AIAA JOURNAL
Vol. 30, No. 6, June 1992

Newton's Method Solver for the Axisymmetric Navier-Stokes Equations

Paul D. Orkwis* and D. Scott McRae†
North Carolina State University, Raleigh, North Carolina 27695

A new two-dimensional Newton's method Navier-Stokes solver is extended to the axisymmetric form of the equations. Roe's flux difference splitting is applied to the axisymmetric equations using Glaister's approach. A geometric conservation law correction term is added to insure satisfaction of the uniform stream reproduction criteria. Higher-order accuracy is attained through the use of Spekreijse's interpolation procedure with Van Albada's continuous limiter. Harten and Hyman's minimum eigenvalue correction is also included. Turbulent flow is simulated with the Baldwin-Lomax turbulence model, and transition is modeled using Guidos and Weinacht's blending polynomial. Results of a grid resolution study for a pointed supersonic projectile are compared with the experimental data of Dolling and Gray.

Introduction

IN recent years, advances in computational hardware have allowed researchers to re-examine previously undesirable schemes in the search for improvements to the accuracy, efficiency, and robustness of existing computational fluid dynamics algorithms. Schemes like the classical Newton's method were once thought to be impractical because of computer memory limitations. However, the recent generation of large-memory supercomputers, like the Cray 2 and Y-MP, finally allow the previously discarded memory intensive methods to be considered.

Researchers have used Newton's method to obtain solutions of the potential equation,[1] the Euler equations,[2-5] and the Navier-Stokes equations.[2,4,6-10] The scheme was recently applied to supersonic viscous flowfields by Orkwis and McRae,[6] who solved the two-dimensional laminar Navier-Stokes equations using Roe's flux difference splitting (FDS) and advanced sparse matrix inversion routines. The current research extends this method to supersonic viscous turbulent axisymmetric flowfields.

This paper begins by describing the governing equations to be solved, the general Newton's method scheme, and the spatial discretization. Glaister's approach[11] to the axisymmetric Roe's FDS, an axial flow freestream geometric conservation law (GCL) correction, and the changes needed for higher-order accuracy and the minimum eigenvalue correction are then explained. The details of the turbulence and transition models are also presented. Finally, a comparison of the results of the new method to the experimental data of Dolling and Gray[12] is included for code validation.

Governing Equations

The equations to be solved are the steady axisymmetric laminar or turbulent Navier-Stokes equations:

$$\frac{\partial yF(U)}{\partial x}+\frac{\partial yG(U)}{\partial y}=H(U) \tag{1}$$

where

$$U=\begin{bmatrix}\rho\\ \rho u\\ \rho v\\ e\end{bmatrix},\qquad F=\begin{bmatrix}\rho u\\ \rho u^2+p-\tau_{xx}\\ \rho uv-\tau_{xy}\\ (e+p)u-b_x\end{bmatrix}$$

$$G=\begin{bmatrix}\rho v\\ \rho uv-\tau_{xy}\\ \rho v^2+p-\tau_{yy}\\ (e+p)v-b_y\end{bmatrix},\qquad H=\begin{bmatrix}0\\ 0\\ p-\tau_{\theta\theta}\\ 0\end{bmatrix}$$

The terms τ_{ij} and b_i are defined in the usual manner, x and y are the axial and radial directions, respectively, ρ the density, u and v the velocity components, p the pressure, e the total energy, μ the viscosity, R_e the Reynolds number, and P_r the Prandtl number.

Equation (1) is transformed into generalized coordinates and put into chain rule conservation law form.[13] The resulting equation set is then discretized with finite differences and solved via Newton's method for systems of nonlinear equations.

Numerical Method

The basic solution procedure is known as Newton's method for systems of nonlinear equations. Systems of this type have the form

$$\mathfrak{F}(\bar{U})=0 \tag{2}$$

The general Newton's method is

$$\left(\frac{\partial\mathfrak{F}}{\partial\bar{U}}\right)^n\Delta^n\bar{U}=-\mathfrak{F}^n(\bar{U}) \tag{3}$$

A solution is obtained by forming $\mathfrak{F}(\bar{U})$ and the Jacobian $\partial\mathfrak{F}/\partial\bar{U}$ at the known nth iterate. The increment $\Delta^n\bar{U}$ is then found by inverting the Jacobian matrix. The value of $\bar{U}$ at the new iterate is given by

$$\bar{U}^{n+1}=\bar{U}^n+\Delta^n\bar{U}$$

In this application, $\bar{U}$ is a vector composed of the local conserved variable vector at each point in the discretized domain. If there are k points, $\bar{U}$ can be written as

$$\bar{U}=[\hat{U}_1,\hat{U}_2,\ldots,\hat{U}_k]^T$$

Received Feb. 4, 1991; presented as Paper 91-1554 at the AIAA 10th Computational Fluid Dynamics Conference, Honolulu, HI, June 24-27, 1991; revision received Sept. 25, 1991; accepted for publication Sept. 27, 1991.

*Research Assistant, Department of Mechanical and Aerospace Engineering, Box 7910; currently Assistant Professor of Aerospace Engineering and Engineering Mechanics, University of Cincinnati, Cincinnati, OH. Student Member AIAA.

†Associate Professor, Department of Mechanical and Aerospace Engineering, Box 7910. Member AIAA.

where

$$\hat{U}_i = [\rho, \rho u, \rho v, e]_i^T$$

For central difference discretizations, the matrix formed by the above Jacobian is block tridiagonal. In general, this matrix is very sparse. Inverting the Jacobian matrix can be accomplished by numerous sparse matrix solvers (i.e., Sparsepak, Yale sparse matrix package), as well as the classic banded matrix solvers. In this work the Boeing RSLIB[5] package with nested dissection was used.

It should be noted that the proper inversion of the Jacobian matrix does not guarantee convergence. If the initial guess is not close enough to the final solution, the solver may diverge. This usually occurs because of spurious oscillations. These oscillations lead Newton's method away from the proper convergence direction. To help avoid this difficulty a timelike $1/\Delta t$ term is added to the diagonal of the Jacobian matrix. As the residual gets smaller, the magnitude of Δt is increased, which in the limit as $\Delta t \to \infty$, returns the modified solver back into the full Newton's method.

Discretization of $\mathfrak{F}(\bar{U})$

Ideally, the solution of the steady-state Navier-Stokes equations should not be affected by the time terms in a time-relaxation scheme, or by the form of the Jacobian in Newton and Newton-like schemes. The quality of the solution should be based entirely on the accuracy of the spatial discretizations of the fluxes. For the inviscid fluxes of the Navier-Stokes equations, recent work by Van Leer et al.[14] has shown that Roe's FDS produces superior results when compared to the flux vector splittings of Van Leer and Steger and Warming. For this reason, Roe's FDS was chosen to discretize the governing equations.

Since the exact Jacobian was used in this work, an analytical expression for the spatial discretization was required. Roe's scheme, as developed by Vatsa et al.[15] produces the desired expressions. This discretization was differentiated by using the symbolic manipulation expert system MACSYMA to develop the exact Jacobian matrix.

Axisymmetric Scheme

The addition of the source term and the y multipliers of the fluxes prevent a straightforward application of Roe's FDS to the axisymmetric equations. The problem is handled in this work by evaluating the source term at the point of discretization and using Glaister's[11] approach for the y multipliers.

Glaister proposed the change of variables

$$\hat{U} = yU$$

$\hat{U}$ may then be written as

$$\hat{U} = \begin{bmatrix} yp \\ y\rho u \\ y\rho v \\ ye \end{bmatrix} = \begin{bmatrix} R \\ Ru \\ Rv \\ E \end{bmatrix}$$

The homogeneous property of the inviscid fluxes allows the following expression to be written

$$yF(U) = F(yU) = F(\hat{U})$$

$$yG(U) = G(yU) = G(\hat{U})$$

To complete the description of the transformed variables, one must also consider the energy and the pressure. Recall that

$$e = \rho e_i + \tfrac{1}{2}\rho(u^2 + v^2)$$

so that

$$E = ye = y\rho e_i + \tfrac{1}{2}y\rho(u^2 + v^2)$$

$$= Re_i + \tfrac{1}{2}R(u^2 + v^2)$$

and since

$$p = (\gamma = 1)\rho e_i$$

a transformed pressure is also defined

$$P = yp = (\gamma - 1)y\rho e_i = (\gamma - 1)Re_i$$

By including the y multipliers, the inviscid fluxes may now be written as

$$F(\hat{U}) = \begin{bmatrix} Ru \\ Ru^2 + P \\ Ruv \\ (E + P)u \end{bmatrix}, \qquad G(\hat{U}) = \begin{bmatrix} Rv \\ Ruv \\ Rv^2 + P \\ (E + P)v \end{bmatrix}$$

This gives the inviscid part of the left-hand side of the axisymmetric equations the same form as the inviscid part of the two-dimensional equations, allowing Roe's scheme to be applied just as for a two-dimensional case. However, a general Roe average of Q now includes y as follows:

$$\tilde{Q} = \frac{Q_L + Q_R\sqrt{(R_R/R_L)}}{1 + \sqrt{(R_R/R_L)}} = \frac{Q_L + Q_R\sqrt{(yR\rho R)/(yL\rho L)}}{1 + \sqrt{(yR\rho R)/(yL\rho L)}}$$

where R and L are the standard right and left Riemann problem interface locations, respectively.

Geometric Conservation Law Correction

A viable discretization of the governing equations must satisfy Hindman's[13] uniform stream reproduction (USR) test. That is, the method must reproduce a freestream when supplied with freestream initial conditions and the associated boundary conditions.

The two-dimensional form of the Roe's FDS discretization satisfies this criterion exactly. In this case, a freestream oriented at any angle to the coordinate system will produce constant conserved variables and fluxes. However, a freestream is not satisfied for the axisymmetric discretization.

Fortunately, the USR test can be satisfied if a correction is added to the scheme. This correction is similar to the GCL term of Gielda and McRae[16] and can be determined by applying a freestream to the discretized equations. Simplification of these equations results in cancellation of the terms that satisfy the GCL. The remaining problem terms are then used to form the GCL correction. These terms are evaluated at the center point and then subtracted from the discretization. When these terms are included in the discretization, the axisymmetric routine satisfies the USR test exactly.

Primitive Variable Interpolation Procedure

The order of accuracy of the previous discretization depends on how U_L and U_R or $\hat{U}_L$ and $\hat{U}_R$ are evaluated. The simplest evaluation uses the variables at the nodes nearest the interface. This produces a first-order method. To obtain higher-order accuracy, one needs to use more complicated expressions based on interpolations of either the conserved or primitive variables. In this work, superior results were obtained from interpolations of the primitive variable vector; $q = [\rho, u, v, p]^T$. Using this approach, Roe's FDS can be considered a function of q_L and q_R.

In this work, Spekreijse's interpolation[17] with Van Albada et al.'s[18] limiter is used. The primitive variables at the right and left interfaces are given by

$$q_{L_{i+\frac{1}{2}}} = q_i + \frac{1}{2}\frac{(2a^2+\delta^2)b + (b^2+2\delta^2)a}{2a^2 - ab + 2b^2 + 3\delta^2}$$

$$q_{R_{i-\frac{1}{2}}} = q_i - \frac{1}{2}\frac{(2b^2+\delta^2)a + (a^2+2\delta^2)b}{2b^2 - ab + 2a^2 + 3\delta^2}$$

where

$$a = q_{i+1} - q_i, \qquad b = q_i - q_{i-1}$$

and δ is a small number taken to be 10^{-5}.

Note that for the axisymmetric equations it is still the primitive variables that are interpolated; therefore, the above equations are not functions of the radial direction. Hence, no change is made in the interpolation procedure for the axisymmetric equation set. The discretization is then completed by using the y values of the nodes nearest the respective interfaces for y_R and y_L.

Entropy Adjustment

A disadvantage of Roe's FDS are the problems encountered in high Mach number flows near stagnation points when the eigenvalues approach zero. Zero eigenvalues effectively remove the natural dissipation inherent in upwind discretizations. A modification introduced by Harten and Hyman[19] provides for a threshold value of the eigenvalues, so that their magnitudes are not permitted to fall below some chosen ϵ. The eigenvalues can then be written as

$$|\lambda_i| \begin{cases} |\lambda_i|, & |\lambda_i| \geq \epsilon \\ (1/2\epsilon)[\lambda_i^2 + \epsilon^2], & |\lambda_i| \geq \epsilon \end{cases}$$

This adjustment increases[20] the dissipation of the scheme when the solution is near a zero eigenvalue. The value of ϵ should be chosen with care because too large a value will result in increased dissipation in the viscous layer, thereby negating one of the primary advantages of Roe's scheme over the flux vector splitting schemes. An alternative that avoids this difficulty is to alter the eigenvalues only in the regions near the stagnation point and leaving eigenvalues in the other regions unchanged.

Turbulence Model

The axisymmetric flowfields to be considered in this work are turbulent in nature. To demonstrate the applicability of the new scheme to flows of this type, the Baldwin-Lomax[21] turbulence model was included. It was chosen because it is a simple algebraic approach that can be easily incorporated into Newton's method. It has also been well used and examined by many previous researchers.

Equation (1) changes only through redefinition of the viscosity and Prandtl number. The modifications are

$$\mu = \mu_l + \mu_t$$

$$\frac{\mu}{P_r} = \left(\frac{\mu}{P_r}\right)_l + \left(\frac{\mu}{P_r}\right)_t$$

The laminar viscosity is given by Sutherland's formula, whereas the turbulent value is found from a two-layer algebraic eddy-viscosity model.

Transition Model

A transition model is needed to activate the turbulent viscosity when transiting from the laminar region to the turbulent region. Baldwin and Lomax suggest a step function turned on when the maximum proposed turbulent viscosity exceeds some constant. This is not well suited to Newton's method procedures because it is discontinuous. Instead, the method used by Guidos and Weinacht[22] was used. This approach follows that of Dhawan and Narashima[23] but uses a simple polynomial expression rather than an exponential. To account for transition, the viscosity is written as

$$\mu = \mu_l + \kappa_t \mu_t$$

κ_t is given by

$$\kappa_t = 10\left[\frac{x-x_1}{x_2-x_1}\right]^3 - 15\left[\frac{x-x_1}{x_2-x_1}\right]^4 + 6\left[\frac{x-x_1}{x_2-x_1}\right]^5$$

where x_1 and x_2 are the x locations at the beginning and end of transition, respectively. The transition model therefore requires inputs of these values. Fortunately, the cases considered in this work are compared to experimental data that provides the transition locations.

Results

In this study, the axisymmetric Newton's method is verified by computing a Mach number 2.95, Reynolds number $3.084 \cdot 10^6$ ogive-tangent cylinder. The ogive is 3 diam long and the cylinder 7 diam long. This flowfield is a good test case because it contains strong curved shock waves, expansion waves, and turbulent flow. The test case was computed using several different grids to gain an understanding of how grid changes affect the results. The solutions obtained on these grids are compared to the experimental data of Dolling and Gray.[12]

Three grids were used for this study, the 176×40 grid shown in Fig. 1, and 176×60 and 278×40 grids with similar outside boundaries. The upper boundary was determined by the cubic

$$y_{\max}(x) = -7.1272 \cdot 10^{-3}x^3 + 4.6716 \cdot 10^{-2}x^2 + 4.8508 \cdot 10^{-1} + 0.2$$

Spacing in the axial direction was governed by the equations

$$x(i) = \begin{cases} 2.1875 \cdot 10^{-2}(i-8) & x \leq 0.7 \\ 1.7846 \cdot 10^{-4}i^2 + 7.7768 \cdot 10^{-3}i + 0.1034 & x > 0.7 \end{cases}$$

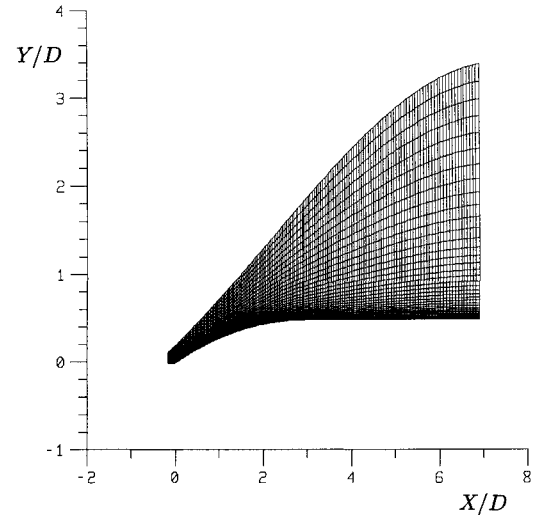

Fig. 1 Computational grid for pointed projectile.

for the 176 axial point grids, and

$$x(i) = \begin{cases} 1.0938 \cdot 10^{-2}(i-8) & x \le 0.35 \\ 7.1145 \cdot 10^{-5} i^2 & \\ \quad + 5.3171 \cdot 10^{-3} i & x > 0.35 \\ \quad + 2.3486 \cdot 10^{-2} & \end{cases}$$

for the 278 axial point grid. Clustering in the y direction was determined from the equation

$$y(j) = y_{\min} + (y_{\max} - y_{\min})\left(1 - s + \frac{2s}{1 + [(s+1)/(s-1)]^{(ny-j)/(ny-1)}}\right)$$

Where $y_{\min}$ and $y_{\max}$ represent the minimum and maximum y values in the column, respectively, ny is the total number of points in the y direction, and s is the clustering parameter; s was constant in both the laminar and turbulent regions and varied through the transition region as follows:

$$s(x) \begin{cases} 1.0004 & x \le 0.85 \\ 1.0004 + 0.01477(x - 0.85) & 0.85 < x \le 1.5 \\ 1.01 & x > 1.5 \end{cases}$$

The solutions obtained on the above grids resulted from successive solutions on smaller 40×40 and 40×60 subgrids. By taking advantage of the supersonic flowfield characteristics, a converged solution is obtained on the first subgrid in the nose region, and then a solution is obtained on the next successive subgrid using the solution from the first subgrid as an initial condition. Solutions are then obtained successively on the remaining subgrids. Five subgrids were used for the 176 axial point meshes and eight subgrids for the 278 axial point mesh. With the exception of the first subgrid, the first column of every subgrid was the 35th column of the previous subgrid.

Slug flow initial conditions (freestream everywhere except at the surface where no-slip is applied) were used on the first subgrid. All other subgrids used the column 3 solution from the preceding subgrid as the initial condition at every column. In general, solutions that were started from slug flow initial conditions required many more iterations to converge than solutions that were started from smooth initial conditions. However, slug flow was used as an initial condition in the nose region to illustrate the effectiveness of the diagonal term modification.

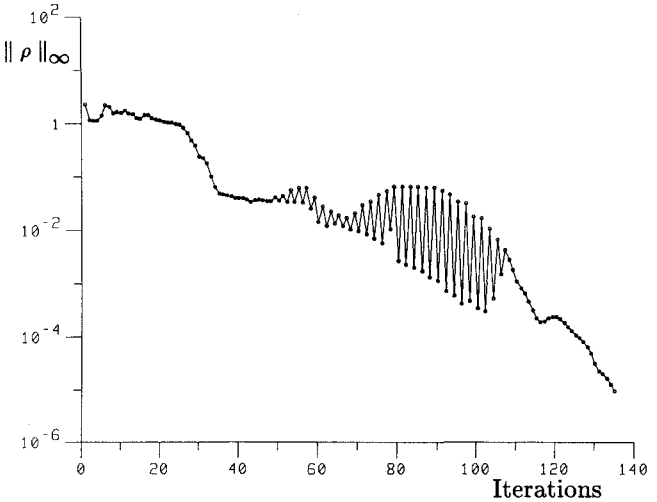

Fig. 2 Density residual for grid 1, 176 × 40 case.

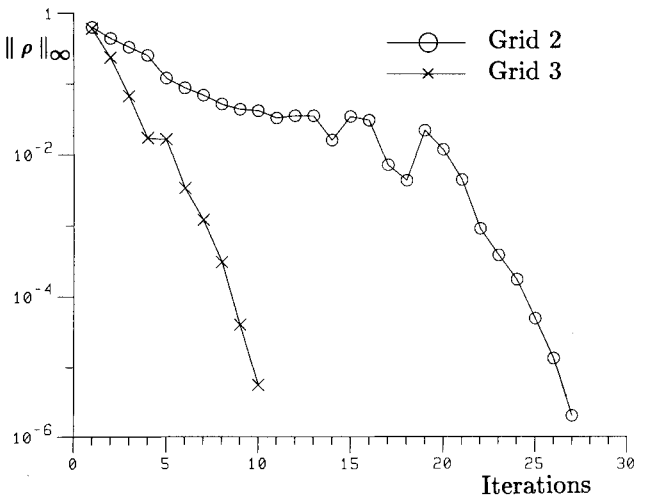

Fig. 3 Density residuals for grids 2 and 3, 176 × 40 case.

The boundary conditions that were used for all three cases were a constant freestream on the inflow and upper boundaries, a no-slip adiabatic wall at the surface of the projectile, extrapolation along the centerline, and outflow integration.

It should be noted that the equations used in the nose region differ slightly from those used elsewhere. Harten and Hyman's[19] minimum eigenvalue correction was used in this region to provide extra numerical dissipation, which was needed because of the decreased grid spacing and the slug flow initial condition. The solution contours and cross plots do not indicate any problems with inconsistent equation sets, which implies that this application of the minimum eigenvalue correction was valid.

Figure 2 shows the density residual for subgrid 1 of the 176×40 case. Note the large oscillations in the subgrid 1 residual. Plots of the maximum residual location show that these oscillations occurred during boundary-layer formation. It is likely that the sign changes of the equation set eigenvalues caused this abnormality. The new method converged in 135 iterations on subgrid 1.

Figure 3 shows the convergence histories along the ogive on subgrids 2 and 3. They indicate that the scheme converged much more quickly aft of the nose region and that near quadratic convergence was achieved on both subgrids. The method required 27 iterations to converge on subgrid 2 and 10 iterations to converge on subgrid 3.

The flow transitioned from laminar to turbulent on subgrid 3 after which the boundary layer became much larger. It should be noted that the slug flow initial condition used on subgrid 1 caused much more convergence difficulty for Newton's method than the transition to turbulent flow. This was because the transition region was modeled by a smooth function and the slug flow initial conditions used on the first subgrid are discontinuous.

Figure 4 shows the density residuals obtained on subgrids 4 and 5. For this case, the formation of the boundary layer resulted in linear convergence in both residuals. It is likely that the reduction in axial direction grid clustering in this region was the cause.

The convergence patterns for the first two subgrids of the 176×60 case are shown in Fig. 5. The addition of normal direction points required more iterations for the method to converge compared to the 176×40 case. This is most evident in the subgrid 2 solution, which required nearly four times the number of iterations to converge as the corresponding coarse

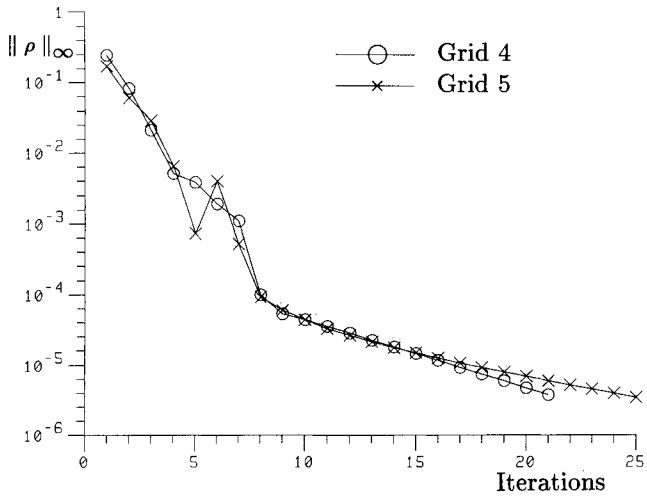

Fig. 4 Density residuals for grids 4 and 5, 176 × 40 case.

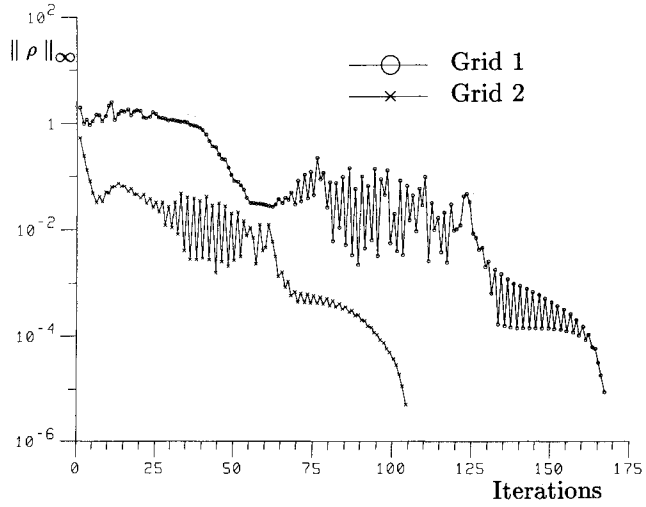

Fig. 5 Density residuals for grids 1 and 2, 176 × 60 case.

grid. In this case subgrid 1 converged in 168 iterations and subgrid 2 in 104 iterations. Oscillations were once again evident in the residual as the scheme formed the boundary layer. The general convergence patterns for the 176 point meshes were the same. Convergence was achieved more quickly for the subgrids farther along the ogive up until the cylinder was reached. In the subgrids aft of the ogive-cylinder junction, a region of linear convergence occurred while the scheme iterated in the boundary layer. Figure 6 shows the residuals for subgrids 3, 4, and 5, which took 12, 15, and 24 iterations to converge, respectively.

Solutions on the 278 × 40 grid produced similar convergence patterns. Figure 7 shows the density residuals for the first two subgrids of this case. Because of reductions in the axial spacing, the first subgrid required 217 iterations to converge. Oscillations were again found once the maximum residual location moved to the boundary layer. It should again be noted that use of the slug flow initial condition required many more iterations for convergence compared to use of smooth initial conditions. In general, the inviscid portion of the solution developed quickly, whereas the viscous boundary layer required many additional iterations to be resolved. A closer initial condition would have probably improved the convergence of this subgrid, however, as before, slug flow initial conditions were used to illustrate the capabilities of the code.

Subgrid 2 required a relatively long 77 iterations to converge. The large number of iterations may have been the result of strong expansion and shock wave gradients aft of the nose region, as this subgrid was farther upstream than the second subgrid in the 176 axial point cases. This occurred in spite of the improved initial condition provided by the solution from the first subgrid. However, oscillations were not present in the residual as the boundary layer was formed and quadratic convergence was obtained.

Figures 8 and 9 show the residuals obtained with the remaining subgrids. Convergence occurred more quickly for the subgrids farther down the ogive up until the ogive-cylinder junction was reached. Subgrid 3 converged in 33 iterations, subgrid 4 in 14 iterations, and subgrid 5 in 10 iterations. The residuals from the three subgrids along the cylinder, shown in Fig. 9, all exhibit a region of linear convergence. This linear region was once again found during the formation of the boundary layer and was probably due to the reduction in axial direction grid clustering in this region. Subgrids 6 and 7 both required 17 iterations, and subgrid 8 required 16 iterations to converge.

Additional confirmation that the increase in grid point spacing led to linear convergence can be found by comparing Figs. 4, 6, and 9. Note that the value of the residual at which linear convergence begins is directly related to how far aft of the ogive-cylinder junction the particular grid lies. Since grid point spacing increases with x, the linear convergence region and the local grid spacing are directly related. This can also be shown to be true for spacing in the y direction, as the residual value at which linear convergence begins was lower for the 176 × 60 grid than the 176 × 40 grid. This indicates that additional grid clustering might remove the region of linear convergence.

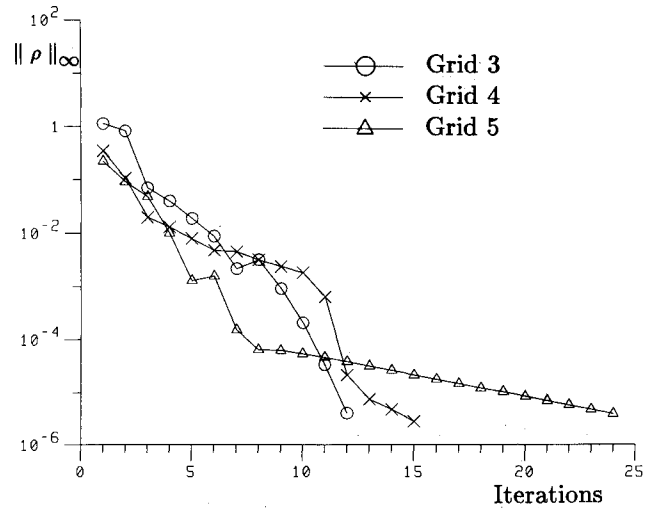

Fig. 6 Density residuals for grids 3, 4, and 5, 176 × 60 case.

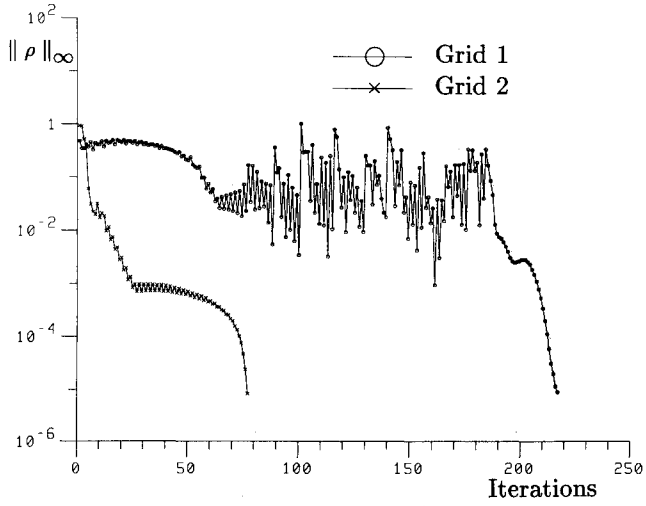

Fig. 7 Density residuals for grids 1 and 2, 278 × 40 case.

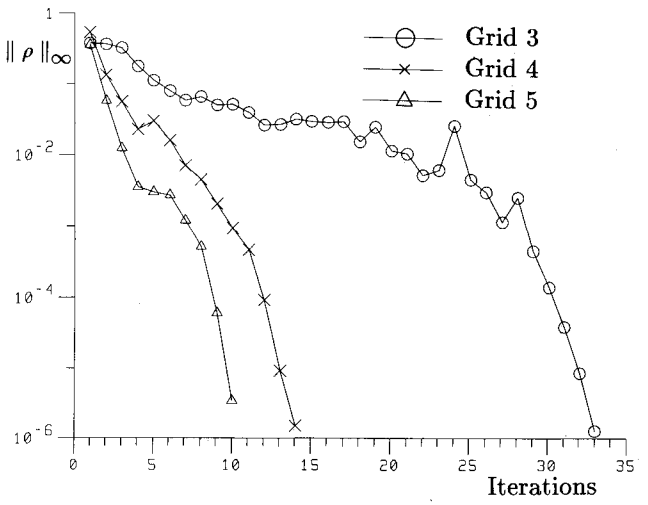

Fig. 8 Density residuals for grids 3, 4, and 5, 278 × 40 case.

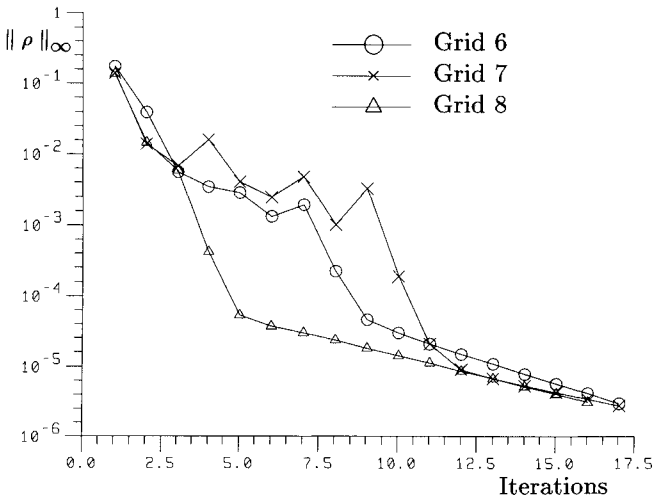

Fig. 9 Density residuals for grids 6, 7, and 8, 278 × 40 case.

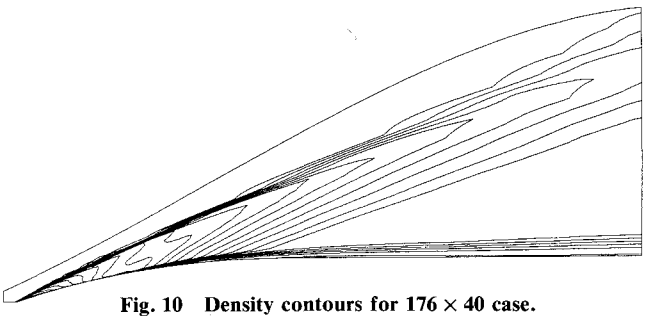

Fig. 10 Density contours for 176 × 40 case.

It should be noted that the first priority of this research was to determine if Newton's method could be applied to the axisymmetric governing equations. Improvements to the matrix inversion routines and implementation of convergence acceleration ideas have not yet been attempted so that the simplest practical form of Newton's method could be used. However, analysis of the current matrix inversion routines by personnel at the North Carolina Supercomputer Center showed that simple changes in the code could produce a 20% savings in computation time. Significant additional savings might also be achieved by using Jacobian freezing and mesh sequencing techniques. These ideas will be implemented in future work.

Computation time was under 30 s per iteration for the 40 × 40 grids and under 50 s for the 40 × 60 cases on the Cray Y-MP. The overall computation time was 1.64 h for the 176 × 40 grid, 3.99 h for the 176 × 60 grid, and 2.04 h for the 278 × 40 grid. This corresponds to 0.838, 1.360, and 0.984 s per grid point for each grid respectively. It is apparent that increasing the dimensions of the grid severely increases convergence times. The 176 × 60 and the 278 × 40 grids required more time per grid point because more overhead was needed due to larger grid dimensions. The computation times for each individual grid are found in Tables 1 and 2. These tables give the CPU times and the fraction of system time to total time. They illustrate the amount of input/output overhead required by the RSLIB matrix inversion routines. Typically, this overhead was between 16 and 17% of the total computation time for the 40 × 40 grids and 24 to 25% for the 40 × 60 grids, but varied with system load and number of iterations.

Table 1 176 × 40 and 176 × 60 subgrid computation times and system to total time fraction

	176 × 40 grid		176 × 60 grid	
Subgrid	CPU	System/total	CPU	System/total
1	3689	0.17	7801	0.31
2	713	0.15	4429	0.25
3	269	0.16	500	0.24
4	577	0.18	620	0.23
5	653	0.14	1010	0.24

Table 2 278 × 40 subgrid computation times and system to total time fraction

Subgrid	CPU	System/total
1	5892	0.16
2	2074	0.16
3	937	0.21
4	378	0.16
5	273	0.17
6	445	0.14
7	9478	0.20
8	463	0.22

The following paragraphs describe the solutions found for the test cases. Figure 10 shows the density contours for the 176 × 40 grid solutions. Results for the 176 × 60 and 278 × 40 grid solutions were virtually identical. Several prominent solution features are visible. A strong shock wave forms from the pointed nose of the projectile, which is curved by the expansion waves emanating from the ogive surface. The transition to turbulent flow is made visible along the ogive surface by a compression wave and an increase in boundary-layer thickness.

Figure 11 shows the $x = 0.35$ boundary layer, and Fig. 12 the $x = 3.07$ shock region comparisons of the density for all three

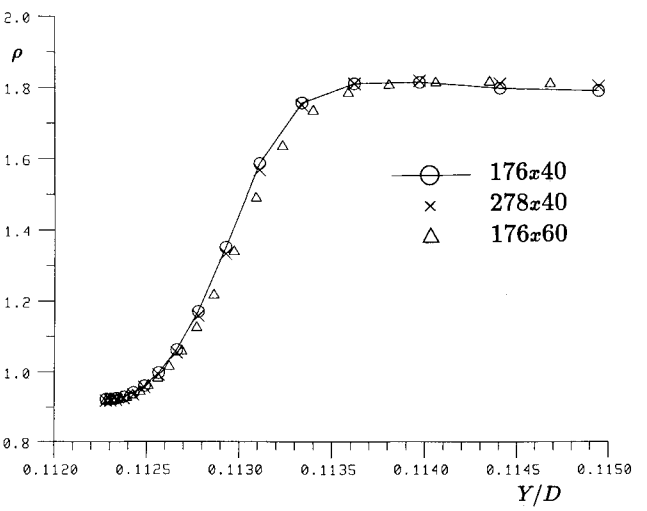

Fig. 11 Boundary-layer density comparison at $x = 0.35$.

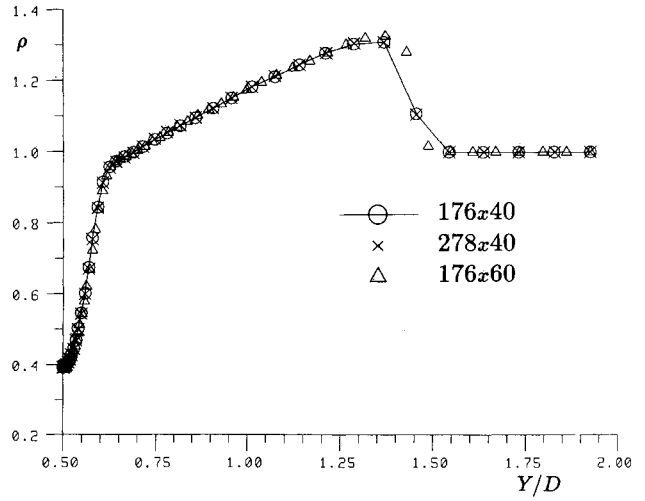

Fig. 12 Inviscid region density comparision at $x = 3.07$.

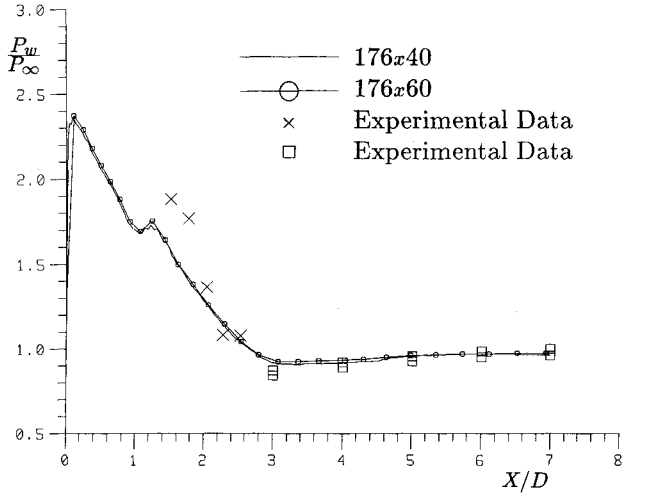

Fig. 13 Surface pressure comparisons for 176×40 and 176×60 grids vs Dolling and Gray's[12] experimental data.

solutions. The results obtained with the 40×40 grids are virtually identical. This indicates that axial clustering had little effect on the solution. On the other hand, improvements to the normal direction clustering did produce small changes. A slightly thicker boundary layer and a sharper shock wave resulted when 60 points were used in the normal direction. The solution improvements indicate that it may still be possible to improve accuracy by using more normal direction points.

To further support this claim, comparisons between the computational and experimental surface pressure data were made. Figure 13 shows the comparisons to the experiment for the 176×40 and 176×60 cases. The results show that increases in the normal direction points moved the computed solution closer to the experimental data. The results of the comparison for the 176×40 and 278×40 cases, shown in Fig. 14, illustrate that additional axial clustering produced virtually no improvement in the results. This again indicates that more axial direction points would not be useful (except possibly in the nose region), but additional normal direction points might produce more accurate solutions. However, the degree of improvement was not thought to be great enough to warrant another expensive fine grid calculation.

The overall agreement between the experimental and numerical pressures was good everywhere except in the region just aft of transition. Not shown in Figs. 13 and 14 is the

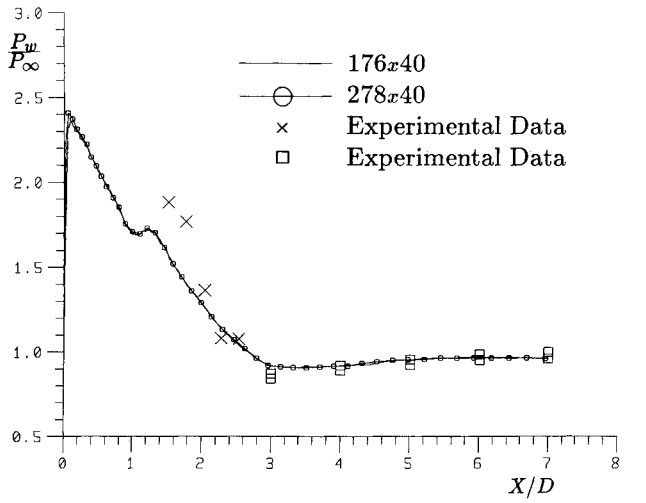

Fig. 14 Surface pressure comparisons for 176×40 and 278×40 grids vs Dolling and Gray's[12] experimental data.

excellent agreement of the surface pressure with that of the conical nose approximation. Possible sources of error that may explain why the experimental data was greater than the computed results in the transition region are Dolling and Gray's claim that this was a difficult location to place pressure taps, the need for more grid resolution, and the choice of turbulence/transition model. In this work, the Baldwin-Lomax turbulence model was used since it has been used quite often by previous researchers. The transition location is a required input for Guidos and Weinacht's transition model, as such the value used from Dolling's observations may be questionable. However, no study was performed to confirm that better results could be obtained by adjusting these parameters.

The results of the three cases served to verify the capabilities of the new method for computing axisymmetric flowfields. It has been shown that Newton's method is capable of computing high-speed viscous projectile flowfields with strong curved shock waves and turbulent flow. These results also verify the use of Glaister's approach to Roe's FDS discretization for the axisymmetric equations.

Conclusions

This research extended the development of the Newton's method solver to supersonic turbulent axisymmetric flowfields. Glaister's approach to applying Roe's flux difference splitting was found to result in an accurate discretization of the axisymmetric equation set. Results obtained with Newton's method applied to this discretization compare quite well with the available experimental data. The results indicate that the inclusion of a turbulence model does not seriously degrade the convergence performance of the new method. However, modifications to the basic Newton's method were found to be necessary to correct for poor initial conditions and complex solution gradients. It was also found that the scheme could degrade to linear convergence if inadequate grid resolution was used. The results show that Newton's method can compute accurately high speed viscous axisymmetric flowfields. However, additional research needs to be performed to transform the basic Newton's method into a competitive scheme.

Acknowledgments

This research was supported by the U.S. Army Research Office, Grant DAAL03-86-0039. The authors wish to thank the personnel at the U.S. Army Ballistic Research Laboratory and at the North Carolina Supercomputer Center for use of their computational resources.

References

[1]Bender, E. E., and Khosla, P. K., "Application of Sparse Matrix Solvers and Newton's Method to Fluid Flow Problems," *Proceedings of the AIAA/ASME/SIAM/APS 1st National Fluid Dynamics Congress,* AIAA, Washington, DC, 1988, pp. 402–408; also AIAA Paper 88-3700, July 1988.

[2]Liou, M. S., and Van Leer, B., "Choice of Implicit and Explicit Operators for the Upwind Differencing Method," AIAA Paper 88-0624, Jan. 1988.

[3]Giles, M., Drela, M., and Thompkins, W. T., "Newton Solution of Direct and Inverse Transonic Euler Equations," *Proceedings of the AIAA 7th Computational Fliud Dynamics Conference,* AIAA, New York, 1985, pp 394–402; also AIAA Paper 85-1530, July 1985.

[4]Hafez, M., Palaniswamy, S., and Mariani, P., "Calculations of Transonic Flows with Shocks Using Newton's Method and Direct Solver, Part II," AIAA Paper 88-0226, Jan. 1988.

[5]Wigton, L. B., "Application of MACSYMA and Sparse Matrix Technology to Multi-Element Airfoil Calculations," *Proceedings of the AIAA 8th Computational Fluid Dynamics Conference,* June 1987, pp. 444–457; also AIAA Paper 87-1142, June 1987.

[6]Orkwis, P. D. and McRae, D. S., "A Newton's Method Solver for the Navier-Stokes Equations," AIAA Paper 90-1524, June 1990; also "A Newton's Method Solver for High-Speed Viscous Separated Flowfields," *AIAA Journal,* Vol. 30, No. 1, 1992, pp. 78–85.

[7]Venkatakrishnan, V., "Newton Solution of Inviscid and Viscous Problems," AIAA Paper 88-0413, Jan. 1988; also, *AIAA Journal,* Vol. 27, No. 7, 1989, pp. 885–891.

[8]Van Dam, C. P., Hafez, M., and Ahmad, J., "Calculation of Viscous Flows with Separation Using Newton's Method and a Direct Solver," AIAA Paper 88-0412, Jan. 1988; also *AIAA Journal,* Vol. 28, No. 5, 1990, pp. 937–939.

[9]Bender, E. E., and Khosla, P. K., "Solution of the Two-Dimensional Navier-Stokes Equations Using Sparse Matrix Solvers," AIAA Paper 87-0603, Jan. 1987.

[10]Venkatakrishnan, V., "Preconditioned Conjugate Gradient Methods for the Compressible Navier-Stokes Equations," AIAA Paper 90-0586, Jan. 1990; also *AIAA Journal,* Vol. 29, No. 7, 1991, pp. 1092–1100.

[11]Glaister, P., "Flux Difference Splitting for the Euler Equations With Axial Symmetry," *Journal of Engineering Mathematics,* Vol. 22, No. 2, 1988, pp. 107–121.

[12]Dolling, D. S., and Gray, W. K., "Experimental Study of Supersonic Turbulent Flow on a Blunted Axisymmetric Body," *AIAA Journal,* Vol. 24, No. 5, 1986, pp. 793–799.

[13]Hindman, G. H., "Geometrically Induced Errors and Their Relationship to the Form of the Governing Equations and the Treatment of Generalized Mappings," AIAA Paper 81-1008, June, 1981.

[14]Van Leer, B. Thomas, J. L., Roe, P. L., and Newsome, R. W., "A Comparison of Numerical Flux Formulas for the Euler and Navier-Stokes Equations," *Proceedings of the AIAA 8th Computational Fluid Dynamics Conference,* AIAA, New York, 1987, pp. 36–41; also AIAA Paper 87-1104, June 1987.

[15]Vatsa, V. N., Thomas, J. L., and Wedan, B. W., "Navier-Stokes Computations of Prolate Spheroids at Angle of Attack," *Proceedings of the AIAA Atomospheric Flight Mechanics Conference,* AIAA, New York, 1987; also AIAA Paper 87-2627, Aug. 1987. See also *Journal of Aircraft,* Vol. 26, No. 11, 1989, pp. 986–993.

[16]Gielda, T. P., and McRae, D. S., "An Accurate, Stable, Explicit, Parabolized Navier-Stokes Solver for High Speed Flows," AIAA Paper 86-1116, May 1986.

[17]Spekreijse, S. P., "Multigrid Solution of the Steady Euler Equations," Ph.D. Dissertation, Centrum voor Wiskunde en Informatica, Amsterdam, 1987.

[18]Van Albada, G. D., Van Leer, B., and Robertrs, W. W., Jr., "A Comparative Study of Computational Methods in Cosmic Gas Dynamics," *Astronomy and Astrophysics,* Vol. 115, No. 2, 1982, pp. 263–269.

[19]Harten, A., and Hyman, J. M., "Self Adjusting Grid Methods for One-Dimensional Hyperbolic Conservation Laws," *Journal of Computational Physics,* Vol. 50, 1983, pp. 235–269.

[20]von Lavante, E., "Accuracy of Upwind Schemes Applied to the Navier-Stokes Equations," *AIAA Journal,* Vol. 28, No. 7, 1990, pp. 1312–1314.

[21]Baldwin, B. S., and Lomax, H., "Thin Layer Approximation and Algebraic Model for Separated Turbulent Flows," AIAA Paper 78-257, Jan. 1978.

[22]Guidos, B., and Weinacht, P., "Comparison of Navier-Stokes Computation and Experiment for Pointed, Spherical and Flat Tipped Shell at Mach 2.95," AIAA Paper 90-0587, Jan. 1990.

[23]Dhawan, S., and Narashima, R., "Some Properties of Boundary Layer Flow During the Transition from Laminar to Turbulent Motion," *Journal of Fluid Mechanics,* Vol. 3, Pt. 4, 1958, pp. 418–436.