Computer Methods in Applied Mechanics and Engineering 94 (1992) 339–351
North-Holland

# A new strategy for finite element computations involving moving boundaries and interfaces — The deforming-spatial-domain/space-time procedure: I. The concept and the preliminary numerical tests*

T.E. Tezduyar, M. Behr

*Department of Aerospace Engineering and Mechanics, Army High-Performance Computing Research Center, and Minnesota Supercomputer Institute, University of Minnesota, 1200 Washington Avenue South, Minneapolis, MN 55415, USA*

J. Liou

*Tulsa Research Center, Amoco Production Company, Tulsa, OK 74102, USA*

Received 23 August 1990
Revised manuscript received 24 October 1990

A new strategy based on the stabilized space-time finite element formulation is proposed for computations involving moving boundaries and interfaces. In the deforming-spatial-domain/space-time (DSD/ST) procedure the variational formulation of a problem is written over its space-time domain, and therefore the deformation of the spatial domain with respect to time is taken into account automatically. Because the space-time mesh is generated over the space-time domain of the problem, within each time step, the boundary (or interface) nodes move with the boundary (or interface). Whether the motion of the boundary is specified or not, the strategy is nearly the same. If the motion of the boundary is unknown, then the boundary nodes move as defined by the other unknowns at the boundary (such as the velocity or the displacement). At the end of each time step a new spatial mesh covers the new spatial domain. For computational feasibility, the finite element interpolation functions are chosen to be discontinuous in time, and the fully discretized equations are solved one space-time slab at a time.

## 1. Introduction

One of the major challenges in computational mechanics and physics is how to handle the moving boundaries and interfaces. There are many applications, practical as well as basic ones, in which this issue must be faced in formulating the computational procedure to be used. To give a few examples, we can mention free-surface flows, two-liquid flows, liquid drops, fluid-structure interactions, large-deformation solid mechanics (including metal forming and penetration mechanics), and melting problems. In all these cases the spatial domain (or the subdomain) deforms with time, and somehow a Lagrangian description of the problem needs to be incorporated.

* This research was sponsored by NASA–Johnson Space Center under grant NAG 9-499 and by NSF under grant MSM-8796352.

A good way to approach the issue is to use the so-called Arbitrary Lagrangian–Eulerian (ALE) finite element procedure (see [1–3]). Simply stated, in the ALE approach, the Lagrangian description is used in zones and directions with 'small' motion, and the Eulerian description is used in zones and directions for which it would not be possible for the mesh to follow the motion. For details of this procedure and various applications we refer the interested readers to [1–3].

The space-time finite element formulation has recently been successfully used for various problems with fixed spatial domains. We are most familiar with [4–8]. The basics of the space-time formulation, its implementation, and the associated stability and accuracy analysis can be found in these references. It is important to remember that the finite element interpolation functions are discontinuous in time so that the fully discrete equations are solved one space-time slab at a time, and this makes the computations feasible.

In the DSD/ST procedure we propose here, the issue of deforming spatial domains is handled by using the space-time finite time element notion. We write the variational formulation of the problem over the associated space-time domain. This way, we automatically take into account the deformation of the spatial domain. The space-time finite element mesh covers the space-time domain of the problem. Therefore, within each time step, the boundary (or interface) nodes move with the boundary (or interface). When the motion of the boundary is unknown, the locations of the boundary nodes at the end of a time step are still not independent unknowns, because the motion of these boundary nodes can be defined in terms the other unknowns (such as the velocity or the displacement) at the boundary.

In general, we have substantial freedom in defining the motion of the mesh. For example, in a free-surface flow problem the only restriction is to move the boundary nodes with the normal velocity of the fluid particles. Sometimes it might also be possible to solve the same problem by moving the boundary nodes with the total velocity of the fluid particles. In fact, in some cases, it might be beneficial to move even the interior nodes with the fluid velocity (or a fraction of it), because this would decrease (relative to the mesh) the local advective effects. We have the same degree of freedom in defining the motion of the mesh in a solid mechanics problem involving free surfaces and large deformations (e.g., metal forming and penetration mechanics). In a melting problem, however, one can only define a physical velocity in the direction normal to the melting surface, and that is the rate with which the melting front moves.

Because the nodes move within each time step, we need to pay particular attention to time stepping to prevent the generation of space-time elements with unacceptable geometries. Because of this, we need to limit the size of the time step to be taken. However, with the space-time finite element formulation, it is quite possible to use spatially local time steps within a temporally accurate formulation (see [5]).

## 2. The stabilized space-time finite element formulation of the incompressible Navier–Stokes equations

We consider a viscous, incompressible fluid occupying at an instant $t \in (0, T)$ a bounded region $\Omega_t$ in $R^{n_{sd}}$, with boundary $\Gamma_t$, where $n_{sd}$ is the number of space dimensions. The velocity and pressure, $\boldsymbol{u}(\boldsymbol{x}, t)$ and $p(\boldsymbol{x}, t)$, are governed by the Navier–Stokes equations:

$$\rho\left(\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u}\right) - \nabla \cdot \boldsymbol{\sigma} = 0 , \quad \text{on } \Omega_t \ \forall t \in (0, T) , \tag{1}$$

$$\nabla \cdot \boldsymbol{u} = 0 , \quad \text{on } \Omega_t \ \forall t \in (0, T) , \tag{2}$$

where $\rho$ is the fluid density, and $\sigma$ is the stress tensor given as

$$\sigma = -p\boldsymbol{I} + 2\mu\, \varepsilon(\boldsymbol{u})\,, \tag{3}$$

with

$$\varepsilon(\boldsymbol{u}) = \tfrac{1}{2}(\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^{\mathrm{t}})\,. \tag{4}$$

Here $\mu$ represents the fluid viscosity while $\boldsymbol{I}$ denotes the identity tensor. Both the Dirichlet and Neumann type boundary conditions are taken into account as shown below:

$$\boldsymbol{u} = \mathbf{g}\,, \qquad \text{on } (\Gamma_t)_{\mathbf{g}}\,, \tag{5}$$

$$\boldsymbol{n}\cdot\sigma = \mathbf{h}\,, \quad \text{on } (\Gamma_t)_{\mathbf{h}}\,, \tag{6}$$

where $(\Gamma_t)_{\mathbf{g}}$ and $(\Gamma_t)_{\mathbf{h}}$ are complementary subsets of the boundary $\Gamma_t$. The initial condition consists of a divergence-free velocity field specified over the entire initial domain:

$$\boldsymbol{u}(\boldsymbol{x},0) = \boldsymbol{u}_0\,, \quad \text{on } \Omega_0\,. \tag{7}$$

In order to construct the finite element function spaces for the DSD/ST method, we partition the time interval $(0, T)$ into subintervals $I_n = (t_n, t_{n+1})$, where $t_n$ and $t_{n+1}$ belong to an ordered series of time levels $0 = t_0 < t_1 < \cdots < t_N = T$. Let $\Omega_t^{\mathrm{h}}$ be the approximation to $\Omega_t$ in $I_n$, and let $\Gamma_t^{\mathrm{h}}$ be the boundary of $\Omega_t^{\mathrm{h}}$. Also let $\Omega_n = \Omega_{t_n}$ and $\Gamma_n = \Gamma_{t_n}$. We shall define the space-time slab $Q_n$ as the domain enclosed by the surfaces $\Omega_n$, $\Omega_{n+1}$, and $P_n$, where $P_n$ is the surface described by the boundary $\Gamma_t^{\mathrm{h}}$ as $t$ traverses $I_n$ (see Fig. 1).

As is the case with $\Gamma_t$, $P_n$ can be decomposed into $(P_n)_{\mathbf{g}}$ and $(P_n)_{\mathbf{h}}$ with respect to the type of boundary condition (Dirichlet or Neumann) being applied. For each space-time slab we define the following finite element interpolation function spaces for the velocity and pressure:

$$(S_{\boldsymbol{u}}^{\mathrm{h}})_n = \{\boldsymbol{u}^{\mathrm{h}} \mid \boldsymbol{u}^{\mathrm{h}} \in [H^{1\mathrm{h}}(Q_n)]^{n_{\mathrm{sd}}},\ \boldsymbol{u}^{\mathrm{h}} \doteq \mathbf{g}^{\mathrm{h}} \text{ on } (P_n)_{\mathbf{g}}\}\,, \tag{8}$$

$$(V_{\boldsymbol{u}}^{\mathrm{h}})_n = \{\boldsymbol{u}^{\mathrm{h}} \mid \boldsymbol{u}^{\mathrm{h}} \in [H^{1\mathrm{h}}(Q_n)]^{n_{\mathrm{sd}}},\ \boldsymbol{u}^{\mathrm{h}} \doteq \mathbf{0} \text{ on } (P_n)_{\mathbf{g}}\}\,, \tag{9}$$

$$(S_p^{\mathrm{h}})_n = (V_p^{\mathrm{h}})_n = \{q^{\mathrm{h}} \mid q^{\mathrm{h}} \in H^{1\mathrm{h}}(Q_n)\}\,. \tag{10}$$

Here $H^{1\mathrm{h}}(Q_n)$ represents the finite dimensional function space over the space-time slab $Q_n$. Over the parent (element) domain, this space is formed by using first-order polynomials in space and, depending on our choice, zeroth- or first-order polynomials in time. Globally, the interpolation functions are continuous in space but discontinuous in time. However, for two-liquid flows, the solution and variational function spaces for pressure should include the functions which are discontinuous across the interface.

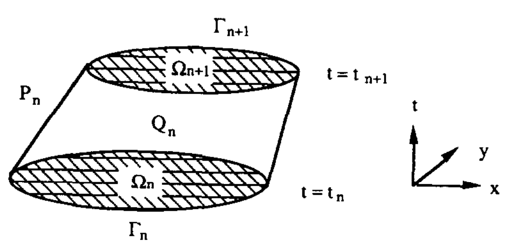


Fig. 1. The space-time slab for the DSD/ST formulation.

The DSD/ST formulation can be written as follows: given $(\boldsymbol{u}^{\mathrm{h}})_n^-$, find $\boldsymbol{u}^{\mathrm{h}} \in (S_u^{\mathrm{h}})_n$ and $p^{\mathrm{h}} \in (S_p^{\mathrm{h}})_n$, such that

$$\begin{aligned}
&\int_{Q_n} \boldsymbol{w}^{\mathrm{h}} \cdot \rho\Big(\frac{\partial \boldsymbol{u}^{\mathrm{h}}}{\partial t} + \boldsymbol{u}^{\mathrm{h}} \cdot \nabla \boldsymbol{u}^{\mathrm{h}}\Big)\,\mathrm{d}Q + \int_{Q_n} \varepsilon(\boldsymbol{w}^{\mathrm{h}}) : \sigma(p^{\mathrm{h}}, \boldsymbol{u}^{\mathrm{h}})\,\mathrm{d}Q \\
&\quad + \sum_{e=1}^{(n_{\mathrm{el}})_n} \int_{Q_n^e} \tau\Big[\rho\Big(\frac{\partial \boldsymbol{w}^{\mathrm{h}}}{\partial t} + \boldsymbol{u}^{\mathrm{h}} \cdot \nabla \boldsymbol{w}^{\mathrm{h}}\Big) - \nabla \cdot \sigma(q^{\mathrm{h}}, \boldsymbol{w}^{\mathrm{h}})\Big] \\
&\qquad\qquad \cdot \Big[\rho\Big(\frac{\partial \boldsymbol{u}^{\mathrm{h}}}{\partial t} + \boldsymbol{u}^{\mathrm{h}} \cdot \nabla \boldsymbol{u}^{\mathrm{h}}\Big) - \nabla \cdot \sigma(p^{\mathrm{h}}, \boldsymbol{u}^{\mathrm{h}})\Big]\,\mathrm{d}Q \\
&\quad + \int_{Q_n} q^{\mathrm{h}} \rho \nabla \cdot \boldsymbol{u}^{\mathrm{h}}\,\mathrm{d}Q + \int_{\Omega_n} (\boldsymbol{w}^{\mathrm{h}})_n^+ \cdot \rho((\boldsymbol{u}^{\mathrm{h}})_n^+ - (\boldsymbol{u}^{\mathrm{h}})_n^-)\,\mathrm{d}\Omega \\
&= \int_{(P_n)_{\mathrm{h}}} \boldsymbol{w}^{\mathrm{h}} \cdot \mathbf{h}\,\mathrm{d}P \quad \forall \boldsymbol{w}^{\mathrm{h}} \in (V_u^{\mathrm{h}})_n \quad \forall q^{\mathrm{h}} \in (V_p^{\mathrm{h}})_n\,.
\end{aligned} \tag{11}$$

This process is applied sequentially to all the space-time slabs $Q_1, Q_2, \ldots, Q_{N-1}$. In the variational formulation given by (11), the following notation is used:

$$(\boldsymbol{u}^{\mathrm{h}})_n^{\pm} = \lim_{\varepsilon \to 0} \boldsymbol{u}(t_n \pm \varepsilon)\,, \tag{12}$$

$$\int_{Q_n} (\cdots)\,\mathrm{d}Q = \int_{I_n} \int_{\Omega_t^h} (\cdots)\,\mathrm{d}\Omega\,\mathrm{d}t\,, \tag{13}$$

$$\int_{P_n} (\cdots)\,\mathrm{d}P = \int_{I_n} \int_{\Gamma_t^h} (\cdots)\,\mathrm{d}\Gamma\,\mathrm{d}t\,. \tag{14}$$

This variational formulation is valid also for two-liquid flows if the surface tension effects at the interface are neglected. The computations start with

$$(\boldsymbol{u}^{\mathrm{h}})_0^- = \boldsymbol{u}_0\,. \tag{15}$$

The coefficient $\tau$ determines the weight of the least-squares terms added to the Galerkin variational formulation to assure the numerical stability of the computations.

*REMARK 1.* This kind of stabilization of the Galerkin formulation is referred to as the Galerkin/least-squares (GLS) procedure [4–9], and can be considered as a generalization of the stabilization based on the streamline-upwind/Petrov–Galerkin (SUPG) procedure [10, 11] employed for incompressible flows. It is with such stabilization procedures that we are able to use an element which has equal-order interpolation functions for velocity and pressure, and which is otherwise unstable. With proper stabilization, elements with equal-order interpolations can be used in place of elements with unequal-order of interpolations [12].

The coefficient $\tau$ used in our formulation is obtained by a simple multi-dimensional generalization of the optimal $\tau$ given in [7] for one-dimensional space-time formulation:

$$\tau = \left(\left(\frac{2}{\Delta t}\right)^2 + \left(\frac{2\|\boldsymbol{u}^{\mathrm{h}}\|}{h}\right)^2 + \left(\frac{4\nu}{h^2}\right)^2\right)^{-1/2}, \tag{16}$$

where $\Delta t$ and $h$ are the temporal and spatial ‘element lengths’. For steady-state computations another definition for $\tau$ is used:

$$\tau = \left( \left( \frac{2\|\boldsymbol{u}^{\mathrm{h}}\|}{h} \right)^2 + \left( \frac{4\nu}{h^2} \right)^2 \right)^{-1/2} . \tag{17}$$

*REMARK* 2. When the mesh movement is prescribed a priori, implementation of (11) is straightforward. More challenging problems, such as free-surface flows and flows with drifting solid objects, involve domains which move and/or deform as functions of the unknown velocity field. In the case of a drifting cylinder, for example, the cylinder moves with unknown linear velocity components $V_1$ and $V_2$ and angular velocity $\dot{\Theta}$. The temporal evolution of these additional unknowns depends on the flow field and can be described by writing the Newton's law for the cylinder:

$$\frac{\mathrm{d}V_1}{\mathrm{d}t} = \frac{D(V_1, V_2, \dot{\Theta}, U)}{m} , \tag{18}$$

$$\frac{\mathrm{d}V_2}{\mathrm{d}t} = \frac{L(V_1, V_2, \dot{\Theta}, U)}{m} , \tag{19}$$

$$\frac{\mathrm{d}\dot{\Theta}}{\mathrm{d}t} = \frac{T(V_1, V_2, \dot{\Theta}, U)}{J} , \tag{20}$$

where $D$, $L$ and $T$ are the drag, lift and torque on the cylinder, while $m$ and $J$ are its mass and polar moment of inertia. The vector of nodal values of velocity and pressure is denoted by $\boldsymbol{U}$. Temporal discretization of (18)–(20) leads to a set of equations which, in abstract form, can be written as

$$\boldsymbol{V} - \boldsymbol{V}^- = \Delta t\, \boldsymbol{D}(\boldsymbol{V}, \boldsymbol{U}) . \tag{21}$$

Here $\boldsymbol{V}$ and $\boldsymbol{V}^-$ are vectors representing the motion of the cylinder, respectively, inside the current space-time slab (unknown) and at the end of the previous one (known). The current slab thickness $t_{n+1} - t_n$ is $\Delta t$. When constant-in-time interpolation is used for $\boldsymbol{V}$ and $\boldsymbol{U}$, (21) takes the form

$$\begin{Bmatrix} (\dot{V}_1)^-_{n+1} \\ (\dot{V}_2)^-_{n+1} \\ (\dot{\Theta})^-_{n+1} \end{Bmatrix} - \begin{Bmatrix} (\dot{V}_1)^-_n \\ (\dot{V}_2)^-_n \\ (\dot{\Theta})^-_n \end{Bmatrix} = \Delta t \begin{Bmatrix} \frac{1}{m} D^-_{n+1} \\ \frac{1}{m} L^-_{n+1} \\ \frac{1}{J} T^-_{n+1} \end{Bmatrix} . \tag{22}$$

For linear-in-time interpolation, on the other hand, (21) takes the form

$$\begin{Bmatrix} (V_1)^-_{n+1} \\ (V_2)^-_{n+1} \\ (\dot{\Theta})^-_{n+1} \\ (V_1)^+_n \\ (V_2)^+_n \\ (\dot{\Theta})^+_n \end{Bmatrix} - \begin{Bmatrix} (V_1)^-_n \\ (V_2)^-_n \\ (\dot{\Theta})^-_n \\ (V_1)^-_n \\ (V_2)^-_n \\ (\dot{\Theta})^-_n \end{Bmatrix} = \Delta t \begin{Bmatrix} \frac{1}{2m} (D^-_n + D^-_{n+1}) \\ \frac{1}{2m} (L^-_n + L^-_{n+1}) \\ \frac{1}{2J} (T^-_n + T^-_{n+1}) \\ \frac{1}{6m} (D^-_n - D^-_{n+1}) \\ \frac{1}{6m} (L^-_n - L^-_{n+1}) \\ \frac{1}{6J} (T^-_n - T^-_{n+1}) \end{Bmatrix} . \tag{23}$$

Based on the general expression (21), we can write the incremental form of (22) or (23) as

$$-\Delta t\left(\frac{\partial \boldsymbol{D}}{\partial \boldsymbol{U}}\right)^{(i)} \Delta \boldsymbol{U}^{(i)} + \left(\boldsymbol{I} - \Delta t\left(\frac{\partial \boldsymbol{D}}{\partial \boldsymbol{V}}\right)^{(i)}\right) \Delta \boldsymbol{V}^{(i)} = \boldsymbol{R}_V(\boldsymbol{U}^{(i)}, \boldsymbol{V}^{(i)}) . \tag{24}$$

Equation (24) is of course coupled with the incremental form of the discrete equation system resulting from (11):

$$(\boldsymbol{M}^*_{UU})^{(i)} \Delta \boldsymbol{U}^{(i)} + (\boldsymbol{M}^*_{UV})^{(i)} \Delta \boldsymbol{V}^{(i)} = \boldsymbol{R}_U(\boldsymbol{U}^{(i)}, \boldsymbol{V}^{(i)}) . \tag{25}$$

In our current implementation, the system (24), (25) is solved by a block iteration scheme in which the term $(\partial \boldsymbol{D}/\partial \boldsymbol{V})^{(i)}$ is neglected. During each iteration (indicated here by subscript i) (25) is solved for $\Delta \boldsymbol{U}$ only, using the value of $\boldsymbol{V}$ from the previous iteration; and then $\boldsymbol{V}$ is updated by (24) while $\boldsymbol{U}$ is held constant. In the future, however, we plan to solve the full system simultaneously to take advantage of larger time steps afforded by an implicit method. Iterating on the solution will still be needed not only because of the nonlinear nature of (1), but also because of the dependence of the element domains $Q^e_n$ on the vector $\boldsymbol{V}$.

## 3. Preliminary numerical tests

We consider four test problems: one-dimensional advection of a discontinuity, travelling cavity, towed cylinder, and drifting cylinder in a uniform flow. In all of these test problems, constant-in-time interpolation functions are used.

*(1) One-dimensional advection of a discontinuity.* This is a one-dimensional, time-dependent, pure advection problem with constant advection velocity. The initial condition is a discontinuity spread across one spatial element. The purpose was to check the mesh moving algorithm, and also to demonstrate that when the mesh is moved with the advection velocity, the local advective effects (relative to the mesh) in minimized and the exact solution is obtained (see Fig. 2). No discontinuity capturing terms were used in the formulation.

*(2) Travelling cavity.* In this problem, a standard lid-driven cavity flow at Reynolds number 400 is 'loaded' on a 'truck' travelling with half the speed of the lid. Consequently the

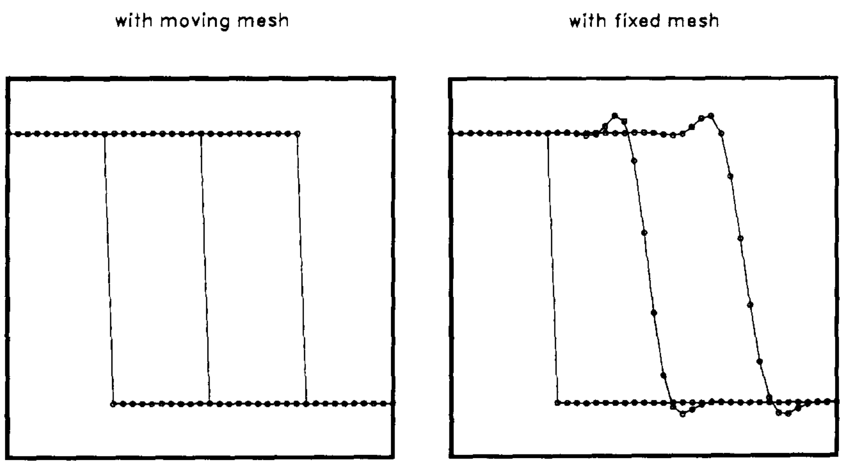


Fig. 2. One-dimensional advection of a discontinuity: initial condition and the solution at $t = 0.25$ and $t = 0.50$, with moving and fixed meshes (no discontinuity capturing).

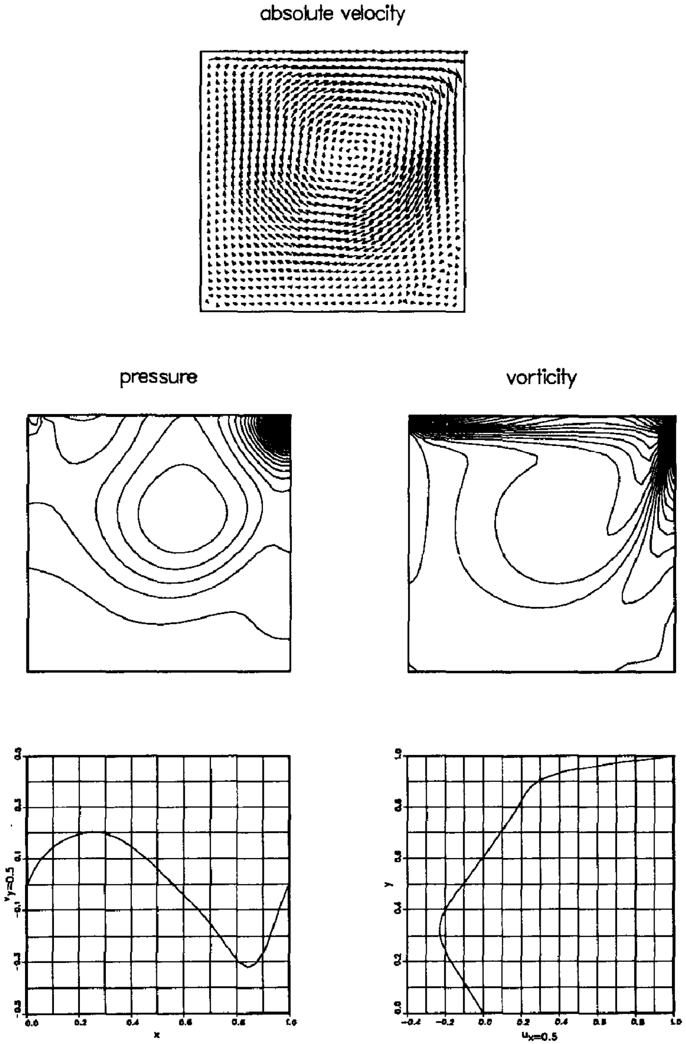


Fig. 3. Stationary cavity: velocity, pressure and vorticity fields, and the velocity profiles on the horizontal and vertical planes passing through the cavity center (steady-state solution).

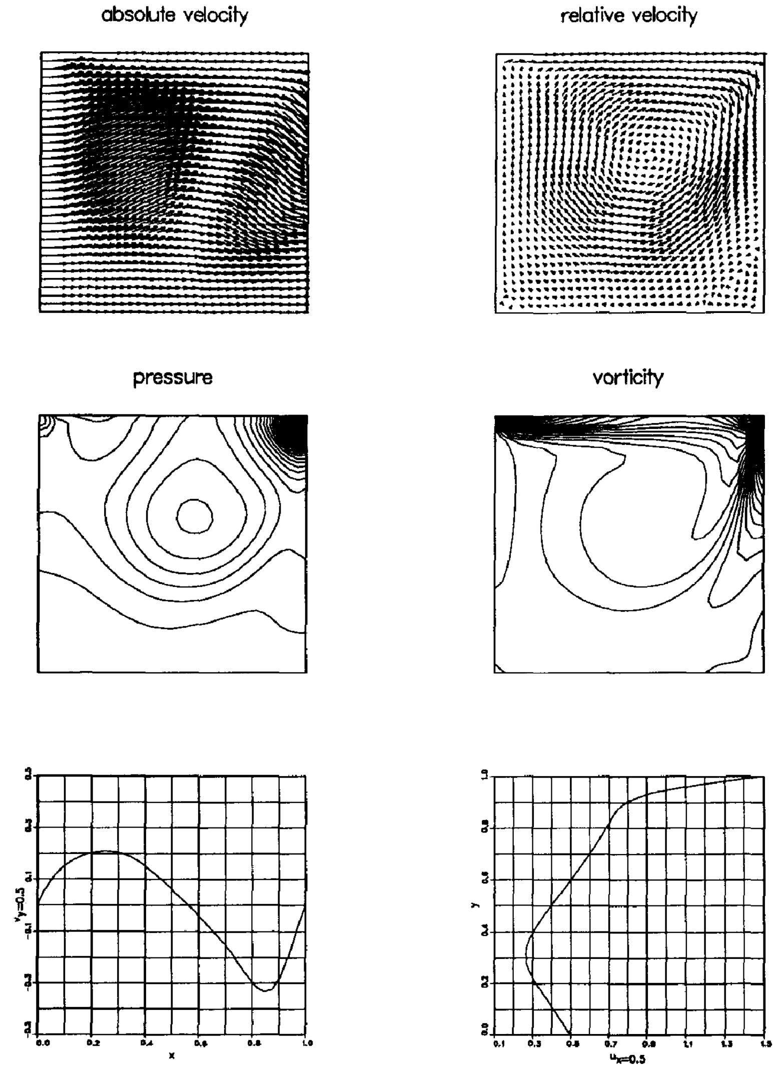


Fig. 4. Translating cavity: velocity, pressure and vorticity fields, and the velocity profiles on the horizontal and vertical planes passing through the cavity center (steady-state solution).

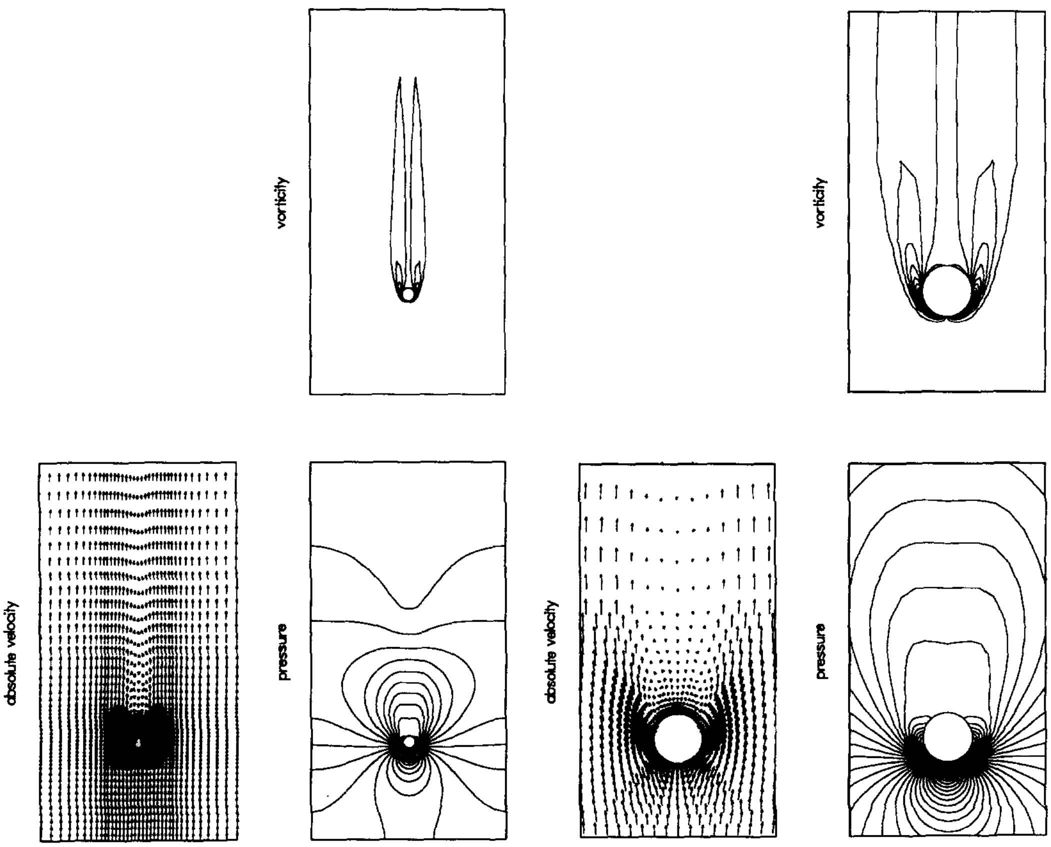


Fig. 5. Stationary cylinder in a uniform flow: velocity, pressure and vorticity fields (steady-state solution).

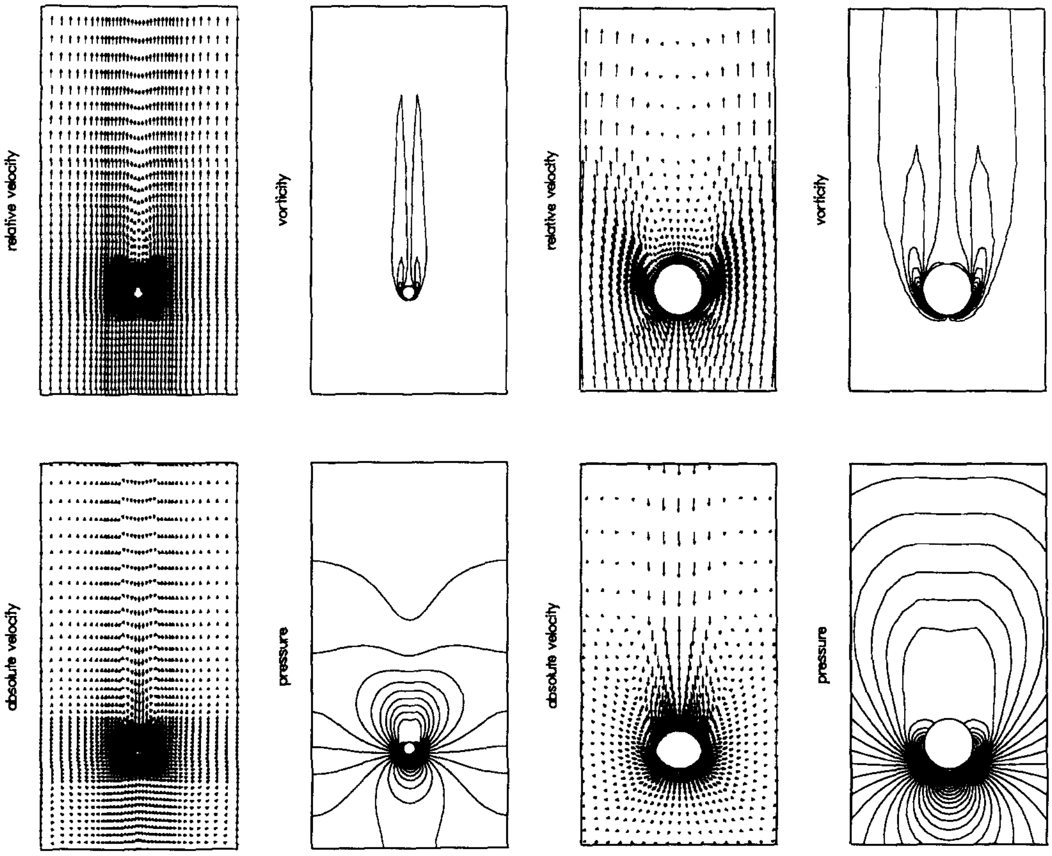


Fig. 6. Towed cylinder in a stationary fluid: velocity, pressure and vorticity fields (steady-state solution).

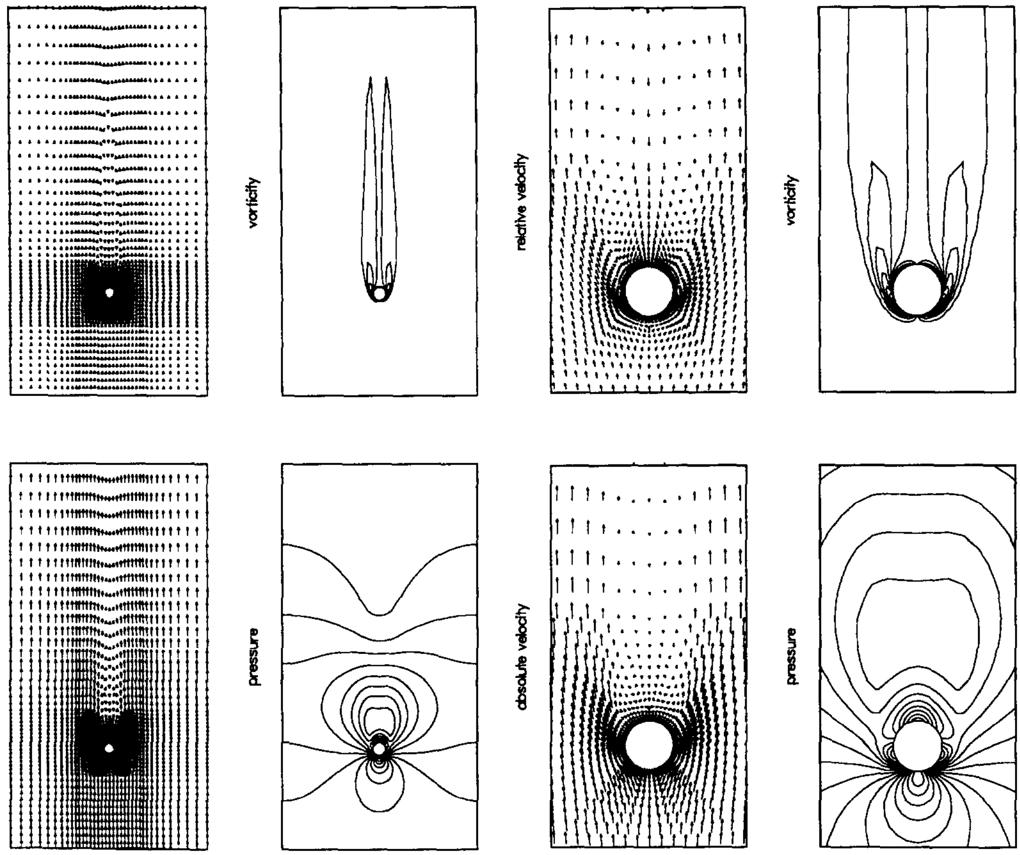


Fig. 8. Drifting cylinder in a uniform flow: velocity, pressure and vorticity fields at $t = 50.0$.

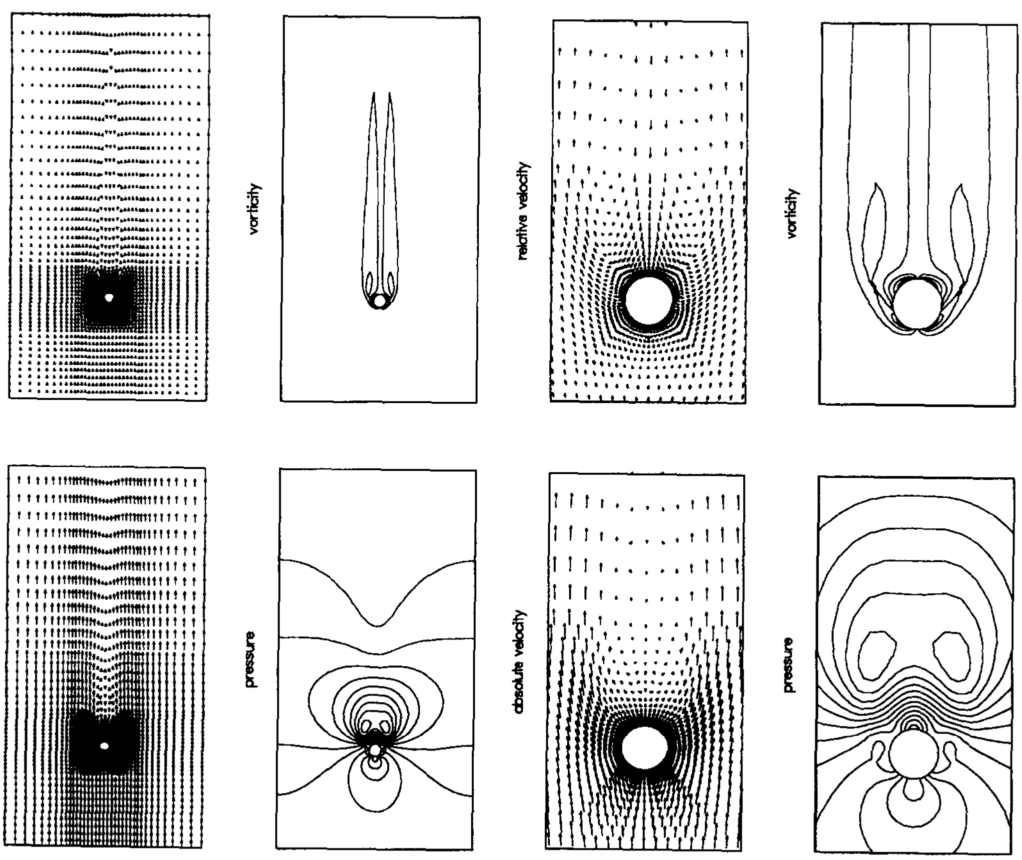


Fig. 7. Drifting cylinder in a uniform flow: velocity, pressure and vorticity fields at $t = 25.0$.

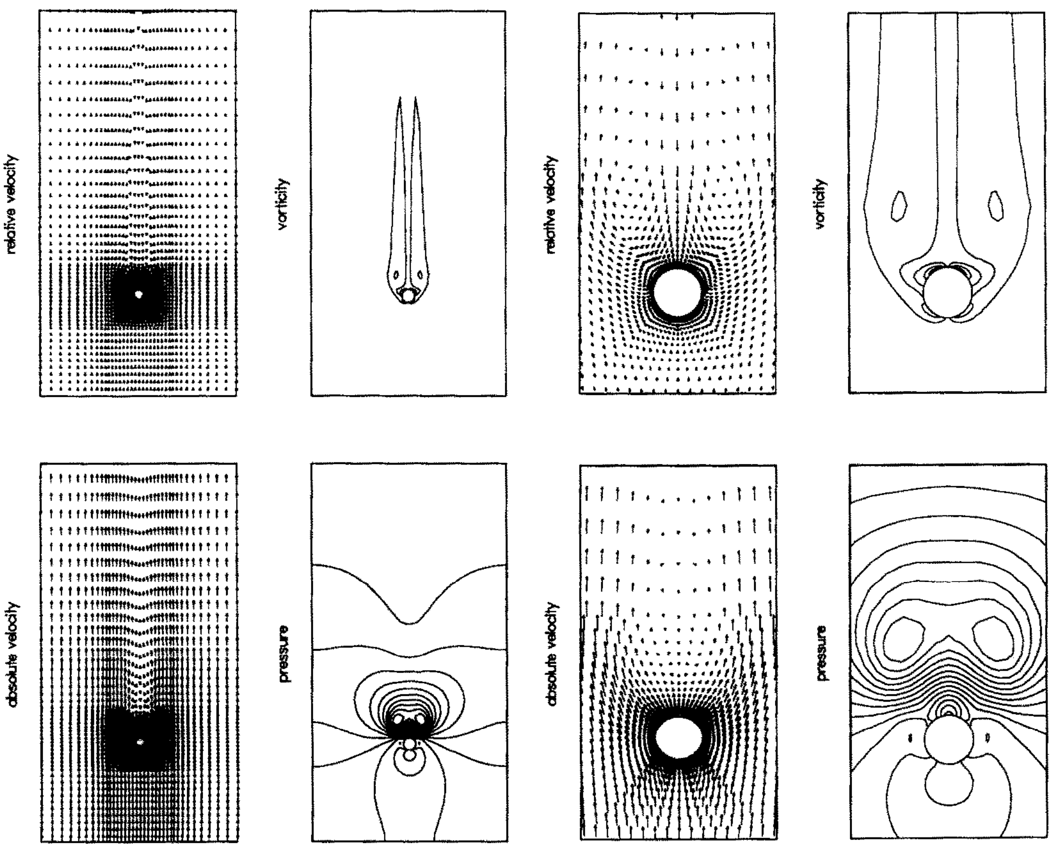


Fig. 9. Drifting cylinder in a uniform flow: velocity, pressure and vorticity fields at $t = 75.0$.

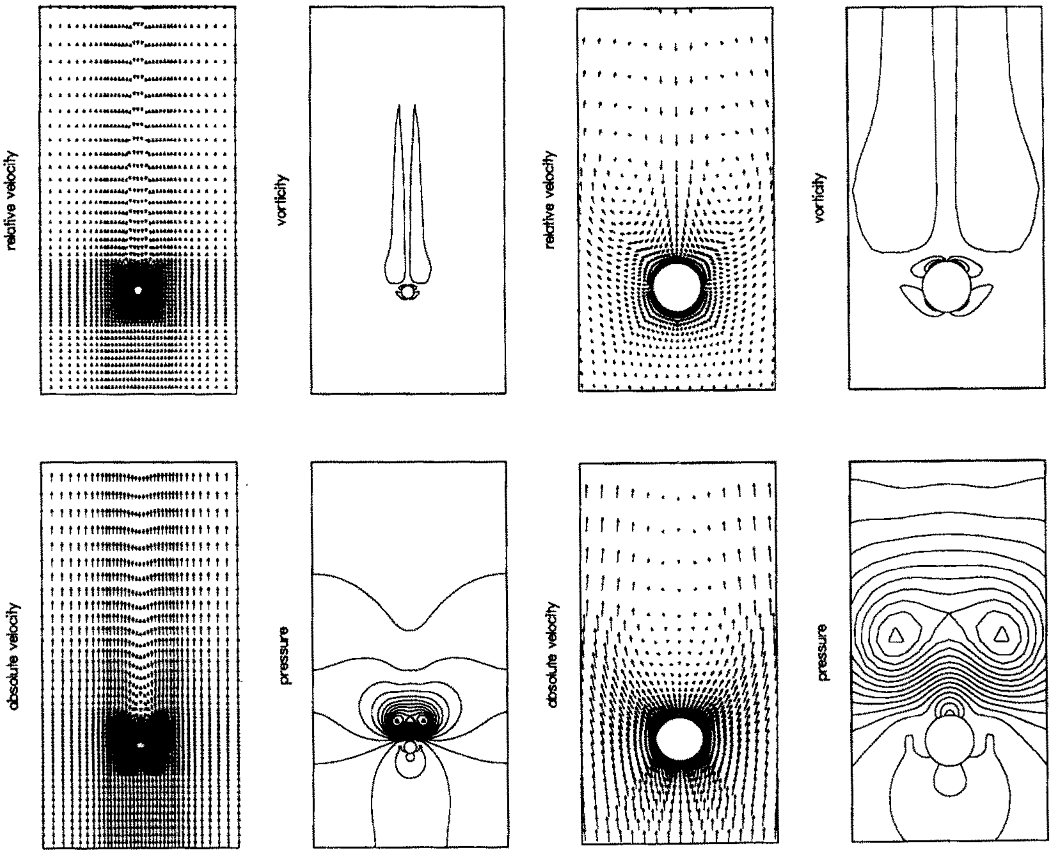


Fig. 10. Drifting cylinder in a uniform flow: velocity, pressure and vorticity fields at $t = 100.0$.

absolute velocity of the lid is 1.5 times that of a stationary cavity. With this seemingly trivial test problem we were able to verify that by moving the mesh with the domain, we can obtain virtually the same steady-state flow field as the one obtained for a stationary cavity (see Figs. 3 and 4).

*(3) Towed cylinder.* In this case a cylinder is being towed in a stationary flow field. The Reynolds number is 100. The mesh is being moved with the cylinder. We impose the free-stream conditions (zero velocity) at the upstream boundary, symmetry conditions at the upper and lower boundaries, and traction-free conditions at the downstream boundary. We compare the steady-state solution with the steady-state solution of flow past a fixed cylinder at Reynolds number 100 (see Figs. 5 and 6). Again the two solutions are in very close agreement.

*(4) Drifting cylinder in a uniform flow.* This test problem involves a cylinder drifting in a uniform flow field. The mesh moves with the cylinder, and the boundary conditions imposed are the same as those for the previous problem, except that the free-stream flow field is not a stationary one but has a uniform magnitude of 0.125. The initial condition is the steady-state solution for the fixed cylinder problem at Reynolds number 100; then we let the cylinder go. The mass and polar moment of inertia of the cylinder are 1.0 and 0.5, respectively. Figures

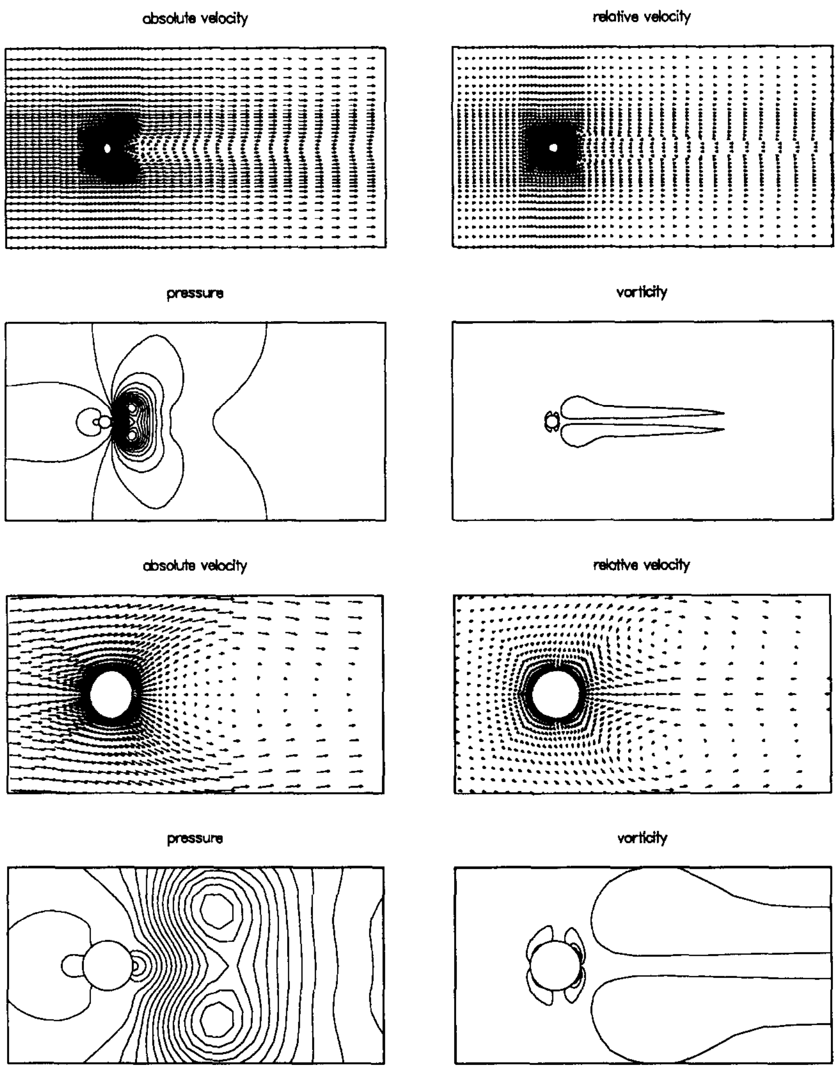


Fig. 11. Drifting cylinder in a uniform flow: velocity, pressure and vorticity fields at $t = 125.0$.

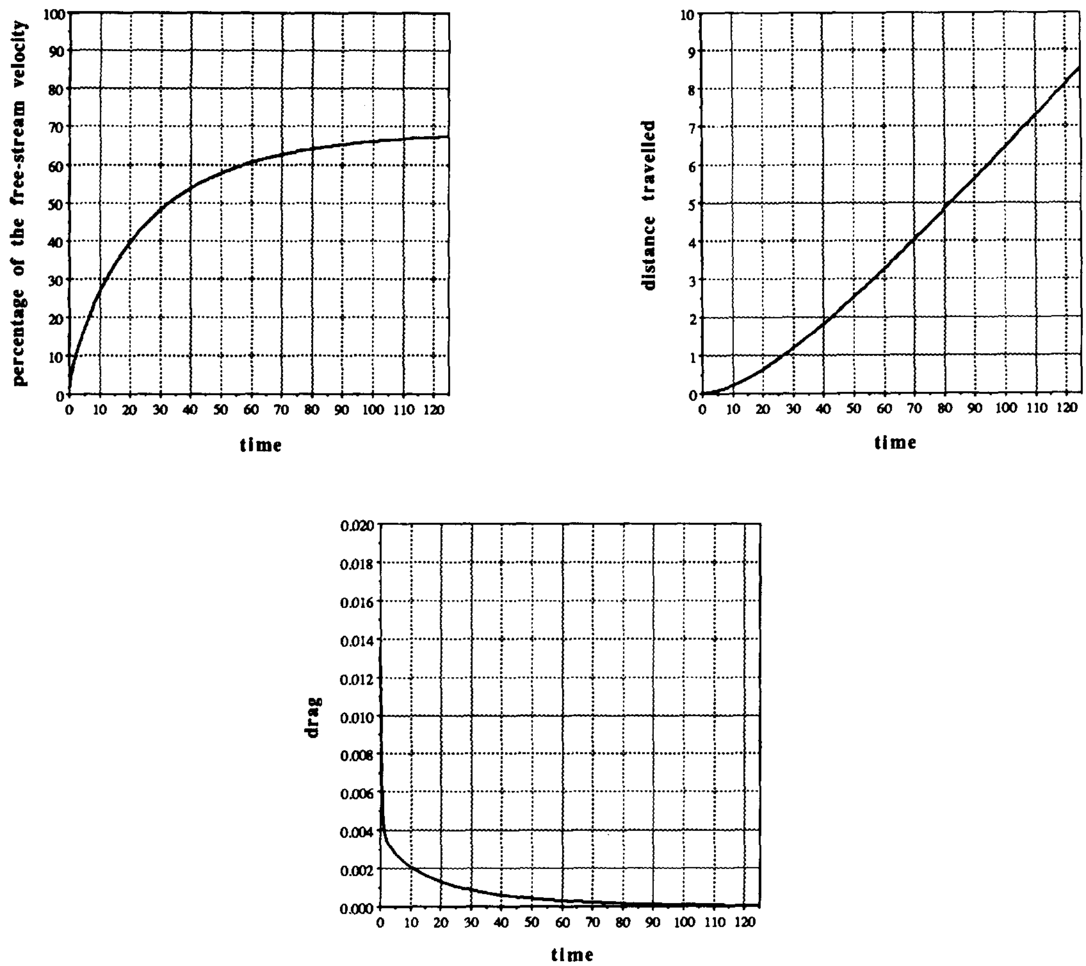


Fig. 12. Drifting cylinder in a uniform flow: time histories of the velocity of the cylinder, distance it travels, and drag on the cylinder.

7–11 show the solution at various instants in time. Figure 12 shows the time histories of the velocity of the cylinder, distance it travels, and drag on the cylinder.

## References

[1] T.J.R. Hughes, W.K. Liu and T.K. Zimmermann, Lagrangian–Eulerian finite element formulation for incompressible viscous flows, Comput. Methods Appl. Mech. Engrg. 29 (1981) 329–349.

[2] W.K. Liu, H. Chang, J.-S. Chen and T. Belytschko, Arbitrary Lagrangian Eulerian Petrov–Galerkin finite elements for nonlinear continua, Comput. Methods Appl. Mech. Engrg. 68 (1988) 259–310.

[3] W.K. Liu, J.-S. Chen, T. Belytschko and Y.F. Zhang, Adaptive ALE finite elements with particular reference to external work rate on frictional interface, Comput. Methods Appl. Mech. Engrg. 93 (1991) 189–216.

[4] T.J.R. Hughes, L.P. Franca and M. Mallet, A new finite element formulation for computational fluid dynamics: VI. Convergence analysis of the generalized SUPG formulation for linear time-dependent multi-dimensional advective-diffusive systems, Comput. Methods Appl. Mech. Engrg. 63 (1987) 97–112.

[5] T.J.R. Hughes and G.M. Hulbert, Space-time finite element methods for elastodynamics: Formulations and error estimates, Comput. Methods Appl. Mech. Engrg. 66 (1988) 339–363.

[6] T.J.R. Hughes, L.P. Franca and G.M. Hulbert, A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/least-squares method for advective-diffusive equations, Comput. Methods Appl. Mech. Engrg. 73 (1989) 173–189.

[7] F. Shakib, Finite element analysis of the compressible Euler and Navier–Stokes equations, Ph.D. thesis, Stanford University, 1988.
[8] P. Hansbo and A. Szepessy, A velocity–pressure streamline diffusion finite element method for the incompressible Navier–Stokes equations, Comput. Methods Appl. Mech. Engrg. 84 (1990) 175–192.
[9] T.E. Tezduyar, R. Shih, S. Mittal and S.E. Ray, Incompressible flow computations with stabilized bilinear and linear equal-order-interpolation velocity-pressure elements, University of Minnesota Supercomputer Institute, Research Report UMSI 90/165, September 1990. ← See update U1 below.
[10] A.N. Brooks and T.J.R. Hughes, Streamline-upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on incompressible Navier–Stokes equations. Comput. Methods Appl. Mech. Engrg. 32 (1982) 199–259.
[11] T.E. Tezduyar, J. Liou and D.K. Ganjoo, Incompressible flow computations based on the vorticity-stream function and velocity-pressure formulations, Comput. & Structures 35 (1990) 445–472.
[12] T.E. Tezduyar, S. Mittal and R. Shih, Time-accurate incompressible flow computations with quadrilateral velocity–pressure elements, Comput. Methods Appl. Mech. Engrg. 87 (1991) 363–384.

**Publication Update**

U1. T.E. Tezduyar, S. Mittal, S.E. Ray and R. Shih, "Incompressible Flow Computations with Stabilized Bilinear and Linear Equal-order-interpolation Velocity-Pressure Elements", *Computer Methods in Applied Mechanics and Engineering*, **95** (1992) 221-242.