JOURNAL OF COMPUTATIONAL PHYSICS 103, 336–349 (1992)

A Primitive Variable Method for the Solution of Three-Dimensional Incompressible Viscous Flows

F. SOTIROPOULOS* AND S. ABDALLAH

Department of Aerospace Engineering and Engineering Mechanics, University of Cincinnati, Cincinnati, Ohio 45221

Received October 30, 1990; revised May 24, 1991

In this paper we present a new primitive variable method for the solution of the three-dimensional, incompressible, Reynolds averaged Navier–Stokes equations in generalized curvilinear coordinates. The governing equations are discretized on a non-staggered grid and the discrete continuity equation is replaced by a discrete pressure-Poisson equation. The discrete pressure equation is designed in such a way that: (i) the compatibility condition for the Poisson–Neumann problem is automatically satisfied, and (ii) the discrete incompresibility constraint is satisfied to, at least, truncation error accuracy while the computed pressure is smooth. The momentum equations are integrated in time using the four-stage Runge–Kutta algorithm while the pressure equation is solved using the point-successive relaxation technique. The method is applied to calculate the turbulent flow field over a ship model. The computed results are in very good agreement with the experimental data.

INTRODUCTION

There are two major difficulties associated with the numerical solution of the incompressible Navier–Stokes equations, in primitive form, on non-staggered grids: (i) the *existence* of a *smooth* solution for the pressure field, and (ii) the satisfaction of the *discrete continuity* equation to machine zero. It is well known, for instance, that on a non-staggered mesh the Neumann boundary conditions for the pressure should be implemented very carefully in order to satisfy the integral compatibility constraint and guarantee the existence of a unique solution for the pressure [1–3]. Moreover, an attempt to satisfy the discrete continuity equation to machine zero may lead to non-smooth solutions for the pressure (odd–even decoupling) [1, 4], while the smoothness of the pressure field can be ensured at the expense of the discrete incompressibility constraint [1–5].

On the other hand, staggered grid primitive variable formulations—which have been extensively pursued by a large number of researchers [6–10]—do not suffer, in general, from the difficulties which are inherent in the use of non-staggered computational meshes. However, when generalized curvilinear coordinates are to be employed, the use of a staggered mesh increases considerably the storage requirements. This is because the metrics of the geometric transformation need to be computed and stored at the center (pressure nodes) as well as at the interfaces (velocity nodes) of each computational cell. This issue becomes particularly important in three-dimensional, high Reynolds number applications where large number of grid nodes are required for the accurate resolution of the flow field. Another difficulty, associated with the staggered mesh in curvilinear coordinates, arises from the fact that the discretization of the continuity equation requires the contravariant velocity components at the interfaces of the computational cell (velocity nodes). This problem can be overcome at the expense of either: (i) storage and complexity, by using contravariant coordinates [9], or (ii) accuracy, by employing physical coordinates (cartesian, cylindrical, etc.) and interpolate to complete the calculation of the contravariant velocity components at the cell interfaces [10, 11]. Finally, in generalized coordinates the use of a staggered mesh does not by default guarantee the smoothness of the computed pressure field. As discussed in Ref. [11], odd–even pressure oscillations may occur if the grid is highly clustered and special treatment of the discrete pressure operator is required in order to eliminate them.

The above discussion suggests that the non-staggered mesh has several advantages, particularly in three-dimensional generalized coordinate systems, since it can reduce storage requirements and simplify the programming of a numerical method [12]. However, choosing to work with a non-staggered mesh requires a very careful treatment of the numerics in order to alleviate the difficulties associated with it (already discussed) and take advantage of its benefits. In Ref. [1] we addressed the difficulties related to the use of the pressure-Poisson approach on a non-staggered grid and we proposed an optimum way to derive a discrete pressure equation. The optimum—among the conventional non-staggered pressure-Poisson formulations [2, 3, 5]—discrete pressure equation [1] was designed to: (i) satisfy automatically the compatibility constraint, and (ii) minimize

* Present address: Institute of Hydraulic Research, The University of Iowa, Iowa City, Iowa 52242.

0021-9991/92 $5.00

the error in the discrete continuity equation with the computed pressure being smooth. However, our analysis was restricted to uniform, cartesian grids.

The main objective of the present paper is to use the ideas of Ref. [1] to develop a method for the solution of the incompressible Navier–Stokes equations in generalized curvilinear coordinates. The discrete continuity is satisfied, through the solution of a discrete pressure-Poisson equation, up to an artificial source term. This artificial source term is at least second order (truncation error of the method) and its size is the minimum required to guarantee the smoothness of the computed pressure field. The momentum equations are integrated to steady state using the four stage explicit Runge–Kutta scheme [13]—enhanced with local time stepping and implicit residual smoothing [14]—while the pressure equation is solved using the successive point relaxation method. The Runge–Kutta scheme has been widely used for solving compressible [13, 15] as well as incompressible (in conjunction with the artificial compressibility method) [16] flowfields in two and three dimensions. Although explicit, it is a scheme that is competitive with implicit approximate factorization techniques (especially in three dimensions) because: (i) it can be easily vectorized and parallel processed, and (ii) the implementation of implicit residual smoothing allows the use of CFL numbers (Courant, Friedrich, and Lewis) similar to those used with multidimensional implicit schemes [16]. In the present study, the Runge–Kutta scheme is used for the first time in conjunction with the pressure-Poisson approach.

To demonstrate the efficiency and the ability of the method to produce a smooth pressure field and a "practically" divergence-free velocity field in generalized curvilinear coordinates, we apply it to calculate the three-dimensional, turbulent flow field around a ship model (Wigley hull). The computed results are in very good agreement with the experimental data.

GOVERNING EQUATIONS IN GENERALIZED CURVILINEAR COORDINATES

The three-dimensional, time dependent, incompressible Reynolds averaged Navier–Stokes equations are written in cylindrical polar coordinates $(x, r, r\theta)$ and then transformed to body fitted, generalized curvilinear coordinates (ξ, η, ζ). The Jacobian J and the contravariant metric tensor $\{g^{ij}\}$ of the geometric transformation $(x, r, r\theta) \to (\xi, \eta, \zeta)$ are defined as

$$J = \frac{\partial(\xi, \eta, \zeta)}{\partial(x, r, r\theta)} = \begin{bmatrix} \xi_x & \xi_r & \dfrac{\xi_\theta}{r} \\ \eta_x & \eta_r & \dfrac{\eta_\theta}{r} \\ \zeta_x & \zeta_r & \dfrac{\zeta_\theta}{r} \end{bmatrix} \tag{1.1}$$

$$g^{ij} = \xi^i_x \xi^j_x + \xi^i_r \xi^j_r + \frac{1}{r^2} \xi^i_\theta \xi^j_\theta \quad \text{for } i, j = 1, 2, 3, \tag{1.2}$$

where $(\xi^1, \xi^2, \xi^3) \to (\xi, \eta, \zeta)$. The transformed governing equations, continuity and momentum, read as follows:

Continuity equation,

$$J\left[\frac{\partial}{\partial \xi}\left(\frac{U}{J}\right) + \frac{\partial}{\partial \eta}\left(\frac{V}{J}\right) + \frac{\partial}{\partial \zeta}\left(\frac{W}{J}\right)\right] = 0 \tag{2}$$

Momentum equation,

$$\frac{\partial Q}{\partial t} + A\frac{\partial Q}{\partial \xi} + B\frac{\partial Q}{\partial \eta} + C\frac{\partial Q}{\partial \zeta} + H - J\left[\frac{\partial E_{v1}}{\partial \xi} + \frac{\partial E_{v2}}{\partial \eta} + \frac{\partial E_{v3}}{\partial \zeta}\right] - H_v = 0. \tag{3}$$

In the above equation, Q is the velocity vector,

$$Q = (u, v, w)^{\mathrm{T}} \tag{3.1}$$

where u, v, and w are the axial, radial, and tangential velocity components, respectively. The matrices A, B, and C, in Eq. (3), are diagonal matrices defined as

$$\begin{aligned} A &= \mathrm{diag}(U, U, U) \\ B &= \mathrm{diag}(V, V, V) \\ C &= \mathrm{diag}(W, W, W), \end{aligned} \tag{3.2}$$

where U, V, and W are the contravariant velocity components in the ξ, η, and ζ directions, respectively,

$$U^i = u\xi^i_x + v\zeta^i_r + \frac{w}{r}\xi^i_\theta \quad \text{for } i = 1, 2, 3, \tag{3.3}$$

with $(U^1, U^2, U^3) \to (U, V, W)$. The viscous flux vectors E_{v1}, E_{v2}, and E_{v3} which appear in the momentum equation (3), read in compact notation as

$$E_{vj} = \frac{\nu_t}{J}\begin{bmatrix} (\xi_x\xi^j_x + g^{1j})\,u_\xi + (\eta_x\xi^j_x + g^{2j})\,u_\eta + (\zeta_x\xi^j_x + g^{3j})\,u_\zeta + S_{1j} \\ (\xi_r\xi^j_r + g^{1j})\,v_\xi + (\eta_r\xi^j_r + g^{2j})\,v_\eta + (\zeta_r\xi^j_r + g^{3j})\,v_\zeta + S_{2j} \\ \left(\frac{1}{r^2}\xi_\theta\xi^j_\theta + g^{1j}\right)w_\xi + \left(\frac{1}{r^2}\eta_\theta\xi^j_\theta + g^{2j}\right)w_\eta \\ + \left(\frac{1}{r^2}\zeta_\theta\xi^j_\theta + g^{3j}\right)w_\zeta + S_{3j} \end{bmatrix}$$

$$\text{for } j = 1, 2, 3, \tag{4}$$

where

$$S_{1j} = \xi^j_r R_{21} + \frac{1}{r}\xi^j_\theta R_{31} \tag{4.1}$$

$$S_{2j} = \xi^j_x R_{12} + \frac{1}{r}\xi^j_\theta \left(R_{31} - \frac{w}{r}\right) \tag{4.2}$$

$$S_{3j} = \xi^j_x R_{13} + \xi^j_r R_{23} + \frac{1}{r}\left(\frac{2v}{r}\xi^j_\theta - w\xi^j_r\right) \tag{4.3}$$

$$R_{ij} = u^i_\xi \xi_{x_j} + u^i_\eta \eta_{x_j} + u^i_\zeta \zeta_{x_j} \quad \text{for } i, j = 1, 2, 3 \tag{4.4}$$

$$(u^1, u^2, u^3) \to (u, v, w) \tag{4.5}$$

$$(x_1, x_2, x_3) \to (x, r, \theta). \tag{4.6}$$

Finally, the source vectors H and H_v, in the momentum equation (3), are given by the expressions:

$$H = \begin{bmatrix} \xi_x P_\xi + \eta_x P_\eta + \zeta_x P_\zeta \\ \xi_r P_\xi + \eta_r P_\eta + \zeta_r P_\zeta - \dfrac{w^2}{r} \\ \dfrac{1}{r}(\xi_\theta P_\xi + \eta_\theta P_\eta + \zeta_\theta P_\zeta + wv) \end{bmatrix} \tag{5}$$

and

$$H_v = \begin{bmatrix} 0 \\ -\dfrac{2v_t}{r}\left(R_{33} + \dfrac{v}{r}\right) \\ \dfrac{v_t}{r}\left(R_{23} + R_{32} - \dfrac{w}{r}\right) \end{bmatrix}. \tag{6}$$

Finally, v_t is the total kinematic viscosity

$$v_t = v_e + \frac{1}{\mathrm{Re}}, \tag{7}$$

where Re is the Reynolds number of the flow and v_e is the eddy viscosity which is introduced through the Boussinesq's hypothesis to model the Reynolds stresses. In the present work we compute the eddy viscosity using the algebraic model of Baldwin and Lomax in its original form. The details of the model can be found in Ref. [17].

Note that, although we use the pressure-Poisson approach to satisfy the continuity equation (2), we do not derive the pressure equation in this section. The reason is that we want to stress the importance of deriving a discrete pressure equation from the discrete approximations of Eqs. (2) and (3). The derivation of the discrete pressure equation is given in the following sections.

NUMERICAL SOLUTION PROCEDURE

In this section we present a method to integrate numerically the three-dimensional equations from an initial guess to a steady state. The governing equations (2) and (3) are discretized on a non-staggered grid and the continuity equation is replaced by a Poisson equation for the pressure. The time dependent momentum equations are integrated in time using the four-stage, explicit Runge–Kutta scheme. The pressure equation is solved by the point-successive relaxation method.

(a) Discretization of the Governing Equations

To discretize in space the momentum equations (3), we use three-point central finite differencing for the pressure gradient and viscous terms, while we employ second-order upwind differencing for the convective terms. The upwind differencing of the convective terms eliminates the need for adding artificial dissipation terms, to the right-hand side of the momentum equations, to stabilize the numerical algorithm. This is due to the fact that a fixed amount of dissipation is inherent in the upwind differencing.

Referring to Fig. 1, we give discrete approximations of convective, pressure gradient and viscous terms, which appear in Eq. (3), as

$$\left[U\frac{\partial u}{\partial \xi}\right]_{i,j,k} = \frac{1}{2}(|U_{i,j,k}| + U_{i,j,k})\,\delta_\xi^- u_{i,j,k} - \frac{1}{2}(|U_{i,j,k}| - U_{i,j,k})\,\delta_\xi^+ u_{i,j,k}$$

$$\left[\xi_x \frac{\partial P}{\partial \xi}\right]_{i,j,k} = (\xi_x)_{i,j,k}\,\delta_\xi P_{i,j,k}$$

$$\left\{\frac{\partial}{\partial \xi}\left[\frac{v_t}{J}(\xi_x^2 + g^{11})\frac{\partial u}{\partial \xi}\right]\right\}_{i,j,k} = \tilde{\delta}_\xi\left[\frac{v_t}{J}(\xi_x^2 + g^{11})\,\tilde{\delta}_\xi u_{i,j,k}\right], \tag{8}$$

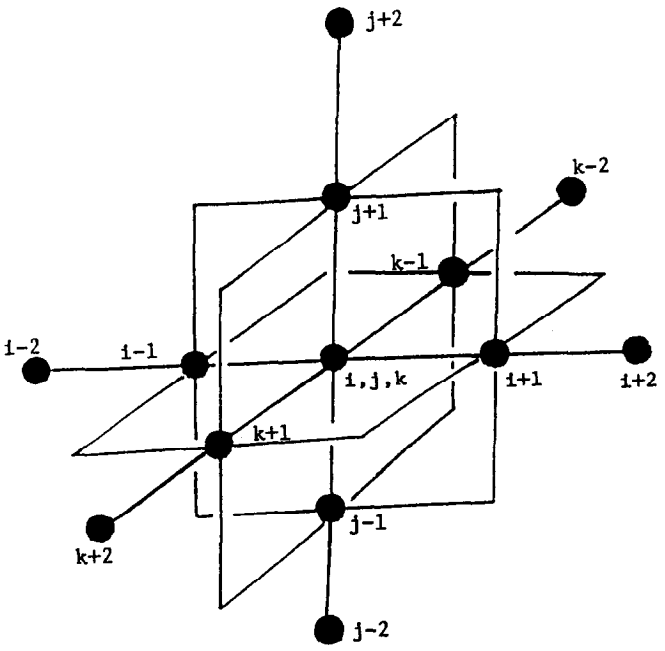

FIG. 1. Three-dimensional computational cell.

where

$$\delta_\xi^{\pm}(\ )_{i,j,k} = \pm\frac{1}{2\Delta\xi}\left[-3(\ )_{i,j,k} + 4(\ )_{i\pm1,j,k} - (\ )_{i\pm2,j,k}\right] \tag{8.1}$$

$$\delta_\xi(\ )_{i,j,k} = \frac{1}{2\Delta\xi}\left[(\ )_{i+1,j,k} - (\ )_{i-1,j,k}\right] \tag{8.2}$$

$$\tilde{\delta}_\xi(\ )_{i,j,k} = \frac{1}{\Delta\xi}\left[(\ )_{i+1/2,j,k} - (\ )_{i-1/2,j,k}\right]. \tag{8.3}$$

The continuity equation (2) is discretized using three-point central finite difference approximations. For the sake of convenience we define the discrete divergence operator as

$$\mathrm{DIV}[Q_{i,j,k}] = J\left[\delta_\xi\left(\frac{U}{J}\right)_{i,j,k} + \delta_\eta\left(\frac{V}{J}\right)_{i,j,k} + \delta_\zeta\left(\frac{W}{J}\right)_{i,j,k}\right], \tag{8.5}$$

where Q is the velocity vector, defined in Eq. (3.1), and U, V, and W are the contravariant components of the velocity, given in Eq. (3.3).

Applying the Runge–Kutta scheme [13] to the system of the governing equations (2) and (3), we obtain (for $l = 1, 2, 3, 4$):

$$\mathrm{DIV}[Q^l_{i,j,k}] = 0 \tag{9}$$

$$Q^l_{i,j,k} = Q^n_{i,j,k} - \alpha_l\,\Delta t_{i,j,k}\,\mathrm{RHS}^{l-1}_{i,j,k}. \tag{10}$$

In the above equations, the superscript "n" denotes the time step at which the solution is known, while the superscript "l" denotes an intermediate time level (or iteration level) used to advance the solution from time step "n" to time step "$n+1$" (we designate $Q^l = Q^n$ for $l = 0$ and $Q^l = Q^{n+1}$ for $l = 4$). For the four-stage scheme, the coefficients a_l's are $\frac{1}{4}$, $\frac{1}{3}$, $\frac{1}{2}$, and 1 for $l = 1, 2, 3, 4$ in sequence. The RHS term, in Eq. (10), denotes the discrete approximation of the right-hand side of the momentum equations (3) at the node (i, j, k):

$$\mathrm{RHS} = A\frac{\partial Q}{\partial\xi} + B\frac{\partial Q}{\partial\eta} + C\frac{\partial Q}{\partial\zeta} + H - J\left[\frac{\partial E_{v1}}{\partial\xi} + \frac{\partial E_{v2}}{\partial\eta} + \frac{\partial E_{v3}}{\partial\zeta}\right] + H_v. \tag{10.1}$$

Also, $\Delta t_{i,j,k}$ in Eq. (10.1) is the time increment which, for reasons we discuss later, varies in space (local time stepping). For the sake of convenience, however, in the rest of our analysis we drop the (i, j, k) subscript.

The major difficulty associated with the solution of the system of the discrete governing equations (9) and (10), is the absence of the time derivative of the pressure from the continuity equation (9). Clearly, the continuity equation (9) is not an evolution equation, which can be used to advance the pressure field in time, but it is rather a constraint imposed on the velocity field. However, the pressure—whose gradient appears in the right-hand side of the momentum equations—serves as the only degree of freedom we can use to satisfy the incompressibility constraint. Therefore, we need to replace the discrete continuity equation with an equivalent discrete equation for the pressure which takes into account the important interaction between pressure and velocity fields. On a non-staggered grid, such an equation must lead to: (i) smooth solutions for the pressure, and (ii) the satisfaction of the discrete continuity equation (9), within acceptable numerical accuracy.

(b) The Pressure-Poisson Equation

In order to derive an equation for the pressure, let us first write out the three scalar momentum equations, implied by Eq. (10), in the form

$$u^l_{i,j,k} = u^n_{i,j,k} - \alpha_l\,\Delta t\left[f^\xi_{i,j,k} + (\xi_x\delta_\xi + \eta_x\delta_\eta + \zeta_x\delta_\zeta)\,P_{i,j,k}\right]^{l-1}$$

$$v^l_{i,j,k} = v^n_{i,j,k} - \alpha_l\,\Delta t\left[f^\eta_{i,j,k} + (\xi_r\delta_\xi + \eta_r\delta_\eta + \zeta_r\delta_\zeta)\,P_{i,j,k}\right]^{l-1}$$

$$w^l_{i,j,k} = w^n_{i,j,k} - \alpha_l\,\Delta t \times\left[f^\zeta_{i,j,k} + \frac{1}{r}(\xi_\theta\delta_\xi + \eta_\theta\delta_\eta + \zeta_\theta\delta_\zeta)\,P_{i,j,k}\right]^{l-1}, \tag{11}$$

where $f^\xi_{i,j,k}$ for example, contains the discrete approximations of the convective and viscous terms which appear in the ξ-momentum equation.

Using Eqs. (11), along with Eq. (3.3), we can eliminate the velocity components, at the "l" level, from the discrete continuity equation (9). Doing so, we arrive at the discrete pressure equation,

$$J\Delta[P_{i,j,k}]^{l-1} = \frac{1}{\alpha_l}\mathrm{DIV}[Q^l_{i,j,k}] - J\sigma^{l-1}_{i,j,k}, \tag{12}$$

where

$$\Delta[P_{i,j,k}] = \left\{\delta_\xi\left[\frac{\Delta t}{J}(g^{11}\delta_\xi + g^{12}\delta_\eta + g^{13}\delta_\zeta)\right] + \delta_\eta\left[\frac{\Delta t}{J}(g^{12}\delta_\xi + g^{22}\delta_\eta + g^{23}\delta_\zeta)\right] + \delta_\zeta\left[\frac{\Delta t}{J}(g^{13}\delta_\xi + g^{23}\delta_\eta + g^{33}\delta_\zeta)\right]\right\}[P_{i,j,k}] \tag{12.1}$$

and

$$\sigma_{i,j,k} = \delta_\xi \left[\frac{\Delta t}{J} (\xi_x f^\xi + \eta_x f^\eta + \zeta_x f^\zeta) \right]_{i,j,k} + \delta_\eta \left[\frac{\Delta t}{J} (\xi_r f^\xi + \eta_r f^\eta + \zeta_r f^\zeta) \right]_{i,j,k} + \delta_\zeta \left[\frac{\Delta t}{rJ} (\xi_\theta f^\xi + \eta_\theta f^\eta + \zeta_\theta f^\zeta) \right]_{i,j,k}. \quad (12.2)$$

For reasons which will become apparent in the process of our analysis, we decompose the pressure operator, Eq. (12.1), into two parts,

$$\Delta[P_{i,j,k}] = L[P_{i,j,k}] + N[P_{i,j,k}], \quad (12.3)$$

where $L[\ ]$ contains the three orthogonal terms in Eq. (7.1),

$$L[P_{i,j,k}] = \left\{ \delta_\xi \left(\frac{g^{11} \Delta t}{J} \delta_\xi \right) + \delta_\eta \left(\frac{g^{22} \Delta t}{J} \delta_\eta \right) + \delta_\zeta \left(\frac{g^{33} \Delta t}{J} \delta_\zeta \right) \right\} [P_{i,j,k}] \quad (12.4)$$

while $N[\ ]$ contains the remaining six cross-derivative terms, which result from the non-orthogonality of the coordinate lines.

Equation (12) is a second-order, elliptic equation for the pressure field which we can use instead of the discrete continuity equation (9). Before we proceed any further, however, we have to make sure that solving Eqs. (11) and (12) is equivalent to solving the original system of Eqs. (11) and (9). In other words, will the velocity field, which results from the solution of Eqs. (11) and (12), satisfy the discrete incompressibility constraint, Eq. (9)? The answer is easy if we employ the same arguments used in Ref. [1] for Cartesian coordinates. The discrete pressure equation (12) is derived from the discrete continuity equation (9) by employing the discrete momentum equations (11) at the nodes, where they are driven to steady state. Therefore, as the solution of Eqs. (11) and (12) approaches a steady state, the discrete divergence operator (Eq. (8.5)) approaches zero. However, the satisfaction of the discrete incompressibility constraint is materialized without the computed pressure being smooth. To make this point clear, let us write out one of the terms appearing in the pressure operator, Eq. (12.1), as

$$\delta_\xi \left(\frac{g^{11} \Delta t}{J} \delta_\xi \right) [P_{i,j,k}] = \frac{1}{4\Delta\xi^2} \left\{ \left(\frac{g^{11} \Delta t}{J} \right)_{i+1,j,k} P_{i+2,j,k} - \left[\left(\frac{g^{11} \Delta t}{J} \right)_{i+1,j,k} + \left(\frac{g^{11} \Delta t}{J} \right)_{i-1,j,k} \right] P_{i,j,k} + \left(\frac{g^{11} \Delta t}{J} \right)_{i-1,j,k} P_{i-2,j,k} \right\}. \quad (13)$$

Such a discretization of the pressure operator produces a pressure field which suffers from odd–even decoupling (checkerboard instability). Therefore, the result of our analysis for three-dimensional, curvilinear coordinates is similar, as one would expect, to the result obtained in Ref. [1] for cartesian coordinates. Namely, on a non-staggered grid the system of the discrete equations (11) and (12) need not have a smooth solution. As a matter of fact, our numerical results indicate that for complex, three-dimensional geometries there may not exist a solution at all, since the oscillations in the ressure can destabilize the entire numerical procedure. Thus, in order to guarantee the existence of a smooth solution, for the discrete Navier–Stokes equations, we have to set aside the idea of satisfying the discrete continuity equation to machine zero [1, 3]. In Ref. [1] we modified the discrete continuity equation by adding to its right-hand side an artificial source term, the size of which is the minimum required to guarantee the existence of a smooth pressure field. The order of this source term is at least equal to the order of the truncation error of the finite difference approximations and, thus, it does not deteriorate the formal accuracy of the solution.

To extend our method [1] to generalized curvilinear coordinates, we first note that the artificial mass source in cartesian coordinates can be interpreted as the difference between the two discrete approximations of the Laplace operator for the pressure: the one that results by discretizing over $2\Delta x$ (invloving the nodes $i+1$, i, and $i-1$) and the one that results by discretizing over $4\Delta x$ (involving the nodes $i+2$, i, and $i-2$). Following this interpretation we seek, in generalized curvilinear coordinates, to satisfy the discrete "continuity" equation,

$$\mathrm{DIV}[Q^l_{i,j,k}] = \varepsilon J(\tilde{L} - L)[P^{l-1}_{i,j,k}], \quad (14)$$

where ε is a positive constant ($\varepsilon \leqslant 1$), used to control the size of the source term. The operator $\tilde{L}$ is defined in a similar fashion as the operator L (see Eq. (12.4)) as

$$\tilde{L}[P_{i,j,k}] = \left\{ \tilde{\delta}_\xi \left(\frac{g^{11} \Delta t}{J} \tilde{\delta}_\xi \right) + \tilde{\delta}_\eta \left(\frac{g^{22} \Delta t}{J} \tilde{\delta}_\eta \right) + \tilde{\delta}_\zeta \left(\frac{g^{33} \Delta t}{J} \tilde{\delta}_\zeta \right) \right\} [P_{i,j,k}], \quad (14.1)$$

where $\tilde{\delta}_\xi$, $\tilde{\delta}_\eta$, and $\tilde{\delta}_\zeta$ are given by Eq. (8.4).

The source term, in the right-hand side of the discrete continuity equation, is proportional to the difference between the two discrete approximations of the orthogonal part of the pressure operator (see Eqs. (12.3) and (12.4)): the one that results by discretizing over $2\Delta\xi$ and the one that results by discretizing over $4\Delta\xi$. The reason that we include only the orthogonal terms of the pressure operator, in the artificial mass source, is that these terms—as numerical experiments indicated—are sufficient to guarantee the smoothness of the pressure field. Therefore, the inclusion of the non-orthogonal terms in the right-hand side of Eq. (14) would only contribute to additional error in the discrete continuity equation.

In order to bring the source term in Eq. (14) to a form which resembles the source term used in Ref. [1], for uniform cartesian coordinates, we use Eqs. (14.1) and (12.4) to obtain

$$\varepsilon J(\tilde{L}-L)[P_{i,j,k}] = -\frac{\varepsilon J}{4}\left\{\Delta\xi^2\,\delta_{\xi\xi}\left(\frac{g^{11}\,\Delta t}{J}\delta_{\xi\xi}\right) + \Delta\eta^2\,\delta_{\eta\eta}\left(\frac{g^{22}\,\Delta t}{J}\delta_{\eta\eta}\right) + \Delta\zeta^2\,\delta_{\zeta\zeta}\left(\frac{g^{33}\,\Delta t}{J}\delta_{\zeta\zeta}\right)\right\}[P_{i,j,k}], \quad (15)$$

where

$$\delta_{\xi\xi}[\ ] = \frac{[\ ]_{i+1,j,k} - 2[\ ]_{i,j,k} + [\ ]_{i-1,j,k}}{(\Delta\xi)^2}. \quad (15.1)$$

To derive Eq. (15), we used averaging to compute the coefficients (coefficients inside the derivatives in Eq. (15)) at the half-grid nodes, where they are needed for the discretization of the $\tilde{L}$ operator. As can be seen from Eq. (15) the source term is at least second order in the transformed computational domain. Therefore, the formal acuracy of the method is not deteriorated. Also, it is easy to show that, for uniform cartesian grids, the right-hand side of Eq. (15) becomes identical to the source term of Ref. [1].

Using Eq. (14) along with Eq. (12.3) and the momentum equations (11), we derive the following discrete pressure equation:

$$(1-\varepsilon)\,L[P^{l-1}_{i,j,k}] + \varepsilon\tilde{L}[P^{l-1}_{i,j,k}] + N[P^{l-1}_{i,j,k}] = \frac{1}{\alpha_1}\,\mathrm{DIV}[Q^{n}_{i,j,k}] - \sigma^{l-1}_{i,j,k}. \quad (16)$$

The discrete pressure operator (left-hand side of Eq. (4.39)) couples together—for sufficiently large values of the ε parameter—the odd and even nodes of the pressure. This can be seen, if we write out a typical term of the pressure operator as

$$\begin{aligned}[(1-\varepsilon)\,\delta_\xi(\beta\delta_\xi) + \varepsilon\tilde{\delta}_\xi(\beta\tilde{\delta}_\xi)][P_{i,j,k}] &= \left[(1-\varepsilon)\frac{\beta_{i,j,k}}{4\Delta\xi^2}\right]P_{i+2,j,k} + \left[\varepsilon\frac{\beta_{i+1/2,j,k}}{\Delta\xi^2}\right]P_{i+1,j,k} \\ &\quad - \left[(1-\varepsilon)\frac{\beta_{i+1,j,k}+\beta_{i-1,j,k}}{4\Delta\xi^2} + \varepsilon\frac{\beta_{i+1/2,j,k}+\beta_{i-1/2,j,k}}{\Delta\xi^2}\right]P_{i,j,k} \\ &\quad + \left[\varepsilon\frac{\beta_{i-1/2,j,k}}{\Delta\xi^2}\right]P_{i-1,j,k} + \left[(1-\varepsilon)\frac{\beta_{i-1,j,k}}{4\Delta\xi^2}\right]P_{i-2,j,k}, \end{aligned} \quad (16.1)$$

where

$$\beta = g^{11}\,\Delta t/J. \quad (16.2)$$

Our numerical experiments indicate that, values of ε of the order of 0.05 are sufficient to guarantee the existence of a smooth pressure field and consequently the stability of the numerical procedure (see results and discussion section).

(b.1) *Boundary Conditions for the Pressure Equation*

A well-posed boundary value problem, for the elliptic pressure equation (16), requires the specification of the boundary conditions on all the boundaries of the computational domain. However, the specification of a proper set of boundary conditions for the pressure is not a trivial task, since the system of the governing equations (2) and (3) is usually closed with boundary conditions on the velocity field. Gresho and Sani [18] investigated the problem of the pressure boundary conditions and derived the relevant pressure equation for nodes near or on the boundary of the computational domain. They also proved that "the Neumann boundary condition (normal momentum equation on the boundary) is always appropriate for the pressure-Poisson equation."

In this section we derive the relevant discrete pressure equation (for the proposed formulation) for grid nodes near the boundary of the solution domain and we show a simple way to build in the discrete pressure equation (16) all the different relevant boundary forms encountered in a three-dimensional domain. For that matter, let us consider the discrete continuity equation written as

$$\frac{U^l_{i+1}-U^l_{i-1}}{2\Delta\xi} + \frac{V^l_{j+1}-V^l_{j-1}}{2\Delta\eta} + \frac{W^l_{k+1}-W^l_{k-1}}{2\Delta\zeta} = \frac{1}{J}\,d_{i,j,k}, \quad (17)$$

where d is the artificial mass source of Eq. (14). Note that at nodes next to the boundary we do not include, in the artificial source term, terms whose evaluation would require nodes outisde the computational domain. For example, next to a $\xi =$ constant boundary we neglect the ξ-direction derivative (see Eq. (15)).

The general form of the pressure equation which can be derived from Eq. (17) and the momentum equations (11), equation (16) (or, equivalently, Eq. (18)) the same way at every computational node, adjacent or not to the boundary, since the various relevant forms of the pressure equation for nodes next to the boundary are recovered automatically.

To calculate the pressure values at the boundary of the solution domain, we employ the normal momentum equation [1, 18] (Neumann condition). At a $\xi =$ constant boundary, for example, the pressure can be computed from

$$\frac{U^n_{i+1}-U^n_{i-1}}{2\Delta\xi}-\frac{1}{2\Delta\xi}\left[(\Delta t\, NM^\xi)_{i+1}-(\Delta t\, NM^\xi)_{i-1}\right]^{l-1}$$
$$+\frac{V^n_{j+1}-V^n_{j-1}}{2\Delta\eta}-\frac{1}{2\Delta\eta}$$
$$\times\left[(\Delta t\, NM^\eta)_{j+1}-(\Delta t\, NM^\eta)_{j-1}\right]^{l-1}$$
$$+\frac{W^n_{k+1}-W^n_{k-1}}{2\Delta\zeta}-\frac{1}{2\Delta\zeta}$$
$$\times\left[(\Delta t\, NM^\zeta)_{k+1}-(\Delta t\, NM^\zeta)_{k-1}\right]^{l-1}=\frac{1}{J}d^{l-1}_{i,j,k}, \quad (18)$$

where NM^ξ, for example, is the normal to a $\xi =$ constant surface steady state momentum equation. Let us assume now, that a $\xi =$ constant boundary is located at $(i-1, j, k)$. Since boundary conditions are specified on the velocity, the ξ-contravariant component of the velocity is known at $(i-1, j, k)$ for all instants in time:

$$U^l_{i-1}=U^n_{i-1}=U_{i-1}. \quad (18.1)$$

Therefore, in order to derive a pressure equation from the discrete continuity equation (17) we do not need to employ the momentum equations at $(i-1, j, k)$ [18]. In other words, incorporating Eq. (18.1) in Eq. (17) and using the momentum equations for the nodes where the velocity is not specified, we obtain

$$\frac{U^n_{i+1}-U_{i-1}}{2\Delta\xi}-\frac{1}{2\Delta\xi}(\Delta t\, NM^\xi)^{l-1}_{i+1}+\cdots=\frac{1}{J}d^{l-1}_{i,j,k}. \quad (18.2)$$

In Eq. (18.2) we omitted the η- and ζ-direction derivative terms which remain the same as in Eq. (18). Equation (18.2) is the relevant discrete pressure equation for a node next to a $\xi =$ constant boundary. In a three-dimensional domain, there are 24 different boundary forms of the discrete pressure equation, the programming of which can be quite complicated. However, all the complications can be avoided if we note that Eq. (18)—which holds for any interior node—reduces automatically to Eq. (18.2)—which holds for a node adjacent to a $\xi =$ constant boundary—if we set the time increment Δt equal to zero at the boundary nodes. By doing so, we can program the discrete pressure

$$\delta^+_\xi(P_{i,j,k})=\frac{1}{g^{11}_{i,j,k}}\left[-g^{12}_{i,j,k}\delta_\eta(P_{i,j,k})\right.$$
$$\left.-g^{13}_{i,j,k}\delta_\zeta(P_{i,j,k})+F^\xi_{i,j,k}\right], \quad (19)$$

where F^ξ contains velocity terms. The one-sided derivative, in the left-hand side of Eq. (19), is forward/backward for a left/right $\xi =$ constant boundary.

Finally, we can easily show that the compatibility condition—which is necessary for a solution to exist—is automatically satisfied. The proof is a straightforward extensionof the detailed proof we gave in Ref. [1].

(c) Summary of the Numerical Algorithm and Convergence Accelaration Techniques

Assuming that we know the solution at the "n" time level, the solution at the "$n+1$" time level is obtained through the following steps:

(1) Compute f^ξ, f^η, and f^ζ (see Eqs. (11)) at the "$l-1$" stage.

(2) Compute the right-hand side of the pressure equation (16) (see Eqs. (8.5) and (12.2)).

(3) Solve the pressure equation (16), using the point-successive relaxation method, to obtain the pressure field at the "$l-1$" stage. Since we are interested inthe steady state solution, only one iterationis performed on the pressure equation.

(4) Using the already computed f^ξ, f^η, and f^ζ terms (step (1)), along with the new pressure field (step (3)), compute the right-hand side RHS (see Eq. (10.1)) of the momentum equations.

(5) Update the velocity field using Eqs. (11).

For the four-stage Runge–Kutta scheme the steps (1) to (5) are repeated for $l=1, 2, 3, 4$, since $Q^l=Q^{n+1}$ for $l=4$.

As we have already mentioned, the f^ξ, f^η, and f^ζ terms contain convective and viscous terms. In order to save computational time, the viscous part of the above terms is updated only at the first stage of the four-stage procedure and kept constant through the subsequent three stages [15]. Also, in the iterative procedure described above (steps (1) to (5) for $l=1, 2, 3, 4$) a total of four iterations is

performed on the pressure equation per time step. Numerical experimentation showed that updating the pressure only once per time step does not affect significantly the convergence rate of the iterative procedure. Therefore, in order to save computational time we chose to update the pressure only at the first stage of the four-stage scheme.

To enhance the convergence rate or the time marching procedure we employ the local time stepping technique along with implicit residual smoothing. The time increment is computed and stored for every node at the beginning of the marching procedure as

$$\Delta t_{i,j,k} = \mathrm{CFL}\, \Delta S_{i,j,k},$$

where

$$\Delta S_{i,j,k} = \min(\sqrt{g_{11}}, \sqrt{g_{22}}, \sqrt{g_{33}}). \tag{20}$$

In the above equations, CFL is the Courant–Friedrich–Lewis number, which is held approximately constant at every node in order to maximize the local damping of the error. Also, g_{11}, g_{22}, and g_{33} are the components of the covariant metric tensor which represent the local arc lengths in the ξ-, η-, and ζ-directions, respectively. As we discussed in the previous section, the time increment is set equal to zero everywhere at the boundary nodes in order to facilitate the application of the pressure boundary conditions.

The implicit residual smoothing was first proposed by Lerat (see, for example, [19]) for use with the Lax–Wendroff scheme and was later adopted by Jameson [14] to accelerate the convergence of Runge–Kutta schemes. In the present study we apply implicit smoothing only to the residual of the momentum equation. More specifically, the residual calculated in Eq. (10) is smoothed by the constant coefficient implicit operator to define a new residual:

$$(1-\varepsilon_\xi \delta_{\xi\xi})(1-\varepsilon_\eta \delta_{\eta\eta})(1-\varepsilon_\zeta \delta_{\zeta\zeta})\,\overline{\mathrm{RHS}}^l = \mathrm{RHS}^l. \tag{21}$$

The constants ε_ξ, ε_η, and ε_ζ are smoothing parameters which are of the order of one and their subscripts indicate that they can be choosen differently for each spatial direction. Equation (21) is solved using the Thomas algorithm and the smoothed residual replaces the residual RHS in Eq. (10). The implemantation of the implicit residual smoothing in the four-stage procedure allows the use of much higher CFL numbers and consequently leads to a significant accelaration of the convergence rate.

RESULTS AND DISCUSSION

To validate the generalized coordinate version of our method we apply it to calculate the three-dimensional, turbulent flow over a ship hull—the parabolic Wigley hull. The geometry of the Wigley hull is described in Ref. [20]. In this section we describe the calculation procedure and we present a sample of the computed results, without emphasizing the physical features of the computed flow. Our main objective is to demonstrate the ability of our method to drive the discrete divergence equation as close to zero as it can be with the computed pressure being smooth. Calculations over other more complex ship models, which have been already completed, along with a detailed discussion of the physics of ship stern and wake flows will be presented in a forthcoming publication.

The physical and computational (transformed) domains are depicted in Fig. 2. The upstream and downstream boundaries are located at $x/L = 0.5$ (midships) and 2, respectively, while the outer boundary are located at $r/L = 0.5$, where L is the length of the hull. At the inlet of the computational domain ($ABCD$) we specify the velocity components from experimental data [19]. The water surface ($ABFEO'$), the keel plane ($DLKGC$), and the wake centerplane ($O'LKE$) are symmetry planes and, therefore, we apply symmetry conditions there. At the exit plane ($EFGK$) we set the streamwise diffusion to zero, i.e., $u_{\xi\xi} = v_{\xi\xi} = w_{\xi\xi} = 0$,

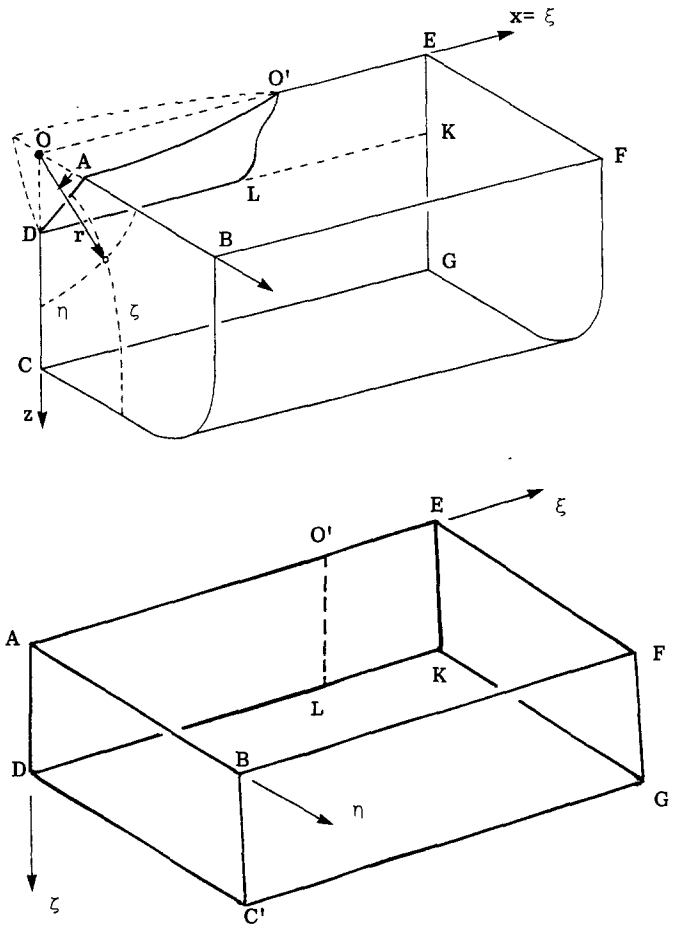

FIG. 2. Physical and computational domains.

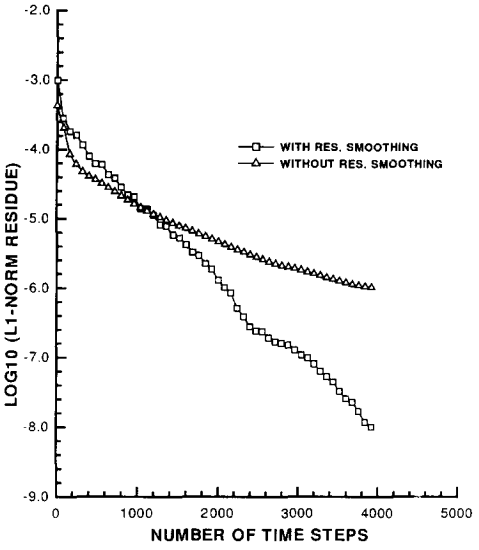

FIG. 3. Convergence history for the residual of the ξ-momentum equation.

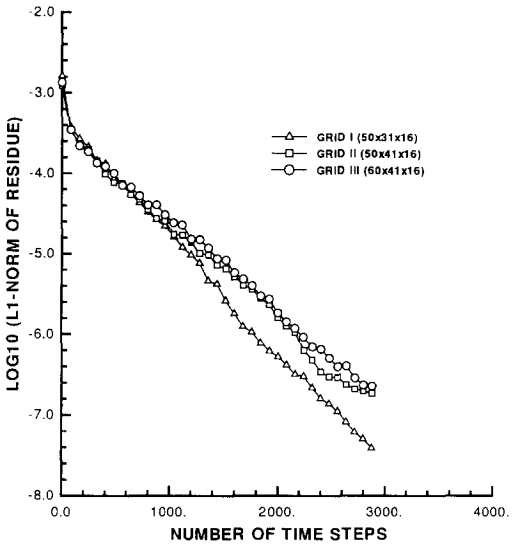

FIG. 5. Convergence histories for the residual of the ξ-momentum equationon grids I, II, and III.

while at the outer boundary ($BCGF$), we set $u_{\eta\eta} = v_{\eta\eta} = w_{\eta\eta} = 0$. Finally, on the hull surface ($ADLO'$) we apply the no-flux and no-slip conditions.

To generate the computational grid, we set $\xi = x$ [10, 21] and specify the desired axial locations of the cross planes. Then we generate a two-dimensional mesh in every cross plane, using the GRAPE method [22]. We complete the three-dimensional grid by connecting the corresponding points of every cross plane. We stretch the grid in the axial direction (around the end of the stern $x/L = 1$), in the radial direction (near the hull surface), and in the girthwise direction (near the keel and water surfaces), using the hyperbolic tangent distribution function. Calculations are performed on three grids: (i) grid I with $50 \times 31 \times 16$ nodes, (ii) grid II with $50 \times 41 \times 16$ nodes, and (iii) with $60 \times 41 \times 16$ nodes in the axial, radial, and girthwise directions, respectively. The first coordinate surface ($\eta = \text{constant}$) just off the hull is located—almost everywhere—at $y^+ < 20$ for grid I and $y^+ < 7$ for grids II and III. The stretching ratio is kept less or equal to 1.3 in all spatial directions and for all the three grids. The Reynolds number of the calculations is $\text{Re} = 4.5 \times 10^6$, based on L and the freestream velocity. All the computed results presented here have been obtained, unless otherwise indicated, on grid II.

The calculations are performed on a CRAY-YMP machine and the CPU time per grid node per iteration is 2×10^{-5} s for the unsmoothed. The application of the implicit residual smoothing, which requires the solution of a scalar tridiagonal system for each one of the velocity components, increases the above time by approximately 10%.

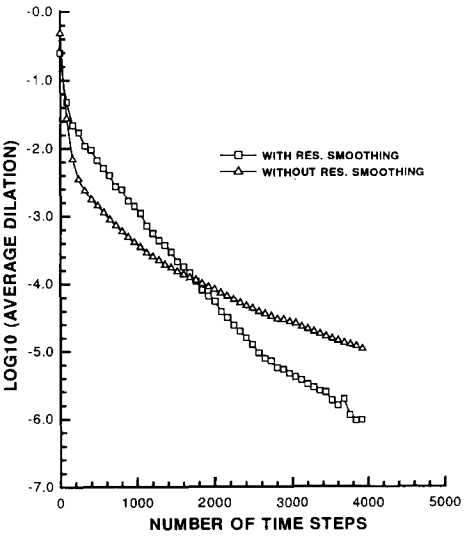

FIG. 4. Convergence history for the average change, between iterations, of the dilation.

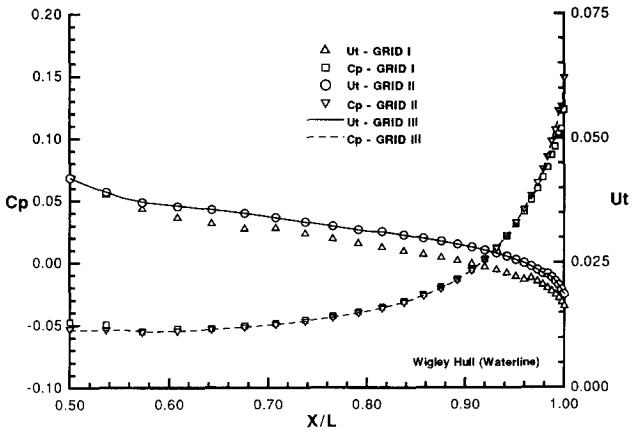

FIG. 6. Grid dependence study.

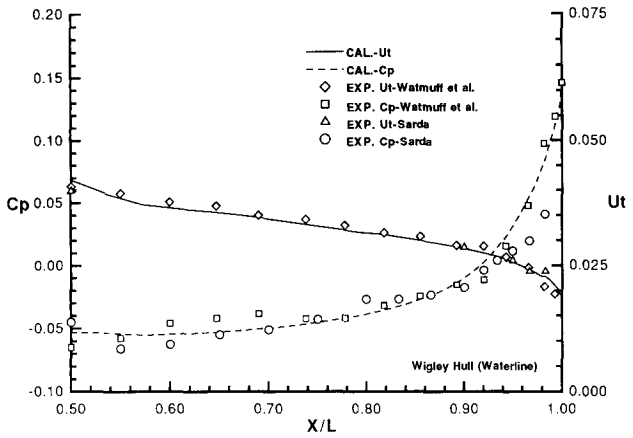

FIG. 7. Variation of the pressure coefficient and the friction velocity along the waterline.

For the unsmoothed scheme the maximum CFL number for stable calculations is 0.7 while the implementation of the implicit residual smoothing ($\varepsilon_\xi = 0.6$, $\varepsilon_\eta = \varepsilon_\zeta = 0.25$) allows us to use a CFL number equal to 2.5. The parameter ε in Eq. (16) is set equal to 0.05 since lower values cause instabilities and the solution eventually "blows up." Also, all calculations are performed using an underrelaxation factor of 0.5 for the solution of the pressure-Poisson equation. Note that we do not make any effort to optimize the parameters which determine the convergence characteristics of the method, namely the CFL number, the implicit smoothing constants, and the underrelaxation parameter for the pressure equation.

The convergence of the $L1$-norm of the residual of the ξ-momentum equation (for grid II) is shown in Fig. 3 for both unsmoothed and smoothed schemes. As can be seen, the unsmoothed scheme converges twice as fast as the unsmoothed one. Moreover, the unsmoothed scheme does not achieve the level of convergence achieved by the smoothed scheme. Clearly, therefore, the implementation of the implicit residual smoothing makes the Runge–Kutta scheme efficient and very effective in handling "stiff" computational meshes (meshes with very high aspect ratios) as those encountered inturbulent flow calculations.

The effect of implicit residual smoothing is even more impressive in the convergence of the continuity equation. Figure 4 shows the convergence of the average, over all the grid nodes, change between iterations of the dilation for both the smoothed and unsmoothed schemes (grid II). The smoothed scheme converges twice as fast as the unsmoothed one and yields a steady state average dilation (see Eq. (8.5) for the definition of the dilation) of 8×10^{-4} in 4000 time steps. The unsmoothed scheme in the same number of iterations yields an average dilation of approximately 5×10^{-3}. This difference in the steady state average dilation is due to the different levels of convergence of the momentum equations. To make this point clear we refer to Eq. (18)—which is another form of the pressure-Poisson equation. As it can be seen from Eq. (18), at any instant in time the dilation is equal to the divergence of the steady state momentum equation plus the artificial source term. At steady state the dilation is driven, at every node, to the artificial source term provided that the three components of the momentum equation are fully converged. For the unsmoothed scheme the momentum residuals never reach full convergence and therefore they contribute to an additional error in the dilation.

As a conclusion we can say that the implicit residual smoothing, althoug it is applied only to the momentum equations, has a very positive impact on the satisfaction of the discrete continuity equation. However, we should note

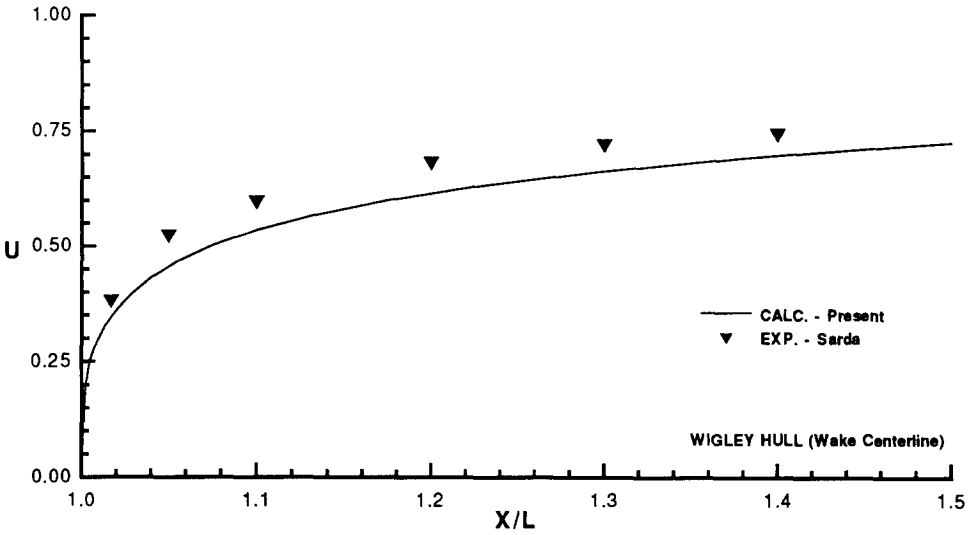

FIG. 8. Variationof the axial velocity component along the wake centerline.

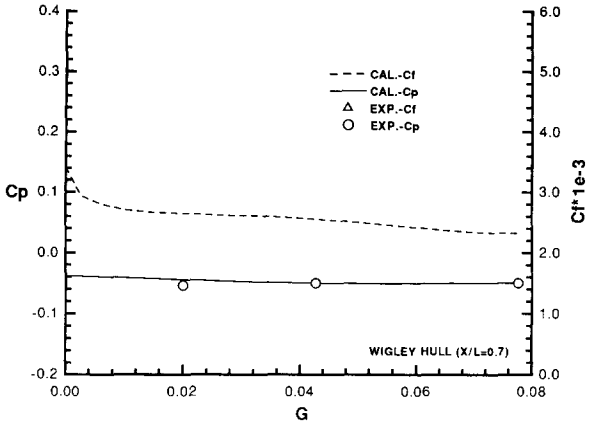

FIG. 9. Girthwise variation of the pressure coefficient and the skin-friction at $X/L = 0.7$.

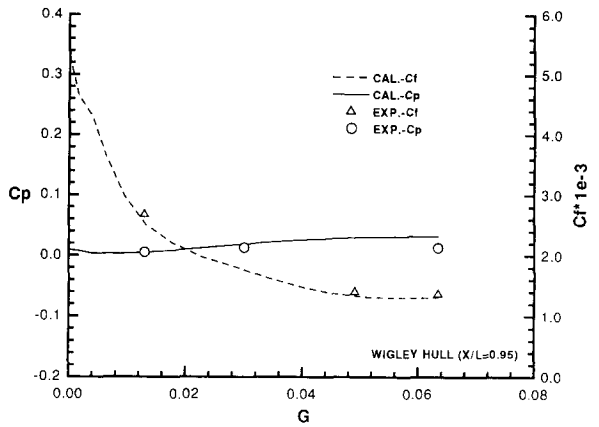

FIG. 11. Girthwise variation of the pressure coefficient and the skin-friction at $X/L = 0.95$.

that both the smoothed and the unsmoothed schemes yield very low steady state values of the dilation.

The convergence curves for the $L1$-norm of the ξ-momentum equation on grids I, II and III are shown in Fig. 5. The linearity of all three convergence curves indicates that, for all the three grids tested, the method converges almost exponentially. Figure 6 shows the variation of the pressure coefficient ($C_p = 2P$) and the friction velocity u_t along the waterline, for grids I, II, and III. Clearly, the two finer grids (II and III) produce identical results. The solution on the coarse grid, on the other hand, shows some departure from the other two—particularly as far as the friction velocity distribution is concerned—presumably due to the inadequate resolution of the near wall region. Recall that on grid I the first coordinate surface just off the wall is located at $y^+ < 20$, almost everywhere, which apparently is not sufficient to capture the near wall steep gradients of the turbulent boundary layer.

The computed variations of the pressure coefficient and the friction velocity along the waterline are compared, in Fig. 7, with the experimental data of Sarda [20] and Watmuff and Joubert [21]. The predicted pressure coefficient agrees well with both sets of experimental data up to $x/L = 0.9$. Beyond this point the calculations follow the measurements of Watmuff and Joubert, since Sarda's measurements predict smaller pressure gradients in the stern region. As pointed out in Ref. [21], this difference between the two sets of data could be due to the fact that Sarda's model was not precisely the same as the mathematical one—his being thicker in the stern region. The computed variation of the friction velocity along the waterline is, also, in very good agreement with the experimental data. Note that the calculations capture accurately the continuous thickening of the boundary layer—implied by the decreasing friction velocity—as the stern is approached.

Figure 8 shows the variation of the axial velocity components along the wake centerline (line $O'E$ in Fig. 2). As can be seen, our calculations underpredict somewhat the

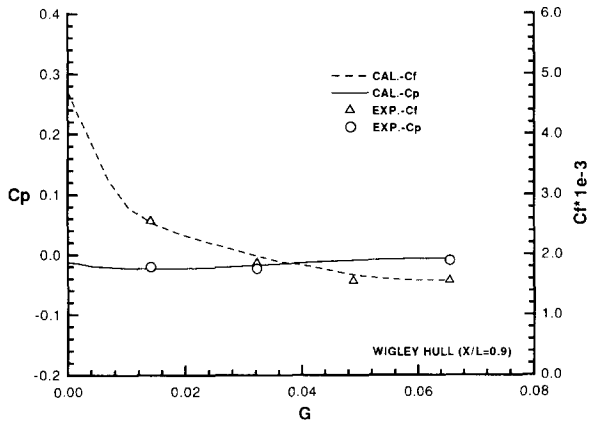

FIG. 10. Girthwise variation of the pressure coefficient and the skin-friction at $X/L = 0.9$.

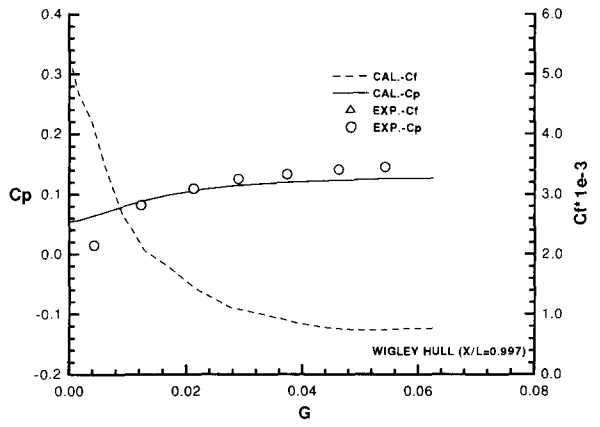

FIG. 12. Girthwise variation of the pressure coefficient and the skin-friction at $X/L = 0.997$.

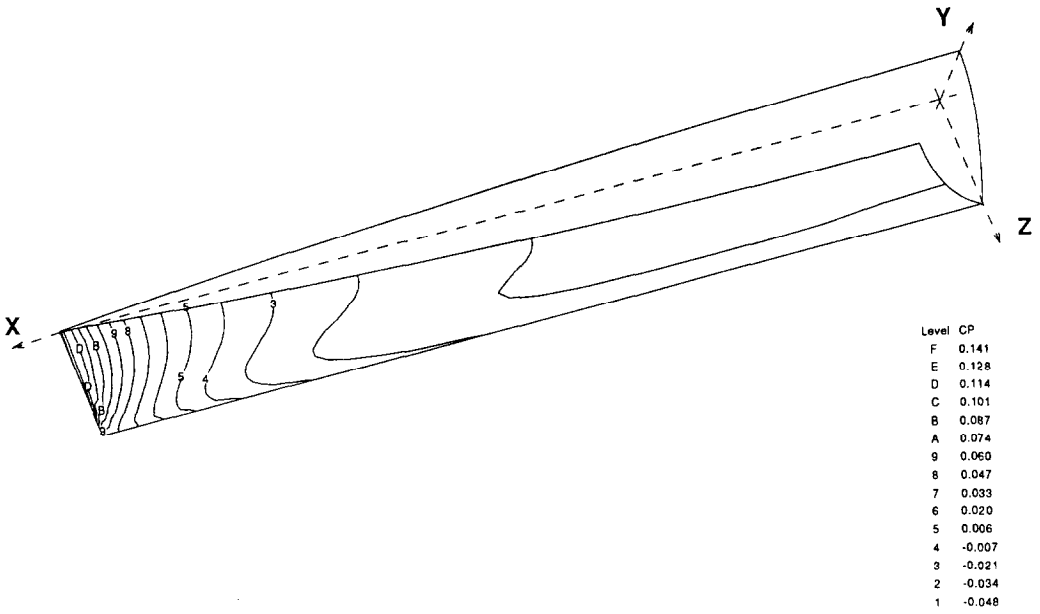

FIG. 13. Contours of constant Cp on the surface of the hull surface.

growth of the wake. This can be attributed to the inadequacy of the turbulence modelling to descride accurately the evolution of a three-dimensional wake.

Figures 9 to 12 depict the variation of the pressure coefficient and the skin-friction ($C_f = 2u_\tau^2$) in the girthwise direction. In all the above figures G denotes the girthwise arc length of the cross section, measured from the keel. As can be seen in Fig. 10, the girthwise variation of the pressure and skin-friction is small at $x/L = 0.7$. However, as the stern is approached—Fig. 11 to 12—the pressure increases from the keel to the waterline and so does the boundary layer thickness, as indicated by the decreasing skin-friction. To demonstrate the ability of the proposed method to produce a smooth pressure field on non-staggered computational

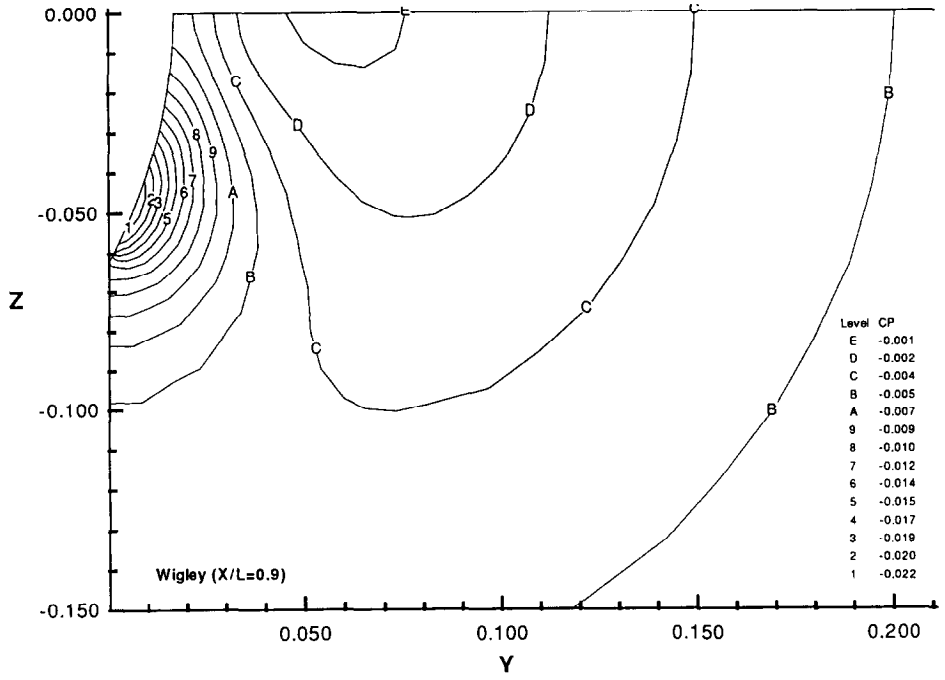

FIG. 14. Contours of constant Cp in the trasverse section $X/L = 0.9$.

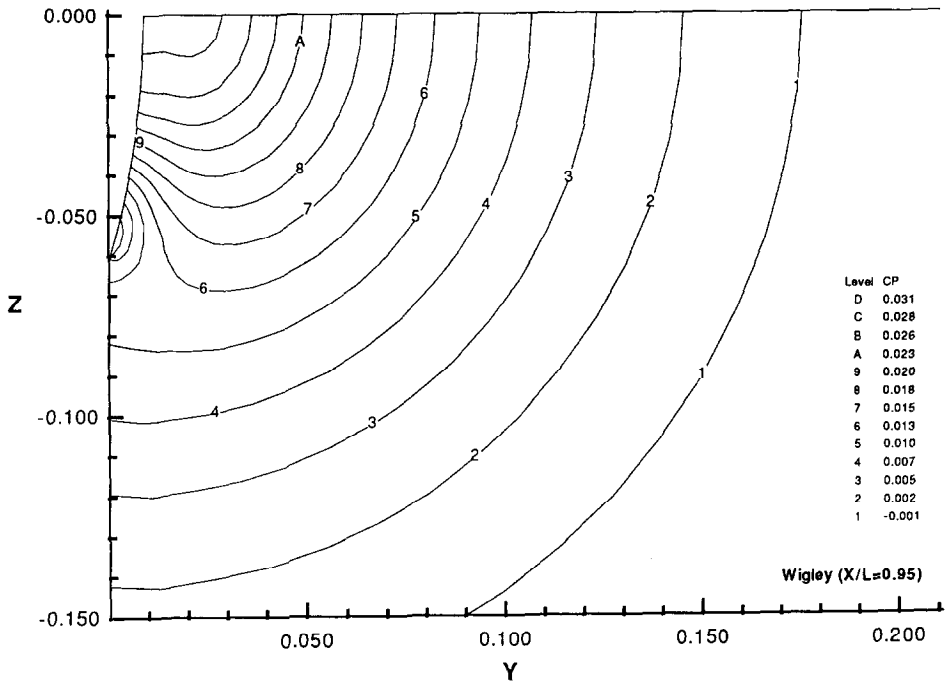

FIG. 15. Contours of constant Cp in the trasverse section $X/L = 0.95$.

meshes, we show in Fig. 13, 14, 15, and 16 the calculated contours of constant pressure on the surface of the hull as well as in several cross-sectional planes.

CONCLUSIONS

We presented an accurate and efficient primitive variable method for the solution of the Reynolds averaged Navier–Stokes equations in generalized curvilinear coordinates on non-staggered grids. Through numerical experimentation we showed that the method produces small values for the discrete dilation with the computed pressure being smooth. We applied the method to calculate the three-dimensional, turbulent flow field over a ship model. The computed results are in very good agreement with the experimental data.

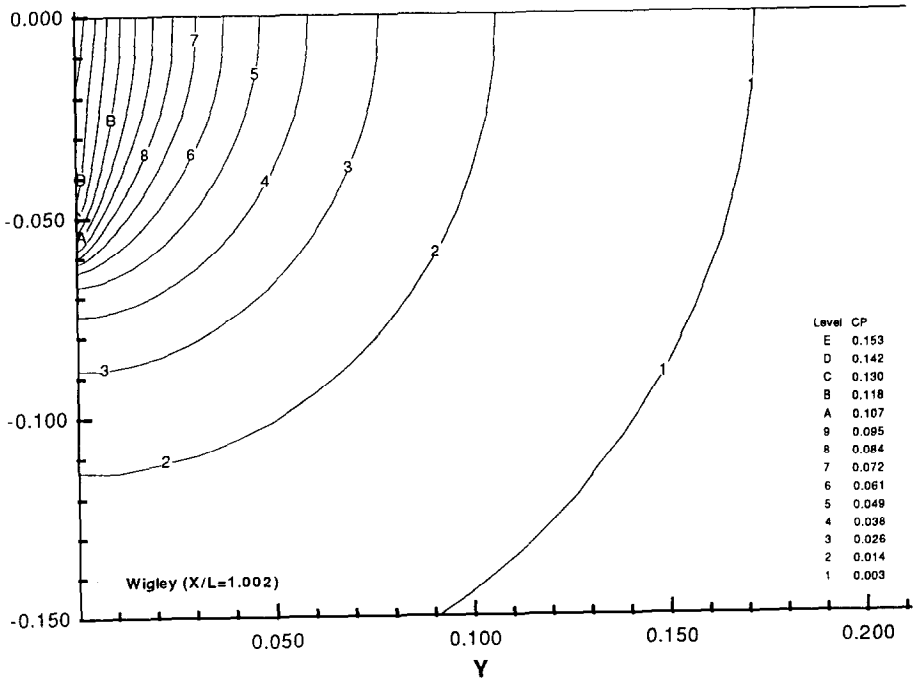

FIG. 16. Contours of constant Cp in the trasverse section $X/L = 1.002$.

ACKNOWLEDGMENT

This work was sponsored by the office of Naval Research, Code 12

REFERENCES

1. F. Sotiropoulos and S. Abdallah, *J. Comput. Phys.* **95** (1991).
2. S. Abdallah, *J. Comput. Phys.* **70**, 193 (1987).
3. K. N. Ghia, W. L. Hankey, and H. K. Hodge, *AIAA J.* **17**, 298 (1989).
4. J. C. Strikwerda and Y. M. Nagel, *J. Comput. Phys.* **78**, 64 (1988).
5. V. Babu and S. A. Korpela, *Comput. Fluids* **18**, 229 (1990).
6. F. H. Harlow and J. E. Welch, *Phys. Fluids* **8**, 2182 (1965).
7. W. Y. Soh and J. W. Goodrich, *J. Comput. Phys.* **79**, 113 (1988).
8. J. Kim and P. Moin, *J. Comput. Phys.* **59**, 308 (1985).
9. H. Miyata, T. Sato, and N. Baba, *J. Comput. Phys.* **72**, 393 (1987).
10. H. C. Chen, V. C. Patel, and S. Ju, *J. Comput. Phys.* **88**, 305 (1990).
11. J. Piquet and M. Visonneau, *Comput. Fluids* **19**, 183 (1991).
12. M. Peric, R. Kessler, and G. Scheuerer, *Comput. Fluids* **16**, 389 (1988).
13. A. Jameson, "Numerical Solution of the Euler Equations by Finite Volume Methods using Runge–Kutta Time-Stepping Schemes," AIAA Paper 81-1259, AIAA 14th Fluid and Plasma Dynamics Conference, Palo Alto, CA, June 1981.
14. [illegible] Complex Configurations," AIAA Paper 85-1929, July 1985.
15. R. V. Chima and J. W. Yokota, *AIAA J.* **28**, 798 (1990).
16. C. L. Merkle and P. Y.-L. Tsai, "Aplication of Runge–Kutta Schemes to Incompressible Flows," AIAA Paper 86-0553, AIAA 24th Aerospace Sciences Meeting, Reno, NV, January 1986.
17. B. S. Baldwin and H. Lomax, "Thin Layer Approximations and Algebraic Model for Separated Turbulent Flows," AIAA Paper 78–257, AIAA 16th Aerospace Science Meeting, Huntsvill, AL, January 1978.
18. P. M. Gresho and R. L. Sani, *Int. J. Numer. Methods Fluids* **7**, 1111 (1987).
19. H. Hollanders, A. Lerat, and R. Peyret, *AIAA J.* **23**, 1670 (1985).
20. O. P. Sarda, Doctoral thesis, Mechanical Engineering, University of Iowa, Iowa City, IA, 1983 (unpublished).
21. V. C. Patel, H. C. Chen, and S. U. Ju, Iowa Institute of Hydraulic Research, University of Iowa, Iowa City, IA, IIHR Report No. 323, 1988 (unpublished).
22. R. L. Sorenson, Ames Research Center, Moffett Field, CA, NASA TM-81198 (unpublished).