43rd AIAA Aerospace Sciences Meeting and Exhibit, 10-13 January, 2005, Reno, NV

A Manufactured Solution for Verifying CFD Boundary Conditions, Part II

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

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 to verify the order of accuracy of a code is the method of manufactured solutions. In this study, a manufactured solution has been derived and implemented that allows verification of not only the Euler, Navier-Stokes, and Reynolds-Averaged Navier-Stokes (RANS) equation sets, but also some of their associated boundary conditions (BC's): slip, no-slip (adiabatic and isothermal), and outflow (subsonic, supersonic, and mixed). Order-of-accuracy verification has been performed for the Euler and Navier-Stokes equations and these BC's in a compressible computational fluid dynamics code. All of the results shown are on skewed, non-uniform meshes. RANS results will be presented in a future paper. The observed order of accuracy was lower than the expected order of accuracy in two cases. One of these cases resulted in the identification and correction of a coding mistake in the CHAD gradient correction that was reducing the observed order of accuracy. This mistake would have been undetectable on a Cartesian mesh. During the search for the CHAD gradient correction problem, an unrelated coding mistake was found and corrected. The other case in which the observed order of accuracy was less than expected was a test of the slip BC; although no specific coding or formulation mistakes have yet been identified. After the correction of the identified coding mistakes, all of the aforementioned equation sets and BC's demonstrated the expected (or at least acceptable) order of accuracy except the slip condition.

1. Introduction

The purpose of this work is to continue 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, we apply it to the verification of Premo,[2] a compressible fluid dynamics code developed at Sandia National Laboratories. Thus, the results in this paper are twofold: first, a manufactured solution for verification of CFD codes and, second, a verification study of Premo using this manufactured solution.

Ref. 1 contains an earlier version of the manufactured solution used in the present work, a derivation of this manufactured solution, and results of tests on the Euler equations and some of the related BC's. The slip condition and the Euler equations using least squares gradient reconstruction failed to verify to the expected order. After Ref. 1 was published, two coding mistakes were identified. The mistake that resulted

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

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

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

§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.

in the poor slip results involved the misuse of a unit normal calculation function intended for planar surfaces only. The mistake that resulted in the poor least squares results involved an indexing mistake in the gradient calculation function. (Ref. 1 draws some incorrect conclusions regarding the least squares results. This will be discussed in detail in Section 7.1.) These problems were corrected, and the present work re-tests these features and extends to options not covered in the previous work.

Although the implementation of the MMS is done for Premo, the derivation respects the needs of general CFD codes and should be useful for their verification. Premo is a compressible flow code used to determine aerodynamic performance of arbitrary bodies, and it is built within the SIERRA framework[3] to allow multi-physics coupling (*e.g.*, ablation or aeroelasticity). It is a parallel, unstructured, edge-based, finite-volume code.

The method of manufactured solutions (MMS) provides order-of-accuracy verification of the implemented equation sets along with their various BC's.[4,5] Such verification is a rigorous way to ensure that the governing equations are correctly solved and thus builds confidence in the user community. MMS derives its name from a solution which is manufactured for the purpose of verifying the code. It is similar to 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 no exact solutions exist or where exact solutions are not sufficiently general to fully test a code. 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 the 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 lines for calculating the source term are added to the code, it can be used to solve the equation set with the additional source term. (The mechanism for this in Premo is a user-defined source routine.) 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 expected order of accuracy, or it will not. In the latter case it may mean there is a coding mistake. Other possibilities are discussed in Ref. 4.

The Reynolds-Averaged Navier-Stokes (RANS) equations are given as

$$\frac{\partial(\overline{\rho})}{\partial t} + \frac{\partial(\overline{\rho}\widetilde{u_j})}{\partial x_j} = 0 \tag{1}$$

$$\frac{\partial(\overline{\rho}\widetilde{u_i})}{\partial t} + \frac{\partial(\overline{\rho}\widetilde{u_i}\widetilde{u_j} + \overline{p}\delta_{ij} - \overline{\tau_{ij}})}{\partial x_j} = 0 \tag{2}$$

$$\frac{\partial(\overline{\rho}\widetilde{e_t})}{\partial t} + \frac{\partial(\overline{\rho}\widetilde{u_j}\widetilde{e_t} + \overline{p}\widetilde{u_j} - \widetilde{u_i}\overline{\tau_{ij}} + \overline{q_j})}{\partial x_j} = 0, \tag{3}$$

where the bar over a symbol indicates Reynolds averaging and the tilde over a symbol indicates Favre averaging. These five transport equations are closed by auxiliary relations for thermodynamic and transport properties. The definitions for internal energy (e) and total energy (e_t) are given as

$$e = \frac{1}{\gamma - 1} RT \tag{4}$$

$$e_t = e + \frac{u_i u_i}{2}. \tag{5}$$

The ideal gas law is used for the equation of state:

$$p = \rho RT. \tag{6}$$

The above relations reference the ratio of specific heats (γ) and the specific gas constant (R). For air, these quantities are 1.4 and 287.0 $\frac{\text{N m}}{\text{kg K}}$, respectively.

The viscous stress tensor and heat flux vector are given by:

$$\overline{\tau_{ij}} = (\overline{\mu} + \mu_t)\left[\left(\frac{\partial \widetilde{u_i}}{\partial x_j} + \frac{\partial \widetilde{u_j}}{\partial x_i}\right) - \frac{2}{3}\delta_{ij}\frac{\partial \widetilde{u_k}}{\partial x_k}\right] \tag{7}$$

$$\overline{q_i} = -(\overline{k} + k_t)\frac{\partial \overline{T}}{\partial x_i}. \tag{8}$$

The molecular viscosity (μ) is given by Sutherland's law:

$$\frac{\mu}{\mu_0} = \left(\frac{T}{T_0}\right)^{\frac{3}{2}} \frac{T_0 + S}{T + S}; \tag{9}$$

while the eddy viscosity (μ_t) is determined by the choice of turbulence model. The molecular and turbulent thermal conductivities are determined by assuming a constant Prandtl number (Pr) and turbulent Prandtl number (Pr_t):

$$k = \frac{\gamma R}{\gamma - 1}\frac{\mu}{Pr} \tag{10}$$

$$k_t = \frac{\gamma R}{\gamma - 1}\frac{\mu_t}{Pr_t}. \tag{11}$$

For this work, the Spalart-Allmaras turbulence model[6] was used. In compressible form, the single transport equation for the model can be expressed as:

$$\frac{\partial(\overline{\rho}\widetilde{\nu})}{\partial t} + \frac{\partial(\overline{\rho}\widetilde{u_j}\widetilde{\nu})}{\partial x_j} = D + S_P - S_D, \tag{12}$$

where

$$S_P = c_{b1}\overline{\rho}\widetilde{\nu}\widetilde{S} \tag{13}$$

$$S_D = c_{\omega 1} f_\omega \overline{\rho}\left(\frac{\widetilde{\nu}}{d}\right)^2 \tag{14}$$

$$D = \frac{1}{\sigma}\frac{\partial}{\partial x_j}\left[(\mu + \overline{\rho}\widetilde{\nu})\frac{\partial \widetilde{\nu}}{\partial x_j}\right] + \frac{c_{b2}\overline{\rho}}{\sigma}\frac{\partial \widetilde{\nu}}{\partial x_j}\frac{\partial \widetilde{\nu}}{\partial x_j}. \tag{15}$$

The closure coefficients and auxiliary relations are given by

$$c_{b1} = 0.1355$$

$$c_{b2} = 0.622$$

$$c_{\nu 1} = 7.1$$

$$\sigma = 2/3$$

$$c_{\omega 1} = \frac{c_{b1}}{\kappa^2} + \frac{1 + c_{b2}}{\sigma}$$

$$c_{\omega 2} = 0.3$$

$$c_{\omega 3} = 2$$

$$\kappa = 0.41$$

$$f_{\nu 1} = \frac{\chi^3}{\chi^3 + c_{\nu 1}^3}$$

$$f_{\nu 2} = 1 - \frac{\chi}{1 + \chi f_{\nu 1}}$$

$$f_{\omega} = g \left[\frac{1 + c_{\omega 3}^6}{g^6 + c_{\omega 3}^2} \right]^{\frac{1}{6}}$$

$$\chi = \frac{\widetilde{\nu}}{\nu}$$

$$g = r + c_{\omega 2} \left(r^6 - r \right)$$

$$r = \frac{\widetilde{\nu}}{\widetilde{S} \kappa^2 d^2}$$

$$\widetilde{S} = \sqrt{2 \Omega_{ij} \Omega_{ij}} + \frac{\widetilde{\nu}}{\kappa^2 d^2} f_{\nu 2}.$$

The eddy viscosity is given by

$$\mu_t = \overline{\rho} \widetilde{\nu} f_{\nu 1}. \tag{16}$$

(Note that $\widetilde{\nu}$ is a working variable, not the Favre averaged kinematic viscosity.)

To manufacture a solution to Eq. (1), (2), (3), (12) and their auxiliary relations, analytic forms of six variables must be defined. These six variables must be independent of one another (*e.g.*, this variable set can not contain ρ, u, and ρu because the third is simply a product of the first two). The most convenient set of variables (whether conservative, primitive, characteristic, or a combination of the three) may depend upon the BC being tested. For this work, the variables p, T, u, v, w, and $\widetilde{\nu}$ were used. This manufactured solution may also be used for the Navier-Stokes set or the Euler set by simply dropping the extraneous turbulence variable.

A general form for a steady manufactured solution of a flow variable ϕ is

$$\phi = \phi_0 \left(1 + f(Ax) f(By) f(Cz) \right), \tag{17}$$

where A, B, and C are constants. The function f must be chosen so that this form satisfies the requirements of Knupp and Salari.[4] To avoid symmetry within the solution, $A \neq B \neq C \neq 0$. If ϕ_0 is a constant, this form is a separated solution plus a constant.[7] The separated form makes deriving manufactured solutions with specific Neumann BC's (such as a vanishing normal derivative) easier. Constant values for ϕ_0 will not be sufficient for all of the BC's of interest, but if ϕ_0 is chosen appropriately, the necessary properties will hold.

2. What Defines a Boundary Condition?

For hyperbolic or mixed character equations, flow of information through the domain and its associated boundaries is restricted by the local mathematical character (*i.e.*, whether the equations are hyperbolic, parabolic, or elliptic at a given point, and if they are hyperbolic, the direction of the characteristics). The concept of information, or small perturbations, propagating into or out of the domain is distinct from the flux of mass, momentum, or energy through the boundary. The eigenvalues of a system of hyperbolic or mixed character equations determine the directions in which the information will propagate. For the 3-D Euler equations, there are five eigenvalues which dictate the flow of information at a bounding surface: one is the velocity normal to the boundary plus the speed of sound, $v_n + a$; one is the normal velocity minus the speed of sound, $v_n - a$, and the other three are the normal velocity, v_n (*i.e.*, this eigenvalue is repeated three times).

Conditions that are enforced on domain boundaries are frequently referred to as BC's in the engineering literature, but some do not satisfy the mathematical definition of a BC. A good example is $\frac{\partial p}{\partial n} = \frac{\rho v_t^2}{R}$ which, although it is enforced on a slip boundary, does not provide any additional constraint to the governing equation and, therefore, is not a mathematical BC. (See Section 3.1.1 for a more thorough explanation of this equation and the slip condition.)

The number of mathematical BC's which need to be enforced at a given boundary segment in order to determine a unique solution is equal to the number of negative eigenvalues. For the Euler equations at a supersonic outflow boundary, there are no negative eigenvalues, so no mathematical BC's need to be enforced on supersonic segments of the boundary. For a subsonic outflow or slip boundary, there is one negative eigenvalue; for a subsonic inflow boundary, there are are four, and for a supersonic inflow, there are five.

Consider a homogeneous differential equation

$$\mathcal{D}\left(u\right) = 0 \tag{18}$$

on a domain Ω and its associated BC's on the boundary of that domain, $\partial\Omega$. (The following discussion is simplified by the assumption of starting with a homogeneous equation, but the conclusions apply equally to a non-homogeneous equation.) For a given manufactured solution, u^*, the source term, S^* generated by operation of the differential operator, $\mathcal{D}$, upon u^* is used to create a new equation which the manufactured solution satisfies

$$\mathcal{D}\left(u^*\right) = S^*. \tag{19}$$

On the interior of Ω, S^* corrects for the failure of u^* to satisfy the homogeneous equation. Since $\mathcal{D}(u) = 0$ is also enforced on $\partial\Omega$, S^* will also correct for the discrepancy between $\mathcal{D}(u)$ and $\mathcal{D}(u^*)$ on $\partial\Omega$. However, S^* does not correct any discrepancy between u^* and the mathematical BC's (the number of which is equal to the number of negative eigenvalues). Thus if a mathematical BC, $\mathcal{B}(u) = 0$, is enforced on some portion of $\partial\Omega$, then a boundary source, $Q^* = \mathcal{B}(u^*)$, can be added to correct for the discrepancy between $\mathcal{B}(u)$ and $\mathcal{B}(u^*)$. Alternatively, the manufactured solution can be designed so that $\mathcal{B}(u^*) = 0$ on the relevant portions of the boundary.

If the interior equations implemented in a code are to be tested using the manufactured solution, then Dirichlet BC's can be set so that the BC's enforced by the code are equal to u^* on $\partial\Omega$. A similar setup can be used for Neumann BC's, provided that the implementation allows for spatially varying (and, if appropriate, temporally varying) values of the partial derivatives of u^* on $\partial\Omega$. However, if a combination of the interior equations and a specialized BC is to be verified using the method of manufactured solutions, then u^* must satisfy the given specialized BC on the portion of $\partial\Omega$ where it is to be enforced.[a] Note that only the BC's by the mathematical definition (constraints corresponding to negative eigenvalues) must be satisfied by the manufactured solution.[b] In many codes, the interior equations and BC's are discretized to different orders of accuracy. In this case, the observed order will be the minimum of the order of the interior equations and the BC's.

3. Solid Surface Boundaries

For testing a code's capability to handle a solid surface boundary, a surface must first be defined. Let $F = F(x, y, z)$ be a function from $\Re^3$ into $\Re$ such that $F = C$ defines a surface for any constant C. The unit

[a]The alternative approach of adding the boundary source, Q^*, where necessary requires additional generality on the part of the code. If the code does not have the capability for introducing Q^*, then enforcing $\mathcal{B}(u^*) = 0$ is the only option. So that the manufactured solution can be used with codes with or without this capability, the $\mathcal{B}(u^*) = 0$ approach was chosen for this work.

[b]There is also the possibility of a code-imposed constraint on a boundary that corresponds to neither the mathematical BC's or the influence of the interior equations. This situation raises numerous code-specific issues for verification that are not discussed here.

normal is defined by

$$\hat{n} = \frac{\nabla F}{\|\nabla F\|}. \tag{20}$$

A manufactured solution can be constructed so that any variable ϕ satisfies either a Dirichlet (*i.e.*, ϕ constrained) or Neumann (*i.e.*, $\frac{\partial \phi}{\partial n}$ constrained) BC along the surface. So that Eq. (20) will represent the outward facing normal, the domain of interest is taken to be on the side $F - C < 0$.

3.1. Conditions on Velocity

The two solid surface conditions discussed in this work are the slip condition ($\vec{v} \cdot \hat{n} = 0$) and the no-slip condition ($\vec{v} = 0$). For a no-slip boundary on a domain where the Navier-Stokes equations are being solved, an additional constraint must be placed upon either the temperature (*i.e.*, Dirichlet BC specifying T) or the heat flux (*i.e.*, Neumann BC specifying $\frac{\partial T}{\partial n}$).

3.1.1. The Slip Condition

The slip condition arises from the impermeability of the solid surface. It states that nothing can be convected through the surface. This can be expressed as $v_n = 0$, where v_n is the velocity component normal to the surface, or equivalently,

$$\vec{\mathcal{F}} \cdot \hat{n} = \begin{bmatrix} 0 \\ p\hat{n} \\ 0 \end{bmatrix}, \tag{21}$$

where $\vec{\mathcal{F}}$ is the flux hypervector and $\hat{n}$ is the unit normal vector of the surface. For an Euler flow, this condition is the actual mathematical BC. The oft quoted $\frac{\partial p}{\partial n} = \frac{\rho v_t^2}{R}$ (where v_t is the velocity component tangent to the surface and R is the surface radius of curvature in the streamwise direction) is a consequence of the governing equations (in this case, the normal momentum equation). Because of this, it is not necessary for the manufactured solution to satisfy any constraint on the pressure (or any variable other than v_n).

For any surface defined by the equation $F = C_{\mathrm{s}}$, impermeability implies that

$$\nabla F \cdot \vec{v} = F_x u + F_y v + F_z w = 0. \tag{22}$$

If analytic forms are chosen for u and v in the definition of the manufactured solution, then w can be defined as follows to ensure that the slip condition holds on any F-constant surface:

$$w = -\frac{F_x u + F_y v}{F_z}. \tag{23}$$

This imposes an additional constraint on the function F: $F_z \neq 0; \forall (x, y, z) \in \Re^3$. This can be easily enforced by defining $F(x, y, z) = \widetilde{F}(x, y) - z$. Thus, the surface $F = C_{\mathrm{s}}$ can be expressed as $z = \widetilde{F}(x, y) - C_{\mathrm{s}}$. With this surface definition, u and v can be left in the general form of Eq. (17) with no additional constraints.

3.1.2. The No-Slip Condition

The no-slip condition is actually slightly simpler. The following equation for u satisfies $u = 0$ on the surface $F = C_{\text{n-s}}$:

$$u = (C_{\text{n-s}} - F) u_0 \left[1 + f(Ax) f(By) f(Cz)\right]. \tag{24}$$

If v and w are of the same form as u, then the manufactured solution satisfies the no-slip condition for velocity along $F = C_{\text{n-s}}$. If, however, w is of the form of Eq. (23), then the manufactured solution will satisfy the no-slip condition on $F = C_{\text{n-s}}$ as well as the slip condition on any $F = C_{\mathrm{s}}$ where $C_{\mathrm{s}} \neq C_{\text{n-s}}$. This is convenient since it allows the same manufactured solution to be used for testing the slip and no-slip conditions.

3.2. Additional Boundary Condition for the Navier-Stokes Equation Set

Because of the presence of thermal diffusion, the Navier-Stokes equation set requires the imposition of one additional BC at a no-slip surface. This BC is most often either a Dirichlet BC (*i.e.*, T specified) or a Neumann BC (*i.e.*, $\frac{\partial T}{\partial n}$ specified). The Neumann condition can also be expressed by specifying the heat flux. If Dirichlet and Neumann BC's are implemented in the code, and they are written very generally (*i.e.*, as functions of spatial and temporal variables), then the manufactured solution choice for temperature does not need to satisfy special requirements at the boundary. In this case, the code can just be given the BC that the manufactured solution already satisfies. It can be expressed as either a Dirichlet or Neumann BC. However, if the code has a special case, such as uniform temperature ($T = T_c$) or adiabatic boundary ($\frac{\partial T}{\partial n} = 0$), then the manufactured solution must satisfy the special case in order to verify its implementation.

3.2.1. Adiabatic Boundary

The temperature at an adiabatic ($\frac{\partial T}{\partial n} = 0$) surface is subject to a Neumann BC. This constraint defines a partial differential equation (PDE) for T:

$$\nabla T \cdot \nabla F = F_x T_x + F_y T_y + F_z T_z = 0. \tag{25}$$

A manufactured solution to T which satisfies this PDE can be used for adiabatic tests. One such solution is given by Eq. (27) in Section 3.2.3.

3.2.2. Isothermal Boundary

If an isothermal (*i.e.*, constant temperature) BC is implemented as a strict Dirichlet BC, then verification by the method of manufactured solutions is probably unnecessary. Simple inspection of the output temperature solution at the isothermal boundary should indicate whether or not the condition is applied correctly. However, in the event that the constant temperature is maintained indirectly through the flux or residual vector formulations, some formal verification is necessary.

The isothermal boundary is formulated along a surface $F = C_{\mathrm{i}}$ similar to the way that the no-slip condition on the velocity was formulated. In this case, it is necessary to ensure that $T = T_c$ at $F = C_{\mathrm{i}}$ and that T has sufficient spatial variation away from the surface. This can be done with the following equation:

$$T = T_c + (C_{\mathrm{i}} - F)\, T_0 \left(1 + f(Ax) f(By) f(Cz)\right). \tag{26}$$

3.2.3. Adiabatic and Isothermal

Because much of the work in verification by the method of manufactured solutions is taken up in calculating the source term and incorporating it into the code, the amount of work can be reduced if the manufactured solution is designed to test multiple BC's. If the constraints mentioned in the previous sections are combined, the following solution is obtained:

$$T = T_c \left(1 + g(x,y,z) \left[C_{\mathrm{a/i}} - F\right]^n\right), \tag{27}$$

where $n > 1$ and $g(x,y,z) \geq 0$. Note that this is a departure from the general form in Eq. (17). The term $\left[C_{\mathrm{a/i}} - F\right]^n$ ensures that $T = T_c$ and $\frac{\partial T}{\partial n} = 0$ on $F = C_{\mathrm{a/i}}$. The function $g(x,y,z)$ is necessary in the event that the function $F(x,y,z)$ does not have enough nonzero partial derivatives to exercise all of the terms in the equations. For example, if $F(x,y,z) = \frac{1}{2}\cos(2x)\cos(3y) - z$, then g must be a function of z so that the leading error terms for the discretization of $\frac{\partial^2 T}{\partial z^2}$ are nontrivial. If $g(z) = 1 + \frac{1}{4}\sin(z)$, this condition is satisfied. Alternatively in this case, n could simply be increased until T has sufficient variance in z, but that has the effect of uniformly raising T, making it more difficult to locate supersonic regions to test supersonic inflow and outflow conditions. Also, raising n would increase solution gradient components in all

three directions, whereas the addition of $g(z)$ only increases the gradient component in the z direction. High solution gradients require higher mesh resolution (*i.e.*, bigger meshes and more expensive computations) to reach the asymptotic regime.

3.3. Turbulence Variable(s) for the RANS Equation Set

For RANS cases, turbulence variable(s) also need to be defined at a solid surface. For the Spalart-Allmaras model, the following equation can be used

$$\widetilde{\nu} = (C_{\text{n-s}} - F)\,\widetilde{\nu}_0\,(1 + f(Ax)f(By)f(Cy))\,. \tag{28}$$

For any of the two-equation models (*e.g.*, k-ϵ or k-ω), this same formula can be used for k. However, the definition of the other turbulence variable (ϵ or ω) may not be zero at the boundary, so a different form must be used. Something similar to Eq. (26) would be sufficient.

4. Inflow and Outflow Conditions for the Euler Equations

For the Euler equations, the number of incoming and outgoing characteristics at any given inflow or outflow surface is determined by the signs of the eigenvalues. Since the manufactured solution needs to match the BC's, the eigenvalues of the operator $\mathcal{D}$ on $\partial\Omega$ must conform to the signs dictating which BC's are matched (*e.g.*, for an inflow surface, $v_n < 0$, and for an outflow surface, $v_n > 0$). Otherwise, the boundary can be over- or under-constrained by the BC's imposed by the code. Each of these surfaces may be either subsonic ($M_n < 1$), supersonic ($M_n > 1$), or a mixture of the two. For example, if a surface is specified as an outflow surface, a Dirichlet condition on the pressure will be enforced on subsonic portions of the boundary with the remainder of the variables determined by the interior equations. All of the flow variables will be determined by the interior equations on a supersonic portion of the boundary. However, if $v_n < 0$ anywhere on this surface, then the necessary constraints arising from the three additional negative eigenvalues will be missing, and the problem will be ill-posed.

Since the previously defined function F used for defining the slip and no-slip surfaces can not also be used for defining an inflow or outflow surface, a new function must be considered. Let $G = G(x, y, z)$ be a function from $\Re^3$ into $\Re$ such that $G = C$ defines a surface for any constant C. The definition of this function and the value of C are both arbitrary. Once G has been defined, it can be used in the manufactured solution to ensure that given inflow and outflow conditions hold on G-constant surfaces.

4.1. Subsonic Outflow Condition

Since the previously derived manufactured solutions for u, v, w, and T leave only one degree of freedom for an Euler solution, an inflow condition (having four or five incoming characteristics, depending on the Mach number) can not be tested. However, an outflow condition (having one or no incoming characteristics, depending on the Mach number) can be tested. The pressure profile can be fixed so that it is constant along a general surface $G(x, y, z) = C_{\text{o}}$. This condition is similar to the isothermal condition at the solid boundary, so an equation for pressure similar to Eq. (26) can be used.

$$p = p_c + (C_{\text{o}} - G)\,p_0\,(1 + f(Ax)f(By)f(Cz)) \tag{29}$$

It is also necessary that $v_n > 0$ and $M_n < 1$. However, these constraints are not very restrictive, so they can be combined with the conditions used to define earlier versions of the velocity profiles. Since the solution is intended to be used for testing all previously mentioned BC's in addition to the outflow condition, w is still defined by Eq. (23). The resulting normal velocity is

$$v_n = \frac{\vec{v}\cdot\nabla G}{\|\nabla G\|} = \frac{\left(G_x - G_z\frac{F_x}{F_z}\right)u + \left(G_y - G_z\frac{F_y}{F_z}\right)v}{\|\nabla G\|}. \tag{30}$$

If each of the two terms in the numerator are greater than zero, then their sum (and thus the normal velocity) is greater than zero. If $\mathcal{A}_1 = G_x - G_z \frac{F_x}{F_z}$, $\mathcal{A}_2 = G_y - G_z \frac{F_y}{F_z}$, $\mathcal{B}_1 = \frac{\mathcal{A}_1}{\sqrt{\mathcal{A}_1^2+\mathcal{A}_2^2}}$, and $\mathcal{B}_2 = \frac{\mathcal{A}_2}{\sqrt{\mathcal{A}_1^2+\mathcal{A}_2^2}}$, then u and v can be defined as

$$u = (C_{\text{n-s}} - F)\,\mathcal{B}_1 u_0\,(1 + f(Ax)f(By)f(Cz)) \tag{31}$$

$$v = (C_{\text{n-s}} - F)\,\mathcal{B}_2 v_0\;\;(1 + f(Ax)f(By)f(Cz))\,. \tag{32}$$

This guarantees that the surface is an outflow surface ($v_n > 0$) if both $(1 + f(Ax)f(By)f(Cz))$ and $(C_{\text{n-s}} - F)$ are positive. The only remaining requirement is that the flow be subsonic, which is true provided u_0 and v_0 are chosen appropriately. These definitions, however, are more complex than necessary. If the terms $\mathcal{B}_1$ and $\mathcal{B}_2$ have sufficient variance in x, y, and z, then u and v can be simplified to

$$u = (C_{\text{n-s}} - F)\,\mathcal{B}_1 u_0 \tag{33}$$

$$v = (C_{\text{n-s}} - F)\,\mathcal{B}_2 v_0. \tag{34}$$

4.2. Supersonic Outflow

Provided that the u_0 and v_0 from Eq. (33) and (34) and n and T_0 from Eq. (27) are defined appropriately, there will exist regions where $M_n > 1$ and regions where $M_n < 1$. Thus, different regions of the solution can be used to test implementation of supersonic and subsonic outflow enforcement at the surface $G = C_{\text{o}}$. For the supersonic case, there is no mathematical BC enforced, so there is no constraint upon p.

5. The Computational Domains

The functions F and G are used to define some of the boundaries of the computational domain. A third function, H 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 exit surface is the $G = 0$ surface. All other F-constant surfaces are slip surfaces. F, G, and H are defined as follows:

$$F = \frac{1}{2}\cos(A_f x)\cos(B_f y) - z \tag{35}$$

$$G = x - \frac{1}{2}\cos(A_g y)\sin(B_g z) - \frac{\pi}{4} \tag{36}$$

$$H = -y. \tag{37}$$

A simple definition of H is permissible since no BC's will be tested on H-constant surfaces.[c] 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. 1 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.

A six-sided domain can be defined by setting maximum and minimum values for F, 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, the supersonic outflow

[c]The sufficiency of a simple definition for H depends upon the BC's being implemented generally in the spatial sense. 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 (ξ,η,ζ) relative to the F, G, and H surfaces will need to be used.

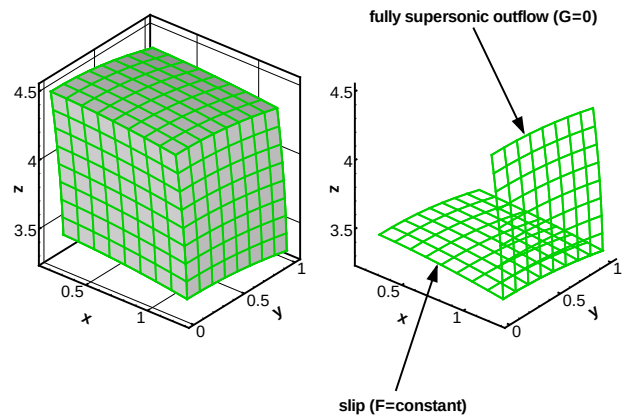

Figure 1. Domain used for testing of slip condition and supersonic outflow.

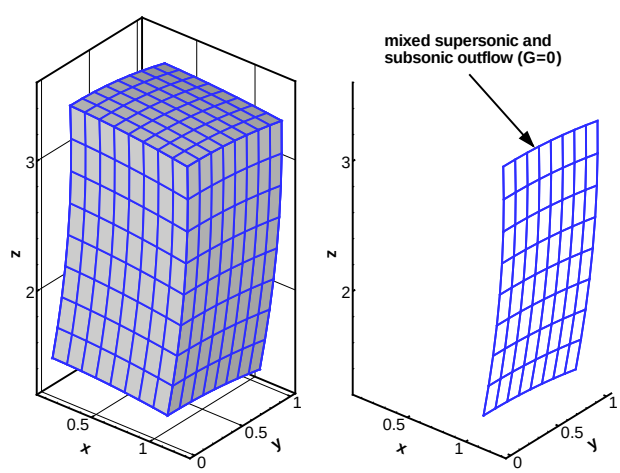

Figure 2. Domain used for testing of mixed outflow.

condition along its $G = 0$ boundary, and the Euler equations on its interior. The domain shown in Figure 2 was used to test a mixed subsonic and supersonic outflow condition along with the Euler equations on the interior. The domain shown in Figure 3 was used for testing the no-slip condition along its $F = 0$ boundary, the subsonic outflow condition along its $G = 0$ boundary, and both Euler and Navier-Stokes equations on its interior.

6. The Manufactured Solution

Once F, G, and H are defined, the manufactured solution is defined as follows:

$$\mathcal{A}_1 = G_x - G_z \frac{F_x}{F_z} \tag{38}$$

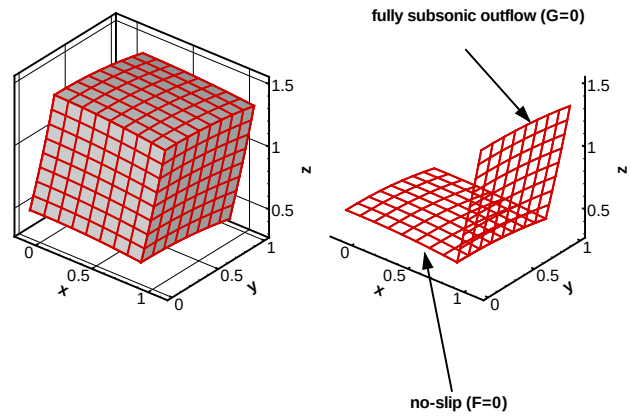

Figure 3. Domain used for testing of no-slip condition and subsonic outflow.

$$\mathcal{A}_2 = G_y - G_z \frac{F_y}{F_z} \tag{39}$$

$$\mathcal{B}_1 = \frac{\mathcal{A}_1}{\sqrt{\mathcal{A}_1^2 + \mathcal{A}_2^2}} \tag{40}$$

$$\mathcal{B}_2 = \frac{\mathcal{A}_2}{\sqrt{\mathcal{A}_1^2 + \mathcal{A}_2^2}} \tag{41}$$

$$u = \frac{3}{4}\left(-F\right)\mathcal{B}_1 a_{\mathrm{sl}} \tag{42}$$

$$v = \frac{3}{4}\left(-F\right)\mathcal{B}_2 a_{\mathrm{sl}} \tag{43}$$

$$w = -\frac{F_x u + F_y v}{F_z} \tag{44}$$

$$T = T_{\mathrm{sl}}\left(1 + \frac{1}{10}\left(1 + \sin(A_t z)\right)\right) F^2 \tag{45}$$

$$p = p_{\mathrm{sl}} - G p_{\mathrm{sl}}\left(1 + \cos(A_p x)\cos(B_p y) cos(C_p z)\right) \tag{46}$$

$$\widetilde{\nu} = -F\widetilde{\nu}_\infty\left(1 + \frac{1}{2}\cos(A_\nu x)\cos(B_\nu y)\cos(C_\nu z)\right), \tag{47}$$

where:

$$a_{\mathrm{sl}} = 340\mathrm{m/s} \tag{48}$$

$$p_{\mathrm{sl}} = 101.3\mathrm{kPa} \tag{49}$$

$$T_{\mathrm{sl}} = 288\mathrm{K} \tag{50}$$

$$\widetilde{\nu}_\infty = 1.95\nu_{\mathrm{sl}} \tag{51}$$

$$\nu_{\mathrm{sl}} = 1.464 \times 10^{-5}\mathrm{m}^2/\mathrm{s}. \tag{52}$$

The version of the solution used in this work uses $A_p = \frac{3}{4}$, $B_p = \frac{1}{4}$, $C_p = \frac{1}{2}$, $A_t = \frac{1}{4}$, $A_\nu = \frac{3}{4}$, $B_\nu = \frac{1}{4}$, and $C_\nu = \frac{1}{2}$. As was the case in the definitions of F and G, this represents a reduction in the wave numbers from Ref. 1 by a factor of four. Other constants have also changed in order to control the locations of subsonic and supersonic regions.

7. Testing of Interior Equations

Before any specialized BC is tested, it is a good idea to test the interior equations using only Dirichlet[d] BC's. This allows the interior equation set to be verified so that any difference between the expected and observed orders of accuracy in the BC tests can be rightfully interpreted as a problem with the BC enforcement, rather than a problem with the interior equation set. However, it is not necessary to verify every possible option[e] for the interior equation set before progressing to the BC's. It is only necessary to verify the same set of options which will be used in the BC tests.

7.1. The Euler Equations

Ref. 1 showed behavior approaching second-order for the Euler equations with 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[8] 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.[f] Figures 4 and 5 show the equally-weighted and inverse-distance-weighted results that were obtained after the coding mistake was fixed. Note that these results

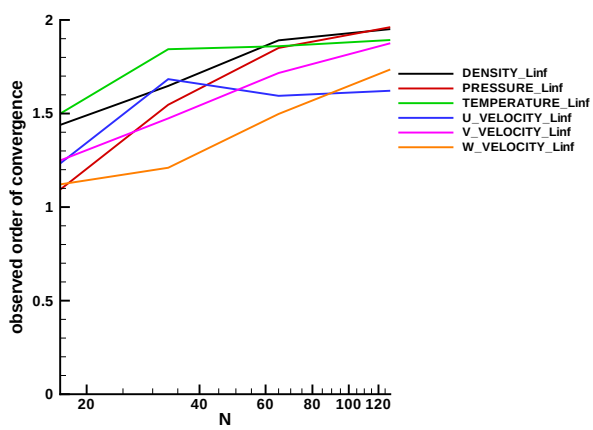

Figure 4. Convergence order for the Euler equations with equally-weighted least squares.

[d]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 frequently does not fit this special case, Premo's user-defined BC option was used to enforce the spatially-varying Dirichlet BC's.

[e]Examples of such options would be choice of gradient operator and choice of auxiliary relations such as the equation of state or transport property definitions. In Ref. 1, both Green-Gauss and least squares gradient operators were tested for the interior equations, but only Green-Gauss was used for the BC test cases. Although some of the BC's depend upon being provided an accurate value for the gradient, their implementation does not change based on how this gradient is calculated. Because of this, one test of each BC with either of the gradient operators is sufficient, rather than tests with both operators.

[f]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.

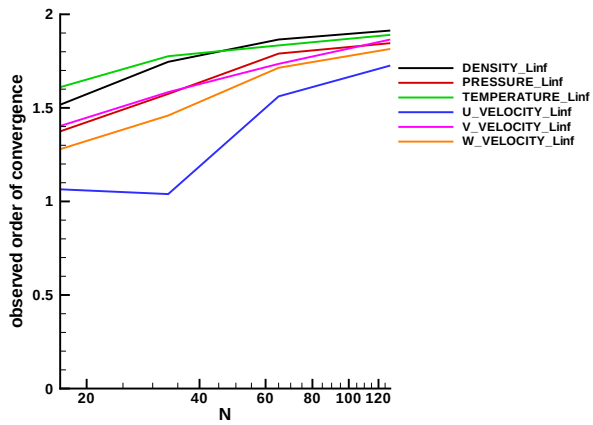

Figure 5. Convergence order for the Euler equations with inverse-distance-weighted least squares.

were obtained using the manufactured solution given in Ref. 1.

Figures 4 and 5 show the observed order of accuracy of the code for each mesh level. 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:

$$\text{observed order} = \frac{\log\left(\frac{\|\epsilon_{N/2}\|}{\|\epsilon_N\|}\right)}{\log(2)}, \tag{53}$$

where ϵ is the error (difference between discrete and exact solutions) for the flow variable of interest, and $\|\epsilon\|$ is any norm of that error. The L_∞ norms are given for the errors in the flow variables ρ, p, T, u, v, and w. The L_∞ norm is used because it tends to indicate non-convergence at a smaller mesh size than the L_1 or L_2 norms. Although no evidence of coding mistakes is shown by either Figure 4 or 5, Figure 4 does not demonstrate asymptotic behavior in the variable u. This and similar problems from Ref. 1 provided the motivation necessary for the wave number reduction in the manufactured solution that was used for the remaining cases in the present work. Figure 5, on the other hand, does appear to be within the asymptotic regime. Although both equally-weighted and inverse-distance-weighted least squares gradient reconstruction are formally second-order, the inverse-distance-weighted is more accurate on non-uniform meshes. Thus, attainment of the asymptotic regime on relatively coarser meshes by the inverse-distance-weighted approach is exactly what should be expected.

7.2. The Navier-Stokes Equations

MMS verification of the Navier-Stokes equations can be performed more on smaller meshes if values for the viscous transport properties μ and κ (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 ($\text{Pr} = \frac{\mu C_p}{\kappa}$, 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 1.458×10^{-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.

MMS tests using the Navier-Stokes equations revealed the presence of a coding mistake immediately, since the observed order of accuracy was zero, even on the coarser meshes (finer mesh runs were never completed). A coding mistake in the CHAD gradient correction[9] was found. An unordered error (*i.e.*, one that does not change with mesh spacing) was present in the implementation. This error did, however, go to zero as the mesh edges were aligned with the coordinate axes. This characteristic explains why the coding mistake went undetected in prior work on uniform, Cartesian meshes.[8]

Once the problem was corrected, tests were performed on the series of meshes using Green-Gauss gradient reconstruction. Since both Green-Gauss and least squares were verified for the Euler equations, and since the gradient calculation does not change with the governing equations, only one gradient option needed to be tested with the Navier-Stokes equations. Figure 6 shows the observed order of convergence of the L_∞ norms of the error.

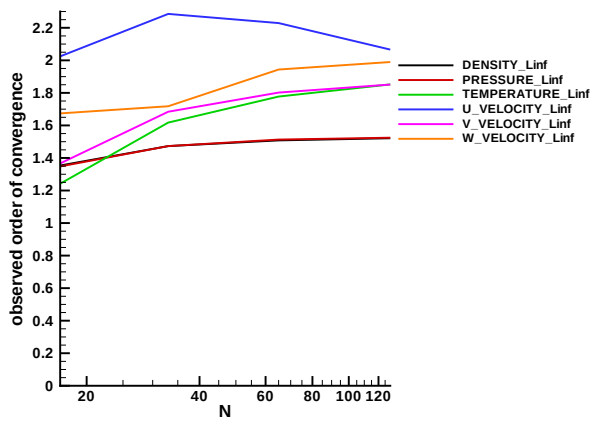

Figure 6. Convergence order for the L_∞ error norms for the Navier-Stokes equations.

The errors norms for ρ and p are nearly equivalent for this case, so their curves lie almost on top of one another. As can be seen in Figure 6, the convergence orders of the L_∞ norms of the error of T, u, v, and w appear to be approaching second-order, while those of ρ and p appear to be approaching order 1.5. This is surprising, since nothing other than an integer order should be expected for the chosen discretization scheme; nor should different orders of convergence be expected for the errors of different variables.

Figure 7 shows that the observed convergence orders of the L_1 and L_2 norms of the ρ error appear to be approaching 2. This implies that the production of less than second-order error is limited to a particular region of the domain. Inspection of the error in p and ρ on the mesh shows it to be concentrated in an area just above the $F = 0$ surface. Since the velocity goes to zero as the $F = 0$ surface is approached, the behavior in this region represents the incompressible limit. It is quite possible that the use of a compressible algorithm in this region results in a local reduction of observed order. A future test will be to perform a Navier-Stokes test on a domain which excludes the extremely low velocity region near the $F = 0$ surface. Although the plots do not support a claim of second-order accuracy for this case, the authors do not presently suspect that a coding mistake is present in the implementation of the governing equations. This is an issue that will need to be resolved in subsequent work.

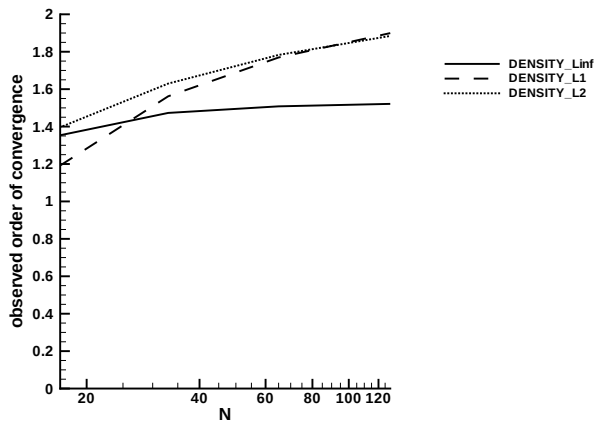

Figure 7. Convergence order for the ρ error norms for the Navier-Stokes equations.

8. 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 (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.[10] 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.[11] This is only 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_∞ 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 performance characteristics of a given BC implementation.

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 verified for fully supersonic outflow in Ref. 1. However, problems with the slip condition were observed in Ref. 1 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 shown in Section 8.1. Sections 8.2 and 8.3 show results for the outflow tests on mixed subsonic and supersonic outflow and for fully subsonic outflow, respectively. 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). However, results for the fully subsonic case on the finest mesh were unavailable at the time this paper was written.

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 Sections 8.4 and 8.5, respectively. For the isothermal case, results for the finest mesh were unavailable at the time this paper was written.

8.1. The Slip Condition

Figure 8 shows the observed order of accuracy versus mesh size for the slip condition on hexahedral meshes. This test was performed with a weak[g] implementation of the BC. This is based on the error calculation

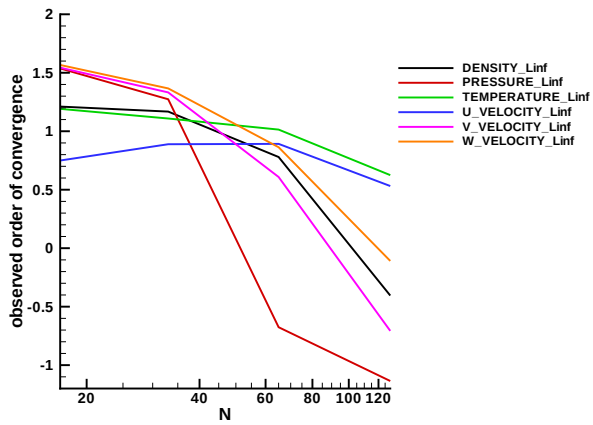

Figure 8. Convergence order for the slip condition on hexahedral meshes.

restricted to the slip boundary (as opposed to the entire domain); however, since the locations of the maximum error are located on the slip boundary, the plots for L_∞ norms look the same whether the error is calculated on the entire domain or restricted to the boundary. Despite the generalization of the unit normal calculation to handle curved surfaces, something is still preventing the expected order of convergence from being observed on the slip boundary.

This test was repeated on a series of tetrahedral meshes to determine if the problem could be isolated to hexahedral meshes. The sequence of tetrahedral meshes was constructed by using all of the nodes in the hexahedral meshes and adding the edges necessary to subdivide each hexahedron into its six constituent tetrahedra (this is not a unique decomposition, but which decomposition is chosen is immaterial). The results are shown in Figure 9.

A first glance at Figure 9 hints that whatever problems are evident on the hexahedral meshes are not present on the tetrahedral meshes. Some first-order error is expected at this boundary, so the L_∞ norm (or any norm for the error calculation restricted to the slip boundary) should detect some first-order error. However, the error in p, which is the first variable to exhibit the problem on hexahedral meshes, is below one and trending downward. The norms of the errors in w, ρ, and u are also trending downward, and none of these curves appear to be asymptotically approaching a constant value. This is alarming, although the presence of less than first-order error can not be confirmed on this series of meshes. Since the nodes of the hexahedral and tetrahedral meshes were collocated, one should not expect the tetrahedral mesh to need more nodes to detect the same problem, but Figure 9 does not provide clear evidence that the problem seen on hexahedral meshes is absent from tetrahedral meshes. However, no specific coding mistake has yet been identified. Whether one exists or whether the problem is with the MMS test itself has not been resolved. Other simluations have been completed using the slip condition without trouble and have shown reductions in errors for specific quantities of interest. Although these simulations are not as thorough as the verification tests performed here, increasing errors with mesh refinement have not been observed outside of the MMS tests. Further investigation is warranted.

[g]A weak formulation of a BC is one that is implemented through the definition of a flux at the boundary, as opposed to the setting of the dependent variable (strong formulation) or modification of the residual vector in the solver (residual formulation).

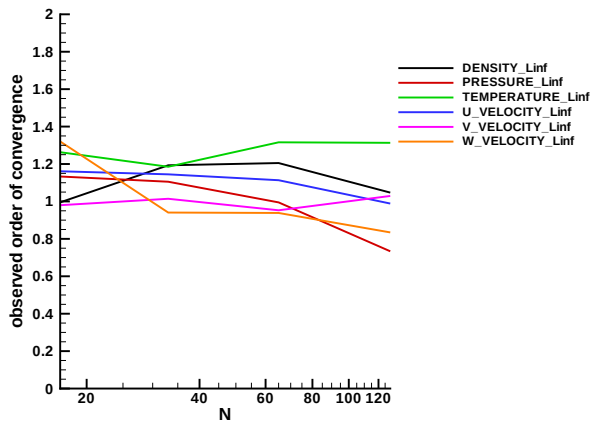

Figure 9. Convergence order for the slip condition on tetrahedral meshes.

8.2. Mixed Subsonic and Supersonic Outflow

Results of the test with a mixed subsonic and supersonic outflow BC are shown in Figure 10. This test

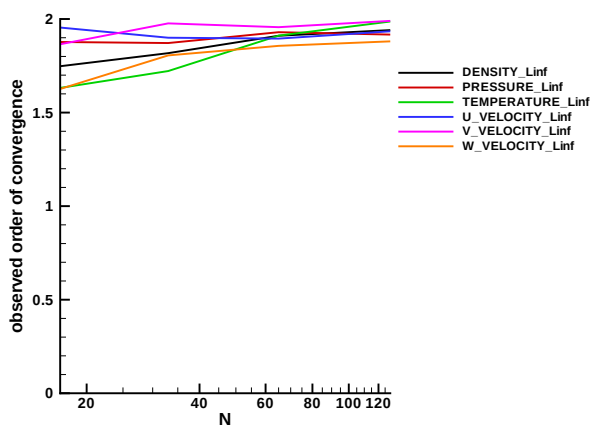

Figure 10. Convergence order for the mixed subsonic & supersonic outflow boundary.

was performed with a weak implementation of the BC. Since this figure shows the L_∞ norms with the error calculation restricted to the exit surface, the second-order results show that the implementation of this BC in this normal Mach number regime does not generate any first-order error. Since there is no first-order error generated, there is no need to examine whether such error, if it existed, would propagate into the interior. Note that if this test case had been performed before the fully-supersonic case in Ref. 1, it would have rendered a fully-supersonic case unnecessary for Premo because a single outflow condition handles all Mach numbers, and this test verified it for both subsonic and supersonic outflow. However, the manufactured solution was designed to also be used for codes that may not be as general. Additionally, in the event that mixed outflow results had revealed problems, the fully subsonic and fully supersonic regions could have been

used to isolate the potential coding mistake.[h]

8.3. Fully Subsonic Outflow

Although the mixed subsonic and supersonic outflow test confirmed that a subsonic outflow boundary would not generate anything less than second-order error, another test was performed with a fully subsonic boundary to examine what would happen when $\vec{v} = 0$. The domain chosen (see Figure 3) has $\vec{v} = 0$ along the bottom edge of the boundary and $v_n > 0$ elsewhere. Although this edge is the intersection of the outflow ($G = 0$) and no-slip ($F = 0$) boundaries, this test was performed with Dirichlet BC's applied at $F = 0$, rather than the no-slip BC.

Figure 11 shows the observed order of convergence for all of the norms of the w error. This test was

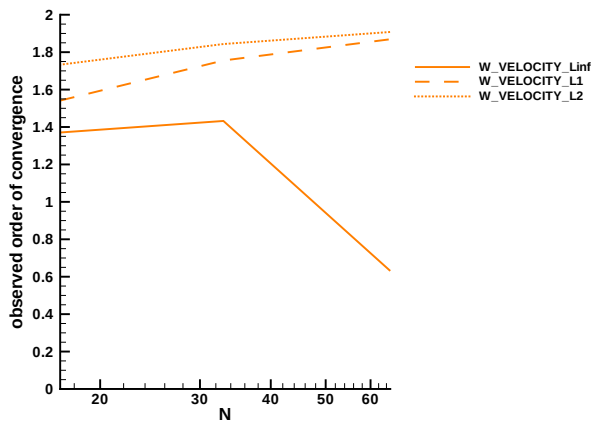

Figure 11. Convergence order for fully subsonic outflow with a $\vec{v} = 0$ line.

performed with a weak implementation of the BC. Error calculations for this plot were performed over the entire domain, but the plot that results from the error calculation being restricted to the exit surface is almost identical. These norms are similar to the norms of the u error, but all of the other variables have their L_∞ norms tending toward 2. Although results for the $N = 129$ case were not available, it appears that there is some error of less than second-order being generated in the $\vec{v} \approx 0$ region. Since the L_1 and L_2 norms do not detect this error, and since the results with the error calculation performed over the entire volume and restricted to the exit surface give similar results, it does not appear that the error generated in this region is propagated to other areas of the domain in detectable quantities. However, the lack of fine mesh results for this case means that this property can not be established with certainty.

8.4. Adiabatic No-Slip Boundary

The no-slip condition enforces $\vec{v} = 0$ by modifying the residual vector at each iteration, so the error in the variables u, v, and w should be zero at the wall for an iteratively converged solution. This is known as a residual enforcement of the BC. The errors of the other variables can be calculated at the wall, and Figure 12 shows their norms.

[h] It is also worth noting that the subsonic portions of the boundary were expected to generate some first-order error, whereas the supersonic portions were expected to verify to second-order. If this had turned out to be the case, then the fully supersonic case would have been useful in demonstrating a higher observed order than the mixed outflow test. The authors presently believe that their initial assessment of the error structure on subsonic portions of an outflow boundary was incorrect, although this has not yet been proven with a re-derivation of the expected order.

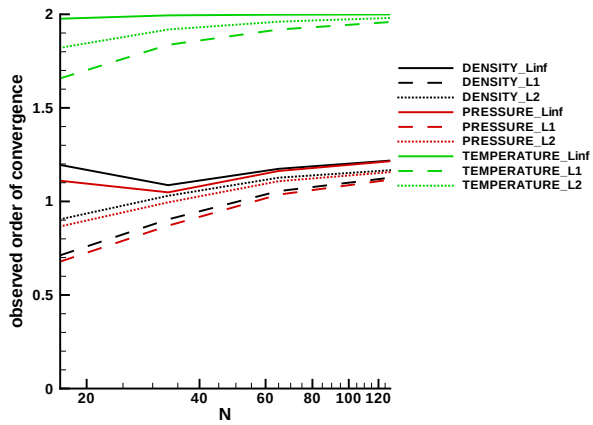

Figure 12. Convergence order for the adiabatic no-slip condition with error calculation restricted to the wall.

Figure 12 does indicate some first-order error is being generated in the ρ and p variables. This is expected, given the implementation of this BC. To determine whether this error propagates into the interior, the L_2 norm of the error of all of the variables was plotted, and this plot is shown in Figure 13.

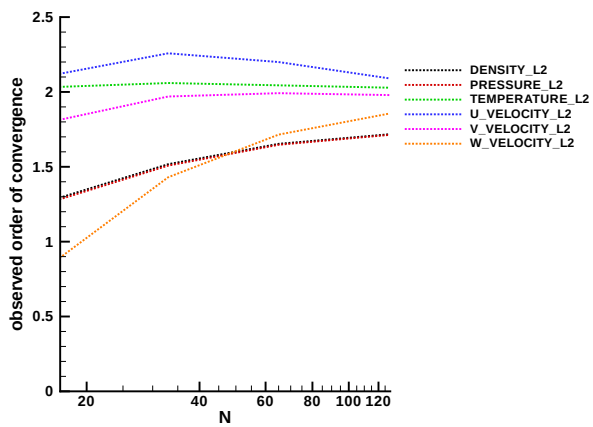

Figure 13. Convergence order for the adiabatic no-slip condition with error calculation performed over the entire domain.

Figure 13 shows the L_2 norms of the errors of all variables to be above 1.5 and converging toward 2. This indicates that the error generated at the surface is not propagating into the interior in measurable quantities.

8.5. Isothermal No-Slip Boundary

Results of the isothermal option for the no-slip condition are similar. In this case, the residual enforcement specifies a constant value of T as well as u, v, and w, so the only non-zero error on the wall is in the p and ρ variables. The L_∞ norms of these errors are given in Figure 14.

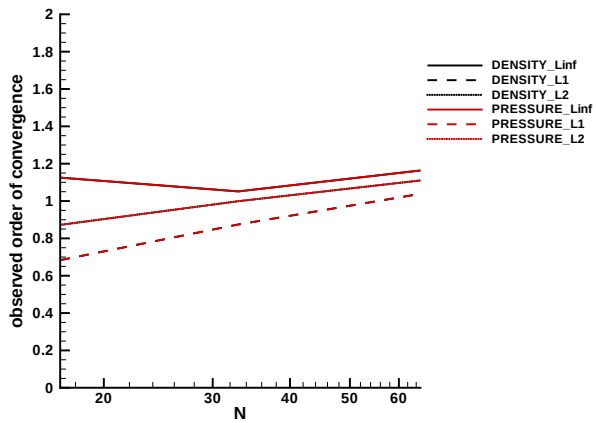

Figure 14. Convergence order for the isothermal no-slip condition with error calculation restricted to the wall.

In this case, the convergence orders for the p and ρ errors are close enough to being equal that the lines lie on top of one another. As was the case for the adiabatic no-slip, first-order error in ρ and p is generated at the wall, but this error does not appear to propagate into the interior (Figure 15). These results are not

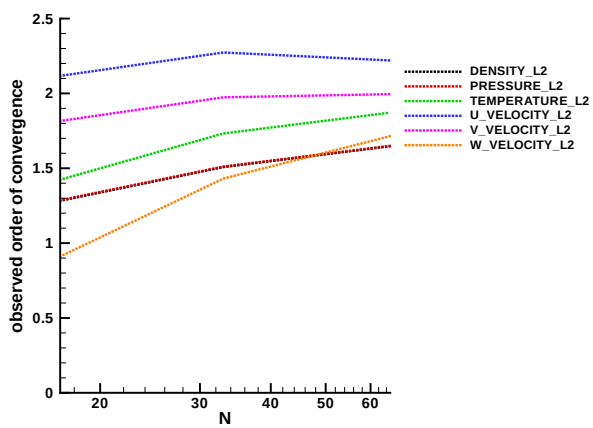

Figure 15. Convergence order for the isothermal no-slip condition with error calculation performed over the entire domain.

entirely conclusive because the $N = 129$ mesh results were unavailable at the time this paper was written.

9. Conclusions

A manufactured solution was 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. Follow-up tests confirmed that a previously identified problem with least squares gradient reconstruction has been corrected and that the Euler equations are

solved to second-order for all gradient reconstruction options. The results of tests with the Navier-Stokes equations using Dirichlet BC's initially revealed a coding mistake, but after that mistake was corrected, no further evidence of mistakes was found, and the Navier-Stokes equations appear to be solved to at least order 1.5. The coding mistake identified would not have been found using Cartesian meshes.

The expected order of accuracy was not observed during tests of the slip condition, despite the correction of a previously identified coding mistake. It has not been determined whether an additional coding mistake is present or whether a problem with the MMS test exists. All other BC's exhibited at least the expected order of accuracy, indicating correct implementations. The outflow boundary implementation was verified to second-order wherever the flow was actually outward ($v_n > 0$), but wherever the flow was stagnant ($\vec{v} \approx 0$), lower-order error was generated. Both the adiabatic and isothermal versions of the no-slip condition implementation generated some first-order error (this was expected). All of the first-order error generated at these boundaries did not propagate into the interior in measurable quantities; however, some of the cases with first-order error present on the boundary did not have fine mesh results to conclusively demonstrate this property of non-propagation. Table 1 gives a summary of results from this work and Ref. 1.

Table 1. Summary of results from present work and Ref. 1

test	lowest order for L_2 norm	lowest order for L_∞ norm
Euler w/ Green-Gauss	2	2
Euler w/ equally-weighted least squares	2	2
Euler w/ inverse-distance-weighted least squares	2	2
Navier-Stokes	2	1.5
slip condition	unordered	unordered
adiabatic no-slip condition	2	1
isothermal no-slip condition	2	1
outflow for $v_n > 0$	2	2
outflow for $\vec{v} \approx 0$	2	?

This work underscores the primary advantage of using skewed, non-uniform meshes for MMS verification: some coding mistakes will be missed if MMS testing is only performed on uniform, Cartesian meshes. This work also addresses the primary challenge of using skewed, non-uniform meshes for MMS verification: finer meshes may be necessary to reach the asymptotic regime than what would be expected for uniform, Cartesian meshes. The current work showed that decreasing the wave numbers in the sinusoidal functions used to generate the manufactured solution makes the asymptotic regime easier to attain, so future work may involve a further reduction in these wave numbers.[i] Another potential area for improvement is the use of inversely-weighted least squares gradient reconstruction for cases which are not specifically intended to test one of the other gradient options. Since this is the most accurate method of calculating the gradient on skewed, non-uniform meshes, it should allow the asymptotic regime to be reached on relatively coarser meshes. Two major objectives of the third and final phase of this work will be to demonstrate attainment of the asymptotic regime on reasonably-sized meshes and to verify the RANS governing equations.

This work demonstrated the value of using the method of manufactured solutions for verifying the BC's of CFD codes. The derived manufactured solution is applicable not only to Premo, but also to general CFD codes.

[i] All of the cases for which no coding mistakes were indicated were in the asymptotic regime by the last two mesh levels for the current version of the manufactured solution. This was not the case for the previous version[1] of the manufactured solution.

10. Acknowledgments

The authors would like to thank Alfred Lorber and Thomas Smith for their help in implementing the manufactured solutions in Premo. They, along with co-author Curtis Ober, are Premo developers.

References

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

[2]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.

[3]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.

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

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

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

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

[8]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.

[9]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.

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

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