JOURNAL OF COMPUTATIONAL PHYSICS 117, 215–227 (1995)

Solution of Flow in Complex Geometries by the Pseudospectral Element Method

HWAR-CHING KU

Johns Hopkins University Applied Physics Laboratory, Johns Hopkins Road, Laurel, Maryland 20723

Received January 21, 1993; revised April 4, 1994

A multi-grid domain decomposition approach with the Schwarz alternating procedure has been developed for the solution of flow in complex geometries by the pseudospectral element method in primitive variable form. The approach for flow problems is first to divide the computational domain into a number of simple blocks (fine-grid or coarse-grid subdomains) with the inter-overlapping zone, of which the overlapped grids may or may not be coincided with each other. An isoparametric mapping is next to map each block onto a simple square (or cube), where c^0 pseudospectral elements (quadrilateral or hexahedral), generated by linearly interpolating the shape function which defines the geomery of their parent elements (blocks), are further used to partition the mapped domain. Schwarz alternating procedure is used to exchange data among the blocks, where the multi-grid (two grid) technique is implemented to remove the high frequency error that occurs when the data interpolation from the fine-grid subdomain to the coarse-grid subdomain is conducted. The solution of the pressure equation in each block can be efficiently solved by the two-level preconditioned minimal residual method in terms of eigenfunction expansion technique, which greatly reduces the inverse of the preconditioned matrix to the simplest "algebraic" form with the least storage requirement, i.e., $O(N^3)$ in 3D and $O(N^2)$ in 2D. Numerical experiments have been performed for both the two-dimensional flow over a cylinder in a channel (Re = 100–1000) and three-dimensional bifurcation flow (Re = 500) to account for the versatility of the proposed technique. The shedding frequency behind a cylinder, Strouhal number increases with increasing Reynolds number, which is different from those found in a square cylinder. As for the bifurcating case, the streamwise velocity profiles of the two-dimensional flow model underestimates the three-dimensional results due to the negligence of boundary effect. © 1995 Academic Press, Inc.

1. INTRODUCTION

Classical fluid mechanics is an approximate description of Nature having wide applicability; however, the nonlinearity of its governing equations (the Navier–Stokes equations) makes fluid flow difficult to understand. The difficulty is compounded when the flow occurs in a domain with complicated geometry, in which case insight must come from experiments or high speed computer simulation. It is not unusual for a complicated flow problem to require more than a million grid points. Even the largest and most modern supercomputers will require huge computing time for such a problem. But that level of calculation still falls far short of a complete numerical attack on the equations of motion, and a lot of ingenuity and attention to computational economy is needed to make realistic simulations practical.

To address these goals, the desired features of numerical methods should have (1) the ability to deal with the variety of geometrical shapes that occur in practice; (2) the high resolution in the interesting areas (locally adaptive scheme); (3) the minimal working space associated with the problem; and (4) the low running time of computation which requires the numerical algorithms that are effective under the multiple processors working environment. Even though the above requirements sound impossible to be met, the fundamental principle still motivates all researchers toward that direction.

How to generate appropriate computational grids is the main issue we often encounter. Generally, grid networks with the desired properties, the comformity close to the boundaries of the region as well as the concentration of points to the area where the steep change of the field variables occurs, can be generated numerically by several techniques, all of which involve mapping physical space (curvilinear coordinates) into a computational space (Cartesian coordinates). When the two coordinate systems are related by an explicit function, the mapping is called "algebraic." If the relationship takes the form of a differential equation, the term "equation-generated" mapping [1] is used. However, the "isoparametric mapping" [2] method, the same set of functions (shape functions) defining both the geometry and the field variables, is adopted to this paper to provide the computational grids.

Perhaps the most commonly used time-step splitting method [3] of satisfying the continuity equation is with the aid of a derived equation, namely the pressure equation $Lp = S$. The pressure equation appearing in curvilinear coordinates contains a nonseparable operator for which there is no easy way for a direct solution, and it is even more difficult in three-dimensional cases. Another alternative for the pressure solution, $L_{ap}^{-1}Lp = L_{ap}^{-1}S$ is the iterative preconditioned minimal residual method [4, 5], which has been widely used for the solution of partial differential equations. An incomplete LU factorization of matrix L is often chosen as the preconditioned matrix L_{ap}, and

0021-9991/95 $6.00

it is constructed from the approximation of banded diagonal elements based on the row sum equality rule [6]. Recently, another promising research on the preconditioned minimal residual method for the solution of the sparse nonsymmetric linear system has been presented by Xu *et al.* [7] who suggested that even with a simple preconditioned matrix of L, like a block diagonal matrix, the convergence rate has been significantly improved by adding a parameter $\lambda(0 < \lambda < 1)$ to the product of the inverse of the preconditioned matrix L_{ap} and matrix L. But in order to minimize the storage size ($O(N^3)$ in 3D) required for the iterative pressure solution the eigenfunction expansion technique [8] previously proposed by the author will be incorporated into the multi-level (two-level) preconditioned minimal residual scheme of Xu *et al.* [7].

The Schwarz alternating procedure (SAP) iterative scheme has long been used for the solution of boundary value problem, solving problems on each subdomain separately and updating the function value on the overlapped interfaces. The convergence rate is proportional to the overlapping area; the more overlapping area, the faster the convergence. The advantage of the SAP scheme is easy to implement, but the iterative solution for data exchange among subdomains is its disadvantage. However, the storage space toward the matrix manipulation, local rather than global, is one of the least expensive schemes that ever exist in nature. The SAP iterative scheme can also be applied to the interface with patched grid, where the boundary information needs to be calculated through the interpolation by extending one grid into the neighboring subdomain, except the convergence rate is slow.

The SAP iterative scheme has been successfully applied to those configurations where the overlapped grids coincide with each other [9, 10] (referred to as a single grid hereafter). But under some circumstances, due to the complexity of the geometrical configuration, such as the submarine and automobile, possible layout of mixed types of grids like Cartesian, ''O'' and ''C'' among the subdomains, or the necessity of applying adaptive fine grids for high resolution in one subdomain and coarse grids for less resolution in other subdomains, the overlapped grids cannot be collapsed at the same position. As pointed out by the authors [11] just by simply exchanging the data through the interpolation in the inter-overlapping areas the high frequency error introduced by the fine-grid subdomain will pollute the results throughout the whole computational domain; it will be especially pronounced for the flow at high Reynolds numbers. The multi-grid technique, which has long been advocated by the finite-difference users [12, 13], employs a sequence of grids to accelerate the convergence of iterative methods. The work rests on ''standard coarsening,'' i.e., doubling the mesh in each directions from one grid to the next coarsest grid. Solve the problem on the coarse grid (low frequency domain) and the coarse-grid correction transfers back (prolongation) to the fine grid (high frequency domain) to gain rapid convergence. The novel multi-grid (two grid) SAP [11] which was strictly used in Cartesian coordinates will be adopted in this paper to eliminate the high frequency error that results from the data interpolation from the fine-grid (''O'' grid) subdomain to the coarse-grid (Cartesian grid) subdomain.

The paper consists of six additional sections. In Section 2, a description of three-dimensional grid generation will be addressed by the isoparametric mapping technique. Section 3 derives a primitive variable form of the Navier–Stokes equations in curvilinear coordinates. Section 4 constructs the preconditioned matrix which facilitates the solution of a three-dimensional pressure equation. Section 5 discusses the multi-grid SAP to iteratively solve the inter-overlapping subdomains where the overlapped grids are not located at the same places. Section 6 presents the detailed numerical results of proposed 2D and 3D problems, and a final section provides the conclusions.

2. ISOPARAMETRIC MAPPING

Let us first define the existence of a mapping function between the physical space (x, y, z) and the computational space (ξ, η, ζ) (a transformed space with Cartesian coordinates). Once such coordinate relationships are known, shape functions defining the geometry can be specified in local coordinates and a one-to-one correspondence between Cartesian and curvilinear coordinates can be established.

An isoparametric mapping, the same set of functions (shape functions) used to define the field variables in an element also being used to define the geometry is applied to map a three-dimensional curved geometry (physical space) onto a cube (computational space). The main objective of the present approach is to provide the three-dimensional computational grids around complex geometries in a structured fashion. The grid generation scheme presented here utilizes a multiple block structure; namely, the global computational domain based on the geometrical configuration is divided into a number of blocks which overlap or not. Each block is then partitioned by the pseudospectral elements. Implicitly, the grid generation is performed in two levels. First, each of these blocks is defined as a parent element. In other words, a parent element in the transformed space, $-1 \le \xi \le 1$, $-1 \le \eta \le 1$, $-1 \le \zeta \le 1$ (Fig. 1), corresponds to an irregular or regular six-faced (hexahedrael) element (block) in the physical space. Next, family elements, each of which contains $N\xi + 1$, $N\eta + 1$, and $N\zeta + 1$ collocation points ($\xi_i = \frac{1}{2}[\cos \pi i/N\xi(\xi_2 - \xi_1) + \xi_2 + \xi_1]$, $\eta_j = \frac{1}{2}[\cos \pi j/N\eta(\eta_2 - \eta_1) + \eta_2 + \eta_1]$, $\zeta_k = \frac{1}{2}[\cos \pi k/N\zeta)(\zeta_2 - \zeta_1) + \zeta_2 + \zeta_1]$), where $\xi_1 \le \xi_i(i = 0, ..., N\ \xi) \le \xi_2$, $\eta_1 \le \eta_j(j = 0, ..., N\ \eta) \le \eta_2$, $\zeta_1 \le \zeta_k(k = 0, ..., N\ \zeta) \le \zeta_2$, linearly (or higher order) interpolating the shape function defining the geometry of their parent elements (or blocks) are allocated within each of these blocks. Once the collocation points (x, y, z) along the curvy boundaries of each parent element are known, the interior points (including the boundaries of the family elements) are interpolated by deforming the (ξ, η, ζ) mesh into its (x, y, z) image using the ''trilinear blending function'' [14], i.e., the grid points $(x, y, z)_{ijk}$ in the physical space are mapped

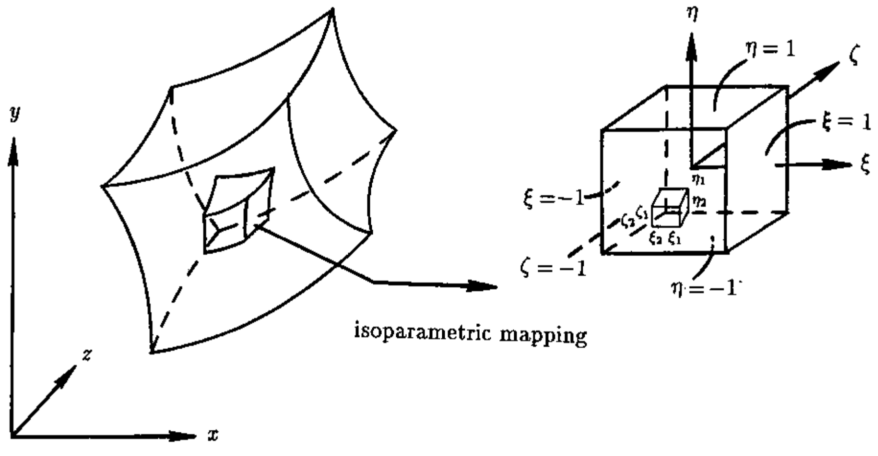

FIG. 1. Three-dimensional isoparametric mapping of family elements.

onto ($\xi = \xi_i$, $\eta = \eta_j$, $\zeta = \zeta_k$) in the transformed space. The following summarizes the technique of an isoparametric mapping for grid generation:

Let ϕ be any value of (x, y, z); the interpolation translates the Booleam sum [15] into a form

$$\begin{aligned}\phi = P_\xi\phi + P_\eta\phi + P_\zeta\phi - P_\xi P_\eta\phi - P_\xi P_\zeta\phi \\ - P_\eta P_\zeta\phi + P_\xi P_\eta P_\zeta\phi,\end{aligned} \tag{1}$$

where the "projectors" P_ξ, P_η, P_ζ interpolate ϕ between two opposing faces of the six-sided region, the double product projector, $P_\xi P_\eta$, interpolate ϕ in two directions from the four edges along which ξ and η are constant, and the triple product projector, $P_\xi P_\eta P_\zeta$ interpolates ϕ from the eight corners.

With linear interpolation functions defined as

$$\begin{aligned} N^{(1)}(\xi) &= \frac{1+\xi}{2}, \quad N^{(2)}(\xi) = \frac{1-\xi}{2} \\ N^{(1)}(\eta) &= \frac{1+\eta}{2}, \quad N^{(2)}(\eta) = \frac{1-\eta}{2} \\ N^{(1)}(\zeta) &= \frac{1+\zeta}{2}, \quad N^{(2)}(\zeta) = \frac{1-\zeta}{2}, \end{aligned} \tag{2}$$

the explicit expressions for the face projectors are

$$\begin{aligned} P_\xi\phi &= N^{(1)}(\xi)\phi(1, \eta, \zeta) + N^{(2)}(\xi)\phi(-1, \eta, \zeta) \\ P_\eta\phi &= N^{(1)}(\eta)\phi(\xi, 1, \zeta) + N^{(2)}(\eta)\phi(\xi, -1, \zeta) \\ P_\zeta\phi &= N^{(1)}(\zeta)\phi(\xi, \eta, 1) + N^{(2)}(\zeta)\phi(\xi, \eta, -1); \end{aligned} \tag{3}$$

and the edge projectors are

$$\begin{aligned} P_\xi P_\eta\phi = {} & N^{(1)}(\xi)N^{(1)}(\eta)\phi(1, 1, \zeta) + N^{(1)}(\xi)N^{(2)}(\eta)\phi(1, -1, \zeta) \\ & + N^{(2)}(\xi)N^{(1)}(\eta)\phi(-1, 1, \zeta) \\ & + N^{(2)}(\xi)N^{(2)}(\eta)\phi(-1, -1, \zeta) \\ P_\xi P_\zeta\phi = {} & N^{(1)}(\xi)N^{(1)}(\zeta)\phi(1, \eta, 1) + N^{(1)}(\xi)N^{(2)}(\zeta)\phi(1, \eta, -1) \\ & + N^{(2)}(\xi)N^{(1)}(\zeta)\phi(-1, \eta, 1) \\ & + N^{(2)}(\xi)N^{(2)}(\zeta)\phi(-1, \eta, -1) \\ P_\eta P_\zeta\phi = {} & N^{(1)}(\eta)N^{(1)}(\zeta)\phi(\xi, 1, 1) + N^{(1)}(\eta)N^{(2)}(\zeta)\phi(\xi, 1, -1) \\ & + N^{(2)}(\eta)N^{(1)}(\zeta)\phi(\xi, -1, 1) \\ & + N^{(2)}(\eta)N^{(2)}(\zeta)\phi(\xi, -1, -1); \end{aligned} \tag{4}$$

and the corner projector is

$$\begin{aligned} P_\xi P_\eta P_\zeta\phi = {} & N^{(1)}(\xi)N^{(1)}(\eta)N^{(1)}(\zeta)\phi(1, 1, 1) \\ & + N^{(1)}(\xi)N^{(1)}(\eta)N^{(2)}(\zeta)\phi(1, 1, -1) \\ & + N^{(1)}(\xi)N^{(2)}(\eta)N^{(2)}(\zeta)\phi(1, -1, -1) \\ & + N^{(1)}(\xi)N^{(2)}(\eta)N^{(1)}(\zeta)\phi(1, -1, 1) \\ & + N^{(2)}(\xi)N^{(1)}(\eta)N^{(1)}(\zeta)\phi(-1, 1, 1) \\ & + N^{(2)}(\xi)N^{(1)}(\eta)N^{(2)}(\zeta)\phi(-1, 1, -1) \\ & + N^{(2)}(\xi)N^{(2)}(\eta)N^{(1)}(\zeta)\phi(-1, -1, 1) \\ & + N^{(2)}(\xi)N^{(2)}(\eta)N^{(2)}(\zeta)\phi(-1, -1, -1); \end{aligned} \tag{5}$$

where the surface function $\phi(\xi, \eta, 1)$ can be adequately represented by an isoparametric tensor product such that

$$\phi(\xi, \eta, 1) = \sum_{i=0}^{N} \sum_{j=0}^{N} N_i(\xi)N_j(\eta)\phi(\xi_i, \eta_j, 1) \tag{6a}$$

and, similarly for the edge function $\phi(\xi, 1, 1)$,

$$\phi(\xi, 1, 1) = \sum_{i=0}^{N} N_i(\xi)\phi(\xi_i, 1, 1). \tag{6b}$$

Here $N_i(\xi)$, $N_j(\eta)$, the shape functions of each parent element derived from the Chebyshev polynomials, read

$$N_i(\xi) = \sum_{m=0}^{N} T_m(\xi)\hat{T}_m(\xi_i) \tag{7a}$$

$$N_j(\eta) = \sum_{n=0}^{N} T_n(\eta)\hat{T}_n(\eta_j), \tag{7b}$$

where the matrices $T_m(\xi)$ and $\hat{T}_m(\xi)$ are the Fourier cosine series and their inverse [8]. Note that the shape functions $N_i(\xi)$, $N_j(\eta)$ satisfy the Kronecker-delta property, i.e., $N_i(\xi_m) = \delta_{im}$, $N_j(\eta_n) = \delta_{jn}$.

In an analogous manner, other undefined surface and edge functions can be derived without any difficulty. It is obvious that Eq. (6a) interpolates the surface boundary functions exactly.

Once the computational grids have been generated according to the aforementioned isoparametric mapping technique, the partial derivatives of a function f in the physical space (x, y, z) can be obtained through the general coordinate transformation; i.e., on the transformed coordinates, the partial derivatives f_x, f_{xx} can be represented by

$$f_x = \xi_x f_\xi + \eta_x f_\eta + \zeta_x f_\zeta \tag{8a}$$

$$f_{xx} = \xi_x (f_x)_\xi + \eta_x (f_x)_\eta + \zeta_x (f_x)_\zeta, \tag{8b}$$

where the subscript denotes the derivative and ξ_x, η_x, ζ_x are the metric coefficients, of which the explicit formulae will be described in the section of primitive variable formulation. The interfacial derivatives at the inter-element points between element e and $e + 1$, or at the inter-block points, are approximated by weighting of the derivatives from each side, namely,

$$f_\xi|_{\text{interface}} = \alpha f_\xi^e + \beta f_\xi^{e+1} \tag{9a}$$

$$f_{\xi\xi}|_{\text{interface}} = \alpha f_{\xi\xi}^e + \beta f_{\xi\xi}^{e+1}. \tag{9b}$$

Choosing α and β to be their respective fraction of the total length of two adjacent elements as described in [9], c^0 continuity is implicitly satisfied for the second derivative calculation, while c^0 continuity is explicitly assumed whenever the calculation of interface values of the first derivative is required. Similarly, c^0 continuity for $f_{\eta\eta}$, $f_{\zeta\zeta}$ stand, too.

In order to set up an efficient matrix operation for derivatives $f_\xi, f_{\xi\xi}, f_\eta, f_{\eta\eta}, f_\zeta, f_{\zeta\zeta}$ by the pseudospectral element (PSE) method, a global-typed differentiation operator, including those blocks with same local coordinates, can be constructed by combining each local element derivative [9].

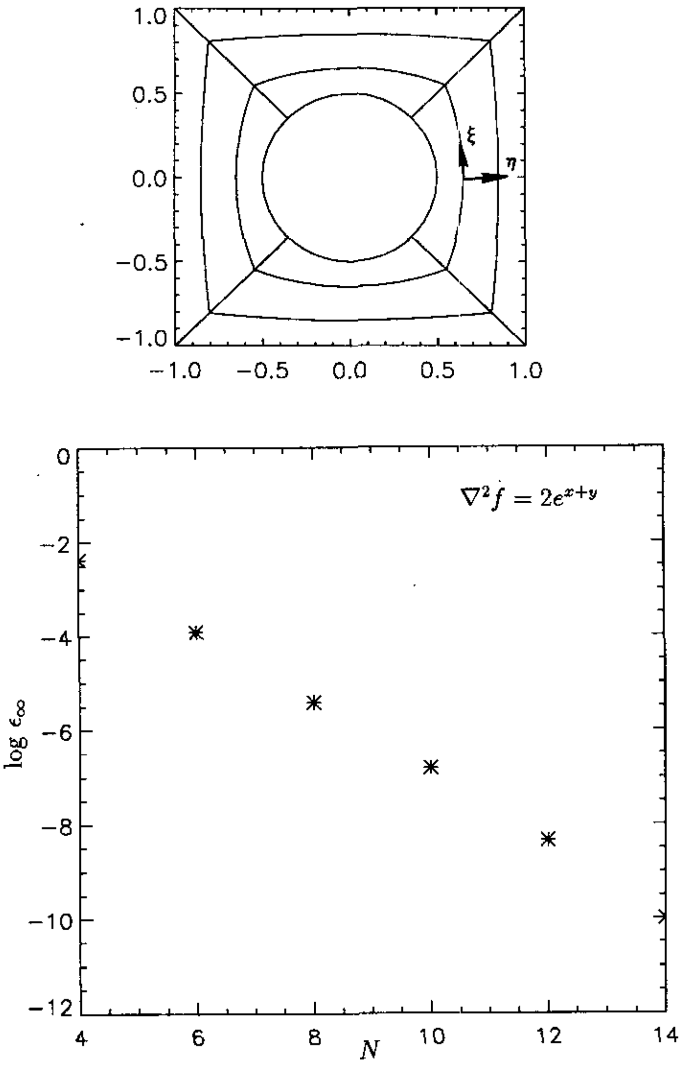

FIG. 2. A plot of the approximation error with four elements in the ξ direction and three elements in the η direction. Here $\varepsilon_\infty = \|f_{\text{numerical}} - f_{\text{exact}}\|_\infty$, and N is the number of points per element.

The proposed PSE method was tested on the two-dimensional Poisson equation, $\nabla^2 f = 2e^{x+y}$, based on the configuration shown in Fig. 2. In conformity with the description in this section, Fig. 2 depicts that four blocks (or parent elements) are chosen in the ξ (circumferential) direction, and each block contains three family elements in the η direction, i.e., $\eta = 1$, 0.4, −0.4, and −1. The numerical results, compared with the exact solution $f = e^{x+y}$, exhibit exponential convergence with an increasing number of points per element.

3. PRIMITIVE VARIABLE FORMULATION

In tensor notation, the time-dependent Navier–Stokes equations in dimensionless form can be described as

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{1}{\mathrm{Re}} \frac{\partial^2 u_i}{\partial x_j^2} \tag{10a}$$

$$\frac{\partial u_i}{\partial x_i} = 0. \tag{10b}$$

Here u_i is the velocity component and Re is the Reynolds number.

The method applied to solve the Navier–Stokes equations is Chorin's [3] splitting technique. According to this scheme, the equations of motion read

$$\frac{\partial u_i}{\partial t} + \frac{\partial p}{\partial x_i} = F_i, \tag{11}$$

where $F_i = -u_j\, \partial u_i/\partial x_j + (1/\mathrm{Re})\, \partial^2 u_i/\partial x_j^2$.

The first step is to split the velocity into a sum of predicted and corrected values. The predicted velocity is determined by time integration of the momentum equations without the pressure term

$$\overline{u}_i^{n+1} = u_i^n + \Delta t F_i^n. \tag{12}$$

The second step is to develop the pressure and corrected velocity fields that satisfy the continuity equation by using the relationships

$$u_i^{n+1} = \overline{u}_i^{n+1} - \Delta t \frac{\partial p}{\partial x_i} \tag{13a}$$

$$\frac{\partial u_i^{n+1}}{\partial x_i} = 0. \tag{13b}$$

Here the superscript n denotes the nth time step. Note that the size of a stable time-step Δt can be increased by using an adaptation of Runge–Kutta techniques [16] for the high Reynolds number and the Stokes solution for the low Reynolds number [9], respectively.

Using the explicit scheme of Chorin's splitting technique, the calculation of the predicted velocity, $\overline{u}_i$, in the three-dimensional transformed coordinates ξ, η, ζ is straightforward. The corrected velocity, Eq. (13a), and the continuity equation, Eq. (13b), now becomes

$$u = \overline{u} - \Delta t(\xi_x p_\xi + \eta_x p_\eta + \zeta_x p_\zeta) \tag{14a}$$

$$v = \overline{v} - \Delta t(\xi_y p_\xi + \eta_y p_\eta + \zeta_y p_\zeta) \tag{14b}$$

$$w = \overline{w} - \Delta t(\xi_z p_\xi + \eta_z p_\eta + \zeta_z p_\zeta) \tag{14c}$$

and

$$\begin{aligned} &\xi_x u_\xi + \eta_x u_\eta + \zeta_x u_\zeta + \xi_y v_\xi + \eta_y v_\eta + \zeta_y v_\zeta \\ &\quad + \xi_z w_\xi + \eta_z w_\eta + \zeta_z w_\zeta = 0, \end{aligned} \tag{15}$$

where u, v, w are the cartesian velocity components in the x, y, and z directions, respectively, and the coefficients inside the brackets of Eqs. (14) correspond to the pressure gradients p_x, p_y, and p_z. The derivatives of the metric coefficients $\xi(x, y, z)$, $\eta(x, y, z)$, and $\zeta(x, y, z)$ related to the transformed coordinates $x(\xi, \eta, \zeta)$, $y(\xi, \eta, \zeta)$, and $z(\xi, \eta, \zeta)$ are given by

$$\begin{aligned} &\xi_x = J^{-1}(y_\eta z_\zeta - y_\zeta z_\eta), \quad \xi_y = J^{-1}(x_\zeta z_\eta - x_\eta z_\zeta), \\ &\xi_z = J^{-1}(x_\eta y_\zeta - x_\zeta y_\eta) \\ &\eta_x = J^{-1}(y_\zeta z_\xi - y_\xi z_\zeta), \quad \eta_y = J^{-1}(x_\xi z_\zeta - x_\zeta z_\xi), \\ &\eta_z = J^{-1}(x_\zeta y_\xi - x_\xi y_\zeta) \\ &\zeta_x = J^{-1}(y_\xi z_\eta - y_\eta z_\xi), \quad \zeta_y = J^{-1}(x_\eta z_\xi - x_\xi z_\eta), \\ &\zeta_z = J^{-1}(x_\xi y_\eta - x_\eta y_\xi) \end{aligned} \tag{16}$$

and the Jacobian is $J = x_\xi y_\eta z_\zeta + x_\zeta y_\xi z_\eta + x_\eta y_\zeta z_\xi - x_\xi y_\zeta z_\eta - x_\eta y_\xi z_\zeta - x_\zeta y_\eta z_\xi$.

Taking the divergence operator to Eqs. (14a), (14b), (14c) in order to satisfy the continuity equation (15), the pressure equation in the interior has the form

$$\begin{aligned} &\xi_x(p_x)_\xi + \eta_x(p_x)_\eta + \zeta_x(p_x)_\zeta + \xi_y(p_y)_\xi + \eta_y(p_y)_\eta \\ &\quad + \zeta_y(p_y)_\zeta + \xi_z(p_z)_\xi + \eta_z(p_z)_\eta + \zeta_z(p_z)_\zeta \\ &\quad = \Delta t^{-1}\{\xi_x \overline{u}_\xi + \eta_x \overline{u}_\eta + \zeta_x \overline{u}_\zeta + \xi_y \overline{v}_\xi \\ &\qquad + \eta_y \overline{v}_\eta + \zeta_y \overline{v}_\zeta + \eta_z \overline{w}_\eta + \zeta_z \overline{w}_\zeta\} \end{aligned} \tag{17}$$

and the supplemental pressure equations at the boundaries are

$$\begin{aligned} &\xi_x(p_x)_\xi + \xi_y(p_y)_\xi + \xi_z(p_z)_\xi \\ &\quad = \Delta t^{-1}\{\xi_x \overline{u}_\xi + \eta_x u_\eta + \zeta_x u_\zeta + \xi_y \overline{v}_\xi + \eta_y v_\eta + \zeta_y v_\zeta \\ &\qquad + \xi_z \overline{w}_\xi + \eta_z w_\eta + \zeta_z w_\zeta\} \end{aligned} \tag{18a}$$

in the ξ direction, as well as

$$\begin{aligned} &\eta_x(p_x)_\eta + \eta_y(p_y)_\eta + \eta_z(p_z)_\eta = \Delta t^{-1}\{\xi_x u_\xi + \eta_x \overline{u}_\eta + \zeta_x u_\zeta \\ &\quad + \xi_y v_\xi + \eta_y \overline{v}_\eta + \zeta_y v_\zeta + \xi_z w_\xi + \eta_z \overline{w}_\eta + \zeta_z w_\zeta\} \end{aligned} \tag{18b}$$

in the η direction and

$$\begin{aligned} &\zeta_x(p_x)_\zeta + \zeta_y(p_y)_\zeta + \zeta_z(p_z)_\zeta = \Delta t^{-1}\{\xi_x u_\xi + \eta_x u_\eta + \zeta_x \overline{u}_\zeta \\ &\quad + \xi_y v_\xi + \eta_y v_\eta + \zeta_y \overline{v}_\zeta + \xi_z w_\xi + \eta_z w_\eta + \zeta_z \overline{w}_\zeta\} \end{aligned} \tag{18c}$$

in the ζ direction. Note that whenever solving Eqs. (17) and (18) the identity of Eqs. (14) should be utilized to absorb the given boundary conditions of the velocity components [8].

4. ITERATIVE PRECONDITIONED METHOD

If p satisfies Eqs. (17) and (18), then $\mathbf{u}^{n+1}$ does indeed satisfy Eq. (13b). The solution of the pressure equations (17) and (18) is the most computationally expensive step. For some simple geometries it can be directly solved numerically by separation of variables [8]. In general there is no simple method for direct solution and one must resort to iterative techniques. The quality of a solution is directly related to the accuracy with which the incompressibility condition is satisfied. Eq. (17) is of the general form

$$Lp = S \tag{19}$$

for some linear operator L on some finite dimensional vector space. The properties of the operator L depend on the methods chosen to represent the fields and their derivatives. Very effective techniques are known in the event that L is positive definite and symmetric.

Although in general the operators L arising in our work are not positive definite symmetric, the alternative is to actually solve an approximate problem whose solution can be easily related to that of the original problem. This technique is called preconditioning. There have been a lot of efforts to choose efficient preconditioned matrices, such as finite difference method [4, 5] or finite element method [17, 18] for spectral solutions of Eq. (19). We have adapted the preconditioned minimal residual methods [4] to our work as follows. Instead of Eq. (19) we solve

$$L_{ap}^{-1} Lp = L_{ap}^{-1} S, \tag{20}$$

where the preconditioner L_{ap}, a certain separable operator which allows a fast solution [19], can be chosen and constructed from the original operator L of Eq. (17), by linearizing every nonconstant metric coefficients; i.e., $\xi_x(\xi, \eta, \zeta)$, $\xi_\eta(\xi, \eta, \zeta)$, and $\xi_\zeta(\xi, \eta, \zeta)$ are approximated by a function of $\tilde{\xi}_x(\xi)$, $\tilde{\xi}_y(\xi)$, and $\tilde{\xi}_z(\xi)$ alone, and so are the other metric coefficients. The chosen structure of L_{ap} should be easily invertable by an eigenfunction technique such that

$$\begin{aligned} L_{ap} p = {} & \tilde{\xi}_x(\xi)[\tilde{\xi}_x(\xi) p_\xi]_\xi + \tilde{\xi}_y(\xi)[\tilde{\xi}_y(\xi) p_\xi]_\xi + \tilde{\xi}_z(\xi)[\tilde{\xi}_z(\xi) p_\xi]_\xi \\ & + \tilde{\eta}_x(\eta)[\tilde{\eta}_x(\eta) p_\eta]_\eta + \tilde{\eta}_y(\eta)[\tilde{\eta}_y(\eta) p_\eta]_\eta + \tilde{\eta}_z(\eta)[\tilde{\eta}_z(\eta) p_\eta]_\eta \\ & + \tilde{\zeta}_x(\zeta)[\tilde{\zeta}_x(\zeta) p_\zeta]_\zeta + \tilde{\zeta}_y(\zeta)[\tilde{\zeta}_y(\zeta) p_\zeta]_\zeta + \tilde{\zeta}_z(\zeta)[\tilde{\zeta}_z(\zeta) p_\zeta]_\zeta, \end{aligned} \tag{21}$$

where ξ_x, ξ_y, ξ_z, η_x, η_y, η_z, ζ_x, ζ_y, ζ_z are metric coefficients.

A good preconditioner requires (i) less memory and inexpensive effort to invert the resulting matrix and (ii) a fast convergence rate. The second requirement implies that the preconditioner L_{ap} should be close to the original operator L; i.e., the spectral condition number κ, the ratio of the maximum and minimum eigenvalues of $\|L_{ap}^{-1} L\|$, should not be large. The iterative procedure then reads as follows:

Given p^0, compute $r^0 = S - Lp^0$, $z^0 = L_{ap}^{-1} r^0$, $h^0 = z^0$. Then, for $k = 0, 1, 2, \ldots$, until $\|r^k\| < \varepsilon$, do

$$p^{k+1} = p^k + \alpha^k h^k \tag{22a}$$

$$r^{k+1} = r^k - \alpha^k L h^k \tag{22b}$$

$$z^{k+1} = L_{ap}^{-1} r^{k+1} \tag{22c}$$

$$h^{k+1} = z^{k+1} - \beta^k h^k, \tag{22d}$$

where

$$\alpha^k = \frac{(r^k, Lh^k)}{(Lh^k, Lh^k)}, \quad \beta^k = \frac{(Lz^{k+1}, Lh^k)}{(Lh^k, Lh^k)}. \tag{22e}$$

Here, (,) denotes the inner product. Let $\mathbf{z}^k$ in Eq. (22c), $k \geq 1$, be expanded in a series of eigenfunctions such that

$$\mathbf{z}^k = \mathbf{E}\xi\, \hat{\mathbf{z}}^k\, \mathbf{E}\eta^{\mathrm{T}}\, \mathbf{E}\zeta^{\mathrm{T}}, \tag{23a}$$

and, similarly, the residual $\mathbf{r}^k$ is expanded such that

$$\mathbf{r}^k = \mathbf{E}\xi\, \hat{\mathbf{r}}^k\, \mathbf{E}\eta^{\mathrm{T}}\, \mathbf{E}\zeta^{\mathrm{T}}. \tag{23b}$$

With the eigenfunction expansion technique, the solution of the three-dimensional preconditioned Eq. (22c) can be reduced to the simplest algebraic form,

$$(\alpha_i + \beta_j + \gamma_k) \hat{z}^k_{i,j,k} = r^k_{i,j,k}, \tag{24}$$

where α_i, β_j, and γ_k are the eigenvalues with respect to the discrete derivative matrices of the linear preconditioner, L_{ap} of Eq. (21), and $\mathbf{E}\xi$, $\mathbf{E}\eta$, $\mathbf{E}\zeta$ are the corresponding eigenvectors associated with each eigenvalue. However, eigenvalues may not be real due to the complexity of a preconditioner. Without putting any restriction on eigenvalues, complex eigenvalues and their associated eigenvectors are permitted if the pressure gradient at the imaginary part vanishes. This is true because only the pressure gradient drives flow, instead of the pressure itself. Note that if there are N degrees of freedom in each direction the overall memory required for finding the solution to the pressure equation in three dimensions is $O(N^3)$. This is the same type of maximally storage efficient scaling that we have for the velocity field.

Based on the Xu *et al.* [7] multi-level preconditioned minimal residual method the adding parameter λ ($0 < \lambda < 1$) improves the preconditioned matrix property (especially pronounced for those problems hard to reach convergence), $\alpha I + L_{ap}^{-1} L$, rather than the original $L_{ap}^{-1} L$, which enables a successful implementation of the preconditioned minimal residual method. In view of this scheme, the pressure equation, Eq. (20), can be adequately replaced by the following form, accordingly,

$$(\lambda I + L_{ap}^{-1} L) p^{m+1} = L_{zp}^{-1} S + \lambda p^m. \tag{25}$$

It is obvious that two iteration loops are required, the outer loop "m" (damping the Jacobi method) and the inner loop "k" the preconditioned minimal residual method alone). The parameter λ $(0 < \lambda < 1)$ is determined by a balanced convergence between these two loops.

Without going through a complicated flow problem, let us examine the same test problem (Poisson equation) in Section 2 by utilizing 49 × 37 points. Starting from the "zero" guess, it takes 20, 40, 60, 80, 100, 120 iterations to reach $\|Lp - S\| = 3.48\text{E} + 1$, 6.01E-1, 1.04E-2, 8.64E-5, 1.34E-7, and 2.63E-10, respectively. The λ effect is not obvious due to the smooth convergence.

In practical application, the λ effect still takes into account the ill-conditioned matrix resulting from the pressure equation. The λ is chosen as 0.75 in all numerical experiments, while the choice of the three outer loops and four to six inner loops will be sufficient to reach the convergence criterion.

5. DOMAIN DECOMPOSITION WITH MULTI-GRID SAP

The solution of flow over complex geometry via the domain decomposition approach consists of first dividing the computational domain into a number of blocks (or subdomains) with inter-overlapping areas, where the grids inside the overlapping area may or may not be located at the same places. Next implement the SAP for exchanging data among blocks, i.e., solving the problem on each block separately and then updating the velocity field on the overlapped interfaces. The extension of the SAP technique for the solution of incompressible flow in curvilinear coordinates will be straightforward by the proposed preconditioned method. The advantages of this approach include (i) less memory access, local rather than global memory, and (ii) easy treatment of complex geometry.

The success of the single-grid SAP scheme applied to the incompressible flow lies in the unique use of the continuity equation as the pressure boundary condition (high-order gradient) in the overlapping area [9]. The residual (l_2 norm) of the flow field, the velocity difference between two subdomains in the overlapping area, can be reduced by one order of magnitude. In most cases, it takes four SAP iterations to sufficiently mark down the residual index to $O(10^{-4})$–$O(10^{-5})$. In addition to the Lagrangian constraint between the pressure and velocity fields, the noncoincident overlapped grids in the inter-overlapping areas among subdomains even enhances the difficulty of applying the multi-grid technique. However, the idea of "coarse-grid correction" is still effective to reduce the high frequency error from the interpolated residual of the fine-grid subdomain. The strategy behind the coarse-grid correction process is to adopt the idea proposed by Thompson and Ferziger [20] and is modified by [11] as

$$\nabla_c \cdot \mathbf{u}_c - \nabla_c \cdot (I_c^f \mathbf{u}_f) = I_c^f (r_f - \nabla_f \cdot \mathbf{u}_f). \qquad (26)$$

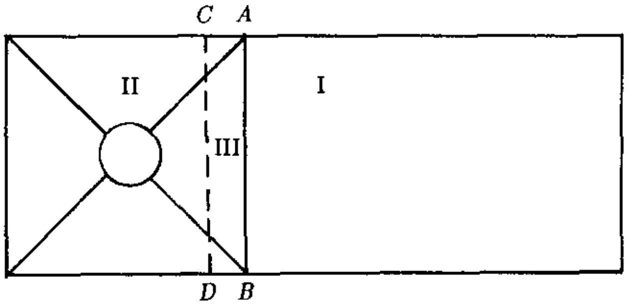

FIG. 3. Configuration of domain decomposition (multiple blocks).

Here $\nabla_c \cdot$ represents the operator of divergence on the coarse-grid subdomain, I_c^f is an interpolation operator from the fine-grid subdomain f to coarse grid subdomain c, and $\mathbf{u}$ is the velocity components; r_f is simply the result of the divergence of the velocity field which should be set to zero. The left-hand side of Eq. (26) is the difference between the coarse-grid operator acting on the coarse-grid subdomain and the coarse-grid operator acting on the interpolated fine-grid subdomain (which is held fixed). The right-hand side of Eq. (26) is the interpolated residual of the fine-grid subdomain. It is obvious that, once the solution of the fine-grid subdomain has been found, the residual will be zero (exactly satisfy the pressure Poisson equation), and it implies that

$$\mathbf{u}_c = I_c^f \mathbf{u}_f. \qquad (27)$$

When the residual is non-zero, Eq. (26) acts as a forcing term for the coarse-grid correction to transfer the correction of velocity field back to the fine-grid subdomain, i.e.,

$$\mathbf{u}_f^{\text{new}} = \mathbf{u}_f^{\text{old}} + I_f^c (\mathbf{u}_c - I_c^f \mathbf{u}_f^{\text{old}}). \qquad (28)$$

This is vital for the success of the scheme. Changes in the velocity field are transferred back to fine-grid subdomain rather than the velocity field itself. Note that when the overlapped grids in the overlapping areas are collapsed at the same places the interpolation operator I_c^f becomes a unitary matrix.

The multi-grid SAP iterative solution of the incompressible Navier–Stokes equation in primitive variable form for flow over a cylinder sketched in Fig. 3, for example, is summarized by the following algorithm:

1. First assume that $\mathbf{u}^{n+1}$ is on $\overline{AB}$. Usually $\mathbf{u}^n$ will be a good initial guess.
2. Solve fine-grid domain II, employing the boundary conditions derived from the divergence of the velocity field, including on $\overline{AB}$, where the pressure solution is obtained by the preconditioned method.
3. With the interpolated solution of $\mathbf{u}^{n+1}$ from step 2 on domain III $\subset$ I, solve the coarse-grid domain I, employing the

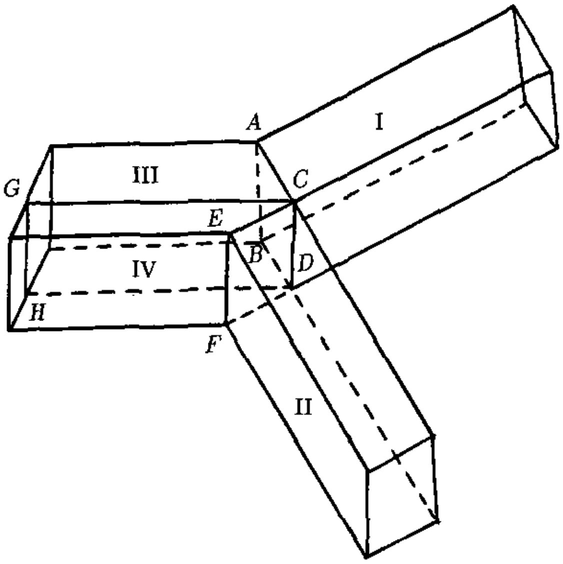

FIG. 4. Three-dimensional configuration of domain decomposition.

same type of boundary conditions, including on $\overline{CD}$, to update $\mathbf{u}^{n+1}$ on domain III $\subset$ II by the coarse-grid correction process.

4. Repeat steps 2 and 3 until the velocity $\mathbf{u}^{n+1}$ on $\overline{AB}$, $\overline{CD}$ does not change.

In order to guarantee that consistent values of velocity (or pressure gradient) be generated in the overlapping domains III, satisfying Eq. (27), the divergence of velocity field $\nabla \cdot \mathbf{u}$ needs to be actually computed in whichever domain I or II is counted [21]. Since $\mathbf{u}$ on domains III is not known a priori, the divergence of the velocity field is only set to zero at the first SAP iteration for step 2. According to this approach, the continuity equation is satisfied on domains I and II, but not on domain III $\subset$ I, which is revealed from Eq. (26) that the continuity equation is only satisfied on the fine-grid domain II. However, the error index of the continuity equation on domain III $\subset$ I will indicate how good the interpolation is (affected by the layout of overlapped grids) and whether any steep change of flow field exists.

The single-grid SAP iterative solution of the three-dimensional bifurcation flow (Fig. 4) is implemented in a similar way and is illustrated as follows:

1. First assume that $\mathbf{u}^{n+1}$ on $\square ABCD$ and $\square CDEF$. Usually $\mathbf{u}^{n}$ will be a good initial guess.

2. Solve domain III $\cup$ IV employing the boundary conditions derived from the divergence of the velocity field on $\square ABCD$ and $\square CDEF$, where the pressure solution is obtained by the preconditioned method.

3. With the solution of $\mathbf{u}^{n+1}$ on $\square CDGH$ from step 2, solve domain I $\cup$ III and II $\cup$ IV, employing the same type boundary conditions on $\square CDGH$ to update $\mathbf{u}^{n+1}$ on $\square ABCE$ and $\square CDEF$.

4. Repeat steps 2 and 3 until the velocity $\mathbf{u}^{n+1}$ on $\square ABCD$, $\square CDEF$, $\square CDGH$ does not change.

6. RESULTS AND DISCUSSION

6.1. *Radiation Boundary Conditions*

Before going through the numerical results of tested problems, let us first focus on the issue of what kind of downstream boundary conditions should be applied. As we know, appropriate outflow boundary conditions are required when the calculations are performed in open domains without making a fully developed flow profile assumption far downstream, and these conditions should have little influence on the upstream flow development. Most often used downstream boundary conditions on the truncated domain, the zero of the first or second derivative of the velocity field in the flow normal direction ($\partial \mathbf{u}/\partial n = 0$ or $\partial^2 \mathbf{u}/\partial n^2 = 0$), are not valid for strong shedding or wave-like flow. Nevertheless, so far the best downstream condition treating wave flow is the Sommerfeld radiation condition [22]. It prescribes that

$$\frac{\partial \phi}{\partial t} + C\frac{\partial \phi}{\partial n} = 0, \tag{29}$$

where ϕ is any variable and C is the phase velocity of waves. However, it is difficult to find the accurate phase speed, and numerous varieties to approximate the phase speed have been given by [22], but none of them can adequately impose the global mass conservation constraint. Gresho [23] further modified the Sommerfeld radiation condition as

$$\frac{\partial u_i}{\partial t} + C\frac{\partial u_i}{\partial n} = 0. \tag{30}$$

Here the constant phase speed C is taken as the average speed of streamwise flow at the downstream position. When integrating Eq. (30) over the cross section where the flow passes through, it is apparent that the downstream velocity predicted according to Eq. (30) indeed satisfies the global mass conservation. Thus, Eq. (30) will be utilized to prescribe the downstream flow conditions for all the numerical simulations.

6.2. *Flow over a Cylinder*

Davis *et al.* [24] investigated the effect of confined walls (the blockage ratio) on the vortex shedding behind a square

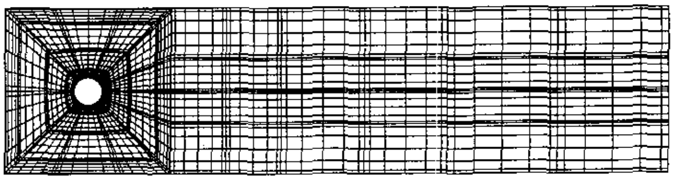

FIG. 5. Grid generation by the isoparametric mapping method.

TABLE I
Summary Chart of Computational Data

Re	100	250	500	1000
Elements[a]	6 × 4, 12 × 5	13 × 8, 16 × 6	16 × 8, 16 × 9	17 × 9, 17 × 10
Δt	0.006	0.005	0.005	0.007
C_D	1.330–1.358	1.185–1.344	1.212–1.481	1.187–1.651
C_L	−0.228–0.228	−0.650–0.650	−1.030–1.030	−1.242–1.242
Strouhal number	0.1675	0.2045	0.2203	0.2326

[a] Six points per element.

cylinder up to the Reynolds number of 1000. They also studied the effect on the resulting forces experienced by the body. For flow passing over a circular cylinder few papers [25, 26] have been experimental in nature and have generally studied the turbulent characteristics for the high Reynolds number, but the study on the laminar flow is far less addressed. The blockage ratio is chosen as $D/H = \frac{1}{6}$ in this numerical experiment, and the diameter D of a cylinder is placed symmetrically between two walls a distance H apart. The inlet parabolic velocity profile is prescribed, while the radiation boundary conditions for the outflow velocity profile are employed. The computational grids generated by the isoparametric mapping are displayed on Fig. 5. Two distinct subdomains can be clearly visualized, one (coarse-grid subdomain) with the Cartesian grids and another

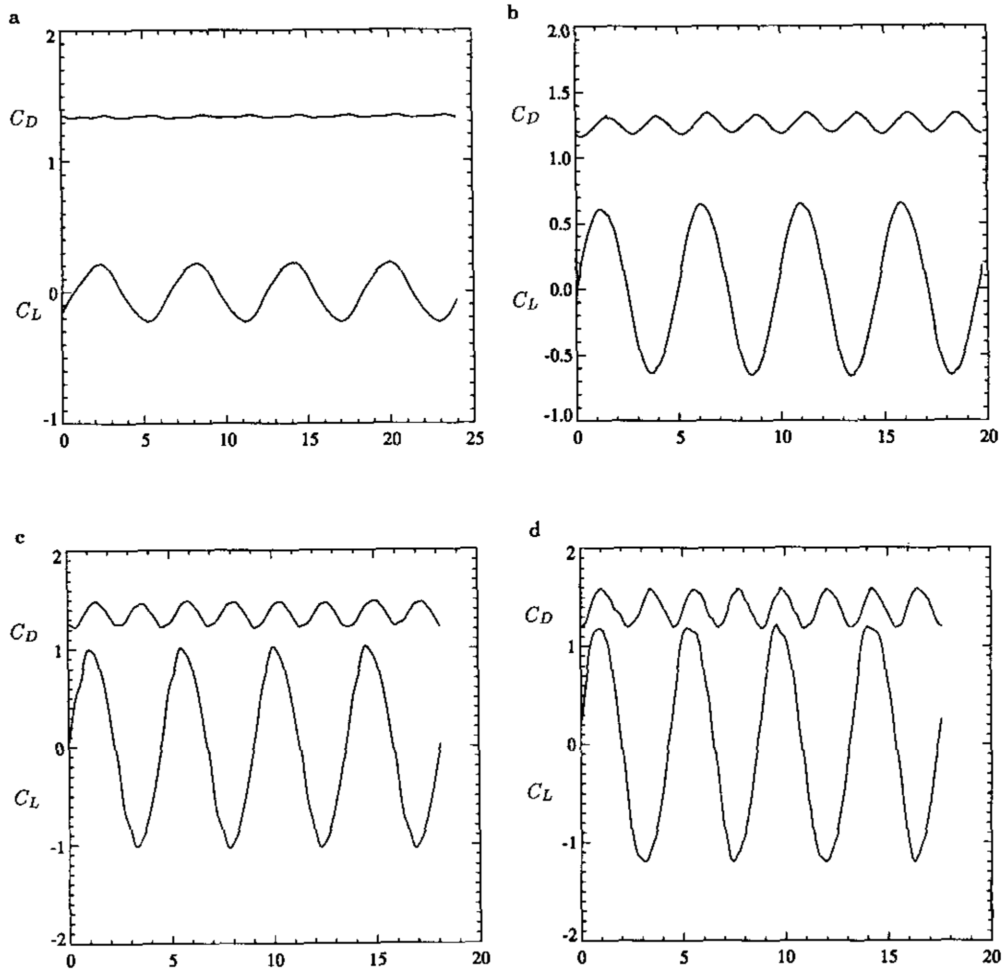

FIG. 6. Time history of drag C_D and lift C_L coefficients for flow over a cylinder at (a) Re = 100, (b) Re = 250, (c) Re = 500, and (d) Re = 1000.

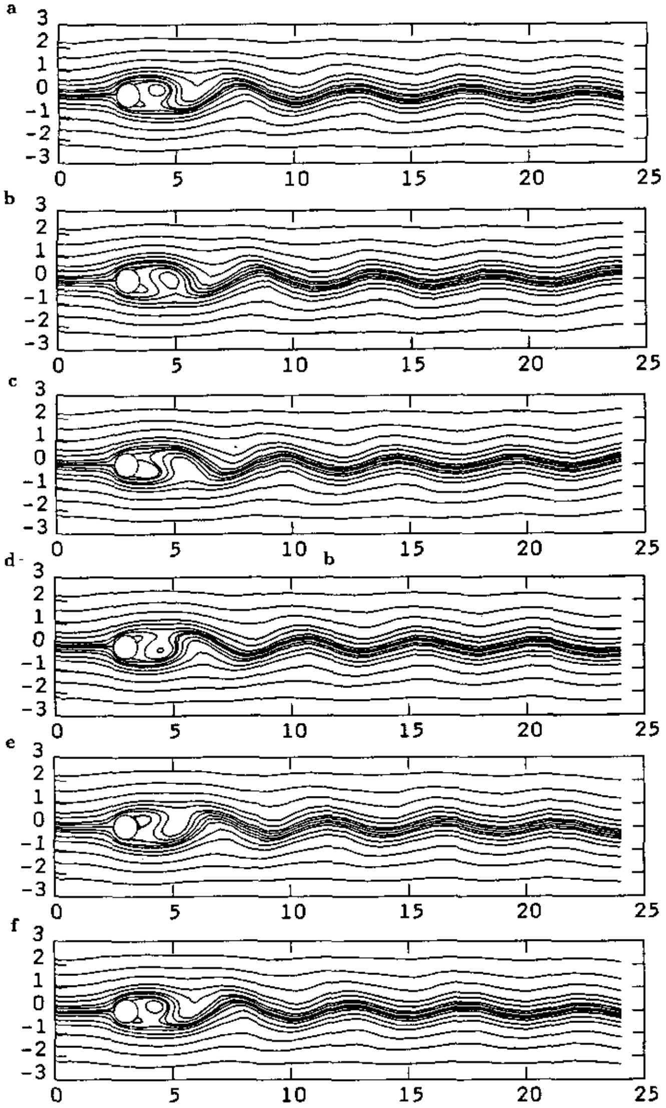

FIG. 7. Full-cycle time history of streamline plots for Re = 100 at time (a) $t = 0$, (b) $t = 0.2T$, (c) $t = 0.4T$, (d) $t = 0.6T$, (e) $t = 0.8T$, and (f) $t = T$.

(fine-grid subdomain) with the "O" grids. The overlapping area is not explicitly shown in the figure, but just imagine the extension of one more element from the Cartesian grids into the "O" grids.

As expected, when the Reynolds number exceeds the critical value any slight disturbance to the flow field will eventually trigger vortex shedding. The simulation starts right after a fully developed flow in a channel is suddenly blocked by a circular cylinder at time $t = 0$. A zero value of vertical velocity along the symmetry line behind the cylinder persists until the loss of symmetry occurs due to the onset of flow instability. All the input data and numerical results are listed in Table I.

The Strouhal number is defined as $S = fD/U_{\max}$, where f is the shedding frequency, D is the diameter of a cylinder, and U is the maximum inlet velocity. The four-cycle time history of the drag coefficient C_D and the lift coefficient C_L are plotted in Fig. 6, and it is no surprise that both increase with increasing Reynolds number. Figures 7–10 provide a full-cycle time history of streamline plots for the different Reynolds numbers, and incidently they all confirm the success of the radiation outflow conditions applied on the truncated domain. Since the intensity of the vertical velocity along the symmetry line behind the cylinder fast grows as the Reynolds number continuously increases, any inappropriate downstream flow conditions will definitely influence the upstream flow development. The Strouhal number tends to increase with increasing Reynolds number and is quite different from the case of a square cylinder investigated by [24], who found the Strouhal number continuously increased up to the maximum at the Reynolds number around

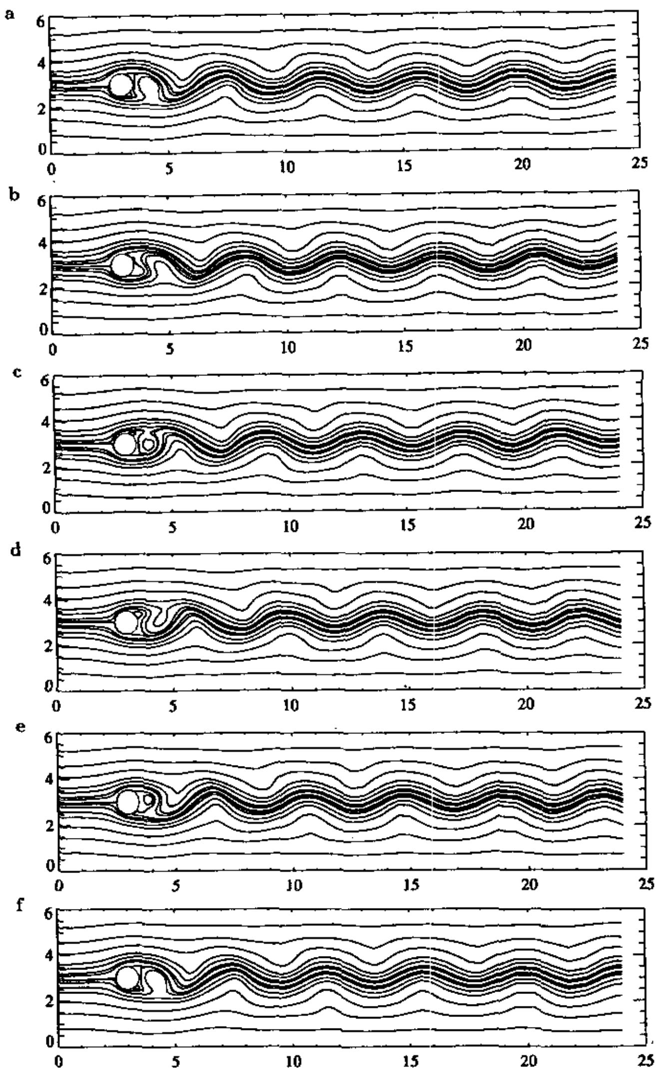

FIG. 8. Full-cycle time history of streamline plots for Re = 250 at time (a) $t = 0$, (b) $t = 0.2T$, (c) $t = 0.4T$, (d) $t = 0.6T$, (e) $t = 0.8T$, and (f) $t = T$.

250 and then slowly decreased to an asymptotic value with increasing Reynolds number. The difference may be attributed to the sharp edge of the square cylinder that causes an additional separation close to the walls at the high Reynolds number, besides the primary circulation in the wake and the secondary bubble on the surface of the cylinder.

The most striking phenomenon observed is that two secondary bubbles coexist on the surface of the cylinder when the Reynolds number becomes 500 (Fig. 9b) or higher (Figs. 10a, 10e, 10f). So far the author could not find any published literature to confirm this point, but he does believe that they indeed exist with accurate numerical schemes.

6.3. *Bifurcation Flow*

Bifurcation flows are encountered in many situations such as blood flow in branching arteries, jet injection into a ducted cross flow, and so on. The understanding of bifurcation flow seems extremely important, especially in the immediate vicinity of the junctions by the bifurcating process, where the steep change of wall shear stress or separation from the bifurcation point might occur. The two-dimensional flow model only provides a simple view of this problem, but many of the important effects cannot be adequately described by it; instead, the three-dimensional flow problem is best modelled towards the realistic approach to give insight into the understanding of bifurcation flow.

The streamfunction-vorticity formulation is widely used for the two-dimensional bifurcation flow simulation [27, 28], while for the three-dimensional flow the primitive variable formulation [29] would be the least complicated form and is superior to the velocity-vorticity formulation. For the two-dimensional

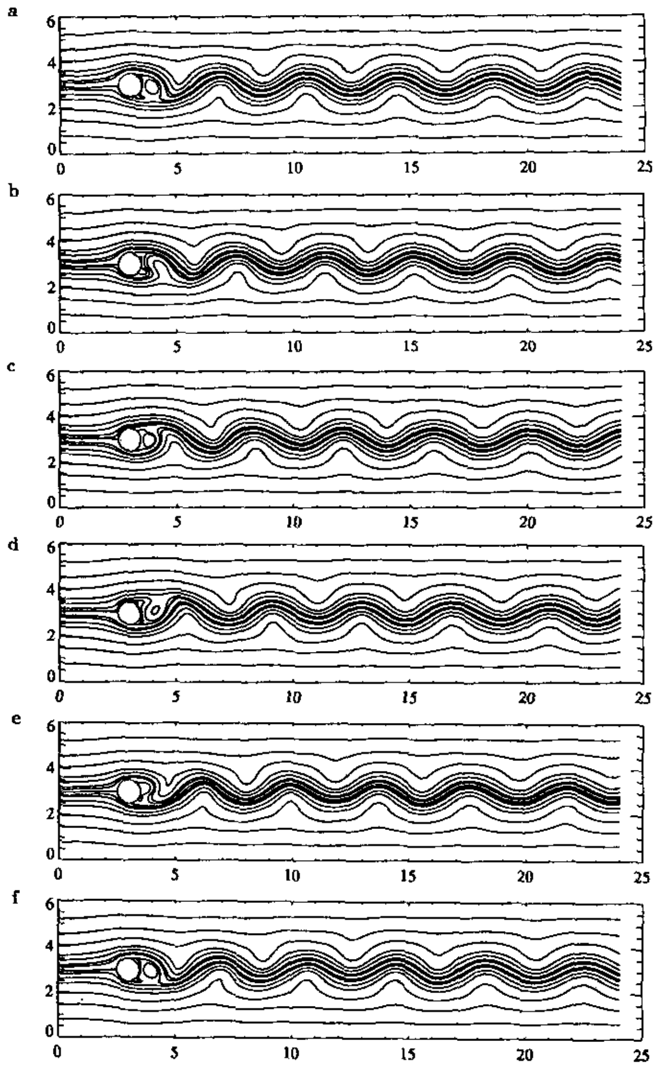

FIG. 9. Full-cycle time history of streamline plots for Re = 500 at time (a) $t = 0$, (b) $t = 0.2T$, (c) $t = 0.4T$, (d) $t = 0.6T$, (e) $t = 0.8T$, and (f) $t = T$.

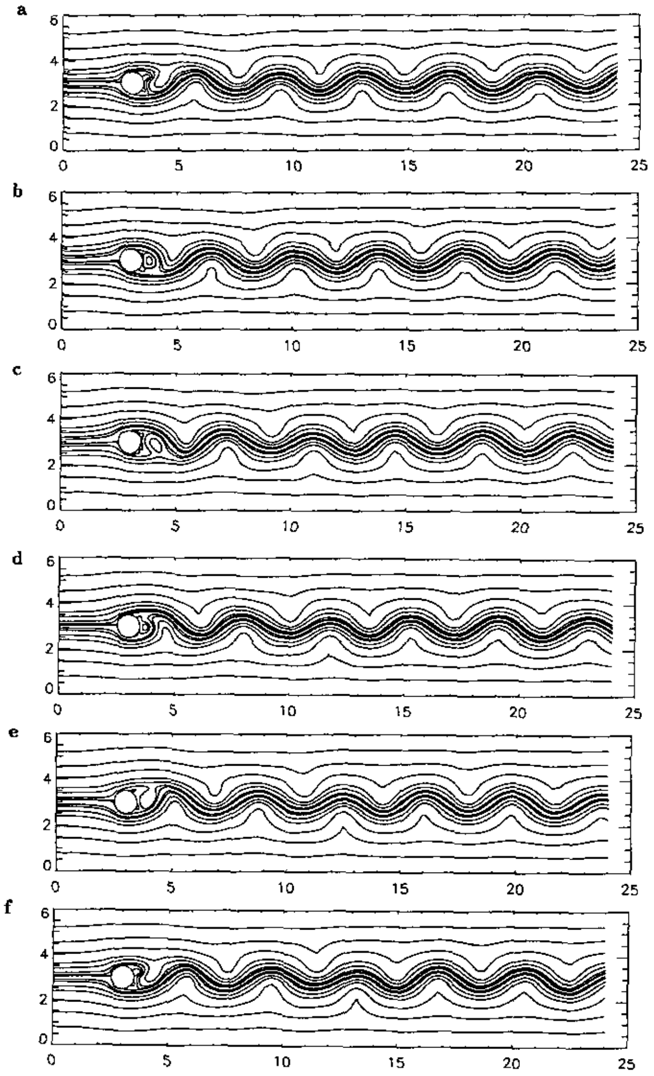

FIG. 10. Full-cycle time history of streamline plots for Re = 1000 at time (a) $t = 0$, (b) $t = 0.2T$, (c) $t = 0.4T$, (d) $t = 0.6T$, (e) $t = 0.8T$, and (f) $t = T$.

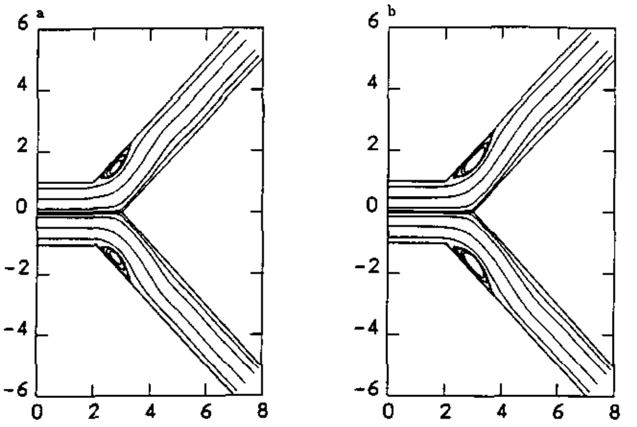

FIG. 11. Streamline pattern for the symmetry bifurcation flow at (a) Re = 500, and (b) Re = 1000.

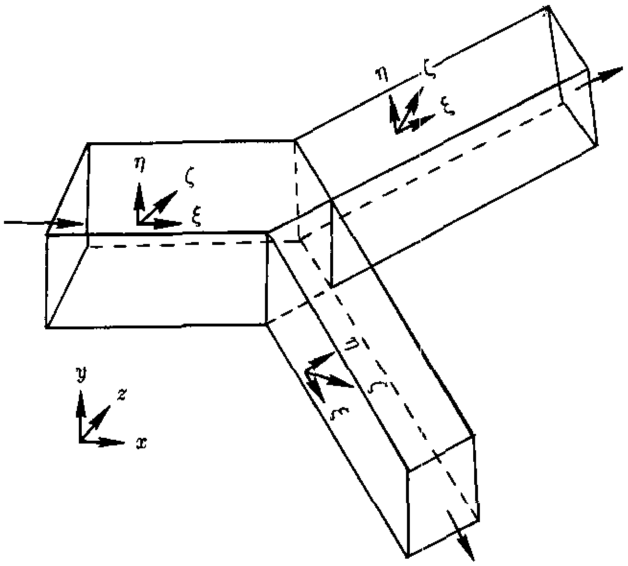

FIG. 12. The local block coordinates of three-dimensional bifurcation flow.

bifurcating case, the characteristic velocity and length defining the Reynolds number is based on the maximum velocity and half-width of the upstream channel; 10 * 4 elements (6 points per element) are allocated in the streamwise and transverse directions along each channel. Parabolic flow profiles are assumed at the inlet. For Re = 500 and 1000 with a branching angle of 90° (symmetry condition), the flow separates immediately from the upper or lower wall of each branch, and the reattachment length, due to the formation of the boundary layer along the upper or lower wall of the bifurcating branch, is also prolonged as the Reynolds number increases. The maximum streamfunction for Re = 500, 100 is $\psi_{\max} = 1.3484$ (Fig. 11a), $\psi_{\max} = 1.3581$ (Fig. 11b), respectively. As for the three-dimensional bifurcation flow, with a uniform depth of $\sqrt{2}$ in the spanwise direction, the local block coordinates are drawn in Fig. 12. In addition to having the same number of elements in the streamwise (ξ) and transverse (ζ) directions as in 2D, three elements were used in the spanwise direction (η). At Re = 500, three-dimensional streamwise (u) and transverse (w) velocity profiles on the spanwise central plane ($\eta = 0.707$) at the bifurcating position $\xi = 0$ and downstream 0.99 (Figs. 13 and 14) were compared with those computed by the two-dimensional flow model. As expected, due to the finite domain in the spanwise direction, the three-dimensional velocity profiles will deviate from the two-dimensional ones: an underestimate of streamwise velocity profiles, as well as a little overestimation of the transverse velocity profiles, by the two-dimensional flow model as seen in Figs. 13 and 14, respectively.

7. CONCLUSIONS

The solution of two- and three-dimensional flow over complex geometries has been solved by the pseudospectral element

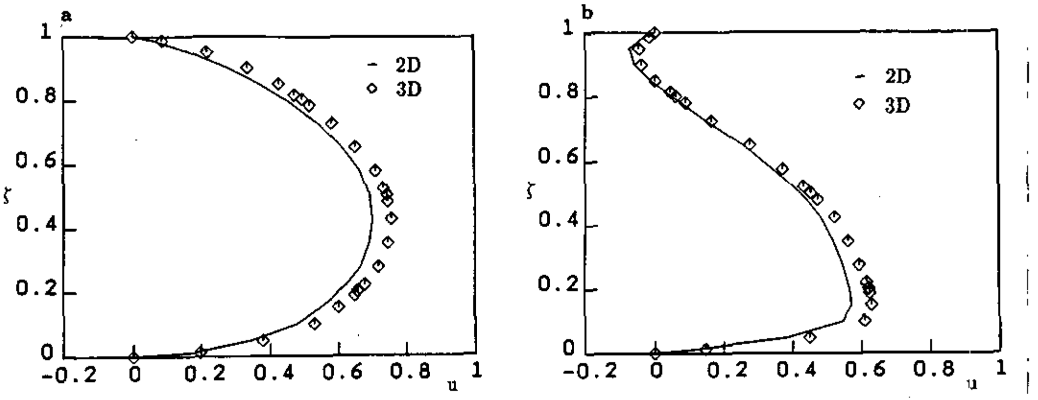

FIG. 13. Three-dimensional streamwise velocity profile for Re = 500 on spanwise central plane at the (a) $\xi = 0$ (bifurcating position), and (b) $\xi = 0.99$ (downstream).

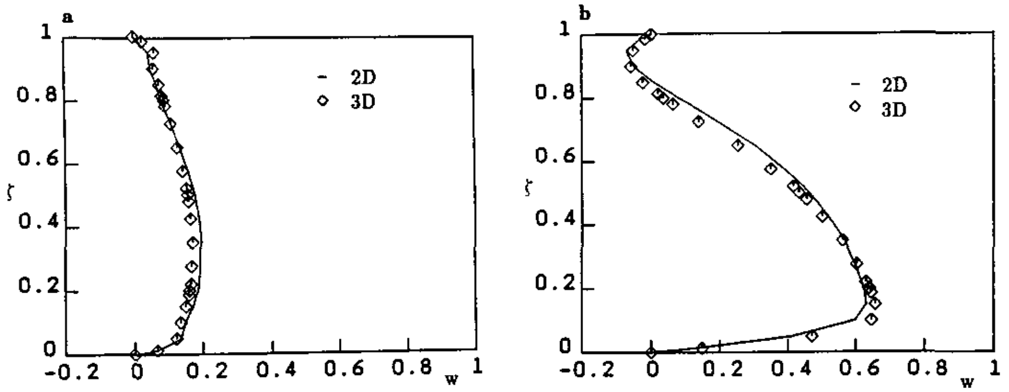

FIG. 14. Three-dimensional transverse velocity profile for Re = 500 on spanwise central plane at the (a) $\xi = 0$ (bifurcating position), and (b) $\xi = 0.99$ (downstream).

method via the multi-grid domain decomposition technique. The governing equations are the Navier–Stokes equations in a primitive variable form. The computational domain divides into a number of simple subdomains with the inter-overlapping zone, of which the overlapped grids may or may not coincide at the same places. During the data exchange among the subdomains, the multi-grid SAP technique is used to eliminate the high frequency error due to the data interpolation between the fine-grid subdomain and coarse-grid subdomain.

With an isoparametric mapping technique, the generation of computational grids for each subdomain could be easily achieved. The resulting pressure equation in each subdomain which contains a non-separable operator can be efficiently solved by the multi-level iterative preconditioned minimal residual method.

The modified radiation outflow boundary conditions have been found to be the best fit to meet the requirement of the global mass conservation constraint, as well as little influence upon the upstream flow development. Both the flow over a cylinder in a channel and the bifurcation flow clearly demonstrate the versatility of the proposed method.

ACKNOWLEDGMENT

This work was partially supported by the SPAWAR under the Contract N00039-91-C-0001.

REFERENCES

1. J. F. Thompson, Z. U. A. Warsi, and C. W. Mastin, *Numerical Grid Generation* (North-Holland, Amsterdam, 1985).
2. O. C. Zienkiewicz, *The Finite Element Method* (McGraw–Hill, London, 1971).
3. A. J. Chorin, *Math. Comput.* **22,** 745 (1968).
4. Y. S. Wong, T. A. Zang, and M. Y. Hussaini, *Comput. Fluids* **14,** 85 (1986).
5. D. Gottlieb and L. Lustman, *Comput. Fluids* **11,** 107 (1983).
6. J. A. Meijerink and H. A. van der Vorst, *J. Comput. Phys.* **44,** 134 (1981).
7. X. Xu, N. Qin, and B. E. Richards, *Int. J. Numer. Methods Fluids* **15,** 613 (1992).
8. H. C. Ku, R. S. Hirsh, and T. D. Taylor, *J. Comput. Phys.* **70,** 439 (1987).
9. H. C. Ku, R. S. Hirsh, T. D. Taylor, and A. P. Rosenberg, *J. Comput. Phys.* **83,** 260 (1989).
10. H. C. Ku, T. D. Taylor, and R. S. Hirsh, *Comput. Methods Appl. Mech. Eng.* **80,** 381 (1990).
11. H. C. Ku and B. Ramaswamy, in *Proceedings, Sixth Copper Mountain Conference on Multigrid Methods, Colorado, 1993.*
12. W. Hackbusch, *Multi-Grid Methods and Applications* (Springer-Verlag, Berlin, 1985).
13. A. Brandt, *Math. Comput.* **31,** 333 (1977).
14. W. J. Gordon, *Approximations with Special Emphasis on Spline Functions,* edited by I. J. Schoenberg (Academic Press, New York, 1969).
15. W. J. Gordon and C. A. Hall, *Numer. Math.* **21,** 109 (1973).
16. H. C. Ku, A. P. Rosenberg, and T. D. Taylor, in *Proceedings, Twelfth Conference on Numerical Methods in Fluid Dynamics,* edited by K. W. Morton (Oxford Univ. Press, Oxford, 1990).
17. M. O. Deville and E. H. Mund, *J. Comput. Phys.* **60,** 517 (1985).
18. P. Demaret and M. O. Deville, *J. Comput. Phys.* **83,** 463 (1989).
19. H. C. Elman and M. H. Schultz, *SIAM J. Numer. Anal.* **23,** 44 (1986).
20. M. C. Thompson and J. H. Ferziger, *J. Comput. Phys.* **82,** 94 (1989).
21. H. C. Ku, T. D. Taylor and R. S. Hirsh, *Comput. Methods Appl. Mech. Eng.* **75,** 141 (1989).
22. I. Orlanski, *J. Comput. Phys.* **21,** 251 (1976).
23. P. M. Gresho, *Annu. Rev. Fluid Mech.* **23,** 413 (1991).
24. D. W. Davis, E. F. Moore, and L. P. Purtell, *Phys. Fluids* **27,** 46 (1984).
25. G. S. West and C. J. Apelt, *J. Fluid. Mech.* **114,** 114 (1982).
26. A. Richter and E. Naudascher, *J. Fluid Mech.* **78,** 561 (1976).
27. J. S. Bramley and S. C. R. Dennis, *Comput. Fluids* **12,** 339 (1984).
28. V. O'Brien, L. W. Ehrlich, and M. H. Friedman, *J. Fluid Mech.* **75,** 315 (1976).
29. M. A. Leschziner and K. P. Dimitriadis, *Comput. Fluids* **17,** 371 (1989).