Journal of Fluids and Structures (1999) **13,** 191–207
Article No.: jfls.1998.0200, available online at http://www.idealibrary.com on IDEAL

SIMULATION OF FLUID–STRUCTURE INTERACTION USING PATCHED-OVERSET GRIDS

C. J. FREITAS AND S. R. RUNNELS

Computational Mechanics, Mechanical and Materials Engineering Division, Southwest Research Institute
P.O. Drawer 28510, San Antonio, Texas 78228-0510, U.S.A.

(Received 31 August 1997 and in revised form 8 October 1998)

The simulation of dynamic fluid–structure interaction has historically presented several challenges when computed on patched grid systems. Traditionally, the structures are static or constrained to predefined motions, as with a piston. However, overset grid methods, in the context of patched grid systems, provide a vehicle for simulating truly dynamic fluid–structure response and interaction for geometrically complex objects. Presented here is a new method for the simulation of dynamic fluid–structure interaction based on patched-overset grid systems, which allows for the coupled dynamics of the problem to define the movement of fluid and structure.

1. INTRODUCTION

THERE ARE MANY APPLICATIONS where it is desirable to simulate the dynamic interaction of solid, deformable objects with a flowing fluid. A current aerospace example is the simulation of store separation from the underside of an aircraft wing during flight (Lijewski & Suhs 1994; Meakin & Suhs 1996), where in practice, it is possible for fluid forces to make the store actually rise after being released, thus striking the wing. Important industrial applications include the simulation of valve dynamics, deformation and fracture of pipelines, and failure of components in power systems (i.e., failure of rotor blades in a turbine). In all of these applications, it is possible to model the solid as a rigid body and thus predict its motion (translation and rotation) using rigid-body Newtonian physics. The net force and moment on the object is computed during the simulation by integrating the stress field imposed by the surrounding fluid. As the object moves through the flow field, the fluid adjusts to the dynamic motion of the object and responds by imposing new stresses on the object. Using the fluid-induced stresses, the deformation as well as position of the solid may be modelled. This is the conceptual framework for the simulation of dynamic, fluid–structure interaction that is developed here.

Computational fluid dynamics (CFD) simulation of fluid–solid interaction is conceptually simplified through the use of overset grids, with the resulting algorithm referred to as the Chimera scheme. One approach to implementing overset grids is to construct a background grid resolving the entire simulation domain. Then, additional grids, each surrounding one of the moving objects, are placed over the background grid. No attempt is made to align the grid points of the background grid with those of the overset grids. Instead, the fluid flow equations are solved on the background grid and then separately on the overset grids in a hierarchical technique that is based on the structure of the grid system and the relationship between grids. Through interpolation, the overset grids use the current flow

0889–9746/99/020191 + 17 $30.00

solution on the background grid as intermediate boundary conditions in an iterative solution. Conversely, the background grid disregards those regions that are lying under an overset grid, instead adopting for the solution the interpolated up-dated values from the overset grid. This iterative cycle of equation solution on a grid and intergrid data exchange is continued until a converged solution is achieved (based on a minimum global residual), and in this way, the solutions on the overset and background grids are fully coupled.

An advantage of this method is that the same flow solver is used for each grid block in the hierarchy of grids, implying that the overset grids may have the same topological structure as does the background grid. For example, the solver may accommodate advanced grid structures on the background grid, such as a nonaligned patched grid system. That same solver will then be applied to the overset grids as well, meaning that the overset grids may also make use of the advanced grid topology (i.e., that an overset grid may consist of a series of patched grids). By applying the same solver to all grid systems, the implementation of the Chimera scheme requires only the addition of data structures and interpolation algorithms to an existing multiblock flow solver.

Steger, Dougherty & Benek (1983) first implemented the Chimera variant of the overset grid technique. Since that time, the technique has been implemented by several other investigators in a broad range of applications. The Chimera scheme is not limited to solid–fluid interaction problems. It has been used to simulate flow around complex stationary objects (Shahcheraghi & Dwyer 1994; Venkata *et al.* 1992; Tu & Fuchs 1992, 1995), as a method to accomplish local adaptive refinement (Moore & Flaherty 1992; Nacul *et al.* 1990; Kao *et al.* 1994), and to track free surfaces (Petersson 1993).

There are a few minor difficulties with the Chimera scheme, all of which were correctly identified by Steger, Dougherty & Benek in their 1983 article. The most obvious difficulty is the development of fast, yet accurate, interpolation schemes that can determine the boundary conditions for the overset grids in a way that minimizes the computational penalty of the method. Several approaches have been explored to address this aspect of the process (Chesshire & Naik 1994; Lijewski & Suhs 1994; Meakin & Suhs 1996). Mass conservation, however, remains the principal difficulty with this method. At first glance, the overset technique seems quite straightforward a procedure. For each overset grid, the solution from the background grid is interpolated and applied as boundary conditions. However, deeper consideration reveals that, because the boundary values are interpolated from the background grid, the corresponding sum of mass flux for the overset grid may not be zero. In fact, it probably will not be zero at most steps within the iterative solution. Hence, traditional interpolation methods introduce a discontinuous mass flux from the background grid to the overset grid. Obviously, as with any discretization scheme, the order of accuracy of the interpolation method will enhance or reduce the fidelity of the conservation of any interpolated variable. Further, this problem is mitigated somewhat through the use of iteration which drives the elemental solution and thus the global solution to convergence and mass conservation. As an alternative to interpolating primitive variables only, mass flux can be interpolated as well. However, for the same reasons cited above, the sum of the interpolated mass flux may not be zero, but the interpolated values may be more easily adjusted so that they do sum to zero. A method for correcting the flux to maintain mass conservation was, again, identified by Steger *et al.* (1983), and has been implemented by other investigators. The bottom line is that many users of the Chimera scheme do nothing to correct for mass conservation, yet report that no appreciable error results.

Several features make the method presented here different from previous work. Primary among these is that the fluid–structure interaction is fully coupled in which the fluid variables (incompressible fluid flow) and structure dynamics are time evolved simultaneously. The structure has no defined motion, so it is free to move as dictated by the system

dynamics. This work presents a method for simulation of a fully coupled and dynamically interacting, incompressible flow field and solid body, using a patched-overset grid technique.

2. THEORETICAL BACKGROUND

2.1. Discretization Method

To provide a context for the overset method, a development of the discrete scheme for a rectangular grid system is given below. Nonrectangular domains are accommodated through a mathematical transformation combined with geometric decomposition, the details of which are omitted for clarity, but may be found in Freitas (1991, 1995).

Consider, then, the x-momentum equation for an incompressible flow, integrated over a rectangular control volume (see Figure 1), which corresponds to a computational cell in a grid aligned with x–y coordinate system, and using a colocated variable arrangement:

$$\left[\rho u - (\rho u)^n \frac{\Delta y \Delta x}{\Delta t}\right] + \left[\left(\rho uu - \mu \frac{\partial u}{\partial x}\right)\Delta y\right]_e - \left[\left(\rho uu - \mu \frac{\partial u}{\partial x}\right)\Delta y\right]_w + \left[\left(\rho vu - \mu \frac{\partial u}{\partial x}\right)\Delta x\right]_n - \left[\left(\rho vu - \mu \frac{\partial u}{\partial y}\right)\Delta x\right]_s = \left[\left(\mu \frac{\partial u}{\partial x}\right)\Delta y\right]_e - \left[\left(\mu \frac{\partial u}{\partial x}\right)\Delta y\right]_w + \left[\mu \frac{\partial v}{\partial x}\Delta x\right]_n - \left[\left(\mu \frac{\partial v}{\partial x}\right)\Delta x\right]_s + (p_w - p_e)\Delta y, \quad (1)$$

where the superscript n refers to the current time step, and no superscript refers to the new time step $(n + 1)$. The, u^n is the x-velocity at time step n, and u is the velocity at time step $n + 1$. An integrated form of the continuity equation, multiplied by u^{n+1} and added to the above equation provides a method for replacing the density (ρ) term with ρ^n. Making that

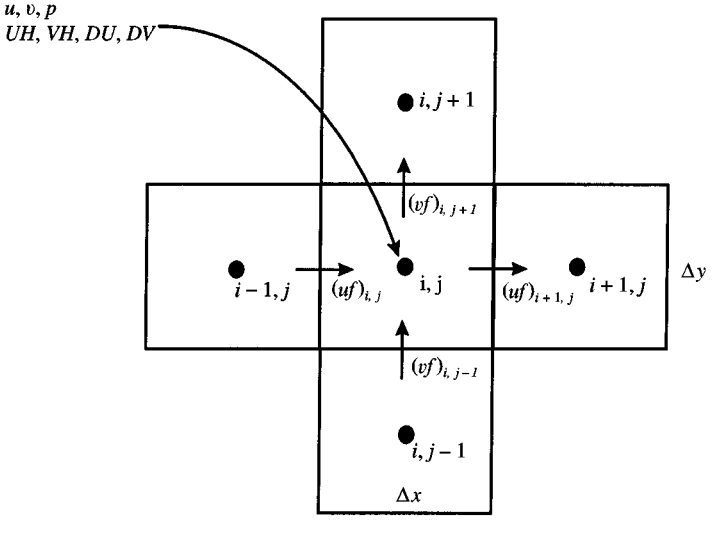

Figure 1. Discretization cell (control volume).

substitution produces the following equation:

$$\rho^n(u-u^n)\frac{\Delta x\Delta y}{\Delta t} - u[(\rho u)_e\Delta y - (\rho u)_w\Delta y + (\rho v)_n\Delta x - (pv)_s\Delta x]$$
$$+\left[\left(\rho uu-\mu\frac{\partial u}{\partial x}\right)\Delta y\right]_e - \left[\left(puu-\mu\frac{\partial u}{\partial x}\right)\Delta y\right]_w$$
$$+\left[\left(\rho vu-\mu\frac{\partial u}{\partial y}\right)\Delta x\right]_n - \left[\left(\rho vu-\mu\frac{\partial u}{\partial y}\right)\Delta x\right]_s$$
$$=\left[\left(\mu\frac{\partial u}{\partial x}\right)\Delta y\right]_e - \left[\left(\mu\frac{\partial u}{\partial x}\right)\Delta y\right]_w + \left[\left(\mu\frac{\partial v}{\partial x}\right)\Delta x\right]_n - \left[\left(\mu\frac{\partial v}{\partial x}\right)\Delta x\right]_s + (p_w-p_e)\Delta y. \quad (2)$$

Approximations to the derivatives in the above equation are used to create a discretized equation. The centroid of the current cell is identified as a "nodal point", and the solution values (u, v and p) are located at that nodal point in a "colocated" variable arrangement. The following approximated derivatives are introduced:

$$\rho^n(u-u^n)\frac{\Delta x\Delta y}{\Delta t} \to \rho^n_{i,j}(u_{i,j}-u^n_{i,j})\frac{\Delta x\Delta y}{\Delta t},$$

$$U[(\rho u)_e\Delta y - (\rho u)_w\Delta y + (\rho v)_n\Delta x - (\rho v)_s\Delta x] \to \rho u_{i,j}[\Delta y(uf)_{i,j} - \Delta y(uf)_{i+1,j} + \Delta x(vf)_{i,j+1} - \Delta x(vf)_{i,j}],$$

$$\left[\left(\rho uu-\mu\frac{\partial u}{\partial x}\right)\Delta y\right]_e \to \rho\Delta y(uf)_{i,j}\,u_e - \frac{\mu\Delta y}{\Delta x}(u_{i+1,j}-u_{i,j}),$$

$$\left[\left(\rho uu-\mu\frac{\partial u}{\partial x}\right)\Delta y\right]_w \to \rho\Delta y(uf)_{i+1,j}\,u_w - \frac{\mu\Delta y}{\Delta x}(u_{i,j}-u_{i-1,j}),$$

$$\left[\left(\rho vu-\mu\frac{\partial u}{\partial y}\right)\Delta x\right]_n \to \rho\Delta x(vf)_{i,j+1}\,u_n - \frac{\mu\Delta x}{\Delta y}(u_{i,j+1}-u_{i,j}),$$

$$\left[\left(\rho vu-\mu\frac{\partial u}{\partial y}\right)\Delta x\right]_s \to \rho\Delta x(vf)_{i,j}\,u_s - \frac{\mu\Delta x}{\Delta y}(u_{i,j}-u_{i,j-1}), \quad (3)$$

$$\left[\left(\mu\frac{\partial u}{\partial x}\right)\Delta y\right]_e \to \frac{\mu\Delta y}{\Delta x}(u_{i+1,j}-u_{i,j}),$$

$$\left[\left(\mu\frac{\partial u}{\partial x}\right)\Delta y\right]_w \to \frac{\mu\Delta y}{\Delta x}(u_{i,j}-u_{i-1,j}),$$

$$\left[\left(\mu\frac{\partial v}{\partial x}\right)\Delta x\right]_n \to (\mu\Delta x)_n\frac{1}{2}\left[\frac{v_{i+1,j+1}-v_{i-1,j+1}}{2\Delta x}+\frac{v_{i+1,j}-v_{i-1,j}}{2\Delta x}\right],$$

$$\left[\left(\mu\frac{\partial v}{\partial x}\right)\Delta x\right]_s \to (\mu\Delta x)_s\frac{1}{2}\left[\frac{v_{i+1,j}-v_{i-1,j}}{2\Delta x}+\frac{v_{i+1,j-1}-v_{i-1,j-1}}{2\Delta x}\right], \quad (4)$$

A variant of Leonard's (1979) QUICK scheme (Freitas 1986) is used to discretize the convective velocity terms. For uniform grid spacing, the "east" x-velocity, u_e, representing

the convective velocity located on the left face of the nodal control volume is evaluated as

$$u_e = \begin{cases} 1/8(6u_{i,j} - u_{i+1,j} - u_{i-1,j} + 1/2u_{i+,1\,j}) & \text{CE} > 0, \\ 1*8(3u_{i,j} + 6u_{i+1,j}) - 1/8u_{i+2,j} & \text{CE} < 0, \end{cases} \tag{5}$$

where CE represents the magnitude and direction of the facial flux on the eastern face of the control volume and is equal to $\rho_e \Delta y_e (uf)_{i+1,j}$. Note that *uf* and *vf* represents the facial velocity normal to a control volume face (refer to Figure 1). An interpolation equation similar to equation (5) is used for each of the facial velocities on the other faces of the control volume. This QUICK scheme is third-order accurate for a uniform grid and represents the most accurate polynomial interpolation scheme possible based on three nodal values. Algebraic substitution of the discrete derivatives of equations (3) and (4) and the QUICK formulated convective velocities, followed by some manipulation, produces a system of algebraic equations of the general form

$$\begin{aligned} (AP)_{i,j}u_{i,j} = (AE)_{i,j}u_{i+1,j} + (AW)_{i,j}u_{i-1,j} + (AN)_{i,j}u_{i,j+1} \\ + (AS)_{i,j}u_{i,j-1} + (SU)_{i,j} + \Delta y(p_w - p_e), \end{aligned} \tag{6}$$

where the coefficients *AP*, *AE*, *AW*, *AN*, and *AS* are functions of the grid spacing and facial fluxes, and *SU* is a function of the grid spacing and the surrounding cell-centered velocities. Employing summation shorthand for these coefficients and introducing an intermediate, temporary velocity variable *UH*, the *x*-momentum equation is written in shorthand as follows:

$$(UH)_{i,j} \equiv \frac{\sum A_{nb}u_{nb} + (SU)_{i,j}}{(AP)_{i,j}}, \qquad (DU)_{i,j} \equiv \frac{\Delta y}{(AP)_{i,j}}, \tag{7}$$

so that

$$u_{i,j} = (UH)_{i,j} + (DU)_{i,j}(p_w - p_e). \tag{8}$$

The above equation is a nonlinear equation for $u_{i,j}$. The term $(UH)_{i,j}$ depends upon cell-centered variables such as $u_{i,j}$, facial velocities, and indirectly on pressure. A similar equation for $v_{i,j}$ may also be defined in analogy to that for $u_{i,j}$.

It is well known that the cell-centered, collocated variable arrangement results in a decoupling of the velocity and pressure fields when solving the incompressible flow equations. This decoupling effect has been overcome by two approaches. One approach uses a staggered grid arrangement of variables, and the other uses a facial flux equation. The facial flux equation essentially provides a staggered arrangement of velocity and pressure. Rhie & Chow (1982) first developed the concept of collocated variable arrangement with these special velocity–pressure relations. Here, the set of cell-facial velocities is defined as follows:

$$\begin{aligned} (uf)_{i,j} &= 1/2[(UH)_{i,j} + (UH)_{i-1,j}] + 1/2[(DU)_{i,j} + (DU)_{i-1,j}](p_{i-1,j} - p_{i,j}), \\ (vf)_{i,j} &= 1/2[(VH)_{i,j} + (VH)_{i,j-1}] + 1/2[(DV)_{i,j} + (DV)_{i,j-1}](p_{i,j-1} - p_{i,j}). \end{aligned} \tag{9}$$

Note that the facial velocities are not merely the average of the cell-centered velocities but instead are computed from the averaged discrete operators $(UH)_{ij}$, $(VH)_{ij}$ and the pressures bounding the control volume face (see Figure 1). If the facial velocities are simply computed by averages of u_{ij} and $u_{i-1,j}$, then velocity–pressure decoupling would result due to the velocity at node *ij* being coupled to pressures at nodes $i-1$, j and $i+1$, j without association to the pressure at node *i*, *j*.

Pressure is computed using the integrated continuity equation

$$(uf)_{i,j}\Delta y - (uf)_{i+1,j}\Delta y + (vf)_{i,j+1}\Delta x - (vf)_{i,j}\Delta x = 0, \tag{10}$$

whereby substituting the equations for $(uf)_{i,j}$ *and* $(vf)_{ij}$ into the continuity equation results in a nonlinear equation for each pressure node. Equation (10) then takes the form

$$\begin{aligned}
&(\rho\Delta y_e)\left[(1-\alpha)(uf)_{i,j}^{I-1} + \frac{\alpha}{2}[(UH)_{i,j}^{I} + (UH)_{i-1,j}^{I} + [(DU)_{i,j}^{I} + (DU)_{i-1,j}^{I}]](p_{i,j}^{I} - p_{i-1,j}^{I})]\right] \\
&+ (\rho\Delta y_e)\left[(1-\alpha)(uf)_{i+1,j}^{I-1} + \frac{\alpha}{2}[(UH)_{i,j}^{I} + (UH)_{i+1,j}^{I} + [(DU)_{i,j}^{I} + (DU)_{i+1,j}^{I}](p_{i+1,j}^{I} - p_{i,j}^{I})]\right] \\
&+ (\rho\Delta x_e)\left[(1-\alpha)(vf)_{i,j}^{I-1} + \frac{\alpha}{2}[(VH)_{i,j}^{I} + (VH)_{i,j-1}^{I} + [(DV)_{i,j}^{I} + (DV)_{i,j-1}^{I}](p_{i,j}^{I} - p_{i,j-1}^{I})]\right] \\
&+ (\rho\Delta x_e)\left[(1-\alpha)(vf)_{i,j+1}^{I-1} + \frac{\alpha}{2}[(VH)_{i,j}^{I} + (VH)_{i,j+1}^{I} + [(DV)_{i,j}^{I} + (DV)_{i,j+1}^{I}](p_{i,j+1}^{I} - p_{i,j}^{I})]\right] = 0,
\end{aligned} \tag{11}$$

where under-relaxation has been employed in defining the (uf) *and* (vf) parameters, and the superscript I refers to the iterate level. This equation may also be reduced to a more compact form,

$$(AP)_{i,j}p_{i,j} = \Sigma A_{nb}p_{nb} + (SU)_{i,j}^{p}. \tag{12}$$

The solution procedure developed here is a modification of that due to Maliska & Raithby (1983). Their formulation was based on a staggered variable arrangement, where here it is built upon a collocated variable arrangement with all the additional subtleties associated with that. The defining aspect of the procedure is that iterative sweeps in the momentum equations contribute a small portion of the overall convergence of the flow field. The critical element is finding the correct solution for the pressure field. Thus, in the procedure, a single momentum sweep is used in each iteration, with the pressure field obtained by solving the exact pressure equation and using the latest velocity values. A single iterative sweep for a grid block then consists of the following steps:

(i) based on the current velocities, compute the temporary velocity variables (UH and VH) and pressure coefficients DU and DV;
(ii) solve the exact pressure equation;
(iii) update the facial velocities (uf) *and* (vf), and nodal velocities u and v;
(iv) check the overall convergence based on residual and point-wise iterative change; if the convergence criteria are not satisfied, return to Step i, otherwise increment time and solution variables and return to Step i.

This procedure has some unique features. In particular, the pressure equation is exact, no iteration or relaxation is performed on the nodal velocities, and the pressure equation may be solved by any type of efficient linear solver.

2.2. Patched Grid Implementation

The above development is for a single rectangular grid. To accommodate more complex domains, several of these rectangular grids may be placed adjacent to each other in a non-overlapping contiguous manner, to fill the domain. The collection of rectangular grids are topologically "patched" together and have only a common interface to each other. In the two-dimensional discussion pursued here, a third subscript on each solution variable

is used to indicate its grid block number. For example, $u_{3,2,9}$ is the cell located on column 3, row 2 of grid block 9.

The patched grid blocks are mathematically linked together through the fact that each grid block cell requires certain values from its neighboring cells. For cells that are on the boundary of a grid block, one of its neighbours will lie in the adjacent grid block. For example, if grid block 1 has 25 cells (5×5) and is to the left of grid block 2 which is also a 5×5 grid block, then the neighbour to the left of cell $(i, j, k) = (1, 2, 2)$ is cell $(5, 2, 1)$. Referencing adjacent grid blocks is accomplished through a four-member array, created for each grid block, that stores the neighbouring grid block numbers. This discussion applies to aligned grid blocks when there is a one-to-one correspondence across the patched grid interface. For the case of non-aligned interfaces, an integral relationship is used to determine the magnitude of the primitive variable to be exchanged between grid blocks at this interface. In this study, all patched grid interfaces are aligned. Only in the overset grids is interpolation used.

2.3. Patched-Overset Grid Implementation

The use of overset grids is a natural extension of the patched grid algorithm described in Section 2.2. Like the background grid, the overset grid may be composed of several grid blocks. Each of these grid blocks is treated exactly the same as those grid blocks of the background grid. On their boundaries, they communicate with each other, just as the background grid blocks do. Furthermore, they are included in the iterative solver loop, just as the background grid blocks are. The only difference for the overset grids is that along their outer boundaries they must interpolate values from the background grid for use as updated boundary conditions. Figure 2 illustrates the bilinear interpolation scheme that is used to obtain boundary values for the overset grid from the background grid. The search engine for determining which background cells to use for interpolation, along with the

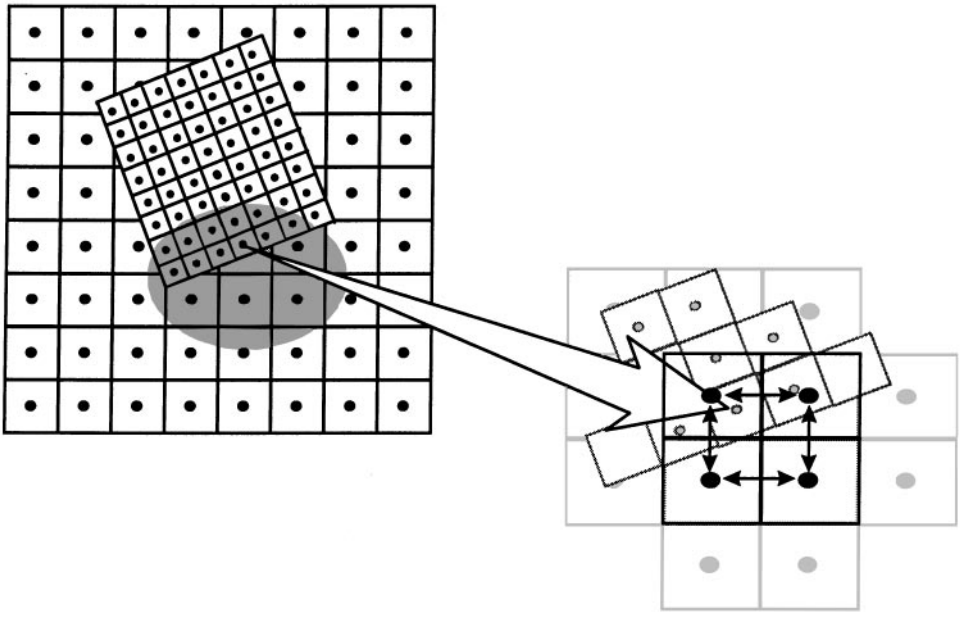

Figure 2. Illustration of bilinear interpolation used for overset grid boundaries.

bilinear interpolation code, augments the existing multiblock code used for inter-grid-block communication.

The augmentation to the multiblock code for the overset grids is accompanied by an analogous modification for the background grid. Solution at those cells lying under the overset grid is not necessary. Instead, the up-dated values for these cells are interpolated directly from the overset grid. The discrete pressure equation for these cells is reduced to a simple identity, setting the new pressure iterate equal to the interpolated value. The linear system for pressure on each background grid is solved in the usual way, with the coefficient matrix appropriately modified.

It is possible to enclose a solid object with the overset grid blocks so as to introduce the object into the flow field without having to change the background grid system. Furthermore, no algorithm modification is required to do so. The solid object must, however, be completely surrounded by overset grid blocks. Along grid block interfaces that coincide with solid-object boundaries, typical no-slip boundary conditions are applied. In this way, both stationary and moving objects can be included in the problem, with fixed objects resolved by the background grid, for example, and moving objects resolved by systems of overset grids.

As with any other overset grid scheme, the straightforward implementation described above suffers only from potential problems with mass conservation. Although the background grid is mass conserving (due to its usually well-defined boundary conditions), the order of interpolation used to extract boundary conditions for the overset grid will result in a non-zero line mass source term. As previous investigators have experienced, this mass conservation problem is neither prohibitive nor is it difficult to circumvent. One approach is to modify the interpolated values used by the overset grid so that, when integrated around the overset grid boundary, the mass flux sums to zero. A difficulty of this approach though, is to know how and where to scale the interpolated values so that a realistic mass balance is achieved. In the method presented here, we rely on an iterative solution technique to smooth the mass flux between grid blocks. Mass conservation is achieved by ensuring that the overall iterative residual, in all variables, is minimized to a small value (where a small value is dictated by the problem at hand).

2.4. Solid–Fluid Interaction

Dynamic fluid–solid interactions are simulated here using the overset grid method in conjunction with additional routines that determine the centre of mass and moment of inertia of the enclosed object. A dynamics routine is also required to compute the current fluid-imposed stresses, integrating them to determine the resultant force and moment on the object and then moving the object and associated overset grid according to the rigid body dynamics equations.

By establishing a coordinate system that is aligned with the boundary of the solid object and the adjoining grid, the stresses on the object are more easily determined. Expressed in tensor notation, the stress at a point in an incompressible Newtonian fluid is

$$\sigma_{ij} = -p\delta_{ij} + \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right), \tag{13}$$

where x_i is the coordinate in the normal and tangential directions. Therefore, the stress at a point on a solid boundary is

$$S_j = n_i\sigma_{ij}, \tag{14}$$

where n_i is the normal to the solid boundary. The stress tensor and the normal, and hence S_j,

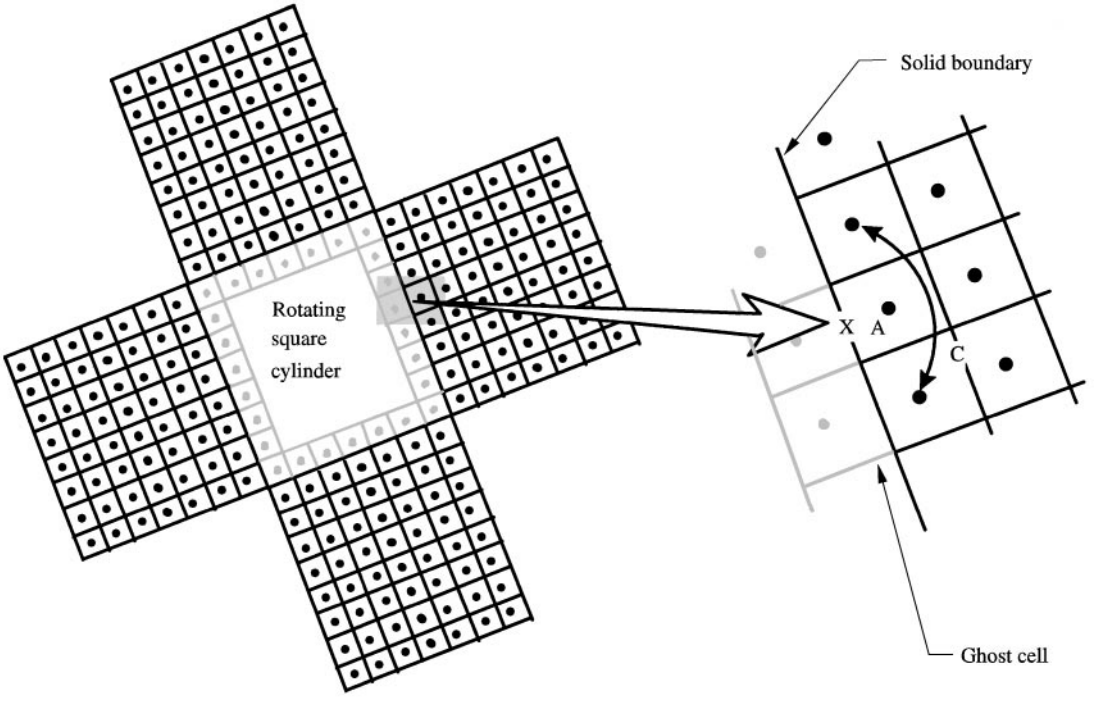

Figure 3. Illustration of process for evaluation of stresses on moving solid object.

is taken to be constant across the face of each computational cell. To evaluate the resultant force on a cell face, the stress is multiplied by the cell or control volume face area so that

$$F_j = An_i\sigma_{ij}, \tag{15}$$

where A is the area of the cell face under consideration and the value of F_j is still in the coordinate system aligned with the solid. To obtain the true force in the global system, it must be transformed according to the rotation of the aligned coordinate system relative to the global system.

Figure 3 illustrates the evaluation of stresses for a cell face on a boundary of the moving object. In the figure, the net forces in the x and y directions are evaluated for the cell face marked with an X. Referring to the equation for S_j, pressure and derivatives of velocity in the normal and tangential directions are required. The value for pressure is determined simply by using the pressure value at the node marked by A. Velocity derivatives in the direction normal to the solid are evaluated using the difference between the A-cell value divided by their distance to the wall. Furthermore, the tangential derivatives are evaluated using the difference of the nodal values marked with arrow C.

3. DEMONSTRATION

Two examples are presented, which both validate the implementation of the overset grid method and demonstrate the necessity of using a dynamic fluid–solid interaction algorithm to simulate the magnitude of the dynamic effect.

3.1. Example 1: Flow Around a Fixed, Square Cross-Section Cylinder

The problem selected to demonstrate these methods is that of flow around a cylinder of square cross-section. This flow was experimentally studied by Durao, Heitor & Pereira (1986, 1988). The experimental configuration was a horizontal flow in a 120 mm × 156 mm rectangular water tunnel with a 20 mm × 20 mm obstacle set across the narrow dimension of the tunnel. The duct extended 1·56 m upstream of the obstacle and 0·44 m downstream of the obstacle. The square cross-section cylinder was centred between the top and bottom walls of the duct. The free-stream reference velocity was 0·68 m/s, and the Reynolds number was 14 000, based on the dimension of the bluff body, and water as the working fluid. The resulting flow is two dimensional and symmetric. The obstacle imposes a 13% blockage to the flow, which is significant and requires that the duct walls be resolved in a numerical simulation. The flow includes regions of recirculating, accelerating, and near uniform velocity, with an unsteady wake region. This flow is turbulent; however, in this paper we focus on the solution method and the mean flow field, and use moderate resolution grids. In a subsequent paper, the details of the turbulence flow structure is studied, computed using Large Eddy Simulation methods and finer grid resolutions.

In this simulation, the base grid domain extends from 0·1 m upstream of the obstacle to 0·4 m downstream of the obstacle, and the width of the domain is 0·156 m. The base grid resolution is 75 horizontal nodes by 30 vertical nodes. In the region of the obstacle, eight overset grid blocks, each with a resolution of 20 × 20 are used to resolve the flow in the vicinity of the obstacle (refer to Figure 5). In this unsteady simulation, the time-step size is maintained to insure a maximum Courant number, anywhere in the computational domain, to be less than 1·0. Figure 4 displays the velocity vectors in both the base grid and the

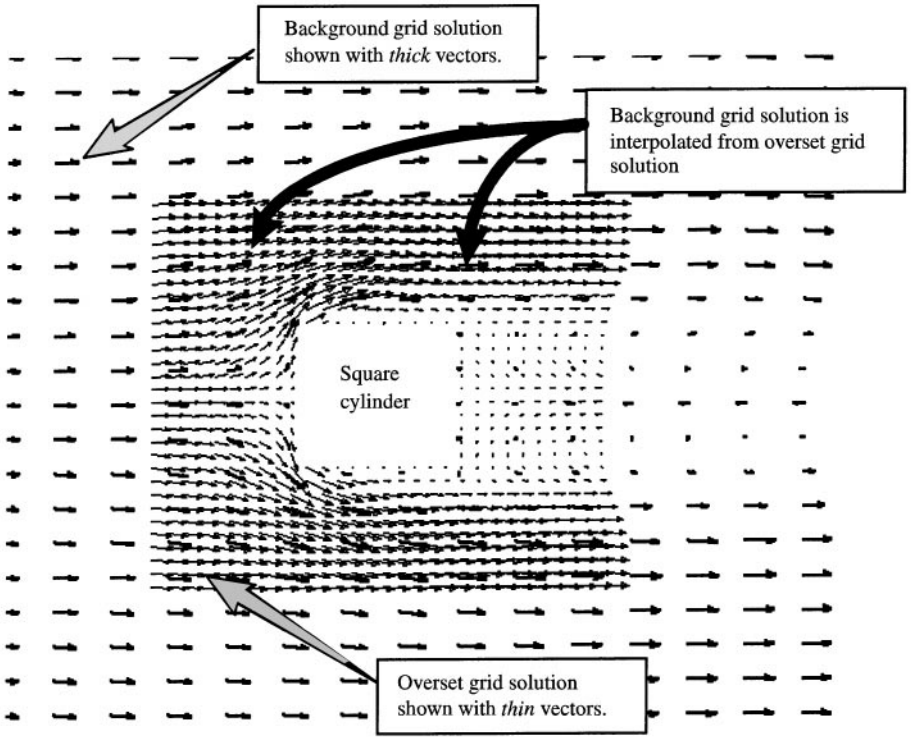

Figure 4. Flow vectors resulting from overset grid computation.

overset grids in the region of the obstacle. Clearly shown is the smoothness of the mapping between the base and overset grids. This smooth variation of the solution across grids is further demonstrated in Figure 5, where pressure contours are displayed. The solution of the pressure field is the most sensitive parameter to inaccurate or nonsmooth interpolation across grid interfaces. A nonsmooth method would exhibit itself by discontinuous contours across the block interfaces, or spatially oscillating contours as they approach a block interface. Clearly, this is not seen in Figure 6. Finally, Figures 11–14 display velocity vectors at different points in time during this unsteady flow simulation. These images display the evolution of the periodic flow in the wake region of the obstacle and, as discussed below,

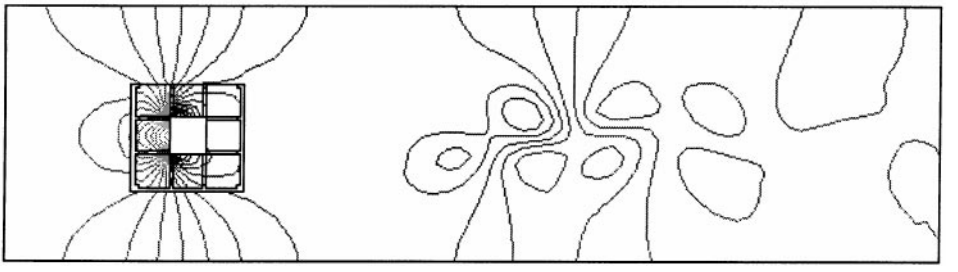

Figure 5. Pressure contours illustrating the continuity of the overset solution. The complete flow domain is displayed.

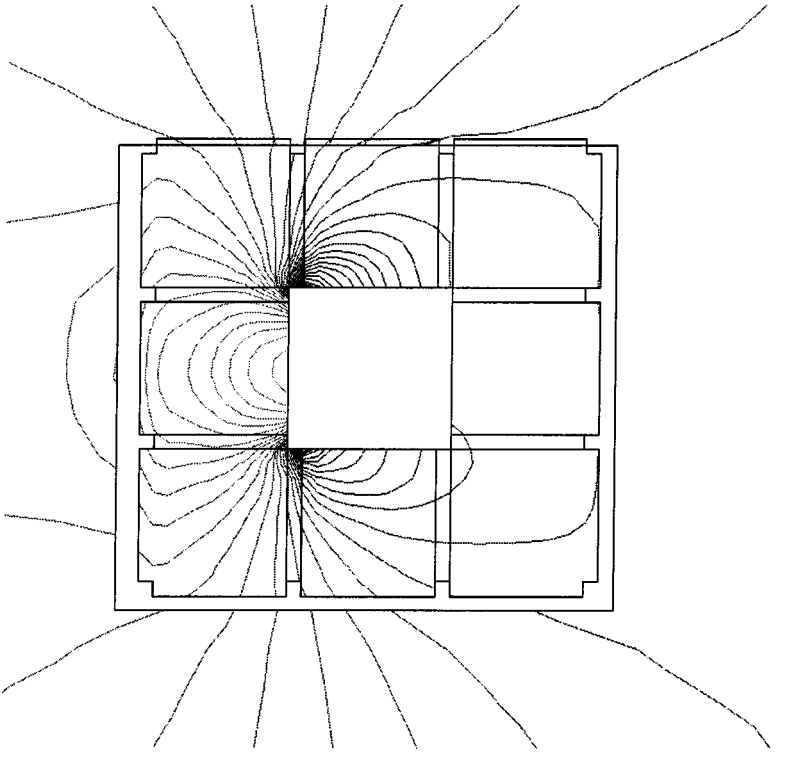

Figure 6. Pressure contours illustrating the continuity of the overset solution. Image is focused on the region of the square cylinder.

these images are contrasted to the dynamic fluid–solid simulation results for the equivalent flow and at equal times.

3.2. Example 2: Flow with a Dynamically Interacting Cylinder

Figure 7 displays the same cylinder as that used in the first example, except here it is tethered with a rigid member of zero mass to a frictionless hinge located on the axial centerline of the cylinder. The stress field imposed by the fluid flow is computed at each time step and the corresponding moment is evaluated. The angular acceleration is computed from the moment and the moment of inertia of the cylinder about the hinge so that the angular position, as a function of time may be determined. The grid resolution used in this example is the same as in Example 1, and the mass per unit length of the square cylinder is taken to be 150 kg/m.

Figure 8 displays a time sequence at 1·2 s intervals of the velocity vector field. Qualitatively, these results are consistent with the type of flow phenomena expected. Here, however,

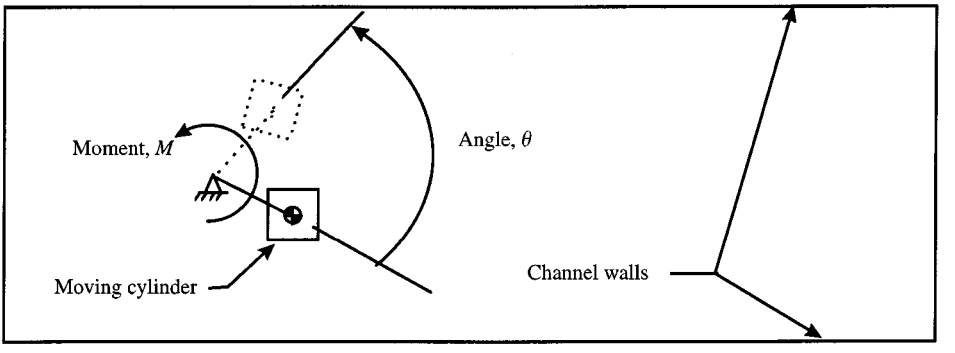

Figure 7. Definition sketch of dynamic simulation problem (Example 2).

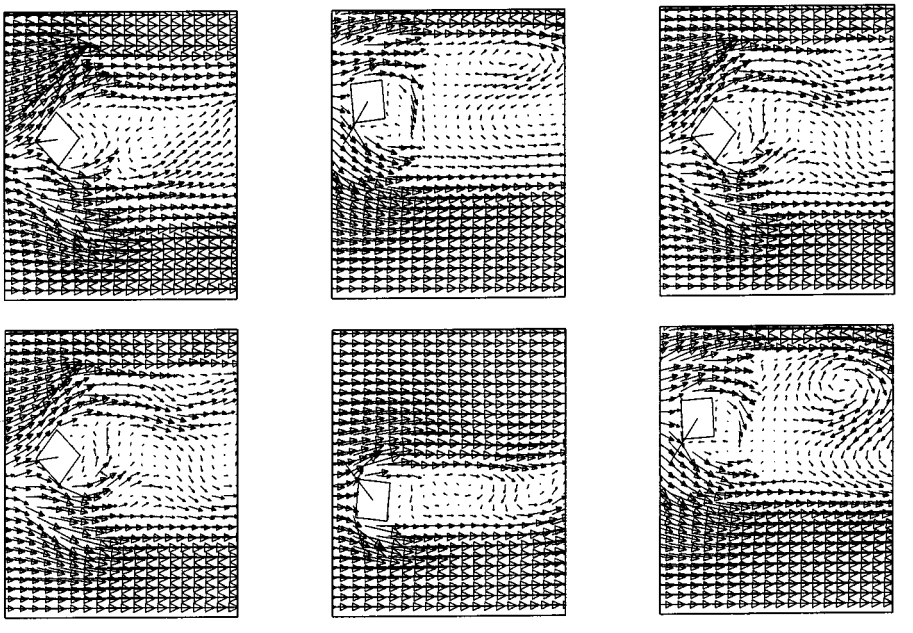

Figure 8. Time sequence of moving-cylinder position at 1·2 s intervals.

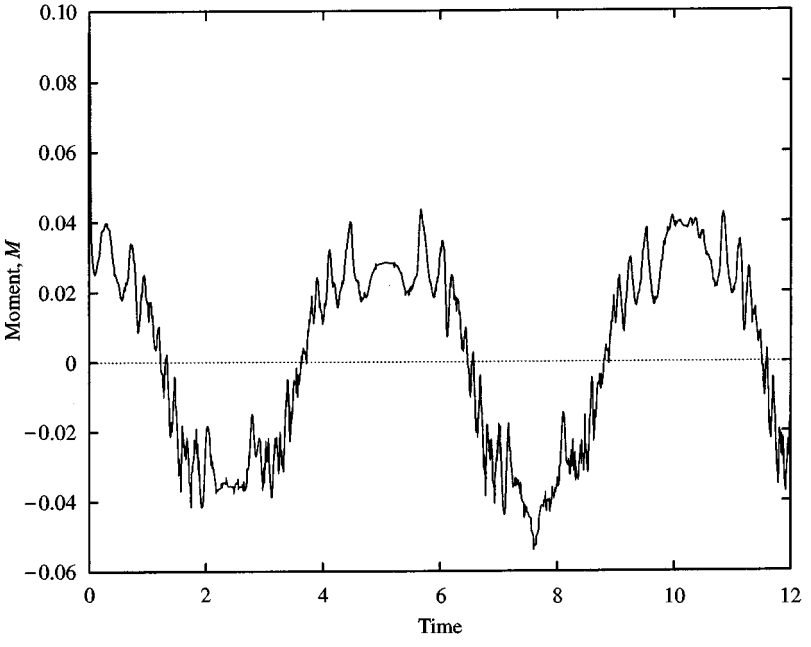

Figure 9. Moment on the solid cylinder induced by the fluid flow field.

the wake region is considerably more dynamic than that of the static case, due to the additional inertial component introduced by the dynamics of the solid object. Figures 9 and 10 show the periodic response of the object due to the asymmetric loading created by the fluid flowing around the object. The angular position of the object is nearly 180° out of phase with the fluid-induced moment on the object. Moreover, careful inspection of Figure 9 reveals an increasing amplitude for the angular displacement. The high-frequency, low-amplitude oscillations in the moment are caused by variations in the flow field around the object. As documented in Durao *et al.* (1995) the frequency of vortex shedding for this configuration is 4·7 Hz which corresponds to the high-frequency oscillations in the moment plot of Figure 9. Because of the relatively large moment of inertia of the object, the high-frequency moment oscillations are averaged over time and are not manifested in the angular displacement.

Finally, Figures 11–14 display the instantaneous velocity vector fields at different times. The top image is the dynamic cylinder, while the lower image displays results for the static cylinder configuration. Again, clearly seen is the amplified structure of the flow in the wake region behind the square cylinder. From these figures, it is clear that the intensity of the flow dynamics is amplified by the dynamic interaction of the movable solid object with the flow field; also, that the patched-overset grid method is capable of resolving the details of this phenomenon. Through the use of this method, one may perform fluid–solid interaction analyses, more accurately accounting for the additional effects due to the interaction.

4. CONCLUSION

A new method for simulating dynamic fluid–structure interaction based on patched-overset grids has been presented. The method extends techniques for patched block grid systems to

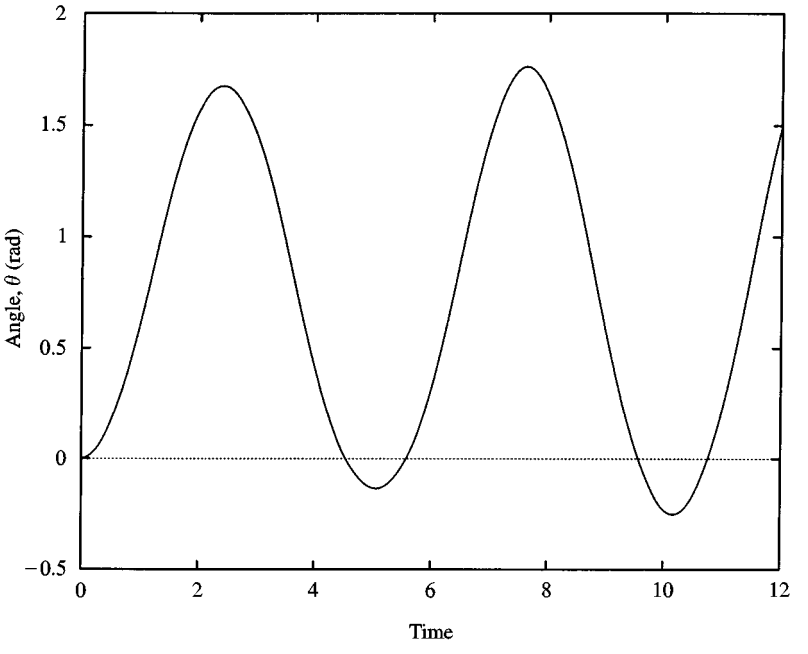

Figure 10. Angular position as a function of time for the solid cylinder.

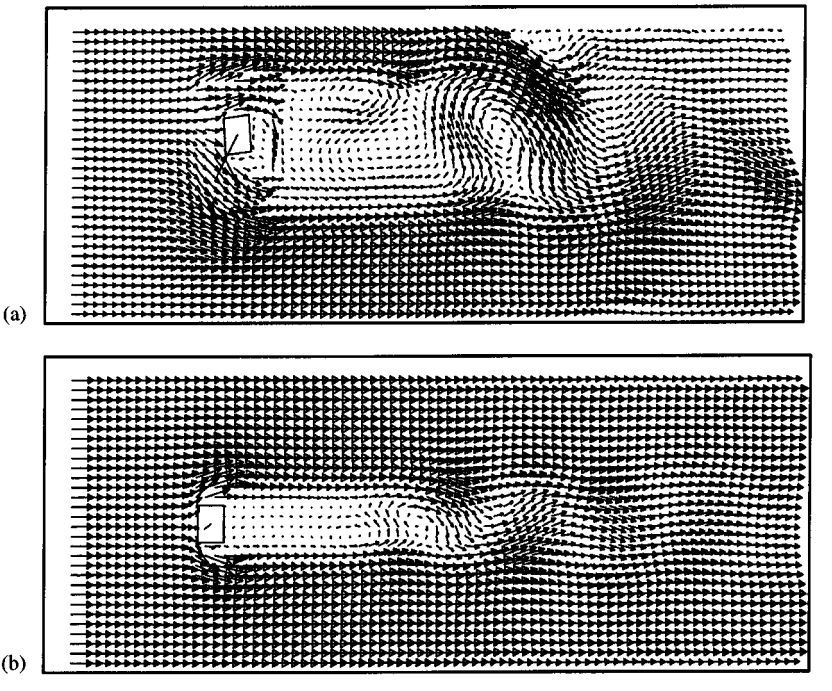

Figure 11. Comparison of flow fields for stationary and dynamic cylinder: (a) result for the dynamic cylinder; (b) result for the static cylinder. Time displayed is 2.4 s after initiation of flow.

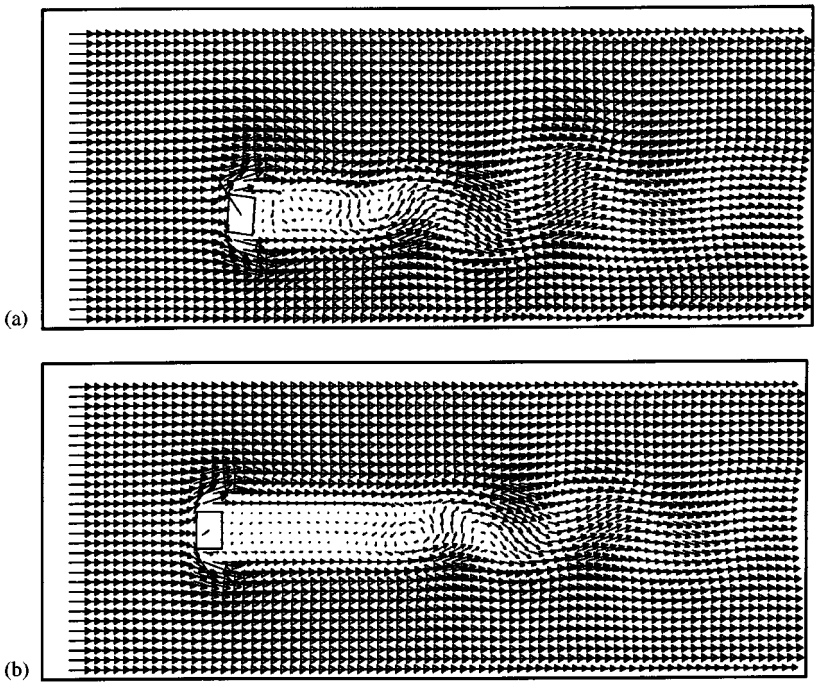

Figure 12. Comparison of flow fields for stationary and dynamic cylinder (time = 4·8 s): (a) for the dynamic cylinder; (b) for the static cylinder.

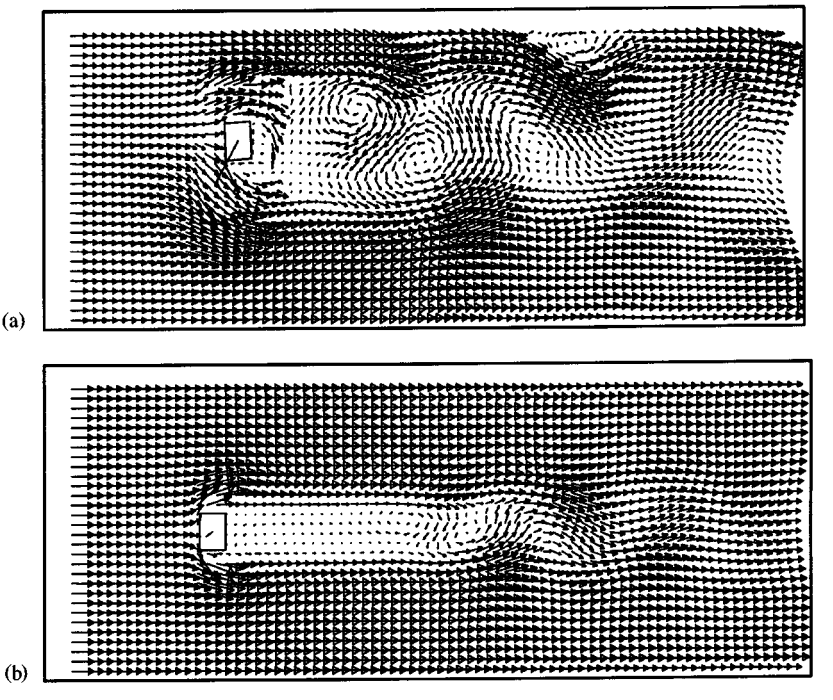

Figure 13. Comparison of flow fields for stationary and dynamic cylinder (time = 7·2 s): (a) for the dynamic cylinder; (b) for the static cylinder.

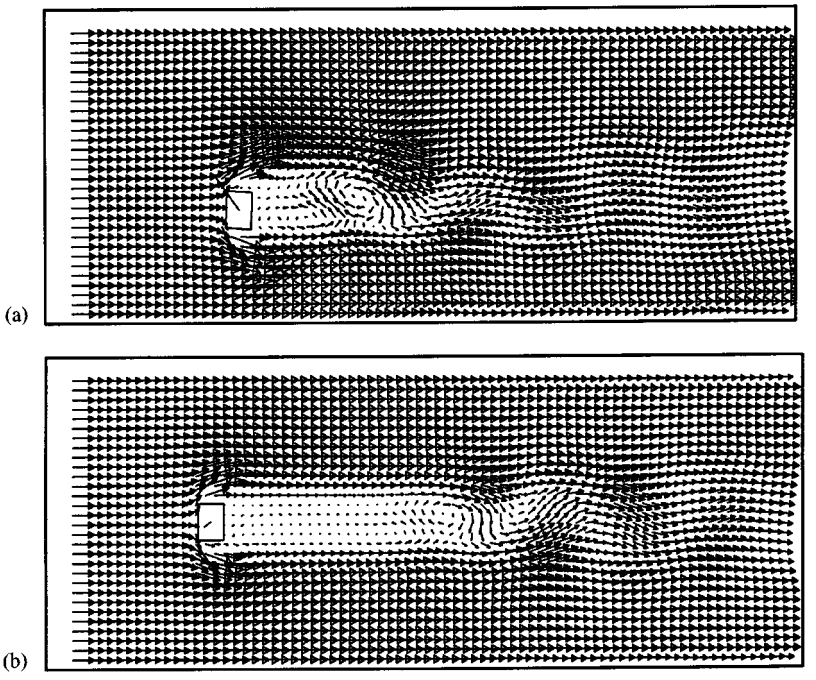

Figure 14. Comparison of flow fields for stationary and dynamic cylinder (time = 8·6 s): (a) for the dynamic cylinder; (b) for the static cylinder.

account for the additional data structures and numerical subtleties of overset grids. The method has been demonstrated on a realistic flow problem and shows the continuity of variables between both patched grid blocks and overset grid blocks. An aspect of this formulation is that the structure has no predefined motion and the interaction between the fluid and structure is fully accommodated. The method is based on an iterative block solution of the incompressible Navier–Stokes equations in which the flow field solution is computed on each grid block, and then primitive variable data is exchanged between appropriate neighbour grid blocks. This approach insures that, at iterative convergence, the critical requirement of continuity in the grid overlap and patched regions is achieved.

References

Atta, E. H. & Vadyak, J. 1982 A grid interfacing zonal algorithm for three-dimensional transonic flows about aircraft configurations. AIAA Paper A82-37477.

Benek, J. A., Donegan, T. L., & Suhs, N. E. 1987 Extended Chimera grid embedding scheme with application to viscous flows. AIAA 8th CFD Conference, 9–11 June, 1987, Honolulu, Hawaii, AIAA Paper 87–1126.

Bramble, J. H., Ewing, R. E., Parashkevov, R. R., & Pasciak, J. E. 1992 Domain decomposition methods for problems with partial refinement. *SIAM Journal on Scientific Computing* **13,** 397–410.

Chesshire, G. & Naik, V. K. 1994 An environment for parallel and distributed computation with application to overlapping grids. *IBM Journal of Research and Development* **38,** No. 3, May.

Durao, D. F. G., Heitor, M. V., & Pereira, J. C. F. 1986 A laser anemometry study of separated flow around a square obstacle. In *Laser Anemometry in Fluid Mechanics III* (eds Adrian, R. G. *et al.* pp. 227–243. Lisbon, Portugal: Ladoan-IST.

Durao, D. F. G., Heitor, M. V., & Pereira, J. C. F. 1988 Measurements of turbulent and periodic flows around a square cross-section cylinder. *Experiments in Fluids* **6,** 298–304.

Fejtek, I. & Roberts, L. 1992 Navier–Stokes computation of wing-rotor interaction for a tilt rotor in hover. *AIAA Journal* **30,** 2595–2603.

Freitas, C. J. 1986 Nonlinear transient phenomena in a three-dimensional cavity flow: a numerical investigation. Ph.D. Dissertation, Standford University, Standford, California, U.S.A.

Freitas, C. J. 1991 A computational fluid dynamics (CFD) program for advanced simulation of complex fluid flows. Final Report, SwRI Project No. 04-9478, October.

Freitas, C. J. 1995 Advanced computational simulation of flow phenomena associated with orifice meters. *Proceedings 3rd International Symposium on Fluid Flow Measurement*, 19–22 March, San Antonio, Texas.

Hubbard, B. J. & Chen, H.-C. 1995 Calculations of unsteady flows around bodies with relative motion using a chimera RANS method. *Proceedings 10th ASCE Engineering Mechanics Conference*, Vol. 2, University of Colardo, Boulder, 21–24 May.

Hubbard, B. J. & Chen, H.-C. 1994 A chimera scheme for incompressible viscous flows with application to submarine hydrodynamics. AIAA Paper 94-2210.

Jones, K. M., Biedron, R. T., & Whitlock, M. 1995 Application of a Navier–Stokes solver to the analysis of multi-element airfoils and wings using multizonal grid techniques. AIAA.

Kao, K.-H., Liou, M.-S. & Chow, C.-Y. 1994 Grid adaptation using chimera composite overlapping meshes. *AIAA Journal* **32,** 942–949.

Leonard, B. P. 1979 A stable and accurate convective modeling procedure based on quadratic upstream interpolation. *Computer Methods in Applied Mechanics and Engineering* **19,** 59–98.

Lijewski, L. E. & Suhs, N. E. 1994 Time accurate computational fluid dynamics approach to transonic store separation trajectory prediction. *Journal of Aircraft* **31,** 886–891.

Maliska, C. R. & Raithby, G. D. 1983 Calculating 3-D fluid flows using nonorthogonal grids. *Proceedings of 3rd International Conference on Numerical Methods on Laminar and Turbulent Flows*, pp. 651–666, Seattle, WA, U.S.A.

Meakin, R. L. & Suhs, N. E. 1989 Unsteady aerodynamic simulation of multiple bodies in relative motion. AIAA Paper 89-1996.

Moore, P. K. & Flaherty, J. E. 1992 Adaptive local overlapping grid methods for parabolic systems in two space dimensions. *Journal of Computational Physics* **98,** 54–63.

Nacul, E. C., Lepretre, C., Pedrosa, O. A. Jr., Petrobras, G. P. & Aziz, K. 1990 Efficient use of domain decomposition and local grid refinement in reservoir simulation. *Society of Petroleum Engineers*, **SPE 20740,** 245–255.

Petersson, N. A. 1993 Computing periodic gravity waves on water by using moving composite overlapping grids. *SIAM Journal on Scientific Computing* **14,** 1339–1358.

Rhie, C. M. & Chow, W. L. 1982 A numerical study of the turbulent flow past an isolated airfoil with trailing edge separation. AIAA Paper 82-0998.

Shahcheraghi, N. & Dwyer, H. A. 1994 Fluid flow and heat transfer over 3-D spherical object in a pipe. In *Proceedings Fundamentals of Heat Transfer in Forced Convection*, HTD-Vol. 285, pp. 93–100. New York: ASME.

Steger, J. L., Dougherty, F. C. & Benek, J. A. 1983 A chimera grid scheme. In *Advances in Grid Generation*, FED-Vol. 15, New York: ASME.

Tu, J. Y. & Fuchs, L. 1992 Overlapping grids and multigrid methods for three-dimensional unsteady flow calculations in IC engines. *International Journal of Numerical Methods in Fluids* **15,** 693–714.

Tu, J. Y. & Fuchs, L. 1995 Calculation of flows using three-dimensional overlapping grids and multigrid methods. *International Journal for Numerical Methods in Engineering* **38,** 259–282.

Venkata, R. G., Oliger, J., & Ferziger, J. 1992 Composite grids for flow computations on complex 3D domains. *Fifth International Symposium on Domain Decomposition Methods for Partial Differential Equations* (eds D. E. Keyes, T. F. Chan, G. Meurant, J. S. Sroggs & R. G. Voigt), pp. 605–613, Philadelphia, PA: SIAM.