

Comprehensive Code Verification for an Unstructured Finite Volume CFD Code

Subrahmanya P. Veluri¹ and Christopher J. Roy²
Virginia Tech, Blacksburg, Virginia, 24061

Edward A. Luke³
Mississippi State University, Starkville, MS, 39762

A detailed code verification study of an unstructured finite volume CFD code is presented. The Method of Manufactured Solutions is used to generate exact solutions to the Euler and Navier-Stokes equations to verify the order of accuracy of the code. Testing is performed on different grid types including triangular and quadrilateral elements in 2D and tetrahedral, prismatic and hexahedral elements in 3D. The requirements for systematic mesh refinement are discussed, particularly in regards to unstructured meshes. Different code options verified include the baseline steady-state governing equations, transport models, turbulence models, boundary conditions and unsteady flows.

I. Introduction

VERIFICATION addresses the mathematical correctness of the simulations. There are two fundamental aspects to verification: code verification and solution verification.¹⁻³ Code verification is the process of ensuring, to the degree possible, that there are no mistakes (bugs) in a computer code or inconsistencies in the solution algorithm. Solution verification is the process of estimating the three types of numerical error that occur in every numerical simulation: round-off error, iterative error, and discretization error. This paper focuses on code verification. Current practices in code verification verify that the observed order of accuracy asymptotically approaches the formal order of accuracy of the discretization scheme as the mesh is systematically refined.

One of the main difficulties in verifying a code is identifying exact solutions to the governing equations which exercise all terms. Traditional exact solutions exist only when the governing equations are fairly simple, which is certainly not the case for modern CFD codes which are expected to handle complex physics (turbulence, combustion, real gas effects, etc.), complex geometries, and significant nonlinearities. The Method of Manufactured Solutions, or MMS, is a general and very powerful approach to code verification.¹⁻³ Rather than trying to find an exact solution to a system of partial differential equations, the goal is to “manufacture” an exact solution to a slightly modified set of equations. Order of accuracy verification is a rigorous code verification assessment which involves comparing the observed order of accuracy for the CFD solutions to the formal order of accuracy of the chosen numerical methods. The observed order of accuracy can be adversely affected by mistakes in the computer code, defective numerical algorithms, solutions which are not sufficiently smooth and numerical solutions which are not in the asymptotic grid convergence range.

The first application of MMS for code verification was by Roache and Steinberg in 1984.⁴ In their pioneering work, they used the MMS approach to verify a code for generating three-dimensional transformations for elliptic partial differential equations. Additional discussions of the MMS procedure for code verification have been presented by Roache.^{1,5} The book by Knupp and Salari⁶ is a comprehensive discussion of code verification, MMS, and order of accuracy verification.

MMS has been used to verify two compressible CFD codes⁷: Premo⁸ (developed by Sandia National Laboratories) and WIND⁹ (developed by the NPARC alliance). In this work, the authors successfully verified both the inviscid Euler equations and the laminar Navier-Stokes equations; however, this study employed only Cartesian grids. An alternative statistical approach to MMS was proposed by Hebert and Luke¹⁰ for the Loci-CHEM combusting CFD code.¹¹ In their approach, they employ a single grid level which is shrunk down (thus providing a

¹ Graduate Research Assistant, Student Member, AIAA.

² Associate Professor, Associate Fellow, AIAA.

³ Associate Professor, Member, AIAA.

locally refined grid) and used to statistically sample the discretization error in different regions of the domain of interest. Their work successfully verified the Loci-CHEM CFD code for the 3D, multi-species, laminar Navier-Stokes equations using both statistical and traditional MMS.

There have been three coordinated efforts to apply MMS to turbulent flows. Pelletier and co-workers have summarized their work on 2D incompressible turbulent shear layers using a finite element code with a focus on a logarithmic form of the k- ϵ two-equation RANS model in Refs. 12 and 13. They employed Manufactured Solutions which mimic turbulent shear flows, with the turbulent kinetic energy and the turbulent eddy viscosity as the two quantities specified in the Manufactured Solution. For the cases examined, they were able to verify the code by reproducing the formal order of accuracy of the code. More recently, Eca and co-workers have published a series of papers on Manufactured Solutions for the 2D incompressible turbulent Navier-Stokes equations.¹⁴⁻¹⁶ They also employed physically-based Manufactured Solutions, in this case mimicking wall-bounded turbulent flow. This group looked at both finite-difference and finite-volume discretizations, and examined a number of turbulence models including the Spalart-Allmaras one-equation model¹⁷ and two two-equation models: Menter’s baseline (BSL) version k- ω model¹⁸ and Kok’s turbulent/non-turbulent k- ω model.¹⁹ While successful in some cases, their physically-based Manufactured Solution often led to numerical instabilities, a reduction in the observed grid convergence rate, or even inconsistency of the numerical scheme (i.e., the discretization error did not decrease as the grid was refined). In order to independently test different aspects of the governing equations, in some cases they replaced certain discretized terms (or even whole equations) with the analytic counterpart from the Manufactured Solution. For the Spalart-Allmaras model they specified the working variable $\tilde{\nu}$, while for the two equation models they specified both the turbulent eddy viscosity and the turbulent kinetic energy. The cases they examined employed a Reynolds number of 10^6 and used Cartesian grids which were clustered in the y-direction towards the wall. Our approach to code verification for RANS models differ from the previous work in number of ways. While the earlier work focused on physically-based solutions with complex exponentials to mimic the turbulence quantities found in real turbulent flows, we simply use sinusoidal functions. Our argument for taking this approach is the goal of code verification is to perform mathematical tests to ensure the discretization approach and the implementation into a code does in fact match the original governing partial differential equations and their solution.

When performing code verification of boundary conditions, there are two approaches of handling boundary conditions.³ The first approach is to impose the mathematically consistent boundary condition that is required according to the mathematical character of the differential equations. The second approach is to simply over-specify the boundary condition from the manufactured solution. In order to verify the implementation of boundary conditions, the manufactured solution must be tailored to exactly satisfy a given boundary condition on a domain boundary and Bond et al.²⁶ developed a general approach for doing the same.

II. Governing Equations

The CFD code verified in the current work is Loci-CHEM¹¹ and was developed at Mississippi State University. Loci-CHEM was developed using the Loci framework²⁰ and can simulate three-dimensional flows of turbulent, chemically-reacting mixtures of thermally perfect gases. It is a library of Loci rules that consists of reusable rules that can be dynamically reconfigured to solve a variety of problems. Two major advantages of the Loci framework are that it does not allow dependencies on un-initialized variables (one of the most common faults in software engineering) and it automatically handles domain decomposition and parallelization. The baseline governing equations include the 3D steady state Euler and Navier-Stokes equations. Euler equations can be obtained from the Navier-Stokes equations by removing the viscous terms.

A. Flow Equations

The 3D, steady state, Favre-averaged Navier-Stokes equations²¹ can be written as

$$\frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \frac{\partial(\rho w)}{\partial z} = 0$$

$$\frac{\partial(\rho u^2 + p - t_{xx} - \tau_{xx})}{\partial x} + \frac{\partial(\rho uv - t_{xy} - \tau_{xy})}{\partial y} + \frac{\partial(\rho uw - t_{xz} - \tau_{xz})}{\partial z} = 0$$

$$\begin{aligned} \frac{\partial(\rho v u - t_{xy} - \tau_{xy})}{\partial x} + \frac{\partial(\rho v^2 + p - t_{yy} - \tau_{yy})}{\partial y} + \frac{\partial(\rho v w - t_{yz} - \tau_{yz})}{\partial z} &= 0 \\ \frac{\partial(\rho w u - t_{xz} - \tau_{xz})}{\partial x} + \frac{\partial(\rho w v - t_{yz} - \tau_{yz})}{\partial y} + \frac{\partial(\rho w^2 + p - t_{zz} - \tau_{zz})}{\partial z} &= 0 \\ \frac{\partial(\rho u h_t - u(t_{xx} + \tau_{xx}) - v(t_{xy} + \tau_{xy}) - w(t_{xz} + \tau_{xz}) + q_{L_x} + q_{T_x})}{\partial x} \\ + \frac{\partial(\rho v h_t - u(t_{xy} + \tau_{xy}) - v(t_{yy} + \tau_{yy}) - w(t_{yz} + \tau_{yz}) + q_{L_y} + q_{T_y})}{\partial y} \\ + \frac{\partial(\rho w h_t - u(t_{xz} + \tau_{xz}) - v(t_{yz} + \tau_{yz}) - w(t_{zz} + \tau_{zz}) + q_{L_z} + q_{T_z})}{\partial z} &= 0 \end{aligned}$$

where t_{ij} is the laminar stress tensor given by

$$\begin{aligned} t_{xx} &= \frac{2}{3}\mu\left(2\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} - \frac{\partial w}{\partial z}\right), & t_{yy} &= \frac{2}{3}\mu\left(2\frac{\partial v}{\partial y} - \frac{\partial u}{\partial x} - \frac{\partial w}{\partial z}\right), & t_{zz} &= \frac{2}{3}\mu\left(2\frac{\partial w}{\partial z} - \frac{\partial u}{\partial x} - \frac{\partial v}{\partial y}\right) \\ t_{xy} &= \mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right), & t_{yz} &= \mu\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right), & t_{xz} &= \mu\left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}\right) \end{aligned}$$

and τ_{ij} is the turbulent stress tensor given by

$$\begin{aligned} \tau_{xx} &= \frac{2}{3}\mu_T\left(2\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} - \frac{\partial w}{\partial z}\right), & \tau_{yy} &= \frac{2}{3}\mu_T\left(2\frac{\partial v}{\partial y} - \frac{\partial u}{\partial x} - \frac{\partial w}{\partial z}\right), & \tau_{zz} &= \frac{2}{3}\mu_T\left(2\frac{\partial w}{\partial z} - \frac{\partial u}{\partial x} - \frac{\partial v}{\partial y}\right) \\ \tau_{xy} &= \mu_T\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right), & \tau_{yz} &= \mu_T\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right), & \tau_{xz} &= \mu_T\left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}\right) \end{aligned}$$

The turbulent and laminar heat flux terms are

$$\begin{aligned} q_{T_x} &= -\frac{\mu_T}{Pr_T}c_p\frac{\partial T}{\partial x}, & q_{T_y} &= -\frac{\mu_T}{Pr_T}c_p\frac{\partial T}{\partial y}, & q_{T_z} &= -\frac{\mu_T}{Pr_T}c_p\frac{\partial T}{\partial z} \\ q_{L_x} &= -\frac{\mu}{Pr}c_p\frac{\partial T}{\partial x}, & q_{L_y} &= -\frac{\mu}{Pr}c_p\frac{\partial T}{\partial y}, & q_{L_z} &= -\frac{\mu}{Pr}c_p\frac{\partial T}{\partial z} \end{aligned}$$

and the total energy and enthalpy are

$$e_t = e + \frac{1}{2}(u^2 + v^2 + w^2) \quad \text{and} \quad h_t = e_t + \frac{p}{\rho}$$

where

$$e = nRT + h_f \quad \text{and} \quad h = (n+1)RT + h_f.$$

The perfect gas equation of state is assumed

$$p = \rho RT$$

and the heat of formation and excited energy mode parameter can be calculated as

$$h_f = h_{ref} - (n+1)RT_{ref} \quad \text{and} \quad n = \frac{c_v}{R}$$

where $h_{ref} = 0$, $T_{ref} = 298$ K, and $n = 5/2$ are used.

B. Turbulence Equations

The turbulence models examined in this work are the baseline (BSL) version of Menter's two-equation k- ω model¹⁸ and the Menter's Shear Stress Transport (SST) k- ω model.¹⁸ One of the difficulties in verifying the two-equation models is the handling of the blending functions and various min and max functions in the models. The general form of the turbulent kinetic energy (k) and the turbulent frequency (ω) for compressible flow are:

$$\begin{aligned} \frac{\partial(\rho uk)}{\partial x} + \frac{\partial(\rho vk)}{\partial y} + \frac{\partial(\rho wk)}{\partial z} - P + \beta^* \rho \omega k - \frac{\partial}{\partial x} \left[(\mu + \sigma_k \mu_T) \frac{\partial k}{\partial x} \right] - \frac{\partial}{\partial y} \left[(\mu + \sigma_k \mu_T) \frac{\partial k}{\partial y} \right] - \frac{\partial}{\partial z} \left[(\mu + \sigma_k \mu_T) \frac{\partial k}{\partial z} \right] &= 0 \\ \frac{\partial(\rho u \omega)}{\partial x} + \frac{\partial(\rho v \omega)}{\partial y} + \frac{\partial(\rho w \omega)}{\partial z} - \frac{\gamma \rho}{\mu_T} P + \beta^* \rho \omega^2 - \frac{\partial}{\partial x} \left[(\mu + \sigma_k \mu_T) \frac{\partial k}{\partial x} \right] - \frac{\partial}{\partial y} \left[(\mu + \sigma_k \mu_T) \frac{\partial k}{\partial y} \right] \\ - \frac{\partial}{\partial z} \left[(\mu + \sigma_k \mu_T) \frac{\partial k}{\partial z} \right] - 2(1-F) \rho \sigma_{\omega 2} \frac{1}{\omega} \left[\frac{\partial k}{\partial x} \frac{\partial \omega}{\partial x} + \frac{\partial k}{\partial y} \frac{\partial \omega}{\partial y} + \frac{\partial k}{\partial z} \frac{\partial \omega}{\partial z} \right] &= 0 \end{aligned}$$

The turbulence production term is given by

$$P = \tau'_{xx} \frac{\partial u}{\partial x} + \tau'_{yy} \frac{\partial v}{\partial y} + \tau'_{zz} \frac{\partial w}{\partial z} + \tau'_{xy} \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right) + \tau'_{yz} \left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} \right) + \tau'_{zx} \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right)$$

where the full compressible turbulent stress tensor is used:

$$\begin{aligned} \tau'_{xx} &= \frac{2}{3} \mu_T \left(2 \frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} - \frac{\partial w}{\partial z} - \rho k \right), \quad \tau'_{yy} = \frac{2}{3} \mu_T \left(2 \frac{\partial v}{\partial y} - \frac{\partial u}{\partial x} - \frac{\partial w}{\partial z} - \rho k \right), \quad \tau'_{zz} = \frac{2}{3} \mu_T \left(2 \frac{\partial w}{\partial z} - \frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} - \rho k \right) \\ \tau'_{xy} &= \mu_T \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right), \quad \tau'_{yz} = \mu_T \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right), \quad \tau'_{zx} = \mu_T \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) \end{aligned}$$

For the BSL Menter model, the turbulent viscosity is given by

$$\mu_T = \rho k / \omega$$

where the closure coefficients are found by combining the coefficients from the k- ω and k- ϵ models as

$$\phi = F \phi_1 + (1-F) \phi_2$$

and the k- ω coefficients (ϕ_1) are

$$\begin{aligned} \sigma_{k1} &= 0.5, \quad \sigma_{\omega 1} = 0.5, \quad \beta_1 = 0.0750 \\ \beta^* &= 0.09, \quad \kappa = 0.41, \quad \gamma_1 = \beta_1 / \beta^* - \sigma_{\omega 1} \kappa^2 / \sqrt{\beta^*} \end{aligned}$$

and the k-ε coefficients (ϕ_2) are

$$\sigma_{k_2} = 1.0, \quad \sigma_{\omega_2} = 0.856, \quad \beta_2 = 0.0828$$

$$\beta^* = 0.09, \quad \kappa = 0.41, \quad \gamma_2 = \beta_2 / \beta^* - \sigma_{\omega_2} \kappa^2 / \sqrt{\beta^*}$$

In the BSL model the blending function is given by

$$F = F_1 = \tanh(\arg_1^4)$$

$$\arg_1 = \min \left[\max \left(\frac{\sqrt{k}}{0.09 \omega d}, \frac{500 \nu}{d^2 \omega} \right), \frac{4 \rho \sigma_{\omega_2} k}{CD_{k\omega} d^2} \right]$$

$$CD_{k\omega} = \max \left[2 \rho \sigma_{\omega_2} \frac{1}{\omega} \left(\frac{\partial k}{\partial x} \frac{\partial \omega}{\partial x} + \frac{\partial k}{\partial y} \frac{\partial \omega}{\partial y} + \frac{\partial k}{\partial z} \frac{\partial \omega}{\partial z} \right), 10^{-20} \right]$$

and d is the distance to the nearest no-slip wall.

For the Shear-Stress Model, the formulation is same as the BSL Menter model except for the constants of ϕ_1 need to be changed.

$$\sigma_{k_1} = 0.85, \quad \sigma_{\omega_1} = 0.5, \quad \beta_1 = 0.0750, \quad a_1 = 0.31$$

$$\beta^* = 0.09, \quad \kappa = 0.41, \quad \gamma_1 = \beta_1 / \beta^* - \sigma_{\omega_1} \kappa^2 / \sqrt{\beta^*}$$

and the eddy viscosity is defined as

$$\nu_t = \frac{a_1 k}{\max(a_1 \omega, \Omega F_2)}$$

where Ω is the absolute value of vorticity and F_2 is given by

$$F_2 = \tanh(\arg_2^2)$$

$$\arg_2 = \max \left(2 \frac{\sqrt{k}}{0.09 \omega d}, \frac{500 \nu}{d^2 \omega} \right)$$

III. Method of Manufactured Solutions

A. Methodology

The most rigorous approach to code verification is the order of accuracy test,^{1,6} which determines whether or not the discretization error is reduced at the expected rate. This test is equivalent to whether the observed order of accuracy matches the formal order of accuracy. The formal order of accuracy is obtained from a truncation error analysis of the discrete algorithm for the finite-difference and finite-volume methods and it is found from interpolation theory for finite-element methods. The observed order of accuracy is directly computed from code output for a given set of simulations. The observed order of accuracy can be adversely affected by mistakes in the computer code, defective numerical algorithms, solutions which are not sufficiently smooth and numerical solutions which are not in the asymptotic grid convergence range.² The asymptotic range is defined as the range of discretization sizes (Δx , Δy , Δt , etc.) where the lowest-order terms in the truncation error dominate.

The accepted approach for code verification used currently is calculating the observed order of accuracy assuming that the exact solution is known. The discretization error is formally defined as the difference between the exact solution to the discrete equations and the exact solution to the governing partial differential equations. Since the exact solution to the discrete equations (which will be different on different mesh levels) is generally not known, the numerical solution on the same mesh level is substituted in its place, thus neglecting iterative and round-off error. Consider a series expansion of the discretization error in terms of h_k , a measure of the element size on the mesh level k

$$DE_k = f_k - f_{exact} = g_p h_k^p + HOT$$

where f_k is the numerical solution on mesh k , g_p is the coefficient of the leading error term, and p is the observed order of accuracy. The main assumption is that the higher-order terms (HOT) are negligible, which is equivalent to saying the solutions are in the asymptotic range. In this case, we can write the discretization error equation for a fine mesh ($k=1$) and a coarse mesh ($k=2$) as

$$DE_1 = f_1 - f_{exact} = g_p h_1^p \quad \text{and} \quad DE_2 = f_2 - f_{exact} = g_p h_2^p$$

Since the exact solution is known, these two equations can be solved for the observed order of accuracy p . Introducing r , the ratio of coarse to fine grid element spacing ($r=h_2/h_1$), the observed order of accuracy becomes

$$p = \ln\left(\frac{DE_2}{DE_1}\right) / \ln(r)$$

Thus, when the exact solution is known, only two solutions are required to obtain the observed order of accuracy. The observed order of accuracy can be evaluated either locally within the solution domain or globally by employing a norm of the discretization error. While we have examined L_1 , L_2 , L_∞ norms of the current code verification study, here we report only L_2 norms for brevity. The discrete L_2 norm for mesh level k is defined as

$$L_{2,k} = \left[\sum_{i=1}^N (DE_{k,i})^2 / N \right]^{1/2}$$

where the i index denotes a cell center value of one of the conserved variables.

When evaluating the observed order of accuracy, round-off and iterative convergence error can adversely affect the results. Round-off error occurs due to finite digit storage on digital computers. Iterative error occurs any time an iterative method is used, as is generally the case for nonlinear systems and large, sparse linear systems. The discretized form of nonlinear equations can generally be solved to within machine round-off error; however, in practice, the iterative procedure is usually terminated earlier to reduce computational effort. In order to ensure that these source of error do not adversely impact the order of accuracy calculation,² both round-off and iterative error should be at least 100 times smaller than the discretization error (i.e., $<0.01 \times DE$). For all cases presented herein, double precision computations are used and the residuals are reduced down to machine zero. For the current computations, this corresponds to a residual reduction of approximately 14 orders of magnitude.

The procedure for applying MMS with order of accuracy verification can be summarized as follows:

1. Choose the specific form of governing equations
2. Choose the manufactured solution
3. Operate the governing equations on the chosen solution, resulting in analytic source terms
4. Solve the modified governing equation (original equation plus source terms) on various mesh levels
5. Compute the observed of accuracy and compare it with the formal order of accuracy

B. Baseline Manufactured Solutions

In our current work, we adhere to the philosophy that code verification is simply a mathematical test to ensure the numerical solution truly represents the solution to the continuum mathematical equations that are being solved. As such, we have specifically chosen Manufactured Solutions which are not physically realistic, but which are simple, smooth, and exercise all terms in the governing equations. The Manufactured Solutions employed here all take the following form

$$\begin{aligned} \phi(x, y, z) = & \phi_0 + \phi_x f_s \left(\frac{a_{\phi_x} \pi x}{L} \right) + \phi_y f_s \left(\frac{a_{\phi_y} \pi y}{L} \right) + \phi_z f_s \left(\frac{a_{\phi_z} \pi z}{L} \right) + \phi_{xy} f_s \left(\frac{a_{\phi_{xy}} \pi xy}{L^2} \right) \\ & + \phi_{yz} f_s \left(\frac{a_{\phi_{yz}} \pi yz}{L^2} \right) + \phi_{zx} f_s \left(\frac{a_{\phi_{zx}} \pi zx}{L^2} \right) + \phi_{xyz} f_s \left(\frac{a_{\phi_{xyz}} \pi xyz}{L^2} \right) \end{aligned}$$

where $\phi = [\rho, u, v, p, k, \omega]^T$ represents any of the primitive variables and the $f_s(\cdot)$ functions represent sine or cosine functions. Some of the baseline Manufactured solutions for the primitive variables and the specific values for constants are mentioned in Refs. 22 and 23. The above form of the Manufactured Solutions is used for the verification of the baseline governing equations. But for the verification of the boundary conditions, the Manufactured Solutions need to be manufactured such that they exactly satisfy a given boundary condition on a domain boundary. For the verification of the time accuracy of the unsteady flows, the Manufactured Solutions will include a time term as well. The Manufactured Solution used for a 2D unsteady problem is of the form given below.

$$\phi(x, y) = \phi_0 + \phi_t f_s \left(\frac{a_{\phi} \pi}{T} \right) + \phi_x f_s \left(\frac{a_{\phi} \pi x}{L} \right) + \phi_y f_s \left(\frac{a_{\phi} \pi y}{L} \right) + \phi_{xy} f_s \left(\frac{a_{\phi} \pi xy}{L^2} \right)$$

IV. Grid Generation and Systematic Mesh Refinement

The Loci-CHEM finite volume CFD code is verified on different mesh types. In order to verify all local mesh transformations used for the convection and diffusion operators, the code is run on the most general grid types^{3,22} which include grids with mild skewness, aspect ratio, and stretching. A 2D hybrid mesh which includes quadrilateral and triangular cells with curvilinear boundaries and stretched cells is considered as the most general mesh topology for 2D verification. Similarly a 3D hybrid mesh with tetrahedral cells, prismatic cells and hexahedral cells is considered as the most general mesh topology for 3D. The hybrid meshes are generated in a curvilinear domain with non-uniform grid distribution so that the cells contain some skewed and stretched cells.

A. Grid Topologies

The code is verified on the hybrid grids and if the verification tests are failed on the complex hybrid grids, then the code is tested on simpler grids which isolate the effect of grid stretching, aspect ratio and curvilinear grids and also the cell topology. A 2D hybrid grid contains quadrilateral cells and triangular cells and a 3D hybrid grid contains hexahedral cells, tetrahedral cells and prismatic cells. The 2D and 3D hybrid grids considered for the verification of the Loci-CHEM CFD code are shown in Figure 1. The 3D hybrid grid has three different layers of cells, each layer consisting of cells of a particular cell topology. In the figure, the bottom layer consists of hexahedral cells, the top layer consists of prismatic cells and the center layer consists of the tetrahedral cells.

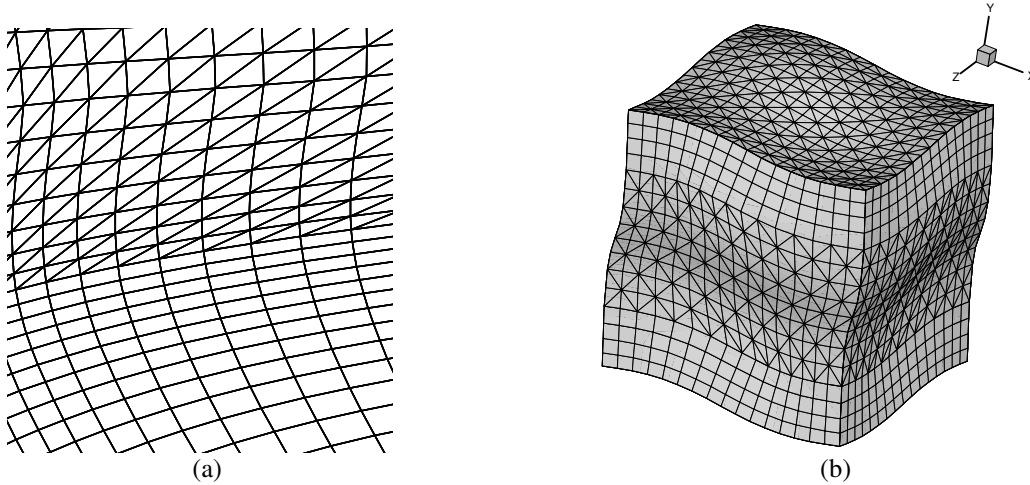


Figure 1 (a) 2D hybrid grid, (b) 3D hybrid grid

Failure of the verification tests on the hybrid grids lead to the testing of the code on structured and unstructured grids separately. In 2D, the code is tested on a curvilinear grid with quadrilateral cells and an unstructured grid with triangular cells. The grid with triangular cells is generated by starting with a structured grid and adding diagonals to quadrilateral cells to make the grid unstructured. Using his procedure a uniform and consistent grid refinement can be achieved for unstructured grids between different grid levels used for the verification purpose. Testing the code separately on the 2D structured and 2D unstructured grids can find the code sensitivities towards the cell topologies. The 2D skewed curvilinear structured grid and the 2D unstructured grid used for testing the finite volume code are shown in Figure 2. The code is tested on even simpler grids²² like the Cartesian, stretched Cartesian and curvilinear grid without skewed cells to isolate the effects of grid stretching, aspect ratio and effect of curved boundaries if the code is not successfully verified on a 2D skewed curvilinear grid.

Similarly in 3D, when the verification test is failed on the 3D hybrid grid, the code is tested on grids with a particular cell topology to test the code sensitivities towards the cell topology. Figure 3 shows the 3D skewed curvilinear grid with hexahedral cells, the 3D unstructured grid with tetrahedral cells and the 3D unstructured grid with prismatic cells. The 3D unstructured mesh with tetrahedral cells is obtained by starting from a structured grid and then splitting each hexahedral cell into five tetrahedral cells. The 3D unstructured mesh with prismatic cells is

obtained by starting with a general 2D unstructured grid and projecting the grid in the third direction normal to the 2D surface. The code is tested on further simpler grids like the 3D Cartesian grid with perfect cubes if required.

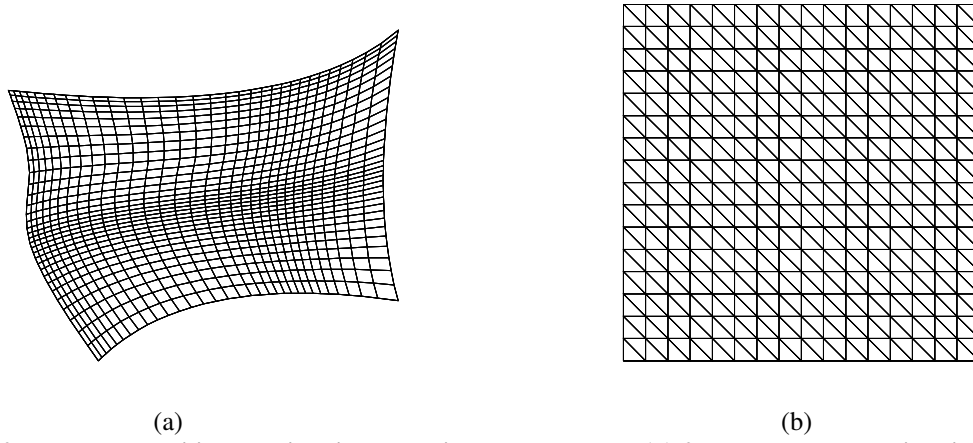


Figure 2 (a) 2D skewed curvilinear grid with quadrilateral cells and, (b) 2D unstructured grid with triangular cells

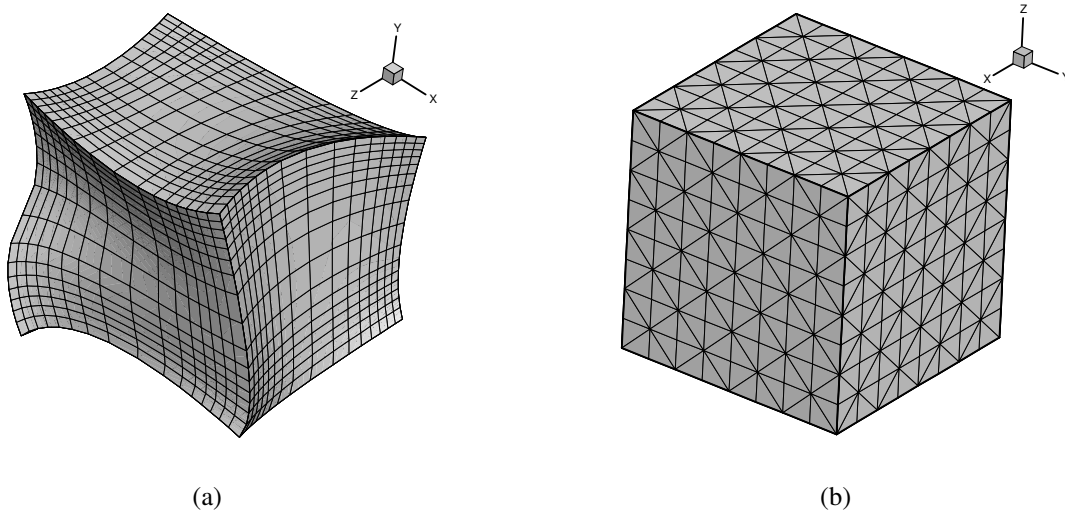


Figure 3 (a) 3D skewed curvilinear structured grid with hexahedral cells, (b) 3D unstructured grid with tetrahedral cells, and (c) 3D unstructured grid with prismatic cells

B. Systematic Mesh Refinement

Systematic mesh refinement³ is defined as uniform and consistent refinement over the spatial domain. A mesh is said to be uniformly refined if the mesh is refined in all the coordinate directions equally and it is said to be consistently refined if the mesh quality stays constant or improves with mesh refinement. For the purpose of code verification, it is necessary to have a systematic mesh refinement.³ In the case of structured grids, refinement/coarsening of the meshes for the verification purpose are straightforward. A coarse mesh is generated from a fine mesh by removing every alternate grid point or grid line to produce grid levels with a refinement factor of two. In the process, the mesh quality can be maintained for the structured meshes. But in the case of unstructured meshes refinement/coarsening of meshes with a uniform refinement factor throughout the domain preserving the mesh quality is more challenging particularly in 3D. In the case of 2D, systematic mesh refinement can be achieved by generating an unstructured mesh from a structured mesh by splitting quadrilaterals into triangles using diagonals.²² To our knowledge, generation of 3D unstructured meshes with uniform refinement preserving the mesh quality cannot be achieved using commercial software. Therefore, for code verification purposes, a grid generation code is developed to generate unstructured tetrahedral meshes from a 3D structured mesh with hexahedral elements. In the process, a hexahedral cell will be split into five tetrahedral cells as shown in Figure 4(a). The unstructured mesh with tetrahedral elements generated using the grid generation code is shown in Figure 4(b). By generating unstructured meshes in this fashion, a uniform and consistent refinement can be achieved by a uniform refinement factor and maintaining the cell quality between the grid levels. Using the concept of generating an unstructured tetrahedral mesh from a structured mesh, another code for generating 3D hybrid grids which contain hexahedral, tetrahedral and prismatic cells with a good connectivity between the different types of cells is also developed. The 3D hybrid grids are generated to satisfy the uniform and consistent refinement criteria between the grid levels.

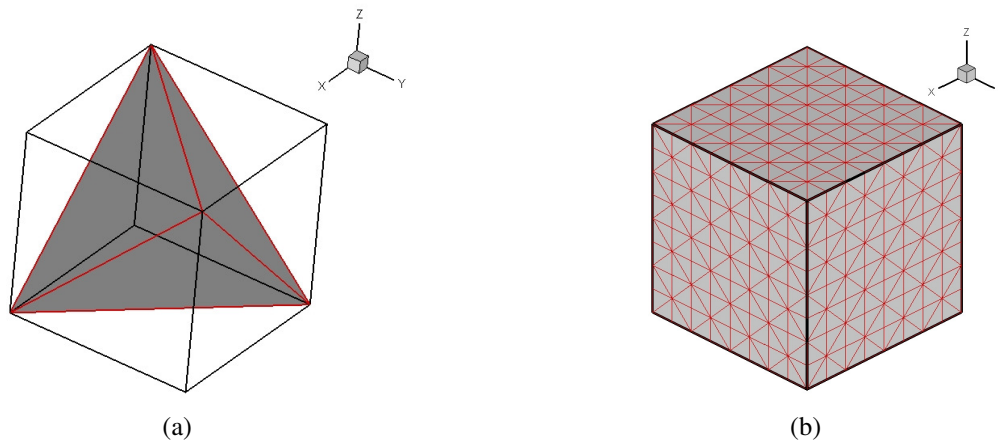


Figure 4 (a) Hexahedral cell split into five tetrahedral cells, (b) 3D unstructured mesh with tetrahedral cells

V. Code Verification

Different code options tested in the Loci-CHEM CFD code include the baseline steady-state governing equations, Sutherland's law for viscosity, thermal conductivity equation, equation of state, boundary conditions (i.e. inflow and outflow boundaries, adiabatic and isothermal no-slip wall, and slip wall boundary conditions), turbulence models (i.e. Spalart-Almaras one equation model²⁵, baseline version of Menter's $k-\omega$ model¹⁸, Menter's SST $k-\omega$ model¹⁸), and time accuracy for unsteady flows. The options in the finite volume CFD code are verified by comparing the observed order of accuracy calculated for the CFD solutions from multiple grid levels to the formal order of accuracy of the chosen numerical methods. An option is considered fully verified if it passes this order of accuracy test on the 3D hybrid grid which has all types of cells with skewness, aspect ratio, and mesh stretching. During the process of code verification, the code options are tested on different grid types and the verification results are shown only on 2D and 3D hybrid grids unless it is required to be verified on other grid types. When the verification test fails on the hybrid grids, then the governing equations are tested on other simpler grids to find whether the discrete formulation of the governing equations is inconsistent on a particular cell topology or on a cell whose quality is affected because of skewness, aspect ratio and curvilinear boundaries.

A. Baseline Governing Equations

Verification of the baseline governing equations includes the testing of the Euler equations and the Navier-Stokes equations on the 2D hybrid and 3D hybrid grids. When the verification test fails on any one of these grids,

then the governing equations are tested on other simpler grids to find whether the discrete formulation of the governing equations is inconsistent on a particular cell topology or on a cell whose quality is affected because of skewness, aspect ratio and curvilinear boundaries. Successful verification of the Euler equations on a particular grid and failure of the order of accuracy test for the Navier-Stokes equations means that there is a mistake in the formulation of the diffusion operator on that particular grid. Mostly successful verification of Navier-Stokes equations on a particular grid topology ensures that the Euler equations are also successfully verified on the same grid topology since Euler equations are only a subset of the Navier-Stokes equations.

The Navier-Stokes equations are verified on the 2D and 3D hybrid grids and the observed order of accuracy approaches two for the numerical scheme with grid refinement on both 2D and 3D hybrid grids. Six grid levels of the 2D hybrid grid are used to run the Navier-Stokes equations and the grid generation code developed to generate the 3D hybrid grids is used to obtain four grid levels in the case of 3D grids. The observed order of accuracy is calculated using L_2 , L_1 and L_∞ norms of the discretization error and checked whether the observed order approaches two when calculated using all the three types of norms. During the verification of the finite volume code, normally the observed order of accuracy calculated using L_2 and L_1 norms has similar behavior and the observed order of accuracy calculated using L_∞ norm asymptotes to the formal order at a slower rate requiring more grids levels to see the asymptotic behavior. In this paper, the observed order of accuracy results shown are calculated using the L_2 norms but the order of accuracy results using the L_∞ norms also approached the formal order. The observed order of accuracy results calculated using the L_2 norm discretization error for the 2D hybrid grid and the 3D hybrid grid are shown in Figure 5(a) and Figure 5(b) respectively. In these plots the calculated order of accuracy ‘p’ is on the y-axis and grid refinement ratio ‘h’ is on the x-axis. The value of ‘h’ equal to one represents the finest grid and hence in these plots the order of accuracy approaches two as the ‘h’ value approaches one.

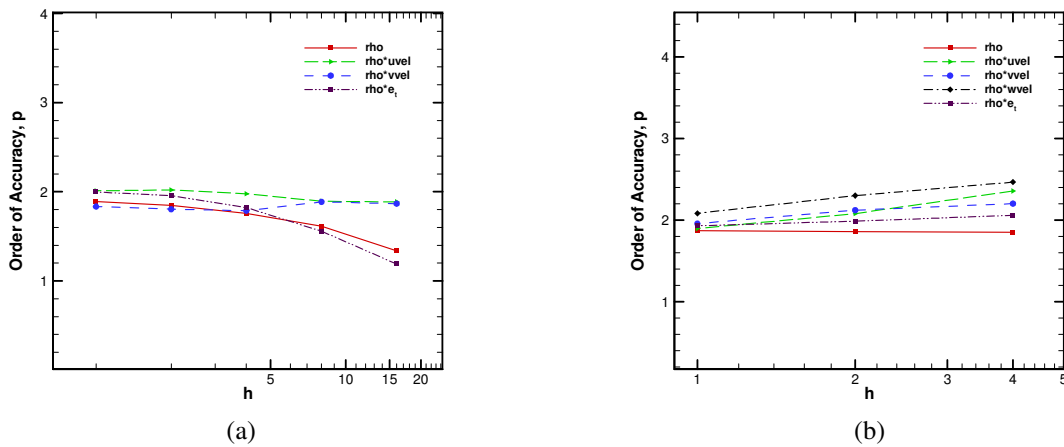


Figure 5 Order of accuracy results for the Navier-Stokes equations using L_2 norm of the discretization error on the (a) 2D hybrid grid and (b) 3D hybrid grid

The Navier-Stokes equations are also tested on the skewed 3D hybrid grid shown in Figure 6(a). The skewed 3D hybrid grid is generated starting from a skewed 3D curvilinear grid shown in Figure 3(a) and using the grid generation code to obtain the skewed 3D hybrid grid. The Navier-Stokes equation are successfully verified to be second order accurate on 3D skewed curvilinear grid with hexahedral cells shown in Figure 3(a), but the verification test failed on the skewed 3D hybrid grid. The observed order of accuracy result for the Navier-Stokes equation on the skewed 3D hybrid grid is shown in Figure 6(b). From the plot, the observed order of accuracy seems to approach to a value less than one with grid refinement. The difference between the 3D hybrid grid (Fig. 1b) and the skewed 3D hybrid grid (Fig. 6a) is only the quality of the cells in the grid; otherwise both the grids have the same grid structure. This indicates a problem in the discrete formulation of the governing equations when the cells are skewed. The L_2 norm of the discretization error for both the 3D hybrid grid and skewed 3D hybrid grid is compared and it is observed that the error is higher for the skewed hybrid grids relative to the hybrid grids for the same number of cells and similar grid structure. The comparison of the L_2 norm of the discretization error is shown in Figure 7. In the plot, errors shown in the solid lines correspond to the 3D hybrid grid and the errors shown in dashed line corresponds to the skewed 3D hybrid grid. This plot explains the effect of cell quality on the error in the solution. The error in the solution either decreases slowly or does not decrease with grid refinement for lower quality grids.

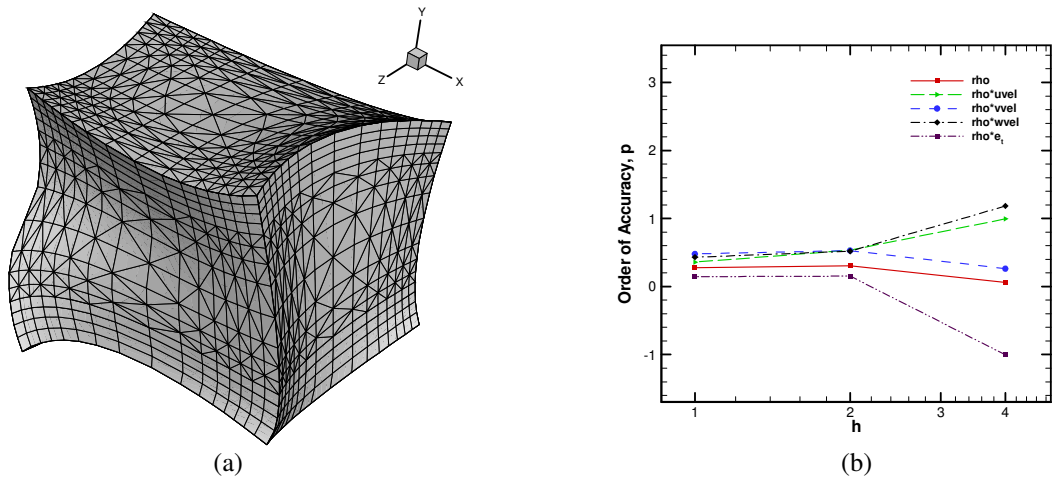


Figure 6 (a) Skewed 3D hybrid grid, and (b) Observed order of accuracy results for the Navier-Stokes equations using L2 norm of the discretization error

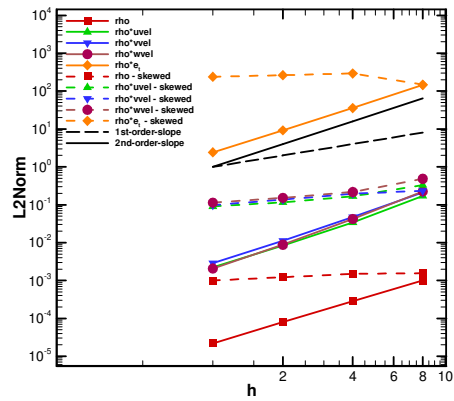


Figure 7 Comparison of discretization error on 3D hybrid grid (Fig. 1b) and skewed 3D hybrid grid (Fig. 6a)

To further study the discrete formulations of the inviscid and viscous terms in the governing equations, the Euler and Navier-Stokes equations are tested separately on grids with a particular grid topology, i.e., either skewed tetrahedral cells or skewed prismatic cells alone. The above analysis explains how the Method of Manufactured Solutions along with the order of accuracy test can be used to find mistakes in the code or the inconsistencies in the discrete formulations by testing on different grids with different cell topologies and different cell quality.

The Navier-Stokes equations are also tested on skewed 2D hybrid grids to look at the effect of cell quality of quadrilateral cells and triangular cells on the discrete formulation of the governing equations. A skewed 2D hybrid grid is shown in Figure 8(a). On the skewed hybrid grid in 2D, the code is successfully verified and the observed order of accuracy approaches two with grid refinement which is shown in Figure 8(b). This shows that the finite volume code works fine on a skewed quadrilateral and triangular cells in 2D but fails the order of accuracy test on skewed tetrahedral and prismatic cells in 3D.

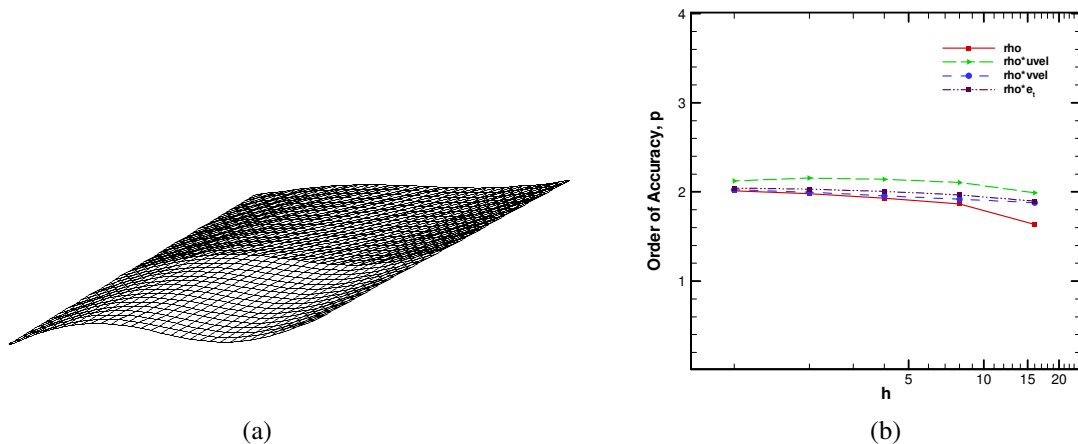


Figure 8 (a) Skewed 2D hybrid grid and (b) observed order of accuracy results for the Navier-Stokes equations using L2 norm of the discretization error

Other equations verified in the finite volume code are thermal conductivity equation, equation of state and Sutherland's law for viscosity. Verifying Navier-Stokes equations in the Loci-CHEM code automatically verifies equation of state. For thermal conductivity equation and Sutherland's law of viscosity, thermal conductivity and viscosity are defined as functions of temperature. Both the equations are tested on 3D hybrid grid and are verified as second order accurate with the observed order of accuracy approaching two with grid refinement.

B. Boundary Conditions

In order to verify the implementation of a boundary condition in a code, the manufactured solution can be tailored to exactly satisfy a given boundary condition on a domain boundary. A general approach for tailoring manufactured solutions to ensure that a given boundary condition is satisfied along a domain boundary was developed by Bond et al.²⁶ The method involves multiplying any standard manufactured solution by a function which has values and/or derivatives which are constant over a specified boundary. The boundary condition options in the Loci-CHEM code are tested on 2D hybrid grids and 3D hybrid grids. For the 2D hybrid grid, a well defined curved boundary on one of the four sides is considered as a no-slip wall and for the 3D hybrid grid a well defined wavy surface on one side of the domain is considered as no-slip wall boundary. The 2D hybrid grid with the bottom curved boundary defined as a no-slip wall is shown in Figure 9(a) and the wavy surface defined as a no-slip wall in a 3D hybrid grid is shown in Figure 9(b). The no-slip wall boundary is tested as an adiabatic and an isothermal wall.

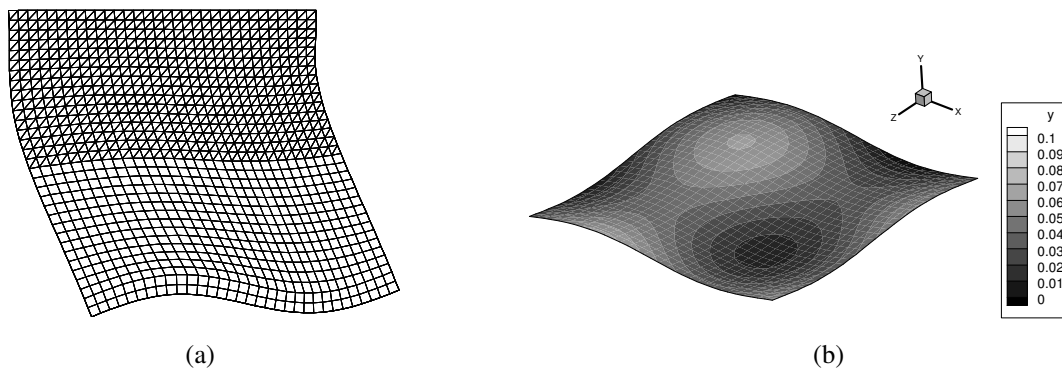


Figure 9 (a) 2D hybrid grid with the bottom curved boundary as no-slip wall and (b) the bottom boundary in a 3D hybrid grid used as a no-slip wall

By testing the no-slip wall as an adiabatic boundary, a Neumann boundary condition, ($dT/dn = 0$) is verified along with the no-slip condition ($V=0$) on a particular boundary. The no-slip wall is tested as an adiabatic boundary on both 2D hybrid and 3D hybrid grids and the observed order of accuracy calculated from the numerical solutions approaches two with grid refinement. The order of accuracy results calculated using the L2 norm of the

discretization error when the no-slip wall tested as adiabatic boundary on 2D hybrid grid and 3D hybrid grid are shown in Figure 10(a) and Figure 10(b), respectively. Similarly, the no-slip wall is tested as an isothermal boundary and which means the testing of a Dirichlet boundary condition, ($T = \text{constant}$) along with the no-slip condition ($V=0$) on a particular boundary. The observed order of accuracy for this case approaches two with grid refinement for both the 2D hybrid and 3D hybrid grids. The order of accuracy results calculated using the L2 norm of the discretization error when the no-slip wall is an isothermal boundary on 2D hybrid grid and 3D hybrid grid are shown in Figure 11(a) and Figure 11(b) respectively.

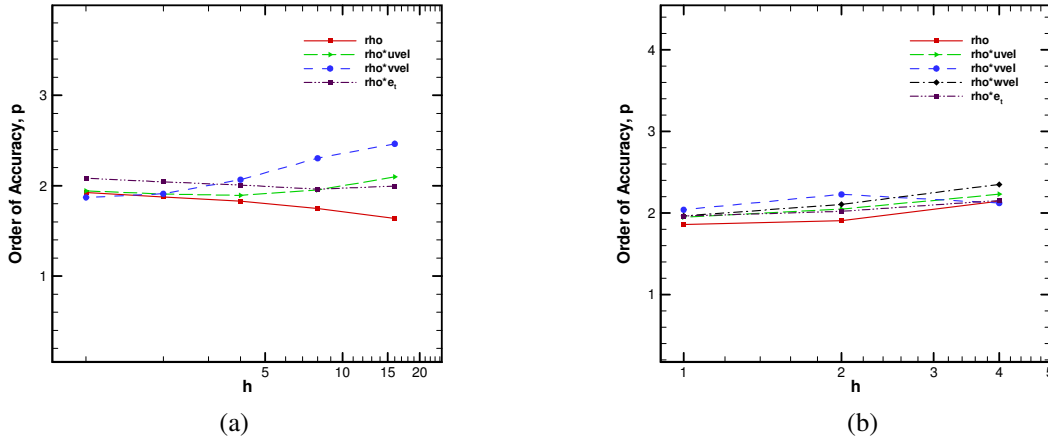


Figure 10 of accuracy results using L2 norm of the discretization error for adiabatic no-slip wall boundary on (a) 2D hybrid grid and (b) 3D hybrid grid

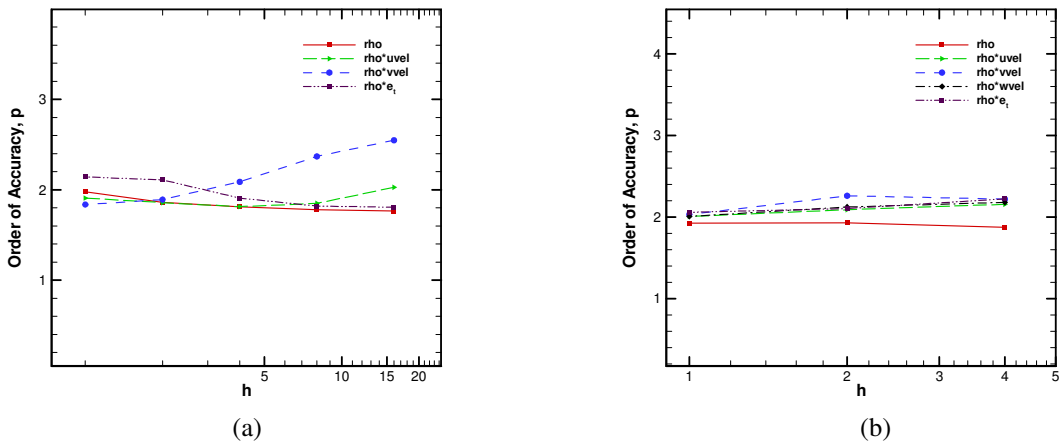


Figure 11 Order of accuracy results using L2 norm of the discretization error for isothermal no-slip wall boundary on (a) 2D hybrid grid and (b) 3D hybrid grid

Testing the slip wall boundary is verifying the slip condition $V_n = 0$ or $V_n = 0$ on a particular boundary where V_n is the velocity component normal to the surface. The slip wall boundary condition option in the finite volume code is tested and found to be inconsistent even on a 2D curvilinear grid with a curved boundary defined as slip wall. The same slip wall option is tested to be second order accurate on the 2D curvilinear grid when the governing equations used are Euler equations rather than Navier-Stokes equations. Implementation of slip wall boundary condition in the Loci-CHEM code of Navier-Stokes equations is currently under investigation. Other boundary condition options tested in the finite volume code include the constant velocity inflow boundary condition and the farfield boundary condition which is an inflow-outflow characteristic based boundary condition suitable for farfield condition in external flow for both supersonic and subsonic flow conditions. Both the boundary condition options are tested on the 3D hybrid grid and are successfully verified to be second order accurate.

C. Turbulence Models

Verification of RANS turbulence models provides additional challenges for MMS for different reasons.²³ One of the reasons is that the turbulence models often employ **min** or **max** functions to switch from one behavior to another, thus causing the source terms to no longer be continuously differentiable. Our approach²³ selects manufactured solutions such that they will only activate one branch of the **min** and **max** functions for a given manufactured solution. The turbulence models tested in the finite volume code are the baseline version of Menter's k- ω model and the Menter's Shear Stress Transport k- ω model. In both the turbulence models, there are two branches to be verified, i.e., the k- ω branch which get activated in the boundary layer region the k- ϵ branch which get activated away from the wall boundaries in free shear layers. See Ref. 23 for additional details.

For the baseline version of Menter's k- ω turbulence model, by setting the blending function, $F_1 = 0$, the model is tested for the k- ϵ branch of the Menter BSL model and in this case, the wall distance d is automatically set to infinity. By setting the wall distance $d = 1 \times 10^{-6}$, the blending function, $F_1 = 1$, and the k- ω branch of the Menter BSL model is tested. The k- ω branch of the Menter BSL model is tested on the 3D hybrid grid and the observed order of accuracy approached two with grid refinement. The order of accuracy result for the k- ω branch of the Menter BSL model on the 3D hybrid grid is shown in Figure 12(a). During the testing of k- ϵ branch, a problem with the ω -equation is observed and the discretization error for that equation did not decrease at the expected rate and the order of accuracy dropped to zero with grid refinement, but all the other equations approached two with grid refinement. The behavior of order of accuracy with grid refinement for the k- ϵ branch of the Menter BSL model on the 3D hybrid grid is shown in Figure 12(b).

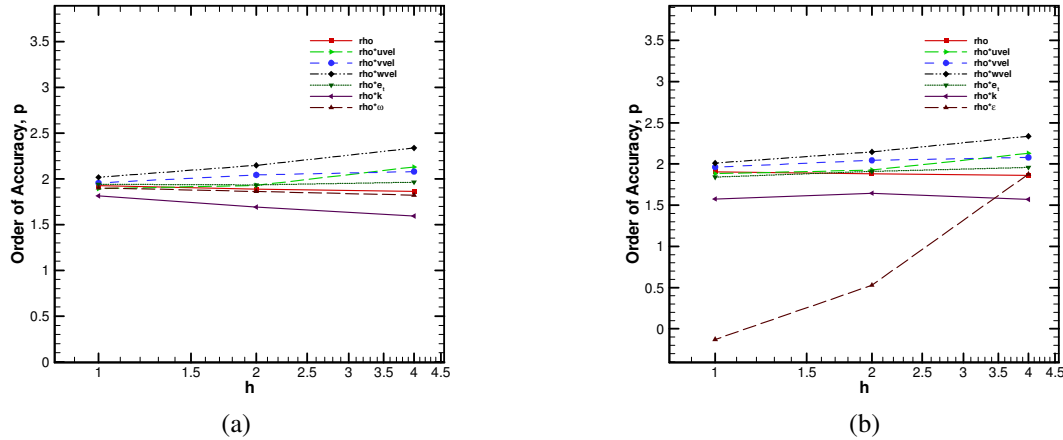


Figure 12 Order of accuracy results for the Menter BSL k- ω turbulence model on the 3D hybrid grid with (a) the k- ω branch activated ($F_1 = 1$) and (b) the k- ϵ branch activated ($F_1 = 0$)

To explore the reason for the failure of the verification test for the k- ϵ branch of the Menter BSL model on the 3D hybrid grid, it is tested on simpler grids. Initially the k- ϵ branch of the Menter BSL model is tested on the 2D hybrid grid and it is observed that the verification is successful with all the norms of the discretization errors approaching two with grid refinement. The order of accuracy result for the k- ϵ branch of the Menter BSL model on the 2D hybrid grid is shown in Figure 13(a). Then the above test is also done on a skewed 3D curvilinear grid with hexahedral cells and the k- ϵ branch of the Menter BSL model is successfully verified with all the norms of the discretization error approaching two with grid refinement. The order of accuracy result for the k- ϵ branch of the Menter BSL model on the skewed 3D curvilinear grid is shown in Figure 13(b).

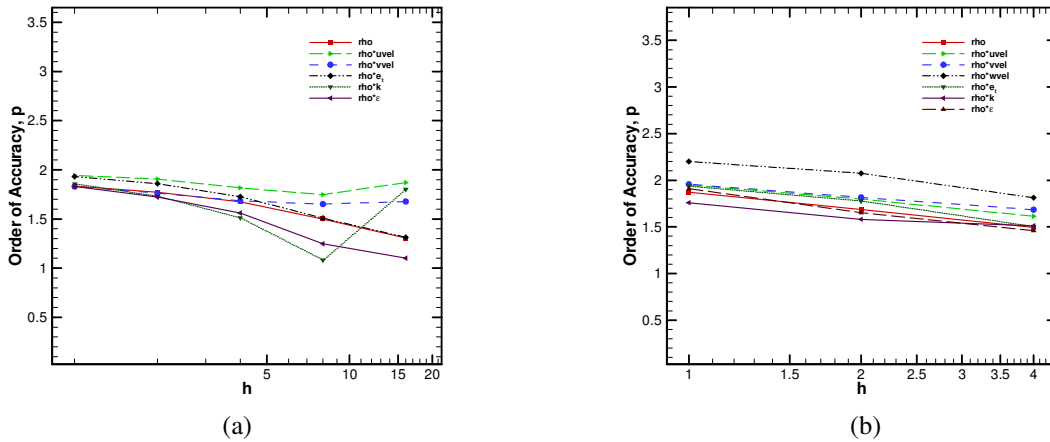


Figure 13 Order of accuracy results for the $k-\epsilon$ branch ($F_1 = 0$) of the Menter BSL $k-\omega$ turbulence model on (a) the 2D hybrid grid and (b) the skewed 3D curvilinear grid with hexahedral cells

Further the $k-\epsilon$ branch of the Menter BSL model is tested on a 3D unstructured grid with tetrahedral cells and it is observed that there is a similar behavior when compared with the 3D hybrid grid and the discretization error for the ω -equation only does not decrease at the expected rate and the order of accuracy dropped to zero with grid refinement. The 3D unstructured grid with tetrahedral cells and the order of accuracy result for the $k-\epsilon$ branch of the Menter BSL model is shown in Figure 14. Testing on different grids, it can be concluded that there is a mistake in the discrete formulation of some of the terms in the ω -equation as it works fine only on 2D grid topologies and 3D structured grid topologies like the hexahedral cells and it fails on the 3D unstructured grid topologies like the tetrahedral cells. From the above analysis the mistake is isolated to the cross-diffusion term in the ω -equation. This analysis holds true even for the $k-\epsilon$ branch of the Menter SST $k-\omega$ turbulence model. The verification procedure for the $k-\omega$ branch of the Menter SST $k-\omega$ turbulence model is still under investigation.

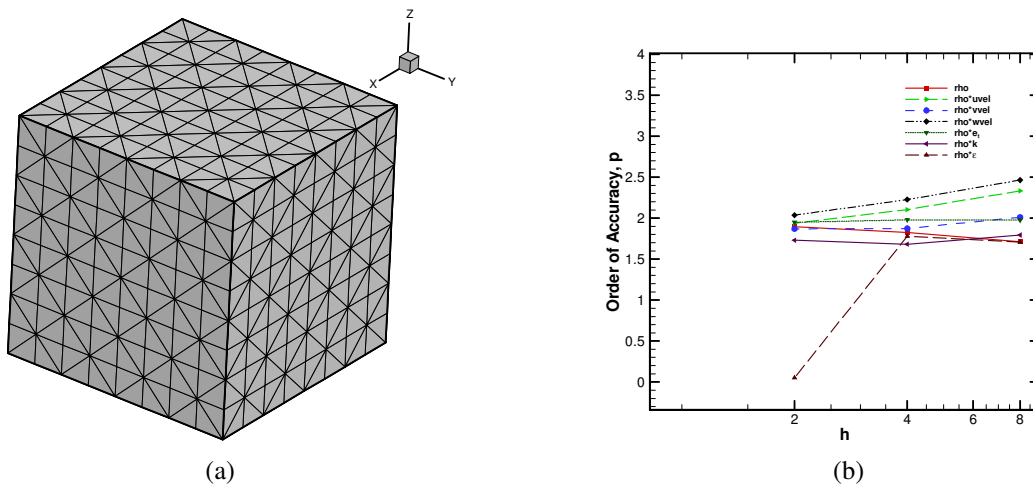


Figure 14 (a) 3D unstructured grid with tetrahedral cells and (b) Observed order of accuracy for $k-\epsilon$ branch of the Menter BSL $k-\omega$ turbulence model

Other turbulence model verified include the Spalart-Almaras turbulence model which is verified on the 3D unstructured mesh and on 3D structured curvilinear mesh using a statistical approach to MMS.¹⁰ In this approach, a single grid is considered and shrunk down providing a locally refined grid and this procedure is used to statistically sample the discretization error in different regions of the domain of interest. Using a statistical approach, the Spalart-Almaras turbulence model is successfully verified to be second order accurate.

D. Time Accuracy of Unsteady Flows

It is more difficult to apply the verification procedure using the order of accuracy test to problems that involve both spatial and temporal discretization, especially when the spatial order of accuracy is different from the temporal order. A combined spatial and temporal order verification method has been developed by Kamm et al.²⁴ In their approach, they use a Newton-type iterative procedure to solve a coupled, nonlinear set of algebraic equations to calculate the coefficients in the spatial and temporal terms and hence the spatial and temporal accuracies. In the present approach, it does not require the solution to a system of nonlinear algebraic equations. Initially, a spatial grid refinement study is performed with a fixed time step followed by a temporal refinement study on a fixed grid. With all the coefficients calculated, the spatial step size and the temporal step size can be chosen such that the spatial discretization error term has the same magnitude as the temporal discretization error term. Once these two terms are approximately the same order of magnitude, combined spatial and temporal order verification is conducted by choosing the temporal refinement factor such that the temporal error term drops by the same order of magnitude as the spatial error term with refinement, i.e., $r_t = r_x^{p/q}$. r_t is the temporal refinement factor, r_x is the spatial refinement factor, p is the spatial order and q is the temporal order. In our case, $p = q = 2$. Using this procedure, the unsteady time term is verified on the 2D hybrid grid and the 3D hybrid grid. The observed order of accuracy on both the grids approached two with grid refinement. The order of accuracy results for the time accuracy of the unsteady flows on 2D hybrid grid and 3D hybrid are shown in Figure 15(a) and Figure 15(b) respectively.

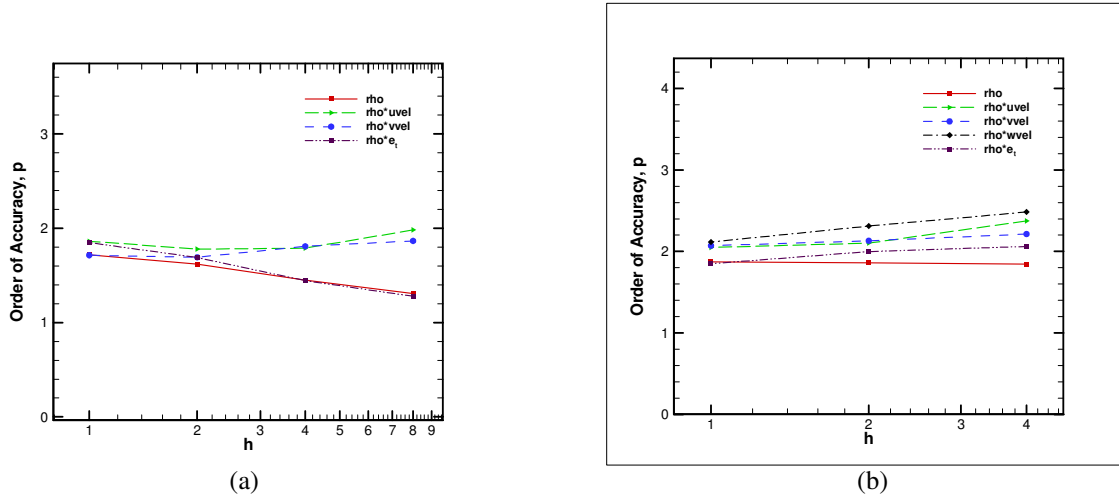


Figure 15 Order of accuracy results for the unsteady time marching term on (a) the 2D hybrid grid and (b) the 3D hybrid grid

A summary of the several options verified in the finite volume Loci-CHEM CFD code is shown in tabular form in Figure 16. The options are successfully verified when they are second order accurate on the most complex grid i.e., the 3D hybrid grid. An option can be assumed verified on a particular grid even though it is not actually run on that grid, if that option is already verified on a complex element type.

| | | 2D Structured | | | 2D Unstructured | | 2D Hybrid | 3D Structured | | 3D Unstructured | 3D Hybrid |
|---------------------|---------------------------------------|-----------------------|---------------------|-------------|-----------------|--------------------|-----------|---------------|-------------|------------------------------|------------------------------|
| | | Rectangular Cartesian | Stretched Cartesian | Curvilinear | Unidirectional | Alternate Diagonal | | Cube | Curvilinear | | |
| | Euler (single species) | | | | | | | | | | |
| | Navier-Stokes (single species) | | | | | | | | | | |
| | Thermally Perfect Thermodynamic Model | | | | | | | | | | |
| | Sutherland | | | | | | | | | | |
| Turbulence Models | Spalart-Allmaras One-equation Model | | | | | | | | | | |
| | Menter BSL k-omega | | | | | | | | | k-omega branch only verified | k-omega branch only verified |
| | Menter SST k-omega | | | | | | | | | | |
| Boundary Conditions | Dirichlet | | | | | | | | | | |
| | No-Slip Adiabatic (Solid Wall) | | | Wavy wall | Wavy wall | | Wavy wall | | Wavy wall | | |
| | No-Slip Isothermal (Solid Wall) | | | Wavy wall | Wavy wall | | Wavy wall | | Wavy wall | | |
| | Slip Wall (Euler) | | | | | | | | | | |
| | Inflow | | | | | | | | | | |
| | Farfield | | | | | | | | | | |
| | Unsteady Time Marching | | | | | | | | | | |

† Cases that were not actually run, but that are assumed to be verified because a complex element type was verified

| | | | |
|--|--|--|--|
| | (Assumed Verified) † | | Verified 2 nd Order - Standard MMS |
| | Verified 2 nd Order - Statistical MMS | | Verified 2 nd Order - Both Statistical and Standard MMS |

Figure 16 Verification of different options in the finite volume Loci-CHEM CFD code

VI. Conclusions

The comprehensive code verification of an unstructured finite volume code was presented. The Method of Manufactured Solutions was used to generate exact solutions. Different options in the finite volume CFD code were verified which included the baseline governing equations, different boundary condition options, turbulence models and time accuracy of unsteady flows. All the options were determined to be verified when the observed order of accuracy matched the formal order on the 2D hybrid and 3D hybrid grid which contained all cell topologies. When the verification process failed on any one of these complex hybrid grids, then simpler grids were considered to isolate the coding mistakes or algorithm inconsistencies.

VII. Acknowledgements

This work is supported by the National Aeronautic and Space Administration's Constellation University Institutes Program (CUIP) with Claudia Meyer and Jeffrey Ryback of NASA Glenn Research Center serving as program managers and Kevin Tucker and Jeffrey West of NASA Marshall Space Flight Center serving as technical monitors.

References

1. Roache, P. J., *Verification and Validation in Computational Science and Engineering*, Hermosa Publishers, New Mexico, 1998.
2. Roy, C. J., "Review of Code and Solution Verification Procedures in Computational Simulation," *Journal of Computational Physics*, Vol. 205, No. 1, 2005, pp. 131-156.
3. Oberkampf, W. L. and Roy, C. J., *Verification and Validation in Scientific Computing*, Cambridge University Press, New York, to appear 2010.
4. Roache, P. J. and Steinberg, S., "Symbolic Manipulation and Computational Fluid Dynamics," *AIAA Journal*, Vol. 22, No. 10, 1984, pp. 1390-1394.

5. Roache, P. J., "Code Verification by the Method of Manufactured Solutions," *Journal of Fluids Engineering*, Vol. 124, No. 1, 2002, pp. 4-10.
6. Knupp, P. and Salari, K., *Verification of Computer Codes in Computational Science and Engineering*, Rosen, K.H. (ed), Chapman and Hall/CRC, Boca Raton, FL, 2003
7. 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 for Numerical Methods in Fluids*, Vol. 44, No. 6, 2004, pp. 599-620.
8. Smith, T. M., Ober, C. C., Lorber, A. A., "SIERRA/Premo – A New General Purpose Compressible Flow Simulation Code, AIAA Paper 2002-3292, 2002.
9. Nelson, C. C. and Power, G. D., "CHSSI Project CFD-7: The NPARC Alliance Flow Simulation System," AIAA Paper 2001-0594, 2001.
10. Hebert, S. and Luke, E., "Honey, I Shrunk the Grids! A New Approach to CFD Verification," AIAA Paper 2005-0685, 2005.
11. Luke, E. A., Tong, X.-L., Wu, J., and Cinnella, P., "CHEM 2: A Finite-Rate Viscous Chemistry Solver – The User Guide," Tech. Rep. MSSU-COE-ERC-04-07, Mississippi State University, 2004.
12. Pelletier, D. and Roache, P. J., "CFD Code Verification and the Method of the Manufactured Solutions," 10th Annual Conference of the CFD Society of Canada, Windsor, Ontario, Canada, June 2002.
13. Pelletier, D., Turgeon, E., and Tremblay, D., "Verification and Validation of Impinging Round Jet Simulations using an Adaptive FEM," *International Journal for Numerical Methods in Fluids*, Vol. 44, 2004, pp. 737-763.
14. Eca, L. and Hoekstra, M., "An Introduction to CFD Code Verification Including Eddy-Viscosity Models," European Conference on Computational Fluid Dynamics, ECCOMAS CFD 2006, Wesseling, P., Onate, E., and Periaux, J. (eds.), 2006.
15. Eca, L. and Hoekstra, M., "Verification of Turbulence Models with a Manufactured Solution," European Conference on Computational Fluid Dynamics, ECCOMAS CFD 2006, Wesseling, P., Onate, E., and Periaux, J. (eds.), 2006.
16. Eca, L., Hoekstra, M., Hay, A., and Pelletier, D., "On the Construction of Manufactured Solutions for One and Two-Equation Eddy-Viscosity Models," *International Journal for Numerical Methods in Fluids*, Vol. 54, No. 2, May 2007, pp. 119-154.
17. Spalart, P. R., and Allmaras, S. R., "A One-Equation Turbulence Model for Aerodynamic Flows," AIAA Paper 92-0439, 1992.
18. Menter, F. R., "Two-Equation Eddy-Viscosity Turbulence Models for Engineering Applications," *AIAA Journal*, Vol. 32, No. 8, 1994, pp. 1598-1605.
19. Kok, J. C., "Resolving the Dependence on Free-Stream Values for the $k-\omega$ Turbulence Model," NLR-TP-00205, 1999.
20. Luke, E., "Loci: A Deductive Framework for Graph-Based Algorithms," Third International Symposium on Computing in Object-Oriented Parallel Environments, S. Matsuoka, R. Oldehoeft, and M. Tholburn (eds.), No. 1732 in Lecture Notes in Computer Science, Springer-Verlag, December 1999, pp. 142-153.
21. Wilcox, D. C., *Turbulence modeling for CFD*, 2nd ed., DCW Industries, La Canada, CA, 1998.
22. Veluri, S. P., Roy, C. J., Hebert, S., and Luke, E. A., "Verification of the Loci-CHEM CFD Code using the Method of Manufactured Solutions," AIAA-2008-661, 46th AIAA Aerospace Sciences Meeting and Exhibit, January 2008.
23. Roy, C. J., Tendean, E., Veluri, S. P., Rifki, R., Luke, E. A., and Hebert, S., "Verification of RANS Turbulence Models the Loci-CHEM using the Method of Manufactured Solutions," AIAA-2007-4203, 18th AIAA Computational Fluid Dynamics Conference, June 2007.
24. Kamm, J. R., Rider, W. J., and Brock, J. S., "Combined Space and Time Convergence Analyses of a Compressible Flow Algorithm," AIAA-2003-4241, 16th AIAA Computational Fluid Dynamics Conference, June 2003.
25. Spalart, P. R., and Allmaras, S. R., "A One-Equation Turbulence Model for Aerodynamic Flows," AIAA-92-0439, 30th Aerospace Sciences Meeting and Exhibit, Jan 1992.
26. Bond, R. B., Ober, C. C., Knupp, P. M., and Bova, S. W., "Manufactured Solution for Computational Fluid Dynamics Boundary Condition Verification," *AIAA Journal*, Vol. 45, No. 9, 2007, pp. 2224-2236.