36th AIAA Fluid Dynamics Conference and Exhibit, 5-8 June, 2006, San Francisco, CA

# A Manufactured Solution for Verifying CFD Boundary Conditions, Part III

Ryan B. Bond*, Curtis C. Ober†, and Patrick M. Knupp‡

*Sandia National Laboratories§, P. O. Box 5800 Albuquerque, NM 87185*

**Order-of-accuracy verification is necessary to ensure that software correctly solves a given set of equations. One method for verifying the order of accuracy of a code is the method of manufactured solutions. Part III of this study completes the development of a manufactured solution that allows verification of not only the Euler, Navier-Stokes, and Reynolds-Averaged Navier-Stokes equation sets, but also some of their associated boundary conditions: slip, no-slip (adiabatic and isothermal), and outflow (subsonic, supersonic, and mixed). In order to demonstrate the usefulness of this manufactured solution, it has been used for order-of-accuracy verification in a compressible computational fluid dynamics code. All of the results shown are on skewed, non-uniform, three-dimensional meshes. Modifications have been made to the manufactured solution and sequence of meshes from previous work to allow asymptotic results to be obtained with less computational cost. In addition to the order of accuracy of the full code for various equation sets and boundary conditions, the order of accuracy of code portions used to calculate solution gradients has been measured as well.**

## 1. Introduction

The purpose of this work is to complete the development of a manufactured solution capable of testing governing equations and boundary conditions (BC's) implemented in computational fluid dynamics (CFD) codes on non-uniform meshes. In order to demonstrate that this manufactured solution is both correct and useful, it is applied to the verification of Premo,[1] a compressible fluid dynamics code being developed at Sandia National Laboratories. Premo is used to determine aerodynamic performance of complex geometries, and it is built within the SIERRA framework[4] to allow multi-physics coupling (*e.g.*, heat transfer and fluids for ablation or structures and fluids for aeroelasticity). It is a parallel, unstructured, edge-based, finite-volume code. Thus, the results in this series of papers are twofold: first, a manufactured solution for verification of CFD codes and, second, a verification study of Premo using this manufactured solution.

Refs. 2 and 3 contain earlier versions of the manufactured solution used in the present work, a derivation of this manufactured solution, and results of tests on the Euler and Navier-Stokes equations and some of the related BC's. The current version of the manufactured solution and current sequence of meshes allow asymptotic behavior to be more easily observed for the order-of-accuracy verification tests with reasonable computational costs. Differences between the current and previous manufactured solutions are detailed in Section 3, and the current sequence of meshes is described in Section 2.

In Ref. 2, the slip condition showed unordered behavior (*i.e.*, non-positive order of accuracy) and the Euler equations using least squares gradient reconstruction demonstrated first-order behavior when second-order was expected. After Ref. 2 was published, two coding mistakes were identified. The mistake associated with the slip BC involved the misuse of a unit normal calculation function intended for planar surfaces only. Ref. 3 repeated the order verification test for the slip BC after the known coding mistake was corrected, but unfortunately, positive order of accuracy (*i.e.*, consistency) was still not observed. The poor least

*Aerosciences and Compressible Fluid Mechanics Department, MS 0825, E-mail: rbbond@sandia.gov. Member, AIAA.

†Computational Sciences Department, MS 0316, E-mail: ccober@sandia.gov. Senior Member, AIAA.

‡Optimization and Uncertainty Estimation Department, MS 1111, E-mail: pknupp@sandia.gov.

§Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed-Martin Company, for the United States Department of Energy's National Nuclear Security Administration under Contract DE-AC04-94AL85000.

This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

squares results in Ref. 2 were caused by an indexing mistake in the gradient calculation function.[a] Ref. 3 demonstrated that second-order behavior was observed after the coding mistake was corrected. Ref. 3 also continued the work by applying the manufactured solution to the verification of the Navier-Stokes equations (another coding mistake was found and corrected during this process). The current work completes the study by modifying the manufactured solution to allow asymptotic results to be obtained at reasonable mesh resolutions and by verifying the observed order of accuracy for the Reynolds-Averaged Navier-Stokes (RANS) equations in Premo. Although the implementation of the manufactured solution is done for Premo, the derivation respects the needs of general CFD codes and should be useful for their verification.

The method of manufactured solutions (MMS) provides order-of-accuracy verification of the implemented equation sets along with their various BC's.[5,6] Such verification is a rigorous way to ensure that the governing equations are solved correctly. Documentation of an observed order of accuracy of certain algorithms, which matches their formal order of accuracy, builds confidence in the user community that those algorithms have been chosen and implemented properly.[b] MMS derives its name from a solution which is manufactured for the sole purpose of verifying the order of accuracy of the code. It is a special case of the method of exact solutions, where discrete solutions on a sequence of uniformly refined meshes are compared to an exact solution. Manufactured solutions are used where

1. no exact solutions exist,
2. exact solutions are not sufficiently general to fully test a code, or
3. exact solutions are not smooth enough to allow proper implementation and error analysis.

Unlike an exact solution, a manufactured solution does not actually need to satisfy the original governing equation set for a given set of initial and boundary conditions. Instead, the original equation set is modified through the addition of a source term so that the manufactured solution is an exact solution to the equation set with this additional source term. The exact form of the source term depends on the particular manufactured solution selected. This form is found by applying the equation operators to the manufactured solution to obtain an analytic formula for the source. The source term is then added to the original equation set in order to balance it. If functions for calculating the source term are added to the code, it can be used to solve the equation set with the additional source term.[c] Then, the discrete solutions produced by the code can be compared to the manufactured solution in order to determine the discretization error. A comparison of the discretization error on a series of uniformly refined meshes will either verify that the observed order of accuracy matches the formal order of accuracy[d], or it will not. In the latter case it may indicate a coding mistake or improper formulation. Other possibilities are discussed in Ref. 5.

Refs. 2 and 3 give detailed descriptions of RANS equations, so no statement of the governing equations is included here.

A proposed form for a steady manufactured solution of any flow variable $\phi$ is

$$\phi = \phi_0 \left(1 + f(Ax)f(By)f(Cz)\right), \tag{1}$$

where $A$, $B$, and $C$ are constants. The function $f$ must be chosen so that this form satisfies the requirements of Knupp and Salari.[5] To avoid symmetry within the solution, $A \neq B \neq C \neq 0$. If $\phi_0$ is a constant, this form is a separated solution plus a constant.[8] The separated form makes deriving manufactured solutions with specific Neumann BC's (such as a vanishing normal derivative) easier. Constant values for $\phi_0$ are not sufficient for all of the BC's of interest, but if $\phi_0$ is chosen appropriately, the necessary properties hold.

Ref. 3 gives a detailed theoretical introduction to the derivation of manufactured solutions for use in order-of-accuracy verification of boundary conditions. That discussion is not repeated here. However, some issues were overlooked in Ref. 3, so additional discussion is necessary.

Earlier versions of the manufactured solution[2,3] had constant pressure ($p$) and varying normal Mach number ($M_n$) along a curved surface which was used for testing of the outflow BC. The surface was defined

[a] Ref. 2 presented some incorrect conclusions regarding the least squares results. A correct analysis of this case is contained in Ref. 3.

[b] One important but often overlooked benefit of formal order verification is that it can not only find implementation errors—it can also identify weaknesses in correctly implemented formulations and algorithms. In many cases, the failure of a code to match a desired order of accuracy may require a different formulation as a remedy, rather than correction of a coding mistake.

[c] The mechanism for this in Premo is a user-defined source routine.

[d] This assumes that the formal order of accuracy is known with certainty, which may not be the case. For cases where it is not, there is probably a desired order of accuracy which the code is meant to achieve.

by the function $G(x, y, z) = C_o$. This constant-$p$ surface was necessary for testing the outflow BC for subsonic conditions ($M_n < 1$). Under subsonic conditions, data corresponding to one incoming characteristic (*i.e.*, one negative eigenvalue) must be specified, and Premo, like many CFD codes, allows a constant $p$ for the outflow surface to be specified by the user.

Tests of the outflow BC in Ref. 3 used meshes which had a boundary defined along the $G = C_o$ surface, allowing tests of the outflow BC at constant $p$ for a wide range of $M_n$. However, since there is a more general case (non-constant $p$) of the supersonic outflow condition than there is for the subsonic outflow condition, it is preferable to have the mesh boundary deviate from the $G = C_o$ (and thus $p$-constant) surface in the supersonic region.[e] The new mesh definition is described in Section 2.

## 2. The Computational Domains

The functions $F(x, y, z)$ and $G(x, y, z)$ are used to define some of the boundaries of the computational domain. A third function, $H(x, y, z)$, is also needed to construct the domain, although no BC's are tested on $H$-constant surfaces (*i.e.*, instead of any special BC's, Dirichlet BC's are enforced on $H$). The no-slip surface is the $F = 0$ surface, and the $p$-constant surface is the $G = 0$ surface. All other $F$-constant surfaces are slip surfaces. $F$, $G$, and $H$ are defined as follows:

$$F = \frac{1}{2}\cos(A_f x)\cos(B_f y) - z \tag{2}$$

$$G = x - \frac{1}{2}\cos(A_g y)\sin(B_g z) - \frac{\pi}{4} \tag{3}$$

$$H = -y. \tag{4}$$

A simple definition of $H$ is permissible since no BC's will be tested on $H$-constant surfaces.[f] The version of the solution used in this work uses $A_f = \frac{1}{2}$, $B_f = \frac{3}{4}$, $A_g = \frac{1}{4}$, and $B_g = \frac{1}{2}$. This represents a reduction in the wave numbers from Ref. 2 by a factor of four. This was done in an attempt to reduce the mesh resolution required to reach the asymptotic regime. It is quite helpful if the mesh resolution is sufficient to identify coding mistakes on coarser meshes. If the observed order of accuracy is poor on coarser meshes, then coding mistakes can be identified and fixed before much computational time is invested in running MMS cases on finer meshes. Finer meshes can later be used to confirm the coarse mesh results.

To address the problem caused by using a constant-$p$ solution on supersonic regions of the outflow boundary in Ref. 3, a new function, $\widetilde{G}$, is defined as follows:

$$\widetilde{G} = \begin{cases} G & : \quad M_n \leq 1 \\ G + \exp\left(\frac{1}{(M_n-2)^2-1}\right) & : \quad 1 < M_n < 2 \\ G + \exp(-1) & : \quad 2 \leq M_n \end{cases} . \tag{5}$$

This function was chosen because $\widetilde{G} = G$ on the subsonic portions of the boundary (where constant $p$ is desired); $\widetilde{G} \neq G$ on the supersonic portions (where non-constant $p$ is desired), and $\widetilde{G}$ is infinitely differentiable. A six-sided domain can be defined by setting maximum and minimum values for $F$, $\widetilde{G}$, and $H$.

Figures 1, 2, and 3 show the three domains used for testing. In each figure, the view on the left shows the volume mesh, while the view on the right shows the boundaries which were used for BC testing. Figure 1 shows the domain that was used for testing the slip condition along its $F_{max}$ boundary with the Euler equations on its interior and the supersonic outflow condition along its $\widetilde{G} = 0$ boundary with the Euler

[e] A coding mistake was actually identified by other means which alerted the authors to the danger of the lack of generality that results from having the supersonic outflow boundary lie along a $p$-constant surface. One of the requirements of manufactured solution design is that the solution be useful for the most general case of code application.[5] The same principle applies to the definition of the mesh. The meshes used in Refs. 2 and 3 were, in effect, blind to this particular type of coding mistake. The mesh definition in the present work tests the most general cases: a subsonic, constant-$p$ outflow boundary; a supersonic, variable-$p$ outflow boundary; or the union of both into a mixed boundary.

[f] The sufficiency of a simple definition for $H$ depends upon the BC's being implemented generally in the spatial sense. If they are not, then something must be done to account for this lack of generality. For example, if a structured mesh code has six separate implementations of each BC corresponding to the six bounding surfaces of the computational domain, then each of the six implementations will need to be tested. If this manufactured solution is to be used for such a case, then several orientations of the computational space ($\xi$,$\eta$,$\zeta$) relative to the $F$, $G$, and $H$ surfaces will need to be used. This is not necessary for Premo, since it is an unstructured code with BC's implemented in a general manner.

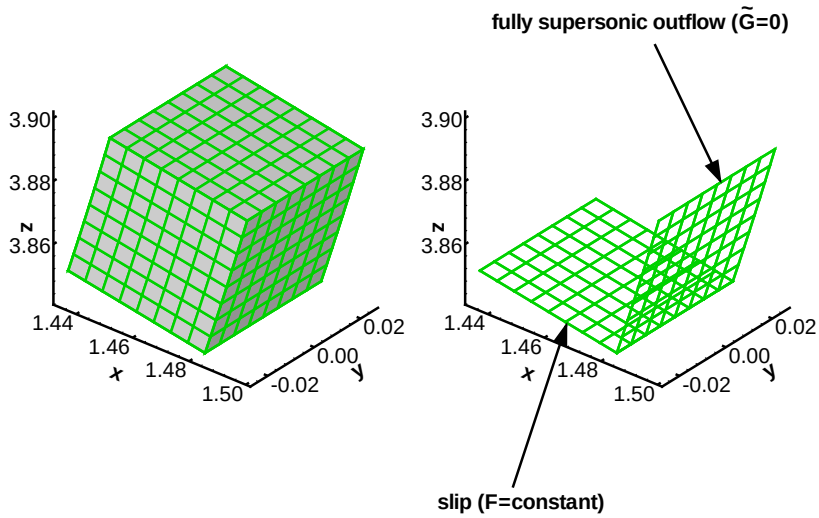


**Figure 1. Domain used for testing of slip condition and supersonic outflow.**

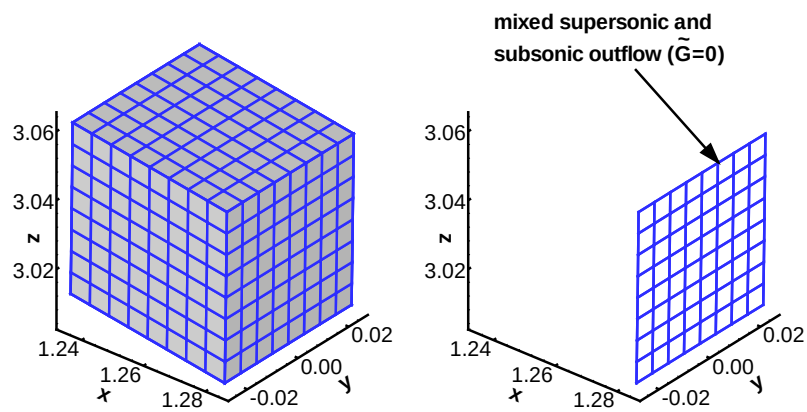


**Figure 2. Domain used for testing of mixed outflow.**

equations on its interior. It is defined in $(F, \widetilde{G}, H)$ space as $-0.05 < F < 0$, $-0.05 < \widetilde{G} < 0$, and $-0.025 < H < 0.025$. The domain shown in Figure 2 was used to test a mixed subsonic and supersonic outflow condition along its $\widetilde{G} = 0$ boundary with the Euler equations on the interior. This domain was also used to test the Euler, Navier-Stokes, and RANS interior equation sets with Dirichlet BC's before testing of the specialized BC's began. It is defined in $(F, \widetilde{G}, H)$ space as $-2.655 < F < -2.605$, $-0.05 < \widetilde{G} < 0$, and $-0.025 < H < 0.025$. The domain shown in Figure 3 was used for testing the no-slip condition along its $F = 0$ boundary with the Navier-Stokes equations on its interior and the subsonic outflow condition along its $\widetilde{G} = G = 0$ boundary with the Euler equations on its interior. It is defined in $(F, \widetilde{G}, H)$ space as

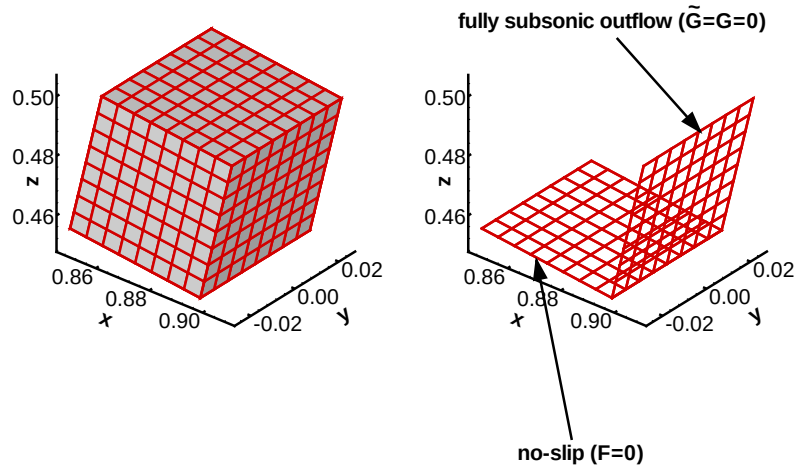


**Figure 3. Domain used for testing of no-slip condition and subsonic outflow.**

$-3.525 < F < -3.475$, $-0.05 < \widetilde{G} < 0$, and $-0.025 < H < 0.025$. These domains are much smaller (in the physical space) than the domains used in Refs. 2 and 3. The small domains were needed to produce convincingly asymptotic results for the observed order of accuracy.

The meshes on these computational domains are hexahedral and evenly spaced in $(F, \widetilde{G}, H)$ space, with $N \in \{9, 17, 33, 65\}$ nodes in each direction. This results in meshes which are non-uniform, skewed, and smooth in $(x, y, z)$ space. Although no results on the $N = 129$ mesh are included in this paper, the mesh exists and will be used in the future to confirm asymptotic behavior of cases with favorable results on the $N \in \{9, 17, 33, 65\}$ meshes.

## 3. The Manufactured Solution

Once $F$, $G$, and $H$ are defined, the manufactured solution can be defined as a function of $F$, $G$, and $H$. Since the manufactured solution is given in terms of $G$, and the meshes are defined in terms of $\widetilde{G}$, the needed $p$ variation is maintained for supersonic outflow tests. Only the parameters of the solution which differ from Ref. 3 are listed here.

$$u = \frac{1}{2}\left(-F\right)\mathcal{B}_1 a_{\mathrm{sl}} \tag{6}$$

$$v = \frac{1}{2}\left(-F\right)\mathcal{B}_2 a_{\mathrm{sl}} \tag{7}$$

$$\widetilde{\nu}_\infty = 1.95 \times 10^7 \times \nu_{\mathrm{sl}} \tag{8}$$

$$A_t = 1 \tag{9}$$

$u$ is the velocity in the $x$-direction; $v$ is the velocity in the $y$-direction; $a_{\mathrm{sl}}$ is a reference speed of sound; $\mathcal{B}_1$ and $\mathcal{B}_2$ are functions of $F$ and $G$ used to determine locations of inflow and outflow boundaries; $\widetilde{\nu}_\infty$ is a reference value for the manufactured solution of $\nu$ (the Spalart-Allmaras[7] working variable); $\nu_{\mathrm{sl}}$ is a reference value for the kinematic viscosity, and $A_t$ is one of the wave numbers in the manufactured solution for the temperature, $T$.

In order for the effects of the Spalart-Allmaras transport equation to be significant on such a smooth solution, the reference value of $\widetilde{\nu}$ was raised by seven orders of magnitude. This is similar to the raising of the molecular viscosity in Refs. 3, 9, and 10, which balanced the viscous and inviscid fluxes. A brief explanation is also given in Section 4.2. The intent of such term balancing is so that the numerical errors generated by

each of the terms in the equations will be of approximately the same magnitude. Otherwise, coding mistakes (which produce unordered error or error of less than the desired order) in small terms may be undetectable. Changes to $u$, $v$, and $A_t$ from Refs. 2 and 3 were made to control the locations of subsonic and supersonic regions.

# 4. Testing of Interior Equations

Before any specialized BC is tested, it is a good idea to test the interior equations using only Dirichlet[g] BC's. This allows the interior equation set to be verified so that any difference between the formal and observed orders of accuracy in the BC tests can be rightly interpreted as a problem with the BC enforcement, rather than as a problem with the interior equation set. However, it is not necessary to verify every possible option for the interior equation set before progressing to the BC's. Examples of such options would be choice of auxiliary relations such as the equation of state or transport property definitions. It is only necessary to verify in advance the same set of options which will be used in the BC tests.

## 4.1. The Euler Equations

### *4.1.1. Summary of Prior Verification of the Implementation of Roe's Method*

Ref. 2 showed behavior approaching second-order for the Euler equations using Roe's[12] approximate Riemann solver with extension to second-order provided by a particular approximate form of Green-Gauss gradient reconstruction. However, use of least squares gradient reconstruction produced only first-order results. It was erroneously concluded that this reduction in order was caused by the use of equally-weighted least squares as opposed to inverse-distance-weighted least squares. Instead, a coding mistake in the least squares gradient calculation was reducing the observed order. Since prior MMS verification[9,10] showed second-order behavior for least squares on uniform meshes, an explanation was sought for why first-order behavior was seen on non-uniform meshes. In fact, the correct explanation was simpler: a coding mistake was introduced in Premo during a revision between the two verification attempts. Because of this, a regression test was added to the Premo test suite to prevent future recurrences.[h] Ref. 3 showed behavior approaching second-order for inverse-distance-weighted least squares. Ref. 3 also showed results for equally-weighted least squares; although no evidence of first-order error was present, the results were not sufficiently asymptotic to make a strong claim for second-order behavior.

More recent results have shown that second-order behavior can not be demonstrated with Premo's approximate form of Green-Gauss gradient reconstruction on hexahedral element types when the meshes are sufficiently refined. However, rather than detecting error of order one or zero, the MMS results simply fail to iteratively converge when mesh resolution is increased beyond a certain threshold. The specific issue with the approximate form of Green-Gauss gradient reconstruction on hexahedral meshes is discussed in Section 6. This point calls into question the results of Ref. 2, and it is one of the reasons that the manufactured solution and associated meshes have been modified for the present work. Section 7 gives more details on this matter. Results obtained with the current manufactured solution and sequence of meshes confirm second-order behavior (as was believed to be the case in Ref. 3) for Roe's method on the Euler equations using inverse-distance-weighted least squares gradient reconstruction. Order plots are omitted here because tests for the Navier-Stokes and RANS equations in Sections 4.2 and 4.3 confirm second-order behavior for the underlying inviscid fluxes as well.

### *4.1.2. STVD Verification Effort*

Although there is some overhead involved in deriving manufactured solutions, providing source terms to computational codes, and performing error analysis after runs, the majority of effort is expended in setting up and completing the first test. Although subsequent tests consume similar amounts of computational

[g]In this case, the term "Dirichlet" is used in its mathematical sense to mean that the value of the dependent variable is specified at various points along the boundary. In Premo, the Dirichlet option enforces the special case where the dependent variable is set to a constant along an entire boundary segment. Since the manufactured solution does not fit this special case, Premo's user-defined BC option was used to enforce the spatially-varying Dirichlet BC's.

[h]This brings up an important point for verification of codes in development. Once a particular set of options has been verified, regression tests must be implemented to prevent code modifications from changing the computational results obtained with this set of options.

resources, very little additional labor is required. This was demonstrated in Premo testing when various options for the freshly-implemented symmetric total variation diminishing (STVD) scheme were evaluated.[11] (Results in Refs. 2 & 3 and Sections 4.2, 4.3, 5.2 & 5.3 use Roe's method.) The initial exploratory space for STVD schemes was very large because

1. different options for adapting the structured grid formulations to Premo's unstructured discretization required consideration;[i]

2. because the limiter is an integral part of the dissipation term, each limiter option results in a slightly different scheme, requiring separate verification, and

3. different options existed for calculating the eigenvector matrix and smoothing the eigenvalues.

In less than a week, 24 separate STVD implementations using gradient-based extrapolation were tested. Several more implementations using collinear edges will be examined in the near future. The MMS tests for these different implementations allowed order-of-accuracy measurements for each scheme on a sequence of meshes as well as error magnitude comparisons between the different schemes at each mesh size. Figure 4

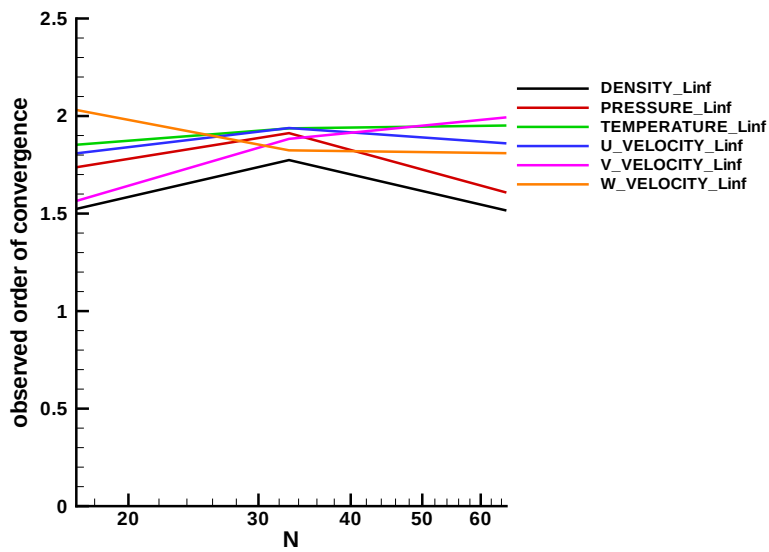


**Figure 4. Convergence order of $L_\infty$ norms for the Euler equations with one of the STVD implementations.**

shows observed order of accuracy of the $L_\infty$ norms of each of the flow variables for one of the schemes.[j] $N$ is the number of nodes in each of the three mesh dimensions. For mesh refinement via doubling, the observed order of accuracy is given by the following equation:

$$\text{observed order} = \frac{\log\left(\frac{\|\epsilon_{N/2}\|}{\|\epsilon_N\|}\right)}{\log(2)}, \tag{10}$$

where $\epsilon$ is the error (difference between discrete and exact solutions) for the flow variable of interest, and $\|\epsilon\|$ is any norm of that error. The convergence orders of the $L_\infty$ norms are given for the errors in the flow variables $\rho$, $p$, $T$, $u$, $v$, and $w$. The $L_\infty$ norm is used because it

[i]Examples include gradient-based extrapolation versus collinear edge for hexahedral meshes, options for modifying the scheme near boundaries, etc.

[j]A detailed comparison of the different STVD versions is well beyond the scope of this paper. The results in Figure 4 correspond to the version with the two-argument Van Albada limiter (left & right arguments for interior edges and "outside" argument replaced by "middle" argument for boundary terminating edges), extrapolated conservative variables and eigenvectors (both via inverse-distance-weighted least squares gradients), and eigenvalue smoothing. This is just an example and is not intended as an endorsement of this particular formulation.

1. tends to indicate non-convergence on coarser meshes than the $L_1$ or $L_2$ norms,

2. generally requires finer meshes to produce convincingly asymptotic behavior for the observed order of accuracy, thus building more confidence that the asymptotic regime has in fact been reached if plots indicate so, and

3. may find local sources of numerical error with order less than the global sources of numerical error if it converges to a value less than the converged values of the $L_1$ and $L_2$ norms (Ref. 3 gives examples where the converged values for the $L_\infty$ norms differ from the $L_1$ and $L_2$ norms, and Ref. 15 gives a thorough theoretical discussion of why).

The results in Figure 4 show observed orders of accuracy between 1.5 and 2.0 for the $L_\infty$ norms of each of the variables. Although numbers closer to 2.0 would be preferable for all of the norms, some degradation in order performance should be expected from the eigenvalue smoothing and influence of the limiter.[k] Once final candidate formulations for the STVD scheme are selected, order behavior without the eigenvalue smoothing will be examined. MMS testing is a critical part of the evaluation of the different STVD implementations because of the hope that STVD will provide better robustness than the previously verified flux schemes. An incorrectly formulated or implemented STVD scheme might show good iterative convergence and robustness if it contains some first-order error, so MMS testing is needed as a "sanity check" to verify proper order-of-accuracy behavior.

### 4.2. The Navier-Stokes Equations

MMS verification of the Navier-Stokes equations can be performed on smaller meshes if values for the viscous transport properties $\mu$ and $k$ (the viscosity and thermal conductivity) cause the viscous fluxes and convective fluxes to be of similar magnitude. If the viscous fluxes are significantly smaller than the convective fluxes, coding mistakes present in viscous momentum or thermal flux calculations may not be observable by MMS tests except on extremely fine meshes.

Since Sutherland's law ($\mu = \frac{C_1 T^{1.5}}{T+C_2}$) and constant Prandtl number ($\mathrm{Pr} = \frac{\mu C_p}{k}$, where $C_p$ is the specific heat at constant pressure) were chosen as the options of highest priority for verification, the constants in Sutherland's law and the value of the constant Prandtl number were changed to allow the viscous momentum and thermal fluxes to be large enough that they were significant compared to the convective fluxes. The value used for $C_1$ was 100, while a typical value for air is $1.458 \times 10^{-6}$. The typical value for air of 110.4 was used for $C_2$. This effectively raised the viscosity by eight orders of magnitude for the Navier-Stokes tests, making them highly sensitive to coding mistakes in the viscous flux calculations. A Prandtl number of 1 was used so that the thermal fluxes would be of the same order of magnitude as the viscous momentum fluxes. (For RANS tests, $C_1 = 1$ was used because of the additional influence of the eddy viscosity.)

Initial MMS tests using the Navier-Stokes equations revealed the presence of a coding mistake. Once this mistake was corrected, results from Ref. 3 showed second-order accuracy for the $L_1$ and $L_2$ norms of the errors in all of the variables as well as the $L_\infty$ norms of the errors of most of the variables. However, the $L_\infty$ norms of the $p$ and $\rho$ errors appeared to be asymptotically approaching 1.5.

Results using inverse-distance-weighted least squares gradient reconstruction on the current sequence of meshes are shown in Figure 5. These results demonstrate behavior approaching second-order for all of the flow variables, in contrast to the results of Ref. 3 This improvement is likely the result of switching from a particular approximate form of Green-Gauss gradient reconstruction to inverse-distance-weighted least squares gradient reconstruction. Section 6 gives more details on the gradient issue. Future work will include an execution of this case on the $N = 129$ mesh to confirm asymptotic behavior.

### 4.3. The RANS Equations

As mentioned in Section 4.2, $C_1$ in Sutherland's law was set to 1 for the RANS tests in order to better balance the terms in the equations. Also, $\mathrm{Pr}_t = 0.9$ was used for the turbulent Prandtl number; $\widetilde{\nu}_\infty = 1.95 \times 10^7 \times \nu_{\mathrm{sl}}$ was used as the reference value for the Spalart-Allmaras working variable, and $\kappa = 1 \times 10^6$ was used for the Kármán constant (the standard value for $\kappa$ is 0.41).

[k] Although the eigenvalue smoothing could be turned off, it is not possible to turn off the limiter for the STVD formulations. The results with Roe's method in Refs. 2 & 3 and Sections 4.2, 4.3, 5.2 & 5.3 do not have eigenvalue smoothing or limiting.

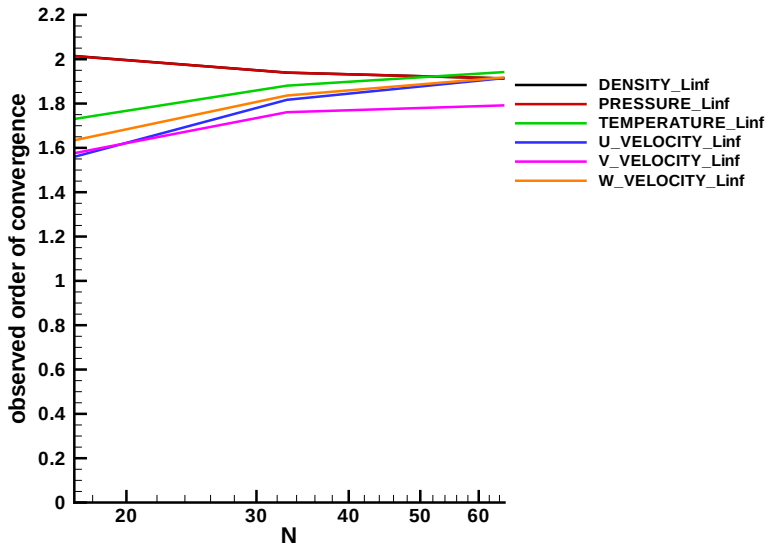


**Figure 5. Convergence order of $L_\infty$ norms for the Navier-Stokes equations.**

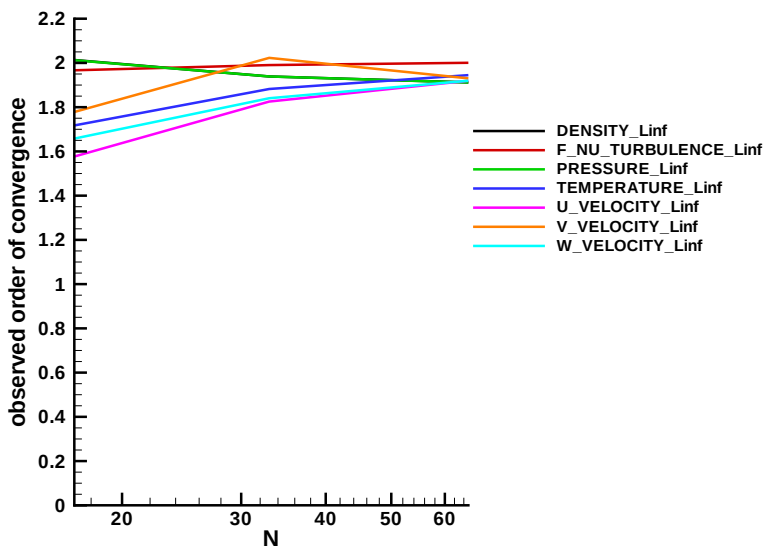


**Figure 6. Convergence order of $L_\infty$ norms for the RANS equations.**

Figure 6 shows the observed order of accuracy of the $L_\infty$ norms of the errors in each of the flow variables for the RANS equations using inverse-distance-weighted least squares gradient reconstruction. These results show behavior in the vicinity of second-order for the interior equations. Future work will include an execution of this case on the $N = 129$ mesh to confirm asymptotic behavior.

## 5. Testing of Boundary Conditions

Once a set of governing equations has been verified on the interior of a domain, the same code options and domain can be used to test a given BC by using it on the appropriate surface instead of the Dirichlet BC. The BC can only be verified to an order equal to or less than the order observed using the same interior equation set with Dirichlet BC's applied. Note that some BC's in many CFD codes with second-order interior schemes (including Premo) should be expected to generate first-order error. Whether or not this error remains local to the boundary or propagates into the interior depends upon the mathematical character at the boundary.[14] If first-order error is generated locally at a boundary and does not propagate into the domain, then it is possible for the $L_1$ or $L_2$ norms of the error to show second-order convergence when the error calculation is performed over the entire domain.[15] This is true for the $L_1$ and $L_2$ norms because the boundary nodes compose a smaller fraction of the total number of nodes as the mesh is refined. On the other hand, if the $L_\infty$ norm is used, or if the calculations for the $L_1$ or $L_2$ norms are restricted to the boundary region generating the first-order error, then the first-order error should be detected. Because of this, it is sometimes necessary to look at different error norms over different subsections of the domain to better determine the order behavior of a given BC implementation.

### 5.1. Summary of Prior BC Testing

The Euler BC's testable by the current manufactured solution are the slip condition and outflow condition with the normal flow subsonic, supersonic, or mixed. The outflow condition implementation was tested for fully supersonic outflow in Ref. 2. It was tested for transonic conditions and near-stagnation conditions in Ref. 3. Updated results are contained in Section 5.2.

Problems with the slip condition were observed in Ref. 2 wherever boundaries for interprocessor communication intersected with the slip surface. In that particular case, a unit normal function valid for planar surfaces was used for the curved slip surface. A new function was written which was valid for curved surfaces as well. After this new function was properly referenced by the slip condition implementation, tests were repeated, and the results are given in Ref. 3. Unfortunately, positive order of accuracy (*i.e.*, consistency) was still not observed in the tests.

Ref. 3 shows preliminary results for the outflow tests on mixed subsonic and supersonic outflow and for fully subsonic outflow. The domain for the fully-subsonic case contains a line with $\vec{v} = 0$ where the outflow boundary meets the no-slip boundary (although a coupled no-slip and outflow test was not attempted). The mixed subsonic and supersonic outflow test performed in Ref. 3 showed behavior approaching second-order, but error of order less than two was observed for the case where the outflow boundary contains a line with $\vec{v} = 0$.

The Navier-Stokes BC's testable by the current manufactured solution are the adiabatic and isothermal no-slip conditions. Results for tests of these cases are shown in Ref. 3. The conclusions of Ref. 3 are that some first-order error is generated by these BC's and that this error does not propagate into the interior in detectable amounts. Updated results which cast doubt upon prior conclusions are discussed in Section 5.3.

### 5.2. Further Testing of the Outflow BC

Figure 7 shows the observed order of accuracy of the $L_\infty$ norms for the mixed subsonic and supersonic conditions, while Figure 8 shows the convergence order of the $L_\infty$ norms in the $\vec{v} \to 0$ limit. Both figures indicate the presence of error of less than second-order, although neither shows asymptotic behavior indicative of first-order error. This is in contrast to the results from Ref. 3, where only the $\vec{v} \to 0$ case showed error of less than second-order. The transonic case presented in Ref. 3 probably lacked sufficient resolution to detect the error of less than second-order. The presence of error of less than second-order is not surprising, since all of Premo's boundary conditions currently rely upon a first-order approximation: Each boundary cell's control point (*i.e.*, its node) is displaced from its volumetric centroid. Which (if any) variables inherit this error and whether or not the error propagates into the interior depend upon the BC implementation and mathematical character of the equations at the boundary. However, if this approximation were the sole source of error of less than second-order, asymptotic results indicative of first-order error should be expected. Another likely source of the error of less than second-order is the reduced accuracy of gradient calculations at the boundary (Section 6 discusses this issue in more detail). The gradient reconstruction options and

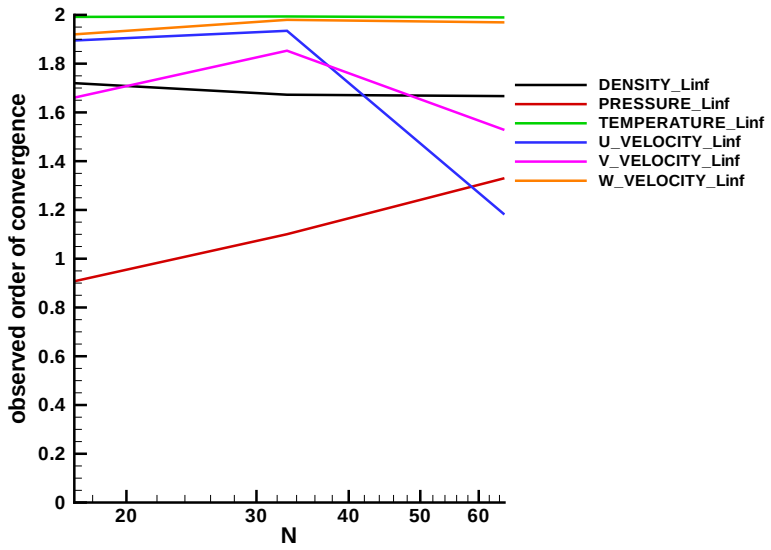


**Figure 7. Convergence order of $L_\infty$ norms for the outflow BC at transonic conditions.**

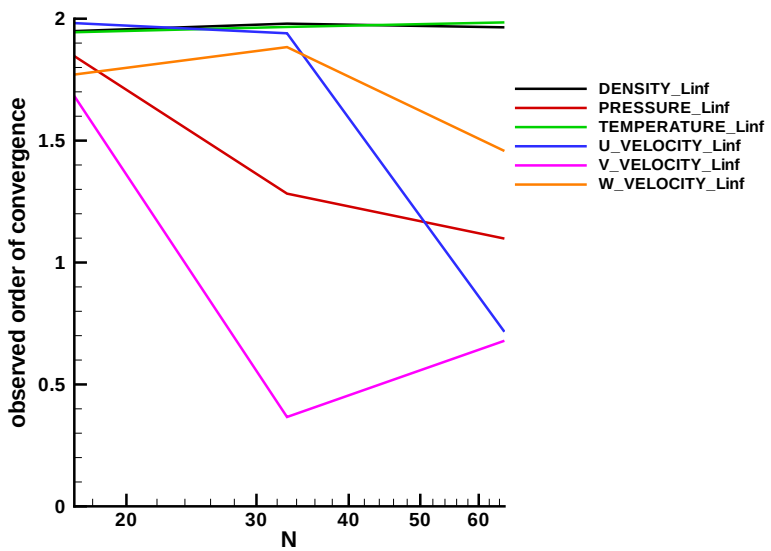


**Figure 8. Convergence order of $L_\infty$ norms for the outflow BC near stagnation conditions.**

BC's in Premo are both currently being overhauled, so these tests will be repeated once the new gradient reconstruction and BC's have been implemented.

Figure 9 shows the observed orders of accuracy of the $L_2$ norms of the errors in each of the flow variables. The observed orders of accuracy are not close enough to 2.0 to claim asymptotic second-order behavior, but no evidence of less than second-order error exists, either. This could indicate that the error of less than second-order evident in Figure 7 is first-order, and that it does not propagate into the interior of the domain. Alternatively, it could simply be the result of second-order error dominating over first-order or unordered error outside of the asymptotic regime.

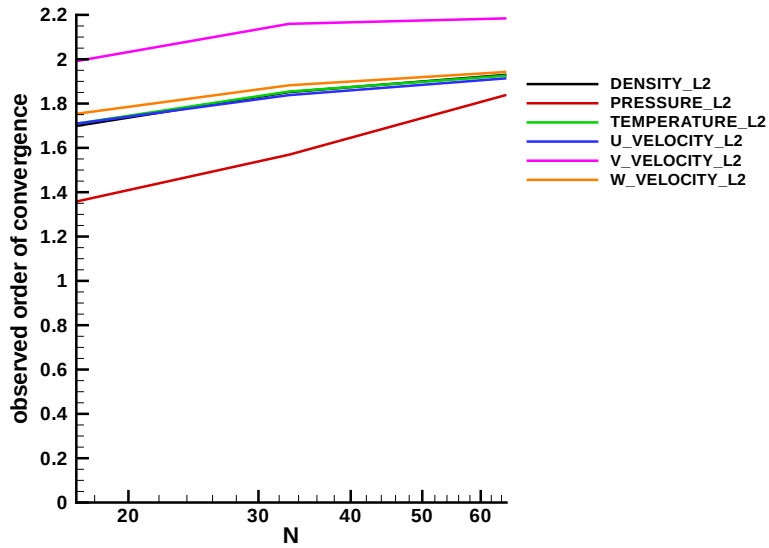


**Figure 9. Convergence order of $L_2$ norms for the outflow BC at transonic conditions.**

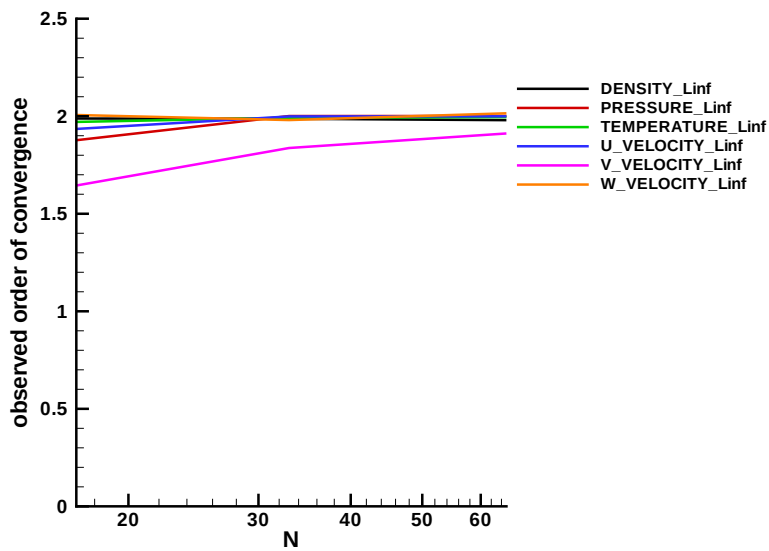


**Figure 10. Convergence order of $L_\infty$ norms for the outflow BC at fully supersonic conditions.**

Figure 10 shows the convergence order of the $L_\infty$ norms of the errors in $\rho$, $p$, $T$, $u$, $v$, and $w$ for the outflow BC at fully supersonic conditions. These results are far better than those in Figures 7 and 8 and show behavior approaching second-order. This indicates that the error of less than second-order evident in Figures 7 and 8 may be a result of unsatisfactory back pressure imposition in subsonic regions. The back pressure is imposed weakly by setting the "outside" state for the Riemann solver at the boundary from the user-specified $p$. Extrapolation of the Riemann invariants is used to determine the "outside" state values for $\rho$, $u$, $v$, and $w$. However, this weak implementation does not strictly enforce the $p$ value on

the boundary nodes.[l] This is a third possible source of the error of less than second-order (along with the previously mentioned boundary gradient inaccuracies and boundary cell centroid/node displacement) evident in Figures 7 and 8.

### 5.3. Further Testing of the No-Slip BC

Further testing of the no-slip BC shows $L_\infty$ results more similar to those in Figures 7 and 8 than those in Ref. 3. Figure 11 shows the observed order of accuracy of the $L_\infty$ norms of the errors of each of the

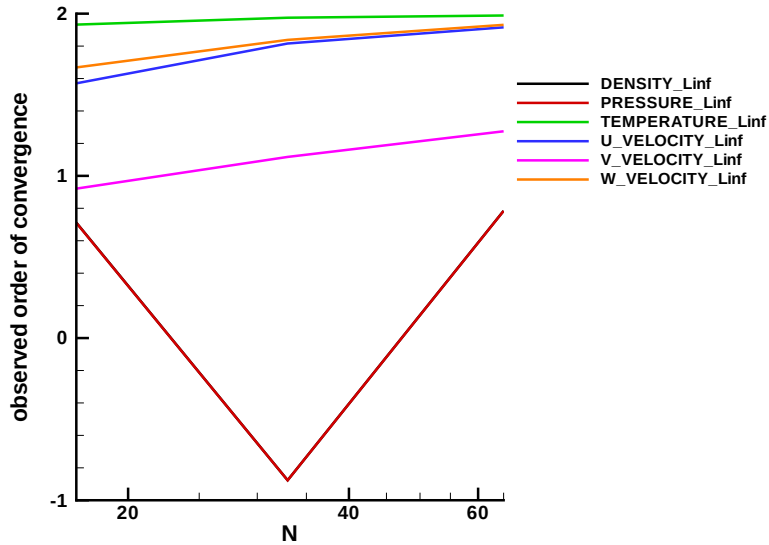


**Figure 11. Convergence order of $L_\infty$ norms for the adiabatic no-slip BC.**

flow variables for the adiabatic no-slip case. It is possible that the same type of error seen in Section 5.2 (bad gradient, centroid/node location discrepancy, or insufficient weak enforcement) is present in the no-slip results, even though it was not detectable using the solution and meshes described in Ref. 3. However, the $L_2$ and $L_1$ behavior of the adiabatic no-slip case is different from what was observed in the outflow cases. Figure 12 shows the observed order of accuracy of the $L_2$ norms of the errors of each of the flow variables for the adiabatic no-slip case. Error of less than second-order is evident in $L_2$ norms, indicating either that the BC generates first-order error that propagates into the domain, that the BC generates unordered error, or both.[m] Differences in error propagation between the no-slip and outflow BC's are not surprising, even if the source of the error is the same, since the mathematical character is different.

## 6. Testing of Gradient Calculations

Robustness issues with calculations of viscous boundary layers on hexahedral grids with high aspect ratios prompted an in-depth study of the gradient calculations. The questions raised were whether or not the gradient reconstruction options in Premo were performing at their desired orders of accuracy, whether or not accuracy was maintained for increasing curvature and aspect ratio, and what the magnitude of the gradient errors were. For all of these tests, the manufactured solution was given to Premo as an initial condition, and the numerical gradients of that solution were written to a file. Figure 13 shows the order of accuracy of the gradient calculations using inverse-distance-weighted least squares. Fifty-four curves (6 variables $\times$ 3 components $\times$ 3 norms) are on this plot, making it difficult to distinguish them, but the

[l]The "inside" Riemann state $p$ is taken to be the boundary node state and is unconstrained by the BC. The residual for the boundary node state's $p$ is formed from the flux balance of the boundary control volume.

[m]Limiting to these possibilities assumes that errors are of integer order, which may not be true in general, but it is believed to be true for the cases under consideration.

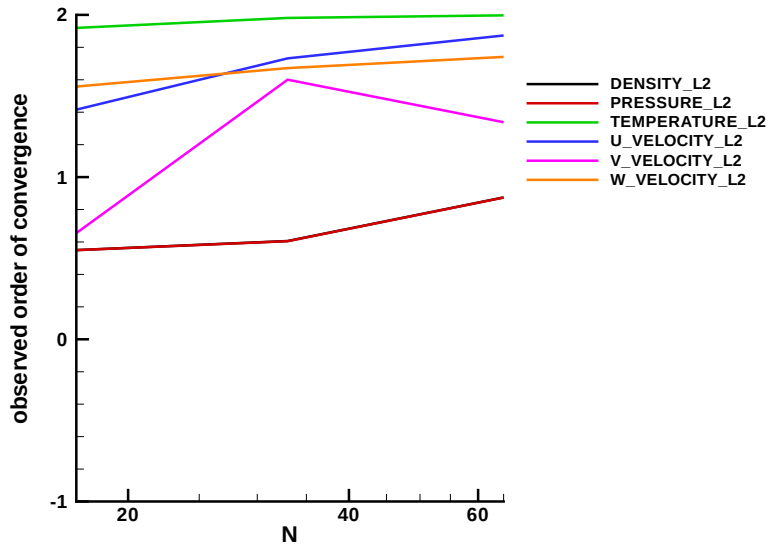


**Figure 12. Convergence order of $L_2$ norms for the adiabatic no-slip BC.**

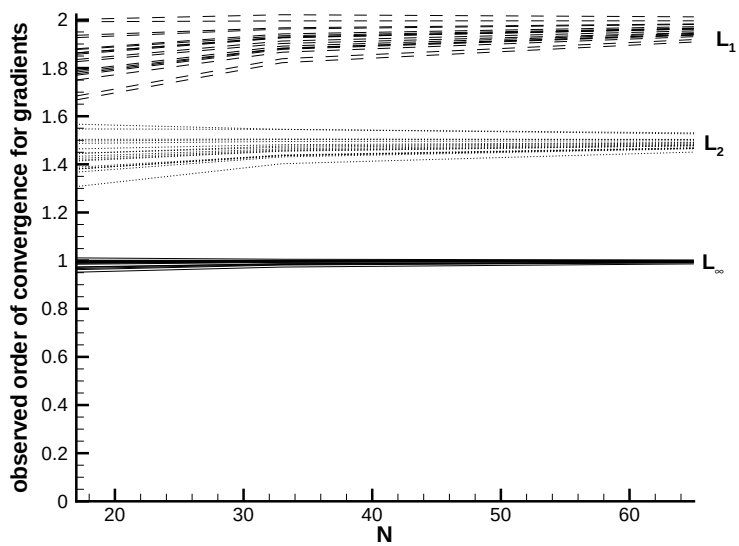


**Figure 13. Convergence order for the gradient calculation using inverse-distance-weighted least squares.**

intent is to demonstrate that all of the $L_1$ norms asymptotically approach 2.0; the $L_2$ norms, 1.5, and the $L_\infty$ norms, 1.0. This indicates that the interior gradient calculations are second-order and that the boundary gradient calculations are first-order.[n] Even though virtual edges[13] are used to support the least squares stencil at boundaries, the error is still first-order at all nodes on boundary faces, boundary edges,

[n]The 1.5 convergence order for the $L_2$ norms does not indicate order 1.5 error; rather, it is the result of how the weights of interior and boundary nodes sum as the mesh is refined.

and boundary corners.[o]

The results shown in Figure 13 are on the meshes described in Section 2. These properties hold as mesh curvature and aspect ratio are increased, although finer meshes are necessary to reach the asymptotic regime. The performance of the inverse-distance-weighted least squares gradient calculations was initially thought to be satisfactory, based on these order analyses. However, inspection of the magnitude of the error on the boundaries indicated serious problems, especially for boundary edge or boundary corner nodes. Even for the smooth manufactured solution at high resolution, the gradients were seen to have high relative error and occasionally even the wrong sign at boundaries. Even though the errors showed first-order behavior for boundary face nodes, boundary edge nodes, and boundary corner nodes, each of these node types showed increasingly higher magnitudes of error.[p] Even though virtual edges are used, the magnitude of the error becomes larger and larger as more dimensions are one-sided in the stencil.

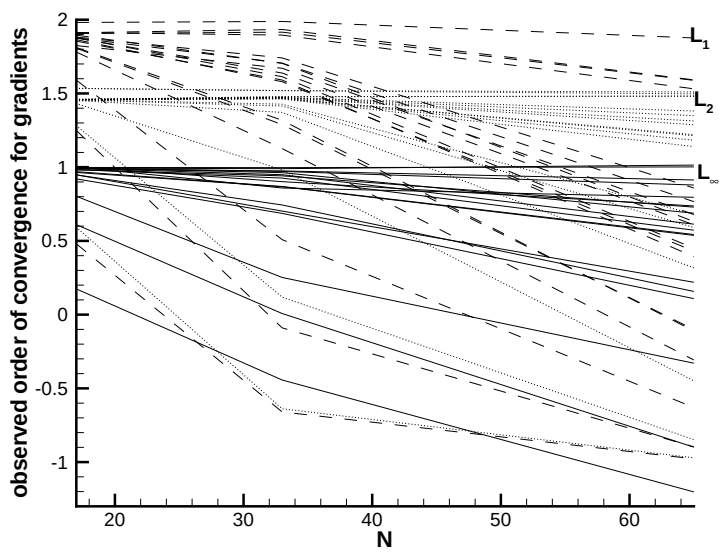


**Figure 14. Convergence order for the gradient calculation using an approximate form of Green-Gauss.**

Figure 14 shows the same fifty-four curves for the gradient calculations using a particular approximate form of Green-Gauss. Obviously, severe problems are indicated in the order behavior. Premo's implementation of Green-Gauss gradient calculations is only strictly valid for tetrahedral element types because it neglects some additional support necessary for non-Cartesian, hexahedral element types. Because of this approximation, poor order performance is not surprising; however, the severity of this problem (negative observed order is evident in some of the curves) even on smooth meshes was vastly underestimated. The use of this approximation to the Green-Gauss gradient calculation is believed to be the source of several of the anomalies documented in Ref. 3. Refs. 16, 17 and 18 compare the formulations of these gradient reconstruction schemes and their performance on more complex mesh topologies.

The gradient calculations in Premo are currently being revised in an attempt to provide increased accuracy and stability to viscous calculations and to alleviate accuracy and robustness issues with the BC's. The viscous calculations are particularly sensitive to the accuracy of gradient calculations, and gradient calculations of the wrong sign (which can happen at boundaries for both inverse-distance-weighted least squares and Green-Gauss) lead to instabilities, since temporal integration of the Navier-Stokes equations is unstable if the viscous fluxes carry a sign error.

[o]On a hexahedral mesh, interior nodes have 6 nearest neighbors; boundary face nodes, 5; boundary edge nodes, 4, and boundary corner nodes, 3.

[p]Boundary nodes with decreasing numbers of nearest neighbors become one-sided in increasing numbers of dimensions. Boundary face nodes are one-sided in 1 dimension; boundary edge nodes, 2, and boundary corner nodes, 3.

## 7. Demonstrating Asymptotic Behavior

Refs. 2, 3, and the current work show an evolution toward lower curvature manufactured solutions and smaller domains (in the physical space) in an effort to produce more convincingly asymptotic results. However, this evolution has demonstrated that results claimed to represent second-order behavior in Refs. 2 and 3 did not have sufficient resolution to uncover sources of error of less than second-order (*e.g.*, second-order accuracy was claimed for some cases where Premo's approximate implementation of Green-Gauss gradients was used).[q]

The authors have observed publications and presentations that are even worse—not even plotting or listing quantitative measurements for order of accuracy. Instead, a common practice is to plot error norms versus mesh size on a logarithmic scale and then compare them with a line (parallel, in the "eyeball" norm) representing a particular integer order of accuracy. Quantitative order of accuracy plots or tables are far more likely to highlight problems with a code's order behavior. Another questionable practice is to quote only $L_1$ or $L_2$ norms, even though the $L_\infty$ norms are much more sensitive to problems.

As the manufactured solution and meshes evolved from those in Refs. 2 and 3 to the current work, the relative errors of some of the variables began to approach the single precision limit (approximately $1 \times 10^{-7}$) and level off. Rather than stemming from a coding mistake in Premo, this problem was a result of the coupling between the compiler used for the user-defined routines and the symbolic manipulation software used to generate the portions of those routines for calculating the MMS source terms.[r] Once this problem was identified, a remedy allowed the MMS tests to be performed in double precision. Although no coding mistake regarding the handling of double precision floating point values was actually present in the Premo source, if there had been one, it would never have been detected if the mesh resolution of the MMS tests was not sufficient to allow relative errors of the flow variables to be below the single precision limit.

All of the cases presented in the present work which show behavior approaching or in the vicinity of second-order have relative errors in the flow variables smaller than the single precision limit, at least on the finest mesh. The authors now use this as an additional constraint for claiming that a code has been verified to a given order and recommend this requirement for use by others as well. Otherwise, the claim that a code is double precision has not been substantiated, and any claims to a given order of accuracy are suspect for high values of relative error, even if the computed order of accuracy appears to be asymptotically approaching a given value (*e.g.*, the Green-Gauss results of Ref. 2). This is especially important when the $L_\infty$ norm is used, since it may not monotonically approach the observed order or get within a small tolerance of an integer-valued observed order except on extremely fine meshes.[s]

## 8. Conclusions

A three-dimensional manufactured solution has been derived that is capable of testing the Euler, Navier-Stokes, and RANS equations on non-uniform meshes along with the slip and no-slip (adiabatic and isothermal) conditions and outflow (subsonic, supersonic, and mixed) boundary. The manufactured solution has been shown useful in verifying the order of accuracy for the interior equation sets in Premo, as well as identifying coding mistakes that prevented initial results from confirming the formal order of accuracy in some cases. The manufactured solution has also proved useful in identifying or clarifying a number of issues with BC's and gradient reconstruction methods, and these portions of Premo are currently being re-designed in an effort to improve accuracy and robustness.

The derived manufactured solution is applicable not only to Premo, but also to general CFD codes. Additionally, the methodology for deriving manufactured solutions for specialized boundary conditions used

[q]Refs. 9 and 10 did not suffer from the Green-Gauss gradient reconstruction issues because they used only Cartesian meshes (for which Premo's approximate implementation of Green-Gauss is valid) and did not claim to have verified the code to be second-order for skewed, non-uniform, hexahedral meshes (as was done in Ref. 2).

[r]Mathematica®was used as the symbolic manipulation software to calculate the source terms and write them out as FORTRAN code. This FORTRAN code was re-formatted, then compiled with g77 before linking with the remainder of the C++ Premo source. However, Mathematica®writes constants out in single precision format (*e.g.*, 1.0) rather than double precision format (*e.g.*, 1.0d0), and the typecasting in g77 does not fill the four least significant bytes of a double precision variable with zeros when a single precision constant is used in an assignment statement. This resulted in a single precision error in the user-defined routine for calculating the MMS source, even though the C++ Premo source was accurate to double precision.

[s]The sometimes erratic behavior of the $L_\infty$ norm is a result of the location of the maximum error moving around as the mesh is refined. This makes it an unpopular choice for order verification, but it is preferred by the authors for the reasons discussed in Section 4.1.2.

in CFD codes has been shown useful, and a similar approach could be used to derive a manufactured solution for other BC's (*e.g.*, inflow boundaries). Unlike the versions of the manufactured solution and sequence of meshes given in Refs. 2 and 3, the current manufactured solution and sequence of meshes are mature enough for other code developers to begin using them for order-of-accuracy verification. It is hoped that such use will not only allow other code developers to find and correct coding mistakes in their own software, but also evaluate different formulations of BC's. Formal order-of-accuracy verification is often heralded for its usefulness in identifying coding mistakes, but its ability to identify weaknesses in numerical formulations and algorithms themselves is as, if not more, significant. The generality provided by using manufactured solutions for formal order-of-accuracy verification allows very rigorous testing of BC formulations. Many BC formulations found in the literature contain approximations of unspecified order of accuracy that are not obvious or well documented, and manufactured solutions derived specifically to evaluate BC's can be used to eliminate some formulations from consideration in the code design process.

## 9. Acknowledgments

The authors would like to thank Alfred Lorber and Thomas Smith for their help in implementing the manufactured solutions in Premo.

## References

[1]Smith, T. M., Ober, C. C., and Lorber, A. A., "SIERRA/Premo—An Unstructured Mesh Compressible Flow Simulation Code," AIAA Paper 2002-3292, June 2002.

[2]Bond, R. B., Knupp, P. M., and Ober, C. C., "A Manufactured Solution for Verifying CFD Boundary Conditions," AIAA Paper 2004-2629, June 2004.

[3]Bond, R. B., Knupp, P. M., and Ober, C. C., "A Manufactured Solution for Verifying CFD Boundary Conditions, Part II," AIAA Paper 2005-0088, January 2005.

[4]Edwards, H. C. and Stewart, J. R., "SIERRA : A Software Environment for Developing Complex Multiphysics Applications", *Proceedings of First MIT Conference on Computational Fluid and Solid Mechanics*, Elsevier Scientific, K. J. Bathe, ed., June 2001.

[5]Knupp, P. and Salari, K., *Verification of Computer Codes in Computational Science and Engineering*, Chapman and Hall / CRC, Boca Raton, FL, 2003.

[6]Roache, P. J., *Verification and Validation in Computational Science and Engineering*, Hermosa, Albuquerque, NM, 1998.

[7]Spalart P. R., and Allmaras, S. R., "A One-Equation Turbulence Model for Aerodynamic Flows," *La Recherche Aerospatiale*, No. 1, 1994, pp. 5–21.

[8]Strauss, W. A., *Partial Differential Equations: an Introduction*, John Wiley and Sons, Inc., New York, 1992, pp. 82–86.

[9]Roy, C. J., Smith, T. M., and Ober, C. C., "Verification of a Compressible CFD Code using the Method of Manufactured Solutions," AIAA Paper 2002-3110, June 2002.

[10]Roy, C. J., Nelson, C. C., Smith, T. M., and Ober, C. C., "Verification of Euler/Navier-Stokes codes using the method of manufactured solutions," *International Journal of Numerical Methods in Fluids*, Vol. 44, 2004, pp. 599–620.

[11]Yee, H. C., "Construction of Explicit and Implicit Symmetric TVD Schemes and Their Applications," *Journal of Computational Physics*, Vol. 49, No. 2, January 1987, pp. 357–393.

[12]Roe, P. L., "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," *Journal of Computational Physics*, Vol. 43, No. 2, October 1981, pp. 357–372.

[13]Haselbacher, A., and Blazek, J., "On the Accurate and Efficient Discretisation of the Navier-Stokes Equations on Mixed Grids," AIAA Paper 99-3363, 1999, and *AIAA Journal*, Vol. 38, 2000, pp. 2094–2102.

[14]Roy, C. J., "Grid Convergence Error Analysis for Mixed-Order Numerical Schemes," *AIAA Journal*, Vol. 41, 2003, pp. 595–604.

[15]Gustafsson, B., "The Convergence Rate for Difference Approximations to Mixed Initial Boundary Value Problems," *Mathematics of Computation* Vol. 29, 1975, pp. 396–406.

[16]Mavriplis, D. J., "Revisiting the Least-Squares Procedure for Gradient Reconstruction on Unstructured Meshes," AIAA Paper 2003-3986, June 2003.

[17]Aftosmis, M., Gaitonde, D., and Tavares, T. S., "Behavior of Linear Reconstruction Techniques on Unstructured Meshes," *AIAA Journal*, Vol. 33, No. 11, November 1995, pp. 2038–2049.

[18]Barth, T. J. and Jesperson, D. C., "The Design and Application of Upwind Schemes on Unstructured Meshes," AIAA Paper 89-0366, January 1989.