

Verification of RANS Turbulence Models in Loci-CHEM using the Method of Manufactured Solutions

Christopher J. Roy,^{*} Eric Tendeau,[†] Subramanya P. Veluri,[‡] and Rifki Rifki[§]
Aerospace Engineering Department, Auburn University, Alabama

Edward A. Luke^{**} and Shelley Hebert^{††}
Simulation and Design Center, Mississippi State University, Mississippi

Abstract

An approach for verifying Reynolds-Averaged Navier-Stokes (RANS) turbulence models in Computational Fluid Dynamics (CFD) codes is presented. This approach relies on smooth, non-physical Manufactured Solutions which are chosen so as to provide contributions from all of the terms in the turbulence transport equations including convection, diffusion, production, and destruction. The Loci-CHEM CFD code is employed to solve the steady-state, compressible RANS equations in two-dimensions. The turbulence model verified is the baseline version of Menter's $k-\omega$ model. Special attention is paid to the blending function which allows the model to switch between a $k-\omega$ and a transformed $k-\epsilon$ model. Results are presented for the observed order of accuracy on families of Cartesian structured grids, skewed curvilinear structured grids, and unstructured triangular grids. The Manufactured Solutions clearly identify problems on skewed meshes which cause the diffusion operator to become inconsistent (i.e., the discretization error does not decrease with mesh refinement). An alternative formulation for the diffusion operator is implemented and the Manufactured Solutions are shown to converge in a second-order manner with mesh refinement for the structured grids. Preliminary investigations on the unstructured meshes indicate that while the code is still consistent, the observed order of accuracy is reduced to first order.

Nomenclature

$CD_{k\omega}$	cross-diffusion term in the turbulence frequency transport equation
c_p	specific heat at constant pressure
c_v	specific heat at constant volume
DE_k	discretization error on mesh level k ; $k=1,2,3,\dots$
d	normal distance to the nearest no-slip wall
e	Favre-averaged internal energy
e_t	Favre-averaged total energy
F	Menter blending function
F_1	Menter blending function for the baseline model
f_{exact}	exact solution to the partial differential equations
f_k	numerical solution to the partial differential equations on mesh level k ; $k=1,2,3,\dots$
h	Favre-averaged enthalpy
h_f	heat of formation

^{*}Assistant Professor, Associate Fellow AIAA.

[†]Graduate Research Assistant, Student Member AIAA.

[‡]Graduate Research Assistant, Student Member AIAA.

[§]Graduate Research Assistant, Student Member AIAA.

^{**}Assistant Professor, Member AIAA.

^{††}Graduate Research Assistant, Student Member AIAA.

h_t	Favre-averaged total enthalpy
h_{ref}	reference heat of formation; $h_{ref} = 0.0 J$
h_k	measure of the element size on mesh level k ; $k=1,2,3,\dots$
g_p	coefficient of the leading error term
k	turbulent kinetic energy
L	reference length
N	number of cells
n	one half the number of fully-excited molecular energy modes; $n = 5/2$
P	turbulence production term
p	Favre-averaged pressure, also observed order of accuracy
Pr	laminar Prandtl number
Pr_T	turbulent Prandtl number
q_{L_i}	laminar heat flux vector
q_{T_i}	turbulent heat flux vector
R	specific gas constant
r	ratio of coarse to fine grid element size
T	Favre-averaged temperature
T_{ref}	reference temperature; $T_{ref} = 298 K$
t_{ij}	laminar stress tensor
u, v	Cartesian velocity components
x, y	Cartesian coordinates
ρ	Reynolds-averaged density
τ_{ij}	turbulent stress tensor for the mean flow equations
τ'_{ij}	full compressible turbulent stress tensor
μ	laminar absolute viscosity
μ_T	turbulent absolute viscosity
ω	turbulence frequency

I. Introduction

COMPUTATIONAL Fluid Dynamics, or CFD, is playing an ever-increasing role in the design, analysis, and optimization of aerospace systems. It is therefore critical that decision makers have confidence in the correctness of the CFD predictions. Verification and validation are a set of formal procedures by which one builds confidence in the results of models and simulations, including CFD.¹ Verification addresses the mathematical correctness of the simulations, while validation addresses the physical correctness of the underlying models and equations through comparisons with experimental data. This paper addresses aspects of the mathematical correctness of CFD simulations, i.e., verification.

There are two fundamental aspects to verification: code verification and solution verification.^{2,3} 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. 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 refined. This procedure thus requires the existence of an exact solution to the governing equations which, ideally, exercises all of the terms in the equations. 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.

One of the chief difficulties in verifying a code is identifying exact solutions to the governing equations which exercise all terms in the equations. 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.^{2,3} 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. For code verification purposes, it is not required (in fact, often not desirable)

that the Manufactured Solution be related to a physically realistic problem; recall that verification deals only with the mathematics of a given problem. The general concept behind MMS is to choose the solution *a priori*, then operate the governing partial differential equations onto the chosen solution, thereby generating analytical source terms which require no discretization. The chosen (manufactured) solution is then the exact solution to the modified governing equations made up of the original equations plus the analytical source terms. Thus, MMS involves the solution to the backward problem: given an original set of equations and a chosen solution, find a modified set of equations that the chosen solution will satisfy. The initial and boundary conditions are then determined from the Manufactured Solution.

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 procedure have been presented by Roache.^{2,5} 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 which is also the subject of the current paper. In their approach, they employ a single grid level which is shrunk down (thus providing a 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.

While MMS has begun to see widespread use for code verification of laminar CFD codes, only a handful of researchers have recently begun to address CFD codes with Reynolds-Averaged Navier-Stokes (RANS) turbulence models. These RANS turbulence models provide additional challenges for MMS for a number of reasons. First, they typically contain strong, highly-nonlinear source terms. Second, when physically-based Manufactured Solutions are employed which mimic actual near-wall turbulence, some turbulence quantities exhibit singular behavior (e.g., the wall-limiting behavior of the turbulence frequency ω often varies as the inverse of the wall distance squared and is thus infinite at the wall). Third, turbulence models often employ \min or \max functions to switch from one behavior to another, thus leading to “kinks” in the source terms since these functions are not continuously differentiable. Even when a turbulence model does not employ such functions, they are often found in the discrete implementation of a turbulence model such as the common practice of limiting the turbulence production to some multiple of the dissipation rate, although such ad hoc modifications are rarely discussed in the literature.

There have been two 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 $\tilde{\nu}_t$, 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 differs from the previous work in a 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 that 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.

The prior work allowed solutions which could select different branches of the \min and \max functions in different spatial regions, thus leading to slope discontinuities in some of the turbulence parameters and source terms. In some cases, they even removed the \min and \max functions altogether from the governing equations. Our approach selects manufactured solutions and assigned wall distance functions which, although not physically realistic, will only activate one branch of the \min and \max functions for a given Manufactured Solution. Finally, the previous work examined different equations or terms separately by replacing certain terms or equations with their analytical counterpart; whereas we simply turn off certain terms in both the numerical code and the manufactured solution to focus on different parts of the equations.

II. Governing Equations

The CFD code employed in the current work is called 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 doesn't allow dependencies on uninitialized variables (one of the most common dynamic faults in software engineering) and it automatically handles domain decomposition and parallelization. Our current efforts consider only two-dimensional steady flows of a perfect gas. The turbulence model to be verified is the Baseline (BSL) version of Menter's $k-\omega$ model. This model combines the accuracy of the $k-\omega$ model for wall-bounded flows with the standard $k-\varepsilon$ model for free shear flows, thus removing the $k-\omega$ model's dependence on the freestream ω value.

A. Flow equations

The 2D, steady-state, Favre-averaged Navier-Stokes equations²¹ can be written as

$$\begin{aligned} \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} &= 0 \\ \frac{\partial(\rho u^2)}{\partial x} + \frac{\partial(\rho uv)}{\partial y} + \frac{\partial p}{\partial x} - \frac{\partial}{\partial x}[t_{xx} + \tau_{xx}] - \frac{\partial}{\partial y}[t_{xy} + \tau_{xy}] &= 0 \\ \frac{\partial(\rho uv)}{\partial x} + \frac{\partial(\rho v^2)}{\partial y} + \frac{\partial p}{\partial y} - \frac{\partial}{\partial x}[t_{xy} + \tau_{xy}] - \frac{\partial}{\partial y}[t_{yy} + \tau_{yy}] &= 0 \\ \frac{\partial(\rho u h_t)}{\partial x} + \frac{\partial(\rho v h_t)}{\partial y} + \frac{\partial}{\partial x}[q_{L_x} + q_{T_x}] + \frac{\partial}{\partial y}[q_{L_y} + q_{T_y}] - \frac{\partial}{\partial x}[u(t_{xx} + \tau_{xx}) + v(t_{xy} + \tau_{xy})] - \frac{\partial}{\partial y}[u(t_{xy} + \tau_{xy}) + v(t_{yy} + \tau_{yy})] &= 0 \end{aligned} \quad (1)$$

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

$$t_{xx} = 2\mu \frac{\partial u}{\partial x} - \frac{2}{3}\mu \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right), \quad t_{yy} = 2\mu \frac{\partial v}{\partial y} - \frac{2}{3}\mu \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right), \quad t_{xy} = \mu \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right)$$

and τ_{ij} is the turbulent stress tensor which is modeled for the mean flow equations as:

$$\tau_{xx} = 2\mu_T \frac{\partial u}{\partial x} - \frac{2}{3}\mu_T \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right), \quad \tau_{yy} = 2\mu_T \frac{\partial v}{\partial y} - \frac{2}{3}\mu_T \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right), \quad \tau_{xy} = \mu_T \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right)$$

The turbulent heat flux terms are

$$q_{T_x} = -\frac{\mu_T}{Pr_T} c_p \frac{\partial T}{\partial x} \quad \text{and} \quad q_{T_y} = -\frac{\mu_T}{Pr_T} c_p \frac{\partial T}{\partial y},$$

the laminar heat flux terms are

$$q_{L_x} = -\frac{\mu}{Pr} c_p \frac{\partial T}{\partial x} \quad \text{and} \quad q_{L_y} = -\frac{\mu}{Pr} c_p \frac{\partial T}{\partial y},$$

and the total energy and enthalpy are

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

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 found via

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

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

B. Turbulence Equations

The turbulence model examined in this work is the baseline (BSL) version of Menter's two-equation $k-\omega$ model.¹⁸ One of the chief difficulties in verifying Menter's model is in the handling of the blending function F and the various \min and \max functions that arise in the model. The general form for the transport of the turbulent kinetic energy (k) and the turbulence frequency (ω) for compressible flow are:

$$\frac{\partial(\rho uk)}{\partial x} + \frac{\partial(\rho vk)}{\partial y} - P + \beta^* \rho \omega k - \frac{\partial}{\partial x} \left[(\mu + \sigma_k \mu_T) \frac{\partial k}{\partial x} \right] - \frac{\partial}{\partial y} \left[(\mu + \sigma_k \mu_T) \frac{\partial k}{\partial y} \right] = 0 \quad (2)$$

$$\begin{aligned} \frac{\partial(\rho u \omega)}{\partial x} + \frac{\partial(\rho v \omega)}{\partial y} - \frac{\gamma \rho}{\mu_T} P + \beta^* \rho \omega^2 - \frac{\partial}{\partial x} \left[(\mu + \sigma_\omega \mu_T) \frac{\partial \omega}{\partial x} \right] - \frac{\partial}{\partial y} \left[(\mu + \sigma_\omega \mu_T) \frac{\partial \omega}{\partial y} \right] \\ - 2(1-F) \rho \sigma_{\omega 2} \frac{1}{\omega} \left[\frac{\partial k}{\partial x} \frac{\partial \omega}{\partial x} + \frac{\partial k}{\partial y} \frac{\partial \omega}{\partial y} \right] = 0 \end{aligned} \quad (3)$$

The turbulence production term is given by

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

where the full compressible turbulent stress tensor is used:

$$\tau'_{xx} = 2\mu_T \frac{\partial u}{\partial x} - \frac{2}{3} \left[\mu_T \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) + \rho k \right], \quad \tau'_{yy} = 2\mu_T \frac{\partial v}{\partial y} - \frac{2}{3} \left[\mu_T \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) + \rho k \right], \quad \tau'_{xy} = \mu_T \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right)$$

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 $k-\omega$ and $k-\epsilon$ models as

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

and the $k-\omega$ coefficients (ϕ_1) are

$$\sigma_{k1} = 0.5, \quad \sigma_{\omega 1} = 0.5, \quad \beta_1 = 0.0750$$

$$\beta^* = 0.09, \quad \kappa = 0.41, \quad \gamma_1 = \beta_1 / \beta^* - \sigma_{\omega 1} \kappa^2 / \sqrt{\beta^*}$$

and the $k-\epsilon$ coefficients (ϕ_2) are:

$$\sigma_{k2} = 1.0, \quad \sigma_{\omega 2} = 0.856, \quad \beta_2 = 0.0828$$

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

In the BSL model, the blending function is given by

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

where

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

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

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

III. Manufactured Solutions

A. Methodology

The most rigorous code verification test is the order of accuracy test,^{2,6} which determines whether or not the discretization error is reduced at the expected rate. This test is equivalent to determining whether the observed order of accuracy matches the formal order of accuracy. For finite-difference and finite-volume methods, the formal order of accuracy is obtained from a truncation error analysis of the discretization algorithm, while for finite-element methods it is found from interpolation theory. Most production-level CFD codes are formally second-order accurate; although in some cases the formal order is reduced to first order in order to ensure solution robustness. In cases where laminar instabilities, turbulence, or acoustics are simulated, higher-order methods are often needed.

The observed order of accuracy is the accuracy that is directly computed from code output for a given simulation or 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.³ 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.

We will now consider a method for calculating the observed order of accuracy assuming that the exact solution is known, which is the currently accepted approach for code verification. The discretization error is formally defined as the difference between the exact solution to the discrete equations and the exact (continuum) solution to the governing partial differential equations f_{exact} . 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 (note that this implies that the round-off and iterative error can be neglected). Consider a series expansion of the discretization error in terms of h_k , a measure of the element size on mesh level k

$$DE_k = f_k - f_{exact} = g_p h_k^p + HOT \quad (4)$$

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:

$$p = \ln \left(\frac{DE_2}{DE_1} \right) / \ln(r) \quad (5)$$

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 , and L_∞ norms for 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 $[\rho, \rho u, \rho v, \rho e, \rho k, \rho \omega]$.

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 sources of error do not adversely impact the order of accuracy calculation,³ 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 (a measure of the iterative error) are reduced down to machine zero. For the flow equations, this corresponds to a residual reduction of approximately 14 orders of magnitude, while for the turbulence equations, a reduction of only 9 orders of magnitude was observed.

B. Baseline Manufactured Solutions

In all of the 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

$$\phi(x, y) = \phi_0 + \phi_x f_s\left(\frac{a_{\phi x} \pi x}{L}\right) + \phi_y f_s\left(\frac{a_{\phi y} \pi y}{L}\right) + \phi_{xy} f_s\left(\frac{a_{\phi xy} \pi xy}{L^2}\right) \quad (6)$$

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. The baseline Manufactured Solutions for the primitive variables are shown below in Figure 1. The specific values for the constants in the above equation are given in the Appendix.

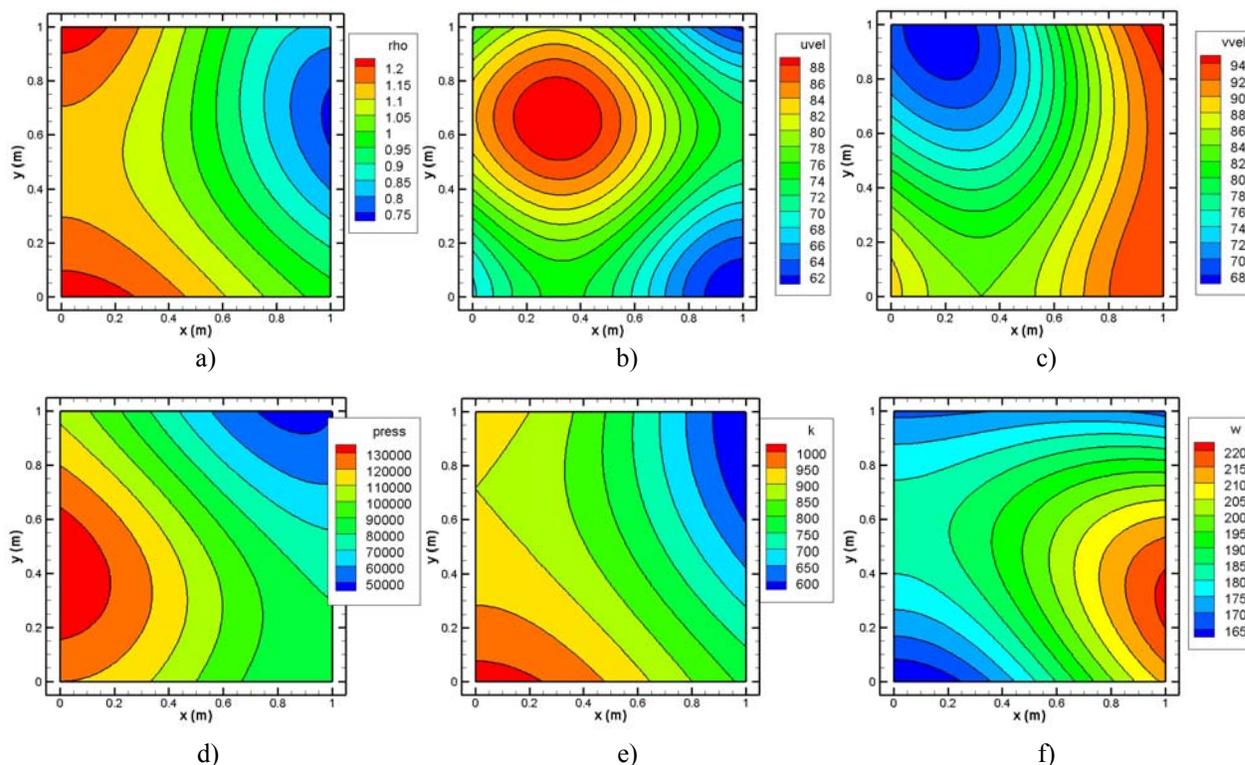


Figure 1 Manufactured Solution for the primitive variables: a) density, b) x-component of velocity, c) y-component of velocity, d) pressure, e) turbulent kinetic energy, and f) turbulence frequency.

C. Source Terms

The manufactured source terms for the basic flow equations will be presented in a future paper. The source terms for the BSL version of Menter's $k-\omega$ turbulence model with and infinite wall distance giving $F_1 = 0$ (i.e., the $k-\varepsilon$ model) are presented below in Figure 2a-b. These source terms exhibit smooth variations in both the x and y directions. Source terms were also generated for the case where the wall distance was set to a constant value of 1×10^{-6} m over the entire domain. This case corresponds to setting $F_1 = 1$ (i.e., the $k-\omega$ model) and the source terms are presented in Figure 2c-d. While the source term for the k -equation shows little change, the source term for the ω -equation shows some differences due to the wall distance being set to a small number.

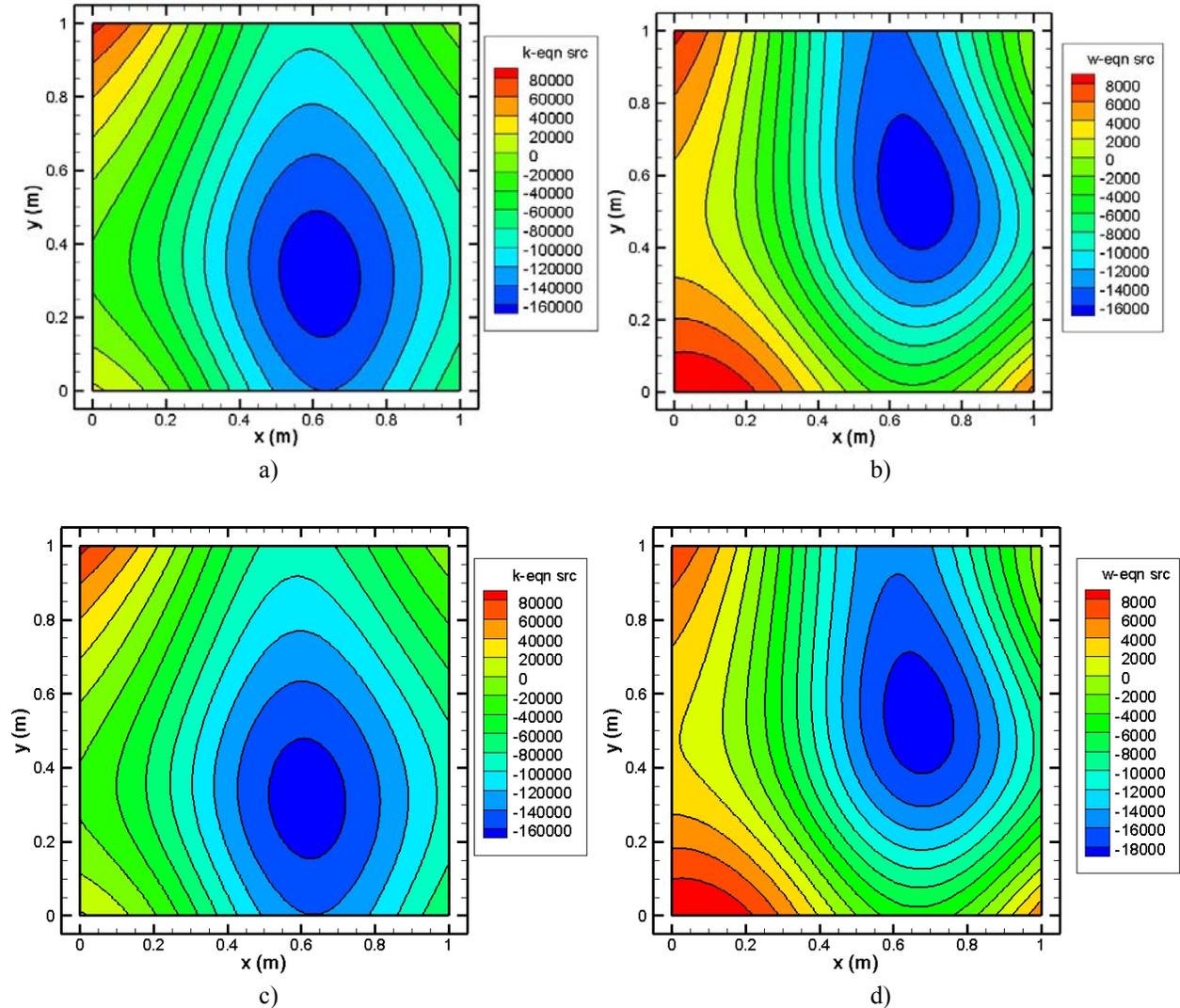


Figure 2 Manufactured Solution source terms for Menter's BSL turbulence model: a) k -equation source term assuming infinite wall distance (i.e., $k-\varepsilon$ model), b) ω -equation source term assuming infinite wall distance (i.e., $k-\varepsilon$ model), c) k -equation source term assuming $d = 1 \times 10^{-6}$ m (i.e., $k-\omega$ model), b) ω -equation source term assuming $d = 1 \times 10^{-6}$ m (i.e., $k-\omega$ model).

D. Ratios of Source Terms

Rather than generate complex, physically-realistic Manufactured Solutions, we have chosen here to employ non-physical solutions. One of our goals in this process was to generate Manufactured Solutions for the turbulence equations such that all of the terms in the turbulence models were roughly the same order of magnitude over some significant region of the domain. We confirm that we have achieved terms which are the same order of magnitude by plotting contours of the ratios of different source terms versus the destruction terms (e.g., $\rho k/\omega$ for the k-equation). The actual destruction term in the k-equation is plotted in Figure 3a along with the ratios of the other terms (convections, diffusion, and production) to the destruction term in Figure 3b-d. It is clear that for at least some region of the domain these source terms are roughly the same order of magnitude. Similar source term ratios for the ω -equation are presented in Figure 4a-e.

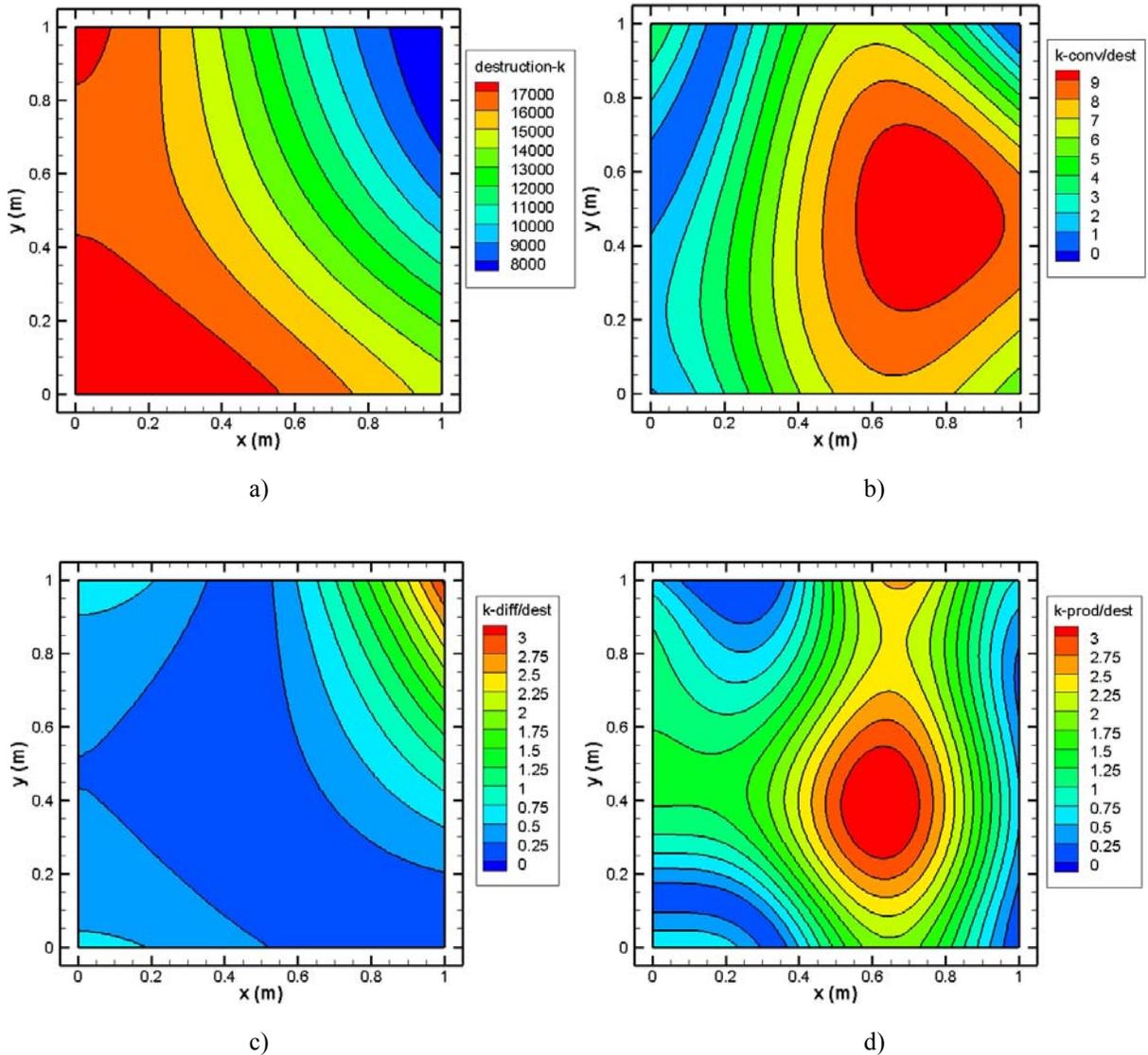


Figure 3 Source term magnitudes for the turbulent kinetic energy equation: a) actual destruction term magnitude, b) ratio of convection terms to the destruction term, c) ratio of diffusion terms to the destruction term, and d) ratio of the production term to the destruction term.

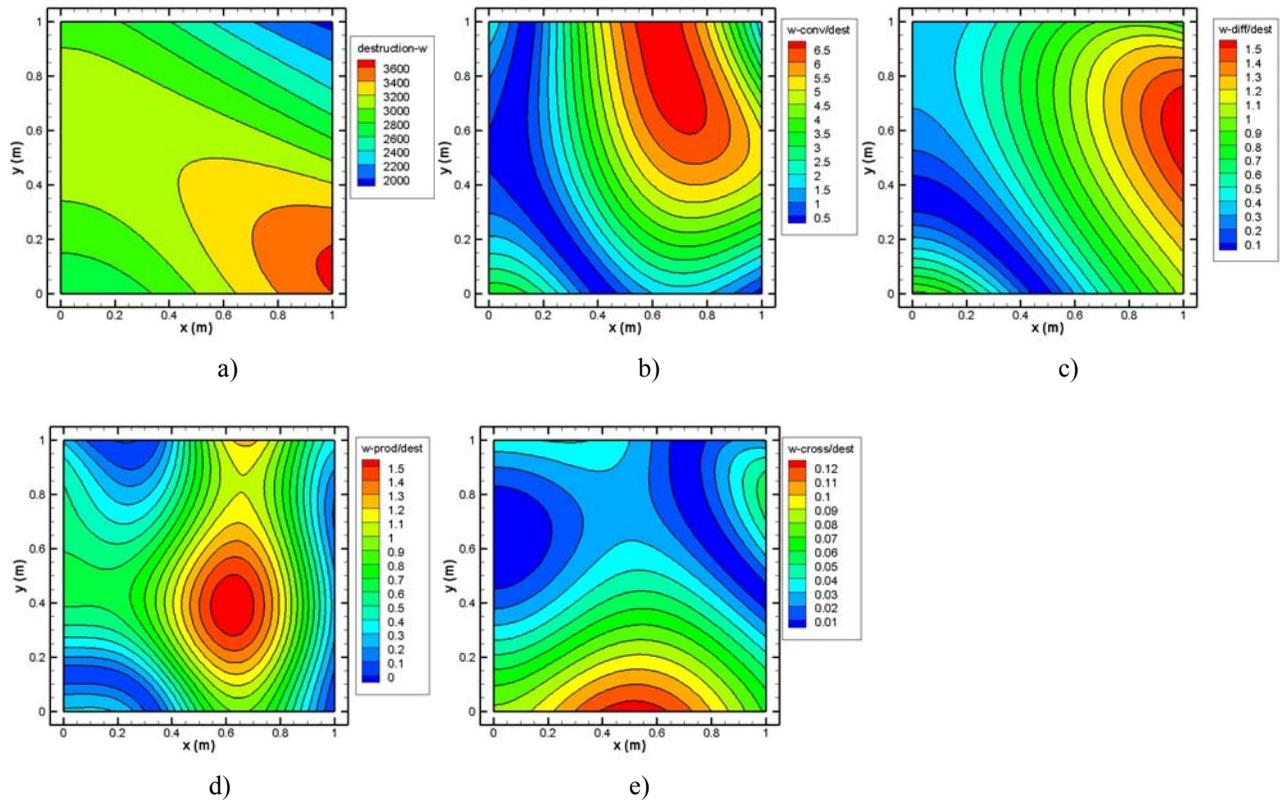


Figure 4 Source term magnitudes for the turbulence frequency equation: a) actual destruction term magnitude, b) ratio of convection terms to the destruction term, c) ratio of diffusion terms to the destruction term, d) ratio of the production term to the destruction term, and e) ratio of cross-diffusion term to the destruction term.

IV. Grids

In order to verify all mesh transformations (either local or global) are coded correctly, the Manufactured Solutions should be run on the most general grid types that will be used by analysts. Grids with excessive skewness, high aspect ratio cells, or large stretching factors should not be used as these grids are likely to lead to a reduction in the observed order of accuracy. However, mild skewness, aspect ratio, and stretching should be examined to ensure the underlying grid transformations are correct. In this section, we present some typical structured and unstructured grid topologies that have been tested. While grids shown tend to be relatively coarse grids, the structured grids employed in this work ranged in size from 8×8 cells to 512×512 cells, with the coarser grids being found by removing every other grid line to ensure uniform coarsening over the domain. The unstructured grids were generated by starting with a given structured grid level and adding diagonals to each structured cell through various strategies to make the grid unstructured. The importance of having families of grids to test which have been uniformly refined over the entire domain with similar topologies can not be overstated.

A. Structured Grid Topologies

A number of different structured grid families were examined ranging from true Cartesian to curvilinear with skewness, nonunity aspect ratio, and grid stretching. Examples of these grid topologies are presented in Figure 5. The stretched Cartesian mesh can be used to isolate out the effects of grid stretching and aspect ratio, the curvilinear (or doughnut) grid can be used to test the effects of curved boundaries without the presence of skewness or stretching, and the skewed curvilinear grid tests all effects on a single grid type.

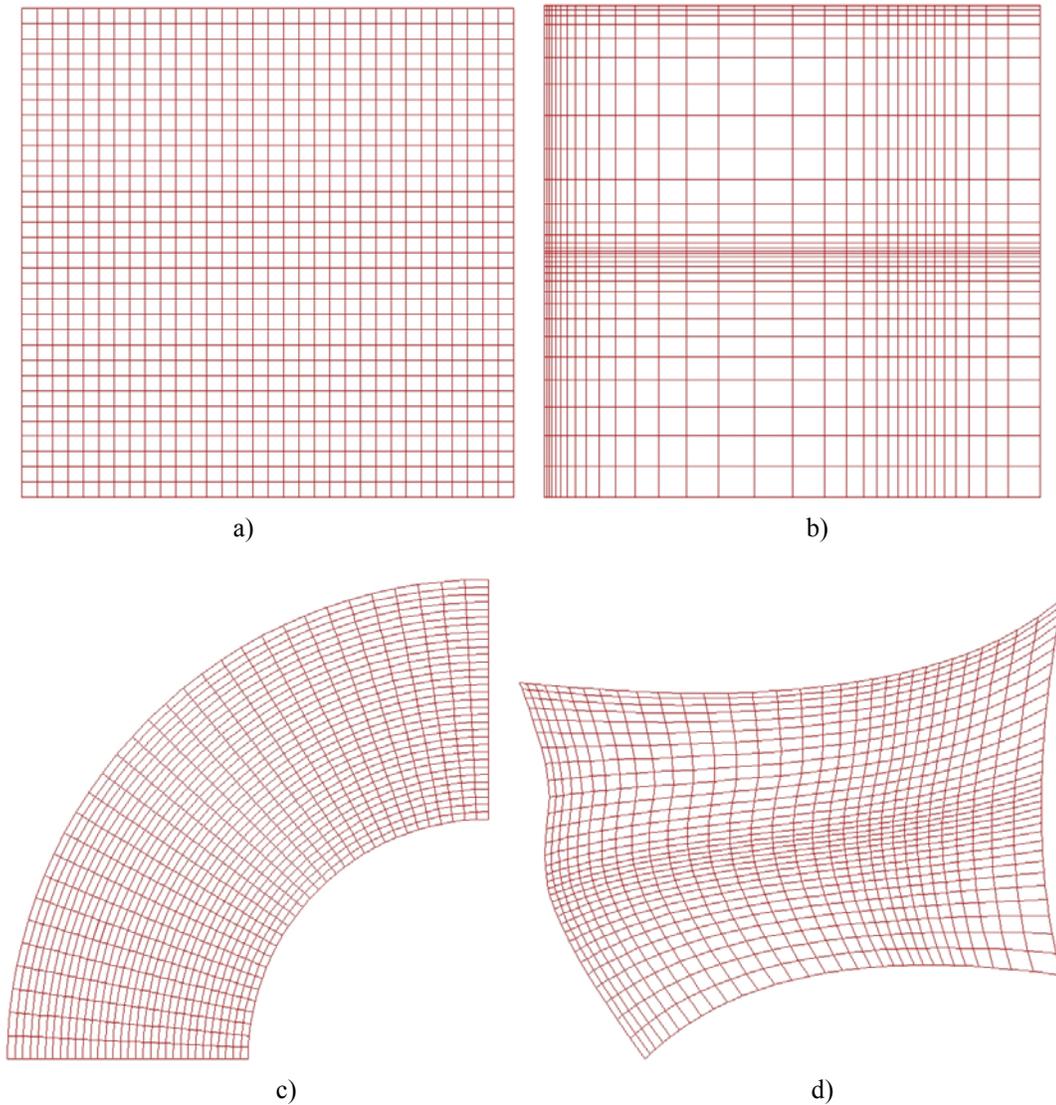


Figure 5 Examples of structured grid topologies examined: a) Cartesian, b) stretched Cartesian, c) curvilinear, and d) skewed curvilinear.

B. Unstructured Grid Topologies

As discussed earlier, the unstructured grids are formed by taking a grid level from the structured grid topologies and then adding in diagonals to convert quadrilateral cells to triangular cells. Examples of some different strategies for adding in these diagonals are given in Figure 6. While all of the unstructured grids shown are based on the Cartesian structured grid, they could just as easily be based on one of the other structured grid topologies.

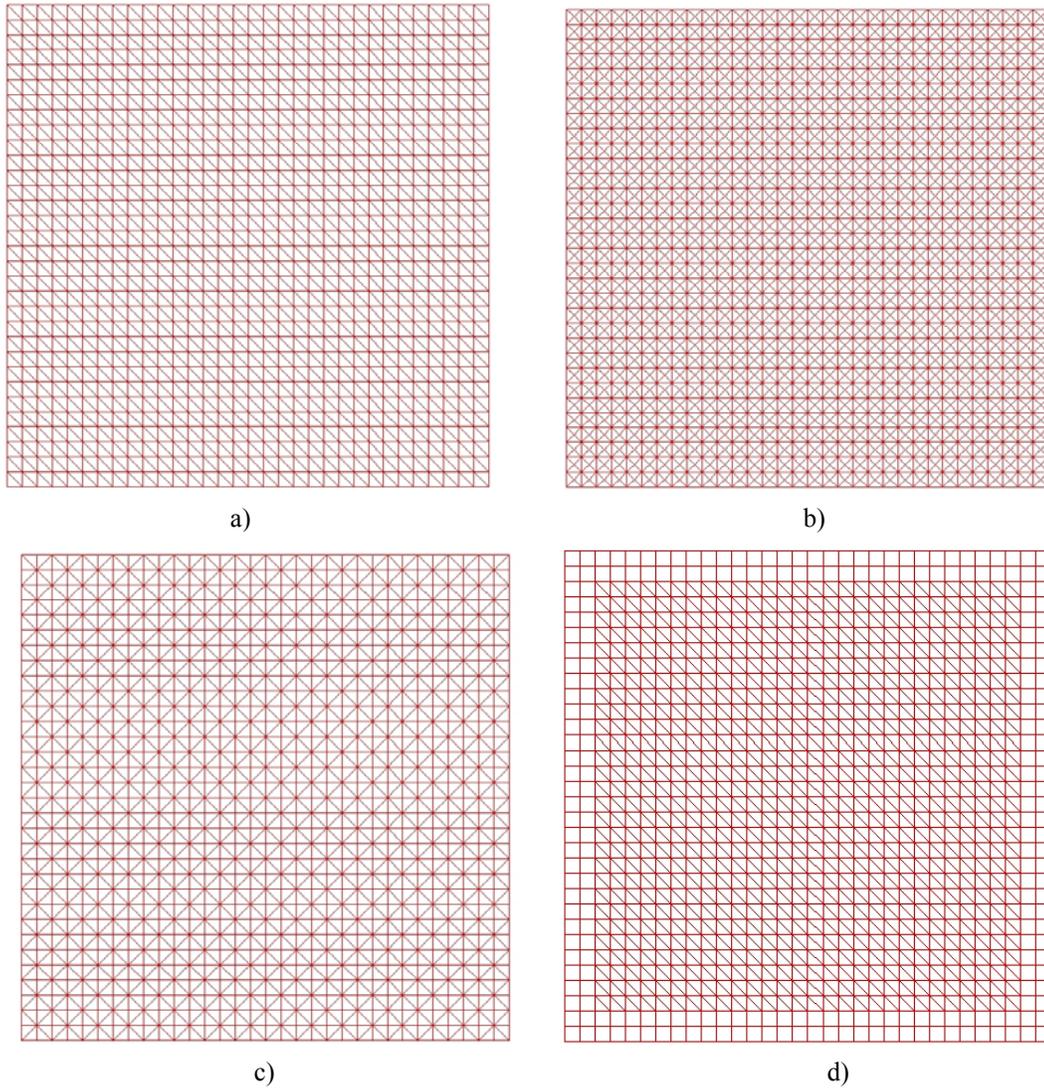


Figure 6 Examples of Cartesian grid-based unstructured grid topologies examined: a) unidirectional diagonal, b) bidiagonal, c) alternating diagonal, and d) hybrid structured-unstructured with unidirectional diagonal.

V. Code Verification Results

A. Laminar Flows

Before getting into the verification of the RANS model, we first performed some initial testing on laminar flows using the skewed curvilinear structured mesh from Figure 5d. For these cases we solved the 2D steady laminar Navier-Stokes equation as given earlier, but with $\mu_T = 0$. The Manufactured Solution was the same form as that given in Figure 1, but with the constants slightly modified. The order of accuracy results for this case as a function of the element size parameter h (see Figure 7a) show that the numerical solutions become inconsistent as the mesh is refined, i.e., the discretization error does not decrease with mesh refinement. Note that $h = 1$ (left side of the plot) represents a fine grid of 256×256 cells used in the order of accuracy calculation. Earlier testing of the Euler equations (i.e., turning off the viscous terms) showed that code was producing an order of accuracy of two for this case, thus suggesting that the problems were in the diffusion terms.

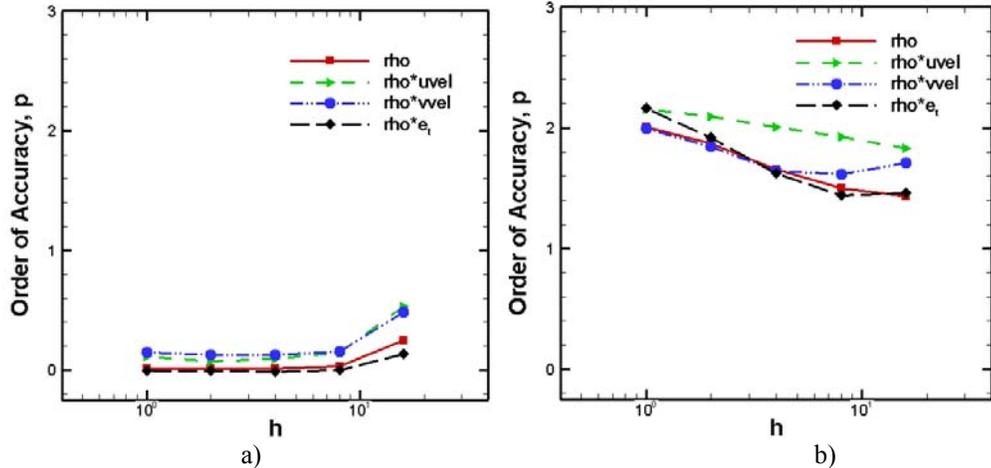


Figure 7 Order of accuracy for the laminar Navier-Stokes equations on the skewed curvilinear mesh: a) original diffusion operator and b) new diffusion operator.

In the diffusion operator, gradients are required at the cell faces to compute the viscous fluxes. Since unstructured finite-volume CFD codes typically compute and store gradients at the cell centers, a mechanism for obtaining gradients at the faces is required. The Loci-CHEM code calculates the gradient at the face by computing both normal and tangential components of the gradient. The original formulation for calculating the normal component of the face gradient utilized the strategy suggested by Strang et al.²² for the Cobalt 60 code. This approach effectively neglects a term in the normal gradient which can lead to stencils with negative weights (which affect the code's stability). Luke²³ has modified the normal gradient calculation such that a limiter is applied to the offending term that both maintains second-order accuracy in smooth regions of the flow and ensures a positive stencil. See Reference 23 for more details. Code verification results for the laminar Navier-Stokes equations using this new diffusion operator are presented in Figure 7b showing that the order of accuracy indeed asymptotes to two.

B. Menter BSL Model on the Cartesian Grid

Results for the Menter BSL model are first presented with the wall distance d set to infinity. For this case, the blending function $F_1 = 0$ and the model defaults to the transformed $k-\epsilon$ model. The order of accuracy as a function of element size parameter h is presented in Figure 8a and shows that the formal order of accuracy of two is indeed matched. Results for the Menter BSL model with $d = 1 \times 10^{-6}$ m are presented in Figure 8b. This choice for the wall distance value results in the blending function $F_1 = 1$ over the entire domain, thus giving the original $k-\omega$ model.

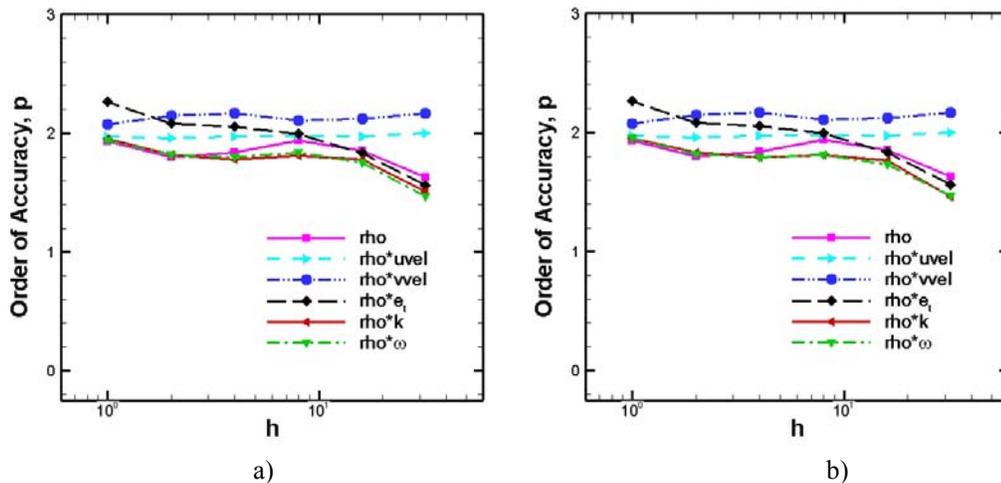


Figure 8 Order of accuracy results for the Menter BSL turbulence model on the Cartesian grid with: a) the transformed $k-\epsilon$ model ($F_1 = 0$) and b) the $k-\omega$ model ($F_1 = 1$).

C. Menter BSL Model on the Skewed Curvilinear Grid

Code verification results for the Menter BSL model are now presented on the skewed curvilinear grid from Figure 5d. The results for the case where the wall distance is set to infinity ($F_1 = 0$, or the transformed k- ϵ model) are given in Figure 9a. The order of accuracy asymptotically approaches the formal order of two as the meshes are refined towards the 512×512 cell mesh. Figure 9b shows the results for the case where the wall distance is set to $d = 1 \times 10^{-6}$ m ($F_1 = 1$, or the k- ω model). Again, the order of accuracy approaches two as the mesh is refined.

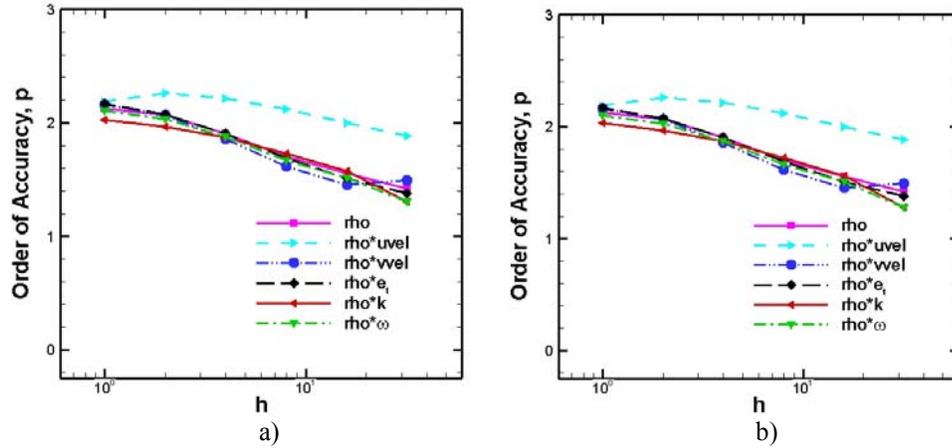


Figure 9 Order of accuracy results for the Menter BSL turbulence model on the skewed curvilinear grid with: a) the transformed k- ϵ model ($F_1 = 0$) and b) the k- ω model ($F_1 = 1$).

D. Unstructured Cartesian-Based Bidiagonal Grid

We have also performed some preliminary investigations into the performance of the Loci-CHEM code on the Cartesian-based unstructured bidiagonal mesh given in Figure 6b. Inviscid results for the Euler equations (with no diffusion terms or turbulence transport equations) show that the convective terms are indeed second-order accurate, as presented in Figure 10a. However, code verification studies of the laminar Navier-Stokes equations (Figure 10b) and the Menter BSL model with the wall distance set to infinity (Figure 10c) both show that the diffusion operator is only first order accurate on this Cartesian-based unstructured bidiagonal grid.

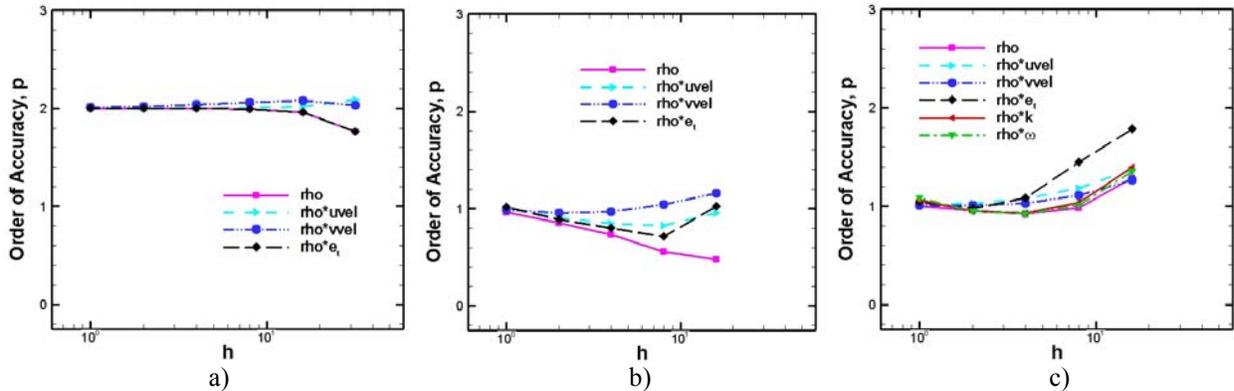


Figure 10 Order of accuracy on the Cartesian-based unstructured bidiagonal grid: a) Euler equations, b) laminar Navier-Stokes equations, and c) Menter BSL turbulence model with the transformed k- ϵ model ($F_1 = 0$).

Insight into the reason for the reduced order of accuracy can sometimes be found by examining the local discretization errors. The local error in the primitive variables for the laminar Navier-Stokes case are presented in Figure 11. The pressure and density have the largest percentage discretization error near the top boundary, while the velocity components show larger error in the domain. The reasons for the reduced order of accuracy on these unstructured meshes is still under investigation.

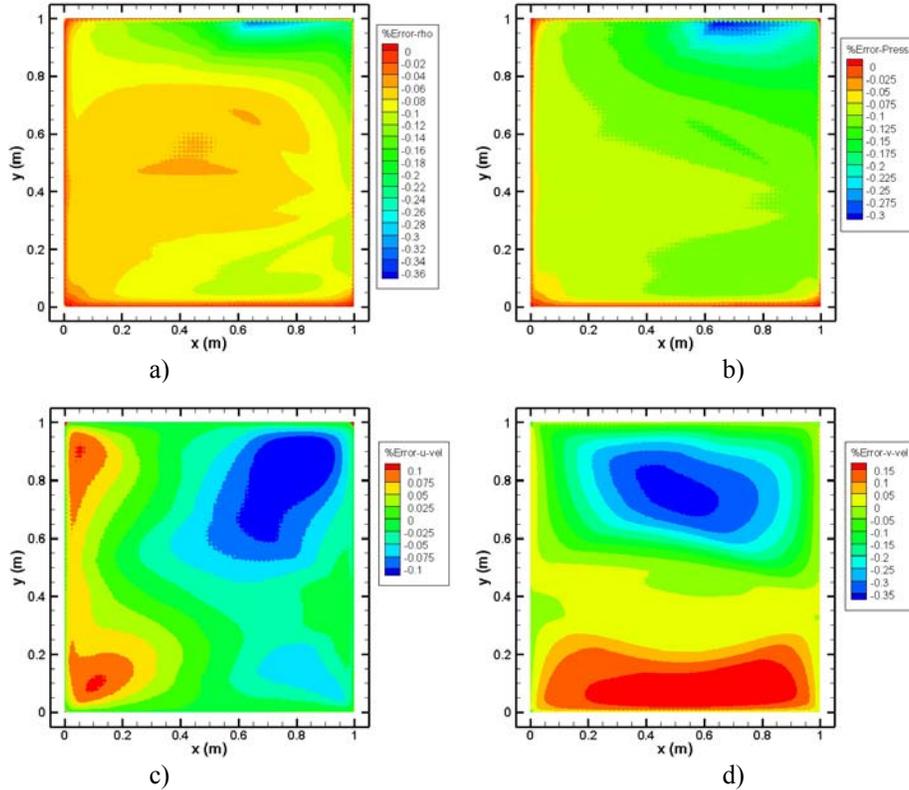


Figure 11 Local percentage errors for laminar Navier-Stokes case on the unstructured bidiagonal grid: a) density, b) pressure, c) u-velocity, and d) v-velocity

VI. Conclusions

An approach for performing code verification studies on RANS turbulence models was presented which relies on smooth, non-physical Manufactured Solutions which exercise all terms in the turbulence transport equations. This approach has been applied to the Loci-CHEM CFD code for the baseline Menter $k-\omega$ turbulence model. The turbulence transport equations, along with the Reynolds-averaged Navier-Stokes equations, have been verified in the Loci-CHEM CFD code by computing the observed order of accuracy for the Manufactured Solution on a series of consistently-refined grids. For the structured grid topologies examined (Cartesian and skewed curvilinear) the observed order of accuracy matched the formal order of two. For the unstructured bidiagonal grid examined, the observed order of accuracy was found to reduce to first order. Manufactured Solutions for the Euler equations and the laminar Navier-Stokes equations suggest that the source of this reduced order of accuracy is the diffusion operator on unstructured grids.

VII. Acknowledgments

This work was supported by the National Aeronautics 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. *Guide for the Verification and Validation of Computational Fluid Dynamics Simulations*, American Institute of Aeronautics and Astronautics, AIAA G-077-1998, Reston, VA, 1998.
2. Roache, P. J., *Verification and Validation in Computational Science and Engineering*, Hermosa Publishers, New Mexico, 1998.
3. 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.

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- ω 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. Strang, W., Tomaro, R., and Grismer, M., "The Defining Methods of Cobalt60: A Parallel, Implicit, Unstructured Euler / Navier-Stokes Flow Solver," AIAA Paper 1999-0786, 1999.
23. Luke, E., "On Robust and Accurate Arbitrary Polytope CFD Solvers," AIAA Paper 2007-3956, 2007.

Appendix: Coefficients in the Manufactured Solutions

The constants used in the Menter BSL manufactured solution (Equation 6) are give in Table A1.

Table A1. Constants for Menter BSL turbulence model Manufactured Solution

Equation, ϕ	ϕ_0	ϕ_x	ϕ_y	ϕ_{xy}	a_{ϕ_x}	a_{ϕ_y}	$a_{\phi_{xy}}$
ρ (kg/m ³)	1	0.15	-0.1	0.08	0.75	1	1.25
u (m/s)	70	7	-8	5.5	1.5	1.5	0.6
v (m/s)	90	-5	10	-11	1.5	1	0.9
p (N/m ²)	1×10^5	0.2×10^5	0.175×10^5	-0.25×10^5	1	1.25	0.75
k (m ² /s ²)	780	160	-120	80	0.65	0.7	0.8
ω (1/s)	150	-30	22.5	40	0.75	0.875	0.6