop*. 28th AIAA Applied Aerodynamics Conference, 28 June – 1 July, 2010, Chicago, IAIAA Paper 2010-4547.

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press

[13] D. A. Venditti, D. L. Darmofal. *Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows*. J. Comp Phys., 187 (2003), 22–46.

[14] V. Venkatakrishnan, S. R. Allmaras, F. T. Johnson, D. S. Kamenetskii. *Higher order schemes for the compressible Navier-Stokes equations*. 16th AIAA Computational Fluid Dynamics Conference. Orlando, Florida, June 23-26, 2003, AIAA Paper 2003-3987.

[15] D. P. Young, R. G. Melvin, M. B. Bieterman, F. T. Johnson, S. S. Samant, J.E. Bussoletti. *A locally refined rectangular grid finite element method: application to computational fluid dynamics and computational physics*. J. Comp Phys., 92 (1991), 1–66.

https://doi.org/10.1051/mmnp/20116301 Published online by Cambridge University Press