presented at the 48th AIAA Aerospace Sciences
Meeting and Exhibit, Orlando, Florida, January 2010
AIAA 2010-0125

The AIAA Code Verification Project - Test cases for CFD Code Verification

Urmila Ghia
University of Cincinnati, Cincinnati, OH 45220 -0072

Sami Bayyuk and Sami Habchi
CFD Research Corporation, Huntsville, AL 35805

Chris Roy
Virginia Tech, Blacksburg, VA 24061-0203

Tom Shih
Iowa State University, Ames, IA 50011-2271

Terrence Conlisk
Ohio State University, Columbus, OH 43210

Charles Hirsch
Vrije Universiteit, Brussels, Belgium

Joseph M. Powers
University of Notre Dame, Indiana 46556-5637

This paper presents a collection of fluid mechanics problems with exact solutions which can be used to verify the numerical accuracy of solutions obtained by CFD codes. This document is a product of the AIAA Committee On Standards (COS). It is intended to serve as the start of a catalog of exact solutions for fluid mechanics problems, and as a complement to the Verification and Validation Guide prepared by the AIAA. While the solutions presented in this paper do not necessarily test all aspects of a code or all terms of the governing equations, they constitute fundamentals tests for identifying coding errors. Hence, documenting these solutions is an important step towards consolidating significant verification cases.

A. Introduction

Following completion of the AIAA guide [1], the AIAA Committee on Standards (CoS) for CFD undertook the verification and validation project as a follow-on task. At about the same time, the ASME PTC61 Committee was pursuing a validation and verification project as well. The latter is focused on validation, and includes a section on verification, with verification there referring to solution verification. It begins with code verification being assumed to have been completed. The AIAA project is focused on code verification. The outcome of this project is a collection of exact solutions that are useful to the code verification process. The present paper describes some of these cases, as listed below:

- Expansion fan
- Oblique shock
- Incompressible Laminar Boundary Layer Flow Solutions in Similarity Variables
- Couette Thermal Flow
- The Method of Manufactured Solutions
- Uniform Stream
- Heat Transfer in a smooth U-Duct with and without rotation
- Detonation wave
- Burger's Equation

All the cases are described using a consistent format. This includes an introduction and motivation, which provide a broad description of the problem, along with the characteristics/features of the governing equations to be tested, and sample applications. This is followed by a single paragraph description of the test case, including a simple sketch. Next, the governing equations and the corresponding boundary conditions are presented. Coordinate transformations are included only where required. Following this, the shape and extent of the computational domain are presented, after which the verification metrics, i.e., specific quantities of interest for comparison are given. The derivation of the exact solution is then presented. Subsequently, a graphical presentation of solution is given (In the AIAA Verification project document, tabulated results will be furnished in an appendix, along with an illustration of the type of quantities to be compared). Typically, this includes a plot of representative results, appropriate values of any parameters, and a description of how the results were obtained. Next, if applicable, a grid refinement study is presented, which is followed, finally, by a summary. References are included at the end of the total document. While all the references used to construct the verification cases presented in this paper have been included at the end, the text of the document cites only those which are reflected in the short description of the verification cases provided in the present paper.

Before proceeding to specific verification cases, it is important to recognize the importance of the method of manufactured solutions (MMS) [10] in the code development and code verification process. MMS serves to provide the exact solution of a modified governing equation operator with an appropriate source term. It is intended to exercise all the terms of the governing equations in the interior, as well as the boundary equations. The modification (source term) is easily determined, even for the most complex form of the assumed solution function, by using symbolic manipulation software. MMS is a purely mathematical process, and the manufactured solutions do not correspond to physically meaningful phenomena. MMS is a highly accurate method for testing numerical codes and algorithms, as manufactured solutions do not suffer from numerical accuracy issues that commonly occur with analytical solutions.

The exact solutions presented in the present paper may not exercise all the terms in the governing equations (here, the Euler or Navier-Stokes equations). These are physical flow cases, while the method of the manufactured solutions may not represent a real flow at all. As might be expected, exact solutions are possible only for simple flow configurations. Since these constitute relatively simple cases, they are therefore more readily tested. Hence, if the code does not reproduce exact values for these simple cases, then one should first review and correct the code to reproduce the exact solutions before proceeding to use the code for more complex flow cases.

Code verification using traditional exact solutions serves to reveal inconsistencies in differencing schemes or discretization, not revealed by the method of the manufactured solutions, such as, for example, the Geometric Conservation Law (GCL), global conservation for finite-difference schemes, etc.

This document is a product of the CFD Verification and Validation Committee of the AIAA, consisting of the authors as well as the following members:

1. John A. Benek
2. Mark K. Bryden
3. Wagdi G. Habashi
4. Hyung B. Lee
5. Ian J. Lyttle
6. Mortaza Mani
7. Marc Poinot
8. Christopher L. Rumsey
9. Ken Tatum
10. Doug Lynch
11. Steve Barson
12. Dominique Pelletier
13. Bill Oberkampf
14. Sam Paulucci
15. Balu Sekar

At this juncture, it is important to recognize noteworthy contributions made by Patrick Roache, William Oberkampf, Ismail Celik and Luis Eca, to the field of CFD Verification and Validation. The use of manufactured solutions and grid convergence studies for code verification was first proposed by Roache and Steinberg [13] in 1984. The term "manufactured solutions" was coined by Oberkampf and Blottner [15]. Oberkampf and Trucano [16] authored a landmark paper which presents recommendations for effective design and use of code verification and validation benchmarks, estimation of experimental measurement uncertainty, validation metrics and role of model calibration in validation. Ismail Celik, Roache and others [31] prepared a comprehensive summary of the processes and procedures to be followed in the application of CFD codes to nuclear reactor safety analysis. Celik also made significant contributions in assessing numerical uncertainty and estimation of error in fluid flow calculations.

This paper is intended to serve as the start of a catalog of exact solutions and as a complement to the

Verification and Validation guide prepared by the AIAA [1]. While these solutions do not necessarily test all aspects of a code or all terms of the governing equations, documenting these solutions is seen as an important step towards consolidating significant verification cases.

B. Description of Test Cases

B.1. Prandtl-Meyer Expansion Fan

B.1.1. Description

A uniform supersonic stream flows along a sudden divergence, turning around the corner through a smooth isentropic expansion fan, to assume its rotated direction. The expansion fan radiates away from the corner, separating two regions of uniform flow. Figure 1.1 shows a schematic of an expansion fan flow, depicting the relative location of the expansion fan and its centering at the corner, the boundary conditions, the two regions of uniform flow, and the wall angle.

The primary variables for the flow are M_1 and δ , and these in turn dictate the value of M_2 (where, subscript 2 indicates the post-expansion state, and subscript 1 indicates the pre-expansion state), the complete self-similar geometry of the expansion fan, and the property variations within the fan. The condition $M_1 \geq 1$ must be satisfied to give rise to the expansion fan, always resulting in $M_2 \geq M_1$.

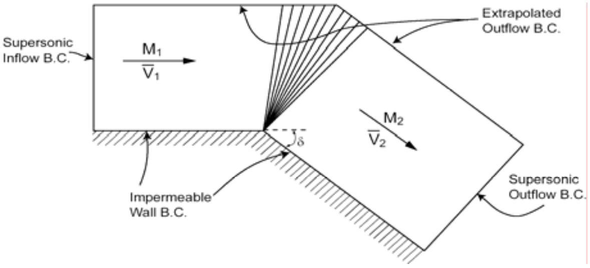

Figure 1.1: Schematic of an Expansion-Fan Flow.

B.1.2. Equations Solved, Boundary Conditions and Constitutive Properties

This verification test case is for the (two-dimensional or three-dimensional) Euler Equations. The boundary conditions are as indicated in Figure 1.1. The specific locations of the boundaries in the computational domain is not critical, provided the boundaries are placed sufficiently far from the corner to allow the creation of sufficiently large, distinct regions of uniform flow, and to allow the fan to develop a sufficient extent relative to the local size of the computational cells. This case can also be used in the vanishing-viscosity limit for the Navier-Stokes Equations, and in that case, the “impermeable wall” boundary conditions shown in Figure 1.1 should be replaced by “symmetry” boundary conditions. This problem does not require coordinate transformations, so these are disregarded here.

The gas is assumed to satisfy the ideal gas relation, pV = nRT, where p is the pressure, V is the volume, n is number of moles, R is the Universal Gas Constant, and T is the absolute temperature. The gas is assumed to have a constant ratio of specific heats. With this Equation of State, the analytical solution is independent of any other properties, except for the ratio of specific heats, γ, which appears as a parameter in the analytical solution, as given in Section B.1.3.

B.1.3. Comparison Metrics

A high level of agreement is expected between all aspects of the computational and the analytical solutions. This includes agreement for the geometry of the expansion fan, for the variation of properties across the fan, and for the post-expansion properties, including the final Mach Number, M_2. In particular, the velocity $\vec{V}_2$ should be parallel to the inclined wall, and the following relations are expected to be satisfied to within no more than 5-10% error, even on a relatively coarse grid:

$$\delta = -\sqrt{b}\left[\arctan \sqrt{\frac{M_2^2-1}{b} - \sqrt{\frac{M_1^2-1}{b}} -}\right] + \left[\arctan \sqrt{M_2^2-1} - \arctan \sqrt{M_1^2-1}\right] \tag{1.2}$$

where,

$$b = \left(\frac{\gamma+1}{\gamma-1}\right).$$

and $\frac{p_2}{p_1}, \frac{\rho_2}{\rho_1}, \frac{T_2}{T_1}$, are all given by isentropic relations.

Depending on the computational scheme and the grid resolution used, the expansion fan may have non-straight edges and may be smeared, and there may be local distortions in the regions where the fan meets the corner or the outflow boundaries, but the smearing should monotonically decrease with increasing grid refinement, and the agreement between all the computed values and the corresponding analytical ones should also monotonically improve with increasing grid refinement. The quality of the computational solution may be highly sensitive to the local grid refinement in the vicinity of the corner. The final expanded state should be accurately predicted by the computational solution, but the level of agreement will be more sensitive to the grid refinement used than, say, for the oblique shock test case given below. In particular, the level of satisfaction of the isentropic ratios given above will be a good measure of the accuracy and the level of numerical dissipation in the computational solution, primarily because the numerical dissipation increases the entropy in the expanded state. The computed solution should be smooth throughout the expansion fan, even on relatively coarse grids.

B.1.4. Approach to Exact Solution

The approach of the computational solution to the self-similar exact solution in this test case depends more on grid resolution than on locations and distances. As described in Section B.1.3 above, the coarser the grid, the more smeared the expansion fan, especially near the corner. However, the overall property ratios across the fan should still be accurately satisfied for reasonably fine grids. The degree of spatial uniformity of the solution on either side of the fan should also be high away from the corner region and the downstream flow originating in that region, even on coarse grids.

For higher-order computational schemes, any oscillations in the solution should be relatively slight, especially around the edges of the fan, and the amplitude of any such oscillations should not continue to increase with increasing grid resolution. The amplitudes of any oscillations around the edges of the fan should not be more than 5% of the total change in the corresponding total property change across the fan, especially well away from the corner.

B.1.5. Mathematically-Complete Definition of the Test Case

Sections B.1.1 and B.1.2 above contain the complete mathematical definition of the generic Prandtl-Meyer Expansion Fan problem. The use of a specific example is discussed below.

B.1.6. Use and Validity of the Verification Data

There is no need to check the validity of the verification for this case as the verification data is from an exact, analytical solution.

B.1.7. Graph of Exact Solution

Several line plots below show the analytical and the computational results for some of the solution variables. Such line plots clearly show the smoothness of variation of the computational solution in the expansion fan, and the locations of the edges of the fan at different distances from the corner.

B.1.8. Specific Example

The following is a specific example for this verification test case:

For M_1 = 2.0, and δ = -10°, the following should result: M_2 = 2.383. The property ratios across the expansion fan should be as follows: p_2/p_1=0.5471, ρ_2/ρ_1=0.6500 and v_2/v_1=1.0931, in satisfaction of the isentropic relations given above.

B.1.9. Computed Solution for Specific Example

Figures 1.2-1.7 below show the Mach number contours for computational solutions obtained using different grid densities and spatial orders of accuracy for the Mach 2.0 specific example given in Section B.1.8 above. The two uniform-flow regions (on the two sides of the expansion fan) of the exact analytical solution are clearly visible in the computational solutions, and so is the correct shape of the fan and the variation of properties across the fan. The computational solutions are therefore seen to agree with the corresponding analytical solution, even on coarse grids. Figures 1.2-1.7 also show how the spatial resolution of the solution monotonically increases with increasing grid resolution. The figures also show how second-order spatial accuracy improves the quality of the solution, as evident in the straighter contours and edges for the fan and a more accurate prediction of the width of the fan as a function of distance from the corner. These figures show that the corner region produces a distortion in the solution that manifests itself through a "slowing down" of the downstream flow that passes through the corner region. The figures show that the "slowing" effect is more spread and slightly more severe for the first-order computations than it is for the second-order computations. The figures also show that the affected region is confined to a layer lying along the inclined boundary and that increasing the grid refinement decreases the thickness of the affected layer, but does not significantly decrease the extent of the slow-down in the layer. This behavior is in agreement with what is expected in a computational solution and is explained further below.

Figures 1.8-1.13 correspond to Figures 1.2-1.7, and show the corresponding entropy-change contours for the different computational solutions. The entropy change here is computed according to the equation

$$S_2 - S_1 = \left(\frac{\gamma R}{\gamma - 1}\right) \ln\left(\frac{T_2}{T_1}\right) - R \ln\left(\frac{p_2}{p_1}\right), \qquad (1.3)$$

(or any of its other equivalents) where the symbols S, T, and p denote the entropy, the absolute temperature, and the pressure, respectively. The subscript 2 denotes the variable, arbitrary state, and the subscript 1 denotes the reference state, which here is taken to be the inlet state. R and γ denote the gas constant and the ratio of specific heats for the specific gas being modeled, respectively; where, in this example they have the values of 287 J/kg/K and 1.4, respectively. Finally, ln(x) denotes the natural logarithm of x.

The flow in the exact analytical solution for this test case is fully isentropic, so any increase in the entropy in any part of the domain of a computational solution is entirely spurious. Figures 1.8-1.13 show that the computational scheme generates a significant amount of entropy (through numerical dissipation) in the corner region, and show that this numerically-generated entropy increase is carried with the downstream flow passing through the corner region. Away from the corner region, the computational solution is observed to preserve the isentropic behavior of the exact solution in the expansion fan. This isentropic behavior away from the corner region is in strong contrast with the oblique-shock test case presented below, where the shock wave generates a physical jump in entropy on which is superposed an additional increase in entropy due to numerical dissipation in the shock-wave region.

Figures 1.8-1.13 also show that, as the grid resolution increases, the thickness of the layer of flow affected by the numerically-generated entropy decreases. This is because the entropy generation is confined along the inclined boundary of the domain (downstream of the corner) in a layer no more than ten or so cells thick, so as the overall cell size decreases, the geometric extent of the affected layer also decreases. It is also seen in Figures 1.8-1.13 that the maximum entropy generation drops slightly in going from a first-order-accurate scheme to a second-order-accurate scheme (from approximately 25J/kg/K to 21J/kg/K), and that, for each scheme, the maximum level of entropy generation is roughly independent of the grid resolution. These overall features of the numerically-generated entropy are in accordance with expectations, and should be observed with any valid computational scheme.

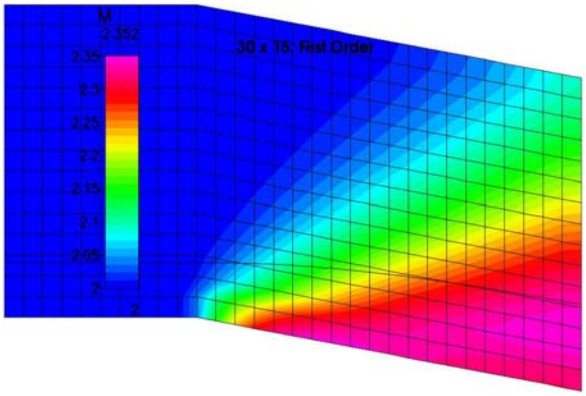

Figure 1.2: Mach Number Contours for a Spatially-First-Order Accurate Computation on a 30x15 Grid.

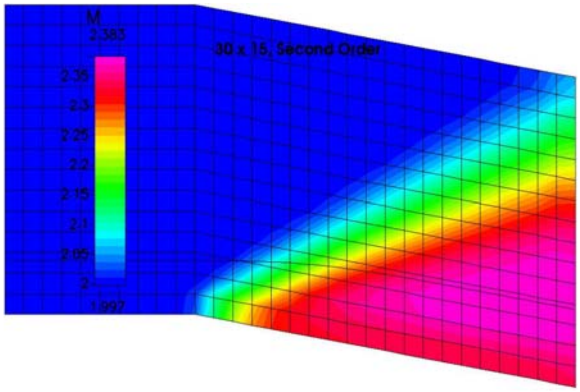

Figure 1.3: Mach Number Contours for a Spatially-Second-Order Accurate Computation on a 30x15 Grid.

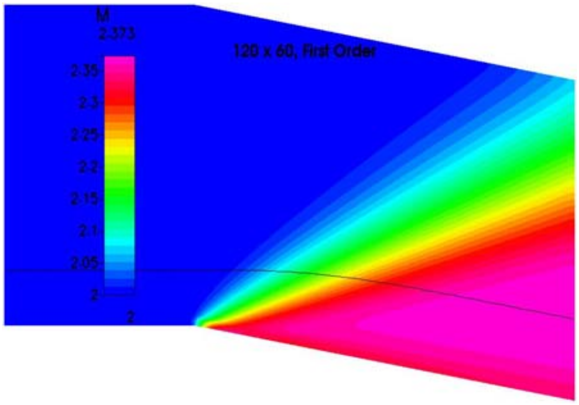

Figure 1.4: Mach Number Contours for a Spatially-First-Order Accurate Computation on an 120x60 Grid.

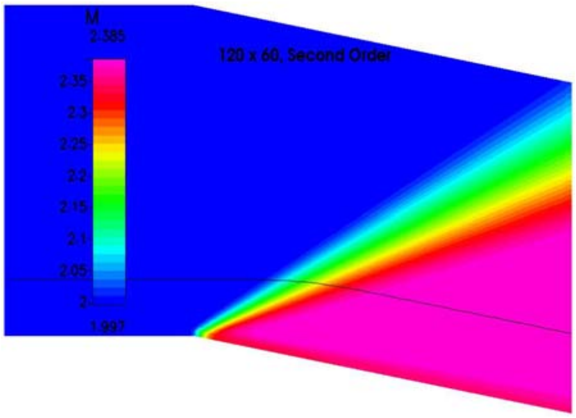

Figure 1.5: Mach Number Contours for a Spatially-Second-Order Accurate Computation on an 120x60 Grid.

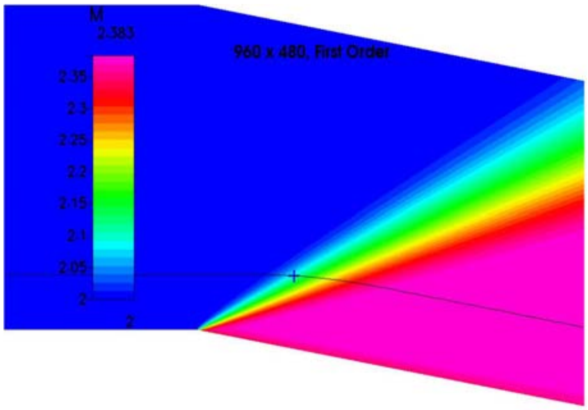

Figure 1.6: Mach Number Contours for a Spatially-First-Order Accurate Computation on a 960x480 Grid.

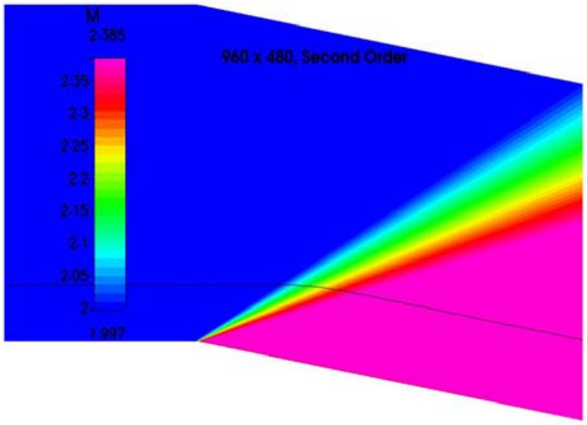

Figure 1.7: Mach Number Contours for a Spatially-Second-Order Accurate Computation on a 960x480 Grid.

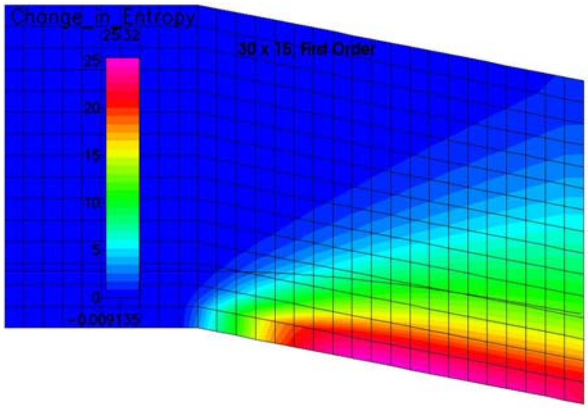

Figure 1.8: Entropy Change Contours for a Spatially-First-Order Accurate Computation on a 30x15 Grid.

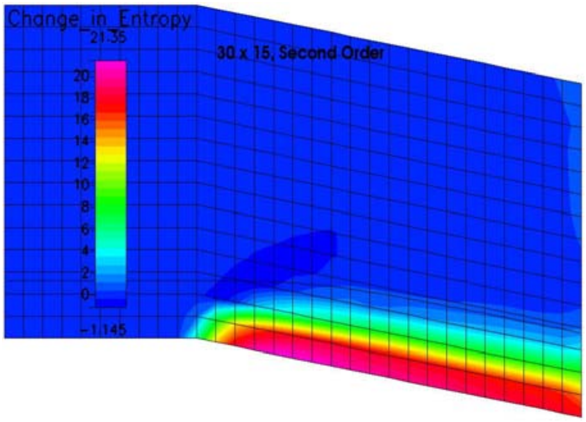

Figure 1.9: Entropy Change Contours for a Spatially-Second-Order Accurate Computation on a 30x15 Grid.

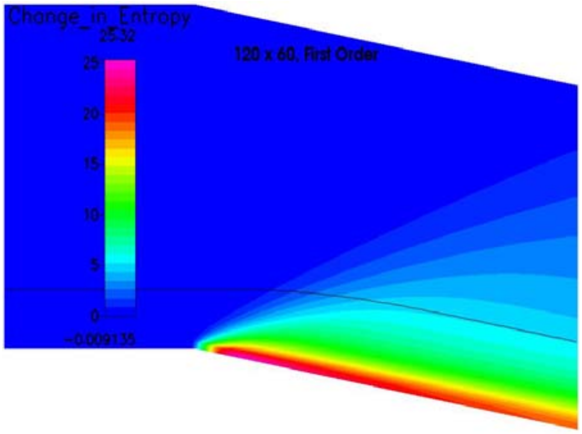

Figure 1.10: Entropy Change Contours for a Spatially-First-Order Accurate Computation on a 120x60 Grid.

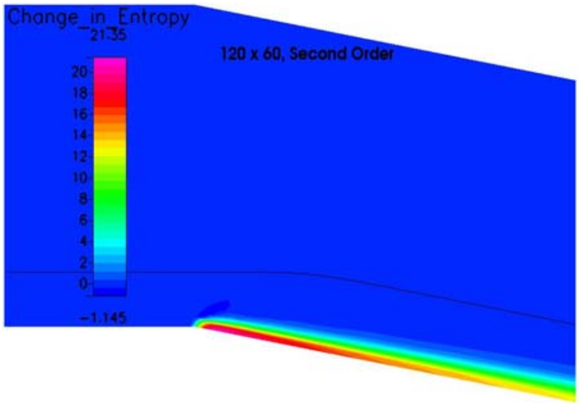

Figure 1.11: Entropy Change Contours for a Spatially-Second-Order Accurate Computation on an 120x60 Grid.

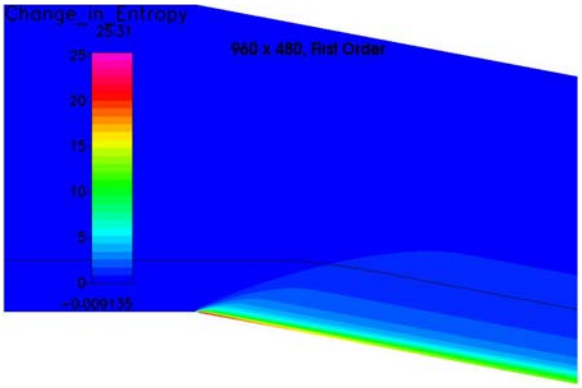

Figure 1.12: Entropy Change Contours for a Spatially-First-Order Accurate Computation on a 960x480 Grid.

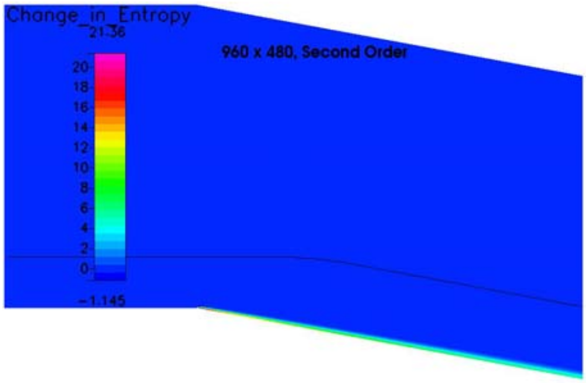

Figure 1.13: Entropy Change Contours for a Spatially-Second-Order Accurate Computation on a 960x480 Grid.

Figures 1.14-1.18 correspond to Figures 1.2-1.7 (but with the solutions on additional grids with intermediate densities also included) and show the pressure distributions along the streamline that is plotted in the latter figures. The line plots in Figures 1.14-1.18 show how the pressure variation across the expansion fan is correctly predicted, even on coarse grids, and how the quality and accuracy of the solution monotonically improves with increasing grid refinement. The figures also show the effect on the spatial resolution of going from first-order accuracy to second-order accuracy for each of the grid densities used, and how the first-order-accurate and the second-order-accurate solutions converge (toward each other) as the grid refinement increases (albeit rather slowly). Figures 1.14-1.18 also show that the computational pressure profile is not precisely centered about the exact-solution profile, and this artifact of the computational solution is probably caused by the numerical diffusion being unequal in the upstream and downstream directions.

The computational solutions from any valid flow solver should exhibit the general behaviors described above in this Section B.1.9.

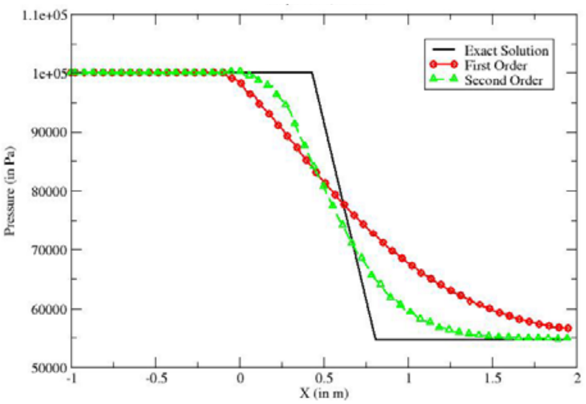

Figure 1.14: Line Plot of the Pressure Distribution along a Streamline on a 30x15 Grid.

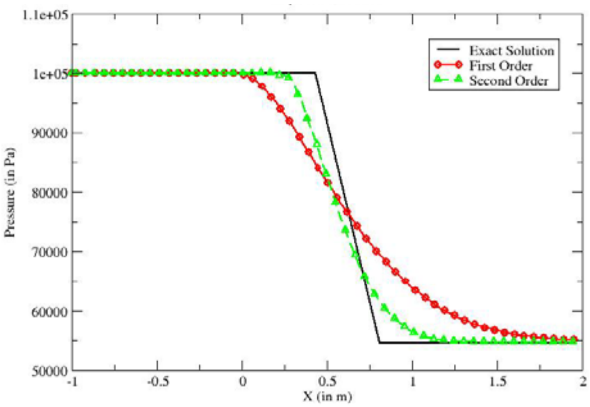

Figure 1.15: Line Plot of the Pressure Distribution along a Streamline on a 60x30 Grid.

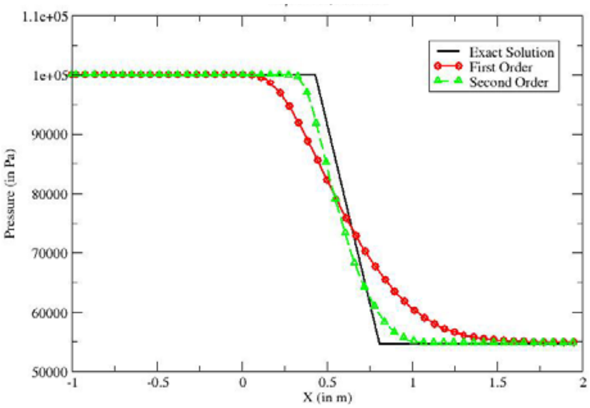

Figure 1.16: Line Plot of the Pressure Distribution along a Streamline on a 120x60 Grid.

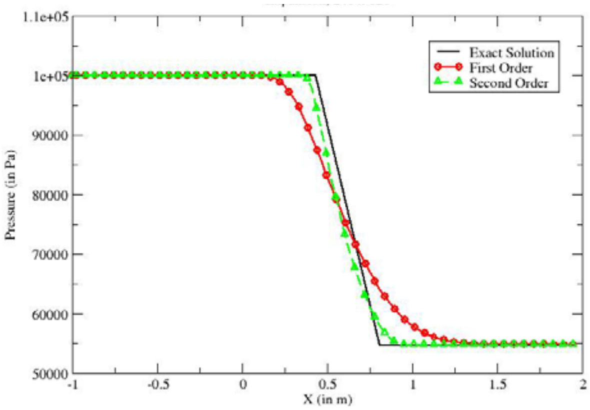

Figure 1.17: Line Plot of the Pressure Distribution along a Streamline on a 240x120 Grid.

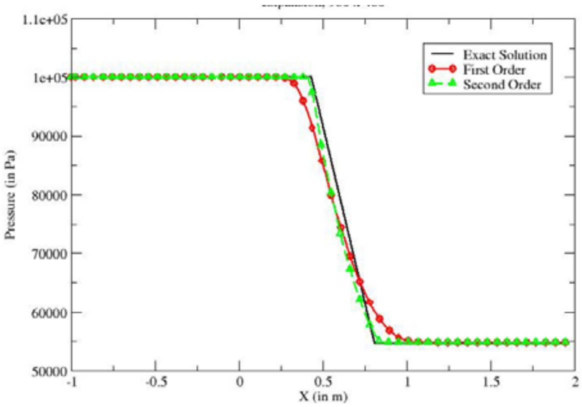

Figure 1.18: Line Plot of the Pressure Distribution along a Streamline on a 960x480 Grid.

Figure 1.19 shows the level of convergence expected in a computational solution for this test case: the residuals should drop by at least 12 orders of magnitude for double-precision computations, and at least six orders of magnitude for single-precision computations, for both the first-order-accurate and the second-order-accurate computations.

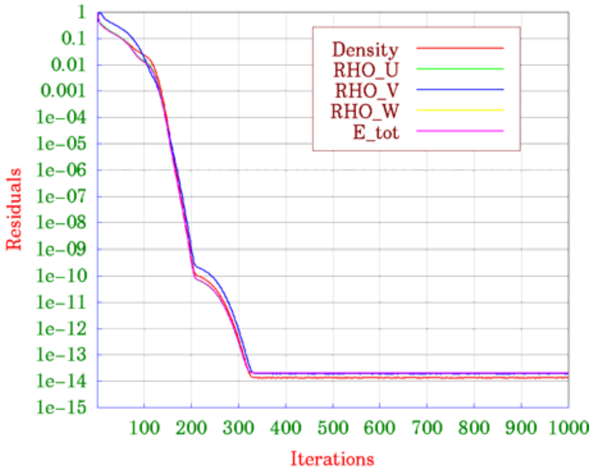

Figure 1.19: Residual Plot Showing Full Convergence for a First-Order-Accurate Double-Precision Computation.

B.1.10. Grid Quality and Grid Refinement Issues

For this validation case, the dependence of the solution quality on grid refinement has been discussed earlier and amply demonstrated in Section B.1.9. The solution quality also depends on grid alignment. In particular, the more closely aligned the grid lines are with the uniform-property radial lines of the analytical solution in the expansion fan, the more accurate the corresponding computational results, and the less the effect of any spurious artifacts in the computed solution, such as incorrect super-imposed local currents resulting from grid-induced artificial mis-alignments of the property variations within the fan. Overall, however, the dependence of the solution quality on grid alignment is lower than it would be for, say, the oblique-shock test case presented below, except perhaps in the corner region.

B.2. Steady-State Oblique Shock Wave

B.2.1. Description

A uniform supersonic stream impinges on a wedge, resulting in a stationary, oblique shock wave, separating two regions of uniform flow. Figure 2.1 shows a schematic of an oblique shock flow, depicting the shock-wave, the boundary conditions, the two regions of uniform flow, the wall angle, and the shock angle.

The primary variables are M_1 and δ, and these in turn dictate M_2 and ε (where the subscript 2 indicates the post-shock state, and the subscript 1 indicates the pre-shock state). The condition $M_1 \sin \varepsilon \geq 1$ must be satisfied to give rise to an oblique shock wave.

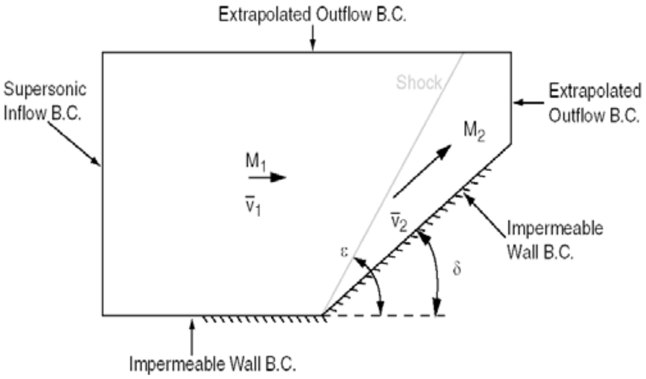

Figure 2.1: Schematic of an Oblique-Shock Flow.

B.2.2. Equations Solved, Boundary Conditions and Constitutive Properties

This verification test case is for the (two-dimensional or three-dimensional) Euler Equations. The boundary conditions are as indicated in Figure 2.1. The specific locations of the boundaries in the computational domain is not critical, provided the boundaries are located sufficiently far from the corner to allow the creation of sufficiently large, distinct regions of uniform flow. This case can also be used in the vanishing-viscosity limit for the Navier-Stokes Equations. In that case, the "impermeable wall" boundary conditions shown in Figure 2.1 should be replaced by "symmetry" boundary conditions. This problem does not require coordinate transformations, so these are disregarded here for this problem.

The gas is assumed to satisfy the ideal gas relation, pV = nRT, where p is the pressure, V is the volume, n is number of moles, R is the Universal Gas Constant, and T is the absolute temperature. The gas is assumed to have a constant ratio of specific heats. With this Equation of State, the analytical solution is independent of any other properties, except for the ratio of specific heats, γ, which appears as a parameter in the analytical solution, as given in Section B.2.3.

B.2.3. Comparison Metrics

A high level of agreement is expected between the computational and the analytical solutions for the shock angle ε , and for the "jump" ratios across the oblique shock. In particular, the velocity $\overline{V}_2$ should be parallel to the inclined wall, and the following "jump" relations are expected to be satisfied to within no more than 10% error, even on coarse grids:

$$\frac{1}{\tan\delta} = \left[\left(\frac{\gamma+1}{2}\right)\left(\frac{M_1^2}{M_1^2 \sin^2\varepsilon - 1}\right) - 1\right]\tan\varepsilon \tag{2.1}$$

$$\frac{p_2}{p_1} = \left(\frac{2\gamma}{\gamma+1}\right)M_1^2\sin^2\varepsilon - \left(\frac{\gamma-1}{\gamma+1}\right) \tag{2.2}$$

$$\frac{\rho_2}{\rho_1} = \frac{\tan\varepsilon}{\tan\delta} = \frac{(\gamma+1)M_1^2\sin^2\varepsilon}{2+(\gamma-1)M_1^2\sin^2\varepsilon} \tag{2.3}$$

$$\frac{|\overline{v}_2|}{|\overline{v}_1|} = \frac{\sin\varepsilon}{\sin(\varepsilon-\delta)}\left[\frac{2}{(\gamma+1)M_1^2\sin^2\varepsilon} + \left(\frac{\gamma-1}{\gamma+1}\right)\right] \tag{2.4}$$

Depending on the computational scheme and the grid resolution used, the oblique shock may be smeared and there may be local distortions in the regions where the shock meets the wall or the outflow boundary condition; but the size of the smear region should decrease monotonically with increasing grid refinement, and the jump conditions should still be accurately satisfied (when moving far enough away from the shock on both sides and far enough away from the wall boundary condition), even on coarse grids. The agreement between the computed values and the corresponding analytical ones should also remain satisfied or monotonically improve with increasing grid refinement.

B.2.4. Approach to the Exact Solution

The approach of the computational solution to the self-similar exact solution for this test case depends more on grid resolution than on locations and distances. As described in Section B.2.3 above, the coarser the grid, the more smeared the shock. However, the jump conditions should still be accurately satisfied for any level of grid refinement. The degree of spatial uniformity of the solution in front of and behind the shock should depend almost entirely on the refinement (and if applicable, the alignment) of the grid and the type of computational scheme used.

For higher-order computational schemes, there should be only relatively weak pre- or post-shock oscillations in the solution, with amplitudes no greater than 10-20% of the amplitude of the jump. The amplitude of any such oscillations should also not continue to increase beyond a certain value with increasing grid resolution.

B.2.5. Mathematically-Complete Definition of the Test Case

Sections B.2.1 and B.2.2 above contain the complete mathematical definition of the generic oblique-shock problem. The application to a specific example is discussed below.

B.2.6. Use and Validity of the Verification Data

There is no need to check the validity of the verification for this case as the verification data is from an exact, analytical solution.

B.2.7. Graph of Exact Solution

Several line plots below show the analytical and the computational results for the pressure along a streamline that crosses the shock wave. These line plots show how the computational solution approximates the exact solution.

B.2.8. Specific Example

The following is a specific example for this verification test case:

For $M_1 = 3.0$, $\delta = 15^o$, and $\gamma = 1.4$, the following should result: $M_2 = 2.255$, and $\varepsilon = 32.24^o$. The property ratios across the shock, which here are given by the oblique-shock relations, should be as follows: $p_2/p_1 = 2.822$, $\rho_2/\rho_1 = 2.0342$, and $v_2/v_1 = 0.888$.

B.2.9. Computed Solution for Specific Example

Figures 2.2-2.7 show the Mach number contours for computational solutions obtained using different grid densities and spatial orders of accuracy for the Mach 3.0 specific example given in Section B.2.8 above. The figures show how the uniform regions are correctly predicted and how the jump conditions across the shock are accurately satisfied, even on coarse grids. Therefore, it is seen that the computational solutions agree with the corresponding analytical solution. The figures show that the spatial resolution of the computational solution monotonically increases with increasing grid resolution. The figures also show how second-order spatial accuracy decreases the width of the shock, and also introduces slight oscillations in the solution. The grid lines are omitted for the finer grids in order to keep the contour lines clearly visible. Closer inspection of these figures reveals that the corner region introduces additional local distortions (due to numerical dissipation) in the computed solution, resulting in a local reduction of the post-shock Mach number below the values obtained away from the corner region (which are closer to the exact value of 2.255).

Figures 2.8-2.13 below correspond to Figures 2.2-2.7, and depict the corresponding entropy-change contours for the different computed solutions. The entropy change here is computed according to the equation

$$S_2 - S_1 = \left(\frac{\gamma R}{\gamma - 1}\right) \ln\left(\frac{T_2}{T_1}\right) - R \ln\left(\frac{p_2}{p_1}\right) \qquad (2.5)$$

(or any of its other equivalents) where the symbols s, T, and p denote the entropy, the absolute temperature, and the pressure, respectively. The subscript 2 denotes the variable, arbitrary state, and the subscript 1 denotes the reference state, which here is taken to be the inlet state, R and γ denote the gas constant and the ratio of specific heats for the specific gas being modeled, respectively, which in this example have the values of 287 J/kg/K and 1.4, respectively. Finally, ln(x) denotes the natural logarithm of x.

The exact analytical solution gives the increase in entropy (that is, the jump in entropy) across the shock wave for this test case as 22.21 J/kg/K. Figures 2.8-2.13 show how the shock in the computational solution generates a uniform jump in entropy across it, in qualitative agreement with the corresponding analytical solution. The figures also show how in the computational solution, the numerical dissipation in the shock region generates an additional increase in the entropy over that of the analytical solution, and how the numerical dissipation in the corner region generates a further significant localized increase in the entropy. Both of these numerically-generated increases are superimposed on the physical jump in the entropy. The results show how the numerically-generated entropy flows downstream with the flow from the point of generation, affecting all downstream points, as expected. This effect is clearly evident for the entropy increase which is numerically generated in the corner region.

Figures 2.8-2.13 show that the numerically-generated entropy increase becomes smaller with increasing grid refinement (albeit very slowly) and then asymptotes to a fixed level. They also depict how the numerically-generated entropy increase drops by about a factor of 25% or so in going from a spatially first-order-accurate to a spatially second-order-accurate computation with the same grid refinement. The results show how the numerically-generated entropy increase in the corner region becomes more localized (into the corner region and in the layer of downstream flow originating in the corner region) with increasing grid refinement and in going from a first-order accurate solution to a second-order solution, as expected. The slight “negative” entropy change shown in some of the figures is non-physical and is due to either boundary effects or to the expected numerical oscillations in second-order-accurate solutions.

Figures 2.14-2.18 correspond to Figures 2.2-2.7 (but with the solutions on additional grids with intermediate densities also included), and show the pressure distributions along the streamline plotted in the latter figures. The line plots in figures 2.14-2.18 show how the jump conditions are accurately satisfied, even on coarse grids, how the computed pressure profiles are closely centered about the corresponding analytical jump profile (as required by the conservation property of the numerical scheme used here), and how the resolution of the jump monotonically improves with increasing grid refinement. The figures also show the effect on the spatial resolution of going from first-order accuracy to second-order accuracy for each of the grid densities used, and how the first-order-accurate and second-order-accurate solutions converge (toward each other) as the grid refinement increases.

The computational solutions from any valid flow solver should exhibit the behaviors described above.

Figure 2.2: Mach Number Contours for a Spatially-First-Order Accurate Computation on a 20x10 Grid.

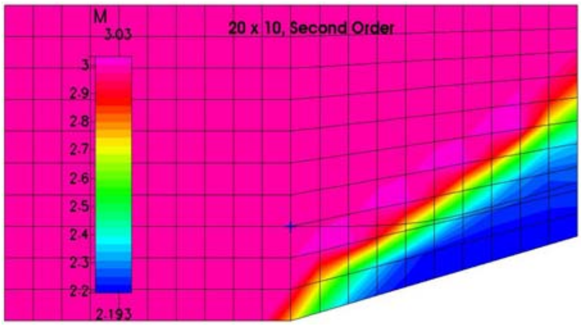

Figure 2.3: Mach Number Contours for a Spatially-Second-Order Accurate Computation on a 20x10 Grid.

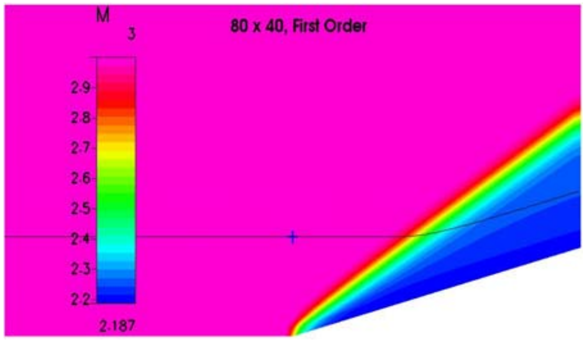

Figure 2.4: Mach Number Contours for a Spatially-First-Order Accurate Computation on an 80x40 Grid.

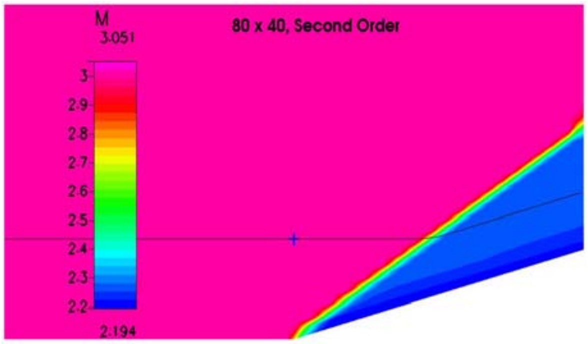

Figure 2.5: Mach Number Contours for a Spatially-Second-Order Accurate Computation on an 80x40 Grid.

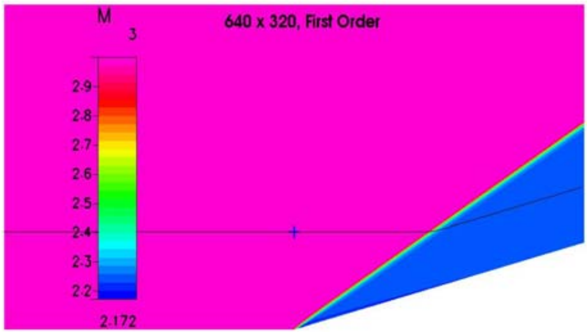

Figure 2.6: Mach Number Contours for a Spatially-First-Order Accurate Computation on a 640x320 Grid.

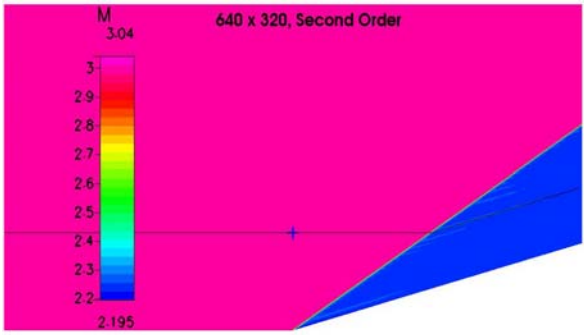

Figure 2.7: Mach Number Contours for a Spatially-Second-Order Accurate Computation on a 640x320 Grid.

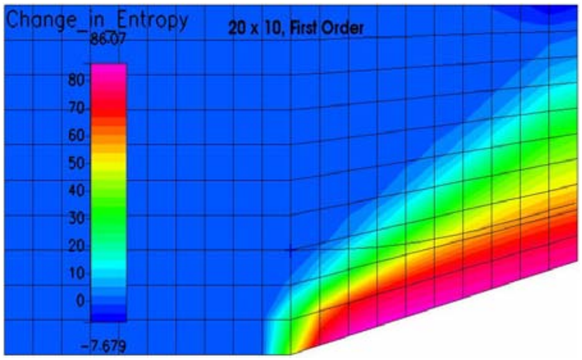

Figure 2.8: Entropy Change Contours for a Spatially-First-Order Accurate Computation on a 20x10 Grid.

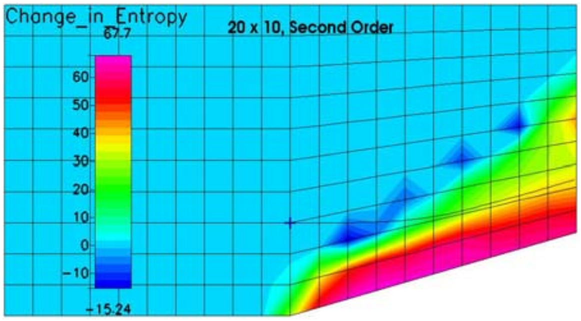

Figure 2.9: Entropy Change Contours for a Spatially-Second-Order Accurate Computation on a 20x10 Grid.

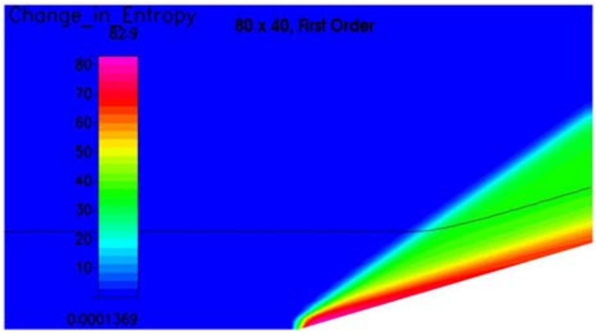

Figure 2.10: Entropy Change Contours for a Spatially-First-Order Accurate Computation on an 80x40 Grid.

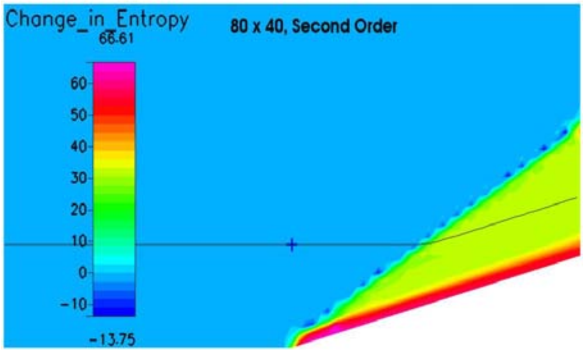

Figure 2.11: Entropy Change Contours for a Spatially-Second-Order Accurate Computation on an 80x40 Grid.

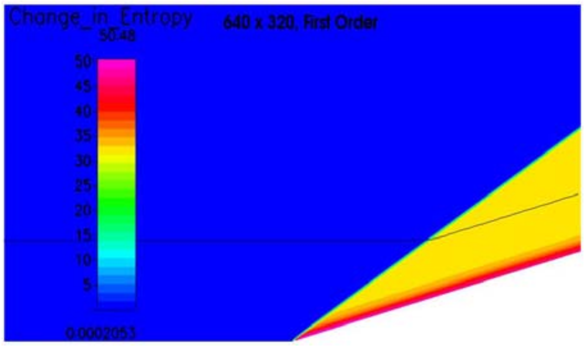

Figure 2.12: Entropy Change Contours for a Spatially-First-Order Accurate Computation on a 640x320 Grid.

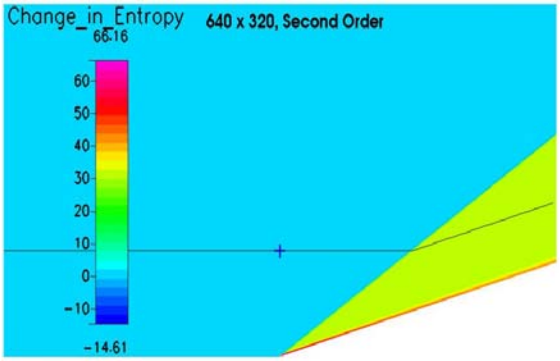

Figure 2.13: Entropy Change Contours for a Spatially-Second-Order Accurate Computation on a 640x320 Grid.

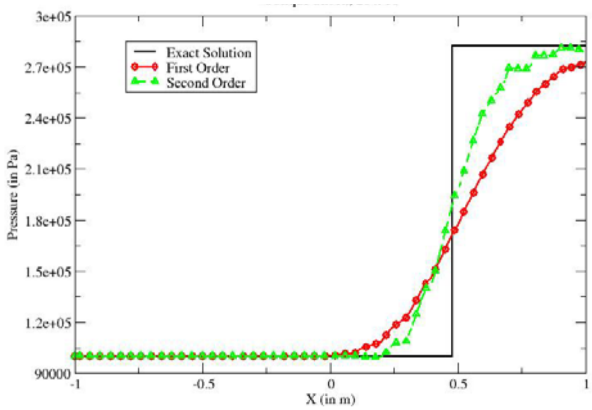

Figure 2.14: Line Plot of the Pressure Distribution Along a Streamline on a 20x10 Grid.

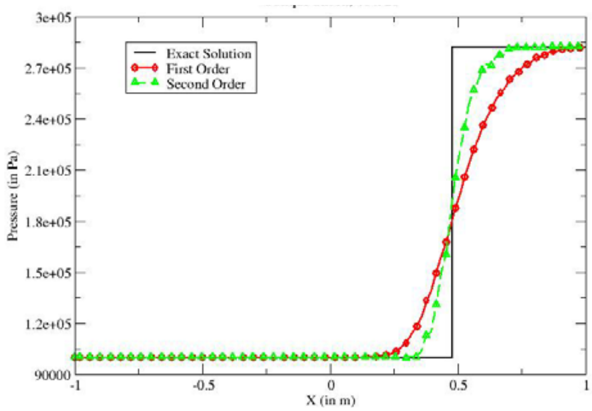

Figure 2.15: Line Plot of the Pressure Distribution Along a Streamline on a 40x20 Grid.

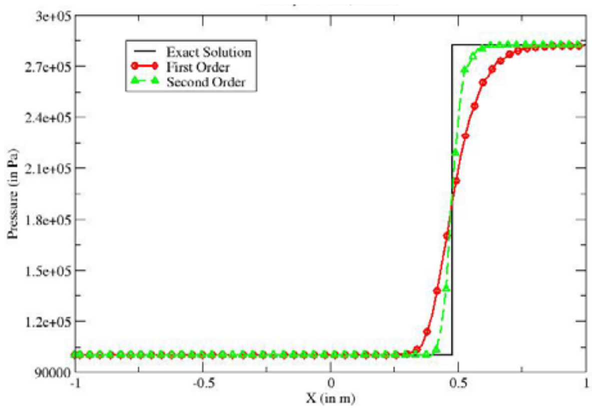

Figure2.16: Line Plot of the Pressure Distribution Along a Streamline on an 80x40 Grid.

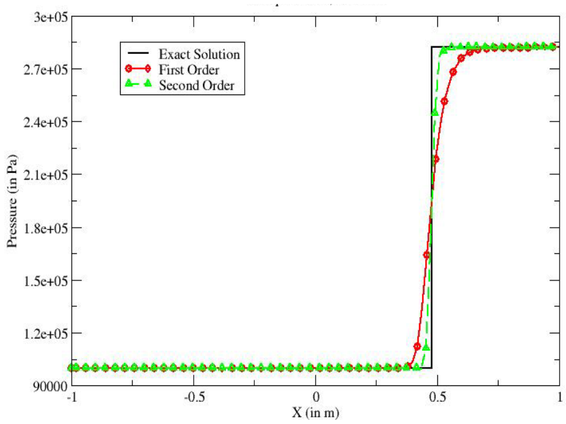

Figure 2.17: Line Plot of the Pressure Distribution along a Streamline on a 160x80 Grid.

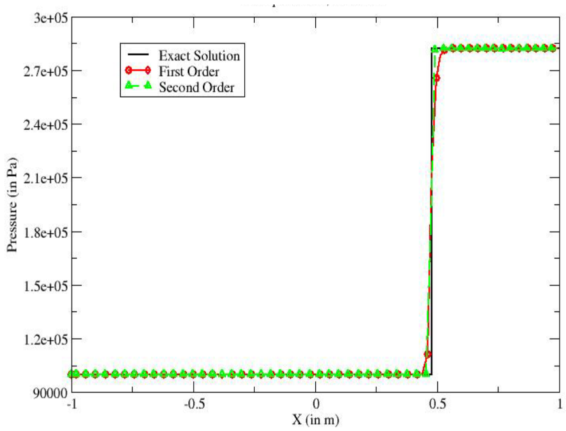

Figure 2.18: Line Plot of the Pressure Distribution along a Streamline on a 640x320 Grid.

Figure 2.19 shows the level of convergence expected in a computational solution for this test case: the residuals should drop by at least 12 orders of magnitude for double-precision computations, and at least six orders of magnitude for single-precision computations, for both the first-order-accurate and the second-order-accurate computations.

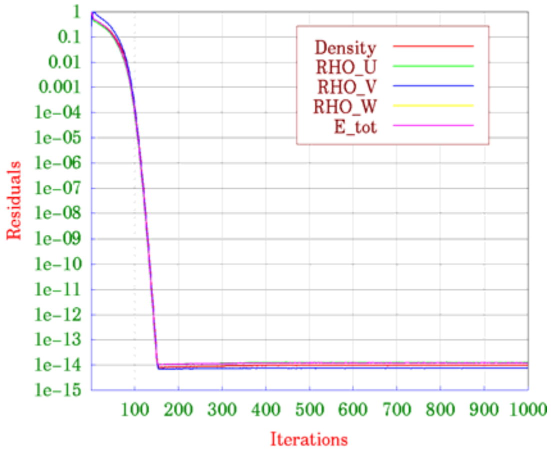

Figure 2.19: Residual Plot Showing Full Convergence for a First-Order-Accurate Double-Precision Computation.

B.2.10. Grid Quality and Grid Refinement Issues

For this validation case, the dependence of the solution quality on grid refinement has been discussed earlier, and amply demonstrated in Section B.2.9. However, the dependence of the solution quality on grid alignment is also high. In particular, the more closely aligned the grid lines are with the reflected shock, the crisper the shock would be, and the lesser the extent of any spurious artifacts in the computed solution, such as incorrect super-imposed local currents resulting from grid-induced artificial misalignments of the property jumps across the shock.

B.3. Incompressible Laminar Boundary-Layer Flow Solutions in Similarity Variables

B.3.1. Introduction and Motivation

This test case provides verification data for the streamwise asymptotic solution for two-dimensional (wedges) and axisymmetric (cones) external flows past semi-infinite bodies. It also serves as an estimate for the local solution for unseparated external wall-bounded flows.

The verification data is from the solution of a set of ordinary differential equations. Since only one coordinate needs to be discretized, these solutions can be obtained with high accuracy, using as many points as needed for achieving the desired accuracy.

B.3.2. Description

A stream of density ρ and viscosity μ flows at constant velocity U past a wedge of included angle α. The flow in the boundary layer that develops at the surface is represented by the boundary-layer equations expressed in terms of surface-normal coordinates (s,n). The pressure gradient along the boundary layer is obtained from the corresponding inviscid flow past the wedge.

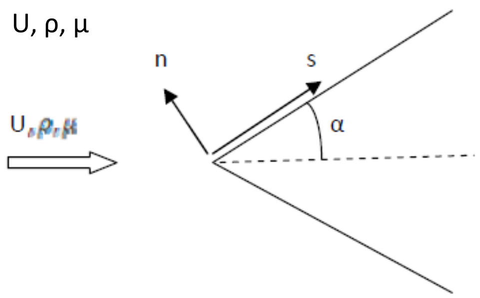

Fig.3.1: Schematic of Flow of Uniform stream Past Wedge

B.3.3. Governing Differential Equations and Boundary Conditions

The equations solved are the incompressible laminar boundary-layer equations for two-dimensional flow expressed in terms of similarity variables, namely, Görtler variables (ξ,η), obtained by a similarity transformation of the physical coordinates (s,n). These equations have the following form:

Momentum Equation:

$$\frac{\partial^2 F}{\partial \eta^2} - V\frac{\partial F}{\partial \eta} - \beta[F^2 - 1] - 2\xi F\frac{\partial F}{\partial \xi} = 0 \qquad (3.1)$$

Equation for V:

$$\frac{\partial V}{\partial \eta} + F + 2\xi\frac{\partial F}{\partial \xi} = 0 \qquad (3.2)$$

Where the pressure gradient parameter β is defined as

$$\beta = \frac{2\xi}{U_e}\frac{dU_e}{d\xi}, \qquad (3.3)$$

and F and V are dimensionless similarity=type variables defined as

$$F = f' = u/U_e \tag{3.4}$$

and

$$V = -f\ . \tag{3.5}$$

Here, $U_e = U_e\ (s)$ is the streamwise velocity at the edge of the boundary-layer $\delta(s)$, and is obtained from the outer inviscid flow as the inviscid velocity at the body surface.

The coordinates ξ and η appearing above, are defined as

$$\xi(x) = \int_0^x U_e dx \tag{3.6}$$

$$\eta(n, s) = n / \delta(s) \tag{3.7}$$

The surface boundary conditions correspond to zero-slip at the non-porous wall. The boundary conditions at the outer edge of the boundary layer are obtained by matching the edge values with the surfaces of the inviscid flow. The outer boundary must be placed sufficiently far from the surface so as to avoid the spatial oscillations ion the velocity profile.

The boundary conditions are stated mathematically as:

$$\begin{aligned} &At\ \ \eta = 0: \ \ F(\xi, \eta) = 0,\ V = 0; \\ &and\ \ as\ \eta \to \infty: \ \ F(\xi, \eta) \to 1 \end{aligned} \tag{3.8a, b}$$

At $\xi = 0$, and as $\xi \to \infty$, the above equations reduce to a self-similar form as follows:

Momentum Equation:

$$\frac{\partial^2 F}{\partial \eta^2} - V\frac{\partial F}{\partial \eta} - \beta[F^2 - 1] = 0 \tag{3.9}$$

Equation for V:

$$\frac{\partial V}{\partial \eta} + F = 0\ \ \ \ . \tag{3.10}$$

Furthermore, the self-similar form for the pressure gradient parameter β becomes

$$\beta = \text{constant}\ \ \ \ . \tag{3.11}$$

The ordinary differential equations, Eqs. (3.9) and (3.10), are the Falkner-Skan equations, and β is related to the wedge angle α as

$$\beta = \frac{2\alpha}{\pi}\ \ \ \ . \tag{3.12}$$

Equations (3.9) and (3.10) for the self-similar flow are solved with the corresponding boundary conditions shown in Eq. (3.8a, b).

B.3.4. Comparison Metrics

For semi-infinite bodies, exact comparison is expected asymptotically far downstream for the entire velocity profile. In addition, exact comparison is expected for the quantities

$$\tau_w = F', \tag{3.13}$$

$$\delta_1 = \eta|_{F=0.99},\ \delta_2 = \eta|_{F=0.999},\ \delta_3 = \eta|_{F=0.9999} \tag{3.14a, b ,c}$$

and,

$$\delta_d = \lim_{\eta \to \infty} (\eta - f) \tag{3.15}$$

where δ_d is the displacement thickness.

In Eq. (3.13), the first derivative is computed using a three-point second-order accurate forward difference.

For other wall-bounded external unseparated (attached) flows, the solution at other streamwise stations should compare qualitatively with the local similarity solution, i.e., with the solution of Eqs. (3.9) and (3.10), with β corresponding to the local conditions at these streamwise locations. Again, the comparison metrics should be the F(η) profile, and the local quantities τ_w, δ and δ_d.

B.3.5. Approach to Exact Solution

The approach to the self-similar exact solution in this test case depends on the grid resolution as well as on the locations of the streamwise station considered. As described in Section B.3.5, a coarse grid can affect the value of τ_w.

B.3.6. Results

The self-similar solutions obtained for β = 1.0, 0.5 and 0 are shown in Fig. 3.2, and correspond to 2-D stagnation-point flow (Hiemenz flow), axisymmetric stagnation-point flow (Homann flow), and flat plate flow (Blasius flow), respectively. These solutions were obtained using 100 points in the η direction.

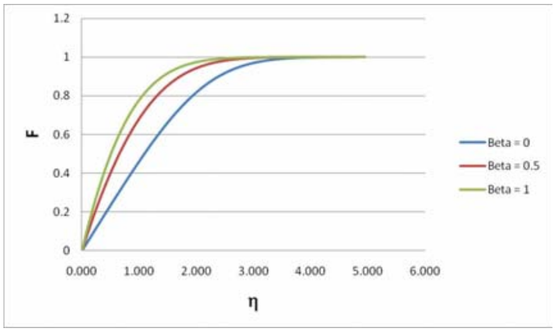

Fig. 3.2: Self-Similar Boundary-Layer Velocity Distribution; β = 0, 0.5, 1.0.

The profiles show that, as β increases from zero to unity, the boundary-layer thickness decreases, the velocity profile is fuller and the wall shear increases. The corresponding numerical values are listed in Table A.1 in Appendix A. These were obtained by solving the self-similar equations (3.9) and (3.10) in terms of η, with the boundary-layer edge being placed at η = 5.0.

B.3.8. Grid Refinement study

In this study we compare the values of F(η) for various values of step size Δη, ranging from 0.05 to 0.005. The maximum value of η was taken to be 5; hence, these increments correspond to 100 to 1000 points, respectively. Results are presented in Figs. 3.3 to 3.7, and Tables A.1 and A.2 in Appendix A.

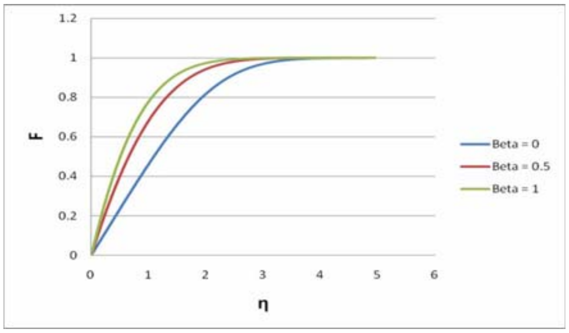

Fig. 3.3: F vs. η evaluated using 100 points

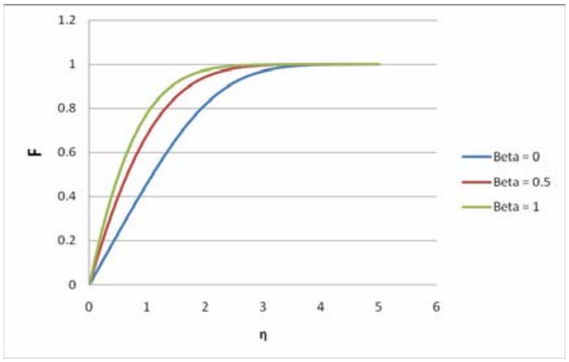

Fig. 3.4: F vs. η evaluated using 200 points

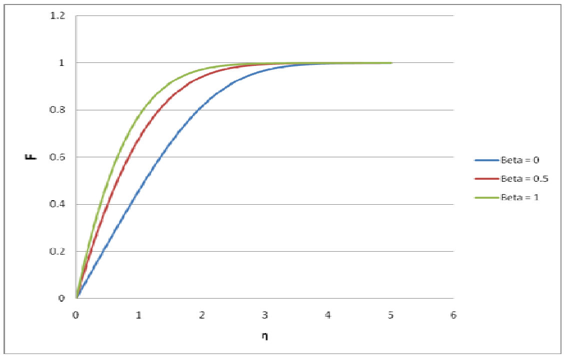

Fig. 3.5: F vs. η evaluated using 300 points

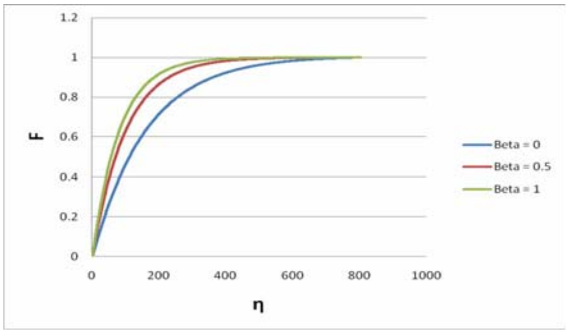

Fig. 3.6: F vs. η evaluated using 800 points

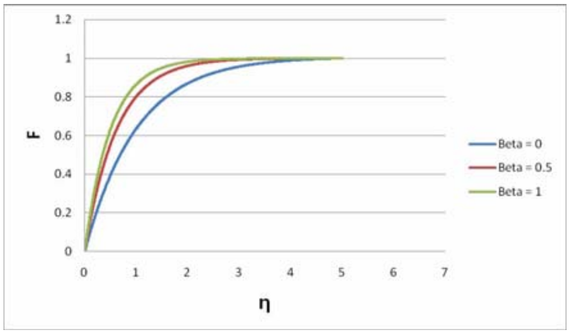

Fig. 3.7: F vs. η evaluated using 1000 points

As Table A.2 shows, a step size of Δη =0.025 is needed in order to achieve 5-decimal place accuracy for τ_w for $\beta = 0$. To achieve 6-decimal place accuracy for the same quantity, it is necessary to use $\Delta\eta = 0.00625$. Similar deductions can be made for other comparison metrics using the grid-refinement results in Table A.2.

B.4. Couette Thermal Flow

B.4.1. Introduction

The following test case is extracted from a book written by Charles Hirsch [21]. It presents one of the simplest cases to verify the discretization of viscous terms in the Navier-Stokes equations: laminar flow between two infinite parallel walls - a fixed wall and a moving wall at a distance L, with the moving wall having a velocity U. In order to include the computation of the thermal fluxes, both walls are considered as isothermal, but at different temperatures. The moving upper wall has a higher fixed temperature than the static wall; see Figure. 4.1. The analytical solution for the incompressible flow situation is easily derived. Since the plates are of infinite length, there is no physically relevant length scale in the streamwise direction and, therefore, all x-derivatives must vanish. In addition, since the flow has only a streamwise velocity component, the normal velocity component is zero everywhere, that is v = 0, reducing the vector momentum equation to its streamwise component only. The shear stress tensor is also reduced to a single component τ_{12},

$$\tau_{12} = \mu \frac{\partial u}{\partial y} \quad all\,other\,\tau_{ij} = 0 \tag{4.1}$$

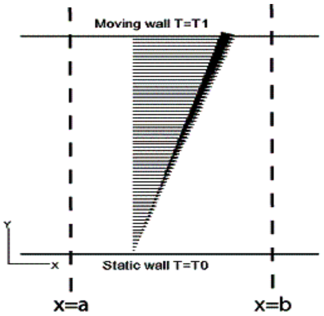

Figure 4.1: Representation of the Couette thermal flow test case

B.4.2. Governing Differential Equations and Boundary Conditions

The Navier-Stokes equations simplify considerably to the one-dimensional system

$$\frac{\partial u}{\partial t} = \vartheta \frac{\partial^2 u}{\partial y^2} \tag{4.2}$$

$$\rho C_v \frac{\partial T}{\partial t} = k \frac{\partial^2 T}{\partial y^2} + \mu \left(\frac{\partial u}{\partial y} \right)^2 \tag{4.3}$$

Note that these equations are parabolic in space and time, since all the convection terms have vanished.

These equations are solved with the following Dirichlet boundary conditions for velocity and temperature

y=0 : u=0, $T=T_0$, (4.4)

y=L: u=U, $T=T_1$, (4.5)

For the steady-state solution, the time derivatives are set to zero, and the momentum equation reduces to

$$\frac{\partial^2 u}{\partial y^2} = 0 \quad , \tag{4.6}$$

leading to a linear velocity profile

$$u(y) = \frac{y}{L} U \quad . \tag{4.7}$$

Introducing this solution in the steady temperature equation, leads to,

$$\frac{d^2 T}{dy^2} = \frac{-\mu}{k} \left(\frac{du}{dy} \right)^2 = \frac{-\mu U^2}{kL^2} \quad . \tag{4.8}$$

This generates a parabolic temperature profile, depending on the parameter in the right hand side. The velocity scale is fixed by the upper wall velocity U, the temperature scale is determined by the temperature difference $\Delta T = T_1 - T_0$, and the length scale by the distance L between the plates. Hence, the non-dimensional variables are defined as

$$\tilde{U} = u / U \qquad \tilde{T} \overset{\Delta}{=} \frac{T(y) - T_0}{T_1 - T_0} \qquad Y = y / L \tag{4.9}$$

$$\frac{d^2 \tilde{T}}{dY^2} = \frac{-\mu U^2}{k \Delta T} \tag{4.10}$$

The right-hand side coefficient can be written as the product of the Prandtl and Eckert non-dimensional numbers:

$$\Pr = \frac{\mu Cp}{k} \quad E_c = \frac{U^2}{Cp\Delta T} \quad \Pr E_c = \frac{\mu U^2}{k\Delta T} \quad . \qquad (4.11)$$

The Eckert number is the ratio of the dynamic temperature induced by fluid motion to the characteristics temperature difference in the fluid. The analytical velocity and temperature profiles are then easily obtained as

$$\begin{aligned} &\tilde{U}(Y) = Y \\ &\tilde{T}(Y) = \frac{T(y) - T_0}{T_1 - T_0} = \left[1 + \frac{1}{2}\Pr E_c (1-Y)\right] Y \end{aligned} \qquad (4.12)$$

The wall heat transfer coefficient is an important quantity in engineering applications, and is generally expressed by the non-dimensional Nusselt number, defined as a measure of the intensity of the heat flux as,

$$Nu \overset{\Delta}{=} \frac{d\tilde{T}}{dY} = \frac{L}{\Delta T}\frac{dT}{dy} \qquad (4.13)$$

For the Couette flow, it takes the following values at the wall

$$\begin{aligned} Nu &= 1 + \frac{\Pr E_c}{2} \qquad at\ y = 0, \\ Nu &= 1 - \frac{\Pr E_c}{2} \qquad at\ y = L. \end{aligned} \qquad (4.14)$$

An interesting property of this solution is that the fluid maximum temperature is greater than the upper wall temperature, when the product PrEc is greater than two.

B.4.3. Numerical simulation conditions

Since this is a one-dimensional flow case, with no changes in the x-direction, it can be simulated by generating a two-dimensional mesh with a limited number of mesh points in the x-direction, and applying periodic boundary conditions at the two ends of the x-domain, i.e., at x=a and x=b, as shown in Figure 4.1. The periodic boundary conditions express that all quantities at x=b are equal to their corresponding values at the same ordinate at x=a. In principle, two or three mesh points should be sufficient in the x-direction. Another issue is connected to the numerical values of the flow variables. Although the non-dimensional solution is independent of the levels of temperature differences and physical distances between the two walls, the flow solver code is written for the physical variables and, consequently, their numerical values can influence the overall accuracy of the computed results. While using a density-based code, we consider low compressible conditions, with a Mach number around 0.1. This is totally acceptable for the analysis of incompressible flows, assuming perfect gas relations for the considered fluid. Because of the thermal effects, we have to ensure that, numerically, the perfect gas relations still remain close to constant-density conditions and, therefore, we have to limit the absolute values of the temperature difference between the two endplates. Since the non-dimensional solution depends only on the product PrEc, this parameter defines completely the numerical solution, when obtained in the non-dimensional form. As this is not the case here, we have to select all the physical quantities of the fluid and the physical set-up, in order to fully define the dimensional form of the solution.

We select here the following values:
The fluid is a perfect gas with the following properties:

Specific Heat: c_p = 1006 J.kg^{-1}.K^{-1}
Gamma: γ = 1.4
Kinematic viscosity: $\nu = 1.57\ 10^{-5}$ m^2/s
Prandtl number: Pr = 0.708.
The Reynolds numbers based on the velocity of the moving wall is 4,000.
The physical conditions of this Couette flow are chosen such that PrEc = 4, with the following variables set according to
L = 0.83 mm
$T_0 = 293^{o}$K
$T_1 = 294^{o}$K
U = 75.4 m/s

Note the very small temperature difference of 1^{o}K selected; which requires double precision arithmetic. Since this problem is slightly more complex than the simple Couette flow without any thermal effect, it is recommended that we verify first the implementation of the viscous effects, without taking into account the thermal fluxes, to confirm that the linear velocity distribution is obtained.

B.4.4. Grid definition

A regular grid has been used for this case with 65 points over a length L, between the walls and 3 points in the axial direction. The two lateral sides of the computational domain are connected assuming a periodic repetition of the channel.

In order to investigate the sensitivity of the numerical results to the grid density, coarser meshes are constructed by simply removing each second point in the vertical wall-to-wall direction. Hereby, we generate four different grids: (65×3), (33×3), (17×3), and (9×3).

B.4.5. Results

The accuracy of the solution is strongly influenced by the grid density used. On the one hand, all the meshes used are able to reproduce the linear distribution of axial velocity between the walls; see Figure 4.2, demonstrating the second-order accuracy of the scheme, ***for which a linear variation has to be exactly reproduced***. On the other hand, the quadratic distribution of temperature is not accurately captured when using very coarse meshes; see figure 4.3. The meshes with 33 and 65 points from wall-to-wall are able to accurately predict the analytical solution. This finding illustrates that sufficient refinement is necessary in order to capture the flow features present in laminar boundary layers.

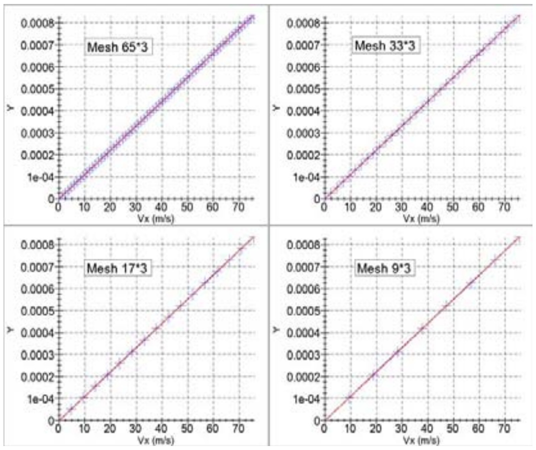

Figure 4.2: Wall-to-wall distribution of axial velocity as deduced from analytical result (continuous line) and from the numerical result (plus signs).

An important aspect of many engineering flow problems is the ability to predict local extrema of important flow variables. Since thermal stresses have an impact on solid structure lifetime, it is of importance to accurately predict the maximum temperature inside a flow. Therefore, we will perform a grid convergence study based on the prediction of the maximum temperature between the solid walls. The maximum temperature inside the flow is obtained at a position y/L=3/4 from the static wall, and its value is 294.125 K. As expected, the maximum-temperature error is reduced if the mesh is refined, and an error lower than 1% of the temperature variation can be obtained with a mesh having at least 33 grid cells in the wall-normal direction. This number has to be doubled if we need a precision of less than 0.1%.

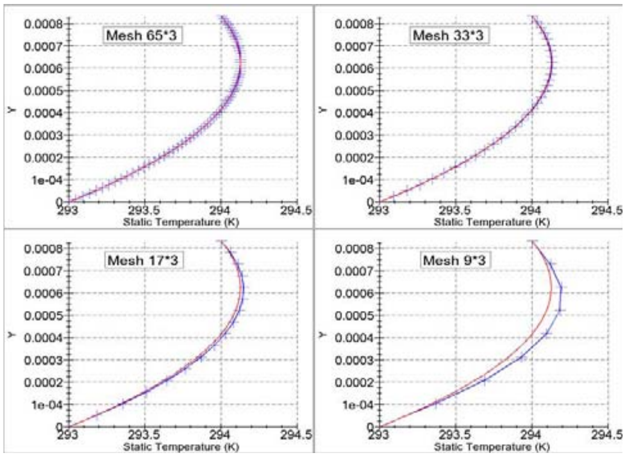

Figure 4.3: Wall-to-wall distribution of static temperature (line with plus signs), compared to the analytical solution (continuous line), for the four different grids.

Hereafter, only a brief write-up is presented about some additional verification cases which admit exact solutions, with additional details provided in the corresponding references.

B.5. Uniform Stream as a Code Verification Case

A uniform stream is a flow with a uniform constant velocity at every point in the flow. Because of its simplicity, and the ready availability of a known exact solution, viz., a constant value of velocity, its use is perforce as the most preliminary test case in the verification of CFD codes.

The uniform stream case is characterized by a constant and uniform velocity at all points in the flow domain, including the boundaries. This ascribes to both internal and external flows. At all boundaries in the flow domain, a velocity of u=U, v=V, w=W is stipulated, where U, V and W are the components of the free stream velocity in the coordinate directions. Regardless of a code's computational ability, or the intricate characteristics of a problem being solved, with or without any coordinate transformations, the uniform stream case should be employed by merely configuring all boundaries to u=U, v=V, w=W.

In problems where one or more boundaries move with respect to time, it becomes necessary to solve the time-dependent-governing equations on a moving mesh. In such cases, geometric quantities, involving the new mesh position and velocity, need to be computed in such a way that a constraint, on the mesh position and velocity, called the Geometric Conservation law (GCL) , as coined by Thomas and Lombard (1979)[22], is satisfied. In the numerical perspective, a discrete version of GCL (DGCL) needs to be implemented. The DGCL establishes a constraint on the mesh position and velocity of the grid, and the numerical scheme employed must satisfy the constraint in order to exactly compute a uniform flow field [23].

Only when the uniform stream case is strictly satisfied, devoid of any numerical complications and uncertainties, should any CFD code be furthered, both in terms of development or usage.

B.6. Method of Manufactured Solutions

The method of manufactured solutions (MMS) [10] is a general and very powerful approach to code verification. Instead of 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. 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 solution.

B.6.1 MMS Procedure

The procedure for applying MMS with order of accuracy verification can be summarized in the following six steps:

Step 1. Choose the form of the governing equations.
Step 2. Choose the form of the manufactured solution.
Step 3. Derive the modified governing equations.
Step 4. Discretize the modified governing equations and solve on multiple meshes.
Step 5. Evaluate the global discretization error in the numerical solution.
Step 6. Apply the order of accuracy test to determine if the observed order of accuracy matches the formal order of accuracy.

The fourth step, which includes the solution to the modified governing equations, may require code modifications to allow arbitrary source terms, initial conditions, and boundary conditions to be used.

Manufactured solutions should be chosen to be smooth, analytical functions with smooth derivatives. The choice of smooth solutions will allow the formal order of accuracy to be achieved on relatively coarse meshes, and trigonometric and exponential functions are recommended. It is also important to ensure that no derivatives vanish, including cross-derivatives. Care should be taken that one term in the governing equations does not dominate the other terms. For example, when verifying a Navier-Stokes code, the manufactured solution should be chosen to give Reynolds numbers near unity so that convective and diffusive terms are of the same order of magnitude. Finally, realizable solutions should be employed, that is, if the code requires the temperature to be positive (e.g., in the evaluation of the speed of sound which involves the square root of the temperature), then the manufactured solution should be chosen as such.

B.6.2 MMS for Euler Equations

MMS has been applied to the Euler equations, which govern the flow of an inviscid (frictionless) fluid [19]. The two-dimensional, steady-state form of the Euler equations is given by

$$
\begin{aligned}
\frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} &= f_m \\
\frac{\partial(\rho u^2 + p)}{\partial x} + \frac{\partial(\rho uv)}{\partial y} &= f_x \\
\frac{\partial(\rho vu)}{\partial x} + \frac{\partial(\rho v^2 + p)}{\partial y} &= f_y \\
\frac{\partial(\rho u e_t + pu)}{\partial x} + \frac{\partial(\rho v e_t + pv)}{\partial y} &= f_e
\end{aligned}
\tag{6.1}
$$

where arbitrary source terms f are included on the right-hand side, and e_t is the specific total energy, which for a calorically perfect gas is given by

$$
e_t = \frac{1}{\gamma - 1} RT + \frac{u^2 + v^2}{2} \tag{6.2}
$$

The final relation needed to close the set of equations is the equation of state for a calorically perfect gas

$$
p = \rho RT \tag{6.3}
$$

The manufactured solution for this case is chosen as

$$
\begin{aligned}
\rho(x,y) &= \rho_0 + \rho_x \sin\left(\frac{a_{\rho x}\pi x}{L}\right) + \rho_y \cos\left(\frac{a_{\rho y}\pi y}{L}\right) \\
u(x,y) &= u_0 + u_x \sin\left(\frac{a_{ux}\pi x}{L}\right) + u_y \cos\left(\frac{a_{uy}\pi y}{L}\right) \\
v(x,y) &= v_0 + v_x \cos\left(\frac{a_{vx}\pi x}{L}\right) + v_y \sin\left(\frac{a_{vy}\pi y}{L}\right) \\
p(x,y) &= p_0 + p_x \cos\left(\frac{a_{px}\pi x}{L}\right) + p_y \sin\left(\frac{a_{py}\pi y}{L}\right)
\end{aligned}
\tag{6.4}
$$

The subscripts here refer to constants (not differentiation) with the same units as the variable, and the dimensionless *a* constants generally vary between 0.5 and 1.5 to provide smooth solutions over an L×L square domain. For this case, the constants were chosen to give supersonic flow in both the positive x and positive y directions (see Table B.1 in Appendix B). While not necessary, this choice simplifies the inflow boundary conditions to Dirichlet values at the inflow and Neumann (gradient) values at the outflow. The inflow boundary conditions are determined from the manufactured solution. Plots of the manufactured solution for the conserved variables are shown in Figures 6.1 to 6.4.

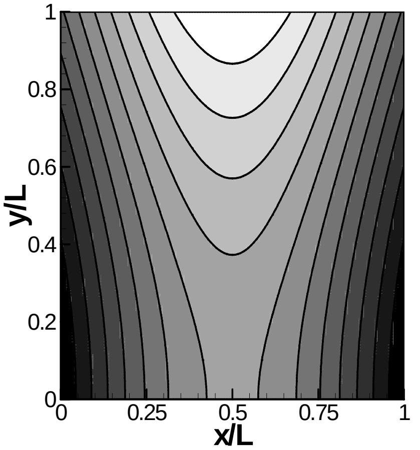

Figure 6.1: Euler manufactured solution for ρ

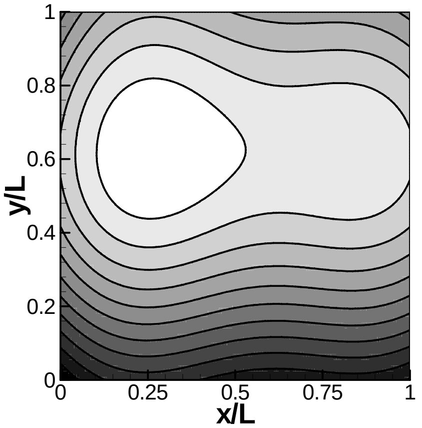

Figure 6.2: Euler manufactured solution for ρe_t

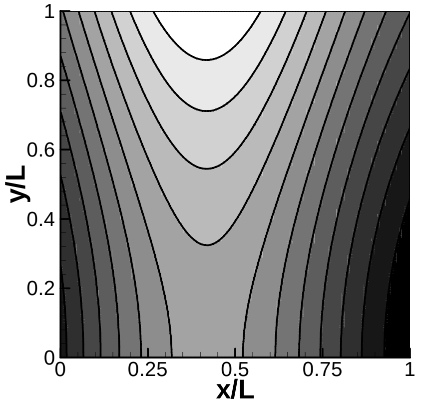

Figure 6.3: Euler manufactured solution for ρu

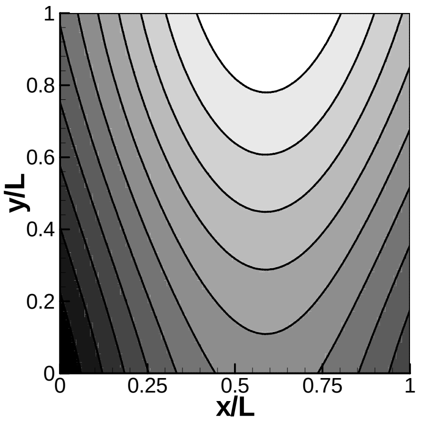

Figure 6.4: Euler manufactured solution for ρv

Substitution of the chosen manufactured solutions into the governing equations allows the analytical determination of the source terms.

As the governing equations get more complex, the use of symbolic manipulation software such as Mathematica™ or Maple™ is strongly recommended. These software packages generally have built-in capabilities to output the source terms as computer code in both FORTRAN and C programming languages. Plots of the source terms for the four governing equations (mass, momentum, and energy) are shown in Figures 6.5 to 6.8.

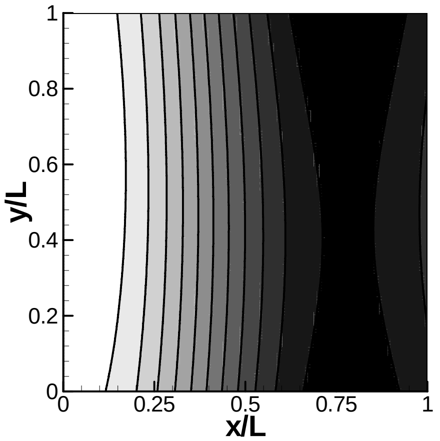

Figure 6.5: Euler source term for mass equation.

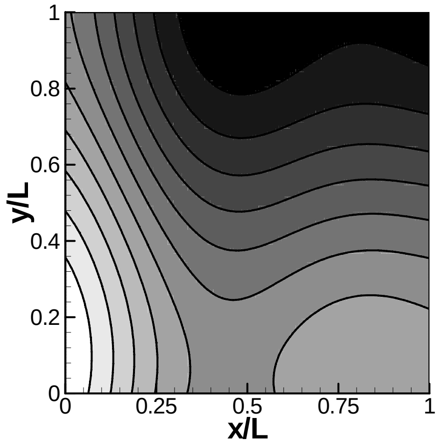

Figure 6.6: Euler source term for energy equation.

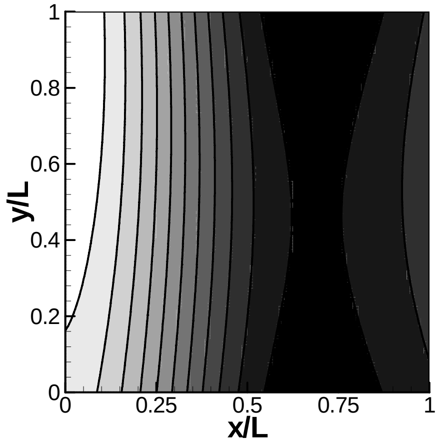

Figure 6.7: Euler source term for x-momentum equation.

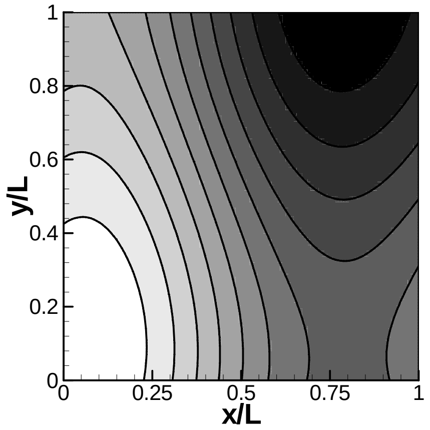

Figure 6.8: Euler source term for y-momentum equation.

The governing equations are then discretized and solved on multiple meshes. In this case, two different finite-volume Navier-Stokes codes were employed: Premo, an unstructured grid code, and Wind, a structured grid code (see Ref. [19] for details). Both codes utilized the second-order Roe upwind scheme for the convective terms, and second-order central differencing for the diffusive terms. The formal order of accuracy of both codes is thus second order. In the current example, five meshes were employed. It is, however, important to note that the computations in this case were performed on Cartesian meshes for simplicity; a more general code verification analysis should employ meshes with significant stretching and skewness, as well as arbitrary orientations for domain boundaries.

MMS can be also be used to exercise the features of a turbulent flow situation. Eca et al, [25] presented manufactured solutions for code verification of two-dimensional, steady, wall bounded, incompressible turbulent flows for six-eddy-viscosity turbulence models. The manufactured solution prescribes the main flow variables and two turbulence quantities: eddy-viscosity and turbulence kinetic energy. In the most common eddy-viscosity models, all other turbulence quantities can be derived from these two turbulence quantities.

B.7. Heat Transfer in a smooth U-Duct with and without rotation

This test case shows that, when computing a compressible flow in a rotating duct of a given geometry and radial distance from the axis of rotation, the inlet Mach number must be specified in addition to the inlet Reynolds number, inlet rotation number, coolant-to-wall temperature ratio, and Prandtl number. This is because the inlet Mach number and other dimensionless parameters collectively fix the rotational speed, which strongly influences centrifugal buoyancy. This case also shows the nature of the three-dimensional flow induced by Coriolis force, centrifugal buoyancy, and a 180-deg bend in a U-shaped square duct with smooth walls for three rotation numbers (0, 0.24, and 0.48), and two Reynolds numbers (2.5×10^4 and 5×10^4). Key flow mechanisms that affect heat transfer are identified. The computed heat transfer coefficient on the leading and trailing faces of the rotating duct compares well with available experimental data. This computational study is based on the ensemble-averaged conservation equations of mass, momentum (compressible Navier- Stokes), and energy closed by $k-\omega$ shear-stress transport model of turbulence that can be integrated to the wall; i.e., wall functions were not used. Solutions were generated by using a cell-centered finite-volume method on a structured grid based on second-order accurate Roe differencing, and on a diagonalized alternating-direction implicit scheme with local time-stepping and V-cycle multigrid. Details on this test case are provided in [26].

B.8. Oblique Detonation with Straight Shock, Wall Curvature, and One-Step Kinetics

This test case discusses an exact solution for an oblique detonation and example of its use to verify a particular computational algorithm. Full details are given by Powers and Aslam [27], from which this document is directly derived. An oblique detonation is defined as a shock-induced combustion process in which the shock has a non-normal angle of inclination to the streamlines of the undisturbed flow. In a detonation, the shock triggers the combustion, and the combustion contributes to the support of the shock. It is generally the case that convective transport dominates diffusive transport. In a typical oblique detonation, an unreacted fluid particle traveling at supersonic velocity in the freestream encounters an oblique shock. Typically, the shock is supported by a stationary downstream solid wedge. As the fluid particle passes through the shock, it is irreversibly compressed such that its temperature rises sufficiently high to induce exothermic reaction. As the particle travels past the shock, it reacts, converting chemical energy into thermal and kinetic energy. This local energy release then propagates via acoustic waves within the domain between the wedge and the shock, and there is a complex series of reflections and interactions. Additionally, the fluid particle turns, so that in the far field, its pathline is parallel to the supporting wedge. An observable global consequence of this energy release and transport is that the shock is displaced further from the wedge than it otherwise would have been had the flow been inert.

The reaction is described by one-step, irreversible kinetics which allows for reactant depletion to extinguish the reaction. Arrhenius effects are embodied in a Heaviside step function which suppresses reaction before the shock and admits reaction following the shock. Consequently, the post-shock reaction is equivalent to an Arrhenius model with zero activation energy. This key assumption allows the equations to be more amenable to analytic solution, and the steady solutions to be less susceptible to instability. Next, an orthogonal coordinate transformation is employed. The flow is taken to be steady and have spatial variation with only one of the variables in the transformed frame. This, along with the assumption of a straight shock attached to a curved wedge, dictates that the flow is irrotational. The conservation of mass, momentum, and energy equations are then seen to admit algebraic solutions, similar to Rankine-Hugoniot relations, which allows one to express all variables as functions of the extent of reaction. These are deployed in the remaining ordinary differential equation for the reaction kinetics, whose simple form admits then an exact solution giving distance as a function of extent of reaction. Consequently, all flow variables have a parametric expression in terms of the extent of reaction. The exact solution is then employed for use in verification of a modern computational algorithm for shock-laden reactive flows. It is shown that a third-order time-accurate numerical algorithm, which can achieve fifth-order spatial accuracy for smooth flows, converges at less than first-order spatial accuracy for a flow containing a captured shock. This result is a feature of all shock-capturing schemes and is consistent with Godunov's Theorem 3, which formally applies only to linear algorithms, in contrast to the nonlinear method employed here.

B.9. Use of Burger's Equation as a Code Verification Tool

Burger's equation is a popular simplified version of the streamwise momentum equation in incompressible viscous flow. It has been used as a simple model of turbulence and for a model of surface growth by a chemical deposition process, where it is known as the Karder, Parisi and Zhang (KPZ) equation [28]. It is a good means of evaluating grid clustering algorithms in that the "viscosity" can be tuned to simulate regions of rapid change in the dependent variable. The equation is subject to the development of shocks which is also useful for evaluating the accuracy of numerical schemes.

Adaptive grid methods are particularly useful in unsteady problems containing regions of rapid variation. Burger's equation is a useful example because it resembles the boundary-layer equations. Originally introduced as a "mathematical model illustrating the theory of turbulence [29]", Burger's (1940) equation is now one of the most commonly used equations for the evaluation of numerical schemes. The viscous form of Burgers equation is a parabolic partial differentiation equation which mimics the boundary layer equations; however, the Burger's equation does not account for a pressure gradient or normal velocity component, and is restricted to flows with uniform velocity profiles. Because of its simple form, Burger's equation is an appropriate model equation for evaluation of numerical schemes employing adaptive-grid methods [30].

C. Closure

The final AIAA document will include the cases presented here, with full details and tabulated data. It will also include additional exact solutions that can be useful for code verification purposes.

D. Acknowledgments

Sami Habchi and Sami Bayyuk acknowledge and thank Vinod Venugopalan, Research Engineer, CFD Research Corporation, for running the test cases shown in Sections B.1 and B.2 and for plotting the results shown for these cases. Urmila Ghia would like to acknowledge Samuel Elekwachi and Santosh Konangi for generating the plots and the tabulated data for the self-similar wedge flow case.

E. References

1. "AIAA Guide for the Verification and Validation of Computational Fluid Dynamics Simulations," American Institute of Aeronautics & Astronautics, 2006.

2. Anderson, J. Jr., "Modern Compressible Flow, with Historical Perspective," Second Edition, McGraw-Hill Publishing Company, 1990.

3. Shapiro, A., "The Dynamics and Thermodynamics of Compressible Fluid Flow," Volume 1, John Wiley and Sons, 1953.

4. Zucrow, M. and Hoffman, J., "Gas Dynamics," Volume 1, John Wiley and Sons, 1976.

5. Evans, H. L., "Laminar boundary-layer theory," Addison-Wesley Publishers, 1968.

6. White, F.M., "Viscous Fluid Flow," McGraw-Hill International, 2006.

7. Rosenhead, L., "Laminar Boundary Layers," University Press, Oxford, 1963.

8. Anderson, D.A., Pletcher, H.R., and Tannehill, J.C., “Computational Fluid Mechanics and Heat Transfer,” Taylor & Francis, 1997.

9. Roy, C.J., “Review of code and solution verification procedures for computational simulation,” J. Comp. Physics 205 (1) (2005) 131-156 (see also Roy, C. J., “Verification of Codes and Solutions in Computational Simulation”, Proceedings of Computational Heat Transfer 2004, ICHMT International Symposium on Advances in Computational Heat Transfer, Norway, April 19-24, 2004).

10. Roache, P.J., “Verification and Validation in Computational Science and Engineering,” Hermosa Publishers, New Mexico, 1998.

11. Knupp, P., Salari, K., “Verification of Computer Codes in Computational Science and Engineering,” K.H. Rosen (ed), Chapman and Hall/CRC, Boca Raton, Florida, 2003.

12. Anderson, D.A., Tannehill, J.C., Pletcher, R.H., “Computational Fluid Mechanics and Heat Transfer,” Hemisphere Publishing Corp., New York, 1984, 70-77.

13. Roache, P.J., Steinberg, S., “Symbolic manipulation and computational fluid dynamics,” AIAA J. 22 (10) (1984) 1390-1394.

14. Roache, P.J., Knupp, P.M., Steinberg, S., Blaine, R.L., “Experience with benchmark test cases for groundwater flow, in Benchmark Test Cases for Computational Fluid Dynamics,” Celik I., and Freitas, C.J. (eds), ASME FED 93, Book No. H00598, 1990, 49-56.

15. Oberkampf, W.L., Blottner, F.G., “Issues in computational fluid dynamics code verification and validation,” AIAA J. 36 (5), 1998, 687-695 (see also Oberkampf, W.L., Blottner, F.G., Aeschliman, D.P., “Methodology for computational fluid dynamics code verification/validation”, AIAA Paper 95-2226, 1995).

16. Oberkampf, W.L, Trucano, G.T, “Verification and validation benchmarks,” Nuclear Engineering and Design 238, 2008, 716-743.

17. Salari, K., Knupp, P., “Code Verification by the Method of Manufactured Solutions,” SAND 2000-1444, Sandia National Laboratories, Albuquerque, New Mexico, 2000.

18. Roache, P.J., “Code verification by the method of manufactured solutions,” J. Fluids Eng. 124 (1), 2002, 4-10.

19. Roy, C.J., Nelson, C.C., Smith, T.M., Ober, C.C., “Verification of Euler / Navier-Stokes codes using the method of manufactured solutions,” Int. J. Numer. Meth. Fluids 44 (6), 2004, 599-620.

20. Roy, C.J., Blottner, F.G., “Assessment of one- and two-equation turbulence models for hypersonic transitional flows,” J. Spacecraft and Rockets 38 (5), 2001, 699-710.

21. Hirsch, Ch., “Numerical Computation of Internal and External Flows – The Fundamentals of Computational Fluid Dynamics”, 2nd edition, Elsevier, 2007.

22. Thomas, P.D., Lombard, C.K., “Geometric conservation law and its application to flow computations on moving grids,” Vol. 17, July 1979.

23. Guillard, H., Farhat, C., “On the significance of the geometric conservation law for flow computations on moving meshes,” Computer methods in applied mechanical engineering, 2000, pg. 1467-1482.

24. Kamakoti, R., Shyy,W., “ Evaluation of geometric conservation law using pressure-based fluid solver and moving grid technique,” International Journal for Numerical Methods in Heat and Fluid flow, Vol. 14, Number 7, pg. 851-865.

25. Eca, L., Hoekstra, M., Hay, A., Pelletier, D., Roache, P.J., 2005, “A Manufactured Solution to for a two-dimensional steady wall-bounded incompressible turbulent flow”, Instituto Superior Tecnico, IST Report D72-34.

26. Stephens, M.A., Shih, T.I-P., “Computations of Flow and Heat Transfer in a Smooth U-Shaped Square Duct with and without Rotation,” AIAA Journal of Propulsion and Power, Vol. 15, No. 2, pp. 272-279, March-April 1999.

27. Powers, J.M., and Aslam, T.D., 2006, “Exact Solution for Multidimensional Compressible reactive Flow for Verifying Numerical Algorithms,” AIAA Journal, Vol. 44, No. 2, pp. 337-344.

28. Karder, M., Parisis, G., Zhang, Y.C., Phy. Rev. let, Vol. 56, 889, 1986.

29. Burgers, J.M, “Application of a model to illustrate some points of the statistical theory of free turbulence,” Proc. Acad. Sci., Vol 43, p. 2, 1940.

30. Adams, E., Conlisk, A.T., Smith, F.T, “Adaptive grid methods for Vorter-Induced Boundary-Layers,” AIAA J., 33, no. 5, p. 864-870, 1995.

31. Johnson, R.W., Schultz, R.R, Roache, P.J, Celik, I., Pointer, W.D, Hassan, Y.A., “Processes and Procedures for application of CFD to Nuclear Reactor Safety Analysis,” Idaho National Laboratory, 2006.

Appendix A:

Table A.1. Self-Similar Boundary-Layer Velocity Distribution; β = 0, 0.5, 1.0.

Beta = 0				Beta = 0.5			Beta = 1		
Eta	**F**	**f**	**δ_d**	**F**	**f**	**δ_d**	**F**	**f**	**δ_d**
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.05000	0.02348	0.00059	0.04941	0.04576	0.00114	0.04886	0.06038	0.00151	0.04849
0.10000	0.04697	0.00235	0.09765	0.09027	0.00454	0.09546	0.11827	0.00598	0.09402
0.15000	0.07045	0.00528	0.14472	0.13353	0.01014	0.13986	0.17368	0.01327	0.13673
0.20000	0.09392	0.00939	0.19061	0.17554	0.01787	0.18213	0.22662	0.02328	0.17672
0.25000	0.11738	0.01468	0.23532	0.21631	0.02766	0.22234	0.27714	0.03588	0.21412
0.30000	0.14083	0.02113	0.27887	0.25582	0.03947	0.26053	0.32526	0.05094	0.24906
0.35000	0.16425	0.02876	0.32124	0.29409	0.05321	0.29679	0.37102	0.06834	0.28166
0.40000	0.18763	0.03755	0.36245	0.33112	0.06884	0.33116	0.41447	0.08798	0.31202
0.45000	0.21098	0.04752	0.40248	0.36691	0.08630	0.36370	0.45567	0.10973	0.34027
0.50000	0.23426	0.05865	0.44135	0.40147	0.10551	0.39449	0.49467	0.13349	0.36651
0.55000	0.25748	0.07094	0.47906	0.43480	0.12641	0.42359	0.53152	0.15915	0.39085
0.60000	0.28062	0.08440	0.51560	0.46691	0.14895	0.45105	0.56630	0.18659	0.41341
0.65000	0.30366	0.09900	0.55100	0.49781	0.17307	0.47693	0.59906	0.21573	0.43427
0.70000	0.32659	0.11476	0.58524	0.52750	0.19871	0.50129	0.62988	0.24645	0.45355
0.75000	0.34938	0.13166	0.61834	0.55601	0.22579	0.52421	0.65882	0.27867	0.47133
0.80000	0.37203	0.14969	0.65031	0.58333	0.25428	0.54572	0.68596	0.31229	0.48771
0.85000	0.39450	0.16886	0.68114	0.60949	0.28410	0.56590	0.71136	0.34722	0.50278
0.90000	0.41679	0.18914	0.71086	0.63450	0.31520	0.58480	0.73510	0.38338	0.51662
0.95000	0.43887	0.21053	0.73947	0.65838	0.34752	0.60248	0.75725	0.42069	0.52931
1.00000	0.46071	0.23302	0.76698	0.68115	0.38101	0.61899	0.77788	0.45907	0.54093
1.05000	0.48231	0.25660	0.79340	0.70282	0.41561	0.63439	0.79707	0.49844	0.55156

1.10000	0.50363	0.28124	0.81876	0.72342	0.45126	0.64874	0.81489	0.53874	0.56126
1.15000	0.52465	0.30695	0.84305	0.74297	0.48792	0.66208	0.83141	0.57990	0.57010
1.20000	0.54535	0.33370	0.86630	0.76150	0.52553	0.67447	0.84669	0.62185	0.57815
1.25000	0.56570	0.36148	0.88852	0.77903	0.56405	0.68595	0.86081	0.66454	0.58546
1.30000	0.58570	0.39026	0.90974	0.79558	0.60341	0.69659	0.87383	0.70790	0.59210
1.35000	0.60530	0.42004	0.92996	0.81119	0.64358	0.70642	0.88581	0.75190	0.59810
1.40000	0.62450	0.45078	0.94922	0.82589	0.68451	0.71549	0.89683	0.79646	0.60354
1.45000	0.64327	0.48248	0.96752	0.83970	0.72615	0.72385	0.90693	0.84156	0.60844
1.50000	0.66160	0.51510	0.98490	0.85266	0.76846	0.73154	0.91619	0.88713	0.61287
1.55000	0.67946	0.54863	1.00137	0.86479	0.81139	0.73861	0.92465	0.93315	0.61685
1.60000	0.69683	0.58303	1.01697	0.87614	0.85492	0.74508	0.93237	0.97958	0.62042
1.65000	0.71371	0.61830	1.03170	0.88672	0.89899	0.75101	0.93940	1.02637	0.62363
1.70000	0.73007	0.65439	1.04561	0.89658	0.94357	0.75643	0.94579	1.07350	0.62650
1.75000	0.74590	0.69129	1.05871	0.90574	0.98863	0.76137	0.95160	1.12094	0.62906
1.80000	0.76120	0.72897	1.07103	0.91424	1.03413	0.76587	0.95685	1.16865	0.63135
1.85000	0.77595	0.76740	1.08260	0.92212	1.08004	0.76996	0.96161	1.21661	0.63339
1.90000	0.79014	0.80655	1.09345	0.92939	1.12633	0.77367	0.96589	1.26480	0.63520
1.95000	0.80378	0.84640	1.10360	0.93611	1.17296	0.77704	0.96976	1.31319	0.63681
2.00000	0.81684	0.88691	1.11309	0.94229	1.21992	0.78008	0.97323	1.36177	0.63823
2.05000	0.82935	0.92807	1.12193	0.94797	1.26718	0.78282	0.97635	1.41050	0.63950
2.10000	0.84128	0.96983	1.13017	0.95317	1.31471	0.78529	0.97914	1.45939	0.64061
2.15000	0.85265	1.01218	1.13782	0.95794	1.36249	0.78751	0.98164	1.50841	0.64159
2.20000	0.86346	1.05508	1.14492	0.96229	1.41049	0.78951	0.98387	1.55755	0.64245
2.25000	0.87371	1.09851	1.15149	0.96625	1.45870	0.79130	0.98585	1.60679	0.64321
2.30000	0.88341	1.14244	1.15756	0.96986	1.50711	0.79289	0.98761	1.65613	0.64387
2.35000	0.89258	1.18684	1.16316	0.97313	1.55568	0.79432	0.98918	1.70555	0.64445
2.40000	0.90122	1.23168	1.16832	0.97610	1.60441	0.79559	0.99056	1.75504	0.64496
2.45000	0.90934	1.27695	1.17305	0.97878	1.65329	0.79671	0.99178	1.80460	0.64540
2.50000	0.91696	1.32261	1.17739	0.98119	1.70228	0.79772	0.99286	1.85422	0.64578
2.55000	0.92409	1.36863	1.18137	0.98337	1.75140	0.79860	0.99381	1.90388	0.64612
2.60000	0.93075	1.41500	1.18500	0.98532	1.80062	0.79938	0.99464	1.95360	0.64640
2.65000	0.93695	1.46169	1.18831	0.98707	1.84993	0.80007	0.99537	2.00335	0.64665
2.70000	0.94272	1.50869	1.19131	0.98863	1.89932	0.80068	0.99601	2.05313	0.64687
2.75000	0.94806	1.55596	1.19404	0.99003	1.94878	0.80122	0.99657	2.10294	0.64706
2.80000	0.95301	1.60348	1.19652	0.99127	1.99832	0.80168	0.99705	2.15278	0.64722
2.85000	0.95758	1.65125	1.19875	0.99237	2.04791	0.80209	0.99747	2.20265	0.64735
2.90000	0.96178	1.69923	1.20077	0.99335	2.09755	0.80245	0.99784	2.25253	0.64747
2.95000	0.96564	1.74742	1.20258	0.99421	2.14724	0.80276	0.99816	2.30243	0.64757
3.00000	0.96918	1.79579	1.20421	0.99497	2.19697	0.80303	0.99843	2.35235	0.64765
3.05000	0.97241	1.84433	1.20567	0.99565	2.24673	0.80327	0.99866	2.40227	0.64773

3.10000	0.97536	1.89302	1.20698	0.99624	2.29653	0.80347	0.99887	2.45221	0.64779
3.15000	0.97804	1.94186	1.20814	0.99675	2.34636	0.80364	0.99904	2.50216	0.64784
3.20000	0.98048	1.99082	1.20918	0.99720	2.39621	0.80379	0.99919	2.55211	0.64789
3.25000	0.98268	2.03990	1.21010	0.99760	2.44608	0.80392	0.99932	2.60208	0.64792
3.30000	0.98467	2.08908	1.21092	0.99794	2.49596	0.80404	0.99942	2.65205	0.64795
3.35000	0.98646	2.13836	1.21164	0.99824	2.54587	0.80413	0.99952	2.70202	0.64798
3.40000	0.98807	2.18772	1.21228	0.99850	2.59579	0.80421	0.99960	2.75200	0.64800
3.45000	0.98952	2.23716	1.21284	0.99872	2.64572	0.80428	0.99966	2.80198	0.64802
3.50000	0.99081	2.28667	1.21333	0.99891	2.69566	0.80434	0.99972	2.85196	0.64804
3.55000	0.99196	2.33624	1.21376	0.99908	2.74561	0.80439	0.99976	2.90195	0.64805
3.60000	0.99298	2.38586	1.21414	0.99922	2.79556	0.80444	0.99980	2.95194	0.64806
3.65000	0.99389	2.43554	1.21446	0.99934	2.84553	0.80447	0.99984	3.00193	0.64807
3.70000	0.99469	2.48525	1.21475	0.99945	2.89550	0.80450	0.99987	3.05192	0.64808
3.75000	0.99540	2.53500	1.21500	0.99953	2.94547	0.80453	0.99989	3.10192	0.64808
3.80000	0.99603	2.58479	1.21521	0.99961	2.99545	0.80455	0.99991	3.15191	0.64809
3.85000	0.99658	2.63460	1.21540	0.99967	3.04543	0.80457	0.99993	3.20191	0.64809
3.90000	0.99706	2.68445	1.21555	0.99973	3.09542	0.80458	0.99994	3.25190	0.64810
3.95000	0.99748	2.73431	1.21569	0.99977	3.14541	0.80459	0.99995	3.30190	0.64810
4.00000	0.99785	2.78419	1.21581	0.99981	3.19540	0.80460	0.99996	3.35190	0.64810
4.05000	0.99817	2.83409	1.21591	0.99984	3.24539	0.80461	0.99997	3.40190	0.64810
4.10000	0.99844	2.88401	1.21599	0.99987	3.29538	0.80462	0.99997	3.45190	0.64810
4.15000	0.99868	2.93394	1.21606	0.99989	3.34537	0.80463	0.99998	3.50189	0.64811
4.20000	0.99889	2.98388	1.21612	0.99991	3.39537	0.80463	0.99998	3.55189	0.64811
4.25000	0.99907	3.03382	1.21618	0.99993	3.44537	0.80463	0.99999	3.60189	0.64811
4.30000	0.99922	3.08378	1.21622	0.99994	3.49536	0.80464	0.99999	3.65189	0.64811
4.35000	0.99935	3.13375	1.21625	0.99995	3.54536	0.80464	0.99999	3.70189	0.64811
4.40000	0.99947	3.18372	1.21628	0.99996	3.59536	0.80464	0.99999	3.75189	0.64811
4.45000	0.99956	3.23369	1.21631	0.99997	3.64536	0.80464	0.99999	3.80189	0.64811
4.50000	0.99964	3.28367	1.21633	0.99998	3.69535	0.80465	1.00000	3.85189	0.64811
4.55000	0.99971	3.33366	1.21634	0.99998	3.74535	0.80465	1.00000	3.90189	0.64811
4.60000	0.99977	3.38364	1.21636	0.99999	3.79535	0.80465	1.00000	3.95189	0.64811
4.65000	0.99982	3.43363	1.21637	0.99999	3.84535	0.80465	1.00000	4.00189	0.64811
4.70000	0.99986	3.48362	1.21638	0.99999	3.89535	0.80465	1.00000	4.05189	0.64811
4.75000	0.99990	3.53362	1.21638	0.99999	3.94535	0.80465	1.00000	4.10189	0.64811
4.80000	0.99992	3.58361	1.21639	1.00000	3.99535	0.80465	1.00000	4.15189	0.64811
4.85000	0.99995	3.63361	1.21639	1.00000	4.04535	0.80465	1.00000	4.20189	0.64811
4.90000	0.99997	3.68361	1.21639	1.00000	4.09535	0.80465	1.00000	4.25189	0.64811
4.95000	1.00000	3.73361	1.21639	1.00000	4.14535	0.80465	1.00000	4.30189	0.64811
5.00000	1.00000	3.78361	1.21639	1.00000	4.19535	0.80465	1.00000	4.35189	0.64811

Table A.2. Local quantities τ_w, δ and δ_d from the similarity solutions.

β	**Grid 1** $\Delta\eta=0.05$	**Grid 2** $\Delta\eta=0.025$	**Grid 3** $\Delta\eta=0.0167$	**Grid 4** $\Delta\eta=0.00625$	**Grid 5** $\Delta\eta=0.005$	
0	0.469669	0.46965	0.469647	0.469645	0.469645	**τ_wall**
0.5	0.927713	0.927689	0.927684	0.927681	0.927681	
1	1.232628	1.232603	1.232596	1.232589	1.232588	
0	3.468768	3.444164	3.452582	3.463079	3.464335	**δ**
0.5	2.749057	2.724565	2.732994	2.743477	2.74473	
1	2.379731	2.354418	2.362762	2.373168	2.374414	
0	1.216392	1.216461	1.216474	1.216483	1.216483	**δ_d**
0.5	0.804649	0.804568	0.804553	0.804543	0.804542	
1	0.64811	0.647952	0.647923	0.647903	0.647902	

Appendix B: Manufactured Solution Constants

Constants employed for the manufactured solutions include L = 1 m, $\gamma = 1.4$, and R = 287.0 N·m/(kg·K). The constants for the supersonic Euler manufactured solution are given in Table B.1. Note that the ϕ constants all have the same dimensions as the primitive variable (listed in the first column), and the *a* constants are dimensionless.

Table B.1: Constants for the supersonic Euler manufactured solution.

Variable, ϕ	ϕ_0	ϕ_x	ϕ_y	ϕ_{xy}	$a_{\phi x}$	$a_{\phi y}$	$a_{\phi xy}$
ρ (kg/m^3)	1	0.15	-0.1	0	1	0.5	0
u (m/s)	800	50	-30	0	1.5	0.6	0
v (m/s)	800	-75	40	0	0.5	2/3	0
p (N/m^2)	1×10^5	0.2×10^5	0.5×10^5	0	2	1	0