Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Computational Physics 191 (2003) 567–600
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

An overset-grid method for 3D unsteady incompressible flows

H.S. Tang, S. Casey Jones, Fotis Sotiropoulos *

School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0355, USA

Received 28 February 2002; received in revised form 12 June 2003; accepted 23 June 2003

Abstract

A composite-grid numerical method is developed for simulating unsteady, three-dimensional (3D), incompressible flows in complex geometries. The governing equations are solved using a second-order accurate, finite-volume method based on the dual time-stepping artificial compressibility approach. Overset grids are employed to discretize arbitrarily complex geometries, and a new interface algorithm is developed to facilitate communication between neighboring grids. The algorithm is inspired by the necessary and sufficient conditions for satisfying global mass conservation in a composite domain and is simple to implement in 3D. Numerical experiments show that the new interpolation scheme is superior to straightforward, trilinear interpolation of all flow variables as it minimizes non-physical spurious oscillations in the overlap region, is less sensitive to grid refinement, and greatly enhances the computational efficiency of the iterative algorithm. The advantages of the new method are especially pronounced when adjacent overset subdomains are discretized with different spatial resolutions. The potential of the method as a powerful technique for simulating complex engineering flows is demonstrated by applying it to calculate vortex shedding from a circular cylinder mounted between two endplates and flow in a rectangular channel with two wall-mounted obstacles.

AMS: 65M55; 76D05

Keywords: Overset grids; Chimera grids; Grid interfaces; Multigrid acceleration; Incompressible Navier–Stokes equations; Artificial compressibility method; Dual time-stepping

1. Introduction

Most flows of practical interest in engineering and bioengineering take place in geometrically complex domains and are dominated by unsteadiness and shedding of three-dimensional (3D) coherent vortices. Typical examples range from the hydraulics of complex bridge foundations in natural rivers [1] to the hemodynamics of mechanical, prosthetic heart valves [2]. A critical prerequisite for the accurate numerical simulation of such flows is the development of robust and accurate algorithms capable of handling complex

* Corresponding author. Tel.: +1-404-894-4432; fax: +1-404-385-1131.
E-mail address: fsotirop@ce.gatech.edu (F. Sotiropoulos).

0021-9991/$ - see front matter © 2003 Published by Elsevier B.V.
doi:10.1016/S0021-9991(03)00331-0

multiconnected geometrical configurations. Constructing a good quality single-block, structured, curvilinear mesh for such complex geometries, however, is often a formidable if not impossible undertaking.

Algorithms employing unstructured grids provide a powerful alternative for simulating flows in arbitrarily complex geometries and have been successfully applied in recent years to a number of complex compressible and incompressible flows (see [3] and references cited there for a review of related literature). Unstructured grids facilitate grid generation in complex domains and allow for the relatively easy implementation of locally adaptive algorithms. Such methods, however, could be memory and computational time intensive and in general are less amenable, as compared to structured grid methods, to the implementation of high resolution, spatial discretization schemes. Moreover, generating a fully unstructured grid near solid boundaries in high Reynolds-number flow simulations is far from trivial. In fact, most available unstructured algorithms have to resort to locally structured grids to accurately resolve near-wall boundary layers.

An alternative to unstructured grids is the so-called composite, overset grid approach pioneered by Benek et al. [4]. A complex geometrical configuration is decomposed into a set of simpler, overlapping subdomains, each designed so that it can be easily discretized with a set of simple, boundary-conforming, curvilinear coordinates. The governing equations are solved independently in each subdomain and information from one subdomain to another is transferred via the specification of interfacial boundary conditions. Compared to their unstructured grid counterpart, composite structured grid algorithms are worth pursuing because they: (1) lead to more efficient numerical algorithms; (2) simplify the implementation of high-resolution discretization schemes; (3) allow for efficient clustering of grid surfaces near solid walls, thus, enhancing accuracy in high Reynolds-number simulations; and (4) provide a natural level of parallelism for execution on massively parallel computers. A major difficulty in the implementation of overset-grid algorithms in complex, unsteady flows, however, stems from the need to specify boundary conditions for all flow variables at interfaces between adjacent subdomains. In complex, unsteady flow simulations such boundary conditions should ensure that vortical structures generated in one subdomain can cross grid interfaces and interact with the flow in adjacent subdomains without distortion.

The simplest approach for specifying interface boundary conditions is to interpolate all primitive variables from one subdomain to the other [4–8]. This approach, however, does not guarantee that the so obtained numerical solution will be globally conservative. The term globally conservative implies that the fluxes of conserved quantities (mass, momentum and energy) integrated over the boundary of the composite domain add up to zero. Lack of global conservation could be detrimental to the accuracy and smoothness of the computed solution, especially in incompressible flow simulations where global mass conservation is a necessary condition for the existence and uniqueness of a smooth pressure field [9]. Conservative composite-grid methods can be developed by patching the grids of adjacent subdomains along a common grid surface. Global conservation can then be ensured using flux interpolation techniques such as those developed by Rai [10], for compressible flows, and Wright and Shyy [11], for incompressible flows. Patched grids have also been used to design locally adaptive [12] and grid embedding algorithms for resolving near-wall regions in DNS and LES [13,14] of turbulent flows. The usefulness of patched grids, however, is rather limited in complex geometries as the requirement to exactly match grids along the interface can be very stringent in arbitrarily complex domains. Overset Chimera grids [4] provide a far more flexible alternative because the grids of adjacent subdomains may be arbitrarily overlapped. The main disadvantage of overset grids, however, is that global conservation is very difficult to enforce [15], especially in 3D. For that reason the prevailing wisdom in the literature is that overset grids should be avoided in flows involving large solution gradients and strong elliptic effects, such as large regions of separated flow [11,15]. In spite of these difficulties, overset-grid approaches for incompressible flow simulations have attracted considerable attention from a number of researchers in recent years. In what follows we present a brief review of recent literature with an emphasis on composite-grid methods employing Chimera, overset grids.

Henshaw [16] developed a pressure-based, fourth-order accurate method for solving the Navier–Stokes equations on overlapping grids. Boundary conditions at grid interfaces were specified for the velocity and pressure fields using interpolation based on Lagrange polynomials. The accuracy of the method was demonstrated by applying it to simulate 2D flow past a cylinder at $Re = 100$. Sample 3D results were also reported for the so-called *twilight-zone flow* (a benchmark, analytical solution) and steady flow past a sphere at an unspecified Reynolds number.

Zang and Street [17] developed a second-order accurate, fractional-step, finite-volume, multigrid method for solving the 3D, unsteady, Navier–Stokes equations on composite overset grids. They employed Lagrangian, bi-quadratic interpolation to transfer data between subdomains. Due to the non-conservative character of their interpolation scheme, Zang and Street [17] had to explicitly implement a mass imbalance correction on the interpolated velocity field to ensure global mass conservation in each subdomain. Zang and Street [17] applied their method to simulate spin-down flow in a circular cylinder, steady flow in a polar cavity, and upwelling flow in a rotating axisymmetric container.

Hubbard and Chen [9] developed a finite-analytic, SIMPLE-type algorithm for solving the Navier–Stokes equations on Chimera overset grids. They reported that using trilinear interpolation at grid interfaces for all flow variables leads to oscillatory pressure and velocity fields and proposed an approach for correcting the interpolated velocity field such that the continuity equation is enforced at every node near a grid interface. Hubbard and Chen simulated steady 2D flow past a circular cylinder at $Re = 30$ and also solved the axisymmetric, Reynolds-averaged Navier–Stokes equations for flow past an axisymmetric submarine-like hull.

Freitas and Runnels [8] developed an overset-grid approach to simulate 2D flows with fluid/structure interaction. They employed bilinear interpolation for all flow variables at grid interfaces and reported that they were able to circumvent the difficulties associated with the lack of conservation at the interface by ensuring that their iterative algorithm converges to a sufficiently small residual level at each time step.

More recently, Burton and Eaton [18] developed a second-order accurate, fractional-step method for solving the unsteady, Navier–Stokes equations on staggered, overset grids. They employed non-conservative, quadratic interpolation to specify boundary conditions at grid interfaces and reported good results for the well known Taylor vortex flow and unsteady 2D flow past a circular cylinder at $Re = 100$.

The above brief literature review points to the progress and growing interest in the development and application of overset-grid techniques for solving the incompressible Navier–Stokes equations. It is important to point out, however, that for the most part previous studies have focused on relatively simple steady and unsteady flows. Therefore, the ability of overset-grid-based algorithms to carry out reliable simulations of massively separated, 3D flows has yet to be demonstrated. It also apparent from the above review that the search for an interface interpolation scheme that is simple to implement in 3D flows and has good conservation properties is still ongoing. In this work, we seek to contribute both toward the development of an improved interface interpolation scheme and to demonstrate the potential of the overset-grid approach in simulations of complex, unsteady, 3D flows.

We propose a practical methodology for solving the unsteady, 3D, Navier–Stokes equations in arbitrarily complex geometries using overset grids. We develop a second-order accurate interface algorithm, which is inspired by the necessary and sufficient conditions required for satisfying global mass conservation in the composite domain. The proposed algorithm determines the velocity components at the interface by ensuring consistency of the mass flux across the interface. Unlike other conservative schemes of flux interpolation type, the present algorithm neither interpolates geometry [10,19] nor requires patching of the overset grids [15] and, thus, is very simple to implement in 3D flows. Furthermore, our mass-flux based interpolation (MFBI) algorithm does not require explicit correction of the interpolated velocity field to ensure global mass conservation (as in [9,17]) – such corrections would be tedious to implement in situations where multiple grids overlap and could complicate the implementation of the method in 3D. The base flow solver is a second-order accurate, dual time-stepping, artificial compressibility method developed by

Sotiropoulos and Ventikos [20]. We compare the MFBI algorithm with an algorithm based on straightforward trilinear interpolation of all flow variables, which we shall refer to in the remainder of this paper as the standard interpolation (SI) method. The two interface treatment algorithms are applied to calculate steady and unsteady flows in a 3D lid-driven cavity and steady flow in a 90°, strongly curved, circular pipe. For both cases, the computations are carried out using overset-grid layouts. We show that MFBI accelerates the convergence of the iterative algorithm and minimizes oscillations of the pressure at discontinuous grid interfaces. We also demonstrate the potential of our method in simulations of complex, massively separated flows for two test cases. The first case is laminar flow past a circular cylinder mounted between two endplates at Reynolds numbers $Re = 100$ and 300, based on the approach velocity and cylinder diameter. Numerical simulations for this case have been recently reported by Mittal [21] using a finite-element method on an unstructured grid. The second test case is flow in a rectangular channel with two wall-mounted obstacles. The computational domain is discretized using six overset grids and calculations are carried out for $Re = 100$ and 300.

The paper is organized as follows. In Section 2, we describe briefly the basic flow solver. In Section 3, we discuss conditions for satisfying global mass conservation in composite domains. In Section 4, we present the MFBI algorithm and describe its implementation in generalized curvilinear coordinates. In Section 5, we discuss issues related to grid connectivity and summarize the entire domain decomposition algorithm. In Section 6, we compare the accuracy and efficiency of MFBI and SI approaches for laminar flows in a lid-driven cavity and through a strongly curved pipe. In Section 7, we report computations for two complex laminar flows involving unsteady shedding from wall-mounted obstacles. A summary of our findings along with conclusions and recommendations for future work are given in Section 8.

2. The base numerical method

In this section we describe the numerical method used to solve the unsteady, 3D Navier–Stokes equations. Various aspects of this method have been described in detail in previous archival publications [20,22] and for this reason we only present a brief summary of the methodology. Issues related to the numerical implementation of the domain decomposition method and a summary of the overall algorithm are discussed in the subsequent section.

We solve numerically the unsteady, 3D, incompressible, Navier–Stokes equations in generalized curvilinear coordinates (repeated indices imply summation – ℓ, $r = 1$–3):

$$\Gamma \frac{\partial Q}{\partial t} + J \frac{\partial}{\partial \xi^\ell} \left(F^\ell - F_v^\ell \right) = 0, \tag{1}$$

where

$$\begin{aligned} \Gamma &= \mathrm{diag}(0,1,1,1), \\ Q &= (p, u_1, u_2, u_3)^{\mathrm{T}}, \\ F^\ell &= \frac{1}{J}\left(U^\ell, u_1 U^\ell + p\xi^\ell_{x_1}, u_2 U^\ell + p\xi^\ell_{x_2}, u_3 U^\ell + p\xi^\ell_{x_3} \right)^{\mathrm{T}}, \\ F_v^\ell &= \frac{1}{JRe}\left(0, g^{r\ell}\frac{\partial u_1}{\partial \xi^r}, g^{r\ell}\frac{\partial u_2}{\partial \xi^r}, g^{r\ell}\frac{\partial u_3}{\partial \xi^r} \right)^{\mathrm{T}}, \end{aligned} \tag{2}$$

where p is the static pressure divided by the density, u_ℓ are the Cartesian velocity components, U^ℓ are the contravariant velocity components ($= u_m \xi^\ell_{x_m}$) and $\xi^\ell_{x_m}$ are the metrics of the geometric transformation, J is the Jacobian of the geometric transformation, $g^{r\ell}$ is the contravariant metric tensor $g^{r\ell} = \xi^r_{x_s}\xi^\ell_{x_s}$, and Re is

the Reynolds number. The system of governing equations (1) is discretized in time using the three-point backward, second-order accurate, Euler implicit temporal integration scheme. The spatial derivatives are discretized in strong conservation form using three-point, second-order accurate central differencing. The resulting discrete approximation of Eq. (1) reads as follows:

$$\Gamma \frac{1}{J}\frac{3Q^{n+1}_{i,j,k} - 4Q^{n}_{i,j,k} + Q^{n-1}_{i,j,k}}{2\Delta t} + \left(\delta_{\xi^\ell}\tilde{F}^{\ell} + \delta_{\xi^\ell}\tilde{F}^{\ell}_{v}\right)^{n+1}_{i,j,k} = 0, \tag{3}$$

where

$$\delta_{\xi^1}()_{i,j,k} = \frac{()_{i+1/2,j,k} - ()_{i-1/2,j,k}}{\Delta\xi^1},$$

and $\tilde{F}$ denotes discrete approximation of the flux F at the cell interfaces. The convective flux vectors at the cell interfaces are approximated as follows:

$$\tilde{F}^{1}_{i+1/2,j,k} = \frac{1}{2}\left(F^{1}_{i+1,j,k} + F^{1}_{i,j,k}\right) + D^{1}_{i+1/2,j,k}, \tag{4}$$

where D^1 is an artificial dissipation flux. Lin and Sotiropoulos [22] have implemented and evaluated several approaches for constructing dissipation fluxes, including scalar and matrix-valued artificial dissipation and flux-difference splitting upwinding. They showed that when all schemes are compared on the same mesh resolution, the matrix-valued scheme [22,23] introduces the least overall amount of artificial dissipation. In this work we adopt the matrix valued scheme and calculate the dissipation flux as follows:

$$D^{1}_{i+1/2,j,k} = \varepsilon\delta_{\xi^1}\left(|A^1|\delta_{\xi^1}\delta_{\xi^1}\right)Q_{i+1/2,j,k}, \tag{5}$$

where $|A^1|$ is the absolute value of the Jacobian matrix $A^1 = \partial F^1/\partial Q$ (see [22] for detailed expressions for the various matrices for the incompressible Navier–Stokes equations) and ε is a constant that controls the amount of artificial dissipation. The calculation of the viscous fluxes at cell interfaces is straightforward. The metrics of the geometric transformation are calculated using simple averaging while the spatial derivatives of the velocity field are calculated using three-point, central differencing (see [22,24] for details).

Since Γ is a singular matrix, Eq. (3) cannot be marched in time to advance the solution to the next time step. Moreover, the temporal discretization scheme is fully implicit so solving Eq. (3) requires an iterative algorithm. Such an algorithm can be designed by adopting the dual- (or pseudo-) time-stepping artificial compressibility approach [25], as follows:

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\left(Q^{m+1}_{i,j,k}\right) + R^{m+1}_{i,j,k} = 0, \tag{6}$$

where τ is the pseudo-time variable, m denotes the iteration level in pseudo-time, and R is the residual vector given by

$$R^{m+1}_{i,j,k} = \Gamma\frac{1}{J}\frac{3Q^{m+1}_{i,j,k} - 4Q^{n}_{i,j,k} + Q^{n-1}_{i,j,k}}{2\Delta t} + \left(\delta_{\xi^\ell}\tilde{F}^{\ell} + \delta_{\xi^\ell}\tilde{F}^{\ell}_{v}\right)^{m}_{i,j,k}. \tag{7}$$

Eq. (6) can be integrated in pseudo-time until the pseudo-time derivative is reduced to a prescribed small tolerance, m and $m+1 \to n+1$, and the governing equations are satisfied at the $n+1$ time level (i.e., $R^{n+1} \to 0$). In this work, we integrate Eq. (6) by adopting the point-wise implicit, four-stage, Runge–Kutta algorithm described in [20]. The convergence of the iterative algorithm is enhanced using local (dual-) time-stepping, implicit residual smoothing, and V-cycle multigrid acceleration (see [20,26] for details). As

discussed in [20], the point-wise implicit discretization of R in pseudo-time (see Eq. (7)) enhances considerably the robustness of the iterative algorithm (by allowing the use of large physical time steps Δt) without the computational overhead associated with a fully implicit (pseudo-) time-marching iteration scheme.

3. Conditions for global mass conservation on overset grids

In this section we discuss the conditions that need to be satisfied to enforce global mass conservation in a composite domain discretized with overset grids. Before we proceed with our discussion, it is important to point out that using conservative schemes in each overset subdomain is not by itself sufficient to guarantee global conservation in the composite domain. As discussed in [15], the presence of grid interfaces creates artificial boundaries in the interior of the flow domain, which must be treated conservatively for global conservation to be satisfied. We should also comment that we focus exclusively on mass conservation because it is the most critical prerequisite for designing well-posed numerical algorithms for solving the incompressible Navier–Stokes equations.

With reference to Fig. 1, consider the composite domain Ω obtained by overlaying domains Ω_A and Ω_B – that is, $\Omega = \Omega_A \cup \Omega_B$. Let Γ_A be the boundary of Ω_A and Γ_a the portion of this boundary that is within the overlap region. By adopting a similar notation for Ω_B, the boundary Γ of Ω can be written as $\Gamma = \Gamma_A - \Gamma_a + \Gamma_B - \Gamma_b$. For incompressible flow, global mass conservation in Ω requires that

$$\int_{\Gamma} \vec{V} \cdot \vec{n}\,\mathrm{d}S = \int_{\Gamma_A} \vec{V}^A \cdot \vec{n}\,\mathrm{d}S - \int_{\Gamma_a} \vec{V}^A \cdot \vec{n}\,\mathrm{d}S + \int_{\Gamma_B} \vec{V}^B \cdot \vec{n}\,\mathrm{d}S - \int_{\Gamma_b} \vec{V}^B \cdot \vec{n}\,\mathrm{d}S = 0, \tag{8}$$

where $\vec{V}$ is the velocity vector, $\vec{n}$ is the outward pointing unit normal vector, and the superscripts A and B indicate the subdomain in which the velocity vector has been calculated. Assuming that conservative numerical schemes are used in each subdomain, it follows that

$$\begin{aligned} &\int_{\Gamma_A} \vec{V}^A \cdot \vec{n}\,\mathrm{d}S = 0, \\ &\int_{\Gamma_B} \vec{V}^B \cdot \vec{n}\,\mathrm{d}S = 0. \end{aligned} \tag{9}$$

From (8) and (9) we obtain:

$$\int_{\Gamma_a} \vec{V}^A \cdot \vec{n}\,\mathrm{d}S + \int_{\Gamma_b} \vec{V}^B \cdot \vec{n}\,\mathrm{d}S = 0. \tag{10}$$

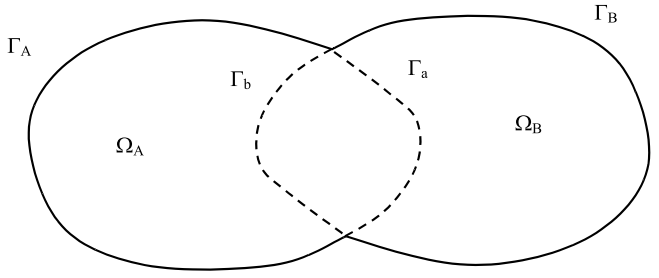


Fig. 1. Schematic of a composite domain with two overset subdomains.

Since the overlap region belongs to both subdomains, the following conservation relations should be satisfied for physically meaningful solutions [15]:

$$\begin{aligned}\int_{\Gamma_a} \vec{V}^A \cdot \vec{n}\,dS + \int_{\Gamma_b} \vec{V}^A \cdot \vec{n}\,dS = 0,\\ \int_{\Gamma_a} \vec{V}^B \cdot \vec{n}\,dS + \int_{\Gamma_b} \vec{V}^B \cdot \vec{n}\,dS = 0.\end{aligned} \tag{11}$$

Combining (10) and (11), we obtain the following conditions for global mass conservation to be satisfied in the composite domain [15]:

$$\int_{\Gamma_a} \vec{V}^A \cdot \vec{n}\,dS = \int_{\Gamma_a} \vec{V}^B \cdot \vec{n}\,dS \quad \text{and} \quad \int_{\Gamma_b} \vec{V}^A \cdot \vec{n}\,dS = \int_{\Gamma_b} \vec{V}^B \cdot \vec{n}\,dS. \tag{12}$$

In other words, assuming that conservative numerical schemes are used in each subdomain, global mass conservation in the composite domain is satisfied if at each interface of the overlap region the mass fluxes calculated using velocities from domains A and B are equal to each other. Obviously, if boundary conditions at the interface are specified using standard interpolation there is no guarantee that Eqs. (12) are satisfied other than in the limit of the grid spacing approaching zero. In fact satisfying Eq. (12) exactly can be very difficult especially in 3D flows. Approaches for enforcing strict conservation have been proposed by Berger [19] and Wang [15] but both are very tedious to implement in 3D geometries. In the following section, we propose a simple approach for satisfying a second-order approximation of Eq. (12). Strictly speaking, our approach satisfies mass conservation exactly only in the limit of zero grid spacing but it is easy to implement and, as will be shown in subsequent sections, greatly improves the efficiency, accuracy, and smoothness of the solution as compared to the standard interpolation.

4. Interface treatment using MFBI

Consider the two overset grids, Ω_A and Ω_B, shown in Fig. 2. Assume that the solution in Ω_B is known and that we seek to construct boundary conditions on the interface S in order to solve the governing equations in Ω_A. We develop an interpolation scheme, which is based on enforcing a second-order accurate approximation of the mass flux constraint given by Eq. (12) on S' – the surface located half a grid node away from the interface S. Note that we enforce the mass-flux constraint on S', rather than on S itself, because we discretize the governing equations in strong-conservation form using three-point central differencing (see Eq. (3)). Thus, summation of the discrete equations over all interior nodes of Ω_A results in boundary fluxes at the grid surface half a grid node away from the boundary.

Using the index and curvilinear coordinate notation of Fig. 2, a discrete approximation of Eq. (12) can be written as follows:

$$\sum_{j,k\in S'} \left(\frac{U^1}{J}\right)_{3/2,j,k} \Delta\xi^2\Delta\xi^3 = \sum_{q\in S'^B} \left(\frac{U^1}{J}\right)^B_q \Delta\xi^2\Delta\xi^3. \tag{13}$$

The summation on the left-hand side of Eq. (13) is over the grid nodes of Ω_A along the S' boundary. The summation in the right-hand side of Eq. (13), on the other hand, is taken over the points q defined by the intersection of the surface S' with the Ω_B grid, denoted as S'^B. As we discussed above, calculating this term exactly, by adopting procedures such as those proposed in [15] and [19], can be very tedious in 3D geometries. We propose, therefore, to approximate Eq. (13) as follows:

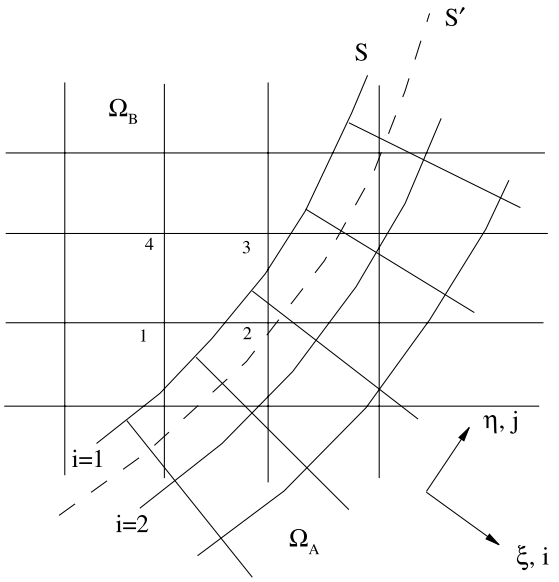


Fig. 2. Schematic of two Chimera overset grids.

$$\sum_{j,k\in S'}\left(\frac{U^1}{J}\right)_{3/2,j,k}\Delta\xi^2\Delta\xi^3=\sum_{j,k\in S'}\left[\frac{\left(\Im_B^A u_\ell\right)\xi^1_{x_\ell}}{J}\right]_{3/2,j,k}\Delta\xi^2\Delta\xi^3, \tag{14}$$

where $\Im_B^A$ is an operator that interpolates the velocity components from grid Ω_B to the nodes of grid Ω_A on the interface S' – second-order accurate trilinear interpolation is used for this purpose. In other words, rather than constructing the mass flux from the Ω_B grid exactly (by considering the contributions from all points on S'^B) we interpolate the Cartesian velocity components from Ω_B to the nodes of Ω_A located on S' and use these interpolated values to construct an approximate mass flux from Ω_B. Obviously the integral constraint given by (14) can be satisfied by setting the U^1 contravariant velocity component at every grid node on S' equal to the U^1 contravariant velocity component calculated using the interpolated values from Ω_B. That is,

$$\left(\frac{U^1}{J}\right)_{3/2,j,k}=\left[\frac{\left(\Im_B^A u_\ell\right)\xi^1_{x_\ell}}{J}\right]_{3/2,j,k}. \tag{15}$$

Since we employ three-point, second-order accurate central differencing to discretize the governing equations, Eq. (15) can be approximated up to second-order accuracy as follows:

$$\left(\frac{U^1}{J}\right)_{3/2,i,k}=\frac{1}{2}\left[\left(\frac{U^1}{J}\right)_{1,j,k}+\left(\frac{U^1}{J}\right)_{2,j,k}\right]=\left[\frac{\left(\Im_B^A u_\ell\right)\xi^1_{x_\ell}}{J}\right]_{3/2,j,k}. \tag{16}$$

The above equation is used to construct an algorithm for calculating $(U^1)^{n+1}$ at every node on S during the iterative solution (pseudo-time iteration) of the governing equations in Ω_A. Using the notation of the dual time-stepping iterative algorithm described in Section 2 (see Eq. (6)), we obtain:

$$\left(\frac{U^1}{J}\right)^{m+1}_{1,j,k} = 2\left[\frac{\left(\mathfrak{I}^{\mathrm{A}}_{\mathrm{B}} u_\ell\right)\xi^1_{x_\ell}}{J}\right]_{3/2,j,k} - \left(\frac{U^1}{J}\right)^{m}_{2,j,k}. \tag{17}$$

The second term in the right-hand side of Eq. (17) can be calculated from the calculated velocity field at the interior nodes $(2,j,k)$. Eq. (17) is not by itself sufficient to specify boundary conditions on Γ for all four primitive variables (velocity components and pressure). To calculate all three Cartesian velocity components on S we set the U^2 and U^3 contravariant velocity components at $(1,j,k)$ equal to those calculated using interpolated velocities from Ω_{B}. That is (for $q = 2,3$),

$$\left(\frac{U^q}{J}\right)^{m+1}_{1,j,k} = \left[\frac{\left(\mathfrak{I}^{\mathrm{A}}_{\mathrm{B}} u_\ell\right)\xi^q_{x_\ell}}{J}\right]_{1,j,k}. \tag{18}$$

Eqs. (17) and (18) form a linear system of three equations, which is inverted to obtain the three Cartesian velocity components at $(1,j,k)$. Finally, the pressure is kept fixed throughout the iteration in pseudo-time and equal to the interpolated value:

$$p^{m+1}_{1,j,k} = \mathfrak{I}^{\mathrm{A}}_{\mathrm{B}} p_{1,j,k}. \tag{19}$$

Note that when the pseudo-time-marching iteration converges, m and $m+1 \to n+1$ and Eqs. (17)–(19) are satisfied at the $n+1$ time level.

An important shortcoming of the SI approach in incompressible flow simulations is the fact that both the velocity vector and the pressure are specified at the interface. A well-posed problem for the incompressible Navier–Stokes equations, however, requires boundary conditions only for the velocity field and the pressure can be obtained as part of the solution [27]. Thus, interpolating and keeping fixed all four primitive variables at the interface over specifies the discrete problem and could result in an ill-posed algorithm. In fact we subsequently show via numerical experiments that when the SI approach is used the numerical problem does indeed become ill posed as indicated by sluggish convergence rates of the iterative algorithm and the appearance of non-physical pressure and velocity oscillations in the computed solutions. The MFBI approach, on the other hand, does not overspecify the discrete problem. This is because only three, Dirichlet-type, boundary conditions are specified at the interface using interpolated variables for (see Eqs. (18) and (19)). The condition enforced by Eq. (17) is actually an integral constraint that guarantees global mass conservation. Note that such a constraint is required for the initial-boundary value problem of the incompressible Navier–Stokes equations to be well posed. As discussed in [27], the Dirichlet conditions for the velocity field should satisfy global mass conservation for a well-posed problem. In other words, the MFBI algorithm can be thought of as specifying three of the four primitive variables at the interface and determining the fourth variable such that the global mass conservation constraint is satisfied (at least in the approximate sense discussed above).

5. Grid connectivity and summary of the domain decomposition algorithm

In this section we discuss issues related to the numerical implementation of the domain decomposition algorithm and summarize the overall solution procedure.

The first step in implementing a domain decomposition method is to decompose the computational domain into a set of subdomains. The decomposition may be carried out to facilitate either the description of the geometry (using subsets of easy to generate curvilinear coordinates), or the resolution of complex flow structures, or the efficient parallelization of the computer code, or a combination of all of the above. After the domain decomposition has been completed, the relative location of the various subdomains and

the manner in which they are connected with each other need to be determined. The solution of this problem, which is collectively referred to as grid connectivity, is not trivial especially when a large number of overset subdomains is employed.

In this work, we develop an efficient procedure for solving grid connectivity using a Newton iteration algorithm. Due to space considerations, we provide only a brief summary of the procedure. The reader is referred to Tang [24] for a detailed description. With reference to Fig. 2, the host cell for node p at the boundary of Ω_A is cell 1–2–3–4 in Ω_B. After the host cell has been determined, the curvilinear coordinates of node p in Ω_B need to be calculated. These coordinates are defined using trilinear interpolation between the eight nodes of Ω_B defining the host cell and their determination requires the solution of a 3×3 non-linear system. The resulting system is solved using Newton iteration, whose convergence rate is quadratic if the initial guess is sufficiently close to the final solution. As shown in [24], very rapid convergence rates can be achieved when the exact solution for the curvilinear coordinates of p under the assumption of a locally uniform Cartesian grid is used to specify the initial guess for the actual curvilinear domain.

The implementation of the present domain decomposition algorithm is based on the Schwarz alternating procedure. That is, the governing equations are solved successively in the various subdomains, alternating from one subdomain to another in a given order. For the solution in a given subdomain, boundary conditions at grid interfaces maybe specified using either MFBI or SI. Moreover, multigrid acceleration is implemented such that the communication between different subdomains takes place only on the finest mesh. This approach has also been used in the past by Hinatsu and Ferziger [6] and Zang and Street [17]. The overall solution procedure for advancing the flowfield from the n to the $n+1$ time levels can be summarized as follows:

1. Specify an initial guess for the flowfield at the $n+1$ level in all subdomains Ω_j ($j = 1$ to j_{max}, where j_{max} is the total number of subdomains) – the solution at the n level can be used for this purpose. Note that this initial guess is also the solution at the $m = 1$ pseudo-time level.
2. For every subdomain Ω_j ($j = 1$ to j_{max}), interpolate the Cartesian velocity components and the pressure on all near-boundary grid nodes of Ω_j from the surrounding subdomains and calculate the interpolated terms in the right-hand side of Eqs. (17)–(19). Calculate the right-hand side of Eq. (17) and invert the system of Eqs. (17) and (18) to obtain the three Cartesian velocity components on the interface at the $m+1$ pseudo-iteration.
3. For every subdomain Ω_j ($j = 1$ to j_{max}), solve Eq. (6) to advance the solution in pseudo-time (from m to $m+1$) at all interior nodes of Ω_j using the multigrid, Runge–Kutta algorithm described in Section 2.
4. Check if convergence in pseudo-time has been achieved in every subdomain.

If yes, the solution in the entire flowfield has been advanced to the $n+1$ level. Return to Step 1 and continue to the next physical time step after replacing the solution at level n with the calculated solution at $n+1$.

If no, update the solution at the $n+1$ level using the computed $m+1$ estimate, return to Step 2 and continue iterating in pseudo-time over all subdomains until convergence is reached.

6. MFBI vs. SI: numerical experiments

In this section, we carry out a series of numerical experiments to evaluate the accuracy and efficiency of the MFBI and assess its performance relative to the standard trilinear interpolation approach (i.e., the specification of all four primitive variables at interfaces via trilinear interpolation). The test cases we consider include steady and oscillatory flow in a lid-driven cavity, and steady flow in a strongly curved, 90° pipe bend. These numerical experiments are designed to investigate the effect of the interface treatment

algorithm on the efficiency of the iterative algorithm and the smoothness and accuracy of the computed solutions (especially when the grid spacing across interfaces is discontinuous).

6.1. Steady lid-driven cavity flow

The Reynolds number for this flow is based on the lid velocity (U) and the cavity height (H). Calculations are carried out for $Re = 400$ and 1000. The flow domain is discretized using two overset grids as shown in Fig. 3. The background cavity domain Ω_A has a hole at its center, which is covered by a $(0.55H)^3$ cube, Ω_B, arbitrarily oriented relative to Ω_A. Both subdomains are discretized with uniformly spaced grids of spacing Δx_A and Δx_B, respectively. For the $Re = 400$ case we investigate the effect of grid spacing continuity across the overset-grid interface – quantified by the ratio $r^x = \Delta x_A/\Delta x_B$ – by considering the following two grid densities: Grid 1, with 33^3 grid nodes in Ω_A, 21^3 grid nodes in Ω_B, and $r^x = r^y = r^z \sim 1.14$ (nearly continuous grids); and Grid 2, with 33^3 grid nodes in Ω_A, 61^3 grid nodes in Ω_B, and $r^x = r^y = r^z \sim 3.4$ (discontinuous grids). For the $Re = 1000$ case we report results for the following grid arrangement: 69^3 grid nodes in Ω_A, 41^3 grid nodes in Ω_B.

We carry out calculations on the two composite domains, Grids 1 and 2, using both the SI and MFBI approaches and compare the computed solutions in Ω_B with the solution obtained on a single zone grid – i.e., using only the background grid Ω_A. For all cases the iterative algorithm is declared converged when all residuals reach machine zero. We should also note that all subsequently presented results (both on the single and composite domains) have been obtained by setting $\varepsilon = 0$ (see Eq. (5)) in order to evaluate the dispersive properties of each interpolation scheme free of artificial dissipation effects.

Fig. 4 compares the calculated steady-state flowfields obtained on Grid 1 in terms of pressure and velocity contours ($Re = 400$). Both interpolation schemes yield identical velocity fields, which are in excellent agreement with the single zone grid solution. Some differences are observed, however, in the computed pressure fields. The MFBI scheme yields nearly smooth pressure contours that are in good agreement with the single-zone grid. The SI scheme, on the other hand, produces weak pressure oscillations throughout the Ω_B subdomain. It should be noted that to some extent the observed pressure oscillations for both schemes are due to the well-known, odd–even decoupling of the pressure nodes, which occurs when the governing equations are discretized on a non-staggered grid with central differencing and no dissipation is introduced

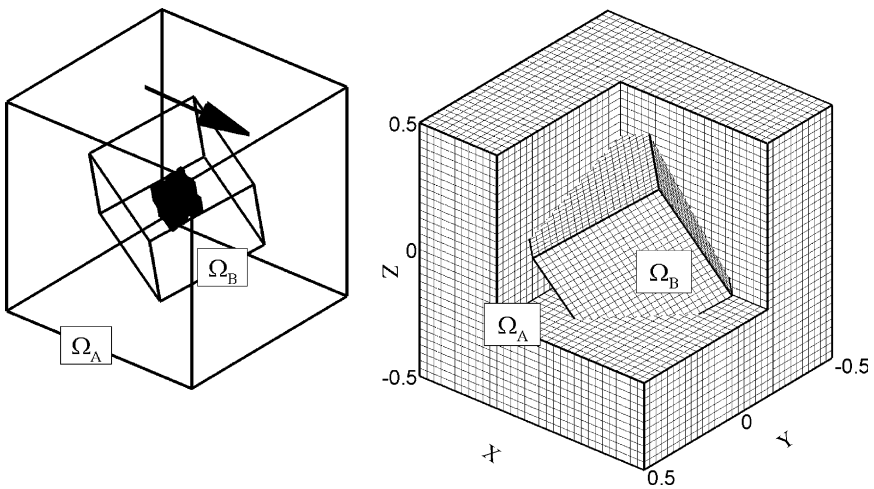


Fig. 3. Overset-grid layout for lid-driven cavity flow.

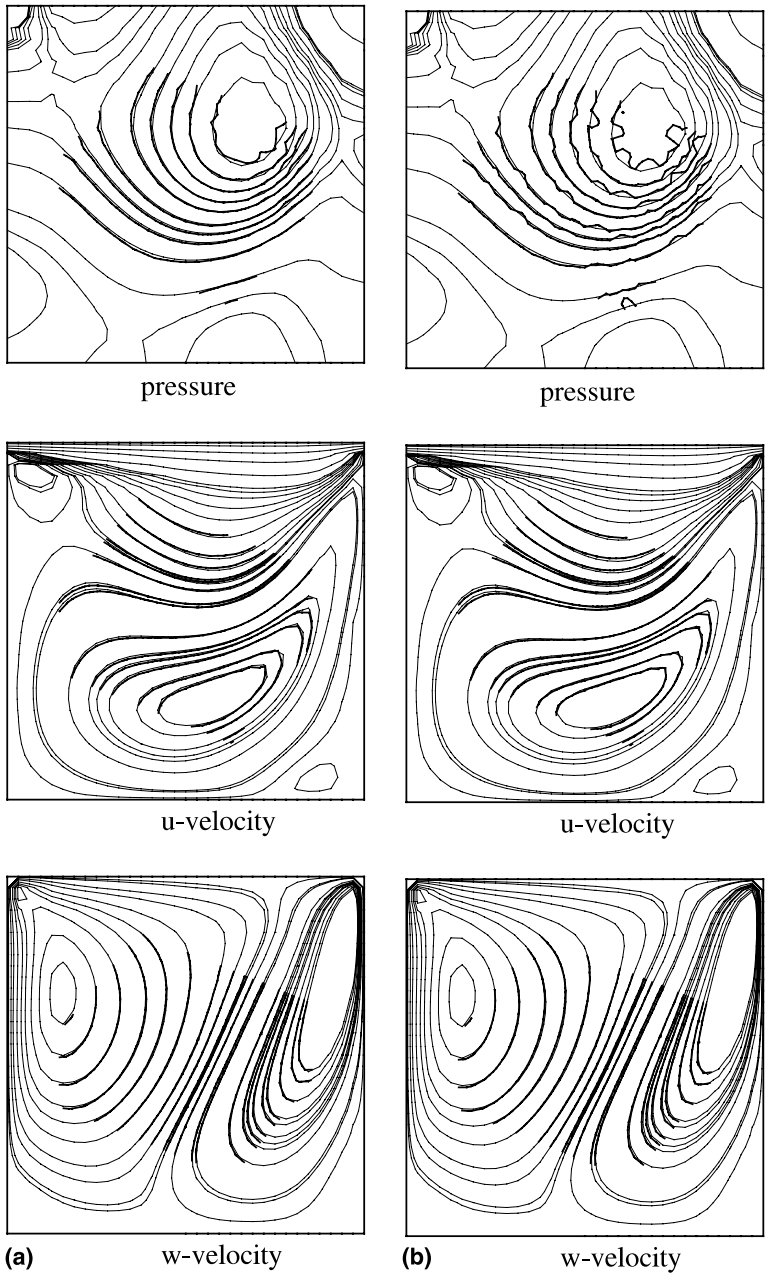


Fig. 4. Steady flow in a lid-driven cavity ($Re = 400$) for continuous overset grids. Contours of pressure and velocity contours (bottom) at the $y = 0$ symmetry plane with MFBI (a) and SI (b). Thin line: Single domain solution; thick line: overset-grid solution in Ω_B (see Fig. 3).

in the discrete continuity equation [28] – weak decoupling of the pressure nodes is also evident in the solution obtained on the single-zone grid. Since, however, all numerical solutions have been obtained with $\varepsilon = 0$ and on identical grids, the pressure comparisons shown in Fig. 4 clearly suggest that the SI approach

tends to exacerbate the decoupling of the pressure nodes even when the grid spacing ratio r is close to one. This conclusion is further reinforced by the results shown in Fig. 5, which compares solutions obtained on the discontinuous Grid 2 arrangement ($Re = 400$). The discontinuity in the grid spacings appears to have

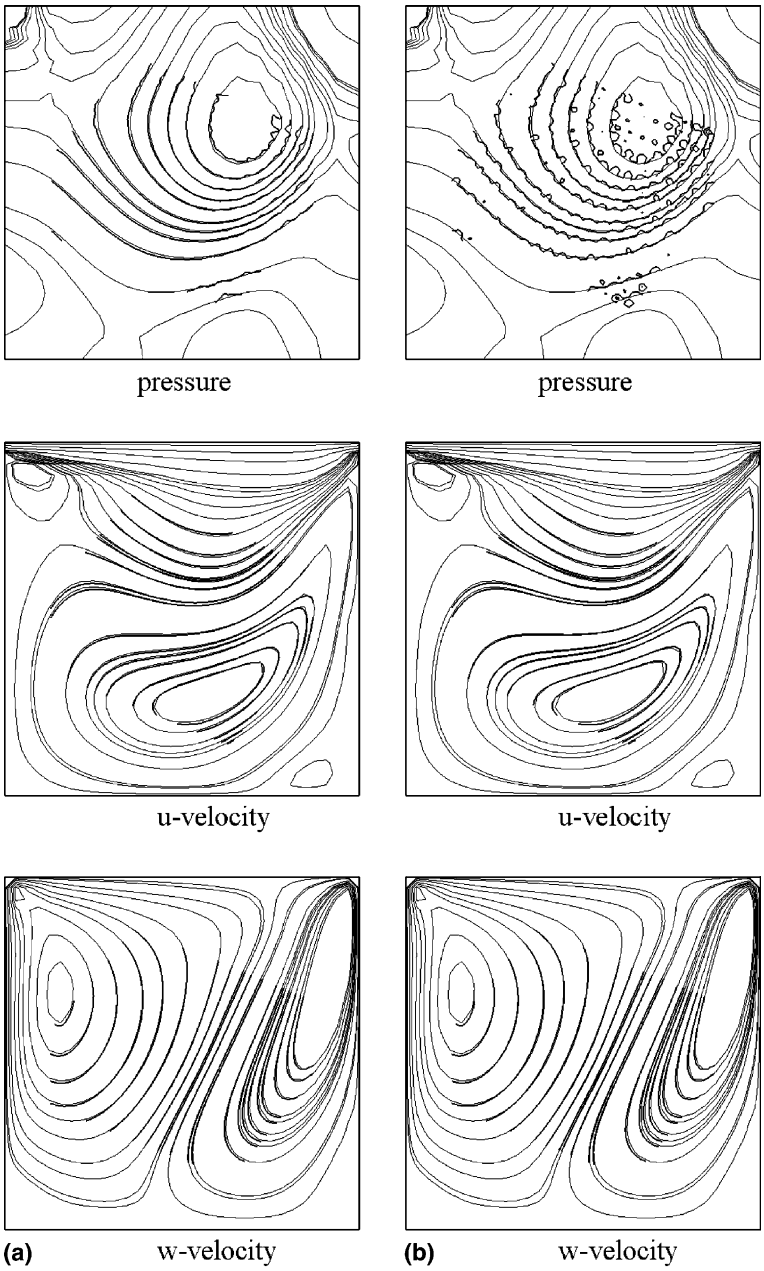


Fig. 5. Steady flow in a lid-driven cavity ($Re = 400$) for discontinuous overset grids. Contours of pressure and velocity contours (bottom) at the $y = 0$ symmetry plane with MFBI (a) and SI (b). Thin line: Single domain solution; thick line: overset-grid solution in Ω_B (see Fig. 3).

little effect on the solution obtained with MFBI while the pressure field obtained with SI is contaminated by spurious oscillations. For both algorithms, however, the calculated velocity fields are identical and in good agreement with the single-zone grid solution.

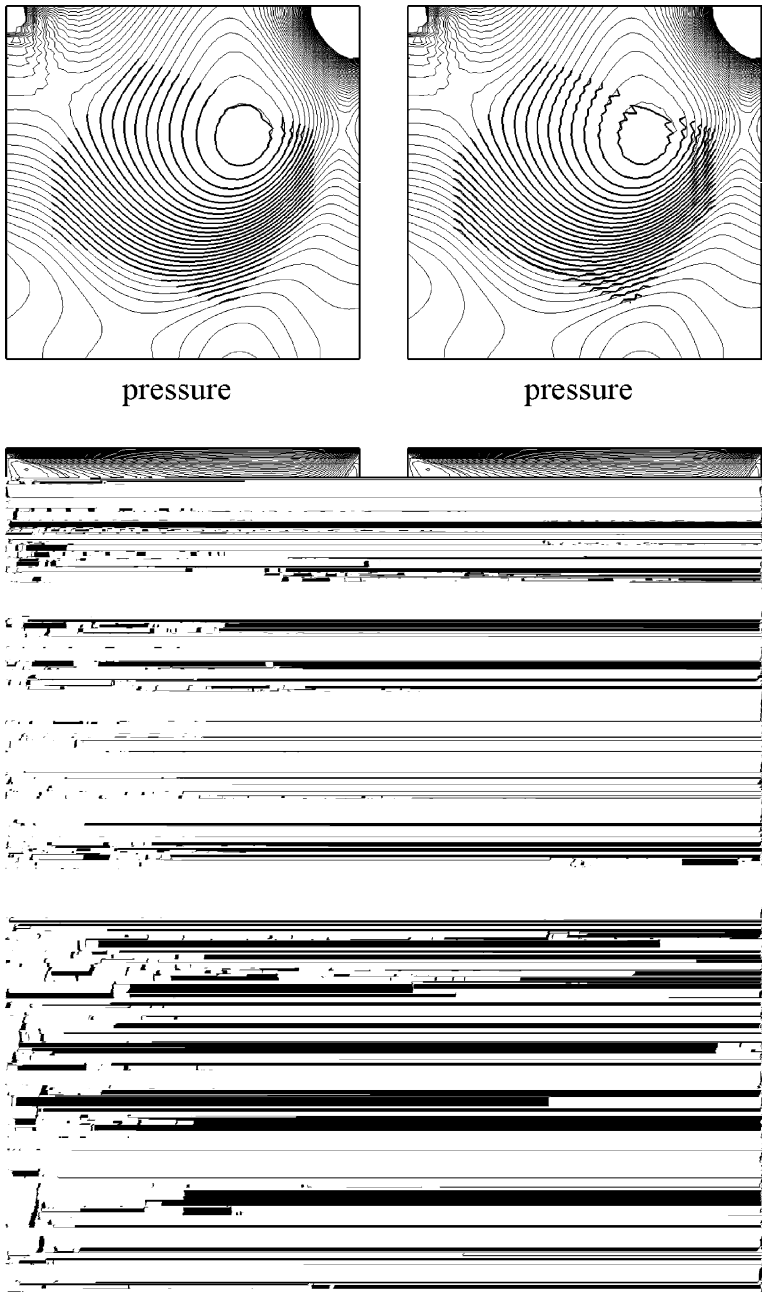


Fig. 6. Steady flow in a lid-driven cavity ($Re = 1000$) for continuous overset grids. Contours of pressure and velocity contours (bottom) at the $y = 0$ symmetry plane with MFBI (a) and SI (b). Thin line: Single domain solution; thick line: overset-grid solution in Ω_B (see Fig. 3).

For the $Re = 1000$ case we carried out similar numerical tests with a small amount of artificial dissipation ($\varepsilon = 0.001$) and the results are shown in Fig. 6 – results with $\varepsilon = 0$, not shown here, were also obtained but they lead to the same conclusions as those derived from Figs. 4 and 5. Artificial dissipation tends to suppress the odd–even decoupling of the pressure nodes for all algorithms but does not change the central conclusion suggested by the results shown in Figs. 4 and 5. As seen in Fig. 6, the results obtained with the SI algorithm continue to exhibit considerable high-frequency oscillations. The addition of artificial dissipation, however, tends to confine the oscillations in the SI pressure field to the overlap region of the two subdomains (see Fig. 6).

6.2. Oscillatory cavity flow

The flow domain is identical to that discussed in Section 6.1 but in this case the lid velocity oscillates in time ($U = \cos(2\pi t)$). Here we compare the MFBI and SI algorithms for an unsteady flow using the overset-grid layout shown in Fig. 7. The Ω_A and Ω_B subdomains are discretized with $25 \times 25 \times 25$ and $17 \times 17 \times 9$ grid nodes, respectively. The grid nodes in each subdomain are distributed uniformly along all three spatial directions such that the resulting grid spacing ratios for the two grids are discontinuous: $r^x = r^y = 2$ and $r^z = 4$. For both interpolation algorithms, calculations are carried out for $Re = 100$ with $\varepsilon = 0$ and $\Delta t = 0.05$.

Figs. 8 and 9 compare the computed solutions at time $t = 0.25$ and 10. Even for this very discontinuous grid arrangement, the MFBI algorithm gives pressure and velocity fields that are nearly continuous in the overlap region – only very weak oscillations are observed in the calculated pressure field. On the other hand, the solutions obtained with the SI algorithm exhibit very pronounced spurious oscillations in the overlap region either in the computed pressure or velocity fields. During the early transient stages of the simulation, see Fig. 8, the calculated pressure exhibits strong oscillations while the

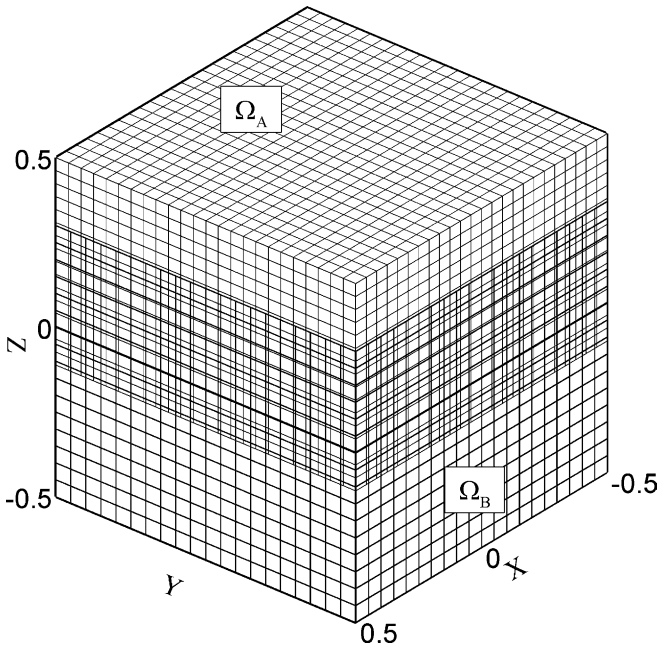


Fig. 7. Overset-grid layout for oscillating cavity flow.

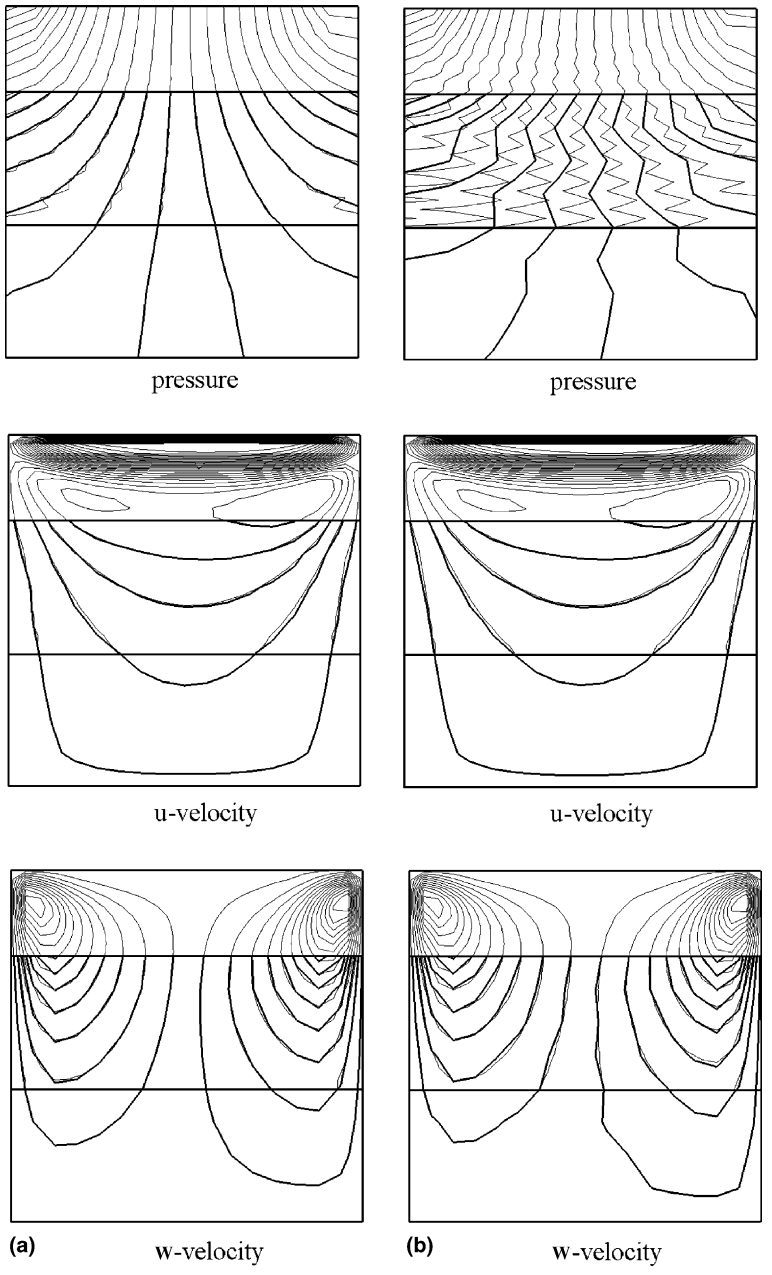


Fig. 8. Oscillating cavity flow ($Re = 100$). Instantaneous ($t = 0.25$) pressure and velocity contours at the $y = 0$ symmetry plane. Thin line: solution in Ω_A; thick line: solution in Ω_B. (a) MFBI; (b) SI.

velocity field is smooth. At later times, however (see Fig. 9) the pressure oscillations diminish considerably, but spurious wiggles grow in the vertical velocity component. Fig. 10 compares the time history of drag coefficient on the moving wall. It is interesting to note that in spite of the previously discussed global differences between the two solutions, the time histories of the drag coefficient are indistinguishable for both algorithms.

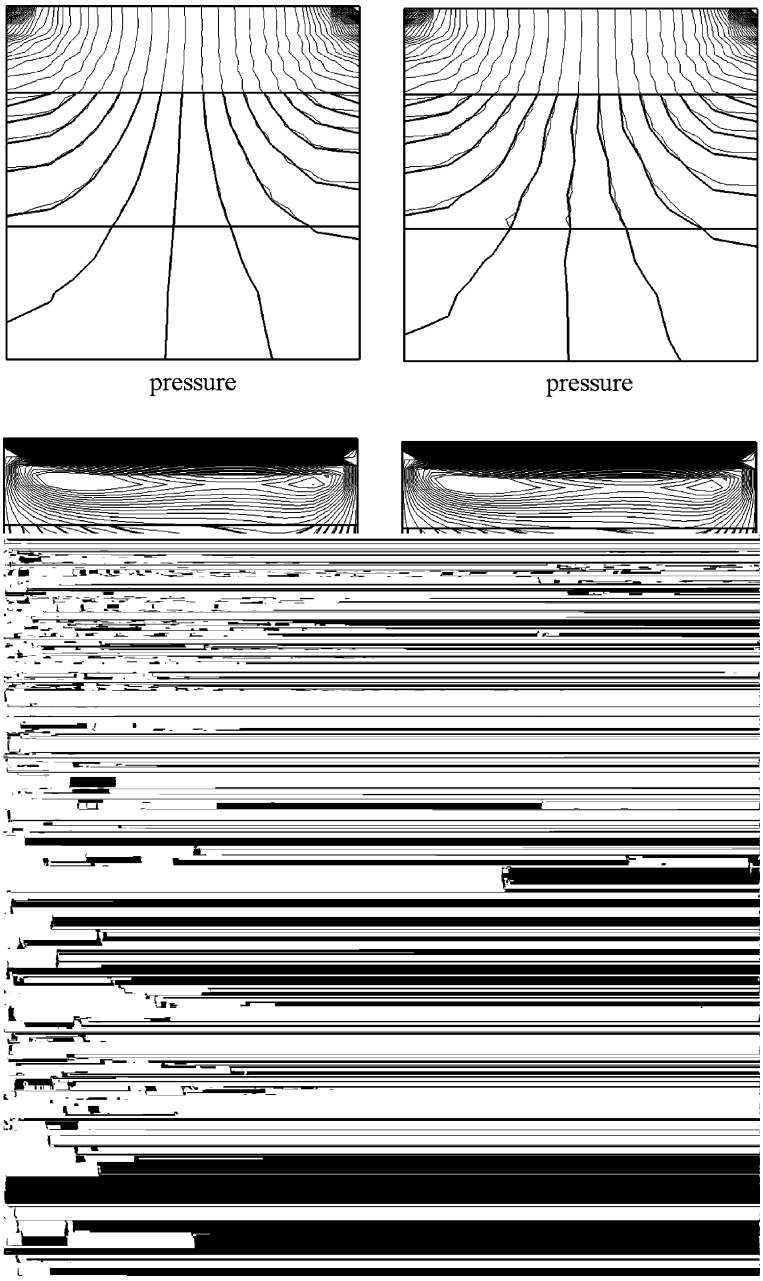


Fig. 9. Oscillating cavity flow ($Re = 100$). Instantaneous ($t = 10$) pressure and velocity contours at the $y = 0$ symmetry plane. Thin line: solution in Ω_A; thick line: solution in Ω_B. (a) MFBI; (b) SI.

To assess the relative efficiency of the two algorithms, we plot in Fig. 11 their respective convergence histories for the first three physical time steps. Note that during every physical time step, 2500 pseudo-time steps were calculated in order to explore whether the residuals reach machine zero and ensure that no long-time numerical instabilities develop. The convergence rate of the MFBI algorithm appears to

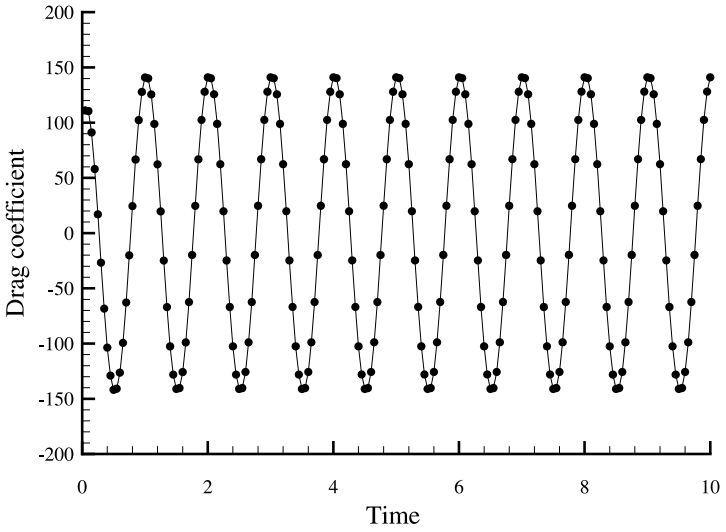


Fig. 10. Oscillating cavity flow ($Re = 100$). Time history for the drag coefficient at the moving lid for Case 2. Solid line: MFBI; points: SI.

be consistent for all simulated time steps as all residuals reach machine zero within 200–300 pseudo-time steps. The SI algorithm, on the other hand, exhibits more erratic convergence histories. In the first six time steps (only three time steps are shown in Fig. 11), the pressure residual (with only exception the second time step) failed to reach machine zero even after 2500 iterations. This trend is consistent with the strong spurious oscillations that develop in the pressure field during the early stages of the calculation (see Fig. 8 above). At later times, however, the convergence of the SI algorithm improves considerably and the pressure residuals decay to machine zero at rates similar to those for the MFBI algorithm.

6.3. Steady flow in a 90° pipe bend

To evaluate the performance of the two interpolation algorithms in a 3D flow with inflow and outflow, we calculated steady flow in a strongly curved 90° pipe bend. This test case was studied experimentally by Bovendeerd et al. [29] who carried out velocity measurements at Reynolds number $Re = 700$ (based on the bulk velocity U and the pipe diameter D). The radius of curvature of the bend is $3D$ and an upstream straight tangent $50D$ long was used in the experiment to ensure fully developed flow at the entrance of the bend.

In the present study, the calculations start $5D$ upstream of the bend where a fully developed (parabolic) velocity profile is specified. The exit boundary is located $7D$ downstream of the end of curvature. We employed the overset-grid layout shown in Fig. 12. The cross section of the bend was discretized using two overlapping subdomains: the global polar grid and the interior, uniformly spaced, Cartesian grid. To avoid numerical difficulties with the singularity on the axis of the bend, the global grid had a hole around the axis as depicted in Fig. 12. For both subdomains the streamwise planes are located at exactly the same locations. Note that we specifically tailored the grid layout shown in Fig. 12 to ensure that the ratios of the grid spacings of the Cartesian subdomain to those of the polar subdomain vary considerably within the overlap region, from close to unity near the edges of the Cartesian box to approximately 7 in the vicinity of the axis. Thus, this test case will allow us to explore the effect of heterogeneously distributed grid spacing ratios in a complex geometrical configuration.

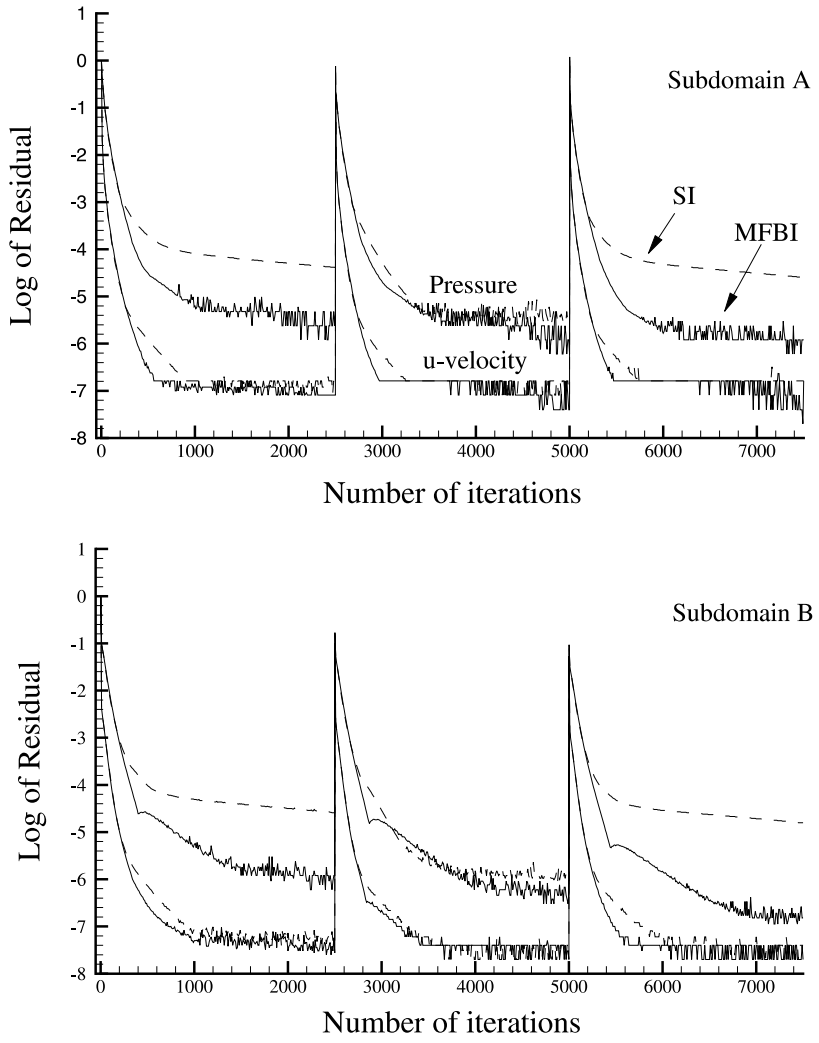


Fig. 11. Oscillating cavity flow ($Re = 100$). Convergence histories for the pressure and U-velocity residuals for three physical time steps in Ω_A (top) and Ω_B (bottom). Solid line: MFBI; dashed line: SI.

To investigate the sensitivity of the calculated flowfields to grid resolution, we carried out calculations on two grids. The coarse mesh consists of $61 \times 17 \times 49$ (streamwise, radial, azimuthal) and $61 \times 13 \times 13$ (streamwise and transverse directions) for the polar and Cartesian subdomains, respectively (a total of 61,112 grid nodes). The respective grid densities for the fine mesh are $85 \times 25 \times 53$ and $85 \times 21 \times 21$ (a total of 150,110 grid nodes).

The calculated streamwise velocity profiles at the plane of symmetry are compared with the experimental measurements [29] in Fig. 13. This figure shows results obtained with the MFBI and SI algorithms on the coarse and fine meshes. As seen, both algorithms yield very similar velocity profiles that are in excellent agreement with the measurements. A notable difference between the two algorithms is that the MFBI appears to be less sensitive to grid refinement. Notice that the coarse and fine mesh velocity profiles in Fig. 13(a) are identical, thus, suggesting that for the MFBI the coarse mesh is sufficient for obtaining grid independent solutions. For the SI, on the other hand, discrepancies between the coarse and fine mesh

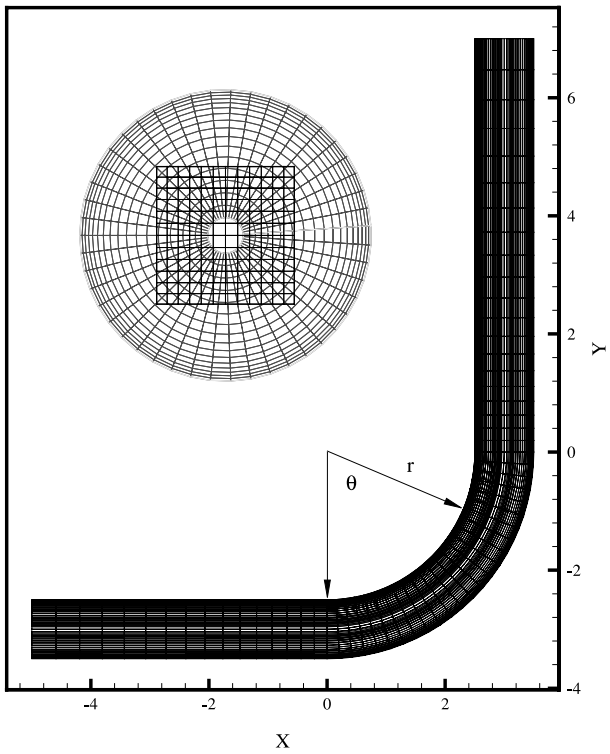


Fig. 12. Overset-grid layout for flow in 90° pipe bend.

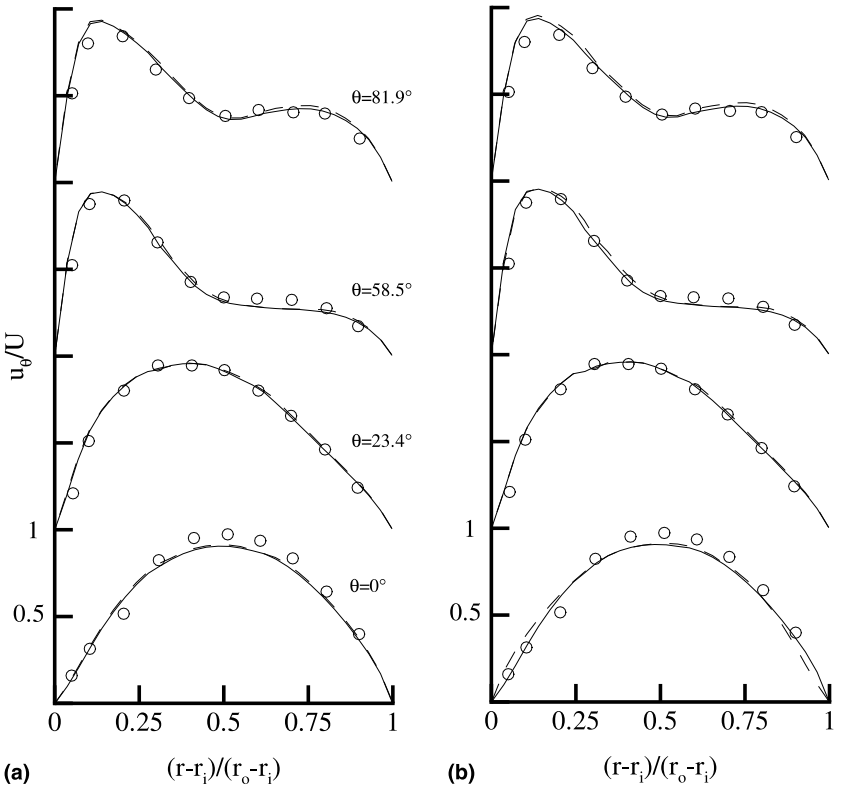
solutions are observed especially near the entrance of the bend. As the mesh is refined, however, the SI solution converges to the solution obtained with the MFBI.

Fig. 14 shows pressure and streamwise velocity contours at the $\theta = 45°$ cross-section within the bend. Only results obtained on the coarse mesh are shown in this figure but similar trends are observed on the fine mesh as well. In spite of the considerable heterogeneity in the grid spacing ratios for this case, the MFBI yields nearly continuous pressure field and velocity contours that are smooth and nearly identical in the overlap region of the two overset grids. The results obtained by the SI approach, on the other hand, are strikingly different with pronounced non-physical oscillations dominating the Cartesian subdomain solution. The velocity contours for this case are also smooth but the computed solutions in the overlap regions are not as well matched as those obtained with the MFBI method (see Fig. 14(a)).

The convergence histories for the two algorithms on the fine mesh are shown in Fig. 15. It is evident that the MFBI converges considerably faster than the SI algorithm. Furthermore, it is important to note that for the MFBI algorithm all residuals in both subdomains reach machine zero after 4000 time steps. The SI algorithm, on the other hand, fails to reach machine zero even after 6000 iterations. In fact, we can estimate from the asymptotic convergence slopes (assuming that this rate of convergence will continue until machine zero is reached) that the SI algorithm may require well over 10,000 iterations before all residuals reach machine zero.

7. Applications to complex unsteady laminar flows

In this section we apply domain decomposition in conjunction with MFBI to calculate massively separated, laminar flows in complex geometrical configurations. We demonstrate that overset grids can



successfully simulate unsteady vortical flows with complex dynamics in multiconnected, 3D geometries. The first case we simulated is vortex shedding from a circular cylinder mounted between two endplates. The calculations are validated by comparing them with available experimental data and observations and numerical results that were previously reported in the literature. The second test case is the flow in a straight rectangular channel with two wall-mounted obstacles: a slender rectangular block attached to the side of the channel and a bottom mounted circular cylinder adjacent to it. Even though no experimental data or previous numerical results are available for this case, the computed flowfields demonstrate further the ability of MFBI to simulate flows with very complex wake dynamics using multiple overset grids.

7.1. Vortex shedding from a circular cylinder mounted between two endplates

The flow past a circular cylinder is an important engineering problem and has been widely studied both experimentally [30–34] and computationally [21,35,36]. In spite of a considerable body of work on this flow, however, a number of fundamental issues about its complex physics are still the subject of interpretation and controversy [32]. Perhaps the most perplexing such issue is the effect that solid endplates, typically used to mount the cylinder in laboratory studies, have on the dynamics of vortex shedding. For low aspect ratios (L/D where L is the cylinder length and D its diameter) endplate effects are known to give rise to very complex 3D phenomena and be the source for the considerable scatter in the available experimental data

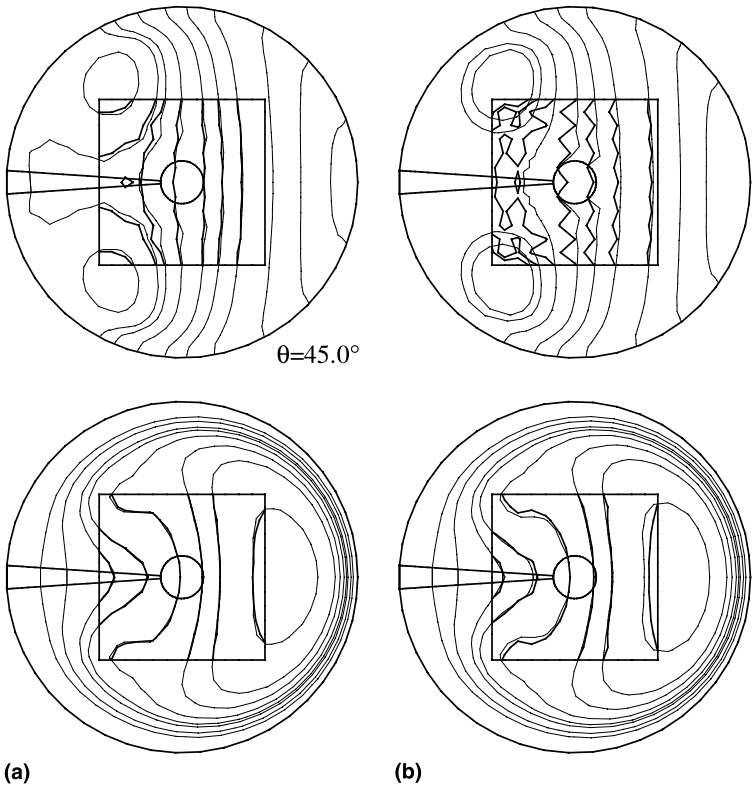
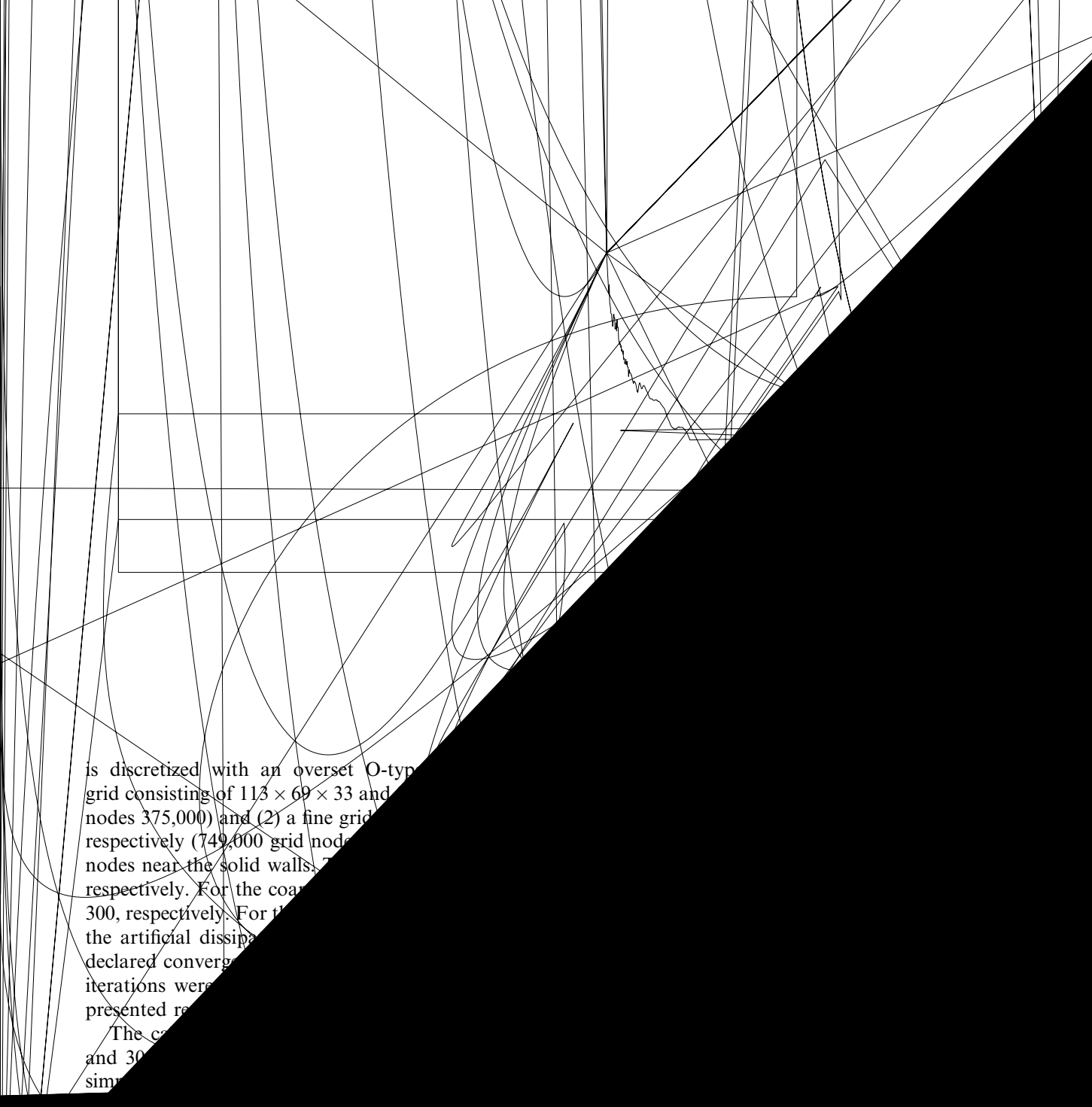


Fig. 14. Pipe bend flow. Calculated pressure (top) and streamwise velocity (bottom) contours at the cross-section halfway through the bend. Thin lines: solution in Cartesian domain; thick lines: solution in polar domain. (a) MFBI; (b) SI.

[31–34]. In a recent computational study, Mittal [21] investigated the effect of endplates on the dynamics of vortex shedding. He carried out direct numerical simulations for flow past a cylinder with $L/D = 16$ at Reynolds numbers ranging from 100 to 1000. We applied the MFBI method to simulate the flow in a geometrical configuration identical to that studied by Mittal so that our computations can be readily compared with his results. Note that Mittal [21] employed a finite-element, Galerkin formulation on a single-block unstructured mesh with approximately 900,000 nodes and with time increments in the range of 0.05–0.25.

The computational domain is identical to that used in [21] and is depicted in Fig. 16. The cylinder is $8D$ long and symmetry boundary conditions are applied at $z = 8D$ to simulate the flow in a domain with aspect ratio $L/D = 16$. The solid endplate, on which no-lip conditions are specified, is located at $z = 0$ and extends from $4.5D$ upstream of the cylinder ($x = -4.5D$) all the way to end of the computational domain ($x = 22.5D$). At the entrance of the computational domain ($x = -12.5D$) a uniform velocity profile is specified. At the top and lateral boundaries, symmetry boundary conditions are specified while at the exit boundary extrapolation is used for the velocity field. At all six planes of the computational domain and the surface of the cylinder the pressure is calculated via linear extrapolation from the interior nodes. We computed two Reynolds numbers, $Re = 100$ and 300. In all subsequently presented results, time, length, velocity, and pressure are non-dimensionalized by D/U, D, U, and ρU^2, respectively.

The overset-grid arrangement we employed is shown in Fig. 16. A global Cartesian mesh (H-type grid) with a hole in the vicinity of the cylinder is used to cover the entire computational domain. The hole region



is discretized with an overset O-typ
grid consisting of 113 × 69 × 33 and
nodes 375,000) and (2) a fine grid
respectively (749,000 grid nod
nodes near the solid walls.
respectively. For the coa
300, respectively. For t
the artificial dissipa
declared converg
iterations were
presented re

The ca
and 30
sim

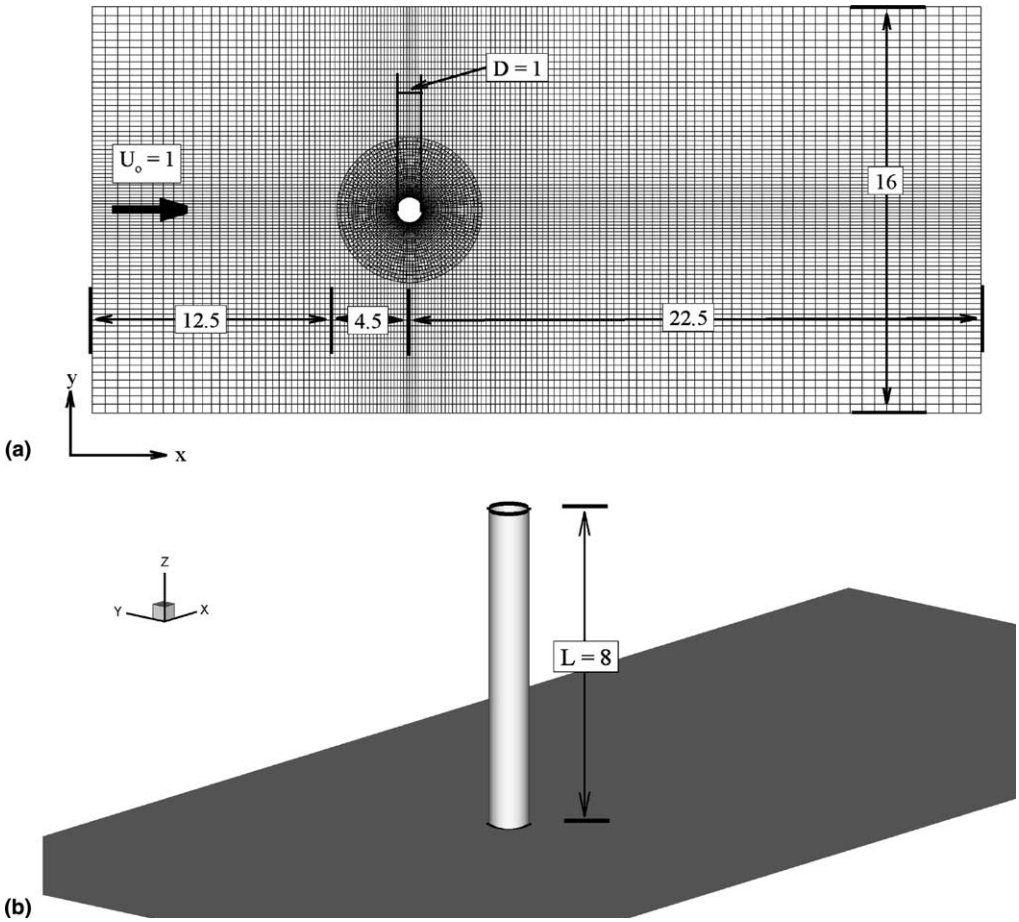
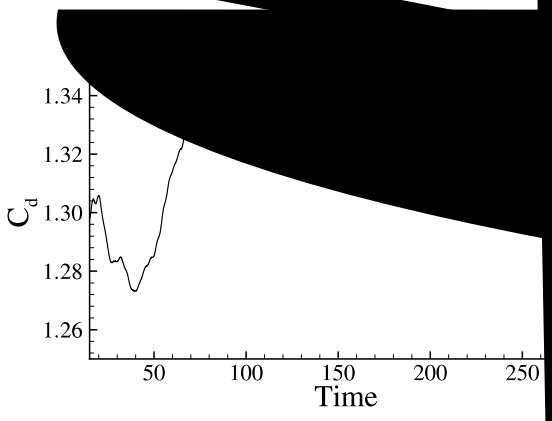


Fig. 16. Wall-mounted cylinder flow. Computational domain and overset-grid layout.

vs. 0.047 [21]. This discrepancy, however, could be due to the different time increments used in the two computations.

The complexity of the flow increases considerably for $Re = 300$ as revealed by the time histories and spectra of lift and drag shown in Fig. 18. Experimental observations [32] reveal that for cylinders of large aspect ratio, at $Re = 180$ the wake undergoes a transition from a simple periodic state to a far more complex and seemingly disorganized state. The first instability is the so-called Mode A shedding, which is characterized by the growth of coherent vortices whose spanwise wave length is approximately equal to 3–4 cylinder diameters and a single, low frequency peak in the lift spectra. As the Reynolds number increases the low frequency peak diminishes and a second peak appears at higher frequencies, which is associated with the so-called Mode-B shedding. This new state is characterized by structures of smaller spanwise extent, typically of the order of one cylinder diameter and an overall more disorganized wake structure. The transition from Mode A to Mode B shedding is also characterized by vortex dislocations, which form as shed vortices go out of phase leading to the growth of large, spot-like, vorticity patches [32]. Mittal's [21] calculations suggest that for $L/D = 16$ and $Re = 300$ the wake is characterized by the intermittent interplay of Modes A and B and the existence of two frequencies in the lift spectra – a high frequency, presumably associated with Mode B shedding, ridding on a low frequency modulation due to Mode A shedding. Our



computed time histories for the lift and drag coefficien
[21] and suggest complex wake dynamics. The drag coe
equal to 1.2, which is very close to the 1.19 value repo
histories reveal the presence of two frequencies, whic
competition of Modes A and B shedding in the wake (s
on the dominant lift frequency is $St_l = 0.195$, which is wi
[21]. Note that Norberg's [34] experiments for the same
($L/D = 20$ in [34] vs. $L/D = 16$ here) yielded a value of
results.

Fig. 19 shows instantaneous contours of u-, v- and
vorticity components at the vertical plane of symmetry
perimental observations of Williamson [31] and the nu
$Re = 100$ is oblique with a single vortical cell across the
vary significantly along the span due to the effects
shedding angle approximately equal to 25° at
ngle we can estimate from our results s
les A and B shedding for th
shows several snapshots of instantaneous
$t = 210$ exhibit small scale structures that are indicativ
is disorganized with multiple shedding cells located acr

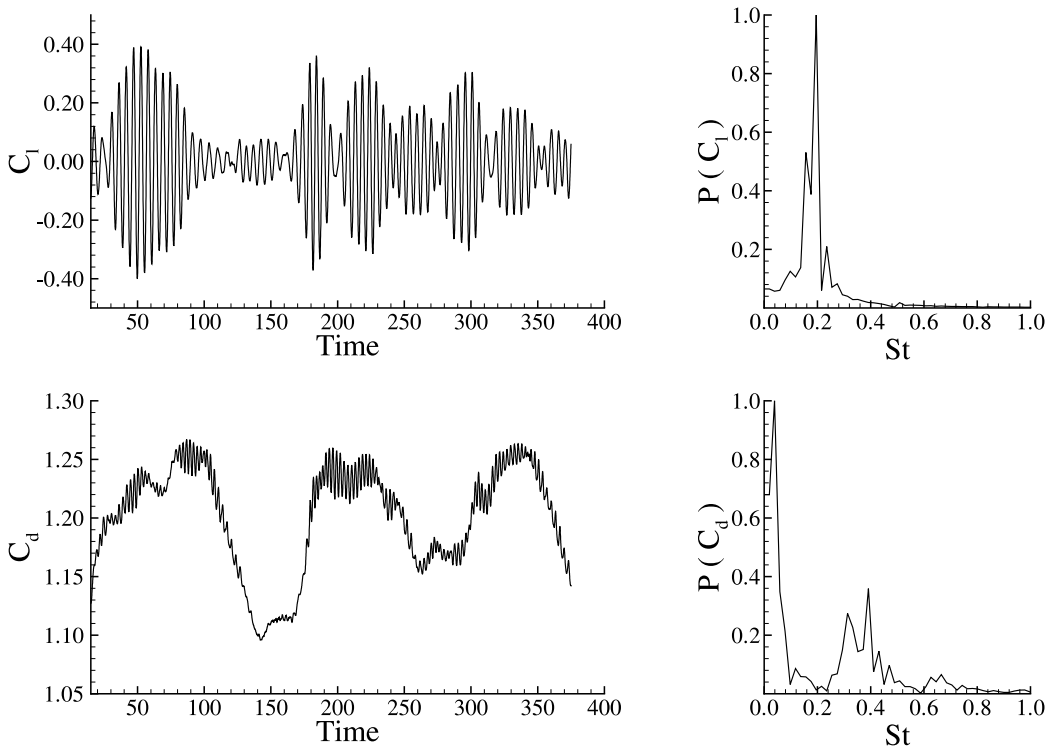


Fig. 18. Wall-mounted cylinder flow ($Re = 300$). Time histories and power spectra of lift (top) and drag (bottom) coefficients.

$t = 210$, there are instants in time during which the wake is characterized by longer wave length cells and exhibits a more organized structure. The contour plot at $t = 300$ is representative of such a state, which would be consistent with Mode A shedding. These results are essentially identical to those reported in [21].

7.2. Flow in channel with wall-mounted obstacles

The final configuration we studied was inspired by the geometry of bridge sections in natural rivers, which typically consist of multiple piers, which support the bridge across the river reach, and abutments, located in the floodplain section of the channel. The computational domain (see Fig. 21(a)) is a rectangular channel containing two bottom mounted obstacles: (1) a circular cylinder mounted on the bottom of the channel and (2) a rectangular block with a semi-circular edge mounted adjacent to the cylinder at the junction between the channel bottom and its side wall. The bottom ($z = 0$) and right side walls ($y = 0$) of the channel are solid walls on which no-slip, no-flux conditions are specified for the velocity field. The left vertical boundary ($y = 10$) and the top boundary ($z = 4$) are treated as symmetry planes. The corner mounted obstacle is a rectangular block of length $L = 2$, height $H = 4$, and width $D = 1$ with half a cylinder attached to it to form a rounded edge. The cylinder (diameter $D = 1$) is attached vertically to the channel bottom at a lateral distance $W = 5$ from the right side wall, as shown in Fig. 21(a). The two obstacles are located across from each other at distance $20D$ from the inlet plane where uniform flow velocity is specified. The outflow plane is located $25D$ downstream of the obstacles. All flow variables at the exit were calculated using linear extrapolation.

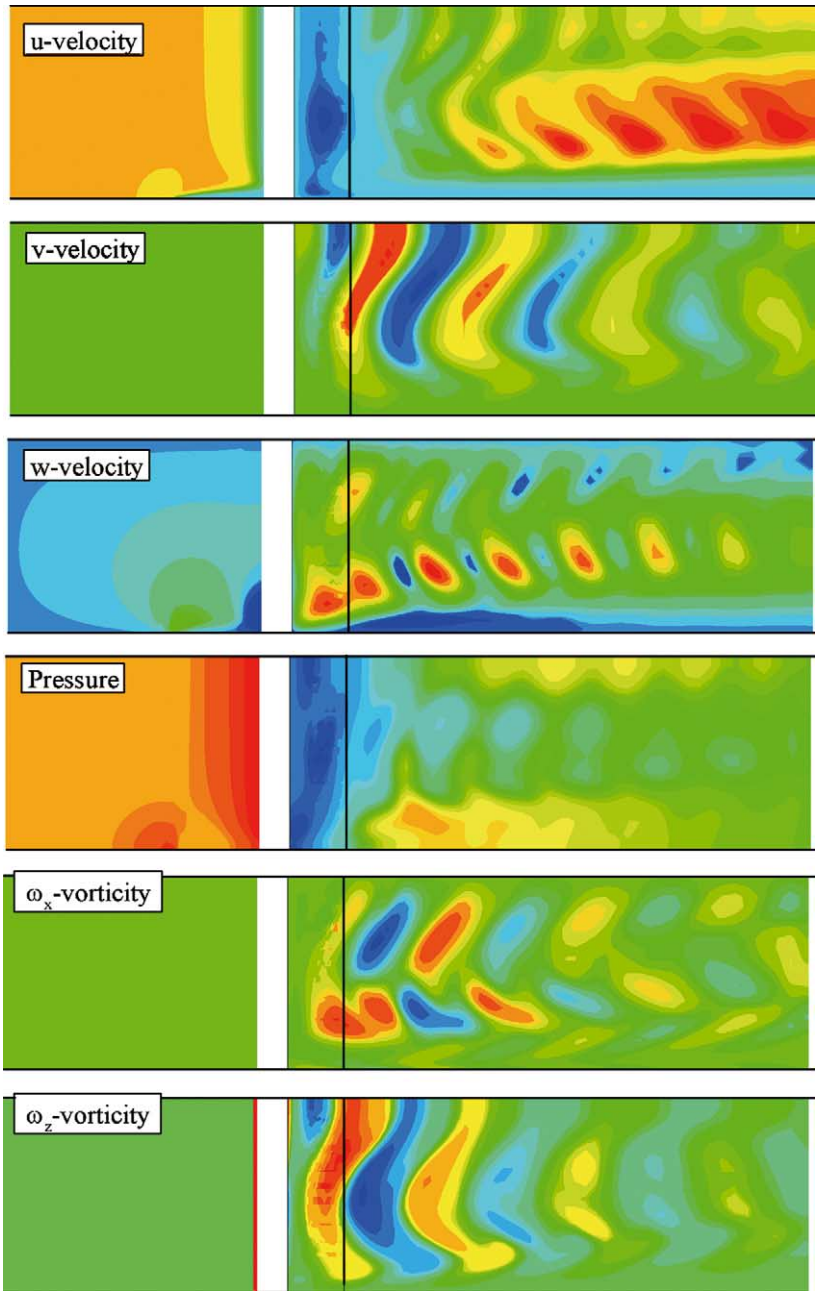


Fig. 19. Wall-mounted cylinder flow ($Re = 100$). Instantaneous contours of velocity components, pressure, and vorticity components at the $y = 0$ plane.

The computational domain is discretized using six overset grids, as shown in Fig. 21(b). The total number of grid nodes for all six subdomains is 336,000. Given the complexity of the flow, this grid resolution is far from adequate for obtaining grid insensitive solutions. However, our objective is to illustrate

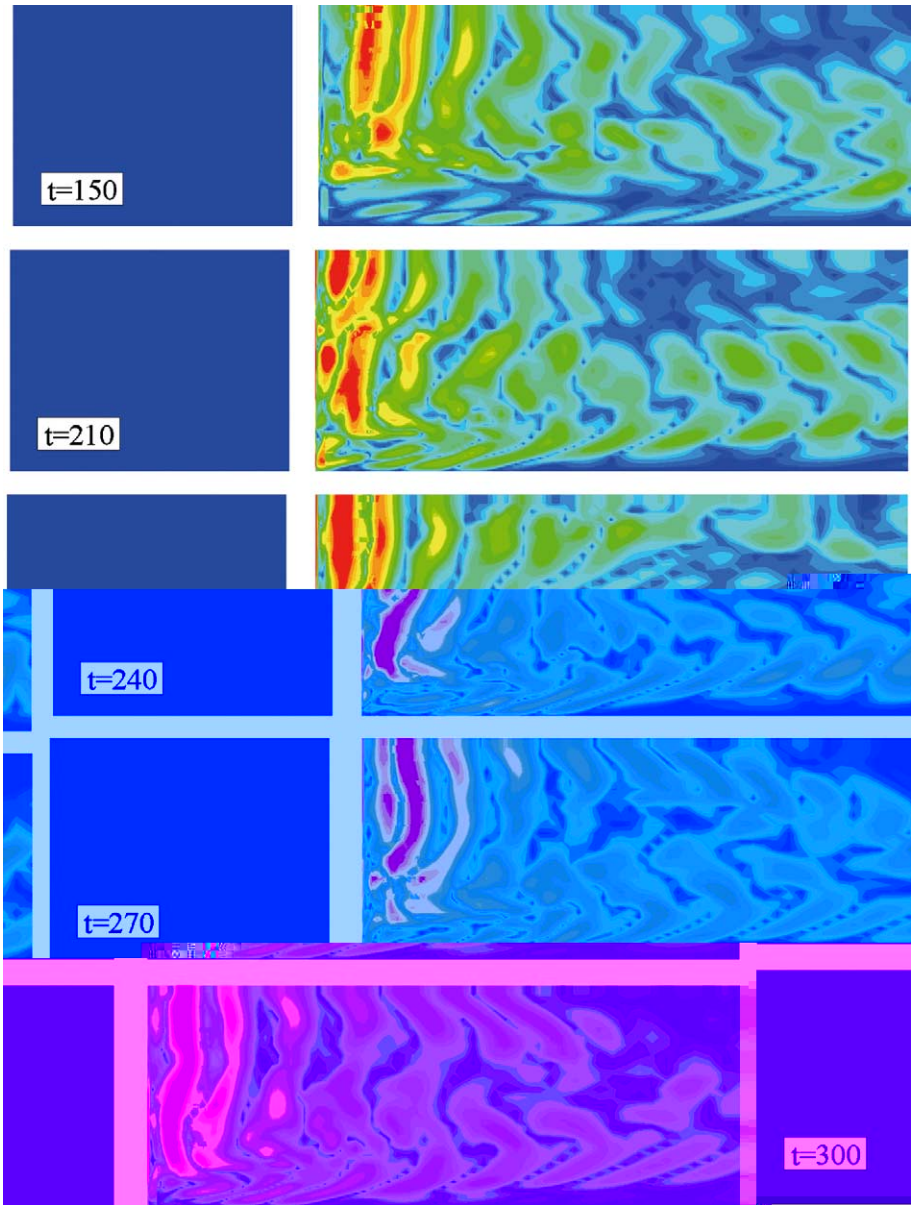


Fig. 20. Wall-mounted cylinder flow ($Re = 300$). Contours of total vorticity at the $y = 0$ plane at five instants in time.

the ability of our domain decomposition method to simulate a complex flow in a complex geometrical configuration. Mesh sensitivity studies and a detailed discussion of the physics of this flow are beyond the scope of this paper and will be reported in a future communication. In the remainder of this section, we present a small sample of our computed results for $Re = 300$. The calculations were carried out with $\Delta t = 0.05$ and $\varepsilon = 0.04$.

Typical time histories of the calculated pressure field at one point in the wake of the two obstacles and its phase portrait are shown in Fig. 22. It is evident that the interaction of the two wakes gives rise to a very

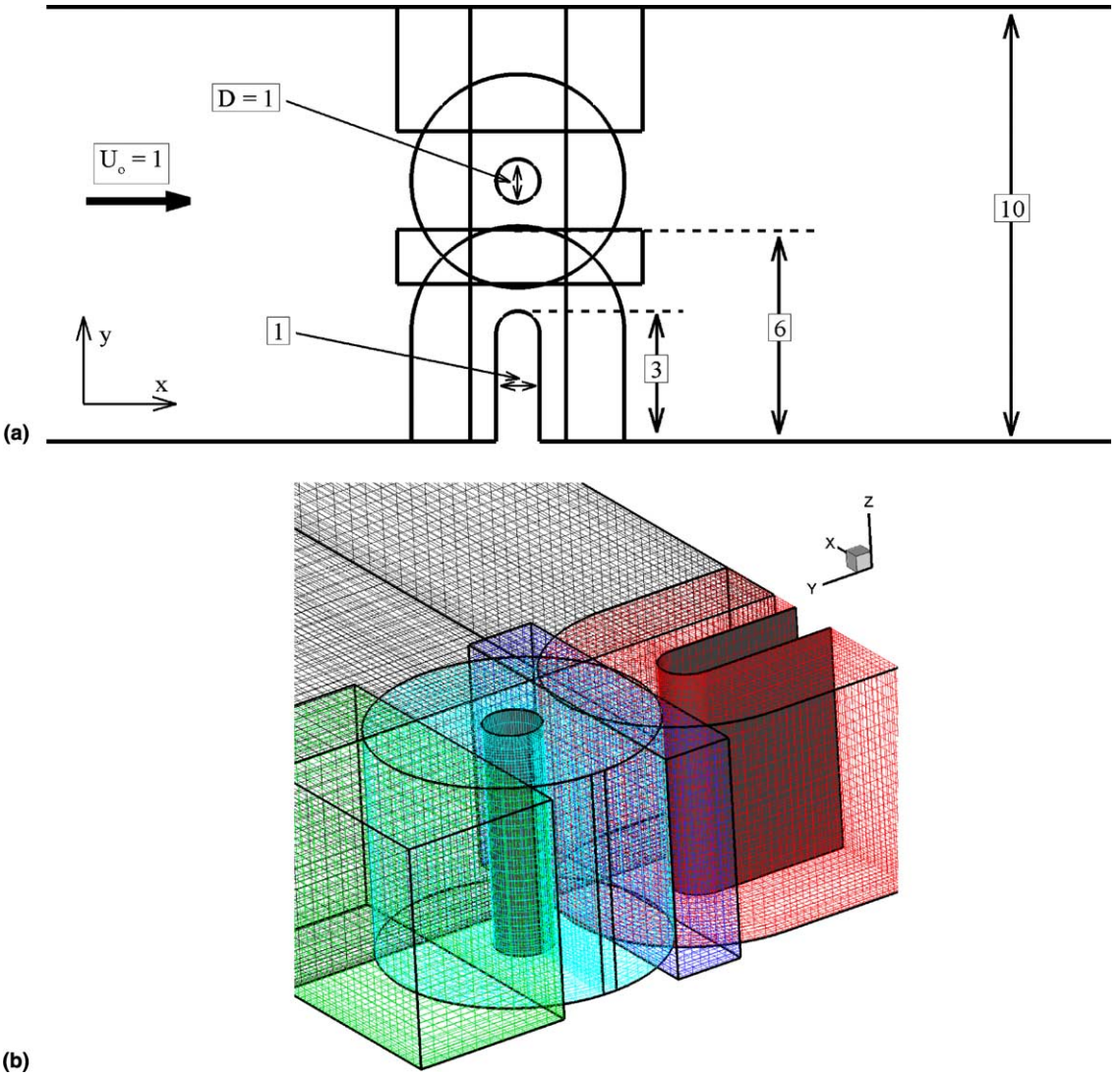
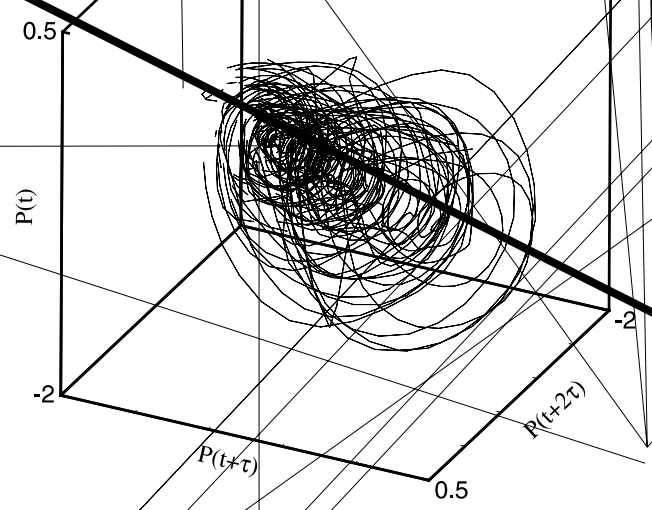


Fig. 21. Channel with two wall-mounted obstacles. (a) Computational domain and (b) overset-grid layout.

complex aperiodic flow. In fact the phase portrait clearly suggests that the dynamics is chaotic – juxtapose the complexity of the dynamics for this case with that shown in Fig. 18 for the single cylinder flow at the same Reynolds number. Contours of instantaneous vorticity contours at the plane of symmetry are shown in Fig. 23. The shear layer emanating from the edge of the block is seen to engage in a very complex interaction with vortices shed from the cylinder. This interaction leads to the break up of the shear layer into small scale eddies almost immediately downstream of the block. Similarly the cylinder wake is characterized by the shedding of smaller scale eddies than those observed for the single cylinder case at the same Reynolds number. The 3D structure of the flow is illustrated in Fig. 24, which visualizes instantaneous coherent structures with the λ_2-method [37]. Large-scale whirlpool-like vortices at the upstream corner of the block, oblique shedding from both the cylinder and the block shear layers, and small scale structures downstream



of the obstacles, emerging due to the complex interaction of the two wakes, dominate the flowfield at all instants in time. Figs. 23 and 24 provide further evidence that with the MFBI approach vortices generated in the various subdomains can cross overset-grid boundaries without distortion and engage in complex, non-linear interactions.

8. Summary and conclusions

We have developed an overset-grid domain decomposition method for simulating unsteady, 3D, incompressible flows in complex geometries. We proposed a mass-flux based, interpolation (MFBI)

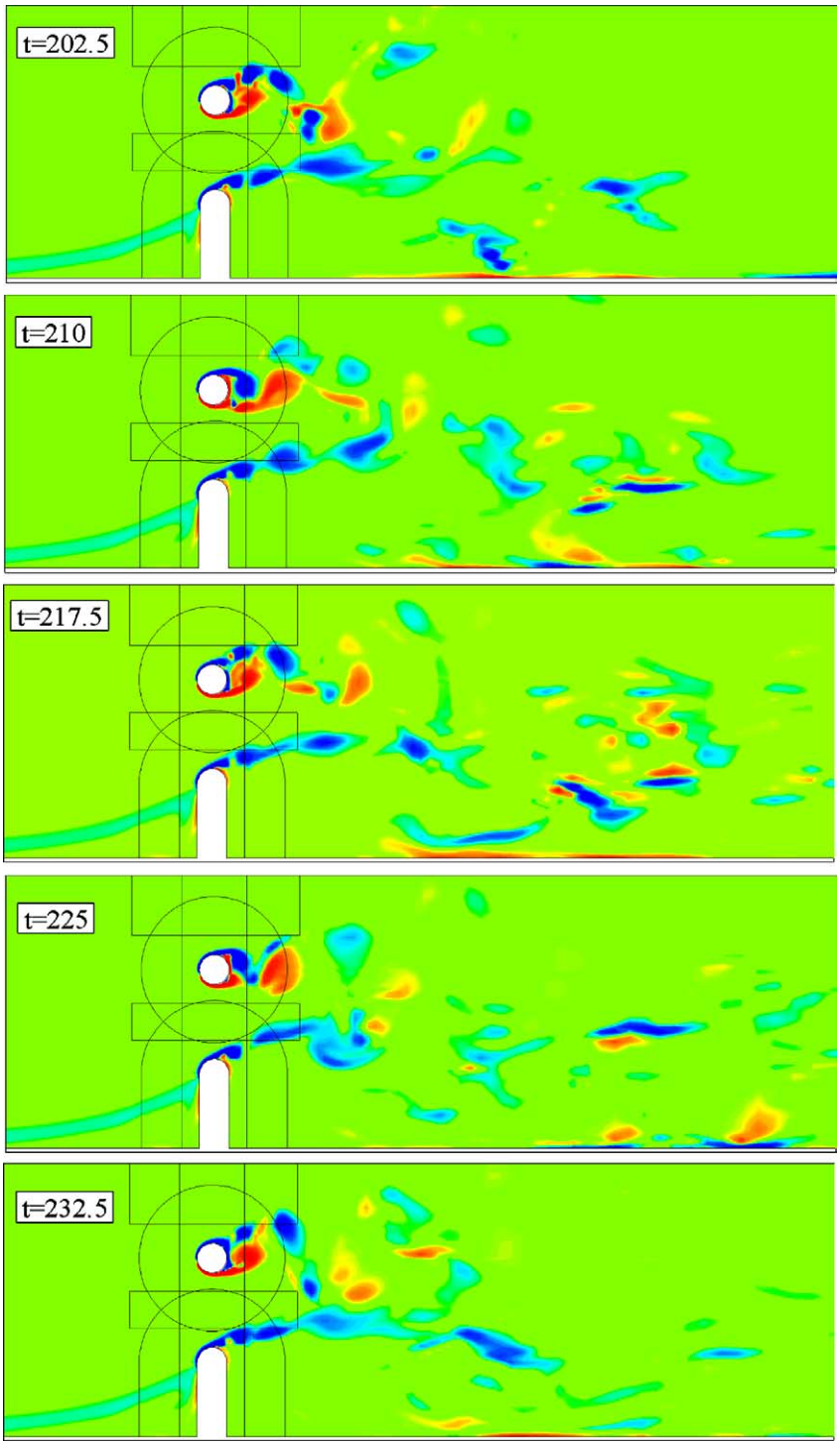


Fig. 23. Channel with two wall-mounted obstacles ($Re = 300$). Contours of vertical vorticity component (Ω_z) at the plane of symmetry at five instants in time. The solid lines mark the boundaries of the various subdomains.

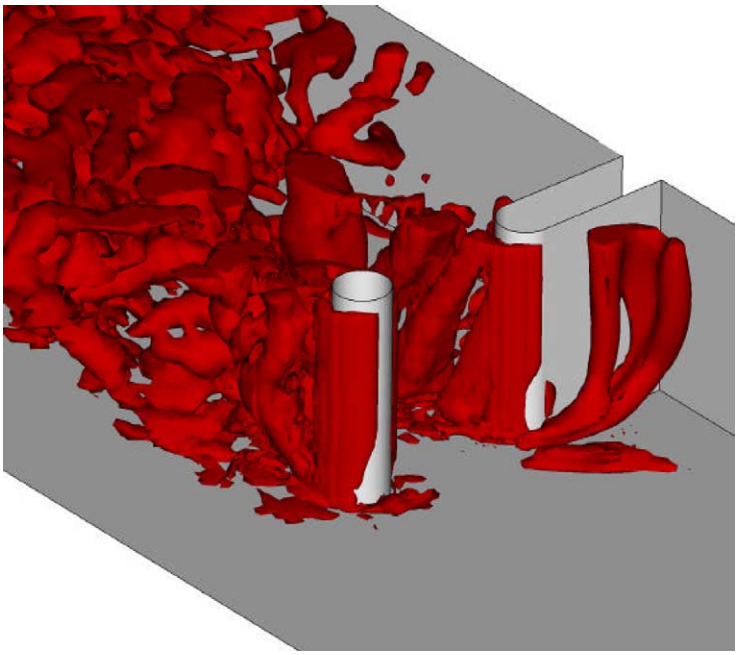


Fig. 24. Channel with two wall-mounted obstacles ($Re = 300$). Instantaneous coherent structures visualized using the λ_2-method [37].

algorithm for calculating the pressure and velocity field at grid interfaces. The algorithm is straightforward to implement in 3D geometries and is inspired by the exact integral constraints that need to be satisfied at overset-grid interfaces to ensure global mass conservation in the composite domain.

We carried out a series of numerical experiments to gauge the accuracy and computational efficiency of the MFBI and SI algorithms. The computed results demonstrated that the SI algorithm tends to give rise to spurious pressure and even velocity oscillations in the overlap region of adjacent subdomains, especially for discontinuous grids, and results in sluggish convergence rates. The new MFBI algorithm, on the other hand, suppresses pressure oscillations at the interface, is less sensitive to grid refinement, and significantly enhances the efficiency of the iterative algorithm. Unlike the SI algorithm, which specifies all four primitive variables via interpolation, MFBI does not over-specify the discrete numerical problem as it requires the specification of three Dirichlet conditions and the satisfaction of a discrete integral constraint. Therefore, MFBI leads to a well-posed discrete problem and this may explain its superior performance relative to the SI algorithm.

We demonstrated the potential of domain decomposition with MFBI as a technique for simulating complex engineering flows by applying it to calculate flow past a circular cylinder mounted between two endplates and flow in a channel with two wall-mounted obstacles. The computed results for the cylinder case were shown to be in good agreement with available experimental data and the recent numerical study of Mittal [21]. Our computations yielded Strouhal numbers and overall shedding patterns in close agreement with the results of Mittal [21]. In particular, we found that at $Re = 100$ oblique shedding of single-cell structures occurs while at $Re = 300$ the competition between Mode A and B shedding patterns dominate the flow.

The simulations presented in this paper have been restricted to laminar flows ($Re \sim 10^2$–10^3). Recently, however, we have successfully adapted our method to solve the unsteady, Reynolds-averaged Navier–Stokes equations and applied it to simulate complex, 3D turbulent flows dominated by unsteady coherent

vortices [1]. We have also applied the method with a great deal of success to simulate vortex shedding in a bileaflet, mechanical prosthetic heart valve at $Re = 1200$ [2]. These recent results along with the computations presented in this paper underscore the potential of the overset-grid approach as a powerful tool for simulating unsteady, 3D flows over a broad range of Reynolds numbers in arbitrarily complex, multi-connected domains. Future work will focus on extending the present algorithm to problems involving moving boundaries.

Acknowledgements

This work was sponsored by NSF, CAREER award 9875691, a grant from Oak Ridge National Laboratory and the DOE monitored by Dr. Michael Sale, and NIH Grant RO1-HL-07262. The authors acknowledge the valuable input of Dr. G.H. Meyer.

References

[1] L. Ge, J. Paik, S.C. Jones, F. Sotiropoulos, Unsteady RANS of Complex 3D Flows Using Overset Grids, in: Proceedings of the 3rd International Symposium on Turbulent and Shear Flow Phenomena, Sendai, Japan, 25–27 June 2003, vol. 1, pp. 67–68.
[2] L. Ge, S.C. Jones, F. Sotiropoulos, T. Healy, A. Yoganathan, Numerical simulation of flow in mechanical heart valves: grid resolution and flow symmetry, ASME Journal of Biomechanical Engineering, 2003, in press.
[3] R. Ramamurti, R. Löhner, W.C. Sandberg, Computation of the 3-D unsteady flow past deforming geometries, Int. J. Comp. Fluid Dyn. 13 (1999) 83–99.
[4] J.A. Benek, J.L. Steger, F.C. Dougherly, A flexible grid embedding technique with application to the Euler equations, AIAA Paper 83-1944 (1983).
[5] G. Chesshire, W.D. Henshaw, Composite overlapping meshes for the solution of partial differential equations, J. Comput. Phys. 90 (1990) 1–64.
[6] M. Hinatsu, J.H. Ferziger, Numerical of unsteady incompressible, flow in complex geometry using composite multigrid technique, Int. J. Numer. Meth. Fluids 13 (1991) 971–997.
[7] H.S. Tang, T. Zhou, On non-conservative algorithms for grid interfaces, SIAM J. Numer. Anal. 37 (1999) 173–193.
[8] C.J. Freitas, S.R. Runnels, Simulation of fluid-structure interaction using patched-overset grids, J. Fluids Struct. 13 (1999) 191–207.
[9] B. Hubbard, H.-C. Chen, A Chimera scheme for incompressible viscous flows with application to submarine hydrodynamics, AIAA Paper 94-2210 (1990).
[10] M.M. Rai, A conservative treatment of zonal boundary scheme for Euler equations calculations, J. Comput. Phys. 62 (1986) 472–503.
[11] J. Wright, W. Shyy, A pressure-based composite grid method for Navier–Stokes equations, J. Comput. Phys. 107 (1993) 225–238.
[12] A.S. Almgren, J.B. Bell, P. Colella, L.H. Howell, M.L. Welcome, A conservative adaptive projection method for the variable density incompressible Navier–Stokes equations, J. Comput. Phys. 142 (1998) 1–46.
[13] A.G. Kravchenko, P. Moin, R. Moser, Zonal embedded grids for numerical simulations of wall-bounded turbulent flows, J. Comput. Phys. 127 (1996) 412–423.
[14] A. Pascarelli, U. Piomelli, G.V. Candler, Multi-block large-eddy simulations of turbulent boundary layers, J. Comput. Phys. 157 (2000) 256–279.
[15] Z.J. Wang, A fully conservative interface algorithm for overlapped grids, J. Comput. Phys. 122 (1995) 96–106.
[16] W.D. Henshaw, A fourth-order accurate method for the incompressible Navier–Stokes equations on overlapping grids, J. Comput. Phys. 113 (1) (1994) 13–25.
[17] Y. Zang, R.L. Street, A composite multigrid method for calculating unsteady incompressible flows in geometrically complex domains, Int. J. Numer. Meth. Fluids 20 (1995) 341–361.
[18] T.M. Burton, J.K. Eaton, Analysis of a fractional-step method on overset grids, J. Comput. Phys. 177 (2) (2002) 336–364.
[19] M.J. Berger, On conservation at grid interfaces, SIAM J. Numer. Anal. 24 (1987) 967–984.
[20] F. Sotiropoulos, Y. Ventikos, Transition from bubble-type vortex breakdown to columnar vortex in a confined swirling flow, Int. J. Heat Fluid Flow 19 (1998) 446–458.
[21] S. Mittal, Computation of three-dimensional flows past cylinder of low aspect ratio, Phys. Fluids 13 (2001) 177–191.

[22] F.B. Lin, F. Sotiropoulos, Assessment of artificial dissipation models for three-dimensional incompressible flows, J. Fluids Eng. Trans. ASME 119 (1997) 331–340.
[23] E. Turkel, V.N. Vatsa, Effect of artificial viscosity on 3-dimensional flow solutions, AIAA J. 32 (1) (1994) 39–45.
[24] H.S. Tang, Numerical simulation of unsteady three dimensional incompressible flows in complex geometries, Ph.D. dissertation, School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA 30332–0355, 2001.
[25] C.L. Merkle, M. Athavale, Time-accurate unsteady incompressible flow algorithm based on artificial compressibility, AIAA Paper 87-1137 (1987).
[26] F.B. Lin, F. Sotiropoulos, Strongly-coupled multigrid method for 3-D incompressible flows using near-wall turbulence closures, ASME J. Fluid Eng. 119 (2) (1997) 314–324.
[27] P.M. Gresho, R.L. Sani, On pressure boundary conditions for the incompressible Navier–Stokes equations, Int. J. Numer. Methods Fluids 7 (1987) 1111–1145.
[28] F. Sotiropoulos, S. Abdallah, The discrete continuity equation in primitive variable solutions of incompressible-flow, J. Comput. Phys. 95 (1) (1991) 212–227.
[29] P.H.M. Bovendeerd, A.A. Steenhoven, F.A. Vosse, G. Vossers, Steady entry flow in a curved pipe, J. Fluid Mech. 177 (1987) 233–246.
[30] A. Roshko, Prespectives on bluff body aerodynamics, J. Wind Eng. Ind. Aerodyn. 49 (1993) 79–100.
[31] C.H.K. Williamson, Oblique and parallel models of vortex shedding in the wake of a circular cylinder at low Reynolds number, J. Fluid Mech. 206 (1989) 579–627.
[32] C.H.K. Williamson, Vortex dynamics in the cylinder wake, Ann. Rev. Fluid Mech. 28 (1996) 477–539.
[33] A. Prasad, C.H.K. Williamson, The instability of the shear layer separating from a bluff body, J. Fluid Mech. 333 (1997) 375–402.
[34] C. Norberg, An experimental investigation of the flow around a circular cylinder: influence of aspect ratio, J. Fluid Mech. 258 (1994) 287–316.
[35] V. Karlo, T.E. Tezduyar, Parallel 3D computations of unsteady flows around circular cylinders, Parallel Comput. 23 (1997) 1235–1248.
[36] H. Persillon, M. Braza, Physical analysis of the transition to turbulence in the wake of a circular cylinder by three-dimensional Navier–Stokes simulation, J. Fluid Mech. 365 (1998) 23–88.
[37] J. Jeong, F. Hussein, On the identification of a vortex, J. Fluid Mech. 285 (1995) 69–94.