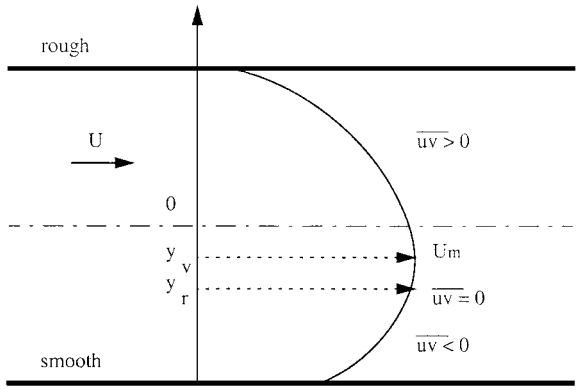

Fig. 1 Asymmetric plane channel flow.

If the SEV model [Eq. (11)] is applied, μ_t inevitably becomes negative, which is not physically meaningful.

By applying the present NAEV model [Eq. (22)] to this flow case, we have

$$\overline{uv} = -\mu_t \frac{\mathrm{d}U}{\mathrm{d}y} - \frac{2c_s k}{3c_1\varepsilon}\frac{\mathrm{d}}{\mathrm{d}y}\left[\frac{k^2}{\varepsilon}\frac{\mathrm{d}}{\mathrm{d}y}\left(\mu_t \frac{\mathrm{d}U}{\mathrm{d}y}\right)\right] \tag{24}$$

where we let $\bar{\mu}_t = \mu_t$. In Eq. (24) we can see that the conditions in Eq. (23) will be satisfied without μ_t becoming negative if

$$\mu_t \frac{\mathrm{d}U}{\mathrm{d}y} < -\frac{2c_s k}{3c_1\varepsilon}\frac{\mathrm{d}}{\mathrm{d}y}\left[\frac{k^2}{\varepsilon}\frac{\mathrm{d}}{\mathrm{d}y}\left(\mu_t \frac{\mathrm{d}U}{\mathrm{d}y}\right)\right] \tag{25}$$

To evaluate Eq. (24) further, the approximate analysis method used by Yoshizawa[7] is applied as follows.

Near the point y_v (<0), the asymmetric velocity profile is expressed as

$$U \approx U_m - b(y - y_v)^2 - c(y - y_v)^3 \tag{26}$$

where constants b and c are positive. For clarity, it is assumed that k, ε, and μ_t are constants near y_v. The combination of Eq. (26) with Eq. (24) gives

$$\overline{uv} \approx 2b\mu_t(y - y_v) + 3c\mu_t(y - y_v)^2 + \frac{4cc_s k^3 \mu_t}{c_1\varepsilon^2} \tag{27}$$

Therefore, the point y_r of zero $\overline{uv}$ is

$$y_r \approx y_v - \frac{3c}{2b}(y - y_v)^2 - \frac{2cc_s k^3}{bc_1\varepsilon^2} < y_v \tag{28}$$

The conditions in Eq. (23) are met without μ_t becoming negative. Note that the second term on the right-hand side of Eq. (24) is the term H_{ij} in Eq. (22). The capability of the present NAEV model to deal with the asymmetric turbulent flow between smooth and rough walls correctly is due to the existence of the term H_{ij} in Eq. (22). Because the NEAV model derived on the basis of Rodi's proposal does not have the term H_{ij}, it, too, will fail to describe the problem. The proposal given in Eq. (13) is the basis for the NAEV model to deal with the complex turbulent flow correctly.

It can be seen from Eq. (28) that the range between y_r and y_v mainly depends on the values of b, c, c_1, c_s, k, and ε. Within the scope of the present analytical work, we cannot provide the range quantitatively. In addition, comparison of the simulation by using SEV and NAEV turbulence models was addressed in Ref. 11 and is not within the scope of this work.

Conclusion

The algebraic Reynolds stress equations obtained on the basis of the proposed approximation are capable of properly dealing with complex turbulent flows such as the asymmetric turbulent channel flows. The application case shows that the NAEV model derived from the present algebraic Reynolds stresses equations can provide correct RS of the turbulent flows, which the SEV model and the ASM model based on Rodi's proposal fail to do.

References

[1]Lumley, J. J., "Toward a Turbulent Constitutive Relation," *Journal of Fluid Mechanics*, Vol. 41, 1970, pp. 413–434.

[2]Launder, B. E., Reece, G. J., and Rodi, W., "Progress in the Development of a Reynolds-Stress Turbulence Closure," *Journal of Fluid Mechanics*, Vol. 68, 1975, pp. 537–566.

[3]Rodi, W., "A New Algebraic Relation for Calculating the Reynolds Stresses," *ZAMM*, Vol. 56, No. 3, 1976, pp. T219–T221.

[4]Launder, B. E., "On Effect of a Gravitational Field on the Turbulent Transport of Heat and Momentum," *Journal of Fluid Mechanics*, Vol. 67, 1975, pp. 569–581.

[5]Gibson, M. M., and Launder, B. E., "On the Calculation of Horizontal, Turbulent Free Shear Flow Using Gravitational Influence," *Journal of Heat Transfer*, Vol. 98C, Feb. 1976, pp. 81–87.

[6]Pope, S. B., "A More General Effective-Viscosity Hypothesis," *Journal of Fluid Mechanics*, Vol. 72, Pt. 2, 1975, pp. 331–340.

[7]Yoshizawa, A., "Statistical Analysis of the Deviation of the Reynolds Stress from Its Eddy-Viscosity Representation," *Physics of Fluids*, Vol. 27, No. 6, 1984, pp. 1377–1387.

[8]Speziale, C. G., "On Nonlinear K–l and K–ε Models of Turbulence," *Journal of Fluid Mechanics*, Vol. 178, 1987, pp. 459–475.

[9]Rubinstein, R., and Barton, J. M., "Nonlinear Reynolds Stress Models and the Renormalization Group," *Physics of Fluids A*, Vol. 2, No. 8, 1990, pp. 1472–1476.

[10]Yakhot, V., and Orszag, S. A., "Renormalization Group Analysis of Turbulence. I. Basic Theory," *Journal of Scientific Computing*, Vol. 1, No. 1, 1986, pp. 3–19.

[11]Ni, W., Kawall, J. G., and Keffer, J. F., "Nonequilibrium Anisotropic Eddy-Viscosity/Diffusivity Turbulence Model for Heated Turbulent Flows," *AIAA Journal*, Vol. 33, No. 9, 1995, pp. 1738–1740.

[12]Hanjalic, K., and Launder, B. E., "Fully Developed Asymmetric Flow in a Plane Channel," *Journal of Fluid Mechanics*, Vol. 51, Pt. 2, 1972, pp. 301–335.

[13]Ni, W., "Derivation of a Non-Linear Reynolds Stress Equation from Its Differential Equation," *Proceedings of the 15th Canadian Congress of Applied Mechanics*, Univ. of Victoria, Victoria, BC, Canada, 1995, pp. 650–651.

R. M. C. So
Associate Editor

Octree-Based Implicit Agglomeration Multigrid Method

Dartzi Pan*
National Cheng-Kung University,
Tainan 70101, Taiwan, Republic of China
and
Maw-Jyi Chao†
Chung Shan Institute of Science and Technology,
Taichung 40701, Taiwan, Republic of China

Introduction

THE agglomeration multigrid method[1] has been shown to be an efficient convergence acceleration method for unstructured grid flow solvers. What is special about an agglomeration multigrid method is that its coarse-level cells are obtained by the direct fusing of neighboring cells of the previous finer mesh. This cell agglomeration procedure is usually accomplished by a weighted graph algorithm, as has been described in Ref. 1. The agglomerated coarse cells are polyhedral, with an arbitrary number of boundary faces, which may be extremely distorted if not properly controlled. In this work a new volume agglomeration scheme is developed to obtain better-quality coarse meshes. The new scheme takes the dual mesh of a tetrahedral mesh as its first-level coarse mesh, utilizing the

Received 27 July 1997; revision received 30 November 1998; accepted for publication 24 March 1999.

*Professor, Institute of Aeronautics and Astronautics. Senior Member AIAA.

†Assistant Researcher, Aeronautics Research Laboratory.

natural intergrid relationships between dual meshes. From the second coarse level and beyond, the fusion of mesh cells is guided by an octree system, generating high-quality coarse cells whose boundary faces roughly follow the edges of an octant.

Implicit Euler Multigrid Solver

The steady-state Euler flow solver developed in this work is a cell-centered, MUSCL-type upwind flux-differencing scheme based on Roe's approximate Riemann solver and Frink's interpolation.[2] Frink's interpolation requires cell-vertex variable values, which are obtained by a volume-weighted averaging of the surrounding cell-centered values. The Euler implicit time integration is used with only first-order spatial accuracy on the implicit side. The implicit operator is split into a diagonal part D, a lower triangular part L, and an upper triangular part U. The approximate LU factorization scheme (ALU) (Ref. 3) on the fine mesh can be written as

$$(D+L)D^{-1}(D+U)\Delta Q^{f,n} = R^f(Q^{f,n}) = \mathrm{Res}^{f,n}$$
$$Q^{f,n+1} = Q^{f,n} + \Delta Q^{f,n} \tag{1}$$

where the superscript f indicates fine grid, n indicates time level, Q is the vector of conserved variables, and $R^f(Q^{f,n})$ is the surface integral of flux functions that equals to the residual $\mathrm{Res}^{f,n}$ on the fine grid. On a multigrid coarse-level mesh the implicit ALU scheme can be written as

$$(D+L)D^{-1}(D+U)\Delta Q^{c,m} = R^c(Q^{c,m}) - R^c(J_f^c Q^f) + I_f^c\,\mathrm{Res}^{f,n} = \mathrm{Res}^{c,m}$$
$$Q^{c,m+1} = Q^{c,m} + \Delta Q^{c,m} \tag{2}$$

where the superscript c indicates coarse grid, m indicates the mth iteration on the coarse grid, and I_f^c and J_f^c are the restriction operators from the fine grid to the coarse grid for the residual and the conserved variables, respectively. A correction for the fine-grid variables can be done by

$$Q^{f,n+1} = Q^{f,n} + J_c^f\left(Q^{c,m} - J_f^c Q^{f,n}\right) \tag{3}$$

where J_c^f is the prolongation operator from the coarse grid to the fine grid. Only first-order spatial accuracy is used for both the implicit and the explicit operators on all levels of coarse grids.

Dual Mesh as a Coarse Mesh

In three dimensions the number of tetrahedral cells is about five to six times the number of the cell vertices. Thus, the dual grid is a natural coarse grid whose cell volume is about five or six times of that of the fine grid. By connecting the cell centroid, the cell-face centers and the midpoints of cell edges, a tetrahedron can be divided into four equal-volume cell quarters, each associated with one cell vertex. The dual cell associated with a cell vertex is the union of cell quarters surrounding that particular vertex. The cell faces of the dual cell are the union of triangular interfaces that border the cell quarters. Because only first-order spatial accuracy is used on the dual cell, each set of triangular interfaces bordering the same two dual cells can be replaced by a single surface vector that equals the sum of the area vectors in the set. On the boundary surfaces, however, we do not agglomerate the triangular interfaces. In fact, each surface triangle on the body surface is split into six subtriangles in the construction of the dual mesh. The identities of these subtriangles are kept unchanged, such that the actual body geometry can be fully preserved on all coarse levels, which is important to the success of the multigrid method when an important geometric peculiarity such as the trail edge is involved.

The flow solver on the dual mesh is still a cell-centered scheme that stores the Q variables at the tetrahedral vertices, which is consistent with the fine-grid solver because the vertex values have already been calculated in Frink's method. Another advantage of the dual mesh is that linear restriction and prolongation operators can be easily constructed. The restriction operator J_f^c for Q is a volume-weighted averaging such that the volume integral QV is conserved. The restriction operator I_f^c for the residual distributes one-fourth of the residual of a fine grid cell to each dual cell associated with its four vertices. The prolongation operator J_f^c is defined such that the correction for each cell-centered Q on the fine grid is the simple average of the corrections computed on the dual cells associated with its four vertices. This is equivalent to a linear interpolation, which is one order higher than a simple injection operator.

Octree-Based Volume Agglomeration

The generation of coarse meshes from the second level and beyond is guided by an octree system. Generating an octree system covering the entire flow domain with a desired octant distribution is easy. That is, those octants cutting the body surface are of the size of surface triangles, and those octants away from the body have their edge length progressively doubled as they move away from the body. Here we use the well-ordered octant sizes to guide the sequence of volume agglomeration. Note that there is a clear parent–children relationship between two consecutive levels of octant. Eight children-level octants can be fused together to form one parent-level octant. This octant agglomeration can be continued until the smallest octant in the domain has reached the desired size. Then the previous finer-grid cells whose centroid falls into one particular octant are agglomerated, generating coarse cells whose boundaries roughly follow the boundaries of the agglomerated octants. Thus, by controlling the sequence of octant agglomeration, one has a high degree of control over the shape and quality of the coarse-grid cells. This octree-based agglomeration scheme can be continued until ultimately there is only one coarse cell left in the final mesh. Note that octants with size greater than that of the agglomerated levels will not be agglomerated, which means that some cells may remain unchanged through several levels of mesh coarsening. The multigrid iterations performed on these unchanged cells are equivalent to iterations carried out on the previous finer mesh.

Similar to the dual grid, each set of triangular surfaces bordering on the same two agglomerated cells is treated as a single interface. The restriction operator I_f^c for residual is a summation such that the surface integral of fluxes is conserved. The restriction operator J_f^c for Q is a volume-weighted averaging such that the volume integral QV is conserved. The prolongation operator J_c^f for the corrections is a simple injection, that is, all finer grid cells inside the same agglomerated cell receive the same amount of corrections. This is equivalent to a zeroth-order interpolation, which is adequate for the Euler equations.

Test Example

All computations presented here are done on a Hewlett-Packard HP-9000/735 (99 Hz) with 176 MB of memory. To save computer resources, only single-precision computations are performed in this work. The language is FORTRAN 77. The Euler solver without multigrid spends 1×10^{-4} s per cell per iteration. Extensive parameter studies[4] have been performed to determine the best parameter set. The best operation in terms of CPU time is the four-level, V-cycle multigrid. The parameters used in this work are as follows. The Courant–Friedrichs–Lewy number based on the average local cell edge is 10,000. Two ALU iterations are performed on each coarse grid, while four ALU iterations (smoothing) are performed on the fine grid between two consecutive V-cycles. Note that one multigrid V-cycle or one fine-grid-alone computation will update the fine-grid variables once, which is counted as one Nstep. In a sense, one multigrid V-cycle can be viewed as a special time-integration method that marches the solution one step forward. One of the convergence indexes is the L_2 norm of the residual vector weighted by the cell volume.

The test example shown here is a transonic flow over a M6 wing. The freestream Mach number is 0.84, and the angle of attack is 3.06 deg. Figure 1 shows the sequence of eight meshes generated by the octree-based volume agglomeration scheme. The fine grid shown in Fig. 1a is generated by a hybrid octree/advancing front method described by Pan and Chao.[5] Figure 1b is the first coarse-level mesh (the dual grid), Fig. 1c the second coarse grid, and so on.

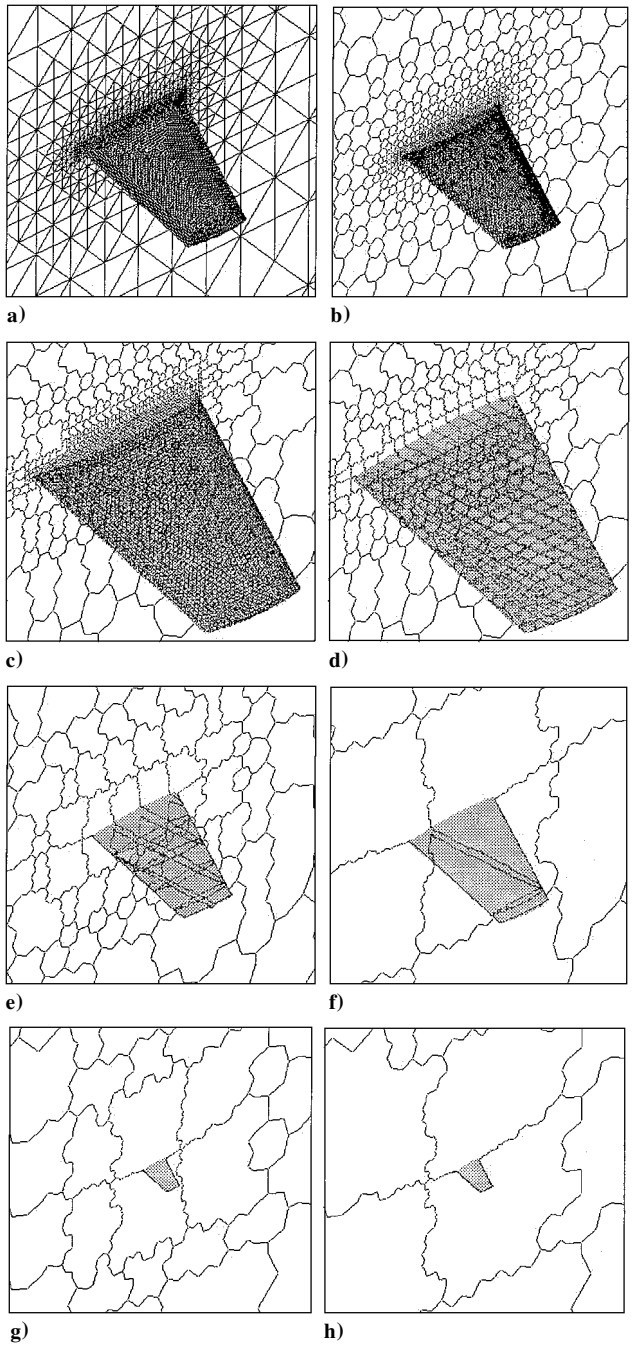

Fig. 1 Sequence of multigrid meshes for a M6 wing generated by the dual-mesh and octree-based volume agglomeration scheme.

The boundaries of the agglomerated cells roughly follow the octant boundaries. The four meshes used in the multigrid V-cycle are in Fig. 1a (236,148 cells), Fig. 1b (45,437 cells), Fig. 1c (16,659 cells), and Fig. 1d (2791 cells). Figures 2a–2d show, respectively, the convergence history of the L_2 norm of the weighted residual, the lift coefficient C_L, and the number of supersonic points N_{sup} in the flowfield for both multigrid computation and fine-grid-alone computation. Note that C_L and N_{sup} have reached their steady values at about 30,000 s. Depending on where the reference point is, the multigrid solution converges three to five times faster than the fine-grid-alone computation in terms of CPU time. Although not shown here because of space limitation, the computed pressure coefficients at various wing sections compare well with experiment.[5] The computed lift coefficient is 0.29011, which is close to the value of 0.2923 by Woodard et al.[6] and 0.2911 by Frink.[2] The computed moment coefficient is −0.1709, whereas it is −0.1717 by Woodard et al.[6] and −0.1726 by Frink.[2] Results similar to Figs. 1 and 2 for transonic flows over a Royal Aircraft Establishment wing-body combination

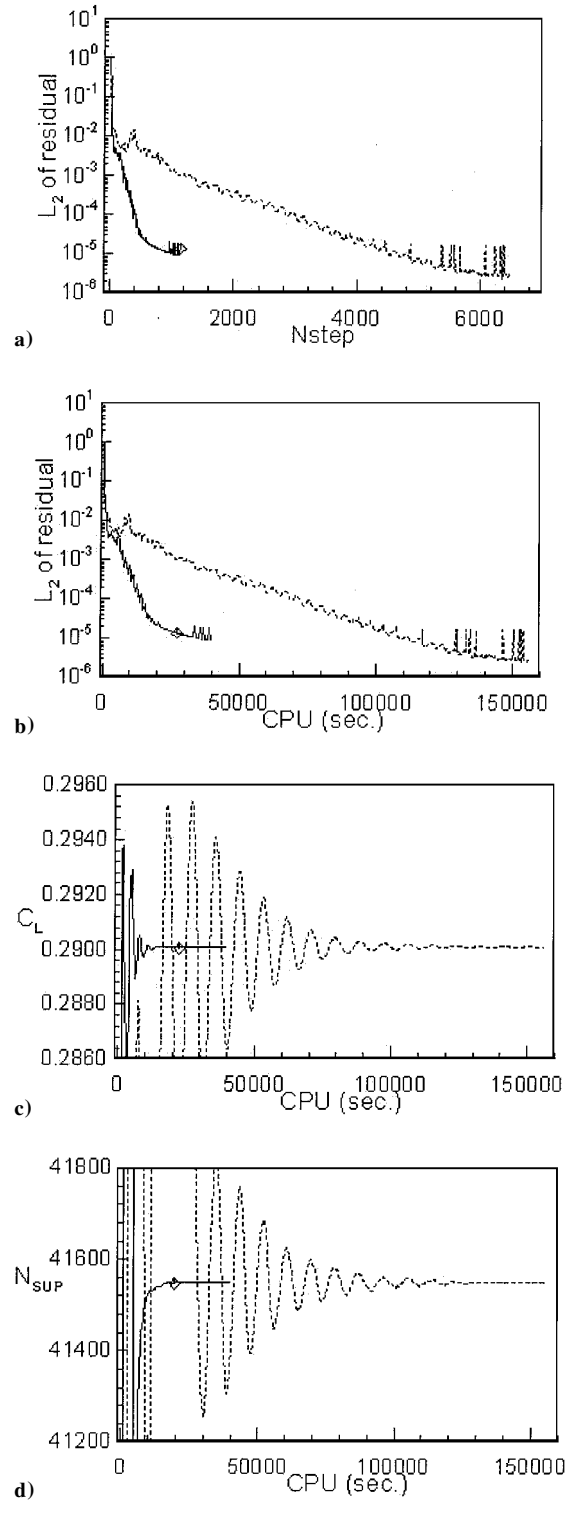

Fig. 2 Convergence histories of M6 wing computation: $M_\infty = 0.84$; $\alpha = 3.06$ deg; - - - -, single-grid calculation; and ——, four-level multigrid.

and an idealized fighter-stores configuration can be found in Ref. 5. The convergence histories shown in Fig. 2 are typical in these cases.

Conclusions

For the steady Euler equations the proposed implicit agglomeration multigrid method uses the naturally well-ordered intergrid relationship between a tetrahedral mesh and its dual mesh. The octree-based agglomeration scheme generates high-quality coarse cells with minimum effort. The proposed agglomeration multigrid method converges about three to five times faster than the single-grid computation in terms of CPU time.

Acknowledgments

This work was partially supported by the National Science Council of the Republic of China under Grants NSC87-2212-E006-093 and NSC87-2212-E006-094, which are highly appreciated.

References

[1]Mavriplis, D. J., "Multigrid Techniques for Unstructured Meshes," Inst. for Computer Applications in Science and Engineering, ICASE Rept. 95-27, Hampton, VA, April 1995.

[2]Frink, N. T., "Upwind Scheme for Solving the Euler Equations on Unstructured Tetrahedral Meshes," *AIAA Journal*, Vol. 30, No. 1, 1992, pp. 70–77.

[3]Pan, D., and Cheng, J. C., "Upwind Finite-Volume Navier–Stokes Computations on Unstructured Triangular Meshes," *AIAA Journal*, Vol. 31, No. 9, 1993, pp. 1618–1625.

[4]Pan, D., and Chao, M. J., "Implicit Unstructured Agglomeration Multigrid Method for the Euler Solutions over Complex Configurations," *Proceedings of the 4th Conference on Computational Fluid Dynamics*, Shi-Tau, Taiwan, ROC, 1997, pp. 269–274.

[5]Pan, D., and Chao, M. J., "Hybrid Octree/Advancing-Front Mesh Generation and Implicit Agglomeration Multigrid Euler Solver," AIAA Paper 99-0918, Jan. 1999.

[6]Woodard, P. R., Batina, J. T., and Yang, H. T., "Unstructured Mesh Quality Assessment and Upwind Euler Solution Algorithm Validation," *Journal of Aircraft*, Vol. 31, No. 3, 1994, pp. 643–650.

J. Kallinderis
Associate Editor

Numerical Tests of Upwind Scheme Performance for Entropy Condition

Ge-Cheng Zha*
General Electric Aircraft Engines, Lynn, Massachusetts 01910

I. Introduction

To treat the flows with shock waves and contact discontinuities, an accurate and efficient upwind scheme used as a Riemann solver to resolve the discontinuities is essential. Such an upwind scheme is particularly important when it is incorporated into a high-order accuracy scheme such as an essentially nonoscillatory scheme to simulate turbulence or acoustic fields with discontinuities, where the nonphysical noise should be minimized and the number of mesh points would be limited due to the computing power.

To achieve efficiency and accuracy, efforts have been made to develop an upwind scheme only using scalar dissipation instead of using matrix dissipation such as that of Roe's[1] flux difference splitting (FDS) scheme. The modification of Van Leer's[2] flux vector splitting (FVS) scheme by Hänel et al.[3] in 1987 began a series of new developments. The advection upstream splitting method (AUSM) suggested by Liou and Steffen[4] in 1993 has achieved high accuracy while the computing work remains as low as that of the Van Leer[2] scheme. Zha and Bilgen[5] suggested a low diffusion FVS scheme in 1993, which may resolve crisp shock profiles, but the contact discontinuity would be smeared. The Zha–Bilgen scheme has been modified to accurately resolve contact discontinuities in Ref. 6. Jameson[7–9] suggested his convective upwind and split pressure (CUSP) schemes and limiters in 1993, which may capture crisp shock profiles, but not contact discontinuities. Liou's[10,11] AUSM$^+$ scheme further improves the accuracy of the AUSM scheme and is able to resolve the exact shock and contact discontinuities and to preserve the constant total enthalpy for steady-state flows. Using methodology similar to the AUSM scheme, Edwards[12,13] developed his low diffusion flux splitting scheme (LDFSS). The scheme shows the best performance in the numerical tests of this paper. The splitting formulations of Mach number and pressure from Van Leer's FVS scheme are employed in both AUSM and LDFSS schemes.

Although progress has been made to reduce the dissipation of upwind schemes and to improve their capability to capture discontinuities, relatively less attention has been paid to answer an important question: Does the scheme satisfy the entropy condition? For FDS schemes such as Roe's[1] scheme and Osher's[14] scheme, it is known that the Roe scheme does not satisfy the entropy condition and the Osher scheme does. For FVS schemes such as the Van Leer[2] scheme and the Steger–Warming[15] scheme, it is not known whether they satisfy the entropy condition.[16,17] The extensive and successful applications of the FVS schemes in practical computations lead to a general impression that a FVS scheme, in particular, the Van Leer scheme, is very robust, and violation of the entropy condition may not be an issue. However, this cannot serve as the answer to the question just raised. For those recently developed schemes, including AUSM family schemes,[10,11] LDFFS schemes,[12,13] CUSP schemes, modified Zha–Bilgen scheme,[6] etc., no definite answers are given regarding whether they satisfy the entropy condition.

The purpose of this Technical Note is to create a numerical test case that may answer the entropy condition question for some of the upwind schemes. If an upwind scheme generates an expansion shock, the scheme would be considered as violating the entropy condition. If an upwind scheme does not generate an expansion shock for the test case, it is not conclusive that the scheme satisfies the entropy condition. To explore the original properties of the upwind schemes, the work in this paper mainly focuses on the one-dimensional piecewise constant first-order discretization.

II. Numerical Procedure

The governing equations are the quasi-one-dimensional Euler equations in Cartesian coordinates:

$$\partial_t \boldsymbol{U} + \partial_x \boldsymbol{E} - \boldsymbol{H} = 0 \tag{1}$$

where

$$\boldsymbol{U} = S\boldsymbol{Q}, \qquad \boldsymbol{Q} = \begin{bmatrix} \rho \\ \rho u \\ e \end{bmatrix}, \qquad \boldsymbol{E} = S\boldsymbol{F}$$

$$\boldsymbol{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ (e + p)u \end{bmatrix}, \qquad \boldsymbol{H} = \frac{\mathrm{d}S}{\mathrm{d}x}\begin{bmatrix} 0 \\ p \\ 0 \end{bmatrix}$$

In the preceding equations, ρ is the density, u is the velocity, p is the static pressure, e is the total energy, and S is the cross-sectional area of the one-dimensional duct. The following state equation is also employed:

$$p = (\gamma - 1)\left(e - \tfrac{1}{2}\rho u^2\right) \tag{2}$$

where γ is the ratio of specific heats and has the value 1.4.

The finite volume method with explicit Euler temporal integration is used to discretize the governing equations. It yields the following formulation at cell i:

$$\Delta \boldsymbol{Q}_i^{n+1} = \Delta t\left[-C\left(\boldsymbol{E}_{i+\frac{1}{2}} - \boldsymbol{E}_{i-\frac{1}{2}}\right) + (\boldsymbol{H}_i/S_i)\right]^n \tag{3}$$

where $C = 1/(\Delta x S_i)$ and n is the time level index.

The following upwind schemes are incorporated into this finite volume solver to evaluate the interface flux $\boldsymbol{F}_{i+1/2}$. They are the Van Leer[2] FVS scheme, the Roe[1] FDS scheme, the Van Leer–Hänel[3] scheme, the Steger–Warming[15] FVS scheme, the Liou[11] AUSM$^+$ scheme, the modified Zha–Bilgen scheme,[6] and the Edwards[12] LDFSS(2) scheme. In Ref. 6, the detailed formulations of each of the cited upwind schemes are listed for the purpose of reproducing the results.

Received 29 January 1998; revision received 1 November 1998; accepted for publication 29 March 1999.

*Lead Engineer, Department of Fan and Compressor Aero Design, Mail Drop 34044, 1000 Western Ave.; Gecheng.Zha@ae.ge.com. AIAA Member.