Probabilistic Error Modeling in Computational Fluid Dynamics

Robert E. Childs and Patrick H. Reisenthel
Nielsen Engineering & Research, Inc.
605 Ellis Street, Suite 200
Mountain View, CA 94043-2241, USA

rechilds123@yahoo.com / phr@nearinc.com

ABSTRACT

The work reported here uses error modeling, both deterministic and probabilistic, as a means of quantifying the inaccuracy in computational fluid dynamics (CFD) results. The two fundamental elements of error modeling are the quantification of error sources and the propagation of error through a solution. The present research focuses on the error sources and treats error propagation either in a deterministic manner or by using a Monte Carlo approach. A nonintrusive form of error modeling based on the concept of "defect correction" is used. The truncation errors, or other error sources, are computed and added to the residual. The equations modified by error sources are then solved, and the error in the solution is the difference between the truncation-error-forced solution and the normal one. The present methods have been applied to four CFD solvers that use structured and unstructured grids, and address examples of truncation error and turbulence modeling error. Both deterministic error correction and probabilistic error modeling are discussed.

1.0 LIST OF SYMBOLS AND ABBREVIATIONS

C_f	skin friction coefficient
C_D	drag coefficient
$C_{D,p}$	pressure drag coefficient
$C_{D,v}$	viscous drag coefficient
C_L	lift coefficient
C_M	pitching moment coefficient
C_p	pressure coefficient
CFD	computational fluid dynamics
DPW	drag prediction workshop
M	Mach number
MC	Monte Carlo
N	number of grid points
n_{perts}	number of perturbations (realizations)
p	probability
PC	polynomial chaos
PDF	probability density function

Childs, R.E.; Reisenthel, P.H. (2007) Probabilistic Error Modeling in Computational Fluid Dynamics. In *Computational Uncertainty in Military Vehicle Design* (pp. 1-1 – 1-22). Meeting Proceedings RTO-MP-AVT-147, Paper 1. Neuilly-sur-Seine, France: RTO. Available from: http://www.rto.nato.int.

P_w	wall pressure
q	exact solution vector
Re	Reynolds number
RMS	root mean square
RN	random number
RSS	root sum of squares
s	entropy
SQA	solution quality assessment
stdev	standard deviation
SVD	singular value decomposition
TE	truncation error
u	CFD solution vector
α	angle of attack
δ_o	boundary layer thickness
σ	standard deviation
ξ	computational streamwise coordinate
η	computational normal coordinate
<.>	average

2.0 INTRODUCTION

The development of simulation-based methods in engineering and science has been relentless. However, there are some problematic aspects of simulation that hinder this progress and must be addressed. Two fundamental challenges in the ideal world are phenomena that have significant influence across very large ranges of scales and those that involve physics that must be modeled, often empirically. A fundamental challenge of the real world is that almost no physical items are exactly as idealized; there are differences between the definition of a simulation and the physical world it nominally models.

Probabilistic simulation methods may be useful in addressing these issues. The challenges cited above are responsible for errors that, while not irreducible, have proven to be difficult and/or expensive to reduce, with no real hope of breakthroughs using existing technologies. If those errors can somehow be bounded and their magnitude quantified, then it is possible to predict the possible range of solutions that may exist, despite the inability to predict the exact solution. In effect, probabilistic error modeling hopes to exchange the impossible task (at least in general turbulent flows) of producing perfect simulation results for the mere Herculean task of producing results that are close enough to the true behavior with a high level of confidence.

Deterministic and probabilistic error modeling represent two levels of estimating the inaccuracy in a simulation. Deterministic error modeling gives the sign and magnitude of predicted error in a solution. Probabilistic error modeling makes some prediction of a confidence interval or perhaps a probability distribution for the error. The error is a single number, and so an error prediction is in some sense as difficult as the original simulation. A probabilistic calculation might substitute a probability distribution function (PDF) for a single variable in the original simulation, which adds a new dimension to the problem, and which potentially raises the difficulty by orders of magnitude. The computational cost of doing probabilistic simulation depends on the technology used and the level of accuracy needed. The Monte Carlo method has been used, despite its high cost, to explore the basic nature of probabilistic error modeling. Despite the potentially greater computational difficulty (cost) of probabilistic methods, they are believed to be essential because they are more reliable than deterministic methods. Deterministic methods have an Achilles heel: the

potential for cancellation between errors of opposite sign, leading to unrealistically low predictions of error. Probabilistic methods can be constructed to minimize the possibility of error cancellation, leading to robust predictions of probable errors in a simulation. Because of these differences, it is important at this time to pursue deterministic and probabilistic methods, as both make important contributions toward improved simulation methods.

The present effort addresses probabilistic methods in the context of error modeling. The development of probabilistic simulation methods is in its infancy, and these methods will not be well validated for many years. Relying on such methods for core simulation capability in critical situations would be risky; the use of these methods in an auxiliary role of error prediction is more likely to be accepted by the engineering community. Furthermore, error modeling via defect correction is readily implemented in existing simulation methods, once the error sources are known. Key processes in error modeling are the ability to compute the sources of error and the validation of this calculation. Examples of these steps are given in the context of deterministic error modeling.

3.0 ERROR EQUATION

Inaccuracy modeling involves two fundamental processes: its generation by sources of inaccuracy, and the propagation of inaccuracy in the solution via convection and diffusion. This can be illustrated in a derivation of the defect-correction form of the error equations. The analytic Navier-Stokes equations in 1D can be written as

$$\partial_t q + \partial_x (F(q)) = 0 \tag{1}$$

in which q is the solution and $F(q)$ includes both the inviscid and viscous fluxes. The discrete form of the Navier-Stokes (CFD) equations are added to both sides of the equation, yielding

$$\delta_t^{(1)} q + \delta_x^{(2)} F(q) = (\delta_t^{(1)} q - \partial_t q) + (\delta_x^{(2)} F(q) - \partial_x F(q)) \tag{2}$$

in which $\delta_t^{(1)}$ and $\delta_x^{(2)}$ are the discrete first-order temporal and second-order spatial derivatives, typical of CFD methods. (CFD methods of different order can, of course, be used.) The CFD equations are on the left and the truncation errors are on the right-hand side of Eq. (2). The exact truncation errors are given by

$$TE_{exact}(q) = (\delta_t^{(1)} q - \partial_t q) + (\delta_x^{(2)} F(q) - \partial_x F(q)) \tag{3}$$

An estimate of the truncation error can be obtained with

$$TE_{est}(q) = (\delta_t^{(1)} q - \delta_t^{(M)} q) + (\delta_x^{(2)} F(q) - \delta_x^{(N)} F(q)) \tag{4}$$

in which M and N are as high an order as possible, but usually just one order higher than the native CFD solver. The use of higher order discretization terms in Eq. (4) highlights a dilemma, namely, why not just use the high-order method in the native CFD solver, rather than compute error estimates? First, true higher order methods have proven difficult to develop and/or costly to use for structured and unstructured methods applied to complex configurations. Second, error estimation is still needed, even for higher order methods.

Let us split the exact solution q into the CFD solution u and the error ε therein,

$$q = u + \varepsilon \ . \tag{5}$$

The defect correction form of the CFD equations has a truncation-error forcing term

$$\delta_t^{(1)}(u+\varepsilon) \; + \; \delta_x^{(2)}F(u+\varepsilon) \; = \; TE_{est}(u+\varepsilon) \tag{6}$$

Finally, assume that the truncation error can be computed from the CFD solution, ignoring the contribution from the, presumably small, error in the solution

$$\delta_t^{(1)}(u+\varepsilon) \; + \; \delta_x^{(2)}F(u+\varepsilon) \; = \; TE_{est}(u) \tag{7}$$

Equation (7) is the defect correction form of the CFD error equations used for error modeling. During the normal CFD process, the standard CFD equations, here taken to be

$$\delta_t^{(1)}(u) \; + \; \delta_x^{(2)}F(u) \; = \; 0 \tag{8}$$

are solved for u, and the error ε is obtained by difference.

Similarities between the original governing equations and the defect correction form used for error modeling indicate that error will have some quasi-physical behavior. Error modeling involves the solution of equations similar to the governing equations, but for a forcing term due to the error source. Inaccuracies in a solution behave much like disturbances in physical flow; they convect and diffuse, and they may amplify or decay depending on flow conditions. It may be helpful to think of error as pollution (e.g., in the accuracy of a solution) that originates from distributed sources and convects through the domain. Error in the solution obeys conservation equations, and it is conserved with respect to transport.

We state without demonstration that errors originating from other sources can be modeled in the context of Eqs. (7) and (8). An obvious extension is the error generated by the use of lower order discretization in a boundary condition. Errors in the modeling of chemical reactions or turbulence can, conceptually, be treated this way. The key issue is that error sources exist where the governing equations are inaccurate, and the errors created in the solution then propagate, like pollution, according to the governing equations.

4.0 ERROR MODELING STRATEGY

The approach described here uses a CFD solver as a “black box,” meaning that it works with no knowledge of the solver's inner workings. The basic strategy used for steady-state results presented here involves 5 steps: (1) iterate the normal CFD solution to convergence and save it; (2) compute and store the truncation error in the normal CFD solution; (3) add the truncation error from the normal solution (which is frozen) to the residual during time integration; (4) iterate the defect correction solution to convergence; (5) compute the error as the difference between normal and defect-corrected solutions. All of these steps except the calculation of the truncation error can be implemented very easily.

The only substantive modification to this process is the option to use a scaling factor to reduce the magnitude of the truncation error used for defect correction; this operation must then be followed by a reciprocal scaling to recover the full error. In the limiting case of an infinitesimally small scaling factor, this has an effect similar to using a truly linearized error equation. This approach could probably be used in all cases, but we have used it only when needed to remedy a divergent defect correction calculation (see Section 10).

5.0 CALCULATION AND VALIDATION OF THE TRUNCATION ERROR

The complement to the simplicity of the error modeling equations is the difficulty of computing the truncation error, and the error sources in general. The truncation error can be computed with good accuracy only when good higher order discretization methods exist. Except for some finite element methods, such as discontinuous Galerkin methods, few 'higher' order methods truly are, when applied on the curvilinear grids

applied to complex geometries. The difficulty of computing the truncation errors accurately is exacerbated at shocks, grid metric discontinuities, and at boundaries; high-aspect ratio cells in unstructured grid methods can generate large truncation errors that are not well-quantified. If truncation errors could be computed accurately, typical CFD methods would be of higher order.

The methods used to construct a higher-order discrete approximation for computing the truncation error depend significantly on the native flow solver, so few general rules apply. Three important issues should be noted: (1) For the higher-order discretization, it is acceptable and perhaps preferable to use a method that minimizes or omits the algorithmic dissipation, which can be the biggest source of truncation error; (2) Due to their errors in terms of wave number, the Euler terms are generally much more important than the viscous terms; (3) Boundaries are very important in general.

We have implemented error modeling for four different CFD solvers, representing three different spatial discretization schemes. The work on these codes is briefly summarized below.

NASA's Overflow and ARC2D both use finite difference methods on structured curvilinear grids. Work with ARC2D was used as a proof-of-concept, and was superseded by work with Overflow. The higher order discretization was implemented using 4-point central interpolation and gradients onto 'cell faces,' followed by differencing of the cell-face fluxes. Algorithmic dissipation was not added to the higher order flux terms. Boundary terms were treated with second-order one-sided formulae. Truncation errors in the Spalart-Allmaras turbulence model were computed in a similar manner, and the gradient terms in the source terms (vorticity) can also be computed with fourth-order central methods. The integrated forces and moments were also computed to higher order. Second-order surface-normal gradients were used, although this is not necessarily a wise step. Higher order grid metrics have been used, but they were not found to be beneficial for 'typical' grids.

FUN3D is a node-based unstructured grid finite-volume method developed by NASA. Its default order of spatial accuracy is second order, obtained using a linear least-squares reconstruction and, commonly, the Roe inviscid flux algorithm. Viscous fluxes are typically computed with a Gauss-Green method, but finite differencing is also available. The 'higher order' discretization was developed using quadratic least-squares reconstruction. Both linear and quadratic methods fit a surface to the cell-averaged solution values cast onto the grid nodes. This approach is order consistent for the second-order linear reconstruction, but it is order deficient for quadratic reconstruction aimed at yielding a third-order method. Thus, true third-order accuracy is precluded. This approximate quadratic fit typically requires a reconstruction stencil that extends beyond nearest neighbor cells. The quality of the reconstruction fit at each grid node was assessed indirectly: a singular-value decomposition (SVD) method was used to solve the least-squares problem, and it signaled rank deficiency. Deficient stencils were not used. They were either expanded until the stencil was sufficient, or the quadratic fit was degraded to linear. The Roe algorithm of the native discretization scheme was not used. Instead simple averaging was applied to the solution or the fluxes, and both methods behaved similarly for the cases studied to date. The same quadratic reconstruction was used for inviscid and viscous fluxes by estimating the value and gradient of the solution.

AVUS (formerly known as Cobalt60) is a cell-based unstructured solver developed by the US Air Force at Wright-Patterson AFB. The grid connectivity is different than in FUN3D which requires a different algorithm for constructing the extended stencil. Other features, such as the use of the SVD algorithm and omission of the Roe fluxes are common with the FUN3D code. No AVUS results are presented herein.

Once the higher order discretization terms are developed, the spatial truncation error is trivial to compute: it is the difference between the normal-order and higher order residuals, as indicated in Eq. (4). We have not yet examined the effects of temporal truncation errors in error modeling, in large part because time-step refinement is easy.

Two methods of validating the truncation error are mentioned here. The defect correction form of the error equation is so simple it can be implemented in a CFD solver with a very low probability of programming errors. A properly computed truncation error will cause defect correction solutions to converge with respect to grid refinement toward the same grid-asymptotic solution as the default CFD methods, and converge more rapidly than the default CFD method. A related and useful test is to compare normal and defect-corrected solutions against solutions with exact properties, like inviscid flows with zero drag and entropy production.

An example of validation of the truncation error by comparison to a flow with exact properties is given in **Figure 1**. Inviscid subsonic flow generates no entropy, so entropy is a measure of error. Calculations of flow past an airfoil were performed with FUN3D, with and without defect correction. A small detail near the leading edge is shown with an overlay of entropy contours from the native solution (black contour lines, white labels) and the defect-corrected solution (white contours, colored labels), both computed on the same grid (purple). The native FUN3D solution has entropy, to the levels plotted, in five layers of cells near the body. Defect correction eliminates the entropy in the outer 4 of these layers, leaving entropy in the body-adjacent cells only. The remaining entropy in the near-boundary cells is probably due to the boundary treatment, which was not at the same level of accuracy as the interior cells. The drag coefficient should theoretically be zero in this flow. The native FUN3D solution yields $C_D = 0.0011$, which drops 62% to 0.00041 with defect correction.

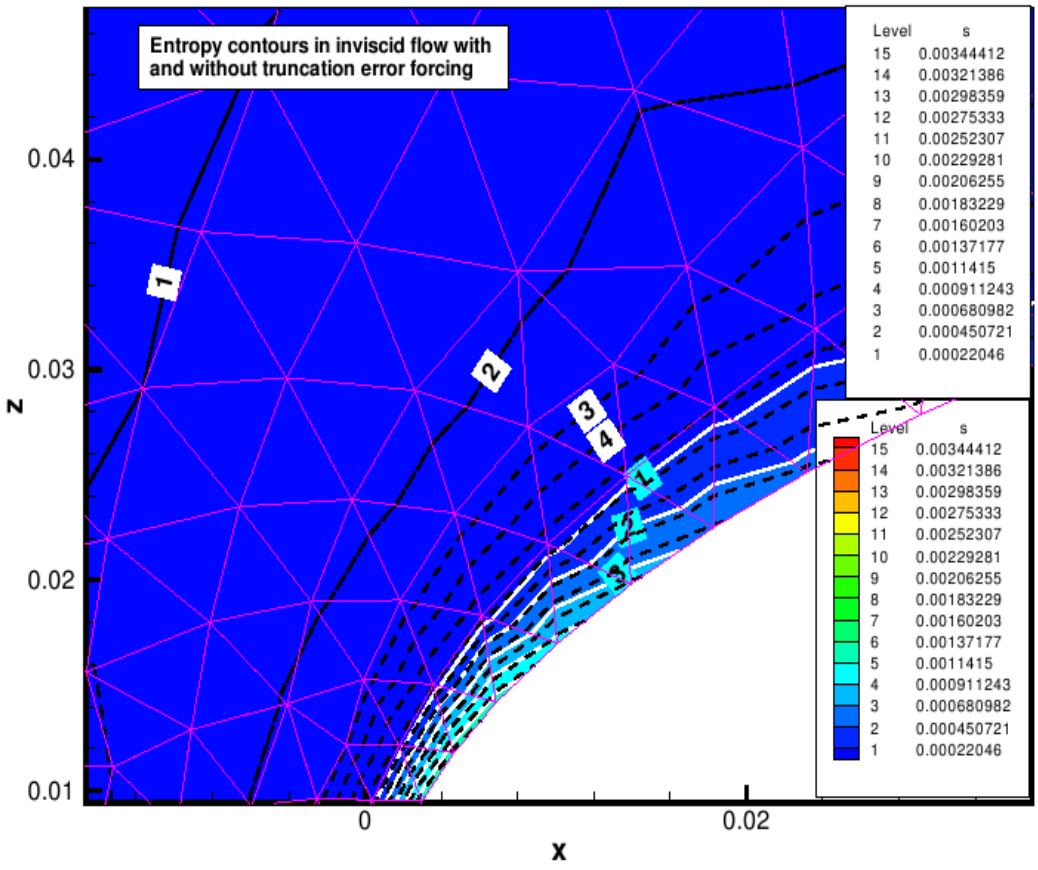

Figure 1: Entropy Contours With and Without Defect Correction.

When validating the truncation error, it is essential to look at many aspects of the solution, and especially not just the integrated forces and moments. A fundamental problem in error modeling is cancellation between errors of opposite sign. This affects validation of the truncation error and is a persistent problem in the use of error modeling. It is appropriate, therefore, to discuss error cancellation as an integral part of truncation error validation.

6.0 ERROR CANCELLATION

The truncation error has sign and magnitude, for each equation, at each grid cell. It is also made up of a sum of parts. The example to be given below involves viscous and inviscid terms of the truncation errors, but individual components like the inviscid terms in, for example, the ξ direction are distinct from those in the η direction of a structured-grid solution. In a solution with many grid points and multiple truncation error terms, some cancellation between truncation error terms of opposite sign is inevitable. Also, truncation errors from different grid cells may cancel as the error propagates through the solution. This occurs at captured shocks, by design. Where errors cancel, the solution is better than should have been expected. Unfortunately, this 'good luck' can disappear in a similar solution, leading to inexplicable error in the CFD results. In error modeling, error cancellation can lead to results that are otherwise inexplicably good or bad. Two examples of error modeling are given below in which error cancellation is significant.

Figure 2 gives the integrated drag on a circular cylinder in low speed laminar flow computed using FUN3D both with and without defect correction. The net drag versus $1/N^{2/3}$ is given, where N is the number of grid points. These are fully 3D grids on a short spanwise section of the cylinder, and grid refinement is approximately isotropic. Grid convergence should be roughly linear in this coordinate system, for this second-order method. The solid lines and solid symbols illustrate this convergence for the standard FUN3D solutions. The convergence is observed to be nearly linear at the two Reynolds numbers Re = 10 and 1. The dashed lines and open symbols are the result of defect correction and are labeled “SQA iv”; the “iv” denotes that inviscid and viscous truncation errors are included, and “SQA” is our name for the process. At Re = 10, the total drag from the SQA iv results is astonishing good, even on the coarsest grid. Conversely, the SQA iv results at Re = 1 are abysmal: they give essentially no improvement over the standard prediction (roughly zero predicted error), and the predicted error *increases* with grid refinement. Further examination shows, however, that at Re = 10, the simple decomposition of the total drag into its viscous and pressure components reveals cancellation between nearly equal and opposite errors in those components. Also, note the line labeled “SQA i” and “SQA v,” which denote defect correction with only the inviscid or viscous components of the truncation error. At Re = 10, the viscous truncation errors are negligible relative to the inviscid ones. (The error bars are discussed later.)

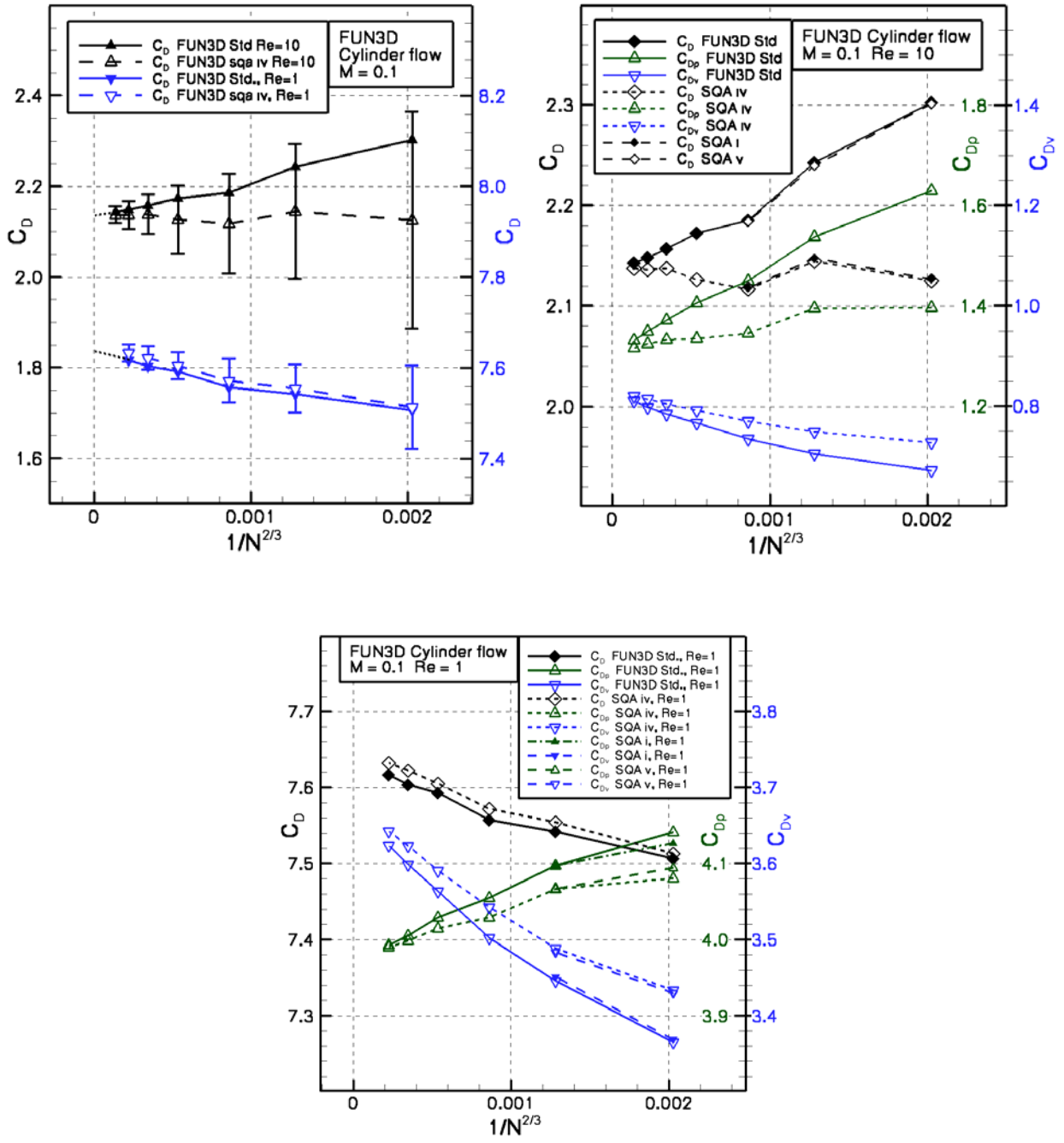

Figure 2: FUN3D Calculation of Flow Past Cylinder at Low Reynolds Number.

In the $Re = 1$ results, the predicted error in the total drag is almost zero on the coarsest grid, and increases slightly with increasing number of grid points ($1/N^{2/3} \to 0$), giving the appearance that error modeling fails. However, defect correction predicts that the viscous and pressure components of the drag have nonzero errors

of opposite sign; as a consequence, the *net* drag is predicted to have essentially zero error on the coarsest grid. The predicted error in the pressure drag converges to zero with grid refinement. On the coarsest grids, the results labeled "SQA i" and "SQA v" demonstrated that the viscous contribution to the drag is dominated by the viscous truncation errors, so for similar flows at Reynolds numbers below 10, the viscous truncation errors are likely to be significant. On the coarsest grid, the inviscid truncation error's contribution to the pressure drag is about 20% of the contribution of the viscous truncation errors. The pressure drag is corrected downward, while the viscous drag is corrected upward by an approximately equal amount. The predicted error in the viscous drag decreases slowly enough that it does not extrapolate to zero at zero grid spacing. We strongly suspect that this is due to the first-order methods used to evaluate the wall shear stress when the viscous drag was integrated, for which the quadratic reconstruction was not implemented. (Although not displayed here, field quantities in these solutions have been examined, and they display convergence behavior consistent with the results for the pressure drag coefficient.)

Another form of error cancellation occurs during spatial integration. For example, similar positive errors in the pressure on the front and back of the cylinder may cancel, in terms of the net drag force they produce. Truncation errors of similar magnitude and opposite sign typically occur on opposite sides of a captured shock; this is how the solution downstream of the shock can be correct, even though the solution inside the shock is not accurate.

Error cancellation is believed to be very common in typical CFD results and therefore in deterministic error modeling of these calculations. The results shown here demonstrate two types of error cancellation: between components of the error sources (truncation error) and between aspects of the solution (e.g., components of the drag). Of all CFD calculations explored by the authors in sufficient detail to assess individual contributions to the error in the solution, the majority have been found to display significant error cancellation. If there is proven consistent error cancellation, as typically occurs at captured shocks, this error cancellation may be relied upon as part of normal CFD practice. However, even at shocks, this cancellation may fail and produce, for example, the carbuncle phenomenon.

7.0 PROBABILISTIC INTERPRETATION OF DETERMINISTIC ERRORS

Options exist for estimating "error bars" from the results of deterministic error calculations. Any method that eliminates some error cancellation can be used to some benefit. The error bars in **Figure 2** were produced by computing a "root sum square" (RSS) of the errors in the pressure and viscous components of drag. The RSS concept is used because it is a valid means of computing the standard deviation of the sum of uncorrelated processes. The use of RSS error bars is approximate in the present application. For example, in a separating flow, there is some physical correlation between the surface velocity gradient and the surface pressure; for general flows, however, this correlation is expected to be weak. The RSS concept is correct for independent Gaussian processes, and the rule applies, strictly speaking, to standard deviations, not the magnitude of the deterministic errors as was done. The results at $Re = 1$ are especially improved by this approach. However, any process that is based on a deterministic solution of an error modeling equation will have limited reliability due to the high probability of cancellation among error source terms, which this RSS postprocessing step can never address.

8.0 PROBABILISTIC ERROR MODELING

Probabilistic error modeling emerges naturally from the acknowledgment that error source terms are approximate and uncertain. Mathematically, this is done by constructing a probabilistic description of the error source terms, for example, in terms of an uncertainty interval or as a probability distribution. In

deterministic modeling, the source term has a specific value at each grid point, for each equation in the set of CFD equations. In probabilistic modeling, every one of those specific values is replaced by a probability distribution. The equation that is the basis for the error modeling, Eq. (6), then becomes probabilistic, and its solution is a probabilistic prediction of the error.

Two important challenges in probabilistic error modeling are (1) the methods needed to estimate the probabilistic error source terms, and (2) an efficient means of solving the probabilistic equations. In this work, we avoid the second issue and use Monte Carlo, which works well, but can be very expensive to use. The probabilistic work reported herein focuses on methods of modeling probabilistic error sources, and demonstrating probabilistic error predictions. There is still much to be learned about the fundamental nature of probabilistic modeling, in addition to the very real need for efficient methods.

Probabilistic error modeling suffers from a lack of useful analytical guidance for computing the uncertainty in the error source terms. The truncation error formula, Eq. (3), provides explicit guidance that can be followed to approximate the deterministic truncation error. One approach is to use the terms neglected in computing the deterministic truncation error to compute the uncertainty in the deterministic truncation error. This analytical approach suggests that either an even-higher-order approximation or a means of estimating the truncation error from $u+\varepsilon$ are candidates for a probabilistic description of the truncation error. Both methods have been explored briefly, with mixed results that exhibit advantages and disadvantages. A different approach is empiricism: propose and test plausible concepts, and use ones that perform well. This is the method with which we achieved the best results in early results, and which has been used exclusively since that time. However, alternate methods deserve additional research.

The general approach we have used is to approximate the probabilistic error source term as the deterministic truncation error plus a zero-mean Gaussian distribution whose standard deviation scales on the magnitude of the deterministic truncation error. Thus, the assumed probability distribution of truncation error (*TE*) having a nominal value TE_0 could be described as

$$p(TE) \;=\; \frac{1}{\beta\left|TE_0\right|\sqrt{\pi}}\;\exp\left(-\left((TE-TE_0)/(\beta\,TE_0)\right)^2\right) \qquad \textbf{(9)}$$

in which TE_0 is the deterministic truncation error and β is a parameter that regulates the uncertainty interval. For example, if β = 1/3, the PDF of the *TE* is a Gaussian centered on TE_0 with a standard deviation of $TE_0/3$, so the 99.7% cut-offs on the *TE* distribution are 0 and $2\,TE_0$. This is a reasonable first guess at a probability distribution for the truncation error.

It is now important to consider how the uncertainty in truncation error is going to be used in a probabilistic calculation. A Monte Carlo simulation consists of many realizations of a deterministic process, and the statistical properties of the aggregate of all of these simulations represent the probabilistic simulation. First, we establish the notation used to define the probabilistic truncation error:

$$TE^m(i,k) \;=\; RN(i,k;m)\;TE_0(i,k) \qquad \textbf{(10)}$$

Here, *i* denotes the *i-th* grid point (for structured or unstructured grids), *k* denotes the *k-th* equation (e.g., density conservation), and *m* is the *m-th* Monte Carlo realization. Monte Carlo simulations are performed by creating a random number field *RN*, and solving Eq. (6) for some number of realizations, and then statistically processing all realizations. Each realization involves a frozen *RN* field.

All that remains to do is determine $RN(i,k;m)$ and the number of realizations needed to achieve a converged simulation. There are constraints on *RN*, some from mathematical considerations, and some from knowledge

of CFD. The statistics of the error source over all realizations must reasonably approximate the intended probability, as in Eq. (9). The random field must be smooth enough so that the spatial discretization of the flow solver can be applied to the random field with minimal truncation error. Truly random fields are very jagged, and have significant energy in frequencies that cannot be resolved by the flow solver, which is also the error solver. If such a random field were to be used (Eq. (10)) to scale an otherwise smooth truncation error field, the CFD solver's algorithmic dissipation would result in severe truncation error and loss of accuracy in the solution of the error equation. Thus we apply iterative second- and fourth-order smoothing (see below) to a field that is initially generated by a classical random number generator. Comments about the actual shape of the probability distribution and effects of that shape are given in the following section. Unless otherwise noted, the random field applied to the density equation has been applied to all equations.

9.0 PROBABILITY DISTRIBUTIONS

A focus of the present work has been on understanding the details of the uncertainty modeling itself, i.e., the role played by the distribution of the random number fields used in the SQA method. As mentioned earlier, errors in the modeling of chemical reactions or turbulence can conceptually be treated in the context of Eqs. (7) and (8). In the case of turbulence modeling for example, in the hierarchy of decisions to be made in a RANS-based SQA simulation, the choice of turbulence model is, in a sense, the leading-order approximation. The next order lies in the identification of which specific terms in the model are contributing sources of uncertainty (for example, rotation-curvature and compressibility correction terms in the Spalart-Allmaras (SA) model). The next level in the SQA modeling must consider the statistics of these terms, i.e., specify the probability distribution functions (PDF) for the random input field *RN*).

To understand the role played by the input PDF, consider the following three probability distributions. The first is a “flat” probability distribution, i.e., equally probable random numbers between specified minimum and maximum values. The second PDF is the result of second- and fourth-order spatial smoothing of this flat distribution, rescaled on output to maintain specified bounds (i.e., the method mentioned above). The rationale behind the smoothing is to produce random number fields which are consistent with the spectral bandwidth of the CFD spatial discretization scheme being used, so that there is no implicit or uncontrolled filtering by the algorithm itself. Due to the wavy nature of the resulting fields, the PDFs associated with this scheme have been found to exhibit a somewhat bimodal character and will be referred to, therefore, as the “bimodal” distribution. The third PDF is a Gaussian distribution, synthesized from a specific sequencing of the bimodal distributions at various amplitudes (the “widths” and amplitudes of the underlying bimodal distributions having been solved independently by an optimization procedure). In this scheme, the random numbers are spatially smoothed in the exact same manner as with the bimodal distribution, except that the amplitudes are varied from realization to realization in such a way that their aggregate results in a near-Gaussian probability distribution. The three input PDFs are shown in the left-hand graph of **Figure 3** in normal form (unit area curves). These PDFs were obtained by collecting the input statistics of 1,700 realizations (also referred to as SQA perturbations) at 146 spatial points, for a total of 248,200 random samples divided into 101 equal-sized *RN* bins between plus and minus four standard deviations.

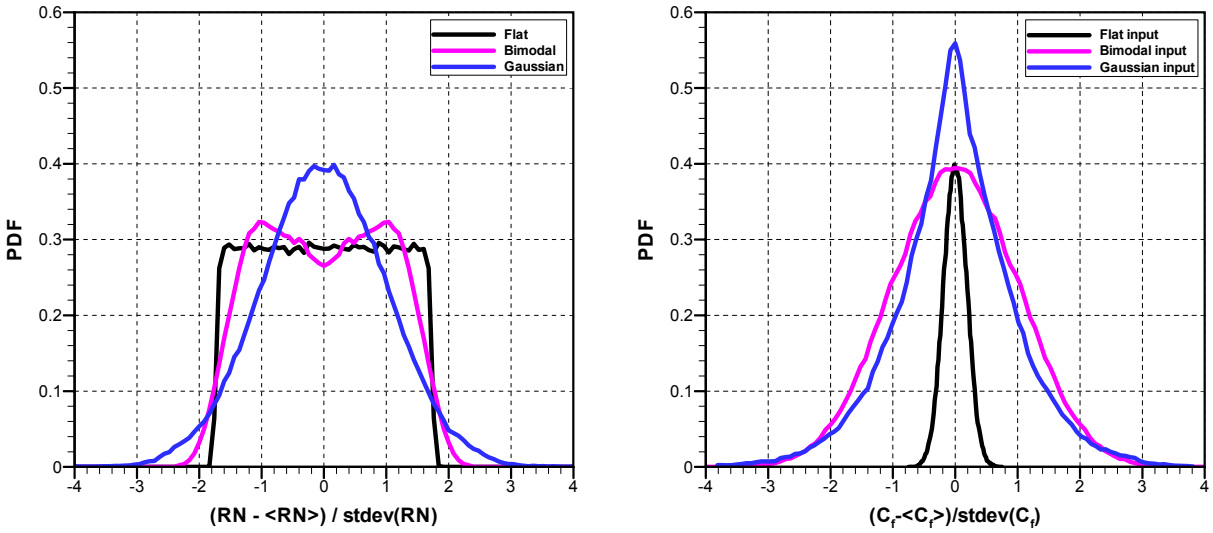

Figure 3: Relationship Between Input (Random Number) PDF Model and Output (C_f) PDF for Three Different Uncertainty Models.

The effect of these PDFs on the output distributions can be seen in the right-hand graph of **Figure 3**. The output statistics shown in this figure were collected in the corner region of a Mach 2.85 16° compression ramp, where 248,200 samples of skin friction are similarly binned, corresponding to 1,700 SQA perturbations in the spatial range corresponding to ±4 boundary layer thicknesses from the corner. Note that, in order to emphasize the relative magnitude of the outputs, the independent variables in the output PDF plots are scaled by the standard deviation of the bimodal case, rather than the individual standard deviations. Because their standard deviations are almost equal, the Gaussian and bimodal output PDFs are essentially shown in normal form. It is interesting to note that the bimodal input distribution generates a Gaussian-like output distribution, while the Gaussian input distribution generates a more pointy and longer-tailed output distribution, quantified by an excess kurtosis of 1.42 (as opposed to -0.11 for the output of the bimodal case). The PDF corresponding to the flat input distribution is also Gaussian-like, with an excess kurtosis of -0.06, but its standard deviation is more than 5 times smaller than either the Gaussian or bimodal input case, as shown by the narrow black curve in the figure. This greater than 80% reduction in the RMS is a measure of the implicit filtering (wavenumber cutoff) caused as a result of the spatial discretization algorithm in the CFD code, and provides a vivid illustration of the need for random number smoothing. Most importantly, the above results demonstrate that the tailoring of the input PDF in a manner consistent with the spectral characteristics of the CFD code can be achieved by means of amplitude sequencing between SQA iterations.

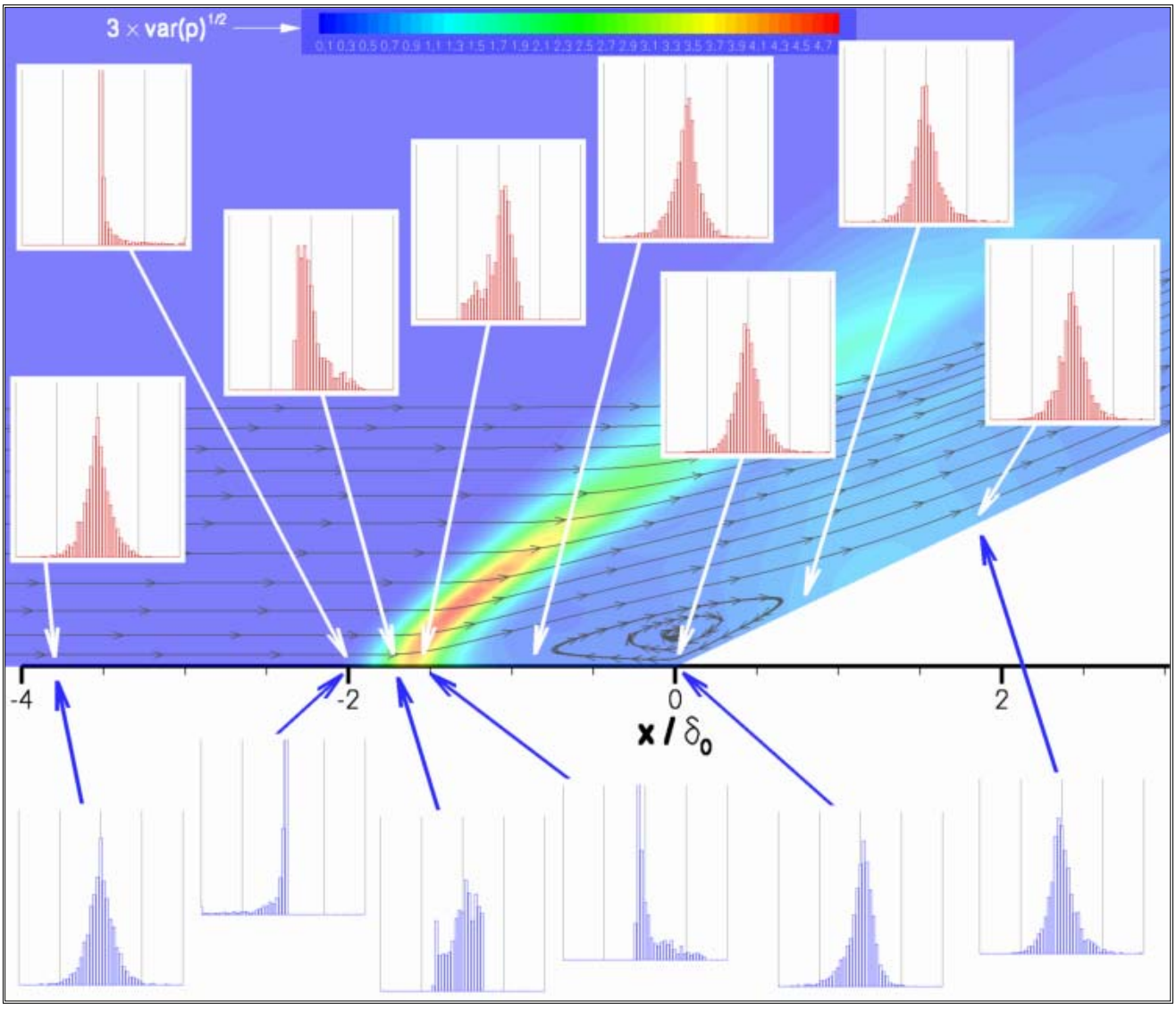

Figure 4: Streamwise Variation of SQA Probability Density Function for Wall Pressure (Top Inserts, Red) and Skin Friction (Bottom Inserts, Blue) in the Separation Bubble Region of a 24-Degree Compression Ramp at Mach 2.85.

When the compression ramp angle increases to 24° and/or the amplitude of the forcing random field is increased, the output PDFs were typically found to exhibit greater asymmetry. The asymmetry is particularly pronounced in the highly nonlinear region associated with the separation shock. This can be seen in the composite picture of **Figure 4** where local PDFs of pressure and skin friction are shown at various wall locations for the case of the 24° compression ramp, using a Gaussian input distribution. The top PDFs (red inserts) correspond to wall pressure. The bottom PDFs (blue inserts) correspond to skin friction. Each PDF is plotted in normal form, and is based on 2,500 samples (2,500 SQA perturbations) collected into 51 bins between plus and minus six standard deviations.

The background in the picture shows the ramp geometry, along with flooded contours for the uncertainty in pressure, defined as three times the pressure standard deviation. The uncertainty in the pressure is helpful in

visualizing the location of the separation shock, as well as the average reattachment, both of which are seen to correlate well with the separation bubble indicated by the mean streamlines.

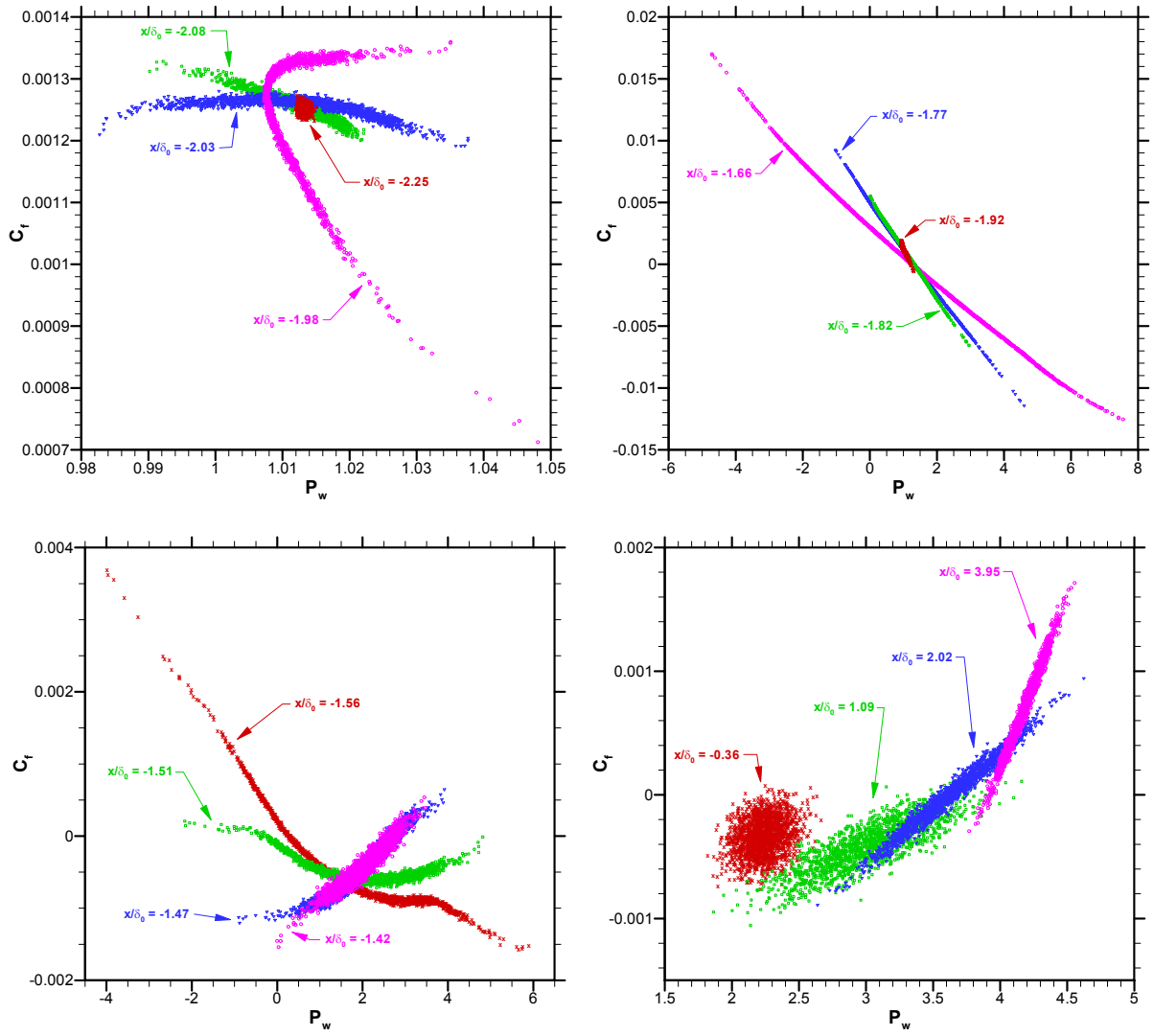

Figure 5: Phase Plots of Skin Friction vs. Wall Pressure at 16 Streamwise Locations in the Vicinity of the Separation Bubble, Shown for 1700 SQA Random Perturbations.

Further evidence of the correlation between skin friction and wall pressure can be seen in the phase plots of **Figure 5**. Much of this correlation is due to the variations in the lambda shock from perturbation to perturbation. The strong correlation between C_f and P_w is virtually assured because each Monte Carlo realization is an attempt at obtaining the "correct" solution by means of a CFD calculation which implements the normal laws of physics. It is easy to conjecture that this and other nonlinear flows will generate highly nonnormal probabilistic error distributions as a result. This may have significant implications for methods that seek to achieve full probabilistic error calculations which are based on Gaussian expansions.

10.0 IMPORTANT DETAILS

There are many details of the error modeling process that have proven to be important. The most relevant ones are listed here.

Truncation Error at Shocks: At shocks, the notion of a convergent discrete approximation series loses validity, and a higher-order approximation may not be a better approximation of physics compared to the normal CFD methods. The truncation error can be computed as described above, and it commonly causes the defect-corrected solution to suffer from Gibbs oscillations. At strong shocks this may result in nonphysical behavior and divergence. An option is to discard the truncation error at shocks. In a very real sense, we concede insufficient knowledge of this topic, and take a path that avoids this issue for now. A sensor flags grid points where the solution is decelerating through a sonic plane, and the truncation error at these and adjacent grid points is set to zero. Results have also suggested that truncation errors are inaccurate at grid metric discontinuities. We have not routinely set these truncation errors to zero in calculations, but this should be considered unless an alternate remedy can be found.

Monte Carlo Convergence Acceleration: Two strategies are used to accelerate the convergence of the Monte Carlo process. The truncation error has a deterministic part (nonzero mean), and a probabilistic part that has a zero mean, using our assumption of a Gaussian distribution. To avoid repeatedly solving the deterministic part of the Monte Carlo simulation, we first solve the deterministic defect-correction problem, and use its solution as the starting point for all of the Monte Carlo realizations. The second step is to ensure that the mean of the random fields is close to zero over small numbers of realizations.

Reduced Forcing: The solution of the full CFD equations with stochastic forcing terms may lead to numerical difficulties. One mechanism to alleviate these difficulties is to effectively linearize the error forcing by scaling down the magnitude of the random number field used, and subsequently inverse scaling the SQA output perturbations. Note that deterministic results may diverge, too. Thus, the concept of reduced error forcing also applies to the defect correction problem.

11.0 DRAG PREDICTION WORKSHOP TEST CASE

Probabilistic error predictions were performed with Overflow and the Monte Carlo (MC) method of the DLR F4 wing-body configuration used in the first AIAA Drag Prediction Workshop (DPW). Calculations were run on a grid coarsened by a factor of two in each direction, and the computed error bars are compared to results on the reference grid. Truncation errors are computed using fourth-order discretization for the high-order part of Eq. (4). The uncertainty in the truncation error was assumed to be the magnitude of the truncation error. The truncation error field for each MC realization was computed as

$$TE(i,k;m) \;=\; TE_0(i,k) + \left|TE_0(i,k)\right| \, RN(i;m) \qquad \textbf{(11)}$$

in which *RN(i;m)* is a smoothed quasi-random field in the range [-1, 1]. The PDF of the random fields was not Gaussian, but it was center-weighted with reduced probability at the extremes (similar to the bimodal distribution shown above, Fig. 3). This calculation involved n_{perts} = 20 MC realizations, which is not converged. Calculations of simpler flows were observed to be within ±32.5% (to within 95% confidence) to ±48.7% (to within 99.7% confidence) of the asymptotic uncertainties with this many realizations. Because the RMS of the standard deviation tends to converge at a rate proportional to $n_{perts}^{-1/2}$, calculations that demand very high accuracy are expensive to do with Monte Carlo; however, quantitative *estimates* of uncertainty can be obtained more affordably. In this case, the Monte Carlo convergence inaccuracies are believed to be small relative to the predicted uncertainties in the CFD results.

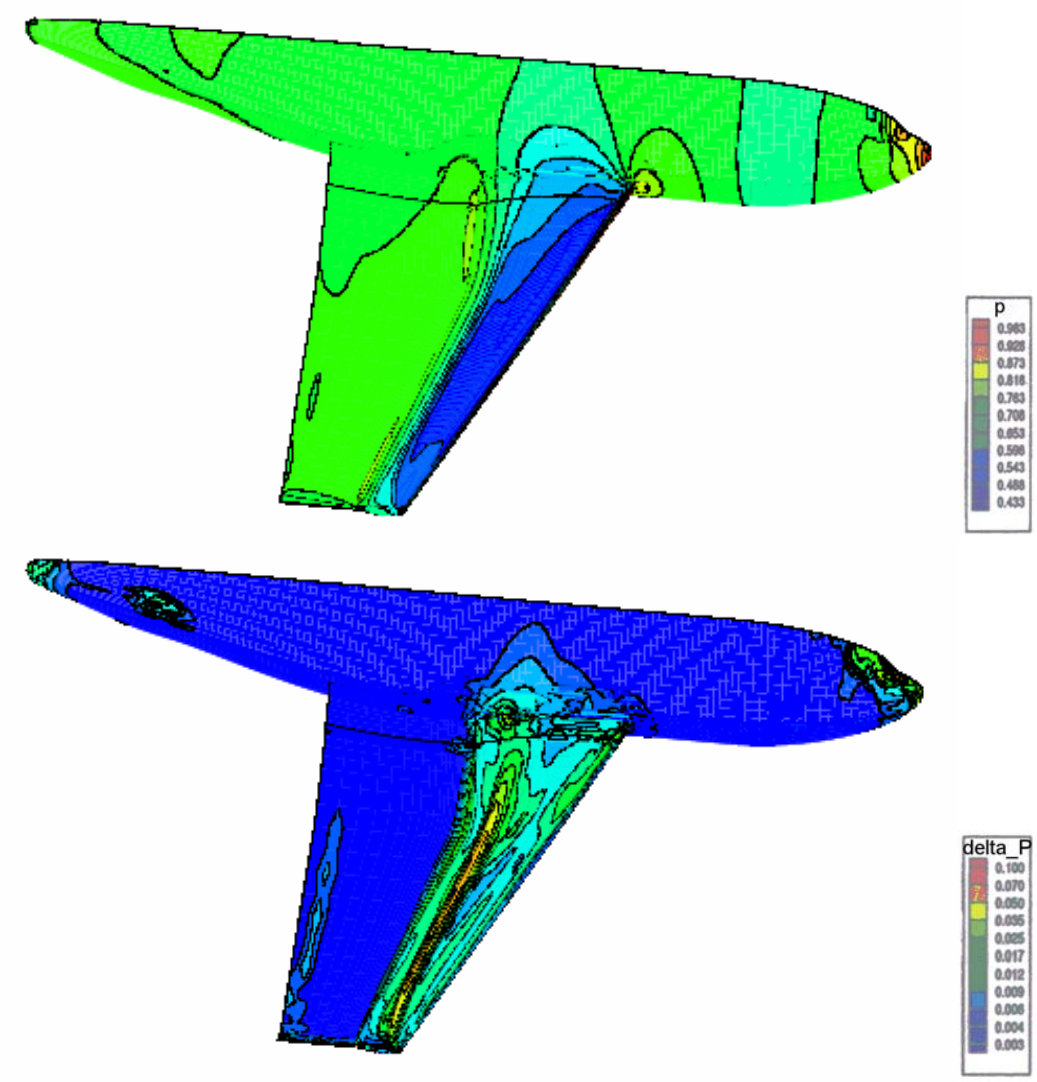

Figure 6: Monte Carlo and Overflow Prediction of (a) Surface Pressure (Top) and (b) Uncertainty in Pressure (Bottom), on the Surfaces and on the Wake Cut Grid, for DPW 1 at M = 0.8, C_L = 0.5.

Figure 6 gives the surface pressure and uncertainty at the 3σ level. The highest uncertainties occur at the shock. A Mach sweep was performed, using the angle of incidence that was found to give $C_L = 0.5$ on the default (finer) grid as part of the DPW effort (Ref. [1]). The 3σ error bars on the lift and pitching moment coefficients (**Figure 7**) are large enough to encompass the finer-grid solution. The drag coefficient is too small by roughly a factor of five. The majority of error in drag is in the viscous component, and there is very small predicted uncertainty in the viscous drag. The very high stretching rate, near 1.3, in the wall-normal grid spacing is the suspected cause of this shortcoming, as good predictions of the uncertainties in skin friction have previously been obtained in simpler flows with better grid resolution.

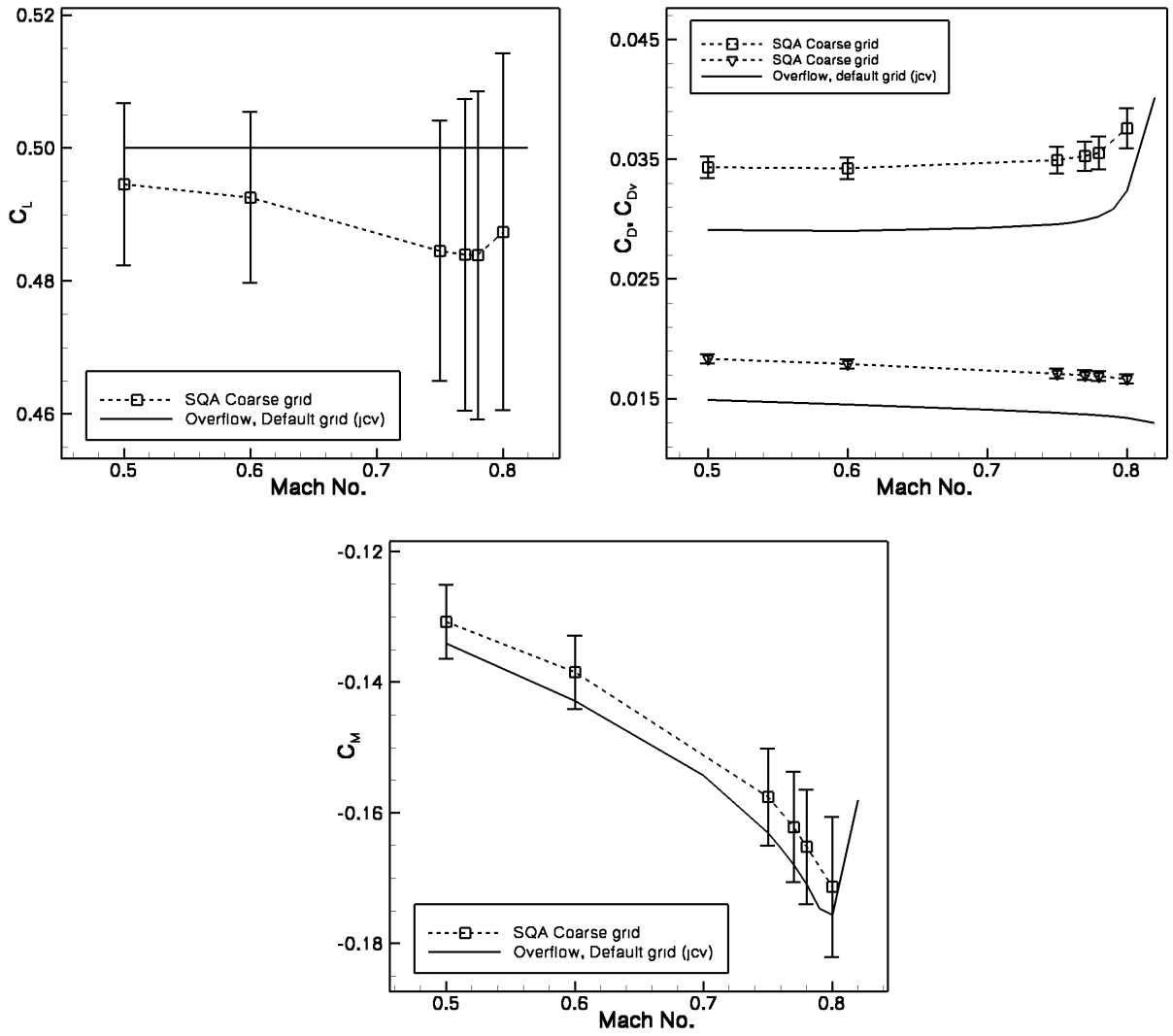

Figure 7: Monte Carlo and Overflow Prediction of Lift, Drag, and Pitching Moment for DPW 1. Mach Sweep at Angles of Incidence Determined in Default-Grid Calculation.

The application of the SQA methodology to the W2 wing-alone test case of the third Drag Prediction Workshop (DPW3, Ref. [2]) is shown next. **Figures 8-10** show the Monte Carlo and Overflow prediction for lift, drag, and pitching moment for three different computational grids as a function of the grid factor, $1/N^{2/3}$. Two of the three grids are the medium and coarse grids of the DPW3 OVERFLOW runs; the third is an 'extra-coarse' grid generated by decimation.

Referring back to the notation of Eq. (11), we consider three analyses for the *TE* source term: (1) *TE* is the truncation error per se ("TE-based SQA", Fig. 8), (2) *TE* represents turbulence modeling uncertainty source terms only ("TM-based SQA", Fig. 9), and (3) *TE* contains both the truncation error and the turbulence modeling uncertainty source terms, added together in the RSS sense ("TETM-based SQA", Fig. 10). In each figure the zero-grid factor extrapolation of Sclafani et al.'s results (Ref. [3]) is indicated by a yellow symbol to represent the asymptotic fine-grid calculation most closely matching the present algorithm. Note that perfect

agreement between the two is not necessarily to be expected because of a number of factors, including versions of OVERFLOW (the host CFD solver integrated with the present software is OVERFLOW 1.8r), possible differences in the Spalart-Allmaras model, and variations in the differencing schemes used.

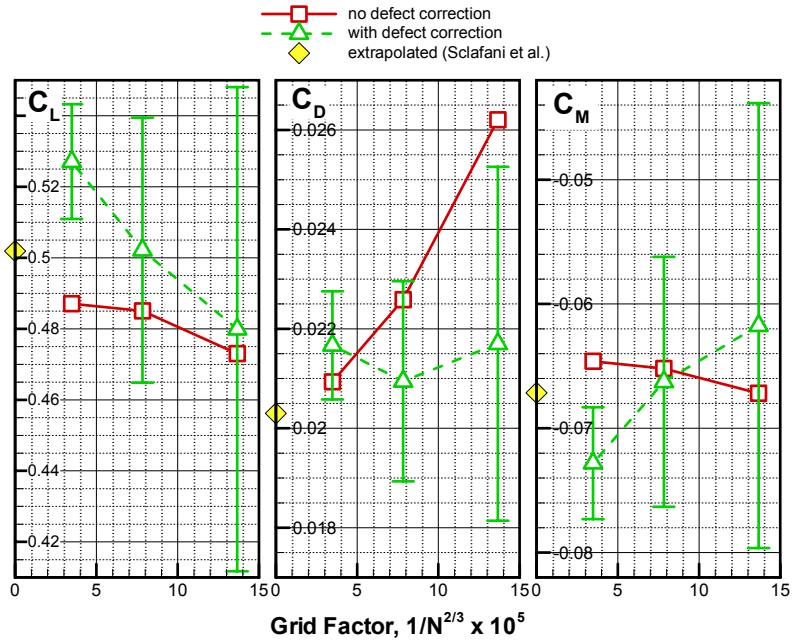

Figure 8: Monte Carlo and OVERFLOW Prediction of Lift, Drag, and Pitching Moment Coefficients for DPW 3 Wing2 Configuration, M = 0.76, α = 0.5°, *Re* = 5 million, TE-Based SQA (Defect Correction and Monte Carlo Perturbations Based on Truncation Error Only).

Figure 9 shows the Monte Carlo predictions when the SQA source term represents the turbulence modeling uncertainty only. In this case, the deterministic correction is not a defect correction per se but, rather, a mean correction to the Spalart-Allmaras (SA) turbulence model, which attempts to take into account the effects of rotation-curvature [4] and compressibility [5] on turbulence. For this analysis, the SQA perturbation parameters were determined from prior empirical calibrations obtained by applying the method to simpler canonical flows, such as the So-Mellor curved pipe flow [6], two-dimensional compressible mixing layers [7], and various supersonic compression ramps [8,9]. As previously mentioned the various uncertainty contributions were added in the RSS sense. Additional testing using compressibility and rotation-curvature corrections separately revealed that the compressibility effects account for approximately 18% of the lift correction, 15% of the drag correction, and 21% of the pitching moment correction; the majority of the SA turbulence modeling correction effects are therefore attributable to the rotation-curvature effects.

When plotted on the same scale as Fig. 8, it is evident that the uncertainty corresponding to the turbulence modeling terms is an order of magnitude smaller than the uncertainty associated with the truncation error.

Figure 10 is an example of a more "comprehensive" uncertainty estimation result, obtained when the SQA source terms include both the truncation error and the turbulence modeling uncertainty. Note that the deterministic (defect correction) error source terms are added algebraically, while the SQA perturbation source term magnitudes are added together in the RSS sense. This has the predictable effect that, since the truncation error defect correction for the lift coefficient is a positive increment (Fig. 8) and the turbulence modeling mean correction is negative (Fig. 9), the net lift increment for the combined analysis is somewhat reduced (Fig. 10). As more independent error components are taken into account, the error bars should increase. In

practice, this may not always be the case, however, due to Monte Carlo variations and small sample size considerations.

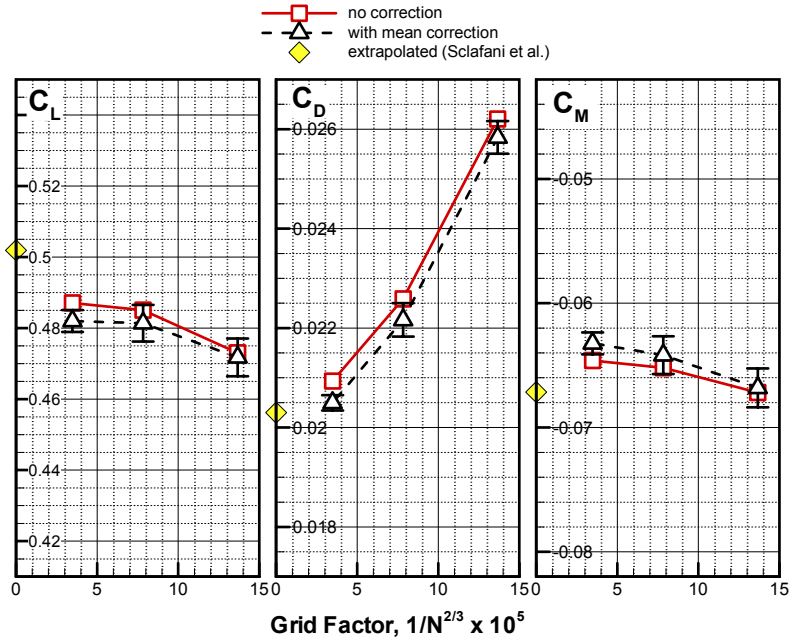

Figure 9: Monte Carlo and OVERFLOW Prediction of Lift, Drag, and Pitching Moment Coefficients for DPW 3 Wing2 Configuration, M = 0.76, α = 0.5°, *Re* = 5 million, TM-Based SQA (Mean Correction and Monte Carlo Perturbations Based on Turbulence Modeling Source Terms Only).

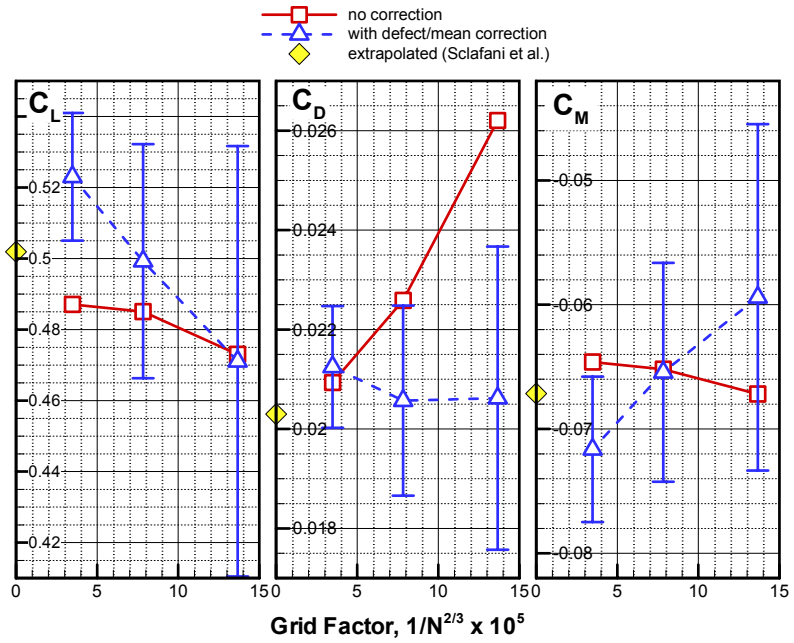

Figure 10: Monte Carlo and OVERFLOW Prediction of Lift, Drag, and Pitching Moment Coefficients for DPW 3 Wing2 Configuration, M = 0.76, α = 0.5°, *Re* = 5 million, TETM-Based SQA (Defect/Mean Correction and Monte Carlo Perturbations Combine Truncation Error and Turbulence Modeling Source Terms).

For the truncation error analysis (Fig. 8), the fact that the predicted error bars decrease with grid refinement is as expected. However, the fact that the defect-corrected means diverge from the OVERFLOW results is and remains a source of concern. The cause of this behavior is believed to be associated with the poor calculation of the truncation error near the trailing edge as a result of metric discontinuities at the square trailing edge. Specifically, the increase in the truncation error source term with grid refinement is reminiscent of the behavior at shocks, i.e., the truncation error is not obeying a "convergent-series" type behavior. Under these circumstances, the defect correction is likely to produce erroneous results, possibly including local divergence. This is a reasonable criterion for discarding the truncation error at a discontinuity. Recent implementation of this idea has had mixed success but suggests a large sensitivity to trailing edge treatment. Thus, the truncation error calculation at metric discontinuities remains a key research challenge.

12.0 FUTURE RESEARCH AREAS

Probabilistic error modeling that uses the adjoint to project the sources of error onto the output functional should be developed. Once the adjoint is computed, an inexpensive inner product between the truncation error and the adjoint gives an estimate of the deterministic error in the output. This deterministic error modeling has been demonstrated by Park [10]. However, deterministic error modeling is observed to be unreliable. Monte Carlo can be used to do economical probabilistic error modeling with the adjoint approach. Because of its anticipated economy, this approach will provide a useful tool for research in how to construct good probabilistic models of the error sources. It also has the potential to be useful in production CFD as the adjoint method for other activities like adaptive mesh refinement and calculation of gradients for design optimization.

The polynomial chaos (PC) method is currently viewed as the best approach to achieve full probabilistic error calculations. Prior PC work has focused on probabilistic boundary conditions, but not 3D field quantities like error sources, to our knowledge. Research on PC error modeling will cover much new ground. One concern is the modeling of probabilistic error sources. Another is the convergence of PC expansions in nonlinear flows that generate highly nonnormal probabilistic errors; many expansion terms may be required, at high computational cost, to accurately portray key features like the tails of PDFs.

Probabilistic modeling of errors caused by physical modeling (transition, turbulence, etc.) is an essential step toward comprehensive quantification of errors in CFD results. Work in this area must focus on the development and validation of models for the probabilistic sources of error in physical models. Error propagation can be treated with probabilistic methods, either defect correction or the adjoint method. Success in this area will spawn an entirely new approach to improving physical modeling. An important benefit will be the ability to optimize designs for a combination of high performance and low sensitivity of that performance to flaws in physical modeling.

13.0 ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of this work by NASA: specifically, NASA Ames Research Center for its support of the early Overflow work, and NASA Langley Research Center for its support of the turbulence modeling and unstructured grid work.

14.0 REFERENCES

[1] Vassberg, J. C., DeHaan, M. A., and Sclafani, A. J., "Grid Generation Requirements for Accurate Drag Predictions Based on OVERFLOW calculations," AIAA 2003-4124.

[2] Vassberg, J. C., Tinoco, E. N., Mani, M., Brodersen, O. P., Eisfeld, E., Wahls, R. A., Morrison, J. H., Zickuhr, T., Laflin, K. R., and Mavriplis, D. J., "Summary of the Third AIAA CFD Drag Prediction Workshop," AIAA 2007-260.

[3] Sclafani, A. J., Vassberg, J. C., Harrison, N. A., DeHaan, M. A., Rumsey, C. L., Rivers, S. M., and Morrison, J. H., "Drag Prediction for the DLR-F6 Wing/Body and DPW Wing Using CFL3D and OVERFLOW on an Overset Mesh," AIAA 2007-257.

[4] Shur, M. L., Strelets, M. K., Travin, A. K, and Spalart, P. R., "Turbulence Modeling in Rotating and Curved Channels: Assessing the Spalart-Shur Correction," *AIAA J.*, Vol. 38, No. 5, pp. 784-792, May 2000.

[5] Forsythe, J. R., Hoffman, K. A., and Squires, K. D., "Detached-Eddy Simulation with Compressibility Corrections Applied to a Supersonic Axisymmetric Base Flow," AIAA 02-0586.

[6] So, R. M. C. and Mellor, G. L., "Experiments on Convex Curvature Effects in Boundary Layers," *J. Fluid Mech.*, Vol. 60, Part 1, pp. 43-62, 1973.

[7] Lele, S. K., "Compressibility Effects on Turbulence," *Ann. Rev. Fluid Mech.*, Vol. 26, pp. 211-254, 1994.

[8] Settles, G. S., Vas, I. E., and Bogdonoff, S. M., "Details of a Shock-Separated Turbulent Boundary Layer at a Compression Corner," AIAA J., Vol. 14, No. 12, pp. 1709-1715, Dec. 1976.

[9] Settles, G. S. and Dodson, L. J., "Supersonic and Hypersonic Shock/Boundary Layer Interaction Database," AIAA J., Vol. 32, No. 7, pp. 1377-1383.

[10] Park, M. A., "Adjoint-Based, Three-Dimensional Error Prediction and Grid Adaptation," AIAA 2002-3286.

Paper No. 1

Discusser's Name: R. Dwight

Question: What were the uncertainty sources in the wing-body case?

Author's Reply: The source term in that case was the randomized truncation error, which is a three-dimensional field which changes at each Monte Carlo realization and with each equation being solved.

Discusser's Name: B. Kleb

Question: Why are only 10-20 Monte Carlo samples sufficient?

Author's Reply: While high numbers of Monte Carlo realizations were used for the PDF results shown here, I acknowledge this is rarely affordable. For the DPW results, the average number of Monte Carlo realizations was about 40. The reason a small number of realizations may be acceptable is that the uncertainty in the RMS due to sampling decays roughly like the square root of the number of samples. Therefore, most of the benefit of reducing sampling uncertainty is realized early on with, say, 10-20 realizations. I should point out that the CL, CM, and CD uncertainties shown for the DPW3 test case are equal to three times the standard deviation, multiplied by an empirical factor to correctly account for this additional sampling uncertainty. If higher-order statistics were required, then a larger number of Monte Carlo realizations would be needed.