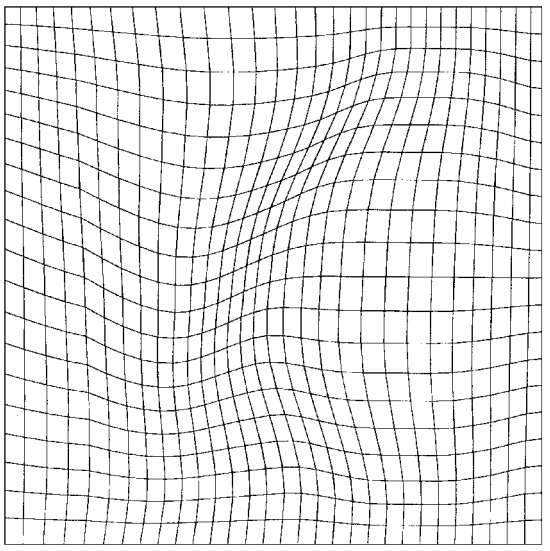

Fig. 2 Background grids on the (U, V) space by the proposed mapping with $n = 4$, $\alpha = 1$, and $\omega = 0.5$.

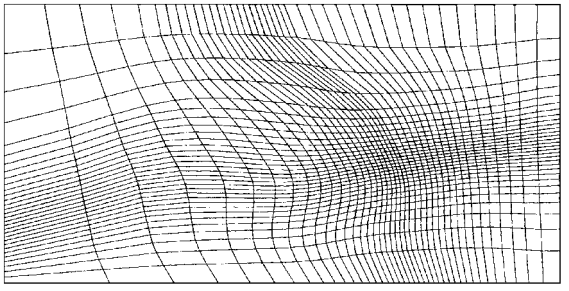

Fig. 3 Top view of the final surface grids.

such that the grid solver is automatically repeated with a larger ω whenever the grid quality is poor.

Conclusions

A modification of APS mapping previously proposed is shown. The new proposed method is a weighted average between UPS mapping and APS mapping that averages arclength over a large region. Numerical tests show that it improves grid smoothness of an extreme case while being very simple.

Acknowledgment

The work presented was supported by Taiwan, Republic of China, National Science Council Grant 85-2212-E006-090.

References

[1]Thompson, J. F., "A Reflection on Grid Generation in the 90s: Trends, Needs, and Influences," *Proceedings of the 5th International Conference on Numerical Grid Generation in Computational Field Simulations*, edited by B. K. Soni, J. F. Thompson, J. Häuser, and P. R. Eiseman, Mississippi State Univ., Mississippi State, MS, 1996, pp. 1029–1110.

[2]Samareh-Abolhassani, J., and Stewart, E., "Surface Grid Generation in a Parameter Space," *Journal of Computational Physics*, Vol. 113, No. 1, 1994, pp. 112–121.

[3]Soni, B. K., "Two- and Three-Dimensional Grid Generation for Internal Flow Applications of Computational Fluid Dynamics," AIAA Paper 85-1526, July 1985.

[4]Jeng, Y. N., and Hsu, W. C., "Enhancements upon the Algebraic Surface Grid Generation Method on Parameter Space," AIAA Paper 97-0203, Jan. 1997.

[5]Jeng, Y. N., and Lee, Z. S., "Revisit to the Modified Muliple One-Dimensional Adaptive Grid Method," *Numerical Heat Transfer*, Pt. B, Vol. 29, No. 3, 1996, pp. 305–323.

[6]Jeng, Y. N., and Liou, S. C., "Modified Multiple One-Dimensional Adaptive Grid Method," *Numerical Heat Transfer*, Pt. B, Vol. 15, No. 2, 1989, pp. 241–247.

[7]Lee, D., and Tusei, Y. M., "A Modified Adaptive Grid Method for Recirculating Flow," *International Journal for Numerical Methods in Fluids*, Vol. 14, No. 7, 1992, pp. 775–791.

[8]Lee, D., and Tusei, Y. M., "A Hybrid Adaptive Gridding Procedure for Recirculating Fluid Flow Problems," *Journal of Computational Physics*, Vol. 108, No. 1, 1993, pp. 122–141.

J. Kallinderis
Associate Editor

Efficiency Improvement of Unified Implicit Relaxation/Time Integration Algorithms

Kozo Fujii*
Institute for Space and Astronautical Science, Sagamihara, Kanagawa 229-8510, Japan

Introduction

FOR space-oriented applications of computational fluid dynamics, subsonic flow regions are locally confined and the steady flowfields to be simulated are mostly governed by supersonic flows. For the simulation of such flows, relaxation-type schemes are much more efficient compared with conventional time-integration schemes. Local iterations at each streamwise station are defined, and the converged solution is obtained there before moving to the next streamwise station. Since the middle of the 1980s quite a few papers have been written on this subject.[1–4] However, conventional time-integration schemes that use only global iterations are still more popular because relaxation-type computations are very efficient for supersonic flows but are usually not as efficient for subsonic and transonic flows. Recent implicit time-integration schemes[5,6] simplified the implicit operations and are much more efficient for subsonic and transonic flows.

In the present Note, relaxation-type algorithms developed in the 1980s are revisited, and their efficiency is improved by adopting the idea of both the lower–upper-symmetric Gauss–Seidel (LU-SGS) and the lower–upper alternating direction implicit (LU-ADI) schemes. Although the code structure is that of relaxation-type schemes, the present scheme is as efficient as conventional efficient schemes in the time-marching mode. When using the local iteration process, the scheme becomes much more efficient for simulations of flows primarily governed by the supersonic region.

Formulations

We start with the two-factored scheme proposed by Yin and Steger.[3] When using diagonally dominant lower–diagonal–upper (LDU) factorization instead of the LU factorization of the original formulation and dividing the equations by Δt, the modified two-factored implicit scheme becomes

$$\left(\frac{I}{\Delta t} - \frac{1}{\Delta \xi}\hat{A}_j^- + \delta_\xi^b \hat{A}^+ + \delta_\zeta \hat{C}\right)\left[\frac{I}{\Delta t} + \frac{1}{\Delta \xi}\left(\hat{A}_j^+ - \hat{A}_j^-\right)\right]^{-1} \times \left(\frac{I}{\Delta t} + \frac{1}{\Delta \xi}\hat{A}_j^+ + \delta_\xi^f \hat{A}^- + \delta_\eta \hat{B}\right)\Delta \hat{Q}^\eta = -\left(\frac{\partial \hat{E}}{\partial \xi} + \frac{\partial \hat{F}}{\partial \eta} + \frac{\partial \hat{G}}{\partial \zeta}\right)^n \tag{1}$$

Presented as Paper 97-2105 at the AIAA 13th Computational Fluid Dynamics Conference, Snowmass Village, CO, June 29–July 2, 1997; received July 26, 1997; revision received Aug. 19, 1998; accepted for publication Sept. 9, 1998.

*Professor, Space Transportation Research Division. Associate Fellow AIAA.

We consider Euler equations here, but the extension to the Navier–Stokes equations is straightforward. Subscript j is the index in the streamwise direction ξ. In this scheme, the flux Jacobian matrix $\hat{A}$ is divided into positive and negative parts, and upwind evaluation is used for the ξ derivatives. The first operator on the left-hand side (LHS) requires forward j sweep with a block matrix inversion of the ζ operator at each j. Similar operations are required for the second operator. The first operator of Eq. (1) at index j can be explicitly written as

$$\left[\frac{I}{\Delta t}+\frac{1}{\Delta \xi}\left(\hat{A}_j^+-\hat{A}_j^-\right)+\delta_\zeta \hat{C}\right]\Delta \hat{Q}^* = -\left(\frac{\partial \hat{E}}{\partial \xi}+\frac{\partial \hat{F}}{\partial \eta}+\frac{\partial \hat{G}}{\partial \zeta}\right)^n+\frac{1}{\Delta \xi}\hat{A}_{j-1}^+\Delta \hat{Q}_{j-1}^* \tag{2}$$

where

$$\Delta \hat{Q}^* = \left[(I/\Delta t)+(1/\Delta \xi)\left(\hat{A}_j^+-\hat{A}_j^-\right)\right]^{-1} \times\left[(I/\Delta t)+(1/\Delta \xi)\hat{A}_j^+ +\delta_\xi^f \hat{A}^- +\delta_\eta \hat{B}\right]\Delta \hat{Q}^\eta$$

Here first-order upwind differencing is used for the implicit operator for simplicity. The implicit operator cannot be diagonalized because it includes the flux Jacobians matrices for both ξ and ζ.

However, when Jacobian matrix $\hat{A}^+$ and $\hat{A}^-$ are approximated as $\hat{A}^+=(\hat{A}+\sigma_\xi)/2$ and $\hat{A}^-=(\hat{A}-\sigma_\xi)/2$ following the LU-SGS algorithm by Yoon and Jameson,[5] the implicit operator of Eq. (2) can be diagonalized by the eigenmatrices of ζ because $\hat{A}_j^+-\hat{A}_j^-=\sigma_{\xi_j} I$. Namely,

$$\left[\left(\frac{I}{\Delta t}+\frac{1}{\Delta \xi}\sigma_{\xi_j} I\right)+\delta_\zeta \hat{\Lambda}_\zeta\right]\left(R_\zeta^{-1}\Delta \hat{Q}^*\right) = -R_\zeta^{-1}\left[\left(\frac{\partial \hat{E}}{\partial \xi}+\frac{\partial \hat{F}}{\partial \eta}+\frac{\partial \hat{G}}{\partial \zeta}\right)^n+\frac{1}{\Delta \xi}\hat{A}_{j-1}^+\Delta \hat{Q}_{j-1}^*\right] \tag{3}$$

by operating R_ζ^{-1} from the left. The LHS operator is similar to that of the diagonal scheme[7] except for an additional diagonal term due to the spectral radius σ_{ξ_j}.

The LU-ADI scheme simplifies the diagonal scheme by replacing the resultant scalar matrix inversion by the two scalar forward and backward sweeps.[6] Here the same strategy can be applied. The LHS operator in Eq. (3) is factorized by the approximate LDU decomposition. The LHS operations at index j in the forward sweep of the ξ direction become

$$\left[\left(\frac{I}{\Delta t}+\frac{1}{\Delta \xi}\sigma_{\xi_j} I\right)+\delta_\zeta \hat{\Lambda}_\zeta\right]\left(R_\zeta^{-1}\Delta \hat{Q}^*\right) = \left[\left(\frac{I}{\Delta t}+\frac{1}{\Delta \xi}\sigma_{\xi_j} I\right)-\frac{1}{\Delta \zeta}\hat{\Lambda}_\zeta^- +\delta_\zeta \hat{\Lambda}_\zeta^+\right] \times\left[\left(\frac{I}{\Delta t}+\frac{1}{\Delta \xi}\sigma_{\xi_j} I\right)+\frac{1}{\Delta \zeta}|\hat{\Lambda}_\zeta|\right]^{-1}\left[\left(\frac{I}{\Delta t}+\frac{1}{\Delta \xi}\sigma_{\xi_j} I\right)+\frac{1}{\Delta \zeta}\hat{\Lambda}_\zeta^+ +\delta_\zeta \hat{\Lambda}_\zeta^-\right]\left(R_\zeta^{-1}\Delta \hat{Q}^*\right) \tag{4}$$

The first operator requires forward L (ζ direction) sweep with a scalar inversion at each L, the second operator requires scalar multiplication, and the third operator requires backward L sweep with a scalar inversion at each L. Similar approximate LDU factorization is applied to the third operator of Eq. (1) for η. In all of the implicit operators, upwind evaluations for the convective terms are obviously used. The resultant scheme consists of 1) a forward sweep in ξ: at each j, forward and backward sweep of ζ; and 2) a backward sweep in ξ: at each j, forward and backward sweep of η.

Only a scalar inversion is required, just as with the LU-ADI scheme. The difference is the addition of the spectral radius to the diagonal term, as can be seen in Eq. (4). Note that the second operator in Eq. (4) can be combined with the backward sweep. The computer code structure is very similar to the LU-ADI scheme except moving the J loop to be outside of the K and L loops, discarding the inversion of the ξ operator, and adding the $j\pm 1$ term on the right-hand side.

We may stop at Eq. (3) if a diagonal scheme is preferred. The modification from the diagonal scheme is almost the same. It is easily vectorizable and parallelizable because the code structure for the implicit operations is essentially the same as the LU-ADI scheme. The LU-SGS algorithm is one of the very efficient implicit schemes, but it is known that convergence is degraded when the ratio of the grid spacing is severely large, which happens in the direction from the body to the outer boundary. The present scheme uses the approximation of the flux Jacobians only in the ξ direction and thus can avoid this feature.

Thus far, we have discussed only time-asymptotic computations (global iteration). The merit of the present scheme lies in that it can be used as an efficient relaxation scheme for supersonic flows. Here we define local and global iterations. Assume that the J loop starts at $J1$ and ends at $J2$. Set the outer forward J loop of ξ for the relaxation mode in addition to the inner J loop (J_{in}) for the global iteration. When only the time marching is required, that is, the global iteration is preferred, set $J1_{\text{out}}=J2_{\text{out}}=1$ and make the outer J loop meaningless. When relaxation is required, that is, local iteration is preferred, set $J1_{\text{out}}=J1$ and $J2_{\text{out}}=J2$ and set $J1_{\text{in}}=J2_{\text{in}}=J_{\text{cur}}$, which is a current J index defined by the outer J loop. Local iterations at each J index are thus defined. Converged solutions are obtained at each J_{cur} before moving to the next J index defined by the outer J loop. In case several local subsonic regions exist, the two described strategies are combined by setting the flag for the span of J to choose local or global iterations.

We call the present scheme the four-factored symmetric Gauss–Seidel scheme (FF-SGS). In all of the computational examples to follow, the steady-state convective terms are evaluated by Roe's flux difference splitting using his approximate Riemann solver. Higher-order extension was done by the MUSCL interpolation. Viscous terms are evaluated by central differencing. Space variable time-stepping technique is used, and only steady-state problems are considered.

Computed Results

Only the computed results for the supersonic freestream are presented, although the algorithm works well for subsonic and transonic flows.[8] The first example is an inviscid supersonic flow over a cone. The computational grids are $51\times 39\times 81$. The freestream Mach number is 2.0, and the angle of attack is 8 deg. The convergence history for the time-marching mode is shown in Fig. 1

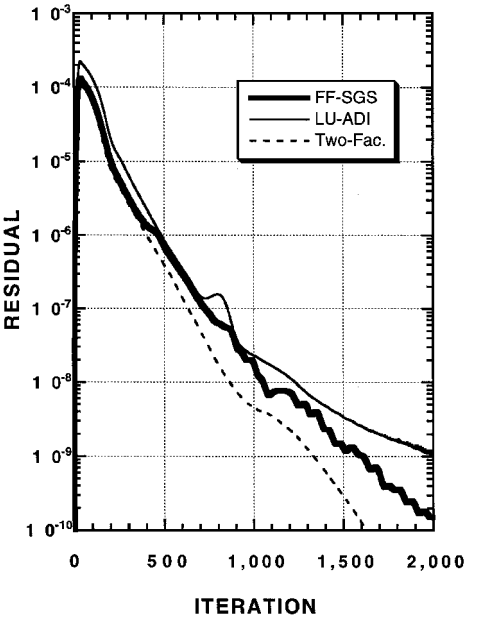

Fig. 1 Residual histories for the supersonic flow over a cone: inviscid, $M=2.0$, and $\alpha=8.0$.

and is compared with those of the LU-ADI and the original two-factored schemes. All of the computational conditions, such as the time step and the coefficients of the numerical dissipation term in the implicit operators, are set to be the same. The computed results by all three schemes are almost identical,[8] and they show good convergence. The result indicates that the simplification of the flux Jacobian matrices does not affect the convergence. The computer times necessary for these schemes are as follows: The present FF-SGS scheme required 1.22×10^{-5} s/iteration/grid. The LU-ADI scheme required 1.65×10^{-5} s/iteration/grid, and the two-factored scheme required 2.55×10^{-5} s/iteration/grid on the same computer. The present scheme shows the best convergence trend from the viewpoint of total computer time.

The same flowfield was computed with the relaxation mode (local iteration) by simply changing the input data. The convergence criterion at each J index is a $3\frac{1}{2}$-order-of-magnitude drop of the residual. The computation was done in $\frac{1}{8}$ of the computer time for the relaxation mode compared with the time-marching mode, and the computed contours are almost the same as those of the global iterations.

The second example is a supersonic flow over a blunted rocket nose. Navier–Stokes equations are used. In this case, a small subsonic region appears in the nose region. The Mach number is 2.0, and the angle of attack is 5 deg. The Reynolds number based on the diameter of the rocket is set at 1.0×10^6, and turbulent flow is assumed. The computational grids are $37 \times 39 \times 31$. The computed pressure contour plots are shown in Fig. 2. The residual history showed a fifth-order drop within 800 iterations and went down to machine zero.[8] Because the subsonic region is confined to the area near the nose, a time-marching/relaxation combined mode was tried. The input data were set so that time-marching mode is used for J to be 1–12, and the relaxation mode is used for the rest of the J indices. The surface pressure on the windward- and leeward-side symmetry planes are shown in Fig. 3. The results are almost identical for both modes, except in the regions of strong expansion and compression. The difference can be eliminated by setting a few hundred global iterations after the local iteration process is finished.

The final example is a supersonic flow over a space plane. The computational grids are $110 \times 119 \times 49$. The Mach number is 2.0, and the angle of attack is 0 deg. The Reynolds number is set at 1.0×10^6 based on the body length, and turbulent flow is assumed. The time-marching/relaxation combined mode was used. Density

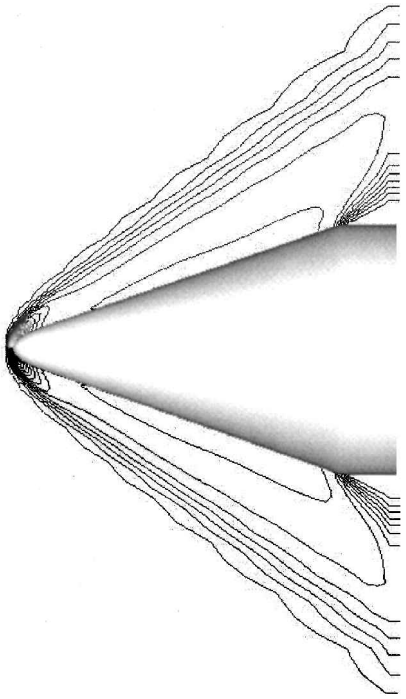

Fig. 2 Pressure contour plots for the nose of a rocket: $M = 2.0$, $\alpha = 5.0$, and $Re = 1.0 \times 10^6$.

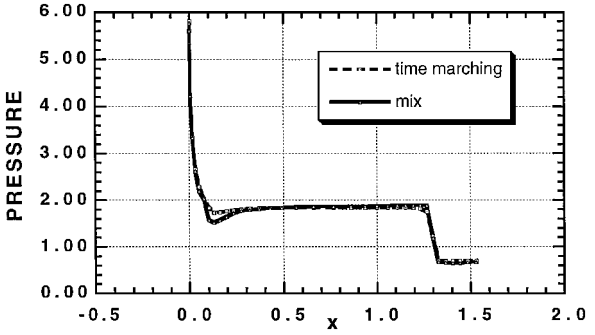

Fig. 3 Chordwise pressure distributions over the rocket nose; comparison of the time-marching and relaxation modes.

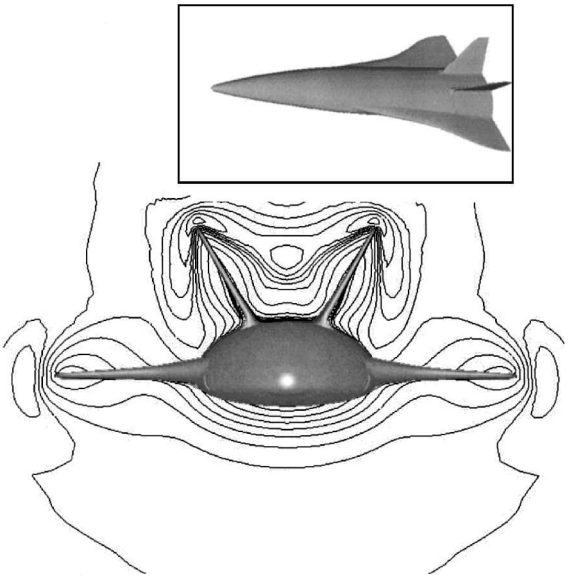

Fig. 4 Chordwise density contour plots for a spaceplane: $M = 2.0$, $\alpha = 0.0$, and $Re = 1.0 \times 10^6$.

contour plots at one chordwise station are shown in Fig. 4 along with the body configuration. The wing shock and the tail shock are both well captured. The result indicates that the present scheme works well for practical problems.

Conclusions

A new implicit time-integration/relaxation scheme was proposed. The scheme is based on the two-factored relaxation-type scheme in which streamwise direction is chosen as a sweep direction. By adopting the approximation of the flux Jacobian matrices used in the LU-SGS scheme for the streamwise direction and LDU approximate decomposition of LU-ADI scheme, a new implicit scheme was constructed. The scheme has four factors, each of which consists of simple scalar forward and backward sweeps and, therefore, is called the FF-SGS scheme. It was shown that the implicit operations in this scheme can be obtained by simply adding the spectral radius of the streamwise direction to the existing LU-ADI operations and, therefore, is easy to develop from the existing code.

Several computations were carried out, and it was shown that the present scheme is as efficient as existing conventional time-marching schemes. For the simulations where most of the flowfield is supersonic, local iterations at each streamwise station make the present scheme more efficient. The results indicated that the present FF-SGS scheme can be a versatile tool for simulations of a wide variety of flowfields.

References

[1]Chakravarthy, S. R., "Relaxation Methods for Unfactored Implicit Upwind Schemes," AIAA Paper 84-0165, Jan. 1984.

[2]Walters, R. W., and Dwoyer, D. L., "An Efficient Iteration Strategy for the Solution of the Euler Equations," *Proceedings of the AIAA 7th Computational Fluid Dynamics Conference*, AIAA, New York, 1985, pp. 381–393 (AIAA Paper 85-1529).

[3]Ying, S., and Steger, J. L., "Numerical Simulation of Unsteady, Viscous, High-Angle-of-Attack Flows Using a Partially Flux-Split Algorithm," AIAA Paper 86-2179, Aug. 1986.

[4]Newsome, R. W., Walters, R. W., and Thomas, J. L., "An Efficient Iteration Strategy for Upwind/Relaxation Solutions to the Thin-Layer Navier–Stokes Equations," *Proceedings of the AIAA 8th Computational Fluid Dynamics Conference*, AIAA, New York, 1987, pp. 126–142 (AIAA Paper 87-1113).

[5]Yoon, S., and Jameson, A., "An LU-SSSOR Scheme for the Euler and Navier–Stokes Equations," AIAA Paper 87-0600, Jan. 1987.

[6]Obayashi, S., Matsushima, K., Fujii, K., and Kuwahara, K., "Improvements in Efficiency and Reliability for Navier–Stokes Computations Using the LU-ADI Factorization Algorithm," AIAA Paper 86-0338, Jan. 1986.

[7]Pulliam, T. H., and Chaussee, D. S., "A Diagonal Form of an Implicit Approximate-Factorization Algorithm," *Journal of Computational Physics,* Vol. 39, 1977, pp. 372–397.

[8]Fujii, K., "Efficiency Improvement of Unified Implicit Relaxation/Time Integration Algorithm," *Proceedings of the AIAA 13th Computational Fluid Dynamics Conference*, AIAA, Reston, VA, 1997, pp. 161–167 (AIAA Paper 97-2105).

J. Kallinderis
Associate Editor

Numerical Study of Dynamic Stall on Several Airfoil Sections

Emmanuel Guilmineau* and Patrick Queutey*
École Centrale des Nantes, 44321 Nantes CEDEX 3, France

Introduction

THE term *dynamic stall* refers to the unsteady separation and stall phenomena of aerodynamic bodies or lifting surfaces that are forced to execute time-dependent motion, oscillatory or otherwise. It is a complex fluid dynamic phenomenon of practical importance and occurs on rapidly maneuvering aircraft, retreating helicopter rotor blades, fluttering compressor blades, and wind turbines. As summarized in extensive reviews by McCroskey[1] and Carr,[2] the unsteady flowfield generated when the airfoil reaches fairly high angles of attack during the oscillatory cycle and past the static stall angle limit is characterized by a massive unsteady separation and large-scale vortical structures. One important difference between this flowfield and that generated by the static stall is the large hysteresis in unsteady separation and reattachment of the vortex.

One reason why the flowfield associated with dynamic stall is more difficult to analyze than the static stall is its dependence on a much larger number of parameters, the most important being airfoil shape, Mach number, reduced frequency, amplitude of oscillations, type of motion (ramp or oscillatory), Reynolds number, three-dimensional effects, and wind-tunnel effects. The primary objective of the present study is to show the influence of the stall on different airfoil shapes.

Numerical Methods

The unsteady incompressible Reynolds-averaged Navier–Stokes equations (RANSE) are used to predict the stall of an airfoil. The RANSE are written in primitive formulation of the partial transformation and in a conservation form. The Cartesian velocity components and pressure share the same location at the center of the control volume. The numerical method uses a consistent physical reconstruction for the mass and momentum fluxes, the so-called consistent physical interpolation approach. This method is presented in detail by Deng et al.[3] for laminar problems and is extended to turbulent flow problems by Guilmineau et al.[4] The momentum and the continuity equations are solved in a segregated way, using the pressure implicit with splitting of operators algorithm. A second-order-accurate, three-level fully implicit time discretization is used.

In the present work, three turbulence models are used: the Baldwin–Barth (B–B) model,[5] which solves an additional equation for the turbulent Reynolds number, and both versions of the Menter K–ω model[6] [baseline version (BSL) and shear-stress transport version (SST)], which solves a first equation for the turbulent energy K and a second equation for the specific turbulent dissipation rate ω.

Results

Before presenting the results, some numerical parameters need to be specified. For each time step, a reduction of nonlinear residuals for the discrete momentum equations is required. By default, we use 10 nonlinear iterations by time step. The presented results focus on the third period of oscillations. All tests have shown that, in contrast to the findings of Raffel et al.,[7] the repeatability of calculations is good between the third cycle and those following. The oscillatory motion about the quarter-chord is defined by the time-dependent angle of attack

$$\alpha = \alpha_0 + \Delta\alpha\cos(2\pi f t) = \alpha_0 + \Delta\alpha\cos(2kt^*) \qquad (1)$$

where k is the reduced frequency, $k = 2\pi f c/2U_\infty$, and t^* the nondimensional time, $t^* = tU_\infty/c$.

All results to be presented were calculated using a 202×90 grid with a nondimensional time step of $\Delta t = 0.005$. The mesh is numerically generated by using a conformal mapping technique. The first points in fluid are located at $y^+ = 1$ away from the wall, and the outer flow boundary is located at 15 chord lengths away from the airfoil. For validation of unsteady, fully turbulent solutions, the experimental measurements[8] are used, which correspond to the deep-stall case. The Reynolds number, based on the airfoil chord length, is $Re = 10^6$. In this Note, the computations were obtained from three airfoils: the NACA 0012, the Sikorsky SC-1095, and the Vertol VR-7.

NACA 0012 Airfoil

The first considered case is that of a NACA 0012 airfoil with $\alpha_0 = 15$ deg, $\Delta\alpha = 10$ deg, and $k = 0.15$. The nondimensional period of the motion is $T^* = 20.94$. Two turbulence models have been tested: the B–B model and the SST K–ω model.

Let us consider an oscillation period starting from the minimum incidence (5 deg). As the airfoil reaches the proximity of the static stall angle ($\alpha_S \approx 12$ deg), the flowfield is still fully attached. Boundary layers on the upper surface, however, have grown considerably. Separation occurs around 20 deg. As the incidence increases, the dynamic stall vortex develops in both cases near the leading edge, although later with the K–ω model than the B–B model. This vortex is well known to be responsible for higher lift forces than those occurring at fixed incidence. At $\alpha = 23.6$ deg, the dynamic stall vortex is composed of a double structure with the B–B model. Twin eddies appear later with the K–ω model. As the airfoil incidence increases, a trailing-edge vortex is observed. At the maximum incidence, the velocity field changes drastically, and the dynamic stall vortex separates from the airfoil near the leading edge. As the trailing-edge vortex grows, the dynamic stall vortex lifts from the upper airfoil surface. This vortex takes place during a small fraction of the cycle, but its growth provokes a large oscillation lift coefficient. As the incidence decreases, the vortex core passes off the trailing edge at 24.6 deg with the K–ω model and at the maximum incidence with the B–B model. Such values can be compared with the value of 24.8 deg observed in the experimental study.[9] The shedding of the trailing-edge vortex is observed at 24.5 deg with the B–B model and at 24 deg with the K–ω model. Some secondary weak vortex structures subsequently develop. They are later shed into the wake. As the incidence decreases, the flow reattachment process starts from the leading edge downward. At $\alpha = 7.4$ deg, the reattachment of the flow occupies 60% of the chord in both turbulence models. In the experimental study,[9] it was reported that the complete reattachment occurred at $\alpha \approx 7$ deg. The evolution and the shedding of vortex

Received March 9, 1998; revision received Sept. 14, 1998; accepted for publication Sept. 16, 1998.

*CNRS Researcher, Laboratoire de Mécanique des Fluides, Centre National de la Recherche Scientifique Unité Mixte de Recherche 6598, Boite Postale 92101.