AIAA JOURNAL
Vol. 32, No. 12, December 1994

Technical Notes

TECHNICAL NOTES are short manuscripts describing new developments or important results of a preliminary nature. These Notes cannot exceed 6 manuscript pages and 3 figures; a page of text may be substituted for a figure and vice versa. After informal review by the editors, they may be published within a few months of the date of receipt. Style requirements are the same as for regular contributions (see inside back cover).

Parallel Processing for Implicit Solutions of the Navier–Stokes Equations

S. Abdallah*
University of Cincinnati, Cincinnati, Ohio 45221

Nomenclature

bc$x$, bc$y$	= boundary conditions in x- and y-directions, respectively
P	= static pressure divided by density
R	= right-hand side of Eq. (3)
Re	= Reynolds number
t	= time
(u, v)	= velocity components in (x, y) directions
(x, y)	= Cartesian coordinates
$(\Delta x, \Delta y, \Delta t)$	= grid spacing in (x, y, t) directions
δ	= discrete operator for space derivatives
σ	= pressure derivatives in x and y directions
ϕ	= velocity components u and v

Subscripts

x, xx	= first- and second-order derivatives, respectively
y, t	= first-order derivative with respect to y and t, respectively

Superscripts

$n, n+1, *$	= t, $t + \Delta t$, and an intermediate time level, respectively

Introduction

THE advantages of using explicit vs implicit schemes for solutions of the Navier–Stokes equations are vectorization and parallel processing. However, severe limitations on time stepping of the explicit schemes[1,2] make them impractical for high Reynolds number calculations. Relaxation of these constraints on the time increment for explicit schemes have been achieved by using residual smoothing and/or multistepping techniques.[3] On the other hand, advancement of the implicit procedures have led to powerful factorization techniques. For example, the Beam and Warming scheme[4] factorizes a two- (three-) dimensional problem into two (three) one-dimensional solutions. Unfortunately, these one-dimensional solutions are not independent of each other and must be computed in series. Thus, parallel processing can not be used with these type of schemes.

In the present study, we propose an implicit scheme which employs superposition of two (three) one-dimensional solutions in two (three) dimensions for the solution of Navier–Stokes equations. Because of superposition, the one-dimensional solutions can be computed simultaneously on separate processors and, therefore, parallel processing can be used to enhance computations of large problems.[5–7] One of the difficulties in dealing with the superposition of several solutions is the determination of proper boundary conditions for the different solutions. In the present procedure, the boundary conditions are implemented in the discrete governing equations before the superposition step takes place. Therefore, no boundary conditions are needed for the one-dimensional solutions. Details of the method and numerical confirmation of the analytical developments are given next.

Received April 12, 1994; revision received June 27, 1994; accepted for publication June 29, 1994.

*Professor, Department of Aerospace Engineering and Engineering Mechanics. Member AIAA.

Analysis

For simplicity and clarity, the analysis is performed for two-dimensional, semi-implicit, and noncoupled Navier–Stokes equations. However, the analysis as well as the conclusions obtained will be directly applicable for three dimensions, implicit, and coupled Navier–Stokes equations without serious modifications.

The Navier–Stokes equations for incompressible flow, for example, are

$$u_x + v_y = 0 \tag{1}$$

$$\phi_t + u\phi_x + v\phi_y = -\sigma + (1/Re)(\phi_{xx} + \phi_{yy}) \tag{2}$$

$$\phi = \begin{pmatrix} u \\ v \end{pmatrix} \tag{2a}$$

$$\sigma = \begin{pmatrix} P_x \\ P_y \end{pmatrix} \tag{2b}$$

where (u, v) are the velocity components in (x, y) directions, P is the static pressure (assumed to be known) divided by the density.

Consider the semi-implicit solution of Eq. (2)

$$\left(1 - \frac{\Delta t}{Re}\delta_{xx} - \frac{\Delta t}{Re}\delta_{yy}\right)\phi^{n+1} = R^n \tag{3}$$

$$R = -(u\delta_x\phi + v\delta_y\phi + \sigma)\Delta t + \phi \tag{3a}$$

where the superscripts n and $n+1$ refer to the time levels t and $t + \Delta t$, respectively. The operators δ_x are δ_{xx} the discrete form of the first- and the second-order derivatives, respectively:

$$\delta_x\phi^{n+1} = \frac{1}{2\Delta x}\left(\phi^{n+1}_{i+1,j} - \phi^{n+1}_{i-1,j}\right) \tag{3b}$$

$$\delta_{xx}\phi^{n+1} = \frac{1}{\Delta x^2}\left(\phi^{n+1}_{i-1,j} - 2\phi^{n+1}_{i,j} + \phi^{n+1}_{i+1,j}\right) \tag{3c}$$

Equation (3) can be rewritten as

$$\left(1 - \frac{\Delta t}{Re}\delta_{xx}\right)\phi^{n+1} - \left(\frac{\Delta t}{Re}\delta_{yy}\right)\phi^{n+1} = R^n \tag{4}$$

Furthermore, the second term in Eq. (4) can be modified by subtracting the second-order term $\Delta t^2\delta_{xx}\delta_{yy}\phi^{n+1}/Re^2$ which does not affect the formal first-order time accuracy of Eq. (3):

$$\frac{\Delta t}{Re}\delta_{yy}\phi^{n+1} = \left(1 - \frac{\Delta t}{Re}\delta_{xx}\right)\left(\frac{\Delta t}{Re}\delta_{yy}\right)\phi^{n+1} + 0(\Delta t^2) \tag{4a}$$

By substituting Eq. (4a) into Eq. (4), one obtains

$$\left(1 - \frac{\Delta t}{Re}\delta_{xx}\right)\left(1 - \frac{\Delta t}{Re}\delta_{yy}\right)\phi^{n+1} = R^n \tag{5}$$

A two-step procedure for solving Eq. (5) is developed by Beam and Warming[4] as follows:

$$\left(1 - \frac{\Delta t}{Re}\delta_{xx}\right)\phi^* = R^n \tag{6}$$

$$\left(1 - \frac{\Delta t}{Re}\delta_{yy}\right)\phi^{n+1} = \phi^* \tag{6a}$$

Solutions for Eqs. (6) and (6a) require boundary conditions on ϕ^{n+1} and ϕ^*. No boundary conditions are readily available for ϕ^*. This difficulty has been resolved by employing the approximate boundary condition[4]

$$\phi^* = \phi^{n+1} \tag{7}$$

We stress here that the two-step approximate factorization scheme [Eqs. (6) and (6a)] requires the calculation of the intermediate variable ϕ^* before the second step can be executed. Thus, parallel processing is not possible for solving these equations.

In the present study, we propose the use of superposition of two one-dimensional solutions for the solution of Eq. (3). We also propose that the boundary conditions on ϕ be incorporated into Eq. (3) before employing the superposition procedure. In this case, Eq. (3) is rewritten as

$$\left(1 - \frac{\Delta t}{Re}\delta_{xx} - \frac{\Delta t}{Re}\delta_{yy}\right)\phi^{n+1} = R^n + \mathrm{bc}x + \mathrm{bc}y \tag{8}$$

Note that the discrete operators δ_{xx} and δ_{yy} are modified at grid points next to the boundaries to incorporate boundary conditions. For example, consider the use of Neumann boundary conditions in the x-direction at the boundary $i=1$, $\partial\phi/\partial x = g$, or in finite-difference form

$$\phi_{1,j}^{n+1} = \Delta x g + \phi_{2,j}^{n+1} \tag{8a}$$

At grid point $i = 2$ next to the boundary, the second term in the left-hand side of Eq. (8) becomes

$$\frac{\Delta t}{Re}\delta_{xx}\phi^{n+1} = \frac{\Delta t}{Re\Delta x^2}\left(\phi_{1,j}^{n+1} - 2\phi_{2,j}^{n+1} + \phi_{3,j}^{n+1}\right) \tag{8b}$$

By substituting Eq. (8a) in Eq. (8b), the term $(\Delta t/Re\Delta x^2)(\Delta x g)$ may be moved to the right-hand side of Eq. (8) and is represented by the symbol bc$x$. The second term in Eq. (8a) modifies the coefficient of $\phi_{2,j}^{n+1}$. Thus, the left-hand side of Eq. (8) contains only unknowns at interior grid points.

We propose the following solution for Eq. (8):

$$\phi^{n+1} = \phi_1^{n+1} + \phi_2^{n+1} - R^n \tag{9}$$

$$\left(1 - \frac{\Delta t}{Re}\delta_{xx}\right)\phi_1^{n+1} = R^n + \mathrm{bc}x \tag{9a}$$

$$\left(1 - \frac{\Delta t}{Re}\delta_{yy}\right)\phi_2^{n+1} = R^n + \mathrm{bc}y \tag{9b}$$

The proof that Eq. (9) is a solution for Eq. (8) can be obtained by substituting Eq. (9) into the left-hand side of Eq. (8) and showing it to be equal to the right-hand side by use of the approximation used in Eq. (4a). It is clear from the discussion leading to Eq. (8a), that no boundary conditions on the one-dimensional solutions of Eqs. (9a) and (9b) are needed.

It can be easily shown that the present scheme has the same order of time accuracy as the Beam and Warming scheme.[4] In other words, the splitting error of Eqs. (9) is of order Δt^2 as is the splitting error of Eq. (5). However, the present scheme employs superposition of two one-dimensional solutions which are independent of each other. These two solutions, Eqs. (9a) and (9b) can be computed simultaneously on two separate processors. This is a major advantage over the splitting procedure of Eq. (5). The use of parallel processing here is not only limited to using one processor for the solution of ϕ_1^{n+1} or ϕ_2^{n+1}. We can employ, for each equation, as many processors as $(M + N - 4)$ where M and N are the maximum number of grid lines in the x and y directions, respectively. This is because Eq. (9a) can be solved for several y locations simultaneously using separate processors (maximum $N - 2$ processors). Similarly, Eq. (9b) can be solved for several x locations simultaneously using separate processors (maximum $M - 2$ processors). For a three-dimensional problem with $(M \times N \times L)$ grid lines we can employ, for each equation, as many processors as $(M + N + L - 6)$.

Applications in Three Dimensions

Equations (9), (9a), and (9b) can be extended to three-dimensional situations as follows:

$$\phi^{n+1} = \phi_1^{n+1} + \phi_2^{n+1} + \phi_3^{n+1} - 2R^n \tag{10}$$

$$\left(1 - \frac{\Delta t}{Re}\delta_{xx}\right)\phi_1^{n+1} = R^n + \mathrm{bc}x \tag{10a}$$

$$\left(1 - \frac{\Delta t}{Re}\delta_{yy}\right)\phi_2^{n+1} = R^n + \mathrm{bc}y \tag{10b}$$

$$\left(1 - \frac{\Delta t}{Re}\delta_{zz}\right)\phi_3^{n+1} = R^n + \mathrm{bc}z \tag{10c}$$

Using the Eqs. (10) and the procedure outlined for two dimensions, one can easily show that Eqs. (10) satisfy the three-dimensional form of Eq. (3).

Applications to the Linearized Implicit Navier–Stokes Equations

For linearized implicit procedures, one writes the discrete form of Eq. (2) as follows:

$$\left(1 + \Delta t u^n\delta_x - \frac{\Delta t}{Re}\delta_{xx} + \Delta t v^n\delta_y - \frac{\Delta t}{Re}\delta_{yy}\right)\phi^{n+1} = \bar{R}^n \tag{11}$$

$$\bar{R} = -\sigma\Delta t + \phi \tag{11a}$$

Similarly, a solution which satisfies Eq. (11) is

$$\phi^{n+1} = \phi_1^{n+1} + \phi_2^{n+1} - \bar{R}^n \tag{12}$$

$$\left(1 + \Delta t u^n\delta_x - \frac{\Delta t}{Re}\delta_{xx}\right)\phi_1^{n+1} = \bar{R}^n + \mathrm{bc}x \tag{12a}$$

$$\left(1 + \Delta t v^n\delta_y - \frac{\Delta t}{Re}\delta_{yy}\right)\phi_2^{n+1} = \bar{R}^n + \mathrm{bc}y \tag{12b}$$

Again, Eqs. (12) can be easily shown, using the same procedure outlined for Eqs. (9), to be a solution for Eq. (11).

Finally, applications for the coupled system of the compressible Navier–Stokes equations can be modeled after Beam and Warming scheme[4] without major changes.

Numerical Results

The solution to a particular problem requires the calculation of the velocity and pressure fields which are obtained using the primitive variables formulation given in Refs. 8 and 9. In this formulation, the momentum equation (2) is solved for the velocity components u and v by marching in time using the semi-implicit scheme Eq. (9). In this case, the pressure is assumed known from the solution at the previous time level or initial guess. Then the pressure is calculated from a Poisson equation which is derived from the divergence of the momentum equation (2) and enforces the continuity equation (1). Numerical solutions for the pressure Poisson equation are obtained using the successive over-relaxation method.

The two-dimensional lid-driven square cavity is used here as a test case. The cavity upper wall moves with steady speed $u = 1$; at Reynolds number, $Re = 400$. Numerical results are obtained on (81×81) uniform grid points in the x and y directions, respectively.

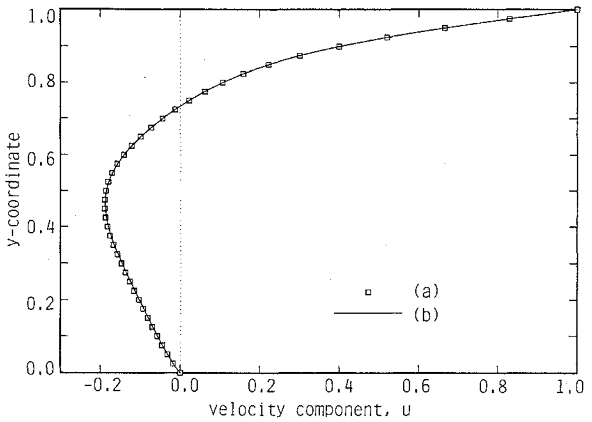

Fig. 1 Horizontal velocity component along the vertical centerline at *Re* = 400: a) present scheme and b) Beam and Warming scheme.

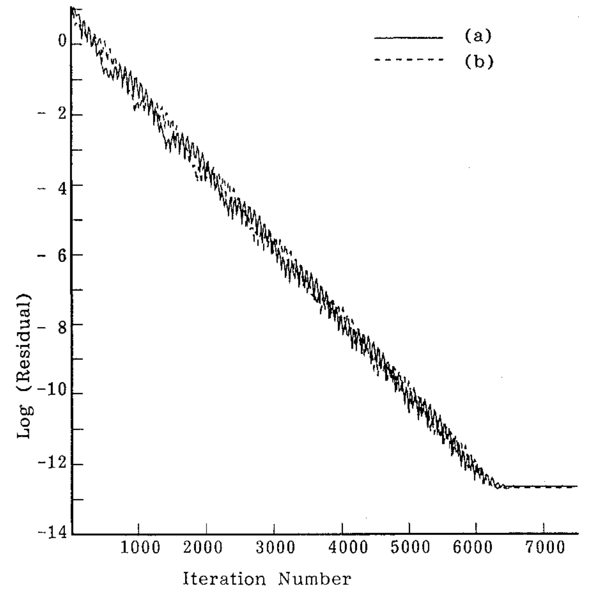

Fig. 2 Convergence history at *Re* = 400: a) present scheme and b) Beam and Warming scheme.

Comparisons of the computed results with numerical results obtained by the Beam and Warming scheme[4] are shown in Fig. 1. The convergence histories for our scheme and the Beam and Warming scheme are also shown in Fig. 2. These comparisons of the computed velocity and the convergence histories confirm the analytical developments of our scheme.

Conclusions

An implicit scheme is developed for the solution of the Navier–Stokes equations (compressible and incompressible). This scheme is based upon the superposition of several one-dimensional solutions which are independent of one another. Therefore, these one-dimensional solutions can be computed simultaneously on separate processors (parallel processing). Also, separate processors may be employed for each coordinate direction. The maximum number of processors which can be employed in each coordinate direction is limited by the maximum number of grid points in each direction. For example, consider a two-dimensional problem with $(M \times N)$ grid lines, the maximum number of processors that can be employed is $k(M + N - 4)$ processors where k is the number of equations. In addition, the present scheme has the advantage that no boundary conditions are needed for the one-dimensional solutions. This is because the boundary conditions on the physical velocity field are implemented in the governing equations before the superposition step takes place. The computed results for the driven cavity problem confirm the analytical developments of the present scheme.

Acknowledgment

This work was sponsored by the Office of Naval Research.

References

[1]Anderson, D. A., Tannehill, J. C., and Pletcher, R. H., *Computational Fluid Mechanics and Heat Transfer*, McGraw-Hill, 1984, pp. 88–97.

[2]Roache, P. J., *Computational Fluid Dynamics*, Hermosa, Albuquerque, NM, 1976, pp. 36–75.

[3]Jameson, A.,"Numerical Solution of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes," AIAA Paper 81-1259, June 1981.

[4]Beam, R., and Warming, R. F., "An Implicit Factored Scheme for the Compressible Navier–Stokes Equations," *AIAA Journal*, Vol. 16, No. 4, 1978, p. 393.

[5]Sotiropoulos, F., and Abdallah, S., "A Primitive Variable Method for the Solution of Three-Dimensional Incompressible Viscous Flows," *Journal of Computational Physics*, Vol. 103, No. 2, 1992, pp. 336–349.

[6]Larson, L., Patel, V. C., and Gilbert, D. (eds.), *Ship Viscous Flow*, Proceedings of 1990 SSPA Maritime Consulting AB—Chalmers University of Technology—Iowa Institute of Hydraulic Research Workshop, Flowtech International AB, Gothenburg, Sweden, 1990, pp. 8–16.

[7]Steger, J. L., "Implicit Finite Difference Simulation of Flow About Arbitrary Two-Dimensional Geometries," *AIAA Journal*, Vol. 16, No. 7, 1978, pp. 679–686.

[8]Abdallah, S., "Numerical Solutions for the Pressure Poisson Equation with Neumann Boundary Conditions Using a Non-Staggered Grid, I" *Journal of Computational Physics*, Vol. 70, No. 1, 1987, pp. 182–192.

[9]Abdallah, S., "Numerical Solutions for the Incompressible Navier–Stokes Equations in Primitive Variables Using a Non-Staggered Grid, II," *Journal of Computational Physics*, Vol. 70, No. 1, 1987, pp. 193–203.

Effects of Spatial Order of Accuracy on the Computation of Vortical Flowfields

J. A. Ekaterinaris*
Navy-NASA Joint Institute of Aeronautics, Moffett Field, California 94035

Introduction

ACCURATE prediction of subsonic, vortical flows over swept wings, slender bodies, and slender body/delta wing combinations at high incidence is of interest to aerodynamicists. The main characteristic of the flow over a swept wing at high incidence is the leading-edge vortex. This vortex is formed by the windward and leeward-side boundary layers which separate at the leading-edge and roll up in a helical fashion. For many practical applications, such as flows over a full aircraft or a canard-wing configuration, the leading-edge vortices generated by a slender wing convect downstream. It is, therefore, important for numerical methods to predict correctly the strength of these vortices and convect them downstream with minimal diffusion, because their presence significantly affects the development of the downstream flowfield.

Numerical methods currently used, such as second-order central difference and upwind schemes, require high grid resolution close to the surface to predict the viscous flow region accurately. In addition, adequate grid density is required in the vortical-flow region to resolve the complex flowfield features. Numerical simulations for flows over delta wings (cf. Refs. 1–4) showed that the lifting characteristics and surface pressure distributions are predicted accurately

Received June 20, 1993; presented as Paper 93-3371 at the AIAA 11th Computational Fluid Dynamics Conference, Orlando, FL, July 6–9, 1993; revision received June 5, 1994; accepted for publication June 6, 1994.

*Research Associate Professor, Naval Postgraduate School, Department of Aeronautics and Astronautics. Senior Member AIAA.