AIAA JOURNAL
Vol. 33, No. 7, July 1995

Variants of Biconjugate Gradient Method for Compressible Navier–Stokes Solver

Herng Lin,* D. Y. Yang,† and Ching-Chang Chieng‡
National Tsing Hua University, Hsinchu, Taiwan, Republic of China

The variants of a biconjugate gradient method with a preconditioner of incomplete lower upper factorization are implemented for solving full Navier–Stokes equations with a k–ϵ two-equation model of turbulence and for resolving the problem of slow convergence, if the widely used approximate factorization method is employed. The conjugate gradient squared biconjugate gradient stable and the transpose of free quasiminimal residual algorithm are the selected approaches to speed up the convergence rate and remove the irregular convergent behavior. The Reynolds averaged Navier–Stokes equations are discretized with an implicit total variation diminishing algorithm. The superiority of the new schemes is demonstrated by the comparisons of the convergence rate with the three variant biconjugate gradient methods and by the approximate factorization method for the computations of the turbulent, transonic, separated flow over an axisymmetric bump, and the projectile of a flat base. The results show that fast convergence rate can be achieved by the preconditioned variants of the biconjugate gradient methods, and the residuals can be further reduced as the iterations are continued. This implies that the computation of the turbulent flow with a two-equation model of turbulence can be a practical possibility in a reasonable computer time and with very good accuracy.

Nomenclature

$\hat{E}, \hat{F}, \hat{H}$	= flux vectors in transformed coordinates
J	= Jacobian
k	= turbulent kinetic energy
L	= length of the body
M	= Mach number
$\hat{M}, \hat{N}$	= viscous flux vectors in transformed coordinates
Q	= vector of dependent variables
t	= time
x, y	= axial and radial coordinate
y_n	= normal distance from the solid wall
ϵ	= turbulent dissipation rate
ξ, η	= transformed coordinates

Subscripts

$i \pm \frac{1}{2}, j \pm \frac{1}{2}$	= located at the interface of cell
i, j	= located on centroid of cell
ℓ	= laminar
t	= turbulent
∞	= freestream value

Superscripts

n	= time step level
T	= transpose
$\tilde{()}$	= approximated numerical flux
$\hat{()}$	= any variable including metric terms

Introduction

THE conventional Navier–Stokes solvers usually employ an algebraic eddy viscosity model such as the Baldwin–Lomax model,[1] although the use of algebraic eddy viscosity models limit the applicability of Navier–Stokes solver to relatively simple flows. Two-equation models of turbulence are very popular in solving complex, incompressible, turbulent flows in various engineering applications and have been applied to high speed compressible flows such as flows past airfoil at angle of attack,[2] compressible corner,[3] cascade,[4,5] backward facing step[6,7] and airfoil/wing.[8,9] The predictions are in good agreement with experiments. However, it is found that it is difficult to achieve convergence in turbulent transonic separated flow computation by employing a k–ϵ two-equation model[10] or an algebraic Reynold stress (ASM)/k–ϵ two-layer model of turbulence.[11] The slow convergence rate encountered is related to strongly nonlinear flow phenomena of turbulence. The acceleration technique is desirable for problems of highly nonlinear systems.

Slow convergence for the solution of a system has been greatly improved by using the conjugate gradient (CG) methods for many applications. Preconditioned conjugate gradient, and its generalizations, have been used for some existing computational fluid dynamics codes,[12,13] and Venkatakrishnan[14] has applied them to obtain solutions of the compressible Navier–Stokes equations for subsonic and transonic flows. Ajmani et al.[15] used the generalized minimal residual (GMRES) method of Saad and Schultz[16] for transonic and hypersonic flow and achieved fast convergence. However, only the algebraic turbulent model of Baldwin–Lomax is incorporated in their studies. The acceleration of convergence has not been tested by the CG-related algorithm for using higher order turbulence closure. The primary goal of this study is to search out efficient methods to improve the convergence rate. In recent years, the conjugate gradient squared (CGS) method[17] has been recognized as an attractive iterative method for the solution of certain classes of nonsymmetric linear systems. Recent studies[18,19] indicate that the method is often competitive in convergence acceleration with the GMRES method and is economical in both storage requirements and computing time. In many situations, however, the CGS method could face with an irregular convergence behavior especially for iterations close to the solution. The irregularities in the iteration process may even spoil the solution. In this study, two variants of the biconjugate gradient (Bi-CG) method are employed to avoid this problem, the biconjugate gradient stable (Bi-CGSTAB) method by Van Der Vorst[18] and the transpose of free quasiminimal residue (TFQMR) algorithm method by Freund and Nachtigal.[19] The incomplete lower upper factorization[20] (ILU) is selected as the preconditioner to improve the rate of convergence in the present study.

The preconditioned variants of the biconjugate gradient method are implemented to the compressible Navier–Stokes solver with a two-equation model of turbulence. The present solver demonstrates

Received April 22, 1993; presented as Paper 93-3316 at the AIAA 11th Computational Fluid Dynamics Conference, Orlando, FL, July 6–9, 1993; revision received Dec. 5, 1994; accepted for publication Dec. 6, 1994.

*Ph.D. Candidate, Department of Power Mechanical Engineering; currently at Chung-Shan Institute of Science and Technology, P.O. Box Lung-Tan Box 90008-15-5, Taiwan, Republic of China.

†Ph.D. Candidate, Department of Nuclear Engineering.

‡Professor, Department of Nuclear Engineering. Member AIAA.

more efficient and exhibits better convergence behavior than the conventional solvers. Axisymmetric transonic turbulent flow over a bump and over the flat base projectile are the test problems. The first test case in challenging in terms of the strength of the interaction between the shock and the boundary layer and the shock-induced separation over the bump. The second test case is more difficult because the flowfield is involved with not only the strength of interaction between the shock and the boundary layer but also the structure of the recirculation flow, the shock wave location, and shock strength. The base flow computation is also an interesting topic in fluid dynamics; base pressure is a part of major contribution of drag for bodies of revolution.

Governing Equations and Numerical Methods

For this study, the time-dependent, mass-averaged, compressible Navier–Stokes equations for axisymmetric flow are employed. The two-equation model used here is Chien's k–ϵ model,[21] which is similar to that of Jones and Launder.[22] The resulting nondimensional equations, in conservation law form, can be formulated in curvilinear coordinates as follows:

$$\partial_t \hat{\boldsymbol{Q}} + \partial_\xi \hat{\boldsymbol{E}} + \partial_\eta \hat{\boldsymbol{F}} = \partial_\xi \hat{\boldsymbol{M}} + \partial_\eta \hat{\boldsymbol{N}} + \hat{\boldsymbol{H}} \tag{1}$$

where $\hat{\boldsymbol{Q}}, \hat{\boldsymbol{E}}, \hat{\boldsymbol{F}}, \hat{\boldsymbol{H}}, \hat{\boldsymbol{M}}, \hat{\boldsymbol{N}}$ are 6×1 column matrices. $\hat{\boldsymbol{Q}}$ is the vector of dependent variables and $\hat{\boldsymbol{Q}} = \boldsymbol{Q}/J = J^{-1}\{\rho, \rho u, \rho v, E_t, \rho k, \rho\epsilon\}$. The vectors of $\hat{\boldsymbol{E}}$ and $\hat{\boldsymbol{F}}$ represent corresponding convective fluxes and contain the convection and pressure terms, but the vectors of $\hat{\boldsymbol{M}}$ and $\hat{\boldsymbol{N}}$ describe diffusive fluxes. The vector of $\hat{\boldsymbol{H}}$ contains the source terms associated with axisymmetric coordinates and terms associated with production, dissipation of turbulent energy, and dissipation rate. The detailed equations can be found in Ref. 10.

The finite volume approach is used for the formulations of difference equations. In this study, the total variation diminishing (TVD) numerical flux (Harten's second-order scheme[23]) is applied for the convective term ($\tilde{E}$, $\tilde{F}$) and the central difference approximation is used for the viscous term ($\tilde{M}$, $\tilde{N}$). The full Navier–Stokes equations and k–ϵ equations are discretized by the implicit backward-Euler scheme for time derivatives and by the upwind differencing in the ξ and η directions for spatial derivatives of $\hat{A}^{\pm}$ and $\hat{B}^{\pm}$. Equation (1) can be written in the form[24]

$$\begin{aligned}
&[I + \Delta t(\partial_\xi^- \hat{A}^+ + \partial_\xi^+ \hat{A}^- + \partial_\eta^- \hat{B}^+ + \partial_\eta^+ \hat{B}^-) - \Delta t \hat{D}]^n (\Delta \hat{Q})_{i,j}^n \\
&\quad = -\frac{\Delta t}{\Delta \xi}\left[(\tilde{E} - \tilde{M})_{i+\frac{1}{2},j} - (\tilde{E} - \tilde{M})_{i-\frac{1}{2},j}\right]^n \\
&\quad - \frac{\Delta t}{\Delta \eta}\left[(\tilde{F} - \tilde{N})_{i,j+\frac{1}{2}} - (\tilde{F} - \tilde{N})_{i,j-\frac{1}{2}}\right]^n + \Delta t \hat{H}_{i,j}^n
\end{aligned} \tag{2}$$

where $\hat{A}^+$, $\hat{A}^-$, $\hat{B}^+$, and $\hat{B}^-$ are the positive and negative splitting Jacobian matrices in the ξ and η directions. The source term $\hat{\boldsymbol{H}}$ associated with the turbulence field variables can be very large, and it is composed of production, dissipation, and decay terms and becomes dominant over the convection and diffusion terms near the wall. The Jacobian matrix $\hat{D}$ is formed by taking the partial derivatives of k–ϵ source terms H_5 and H_6 with respect to conservative variables ρk and $\rho\epsilon$ (Ref. 10).

Equations (2) can be rearranged in the following form:

$$\begin{aligned}
&A_{i,j}(\Delta Q)_{i-1,j} + C_{i,j}(\Delta Q)_{i+1,j} + E_{i,j}(\Delta Q)_{i,j-1} \\
&\quad + F_{i,j}(\Delta Q)_{i,j+1} + D_{i,j}(\Delta Q)_{i,j} = B_{i,j}
\end{aligned}$$

where

$$\begin{aligned}
B_{i,j} &= \{\text{RHS of Eq. (2)}\}/\text{Vol}_{i,j} \\
A_{i,j} &= -(\Delta t/\text{Vol}_{i,j})\hat{A}^+_{i-\frac{1}{2},j} \\
C_{i,j} &= -(\Delta t/\text{Vol}_{i,j})\hat{A}^-_{i+\frac{1}{2},j} \\
E_{i,j} &= -(\Delta t/\text{Vol}_{i,j})\hat{B}^+_{i,j-\frac{1}{2}} \\
F_{i,j} &= -(\Delta t/\text{Vol}_{i,j})\hat{B}^-_{i,j+\frac{1}{2}} \\
D_{i,j} &= (\Delta t/\text{Vol}_{i,j})\left(\hat{A}^+_{i+\frac{1}{2},j} - \hat{A}^-_{i-\frac{1}{2},j} + \hat{B}^+_{i,j+\frac{1}{2}}\right. \\
&\quad \left. - \hat{B}^-_{i,j-\frac{1}{2}}\right) + I + (\Delta t)\hat{D}_{i,j}
\end{aligned} \tag{3}$$

Equation (3) forms a diagonally dominate block pentadiagonal matrix system of equations, $Ax = b$, where A is an $(NI\times NJ\times 6)\times(NI\times NJ\times 6)$ banded matrix, x and b are $(NI\times NJ\times 6)$ column matrices. The system equations can be represented in the matrix form as Fig. 1. The coefficient matrices $A_{i,j}$, $C_{i,j}$, $D_{i,j}$, $E_{i,j}$, and $F_{i,j}$ are 6×6 square matrices and the right-hand-side matrix $\boldsymbol{B}_{i,j}$ is a 6×1 vector. The unknown $(\Delta Q)_{i,j}$ is conventionally evaluated by using the block Gauss–Seidel method with slow convergence. The approximate factorization (AF) method is conventionally employed to achieve better convergence for many Navier–Stokes solvers, but the present work proposes the use of variants of the biconjugate gradient iterative method to accelerate and stabilize the convergence; a comparison will be made with the conventional AF method.

For solving the linear system $Ax = b$ with A symmetric positive definite (SPD), the classical conjugate gradient method of Hestens and Stiefel[25] is one of the most powerful iterative schemes, especially when combined with preconditioning techniques. The biconjugate gradient algorithm[26] is an extension of the CG method to a linear system with general nonsymmetry nonsingular coefficient matrices. The cost of using the Bi-CG algorithm is twice the computational work for the CG algorithm, and the manipulation between the A^T (transpose matrix of A) and some vectors included in the algorithm causes very cumbersome programming. Several variants of the biconjugate gradient method have been proposed and recognized to resolve these negative effects, such as CGS,[17] Bi-CGSTAB,[18] and TFQMR.[19] These methods are tested and compared for the two test cases in the present study.

Since convergence acceleration is one of the major concerns for this study, preconditioning is employed to improve the matrix conditioning due to the high dependence of the convergence rate on the eigenvalue distribution of the coefficient matrix and the condition number, i.e., instead of solving the original linear system $Ax = b$, the preconditioned conjugate gradient methods solve a related linear system $K^{-1}Ax = K^{-1}b$ by a conjugate gradient method with an auxiliary matrix K. The matrix production of $K^{-1}A$ has a more favorite eigenspectrum distribution than A has, i.e., the eigenvalues are more clustered near 1 or the condition number of $K^{-1}A$ is smaller than that of A (Ref. 27). The choice of the preconditioner in the present study is based on the incomplete factorization of the matrix A, which consists of a strictly block-lower triangular matrix of A, $\mathcal{L}$, a strictly block-upper triangular matrix of A, $\mathcal{U}$, and a nonzero block-diagonal matrix $\mathcal{D}$, satisfying $A = (\mathcal{L}+\mathcal{D})\mathcal{D}^{-1}(\mathcal{U}+\mathcal{D})+\mathcal{E}$, where $\mathcal{E}$ is the deviation matrix. The deviation matrix $\mathcal{E}$ is often very sparse itself. This factorization method is called the incomplete line LU (ILU) factorization. Preconditioning with ILU is done by choosing $K = (\mathcal{L}+\mathcal{D})\mathcal{D}^{-1}(\mathcal{U}+\mathcal{D})$ (Ref. 20), where $\mathcal{D}$ is slightly modified form D and is evaluated as follows[18]:

$$\mathcal{D}_{i,j} = D_{i,j} - E_{i,j}\mathcal{D}^{-1}_{i,j-1}F_{i,j-1} - A_{i,j}\mathcal{D}^{-1}_{i-1,j}C_{i-1,j}$$

The CGS method produces x_n iterates of which the residual satisfies $r_n = P_n^2(A)r_0$, where $P_n(A)$ is a special polynomial of degree less than or equal to n. The CGS method has been recognized as an attractive variant of the Bi-CG method for the solution of certain classes of nonsymmetric linear systems. In many situations, however, one is faced with quite irregular convergence behavior of CGS, in particular, in situations when the iteration starts close to the solution. A more smoothly converging variant of Bi-CG, without giving up the attractive speed of convergence of CGS, is derived by Van Der Vorst.[18] He modified the relation $r_n = P_n^2(A)r_0$ by introducing a special polynomial $Q_n(A)$ and the residual vector r_n satisfies the relation $r_n = Q_n(A)P_n(A)r_0$. It is the scheme in which the residuals $Q_n(A)P_n(A)r_0$ is developed and it uses local steepest descent steps to obtain a more smoothly convergent CGS-like process. Because of its similarity to CGS, its favorable stability properties, and its relation with Bi-CG,[30] it was named the Bi-CGSTAB method. Although Bi-CGSTAB seems to work well in many cases, it still exhibits irregular convergence behavior and a slower convergent rate than the CGS method for some difficult problems.

Freund[19] proposed the TFQMR method which is defined by a quasiminimization of the residual norm without working with A^T. He demonstrated that the resulting TFQMR method can be implemented very easily by changing a few lines in the standard CGS

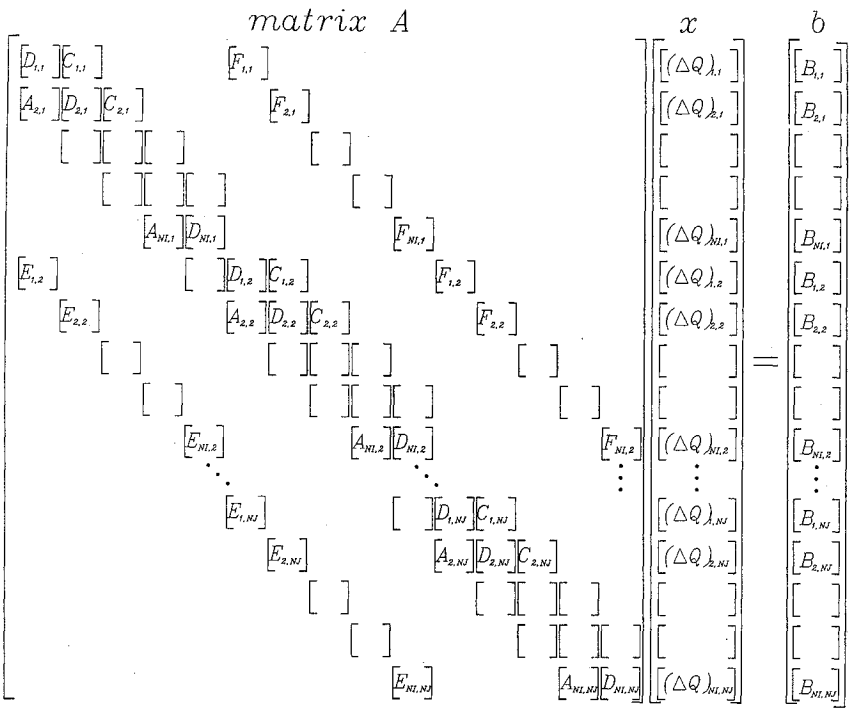

Fig. 1 Block pentadiagonal matrix system of Eq. (3).

algorithm, and the quasiminimization property may remedy the irregular convergence behavior with wild oscillations in the residual norm resulting from the CGS method. The work and storage per iteration are roughly the same for all three methods.

Initial conditions are chosen to be a uniform flowfield of freestream conditions for mean flow equations. The initial condition to start the k–ϵ calculations is a uniform field of 2% turbulence intensity,

$$k_0 = 1.5(0.02v_\infty)^2; \qquad \epsilon_0 = k_0^{1.5}/0.01$$

where v_∞ is the freestream speed.

The characteristic extrapolating technique is applied at the far field ($\eta = \eta_{\max}$) and the outflow positions (i.e., $\xi = \xi_{\max}$). The convergence of the present Navier–Stokes solver is reached faster by this method than by setting a constant of freestream value at the far field boundary. A no-slip boundary condition is adopted on the solid surface of the projectile for velocities u and v. The point $J = 2$ is very close to the wall, hence, the density and pressure on the wall are set to be equal to the values of node points next to the wall.

For the k–ϵ calculation, a zeroth-order extrapolation of the k–ϵ values is used to specify conditions at the outflow and axisymmetric boundary. For the far field inflow condition, the values of turbulent viscosity μ_t, k, and ϵ are set to be small (i.e., 10^{-10} for k_∞ and $\mu_{t,\infty}$, and 10^{-8} for ϵ_∞).

Results and Discussions

Numerical computations have been made for a transonic turbulent flow past an axisymmetric circular-arc bump for freestream Mach number of 0.875 and a unit Reynolds number of 13.6×10^6/m. The computed results of the mean flow and turbulent properties are compared with experimental data and with the results using different implicit methods. The iteration histories using the AF algorithm, CGS method, and the Bi-CGSTAB and TFQMR algorithms with ILU preconditioner are compared to the maximum possible Courant number according to stability restriction.

The grid systems used in the numerical computations are established using a hyperbolic solver.[28] A schematic diagram of the axisymmetric bump is shown in Fig. 2a. The circular-arc bump has a thickness of 1.91 cm and a chord length of 20.32 cm and is affixed to a thin-walled cylinder of outer diameter 15.2 cm. Its leading edge is jointed to the cylinder by a smooth circular arc

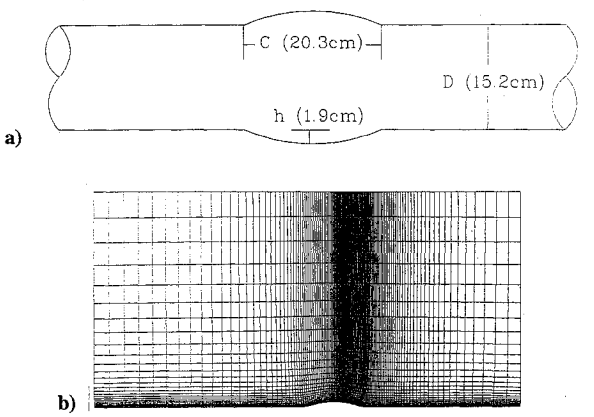

Fig. 2 Axisymmetric bump: a) geometry and b) 135 × 60 grid.

of 18.3-cm radius, which is tangent to the cylinder 3.33 cm upstream and to the bump at 2.05 cm downstream of the intersection of the bump arc with the cylinder. The grid system around the bump is a 135 × 60 H-type grid (Fig. 2b) with 45 points behind the trailing edge of the bump. The grid points in the normal direction are exponentially stretched away from the wall; the first grid line is at a distance of 0.00001 C (chord) off the wall with a y^+ value of ~0.55; 37 points are inside the boundary layer with $y^+ \leq 7000$. The boundaries of the computational mesh extend $4.5C$ in the normal direction and from $-4C$ to $4C$ in the flow direction. The calculated results are compared with experimental data.[29]

The convergence histories with different schemes are plotted in Fig. 3a. The L_2 residual, defined as the square root of

$$\sum_{i=1}^{N}\left[\left(\frac{\Delta\rho}{\Delta t}\right)_i^2 + \left(\frac{\Delta\rho u}{\Delta t}\right)_i^2 + \left(\frac{\Delta\rho v}{\Delta t}\right)_i^2 + \left(\frac{\Delta E_t}{\Delta t}\right)_i^2\right] \Big/ N$$

displays the overall convergence properties of the flowfields, where N is the total number of computational cells. The L_2 residual by the AF method is reduced from 10^{-2} to 10^{-4} after 4800 iterations, but it can not be reduced to 10^{-5} even after another 7000 iterations are

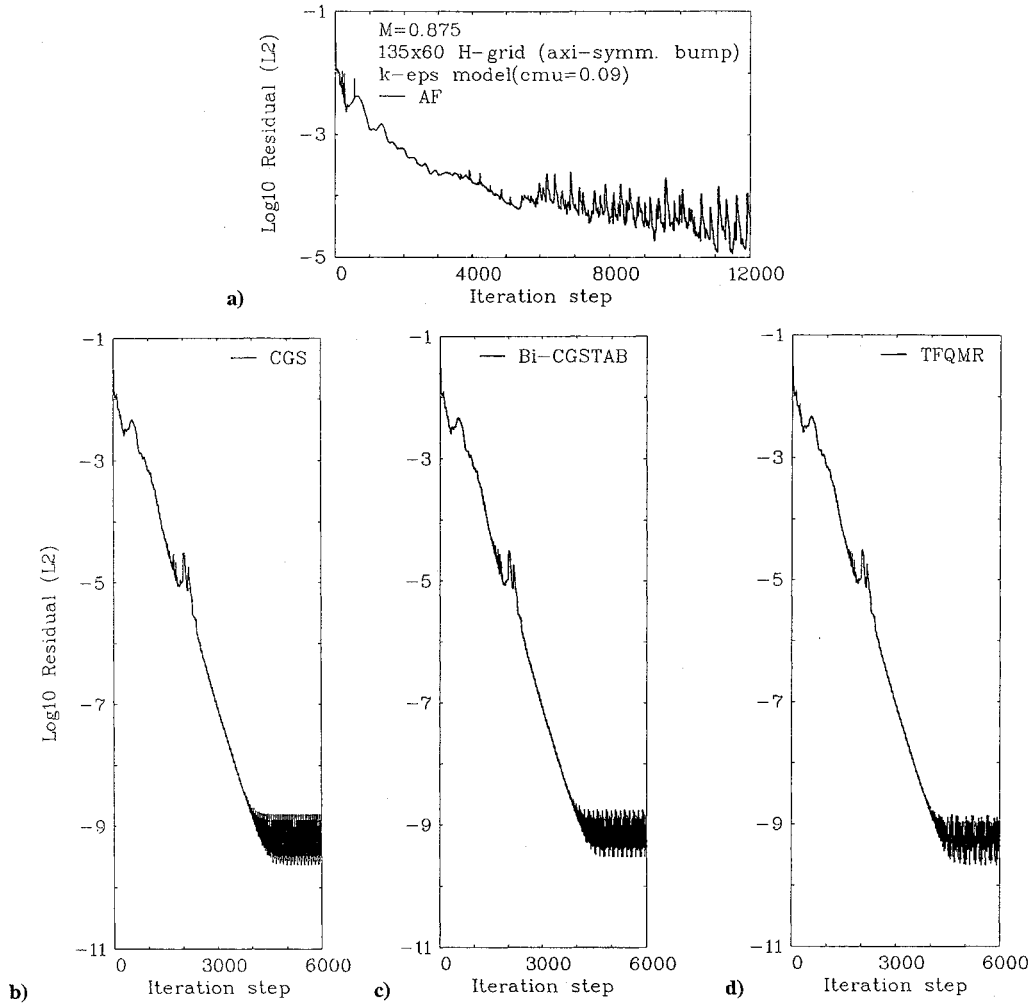

Fig. 3 Bump case convergence histories.

completed. A small time step (0.6) and a relaxation factor (0.12) for computing k and ϵ must be applied for the 12,000 iteration steps to keep the convergence stable using the AF method. The tolerance level is chosen as 10^{-5} for each time iteration step; approximately 1–2 iterates for the first 3500 time iteration steps and 3–4 iterates for later time steps are used for these variants of the BI-CG algorithms. The larger tolerance level implies a lower number of iterates for each time step and less CPU time per time step; however, more time step numbers are needed to reach the same low convergence level. The convergence may be broken down by using the CGS method if the number of iterates is small (for example, one iterate per time step) and/or a large time step is used. The BICGSTAB and TFQMR methods give more stable convergence behavior. The relaxation factor in computing k and ϵ and α_k and α_ϵ, is defined as $\rho k^{m+1} = \rho k^m + \alpha_k(\Delta\rho k)^m$, and $\rho\epsilon^{m+1} = \rho\epsilon^m + \alpha_\epsilon(\Delta\rho\epsilon)^m$ for the mth time step. The local time steps using the three variant BI-CG methods, CGS method, BI-CGSTAB method, and TFQMR method, can be linearly enlarged to 3.0 up to 1500 time steps, although the time steps are 0.6 initially for the first 300 steps. The local time step size is defined as $(\Delta t)_{\text{ref}}/(1 + \sqrt{J})$, where $(\Delta t)_{\text{ref}} = 3.0$ is equivalent to a CFL number of 345 for this test case. The relaxation factors for k and ϵ can be set to 1. The L_2 residuals are reduced to 10^{-5} after 2000 iterations and further reduced to 10^{-9} after another 2000 iterations; however, all three methods wildly oscillate after the 4000th iteration.

The CPU times needed for each iteration using the different implicit solvers are different, i.e., 4.4, 13.3, 13.5, and 15.2 for the AF method, the CGS method, the BI-CGSTAB method, and the TFQMR method, respectively, if the computations are performed by an HP-9000/730 scalar workstation. One "work unit" is defined as the CPU time required for an iteration step of the AF method. Each iteration step of the variants of the BI-CG method requires about 3 work units. The L_2 residuals can be reduced almost to a level of $\sim 10^{-5}$ (Fig. 3a) by using the AF method during 12,000 work units, but the residual can be reduced to the same level after approximately 6000 work units for the BI-CG-like method. A 50% overall gain in CPU time saved is achieved by using variants of the BI-CG method if the convergence level of a residual of 10^{-5} is acceptable for this test problem. The computed flowfields are still changing and oscillating after 12,000 iteration steps and more iterations need to be performed to reach a well-converged solution. The convergence behaviors of the three variants of the BI-CG method are very similar; the improvements in the CGS algorithm by using BI-CGSTAB or TFQMR are very limited for this problem because of the small time step and the higher number of iterates (3–4) for each time step used (Figs. 3b–3d).

The emphasis of present study is the convergence characteristics of the numerical methods and the influence of the convergence level on the computed results. Figure 4 shows the surface pressure coefficient distributions obtained by the different schemes. The surface pressure distribution by the AF method after 12,000 steps shows an insignificant difference from the results of the three variants of the Bi-CG method after only 3000 iteration steps, although they are not converged to the same convergence level. The mean velocity profiles predicted by the different numerical methods at various axial stations are compared with the measured data in Fig. 5. It shows that the computed results give good agreement in the region downstream of reattachment ($X/C \geq 1.25$) and before the shock position ($X/C \leq 0.875$). The results by three variants of the Bi-CG method are close to each other, except for those of the AF method locally. The turbulent kinetic energy profile predicted by the four implicit schemes indicates that Chien's k–ϵ model[21] overestimates

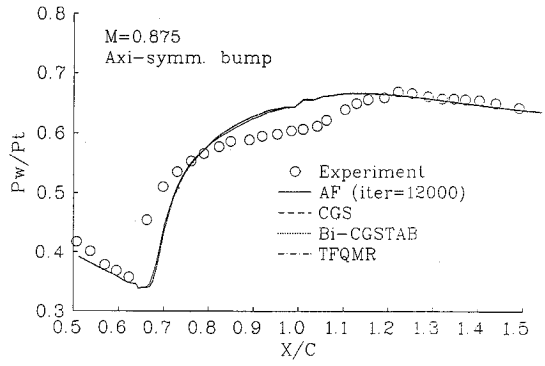

Fig. 4 Surface pressure distributions by different schemes for the bump case.

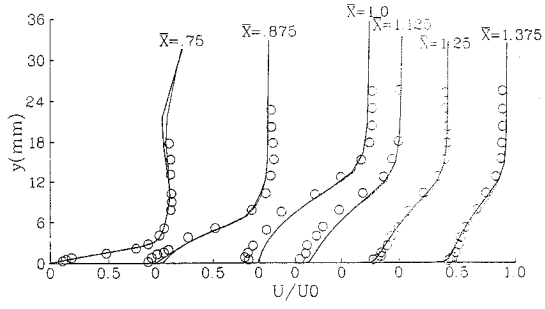

Fig. 5 Velocity profiles at various axial stations by different schemes for the bump case: M = 0.875; 135 × 60 grid; ———, AF(iter = 12,000); ....., Bi-CGSTAB; ○, experiment; - - - -, CGS; ·····, TFQMR.

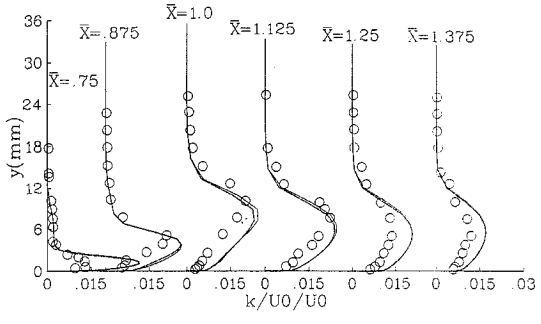

Fig. 6 Turbulent kinetic energy profiles at various axial stations by different schemes for the bump case: M = 0.875; 135 × 60 GRID; ———, AF(iter = 12,000); ....., Bi-CGSTAB; ○, experiment; - - - -, CG-SQUARE; ·····, TFQMR.

the magnitudes of the peaks (Fig. 6). The turbulent quantity result of the AF method is also apparently different from that of the another three solvers. The comparison of the computed results of surface pressure coefficient, velocity profile, and turbulent kinetic energy profile shows that the reduction of the convergence criteria lower than to 10^{-9} influences only the results of turbulent properties. It seems that the well-converged methods were not vital, as shown in this test problem. However, the following test problem will give a different implication.

Transonic flow past a secant-ogive-cylinder-boat tail (SOCBT) projectile with a flat base at zero angle of attack is another testing problem with experimental data from Chien and Danberg.[30] The flow pattern consists of two zones; one is boundary-layer-type flow around the projectile, the other is the recirculation zone after the base. The projectile model is shown in Fig. 7a with a half-O-type grid of 133 × 76 points, with 40 points lying on the half-base (Fig. 7b). The stations with numbers in Fig. 7a are the locations where the comparisons are made to the different schemes. The first grid line is at a distance of $D = 2 \times 10^{-5}$ with a y^+ value of 0.5. There are 30 points inside the boundary layer. A small time step of 0.5 and $\alpha_k = \alpha_\epsilon = 0.12$ are used during all 15,000 iterations for the

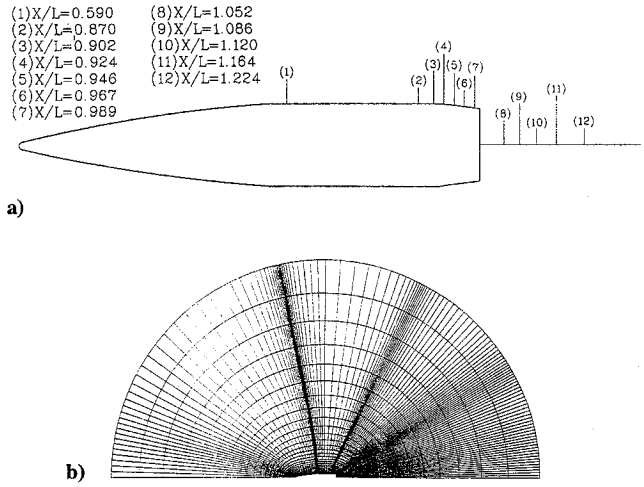

Fig. 7 SOCBT projectile a) geometry and b) 133 × 76 grid.

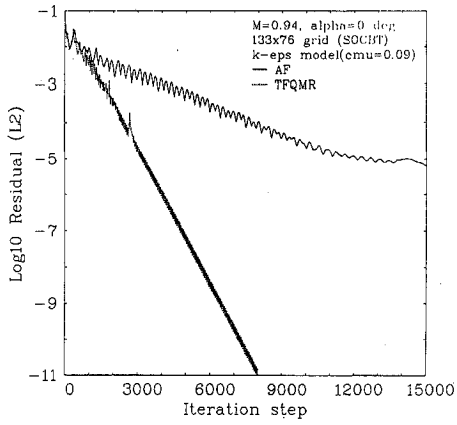

Fig. 8 Convergence histories by different schemes for the SOCBT case.

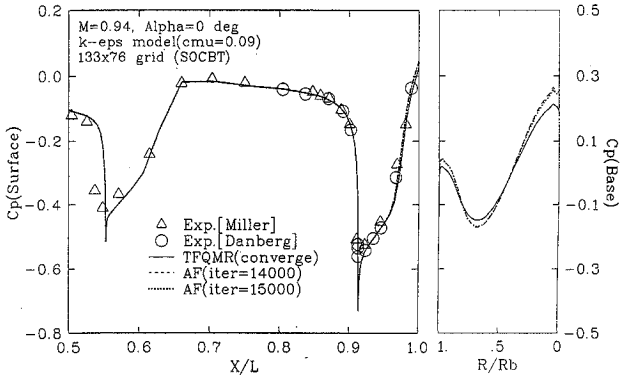

Fig. 9 Surface pressure distributions by different schemes for the SOCBT case.

AF method. For the computations by the TFQMR method, the time step can be enlarged linearly from 0.6 to 3.6 between 300 and 2000 iterations and kept at 3.6 after 2000 iterations. The relaxation factors of α_k and α_ϵ are 0.95. The time step size of 3.6 is equivalent to a CFL number of 300 for this case. The convergent histories by the AF and the TFQMR algorithms are plotted in Fig. 8. The L_2 residuals of the TFQMR method per iteration steps can be reduced both continuously and much faster than those of the AF method. The L_2 residuals of the AF method can only be reduced to 10^{-5} after 15,000 iteration steps and it is difficult to reduce them further, but the L_2 residuals of the TFQMR methods can be reduced to under 10^{-5} after only 3000 steps and to 10^{-11} after 8000 steps. Since one iteration step of

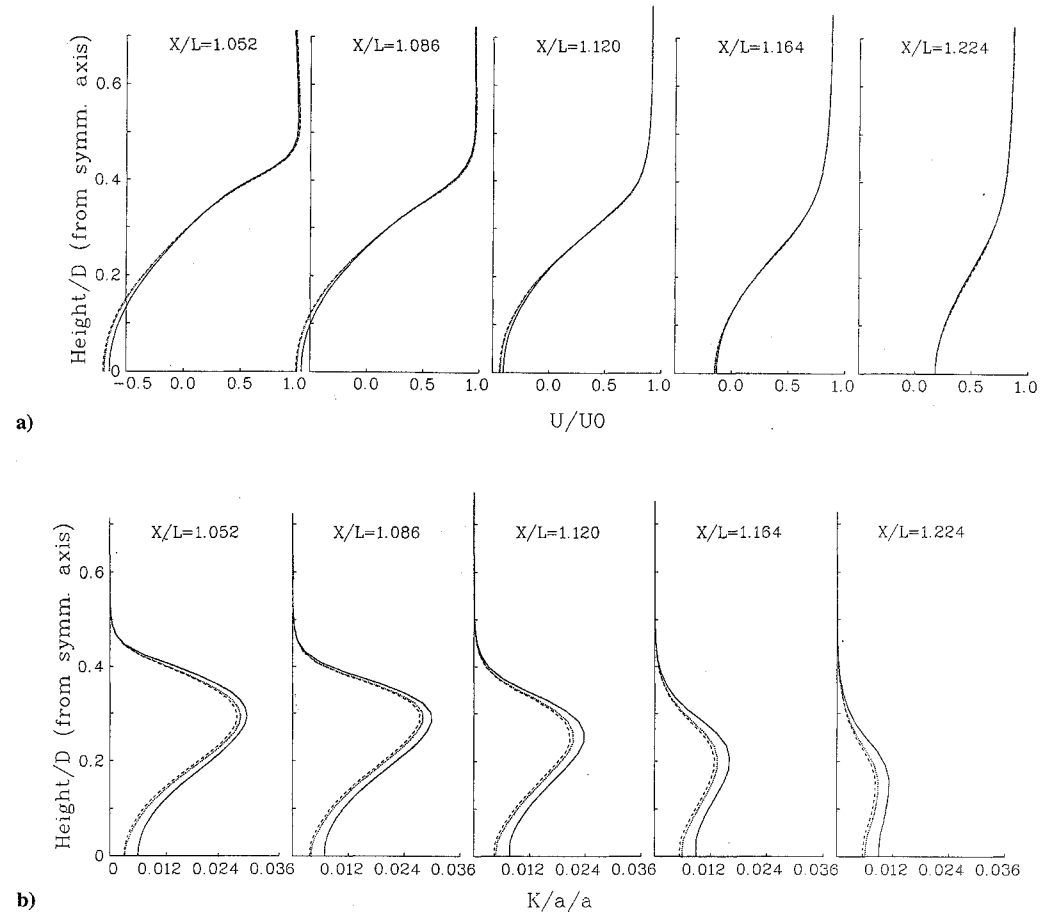

Fig. 10 Results by different schemes for the SOCBT case: a) axial velocity and b) turbulent kinetic energy profiles. M = 0.94, alpha = 0 deg, k-eps model-(cmu = 0.09), 133 × 76 grid-(SOCBT); ———, TFQMR-(converge); - - - -, AF-(iter = 14,000); ....., AF-(iter = 15,000).

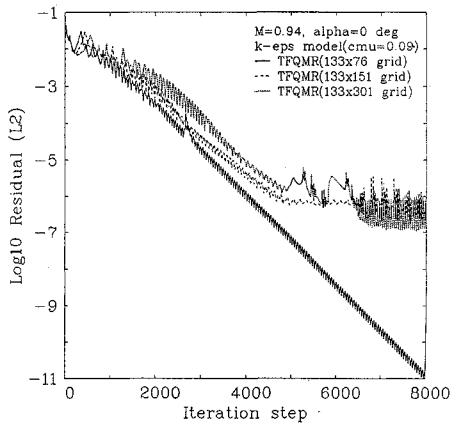

Fig. 11 Convergence histories by different grids for the SOCBT case.

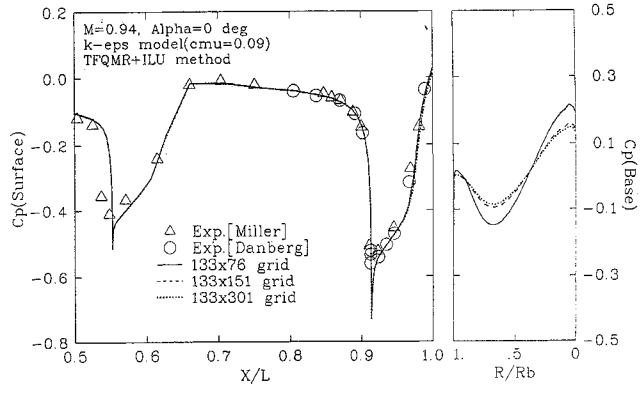

Fig. 12 Surface pressure distributions by different grids for the SOCBT case.

the TFQMR method requires 3 work units, there is a 40% gain in CPU time saved by using the TFQMR method as compared to the AF method if the convergence level of 10^{-5} is considered as converged. The computed flowfield parameters, such as surface pressures, velocity profiles (Fig. 10a) and turbulent properties (Fig. 10b) using the AF method are very close but different for the 14,000th and 15,000th time steps with an L_2 residual level of approximately 10^{-5}. This means that the convergence is not achieved yet.

Figure 9 indicates a small difference on the pressure distribution on the boattail surface but a significant difference on the base surface between the results by the AF method at the 15,000th iteration step and the results by the TFQMR method at the 4000th iteration (convergence level of 10^{-6}). This implies that an insufficient convergence leads to a significant deviation on the base pressure prediction and a small deviation on the boattail pressure prediction. Figure 10a plots the axial velocity profile at different X/L stations which are behind the base ($X/L = 1$) and it illustrates that the difference of the velocity profiles by the AF and TFQMR methods are larger at stations closer to the base, i.e., in the recirculation zone. Furthermore, great differences are observed in the turbulent properties computed by these two methods, with different convergence criteria as shown in Fig. 10b with the largest difference located on the axisymmetric axis.

It is well known that finer meshes result in more difficult convergence. However, grid independence is required for reliable computation. The recirculating flow behind the base needs a reasonably large number of grid points for good computation. Two other sets of grid systems are chosen to perform the computation: 133 × 151 and 133 × 301 points, i.e., more points are implemented in the η

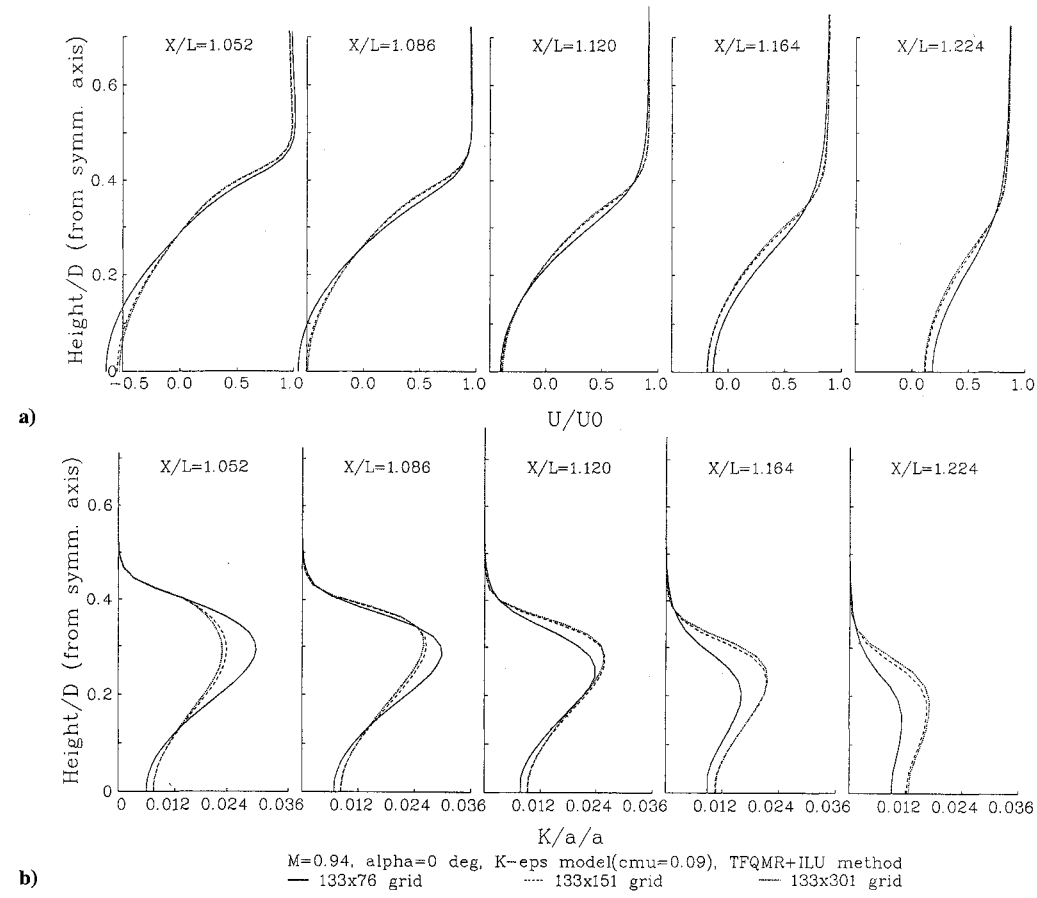

Fig. 13 Results by different grids for the SOCBT case: a) axial velocity and b) turbulent kinetic energy profiles.

direction and in the recirculation zone. The time steps for the computation of two finer grids are chosen to be equal to that of the case of the 133 × 76 grid. Figure 11 shows the convergence histories of these three grids using the same time step size. The coarse grid (133 × 76) exhibits the best convergence characteristics but the other two grids yield the L_2 residual flattened and it oscillated after 5000 iteration steps, the finest grid (133 × 301) starts the earliest and most severe oscillation. The computations using the AF method with the two finer grids diverge quickly even with much smaller time steps (for example, $\Delta t_{\text{ref}} = 0.05$), therefore, the computed results using the AF method with the finer grids are not included in this paper. Figure 12 plots the surface pressure coefficients along the projectile and its flat base. It indicates that finer meshes are needed to achieve grid independence, i.e., there is an apparent difference of computed pressure on the base surface although only a small difference on the boattail surface. Figures 13a and 13b show the mean velocity profiles and turbulent intensities at various axial stations using different grids; significant differences are observed in the recirculating flow behind the base, and an insignificant difference is shown on the boattail except at the station very close to the base corner.

In conclusion, an efficient method such as presently proposed variant of the Bi-CG method (TFQMR) must be used to obtain a well-converged result especially if a fine-enough mesh is needed. Otherwise, there is very little chance to reach grid-independent results for the second test problem.

Conclusion

Three variants of preconditioned biconjugate gradient methods have been successfully implemented in the compressible Navier–Stokes solvers with the k–ϵ two-equation model for transonic separated flows. This study concludes that the convergence characteristics are largely improved by the application of the Bi-CGSTAB method, the TFQMR method and CGS method as compared to the AF method there is a 40–50% CPU time savings for the two test cases. The BICGSTAB method is the most highly recommended among the three variants of the BI-CG method because of a more stable convergence behavior than the CGS method and less CPU time with the same convergence rate compared to the TFQMR method. For the flow over an axisymmetric bump, there is a very slight difference in the surface pressure coefficient and a slight difference in the profiles of velocity and turbulent quantities observed if the L_2 residual levels using the different numerical schemes can not be reduced to the same low residual level. For the separated flow over the SOCBT projectile with a flat base, convergence is very difficult to achieve, especially for fine meshes if the conventional AF method is employed. Looking at even the "acceptable" converging results with some acceptable residual level by using the AF method, these AF computed results may be apparently different from the well-converged results obtained by the more efficient methods, such as TFQMR methods, with a much lower convergence level. For example, the parameters such as the distribution of pressure on the flat base, the computed velocity profiles, and the turbulent properties in the recirculation zone behind the base, are significantly different by AF and TFQMR methods. Therefore, use of the variants of Bi-CG methods can serve the purpose of fast and well-converged for transonic turbulent separated flows.

Acknowledgments

The authors want to thank the National Science Council of Republic of China for support under Contract NSC-82-0413-E-007-109. The authors also want to thank W. W. Lin of Applied Mathematics, National Tsing Hua University, for discussions and suggestions regarding the use of three variants of the Bi-CG method and T. M. Hung of NASA Ames for his kind advice.

References

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

[2]Coakley, T. J., "Turbulence Modeling Methods for the Compressible Navier–Stokes Equations," AIAA Paper 83-1693, July 1983.

[3]Viegas, J. R., and Rubesin, M. W., "On the Use of Wall Function as Boundary Conditions for Two-Dimensional Separated Compressible Flows," AIAA Paper 85-0180, Jan. 1985.

[4]Liu, J. S., "Navier–Stokes Cascade Analysis with a Stiff k–ϵ Turbulence Solver," AIAA Paper 88-0594, Jan. 1988.

[5]Yokota, J., "A Diagonally Inverted LU Implicit Multigrid Scheme for the 3-D Navier–Stokes Equations and a Two Equation Model of Turbulence," AIAA Paper 89-0467, Jan. 1989.

[6]Nichols, R. H., "A Two-Equation Model for Compressible Flows," AIAA Paper 90-0494, Jan. 1990.

[7]Goldberg, U., and Ota, D., "A k–ϵ Near-Wall Formulation for Separated Flows," AIAA Paper 90-1482, June 1990.

[8]Marcum, D. L., and Agarwal, R. K., "A Three-Dimensional Finite Element Navier–Stokes Solver with k–ϵ Turbulence Model for Unstructured Grids," AIAA Paper 90-1652, June 1990.

[9]Mavriplis, D., "Multigrid Solution of Compressible Turbulent Flow on Unstructured Meshes Using a Two-Equation Turbulence Model," AIAA Paper 91-0237, Jan. 1991.

[10]Lin, H., and Chieng, C. C., "Comparisons of TVD Schemes for Turbulent Transonic Projectile Aerodynamics Computations with a Two-Equation Model of Turbulence," *International Journal for Numerical Methods in Fluids*, Vol. 16, No. 5, 1993, pp. 365–390 (AIAA Paper 91-2858, Aug. 1991).

[11]Chuang, C. C., and Chieng, C. C., "Transonic Turbulent Separated Flow Prediction using a Two-Layer Turbulence Model," *AIAA Journal*, Vol. 31, No. 5, 1993, pp. 816, 817 (AIAA Paper 92-0518, Jan. 1992).

[12]Wigton, L. B., Yu, N. J., and Young, D. P., "GMRES Acceleration of Computational Fluid Dynamic Codes," AIAA Paper 85-1494, July 1985.

[13]Obayashi, S., "Numerical Simulation of Underexpanded Plumes Using Upwind Algorithms," AIAA Paper 88-4360, Aug. 1988.

[14]Venkatakrishnan, V., "Preconditioned Conjugate Gradient Methods for the Compressible Navier–Stokes Equations," *AIAA Journal*, Vol. 29, No. 7, 1991, pp. 1092–1100.

[15]Ajmani, K., Ng, W.-F., and Liou, M.-S., "Generalized Conjugate-Gradient Methods for the Navier–Stokes Equations," AIAA Paper 91-1556, 1991.

[16]Saad, Y., and Schultz, M., "GMRES: A Generalized Minimum Residual Algorithm for Solving Nonsymmetric Linear Systems," *SIAM Journal on Scientific and Statistical Computing*, Vol. 7, No. 3, 1986, pp. 856–869.

[17]Sonneveld, P., "CGS, a Fast Lanczos-type Solver for Nonsymmetric Linear Systems," *SIAM Journal on Scientific and Statistical Computing*, Vol. 10, No. 1, 1989, pp. 36–52.

[18]Van Der Vorst, H. A., "Bi-CGSTAB: A Fast and Smoothly Converging Variant of Bi-CG for the Solution of Nonsymmetric Linear Systems," *SIAM Journal on Scientific and Statistical Computing*, Vol. 13, No. 2, 1992, pp. 631–644.

[19]Freund, R. W., "A Transpose-Free Quasi-Minimal Residual Algorithm for Non-Hermitian Linear Systems," *SIAM Journal for Scientific and Statistical Computing*, Vol. 14, No. 2, 1993, pp. 470–482.

[20]Meijerink, T. A., and Van Der Vorst, H. A., "Guidelines for the Usage of Incomplete Decompositions in Solving Sets of Linear Equations as they Occur in Practical Problems," *Journal of Computational Physics*, Vol. 44, No. 1, 1981, pp. 134–155.

[21]Chien, K. Y., "Predictions of Channel and Boundary-Layer Flows with a Low-Reynolds-Number Turbulence Model," *AIAA Journal*, Vol. 20, No. 1, 1982, pp. 33–38.

[22]Jones, W. P., and Launder, B. E., "The Prediction of Laminarization with a Two-Equation Model of Turbulence," *International Journal of Heat and Mass Transfer*, Vol. 15, No. 2, 1972, pp. 301–314.

[23]Yee, H. C., and Harten, A., "Implicit TVD Schemes for Hyperbolic Conservation Laws in Curvilinear Coordinated," *AIAA Journal*, Vol. 25, No. 2, 1987, pp. 266–274.

[24]MacCormack, R. W., "Current Status of Numerical Solutions of the Navier–Stokes Equations," AIAA Paper 85-0032, Jan. 1985.

[25]Hestenes, M. R., and Stiefel, E., "Methods of Conjugate Gradients for Solving Linear Systems," *Journal of Research of the National Bureau of Standards*, Vol. 49, No. 6, 1952, pp. 409–436.

[26]Fletcher, R., "Conjugate Gradient Methods for Indefinite Systems," *Lecture Notes in Mathematics*, Vol. 506, Springer–Verlag, Berlin, 1976, pp. 73–89.

[27]Van Der Vorst, H. A., "Preconditioning by Incomplete Decompositions," Ph.D. Thesis, ACCU-Series 32, Academic Computer Centre Utrecht, Univ. of Utrecht, The Netherlands, 1982.

[28]Steger, J. L., and Chaussee, D. S., "Generation of Body Fitted Coordinates Using Hyperbolic Differential Equations," Flow Simulations, Inc., FSI Rept. 80-1, Sunnyvale, CA, Jan. 1980.

[29]Johnson, D. A., and Horstman, C. C., "Comparison Between Experiment and Prediction for a Transonic Turbulent Separated Flow," *AIAA Journal*, Vol. 20. No. 6, 1982, pp. 737–744.

[30]Chieng, C. C., and Danberg, J. E., "Navier–Stokes Computations for a Transonic Projectile at Angle of Attack and Comparison with Experimental Data," AIAA Paper 88-2584, June 1988.