THE AMERICAN SOCIETY OF MECHANICAL ENGINEERS
345 E. 47 St., New York, N.Y. 10017

89-GT-45

Accelerated Computation of Viscous, Steady Incompressible Flows

SEUNGSOO LEE GEORGE S. DULIKRAVICH
Department of Aerospace Engineering
The Pennsylvania State University
University Park, PA 16802, U.S.A.

ABSTRACT

Based on an artificial compressibility method, the explicit Runge-Kutta time stepping finite difference algorithm was applied to steady, incompressible, Navier-Stokes equations. A two-dimensional analysis computer code in a generalized curvilinear coordinate system was developed and its accuracy has been compared to known numerical solutions. The algorithm has been accelerated using our new Distributed Minimal Residual (DMR) method, which allows each equation in the system to advance in time with its own optimal speed. The effectiveness of the DMR method was examined for a number of test cases. The accelerated algorithm offers substantial savings of the computing time.

INTRODUCTION

Unlike the algorithms for computing compressible viscous flows, computational methods for incompressible viscous flows have been developed relatively slowly. The main difficulty which arises during the incompressible flow computation comes from the lack of a time derivative term in the continuity equation.

One of the methods for overcoming this problem is the artificial compressibility concept originated by Chorin (1972). He solved incompressible Navier-Stokes equations using the explicit leap-frog/DuFort Frankel method. In this concept, an artificially time dependent derivative term $\frac{\partial(p/\beta)}{\partial t}$ is added to the continuity equation with a user specified control parameter β. Since the artificial time derivative added to the continuity equation diminishes as the solution converges, the added term can be chosen so as to achieve the maximum convergence rate. The added term forces the system to be of a mixed parabolic-hyperbolic type, which allows the use of the time marching techniques. Later, Kwak et al. (1986) and Choi and Merkle (1985) and Merkle and Tsai (1986) used the Approximate Factorization (AF) method in conjunction with the artificial compressibility method.

In this paper, the explicit Runge-Kutta time stepping method of Jameson (1981) is used as a basic algorithm and the DMR (Distributed Minimal Residual) method developed by Lee et al. (1989) and Dulikravich et al. (1988) is incorporated in the artificial compressibility code in order to maximize the convergence rate. The DMR method belongs to a general class of the extrapolation techniques in which the solution is updated using information from a number of consecutive time steps in such a way that the L_2 norm of future residual is minimized. Unlike in Minimal Residual Method of Hafez (1985) and the General Nonlinear Minimal Residual Method of Huang and Dulikravich (1987), each component of the solution vector is updated using a separate sequence of acceleration factors. The idea of using different acceleration factors for each component of a solution vector achieves similar effects as dynamic preconditioning.

The objective of this paper is to present the DMR method as applied to the artificial compressibility method, and to demonstrate the advantages of the DMR method with a number of computational examples.

Presented at the Gas Turbine and Aeroengine Congress and Exposition—June 4-8, 1989—Toronto, Ontario, Canada

NAVIER-STOKES EQUATIONS AND THE NUMERICAL ALGORITHM

The two dimensional Navier-Stokes equations expressed in the general curvilinear nonorthogonal coordinates ξ,η can be written as

$$\frac{\partial Q}{\partial t} = -\frac{\partial E}{\partial \xi} - \frac{\partial F}{\partial \eta} + D^2(JQ) - D(JQ) \equiv -\mathbf{R}^t, \text{ the residual} \qquad (1)$$

where p, u, v, U,and V are the pressure, Cartesian velocity components along x and y direction, and contravariant velocity components normal to constant ξ and η lines, respectively. The general solution vector and flux vectors are defined as

$$Q = \frac{1}{J}\begin{bmatrix} 0 \\ u \\ v \end{bmatrix} \quad E = \frac{1}{J}\begin{bmatrix} U \\ Uu+\xi_x p \\ Uv+\xi_y p \end{bmatrix} \quad F = \frac{1}{J}\begin{bmatrix} V \\ Vu+\eta_x p \\ Vv+\eta_y p \end{bmatrix} \qquad (2)$$

where the subscripts designate partial derivatives and J is the Jacobian $J = \det(\partial(\xi,\eta)/\partial(x,y))$. The physically dissipative viscous term in the general coordinates is given by

$$D^2(JQ) = \left[\frac{S}{J} g_{ij} (JQ)_j\right]_i \qquad (3)$$

where g_{ij} is the contravariant geometric transformation matrix tensor

$$g_{ij} = \begin{bmatrix} \nabla\xi\nabla\xi & \nabla\xi\nabla\eta \\ \nabla\eta\nabla\xi & \nabla\eta\nabla\eta \end{bmatrix} \qquad (4)$$

and $S=\frac{1}{Re}\text{diag}(0, 1, 1)$, and Re is the Reynolds number. D(JQ) is an artificial dissipation term to be described later. Since the solution vector Q does not have the first component, it is impossible to solve the system simultaneously. This difficulty can be resolved by, say, adding $\frac{\partial(p/\beta)}{\partial t}$ to the continuity equation (1), that is, $Q=(p/\beta, u, v)^*$. Subscript * designates transpose of a vector or a matrix. Because the time derivatives vanish as the steady state is approached, the added term does not alter the final steady solution. According to the eigenvalue analysis for the system, the Jacobian matrices $\mathbf{A} = \frac{\partial \mathbf{E}}{\partial \mathbf{Q}}$ and $\mathbf{B} = \frac{\partial \mathbf{F}}{\partial \mathbf{Q}}$ have real eigenvalues,

$$\mathbf{A} = \frac{\partial \mathbf{E}}{\partial \mathbf{Q}} = \mathbf{K}(U,\xi_x,\xi_y)$$
$$\mathbf{B} = \frac{\partial \mathbf{F}}{\partial \mathbf{Q}} = \mathbf{K}(V,\eta_x,\eta_y) \qquad (5)$$

where

$$\mathbf{K}(k,k_1,k_2) = \begin{bmatrix} 0 & k_1 & k_2 \\ \beta k_1 & k+k_1 u & k_2 u \\ \beta k_2 & k_1 v & k+k_2 v \end{bmatrix} \qquad (6)$$

and

$$k = k_1 u + k_2 v \qquad (7)$$

The eigenvalues of the matrix **K** are given by

$$\Lambda = \mathbf{diag}(k - c, k + c, k) \qquad (8)$$

where the equivalent speed of sound, c, is given as

$$c = \sqrt{k^2 + \beta(k_1^2 + k_2^2)} \qquad (9)$$

Notice that one of the eigenvalues is negative. This means that the incompressible flow is equivalently "subsonic" in the sense of different signs of the eigenvalues.

SPATIAL DISCRETIZATION AND THE RUNGE-KUTTA TIME STEPPING

The residual of the Navier-Stokes equations, Eq. (1), is discretized by the second order central difference approximation. For example, the first derivative is approximated as

$$\frac{\partial E}{\partial \xi} = \frac{E_{i+1j} - E_{i-1j}}{2\Delta\xi} \qquad (10)$$

The second derivative is discretized as

$$\frac{\partial}{\partial\xi}\left\{\frac{S}{J} g_{11} \frac{\partial (JQ)}{\partial \xi}\right\}$$
$$= \frac{1}{\Delta\xi^2}\left[\left\{\frac{S}{J} g_{11}\right\}_{i+1/2j}\left\{(JQ)_{i+1j} - (JQ)_{ij}\right\}\right]$$
$$- \frac{1}{\Delta\xi^2}\left[\left\{\frac{S}{J} g_{11}\right\}_{i-1/2j}\left\{(JQ)_{ij} - (JQ)_{i-1j}\right\}\right] \qquad (11)$$

The mixed derivative is discretized as

$$\frac{\partial}{\partial\xi}\left\{\frac{S}{J} g_{12} \frac{\partial (JQ)}{\partial \eta}\right\}$$
$$= \frac{1}{4\Delta\xi\Delta\eta}\left\{\frac{S}{J} g_{12}\right\}_{i+1/2j}\left\{(JQ)_{i+1j+1} - (JQ)_{i+1j-1} + (JQ)_{ij+1} - (JQ)_{ij-1}\right\}$$
$$- \frac{1}{4\Delta\xi\Delta\eta}\left\{\frac{S}{J} g_{12}\right\}_{i-1/2j}\left\{(JQ)_{ij+1} - (JQ)_{ij-1} + (JQ)_{i-1j+1} - (JQ)_{i-1j-1}\right\} \qquad (12)$$

where $\Delta\xi = \Delta\eta = 1$ = constant.

After spatial derivative terms are discretized, the governing set of partial differential equations becomes a set of ordinary differential equations, which can be integrated by the Runge-Kutta time-stepping algorithm. The Runge-Kutta algorithm can be written as

$$Q^0 = Q^t$$
$$\Delta Q^k = -\alpha_k \Delta t R^{k-1} \quad k = 1,2,...,K \qquad (13)$$
$$Q^{t+1} = Q^t + \Delta Q^K$$

where α_k are the coefficients for each of the K stages of the Runge-Kutta scheme required to advance the solution from the time level t to the time level t+1. For example, α_k = 1/4, 1/3, 1/2 and 1.0 for the four-stage Runge-Kutta scheme. To reduce the computational effort, we calculate the viscous part of the residual only once every global time level and keep it unchanged during the four steps of the Runge-Kutta scheme. This does not deteriorate the stability of the time stepping algorithm.

STABILITY AND THE ARTIFICIAL DISSIPATION

The allowable time increments of the explicit scheme are severely limited by the stability conditions. We will follow the suggestions of MacCormack and Baldwin (1975) and define the time step by considering the hyperbolic part of the system and the parabolic part of the system separately and by combining these time steps. The system becomes hyperbolic when viscosity is neglected. Then, the stability bound of the resulting system is determined by the CFL (Courant-Friedrichs-Lewy) number. The maximum time steps for each of the two coordinate directions are defined as

$$\Delta t_{h\xi} = \frac{CFL}{|U| + c_\xi} \qquad \Delta t_{h\eta} = \frac{CFL}{|V| + c_\eta} \tag{14}$$

and the combined maximum time step for the hyperbolic part of the system is defined by

$$\Delta t_h = \frac{\Delta t_{h\xi}\Delta t_{h\eta}}{\Delta t_{h\xi} + \Delta t_{h\eta}} \tag{15}$$

When the convective part of the acceleration is neglected, the system becomes parabolic. The stability of the parabolic type svstem is dictated by the non-dimensional von Neumann number, σ. For each direction, the maximum time steps are defined as

$$\Delta t_{p\xi} = \frac{\sigma}{g_{11}Re} \qquad \Delta t_{p\eta} = \frac{\sigma}{g_{22}Re} \tag{16}$$

and the combined maximum time step for the parabolic part is given by

$$\Delta t_p = \frac{\Delta t_{p\xi}\Delta t_{p\eta}}{\Delta t_{p\xi} + \Delta t_{p\eta}} \tag{17}$$

The total maximum time step is estimated conservatively as

$$\Delta t = \frac{\Delta t_h \Delta t_p}{\Delta t_h + \Delta t_p} \tag{18}$$

It is known that the solution obtained from the central difference approximation tends to decouple at even and odd numbered grid point, resulting in numerical oscillations. These oscillations can be removed by adding a small amount of artificial dissipation to the residual. We use the fourth order artificial dissipation suggested by Steger and Kutler (1977)

$$D(JQ) = \frac{\varepsilon}{8J\Delta t}\nabla^4[JQ] \tag{19}$$

where ∇^4 denotes the fourth order elliptic operator and ε is the control parameter. The artificial dissipation term was kept unchanged during the the four steps of the Runge-Kutta scheme.

BOUNDARY CONDITIONS

At higher Reynolds numbers, the viscous effects are dominant only in the vicinity of the solid boundary. Therefore, for the purpose of enforcing boundary conditions, we assume that the flow is inviscid at the inlet and the exit plane. After neglecting the viscous terms, the system of equations becomes hyperbolic in time near the inlet and exit. For the compatibility condition, therefore, we have to consider the direction of the characteristics. As stated earlier, the incompressible flow has two positive eigenvalues and one negative eigenvalue. Thus one equation of motion should be considered with two boundary conditions at the inlet, while at the exit two equations with one boundary condition must be applied. At the inlet we specify U and V/U , while the back pressure p is specified at the exit. Also, the flow is assumed to be locally one-dimensional at the inlet and exit boundaries in order to transform the equation into the characteristic form. The similarity transform matrix for the two dimensional case can be derived from the Jacobian matrix **A** in the general coordinates, and is given by

$$\mathbf{M}_\xi^{-1} = \begin{bmatrix} -\frac{U+c_\xi}{2c_\xi^2} & \frac{\xi_x}{2c_\xi^2} & \frac{\xi_y}{2c_\xi^2} \\ -\frac{U-c_\xi}{2c_\xi^2} & \frac{\xi_x}{2c_\xi^2} & \frac{\xi_y}{2c_\xi^2} \\ \frac{(U^2-c_\xi^2)v+\beta U\xi_y}{c_\xi^2} & -\frac{\xi_x(Uv+\beta\xi_y)}{c_\xi^2} & 1-\frac{\xi_y(Uv+\beta\xi_y)}{c_\xi^2} \end{bmatrix} \tag{20}$$

Premultiplying Eq. (13-b) by $\mathbf{M}_\xi^{-1}$(in ξ direction) results in the characteristic form of the equations. At the inlet, the equation corresponding to the negative eigenvalue should be selected, while at the exit the equations corresponding to positive eigenvalues are chosen. This selection procedure can be thought of as a matrix operation, and we designate the operator as **L**. If the boundary condition is given by Ω, then

$$\Omega^{t+1} = \Omega^t + \frac{\partial\Omega}{\partial Q}\Delta Q$$

or

$$\frac{\partial\Omega}{\partial Q}\Delta Q = 0 \tag{21}$$

and Eq. (21) is added to the transformed-selected equations, that is,

$$\left[\mathbf{L}\mathbf{M}_{\xi}^{-1} + \frac{\partial \Omega}{\partial \mathbf{Q}}\right]\Delta\mathbf{Q} = -\alpha_k \Delta t \mathbf{M}_{\xi}^{-1}\mathbf{R} \qquad (22)$$

At the solid wall, the contravariant velocities U and V are set to zero, and the surface pressure is extrapolated from that of the adjacent grid point. Artificial dissipation and its normal derivative at the boundaries are set to zero.

DISTRIBUTED MINIMAL RESIDUAL (DMR) METHOD

The local residual at time level t+1 is given by

$$\mathbf{R}^{t+1} = \frac{\partial \mathbf{E}^{t+1}}{\partial \xi} + \frac{\partial \mathbf{F}^{t+1}}{\partial \eta} - D^2(\mathbf{J}\mathbf{Q}^{t+1}) + \mathrm{D}(\mathbf{J}\mathbf{Q}^{t+1}) \qquad (23)$$

Assume that the solution at time level t+1 is extrapolated from the previous M consecutive time levels. Then, we can say that

$$\mathbf{Q}^{t+1} = \mathbf{Q}^{t} + \sum_{m=1}^{M} \Theta^{m} \qquad (24)$$

where

$$\Theta^{m} = \begin{bmatrix} \omega_1^m \Delta_1^m \\ \omega_2^m \Delta_2^m \\ . \\ . \\ . \\ \omega_L^m \Delta_L^m \end{bmatrix} \qquad (25)$$

Here, ω's are the acceleration factors to be calculated, Δ's are the corrections computed with the original scheme after each of the consecutive time steps, M denotes the total number of consecutive time steps combined, and L denotes the total number of equations in the system. For example, L = 3 for the incompressible, two-dimensional flow problem accounting for one mass conservation equation and two momentum equations.

Using Taylor series expansion in time and neglecting the terms that are higher than first order in Δt, Eq. (23) becomes approximately

$$\mathbf{R}^{t+1} = \mathbf{R}^{t} + \sum_{m=1}^{M} \left[\frac{\partial}{\partial \xi}\mathbf{A}^{t}\cdot + \frac{\partial}{\partial \eta}\mathbf{B}^{t}\cdot - D^2\mathrm{J}\cdot + \mathrm{DJ}\cdot\right]\Theta^{m} \qquad (26)$$

The global residual for the entire domain can be defined as

$$\mathrm{R}^{t} = \sum_{\mathrm{D}} \mathbf{R}^{t^*}\,\mathbf{R}^{t} \qquad (27)$$

where $\sum_{\mathrm{D}}$ denotes summation over the computational domain, and $(.)^*$ represents transpose of a vector. In order to minimize the R^{t+1}, we can determine the ω's from the following system of algebraic equations

$$\frac{\partial \mathrm{R}^{t+1}}{\partial \omega_r^m} = 0 \qquad (28)$$

that is,

$$-\sum_{\mathrm{D}} \mathbf{R}^{t^*}\left[\frac{\partial}{\partial \xi}\mathbf{A}^{t}\cdot + \frac{\partial}{\partial \eta}\mathbf{B}^{t}\cdot - D^2\mathrm{J}\cdot + \mathrm{DJ}\cdot\right]\frac{\partial \Theta^{m}}{\partial \omega_r^{m}}$$

$$= \sum_{\mathrm{D}}\sum_{n}^{M}\left\{\left[\frac{\partial}{\partial \xi}\mathbf{A}^{t}\cdot + \frac{\partial}{\partial \eta}\mathbf{B}^{t}\cdot - D^2\mathrm{J}\cdot + \mathrm{DJ}\cdot\right]\Theta^{n}\right\}^{*} \cdot\left\{\left[\frac{\partial}{\partial \xi}\mathbf{A}^{t}\cdot + \frac{\partial}{\partial \eta}\mathbf{B}^{t}\cdot - D^2\mathrm{J}\cdot + \mathrm{DJ}\cdot\right]\frac{\partial \Theta^{m}}{\partial \omega_r^{m}}\right\} \qquad (29)$$

where

$$\frac{\partial \Theta^{m}}{\partial \omega_r^{m}} = \left\{\Delta_p^m \delta_{pr}\right\} \qquad (30)$$

and δ_{pr} is the Kronecker delta. Notice that from Eq. (25),

$$\Theta^{n} = \sum_{q}^{L} \omega_q^n \frac{\partial \Theta^{n}}{\partial \omega_q^{n}} \qquad (31)$$

Also, notice that $\frac{\partial \Theta^{n}}{\partial \omega_q^{n}}$ is not a function of ω.

Let

$$\mathbf{a}_q^m = \left[\frac{\partial}{\partial \xi}\mathbf{A}^{t}\cdot + \frac{\partial}{\partial \eta}\mathbf{B}^{t}\cdot - D^2\mathrm{J}\cdot + \mathrm{DJ}\cdot\right]\frac{\partial \Theta^{m}}{\partial \omega_q^{m}} \qquad (32)$$

Then Eq. (29) becomes

$$-\sum_{\mathrm{D}} \mathbf{R}^{t^*}\mathbf{a}_r^m = \sum_{n}^{M}\sum_{q}^{L}\sum_{\mathrm{D}} \omega_q^n \mathbf{a}_q^{n^*}\mathbf{a}_r^m \qquad (33)$$

For simplicity, let

$$c_{qr}^{nm} = \sum_{\mathrm{D}} \mathbf{a}_q^{n^*}\mathbf{a}_r^m \qquad (34)$$

and

$$b_r^m = -\sum_{\mathrm{D}} \mathbf{R}^{t^*}\mathbf{a}_r^m \qquad (35)$$

Then Eq. (35) can be written as

$$\sum_{n}^{M}\sum_{q}^{L} \omega_q^n c_{qr}^{nm} = b_r^m \qquad (36)$$

representing the system of M×L linear algebraic equations for the L sets of M optimum acceleration factors ω. For example, if we are to combine M = 2 consecutive time steps to extrapolate the solution and to solve the two-dimensional incompressible Navier-Stokes equations (L = 3), we need to solve 6 equations for 6 values of ω.

Using a different sequence of acceleration factors for each component of the solution vector is equivalent to using a different time step for each equation, which allows acceleration of the scheme. This does not mean that the system is decoupled since all ω's are implicitly interrelated.

COMPUTATIONAL RESULTS AND DISCUSSION

In all computational examples, the value of β (artificial compressibility parameter) was $\beta = 1.0$ in order to test the effects of the DMR method alone.

The two-dimensional code was tested on a classical example of a driven cavity where the upper wall is steadily moving. A mildly clustered grid of 45×45 points was used with Re=400, CFL = 2.8 and $\sigma = 1.0$. The convergence history (Fig. 1 and 2) indicates that the $\log_{10}(\text{Res})$, where $\text{Res} = \frac{R^t}{LN}$ and N is the number of grid points, is reduced by four orders of magnitude after 2500 iterations with the DMR method. For the basic algorithm, however, the logarithm of Res was reduced by only three orders of magnitude after 2500 iterations . Figure 3 depicts the computed values of u component of velocity along a vertical line passing through the center of the cavity, and the computed values of v component of velocity along the horizontal line passing through the center of the cavity, showing excellent agreement with the published results of Athavale and Merkle (1988). From the computed velocity vector field (Fig. 4) and the isobar lines (Fig. 5), we can see the large vortex core.

The code was then applied to a flow through a non-staggered cascade of NACA0012 airfoils with the specified inlet flow angle of 0°. An H-type clustered grid consisting of 40×30 grid points (Fig. 6) was used with Re = 500, CFL = 2.8, $\sigma = 1.4$. A smooth convergence history was obtained (Fig. 7 and Fig. 8) and the computing time was reduced by 70% when using the DMR method. The computed velocity vector field (Fig. 9) shows well developed boundary layers and the wake. Figure 10 shows the computed field of isobars.

Finally, flow through a cascade of NACA0012 airfoils with inlet flow angle of 10° was computed (Fig. 11) with the same conditions as in the second test case. The convergence rates with and without the DMR are compared in Fig. 12 and 13 in terms of the number of iterations and CPU time indicating that the DMR method reduces computing time by 35%. Figure 14 shows the wake zone behind the trailing edge of the airfoil. In Fig. 15, the computed isobar contours are plotted.

In all examples, we combined only two (M=2) consecutive time steps when using the DMR method which we applied after every thirty steps performed by the basic algorithm

CONCLUSIONS

The Distributed Minimal Residual (DMR) method was applied to the acceleration of the explicit Runge-Kutta method for the incompressible Navier-Stokes equations. The artificial compressibility method was used to make the incompressible Navier-Stokes equations of mixed parabolic-hyperbolic type. The DMR method uses a separate sequence of optimal acceleration factors for each of the equations in the system, which allows the acceleration of the scheme. The DMR method offers between 35% and 70% reduction in computing time when integrating incompressible Navier-Stokes equations on clustered and nonorthogonal grids. The new acceleration method is stable and does not seem to depend on the Reynolds number. It requires approximately double computer storage as compared to the non-accelerated explicit method. In comparison, the GMRES conjugate gradient method of Wighton et al. (1985) needs between twenty and eighty consecutive solutions to be stored. The DMR method can also be applied to implicit algorithms, as our preliminary findings confirm with the Beam-Warming (1976) type ADI algorithm for incompressible Navier-Stokes equations.

ACKNOWLEDGMENTS

This work was partially supported by the grant AFOSR-87-0121 supervised by Lt. Col. Dr. David Nelson and Maj. Dr. John Thomas. The authors are grateful for the computing equipment donated by the APPLE computer, Inc.

REFERENCES

Athavale, M. and Merkle, C. L., 1988,"An Upwind Differencing Scheme for Time-Accurate Solutions of Unsteady Incompressible Flow," AIAA Paper 88-3650-CP, Cincinnati, OH, July.

Beam, R. M. and Warming, R. F., 1976,"Implicit Finite-Difference Procedures for Hyperbolic Systems in Conservation-Law Form," *Journal of Computational Physics*, Vol. 22, September, pp. 87-110.

Choi, D. and Merkle, C. L., 1985," Application of Time-Iterative Schemes to Incompressible Flow," AIAA Journal, Vol. 23, No.10, October, pp. 1518-1524.

Chorin, A. J., 1967,"A Numerical Method for Solving Incompressible Viscous Flow Problems," *Journal of Computational Physics*, Vol. 2, pp. 12-26.

Dulikravich, G. S., Dorney, D. J. and Lee, S., 1988,"Iterative Acceleration and Physically Based Dissipation for Euler Equations

of Gasdynamics," Symposium on Advances and Applications in Computational Fluid Dynamics, ASME Winter Annual Meeting, 'hicago, IL, Nov. 28-Dec. 2.

Hafez, M., Parlette, E., and Salas, M. D.,"Convergence Acceleration of Iterative Solutions of Euler Equations for Transonic Flow Computations," AIAA Paper 85-1641, July.

Huang, C. Y. and Dulikravich, G. S., 1985,"Fast Iterative Algorithms Based on Optimized Explicit Time-Stepping," *Computer Methods in Applied Mechanics and Engineering*, 63, August, pp. 15-36.

Jameson, A., 1981,"Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Scheme," AIAA Paper 81-1259, AIAA 14th Fluid and Plasma Dynamics Conference, Palo Alto, CA, Jun. 23-25.

Kwak, D., Chang, J. L. C., Shanks, S. P. and Chakravarthy, S. R., 1986,"An Incompressible Navier-Stokes Flow Solver in Three Dimensional Curvilinear Coordinate System Using Primitive Variables," *AIAA Journal*, Vol. 24, No. 3, March, pp. 390-396.

Lee, S., Dulikravich, G. S. and Dorney, D. J., 1989,"Acceleration of Iterative Algorithms for Euler Equations of Gasdynamics," AIAA Paper 89-0097, presented at the AIAA Aerospace Sciences Meeting, Reno, NV, Jan.9-12; Also, accepted for publication in *AIAA Journal.*

MacCormack, R. W. and Baldwin, B. S., 1975,"A Numerical Method for Solving the Navier-Stokes Equations With Application to Shock-Boundary Layer Interaction," AIAA Paper 75-1, January.

Merkle, C. L. and Tsai, Y.-L., 1986,"Application of Runge-Kutta Schemes to Incompressible Flows," AIAA paper 86-0553, Reno, NV, January.

Steger, J. L. and Kutler, P., 1977, "Implicit Finite-Difference Procedure for the Computation of Vortex Wakes," *AIAA Journal*, Vol. 15, No. 7, July, pp. 581-590.

Wighton, L., Yu, J. N. and Young. D., 1985,"GMRES Acceleration of Computational Fluid Dynamics Codes," AIAA paper 85-1494, Cincinnati, OH, June.

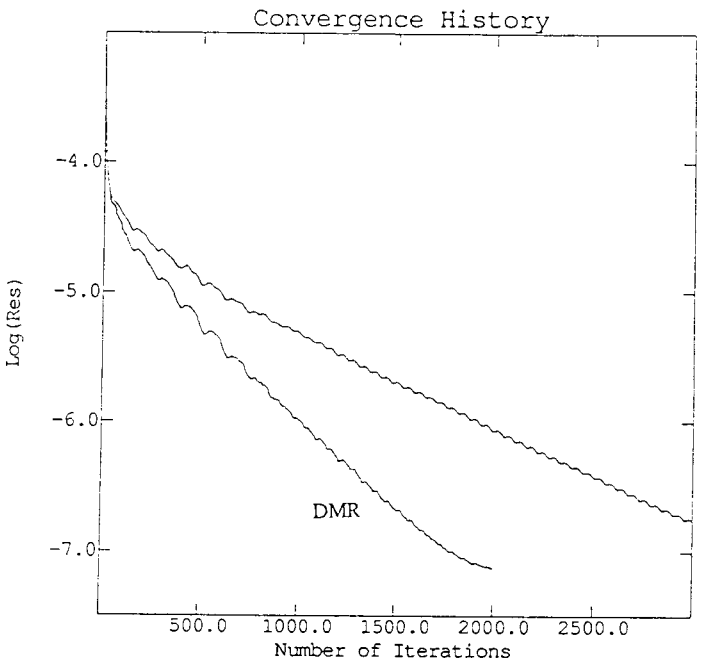

Fig. 1 Residual vs. number of iterations for the driven cavity

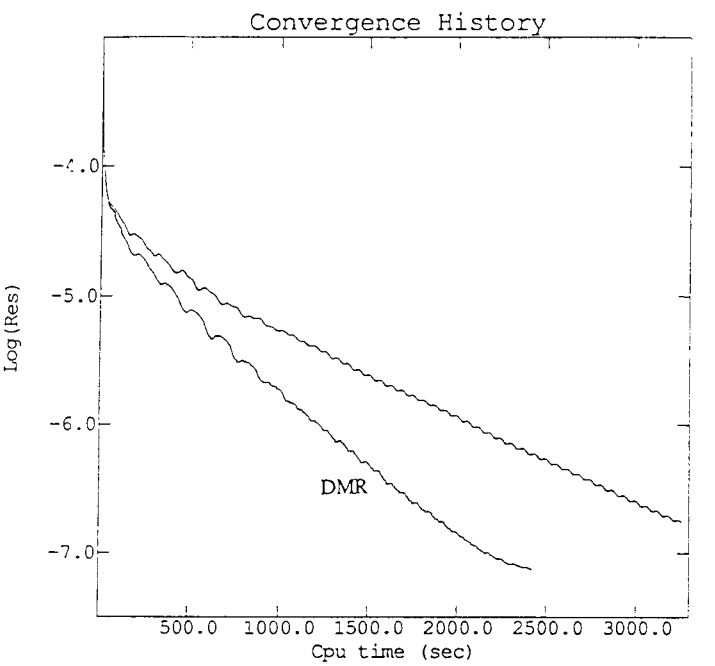

Fig. 2 Residual vs. CPU time for the driven cavity

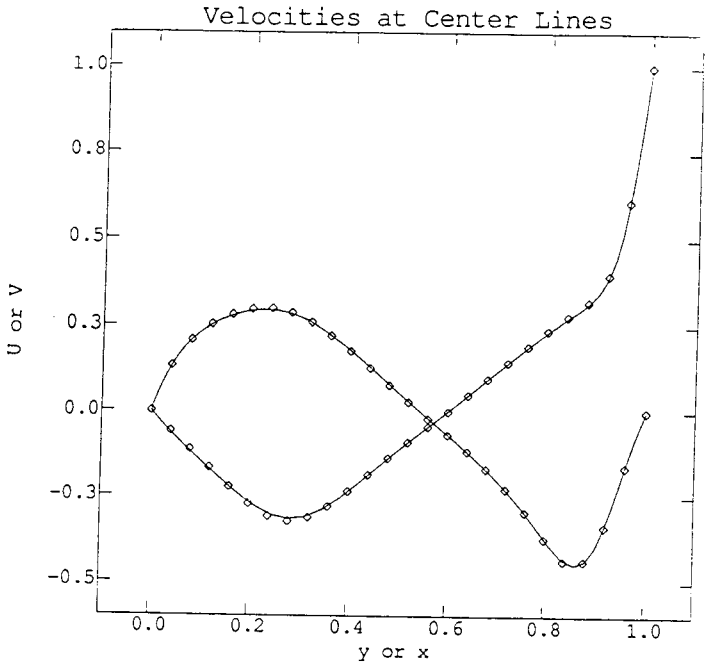

Fig. 3 Velocities at the center lines of the driven cavity (◇◇◇) Athavale et al., (——) present method

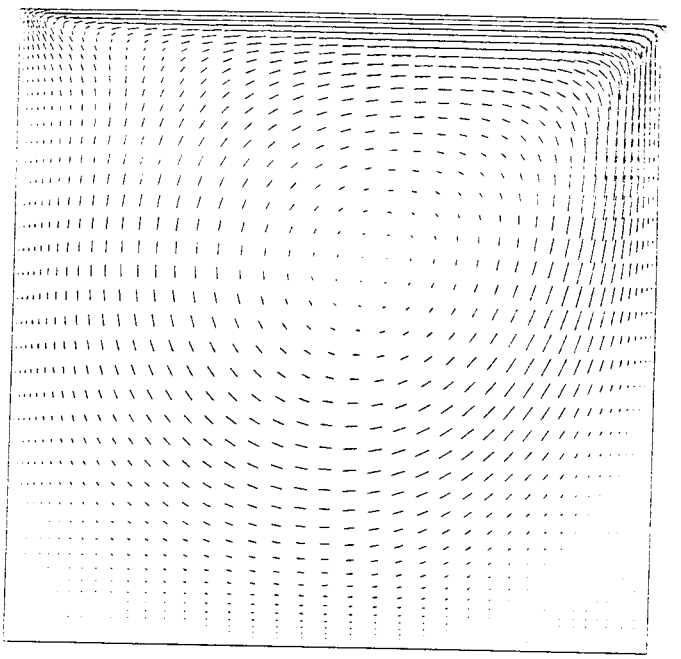

Fig. 4 Velocity profiles of the driven cavity

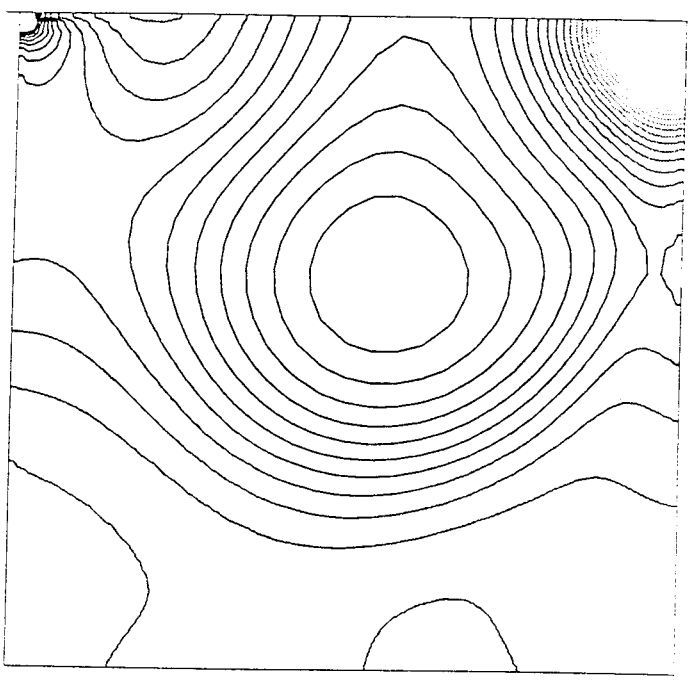

Fig. 5 Isobar contour lines of the driven cavity : with DMR

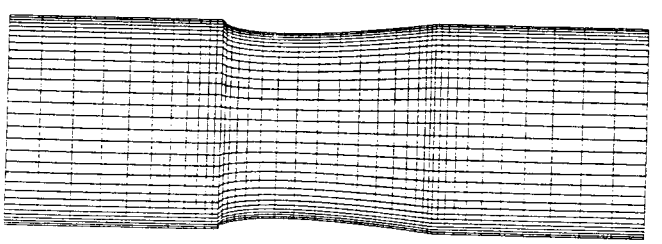

Fig. 6 Computational grid for the cascade of NACA0012 airfoil with flow angle 0°

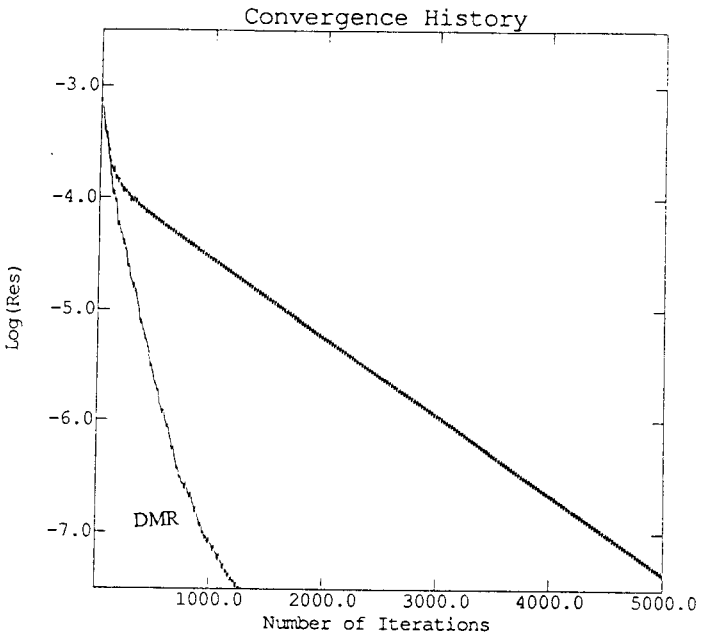

Fig. 7 Residual vs. number of iterations for the cascade

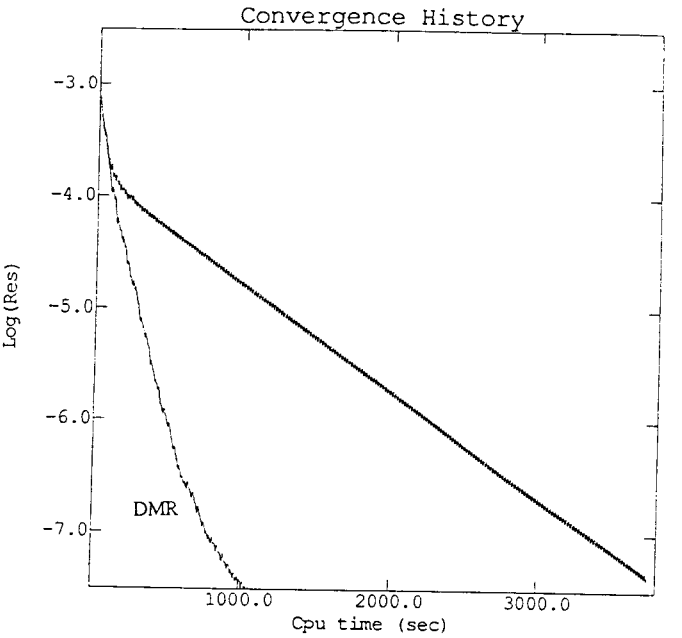

Fig. 8 Residual vs. CPU time for the cascade

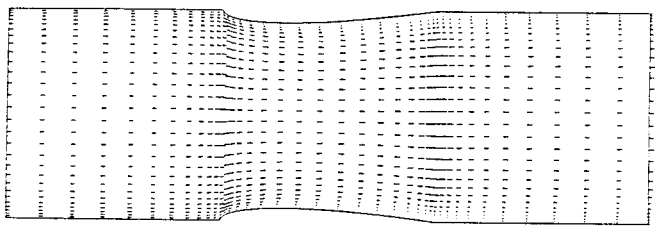

Fig. 9 Velocity profiles of the cascade

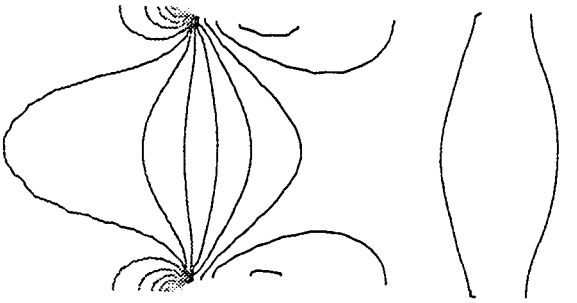

Fig. 10 Isobar contour lines of the cascade

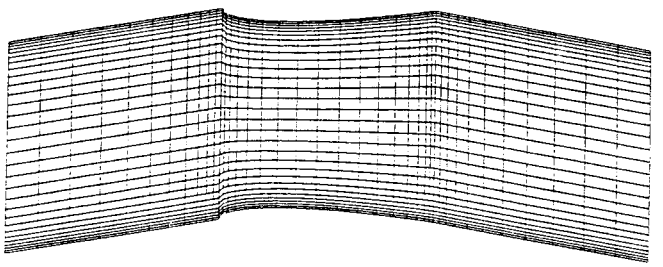

Fig. 11 Computational grid for the cascade of NACA0012 airfoil with flow angle 10°

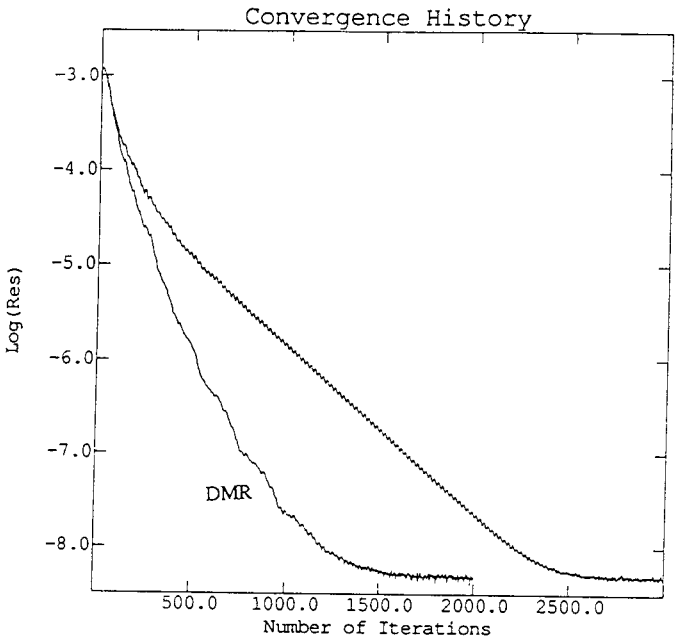

Fig. 12 Residual vs. number of iterations for the cascade

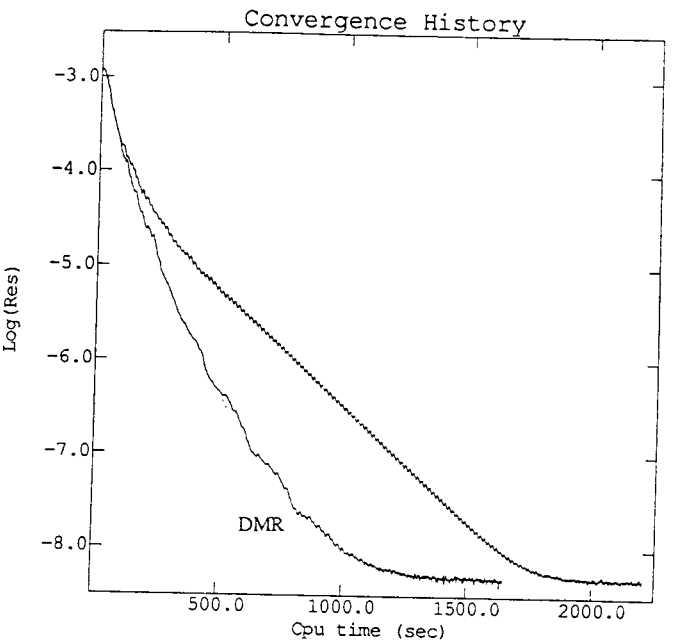

Fig. 13 Residual vs. CPU time for the cascade

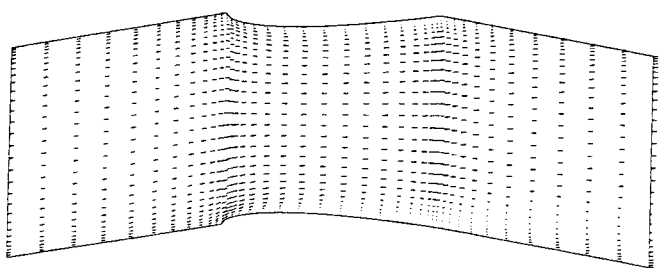

Fig. 14 Velocity profiles of the cascade

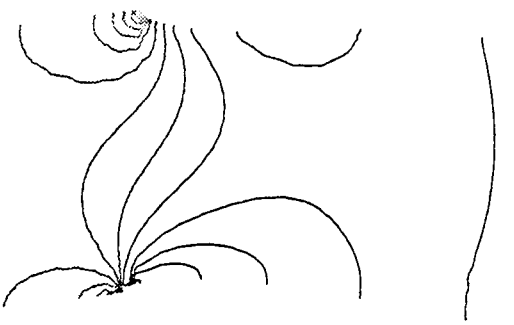

Fig. 15 Isobar contour lines of the cascade