A NEW STRATEGY FOR FINITE ELEMENT COMPUTATIONS INVOLVING MOVING BOUNDARIES AND INTERFACES— THE DEFORMING-SPATIAL-DOMAIN/SPACE-TIME PROCEDURE: II. COMPUTATION OF FREE-SURFACE FLOWS, TWO-LIQUID FLOWS, AND FLOWS WITH DRIFTING CYLINDERS

T.E. Tezduyar, M. Behr, S. Mittal
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

23 August 1990
Revised 24 October 1990

Abstract

New finite element computational strategies for free-surface flows, two-liquid flows, and flows with drifting cylinders are presented. These strategies are based on the DSD/ST (Deforming-Spatial-Domain/Space-Time) procedure. In the DSD/ST approach, the variational formulations for this type of flow problems are written over their space-time domains. One of the important features of the approach is that it enables one to circumvent the difficulty involved in remeshing every time step and thus reduces the projection errors introduced by such frequent remeshings. Computations are performed for various test problems mainly for the purpose of demonstrating the computational capability developed for this class of problems. In some of the test cases, such as the liquid drop problem, surface tension is taken into account. For flows involving drifting cylinders, the mesh moving and remeshing schemes proposed are convenient and reduce the frequency of remeshing.

1. Introduction

The DSD/ST (Deforming-Spatial-Domain/Space-Time) procedure was proposed in [1] for finite element computations involving moving boundaries and interfaces. The procedure is based on the space-time finite element formulation which has been successfully used for various compressible and incompressible flow problems with fixed spatial domains (see for example [2–6]). In the DSD/ST procedure, deformation and/or motion of the spatial domain is taken into account automatically by writing the variational formulation of the problem

over the associated space-time domain. For more on the general concept of the DSD/ST procedure, various related issues, and the general variational formulation, we refer the interested readers to reference [1].

We expect that the relationship and the similarities between the DSD/ST procedure and the Arbitrary Lagrangian-Eulerian (ALE) formulation [7–9] will be a source of discussion in the future. At this time it is perhaps too early to debate the relative merits of these methods; however, we can say that the two methods were designed to serve roughly the same purpose, with what appears to be similar philosophies. We can also say that the DSD/ST procedure is conceptually easier to understand, and therefore is simpler to implement.

It was stated in [1] that the DSD/ST procedure can be applied to a wide range of problems involving moving boundaries and interfaces. Among the exemplary areas of applications listed in [1] were free-surface flows, two-liquid flows and liquid drops. In this paper we perform computations for the purpose of demonstrating the computational capability of the DSD/ST procedure for this type of flows. While some of these computations will be quite preliminary, for some others we will be able to compare our results with previously published results obtained with different methods. Also in [1], some test computations were performed for flow problems involving moving cylinders, such as towing a cylinder in a stationary fluid or letting a cylinder drift in a uniform flow field. For example, it was shown that the steady-state solution obtained for the towed cylinder was virtually identical to the steady-state solution of flow past a fixed cylinder for the same Reynolds number. In this paper, we will also continue to give numerical examples to demonstrate the computational capability developed for moving solid objects in general, and moving cylinders in particular. For all flow problems considered in this paper, the purpose will not be to conduct an in-depth physical investigation, but to show what we can do with the DSD/ST procedure for this type of problems.

In the DSD/ST approach, the motion of free-surfaces, interfaces and solid boundaries is accommodated by moving the boundary nodes. Therefore there will not always be a need for remeshing. Here we define remeshing as the process of generating a new mesh at the end of a time step and projecting the solution from the old mesh to the new mesh. Since remeshing, in general, involves projection errors, we try to minimize the frequency of remeshing. In the case of a moving cylinder, in addition to minimizing the frequency of remeshing, with the particular mesh we use, we will be able to simplify the remeshing process significantly. In some cases, it is possible to get automatic mesh refinement as a byproduct of the moving mesh feature of the DSD/ST procedure. We will discuss the issues related to mesh moving and remeshing further in Section 3.

In Section 2 we describe the incorporation of surface tension effects into the formulation which was given in [1]. Numerical examples and concluding remarks will be given in Sections 4 and 5.

2. Incorporation of the Surface Tension into the Formulation

In this section we describe how the surface tension effects are incorporated into the formulation which was given in [1]. Consider two viscous, immiscible, incompressible fluids, A and B, occupying at an instant $t \in (0, T)$ a bounded region Ω_t in $\mathbb{R}^{n_{\text{sd}}}$, with boundary Γ_t, where n_{sd} is the number of space dimensions. Let $(\Omega_A)_t$ denote the subdomain occupied by

fluid A, and $(\Gamma_A)_t$ denote the boundary of this subdomain. Similarly, let $(\Omega_B)_t$ and $(\Gamma_B)_t$ be the subdomain and boundary associated with fluid B. Furthermore, let $(\Gamma_{AB})_t$ be the intersection of $(\Gamma_A)_t$ and $(\Gamma_B)_t$, i.e., the interface between fluids A and B.

The kinematical conditions at the interface $(\Gamma_{AB})_t$ are automatically satisfied because the discretized subdomains $(\Omega_A)_t$ and $(\Omega_B)_t$ share the nodes at this interface. The dynamical conditions at the interface, for two-dimensional problems, can be expressed by the following equation:

$$\mathbf{n}_A \cdot \boldsymbol{\sigma}_A + \mathbf{n}_B \cdot \boldsymbol{\sigma}_B = \mathbf{n}_A \gamma / R_A, \quad \text{on } (\Gamma_{AB})_t \quad \forall t \in (0, T), \tag{1}$$

where $\mathbf{n}_A$ and $\mathbf{n}_B$ are the unit outward normal vectors at the interface, $\boldsymbol{\sigma}_A$ and $\boldsymbol{\sigma}_B$ are the stress tensors, γ is the surface tension coefficient, and R_A is the radius of curvature defined to be positive when $\mathbf{n}_A$ points towards the center of curvature. To impose this condition, we add the term

$$\int_{(P_n)_{AB}} \mathbf{w}^h \cdot \mathbf{n}_A \gamma / R_A dP \tag{2}$$

to the variational formulation given in [1]. Here $\mathbf{w}^h$ is the weighting function belonging to a suitable function space, and $(P_n)_{AB}$ is the space-time surface described by the boundary $(\Gamma_{AB})_t$ as t traverses the time interval (t_n, t_{n+1}). The variational formulation modified this way to account for surface tension effects will be formally valid also for free surface flows (i.e. when the second fluid does not exist), provided that subdomain $(\Omega_A)_t$ is the one assigned to be occupied by the fluid. Certainly, the modified variational formulation can easily be extended to more than two fluids.

3. Moving the Mesh and Remeshing

In the DSD/ST procedure, to facilitate the motion of free-surfaces, interfaces and solid boundaries, we need to move the boundary nodes with the normal component of the velocity at those nodes. Except for this restriction, we have the freedom to move all the nodes any way we would like to. With this freedom, we can move the mesh in such a way that we only need to remesh when it becomes necessary to do so to prevent unacceptable degree of mesh distortion and potential entanglement. By minimizing the frequency of remeshing we minimize the projection errors expected to be introduced by remeshing. In fact, for some computations, as a byproduct of moving the mesh, we may be able to get a limited degree of automatic mesh refinement, again with minimal projection errors.

Figure 1 shows the topologically typical (but not actual) meshes we use for the computation of flow fields involving a cylinder drifting in a bounded flow domain. We can illustrate some of the concepts related to moving the mesh, mesh refinement and remeshing, by referring to these meshes.

Let us assume that the first mesh is what we have initially when we let the cylinder go. As the cylinder drifts horizontally and vertically it carries the mesh within the square region with it, as shown in the next two frames. This way the cylinder can travel several radii without

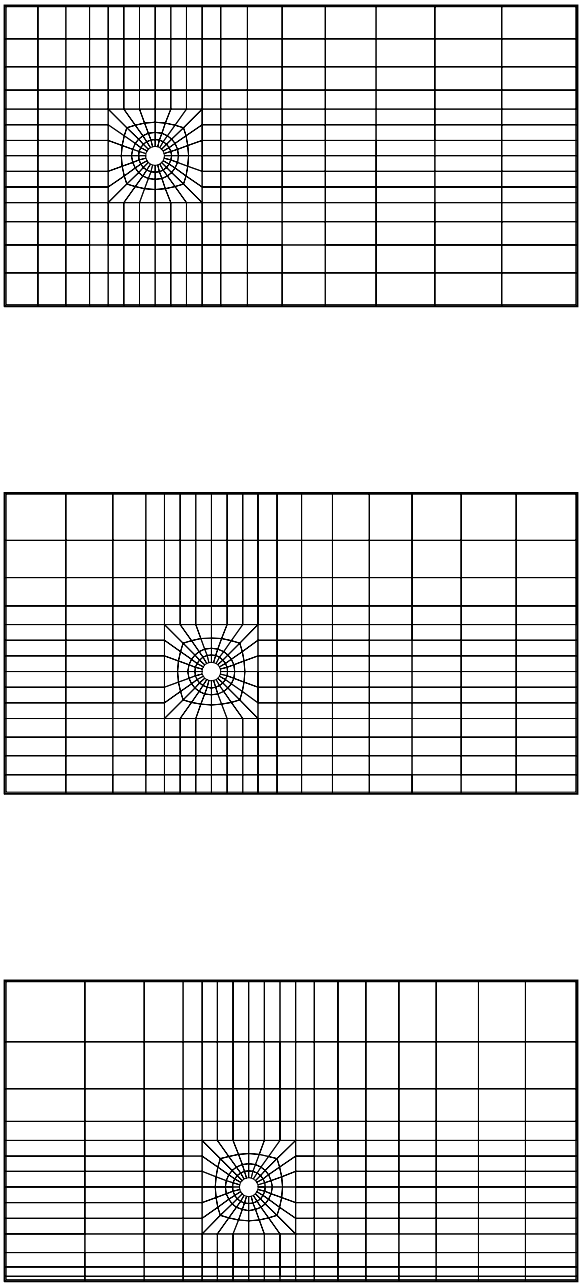

Figure 1. Typical (but not actual) meshes used for the computation of flow fields involving a cylinder drifting in a bounded domain.

severe distortion of the elements and therefore with no need for remeshing. Also because the mesh within the square region travels with the cylinder, the mesh refinement around the cylinder is always exactly the same, and thus the level of mesh refinement required for sufficient accuracy is retained. It is also interesting to note that, as the cylinder gets closer to the lower wall, the refinement in the region between the cylinder and the wall increases.

If remeshing becomes necessary, it still will not be a very difficult process. This is because only the region outside the square region needs to be remeshed, and that part of the mesh is

quite structured, and therefore it will be quite easy to remesh and project the solution from the old mesh to the new one. For the mesh within the square region around the cylinder, there will be no remeshing and thus no projection error. Of course, if the cylinder gets too close to the wall, then a new and smaller square region of mesh will need to be generated around the cylinder.

For flows involving a cylinder drifting in an unbounded flow domain, naturally, the entire mesh translates with the cylinder, and therefore there is no issue of remeshing (see [1]).

4. Numerical Examples

All computations were performed with linear-in-time interpolation functions.

(1) Free-surface wave-propagation problem. This is a problem that was considered in [7]. Initially the fluid is stationary and occupies a long rectangular region with dimensions $L \times D$, where $L = 949.095$ and $D = 10$. The flow is assumed to be inviscid, and both the density and the gravity is set to 1.0. The mesh consists of 320 elements, with two elements through depth. The wave is generated by prescribing the velocity along the left-hand boundary of the domain according to the following expression:

$$u_1 = \frac{Hc}{D} \operatorname{sech}^2(c\kappa t/D - 4), \tag{3}$$

where

$$c = (g(D + H))^{1/2}, \tag{4}$$

$$\kappa = (3H/4D)^{1/2}, \tag{5}$$

with $g = 1$, and $H = 0.86$. The time step is 1.789. The surface nodes are moved with the local velocity component normal to the surface. Figure 2 shows the solutions obtained at various time steps. After 160 time steps, the wave retains 94.4% of its initial amplitude. These solutions compare well with those presented in [7]. When we solved the same problem by using a mesh which consists of 160 elements with only one element through depth, the solutions obtained were nearly the same.

(2) Pulsating drop problem. In this problem the drop is initially of elliptical shape with horizontal and vertical axes 1.25 and 0.80. The density, viscosity and the surface tension coefficient are 1.0, 0.001 and 0.001, respectively. We neglect the effect of gravity. The number of elements is 380, and the time step is 1.0. The surface nodes are moved with the nodal velocity while the nodes in the square core are kept fixed. Figure 3 shows the time history of the principal axes of the drop. Figures 4 (a)–(d) show the flow field and finite element mesh corresponding, approximately, to points a, b, c, and d in Fig. 3.

(3) Two-liquid interface problem. In this problem, two liquids, with densities 1.0 and 2.0, occupy a closed "tank" with dimensions 0.8×0.6. The interface is initially linear with an average height of 0.3 and a slope of -0.250, and the lighter liquid is on top of the heavier one. The dynamic viscosity for both liquids is 0.001. The gravity is 0.294, and the surface tension is neglected. The number of elements is 882, equally distributed between the two

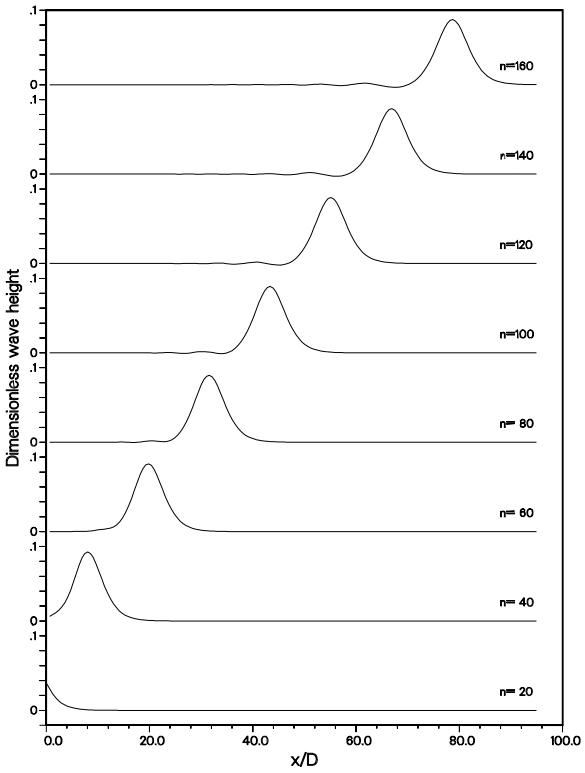

Figure 2. Free-surface wave-propagation problem.

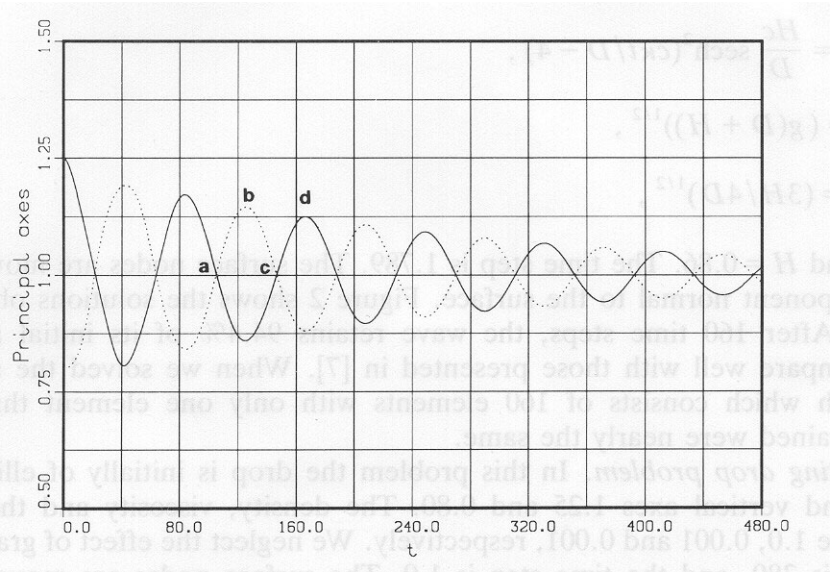

Figure 3. Pulsating drop problem: time histories of the principal axes of the drop.

liquids; the time time step is 0.5. The interface nodes are moved with the nodal velocities, and the nodes at the top and bottom walls of the "tank" remain still. Figure 5 shows the time history of the vertical location (relative to the average height of 0.3) of the interface along the left and right hand sides of the "tank". Figures 6 (a)–(d) show the flow field and

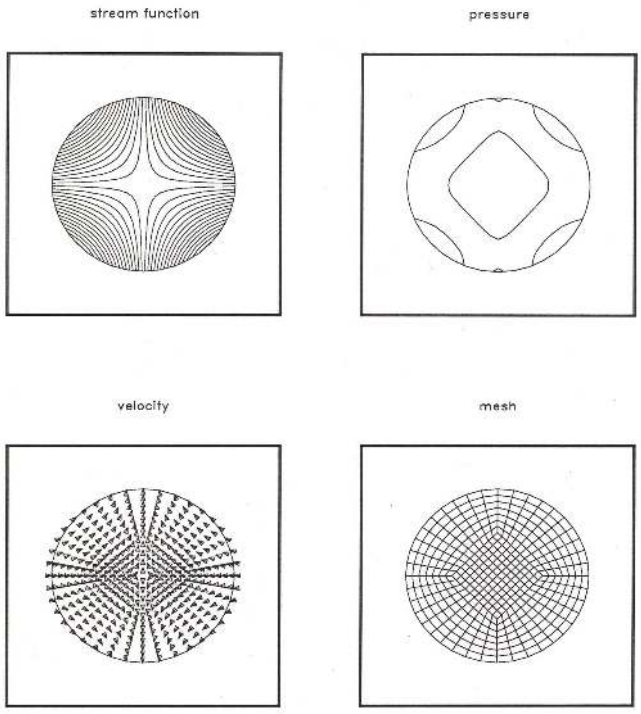

Figure 4. (a) Pulsating drop problem: the flow field and finite element mesh corresponding, approximately, to point a in Fig. 3.

finite element mesh corresponding, approximately, to points a, b, c, and d in Fig. 5.

(4) Large-amplitude sloshing. This problem is very similar to the one that was considered in [9]. Initially the fluid is stationary and occupies a 2.667×1.0 rectangular region. The density and viscosity are 1.0 and 0.002. The gravity is 1.0, and the surface tension is neglected. The wave is created by applying a horizontal body force of $A\sin(\omega t)$, where $A = 0.01$ and $\omega = 0.978$. The number of elements is 441, and the time step is 0.107. With these values of the frequency and the time step size, a single period of the forcing function takes 60 time steps. The surface nodes are moved with the local velocity component normal to the surface. Figure 7 shows the time history of the vertical location (relative to the average height of 1.0) of the free-surface along the left and right hand sides of the "tank". Figures 8 (a)–(d) show the flow field and finite element mesh corresponding, approximately, to points a, b, c, and d in Fig. 7. In [9] the horizontal body force is removed after ten cycles; in our case, on the other hand, this force is maintained during the entire computation. Therefore it takes longer time for our solution to reach its fully periodic pattern, and the waves we compute have larger amplitudes. It should also be noted that while Q1P0 (bilinear velocity–constant pressure) elements are used in [9], we use stabilized Q1Q1 (bilinear velocity–continuous bilinear pressure) elements.

(5) Cylinder drifting in shear flow. This test problem involves a cylinder (with unit radius) drifting in a shear flow in a 61×32 bounded domain. The density and viscosity are 1.0 and 0.005. The upper and lower walls move with velocities 0.156 and 0.094. The upstream velocity profile is assumed to be a linear interpolation of the velocities at the upper and lower boundaries. At the downstream boundary normal and shear stresses are specified

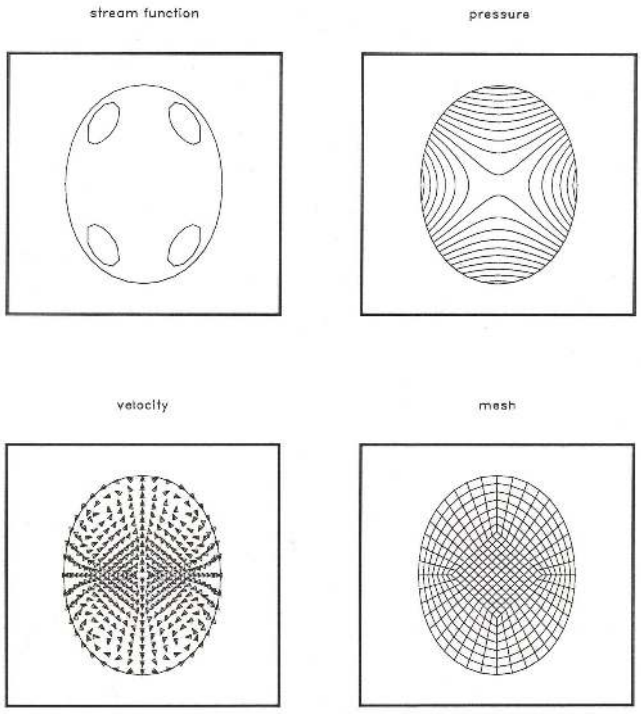

Figure 4. (b) Pulsating drop problem: the flow field and finite element mesh corresponding, approximately, to point b in Fig. 3.

consistent with the undisturbed shear flow. The Reynolds number based on the average upstream velocity and the cylinder diameter is 50. The mass and polar moment of inertia of the cylinder are 2π and π, respectively. The initial condition is the steady-state solution for the fixed cylinder located at (16,16) relative to the lower left corner; then we let the cylinder go. The number of elements is 1152, and the time time step is 0.125. To prevent temporal oscillations due to the start-up, for the first few time steps we subdivide each time step and use these smaller step sizes. The mesh within the square region around the cylinder moves with the cylinder as described in Section 3. Figure 9 shows, for the cylinder, the time history of the drag, lift, torque, linear and angular velocity components, displacement components, and rotation. Figures 10 (a)–(e) show the flow field and finite element mesh at $t = 0.00$, 31.25, 62.50, 93.75, and 125.00.

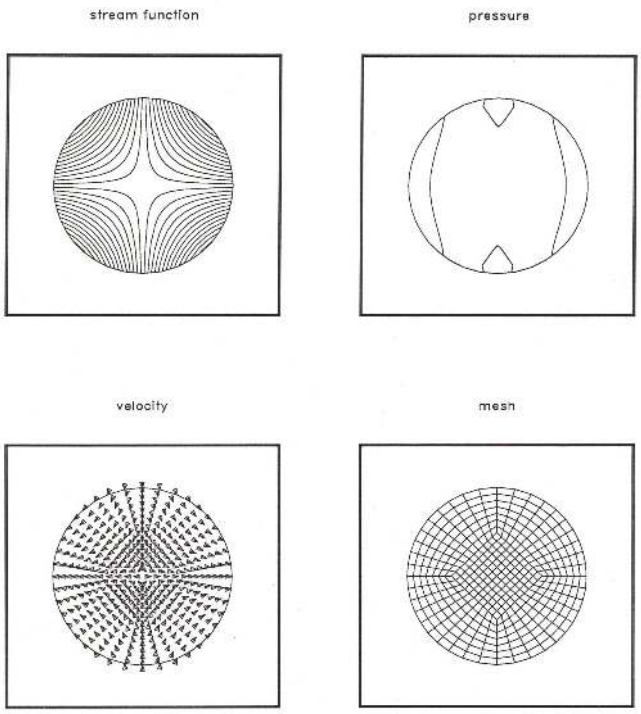

Figure 4. (c) Pulsating drop problem: the flow field and finite element mesh corresponding, approximately, to point c in Fig. 3.

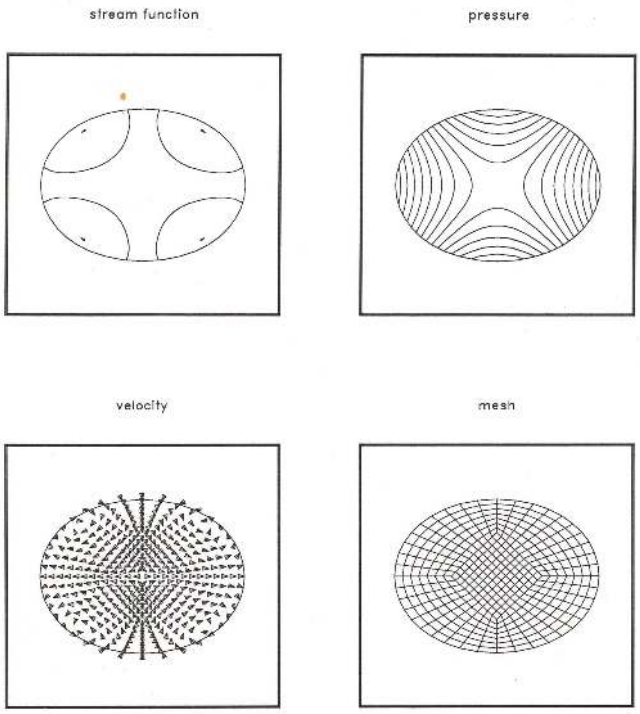

Figure 4. (d) Pulsating drop problem: the flow field and finite element mesh corresponding, approximately, to point d in Fig. 3.

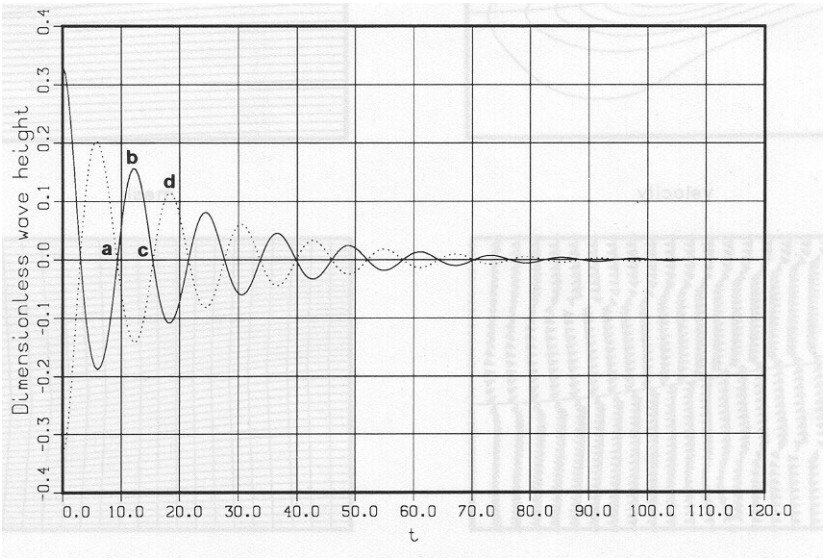

Figure 5. Two-liquid interface problem: time histories of the vertical location (relative to the average height of 0.3) of the interface along the left- and right-hand sides of the "tank".

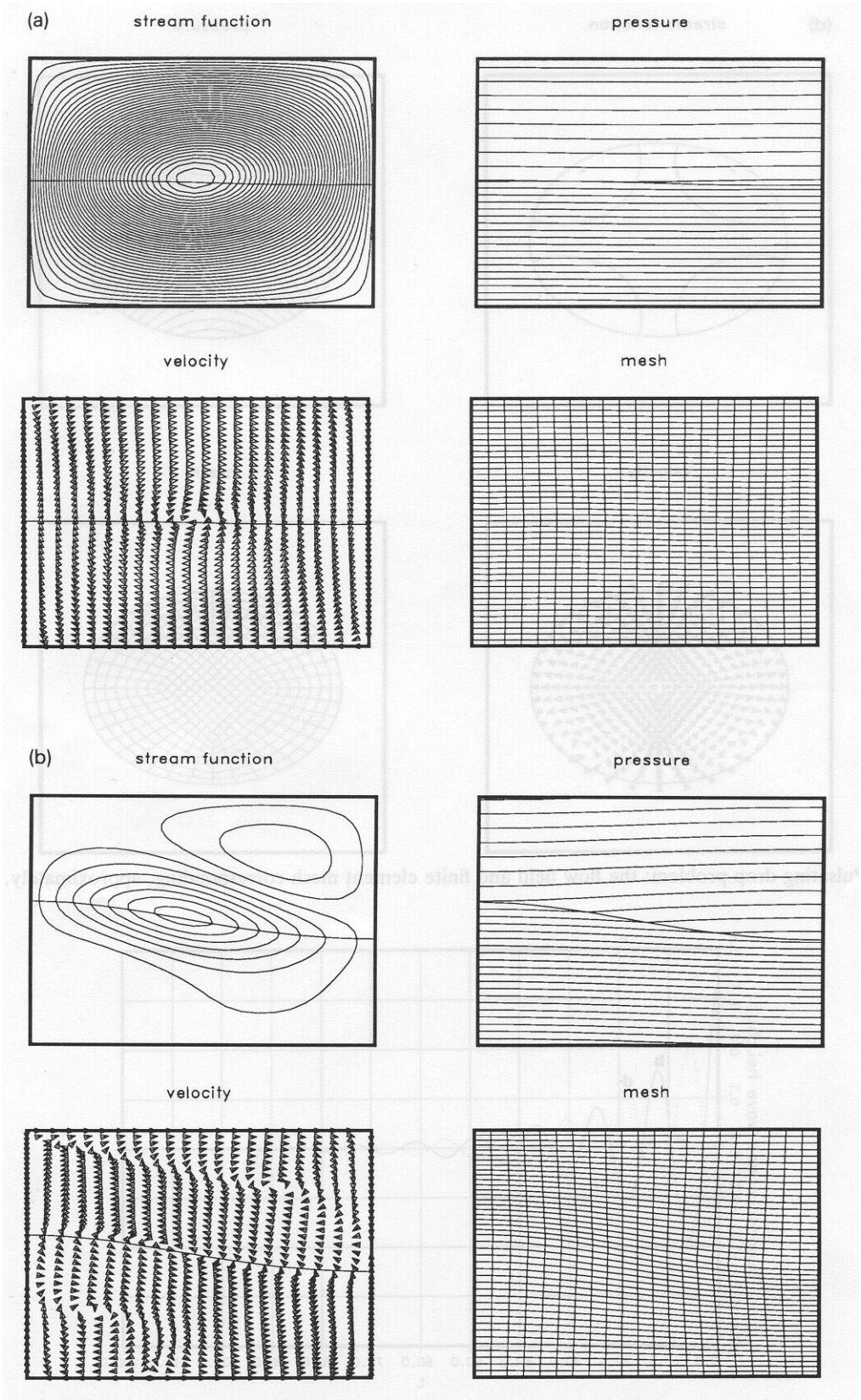

Figure 6. Two-liquid interface problem: the flow field and finite element mesh corresponding, approximately to (a) point a in Fig. 5; (b) point b in Fig. 5.

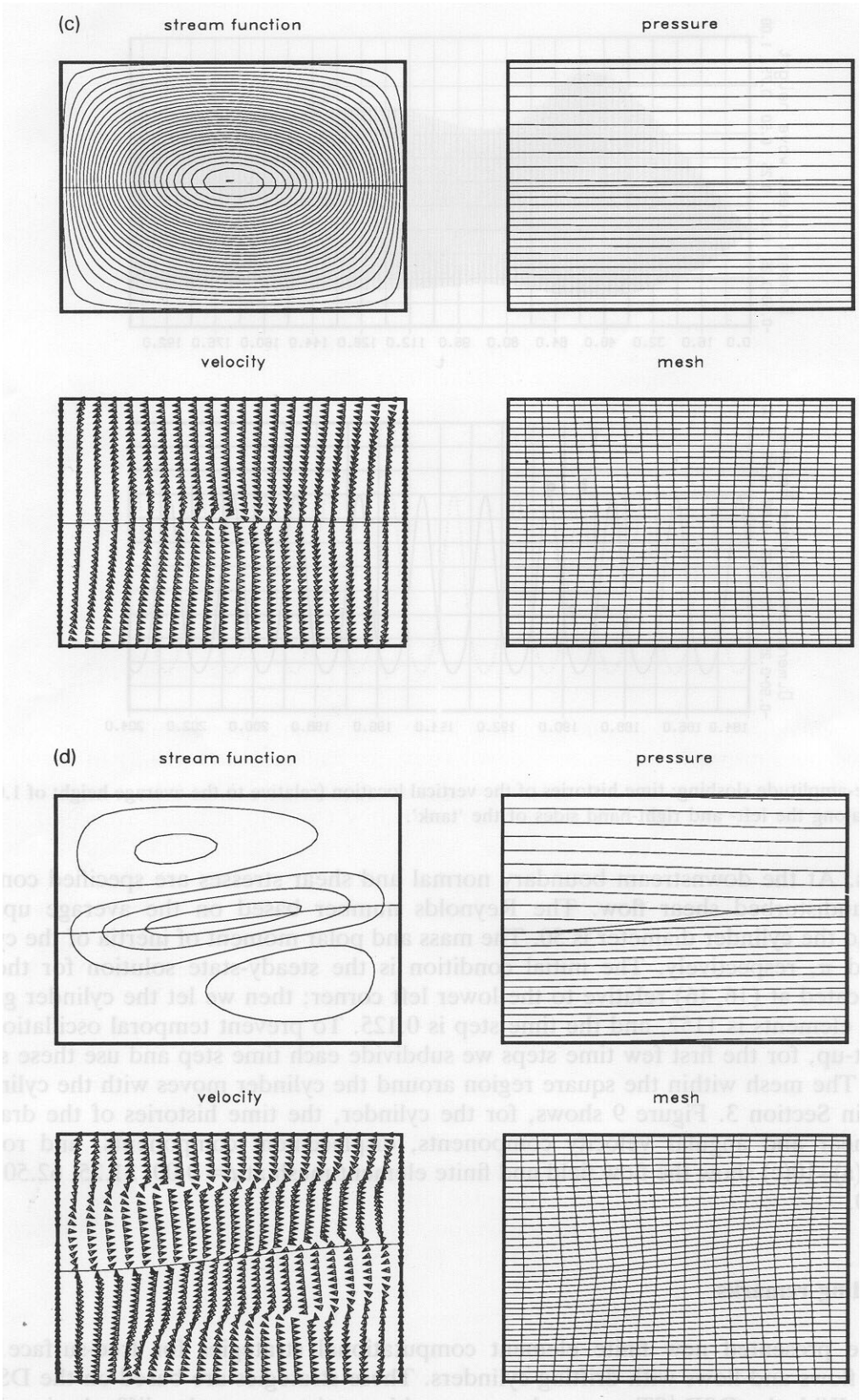

Figure 6. Two-liquid interface problem: the flow field and finite element mesh corresponding, approximately to (c) point c in Fig. 5; (d) point d in Fig. 5.

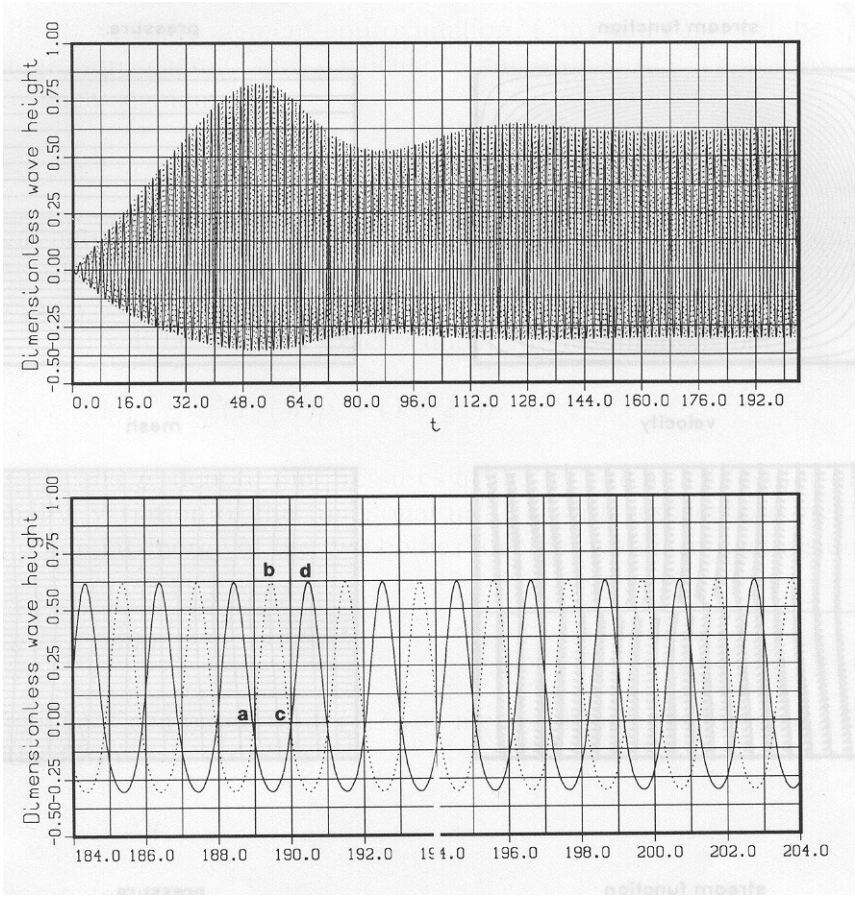

Figure 7. Large-amplitude sloshing: time histories of the vertical location (relative to the average height of 1.0) of the free-surface along the left- and right-hand sides of the "tank".

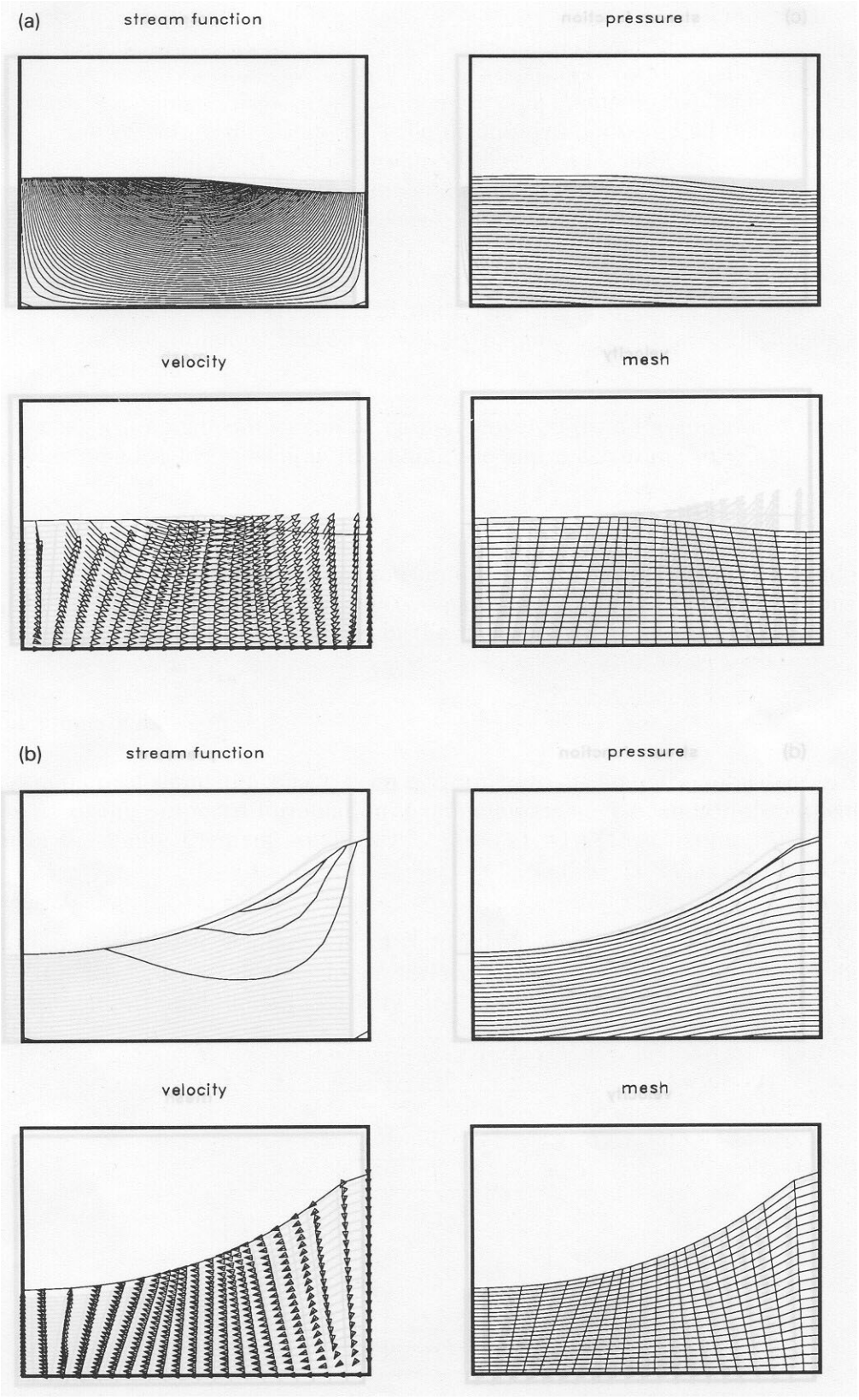

Figure 8. Large-amplitude sloshing: the flow field and finite element mesh corresponding, approximately, to (a) point a in Fig. 7; (b) point b in Fig. 7.

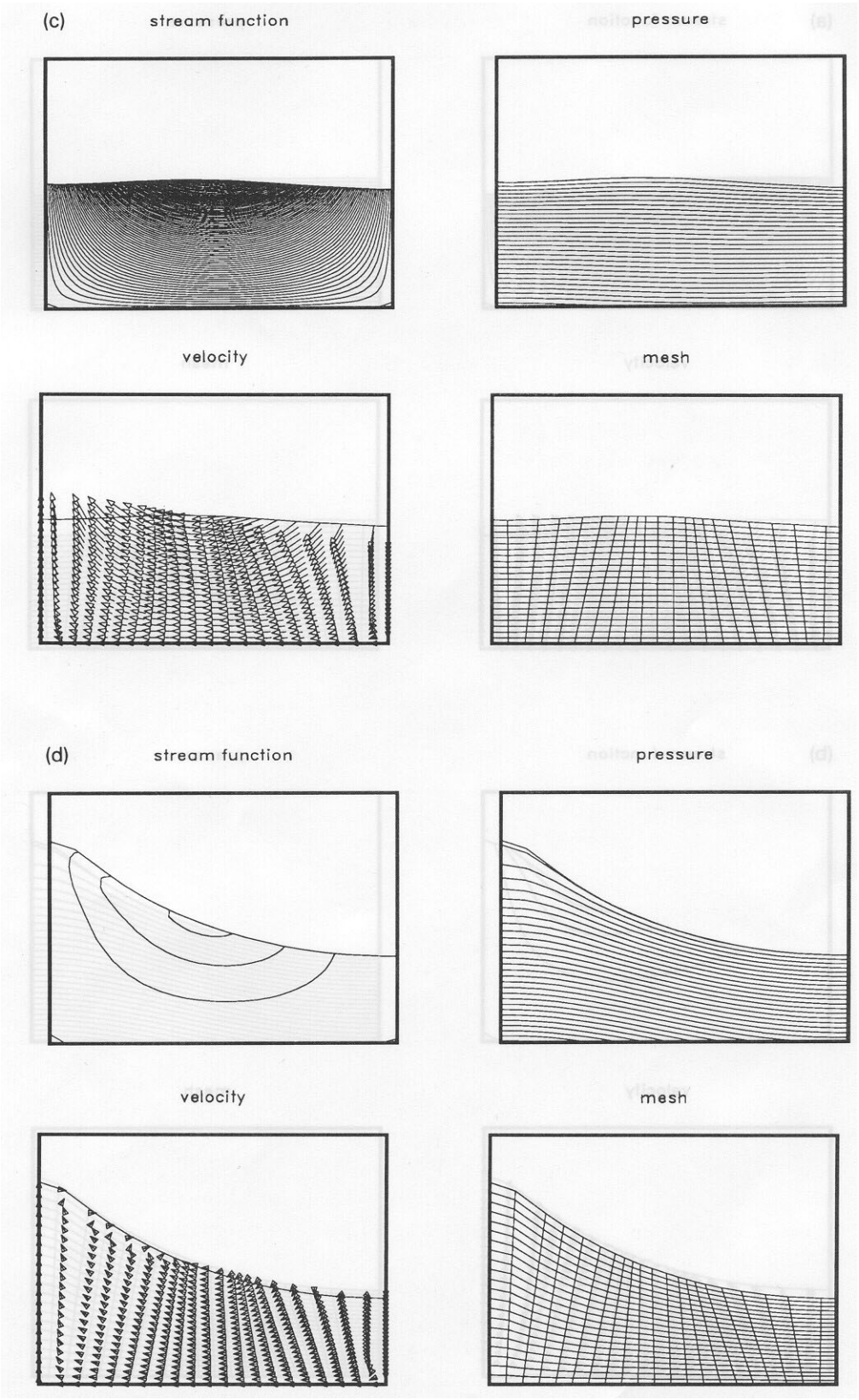

Figure 8. Large-amplitude sloshing: the flow field and finite element mesh corresponding, approximately, to (c) point c in Fig. 7; (d) point d in Fig. 7.

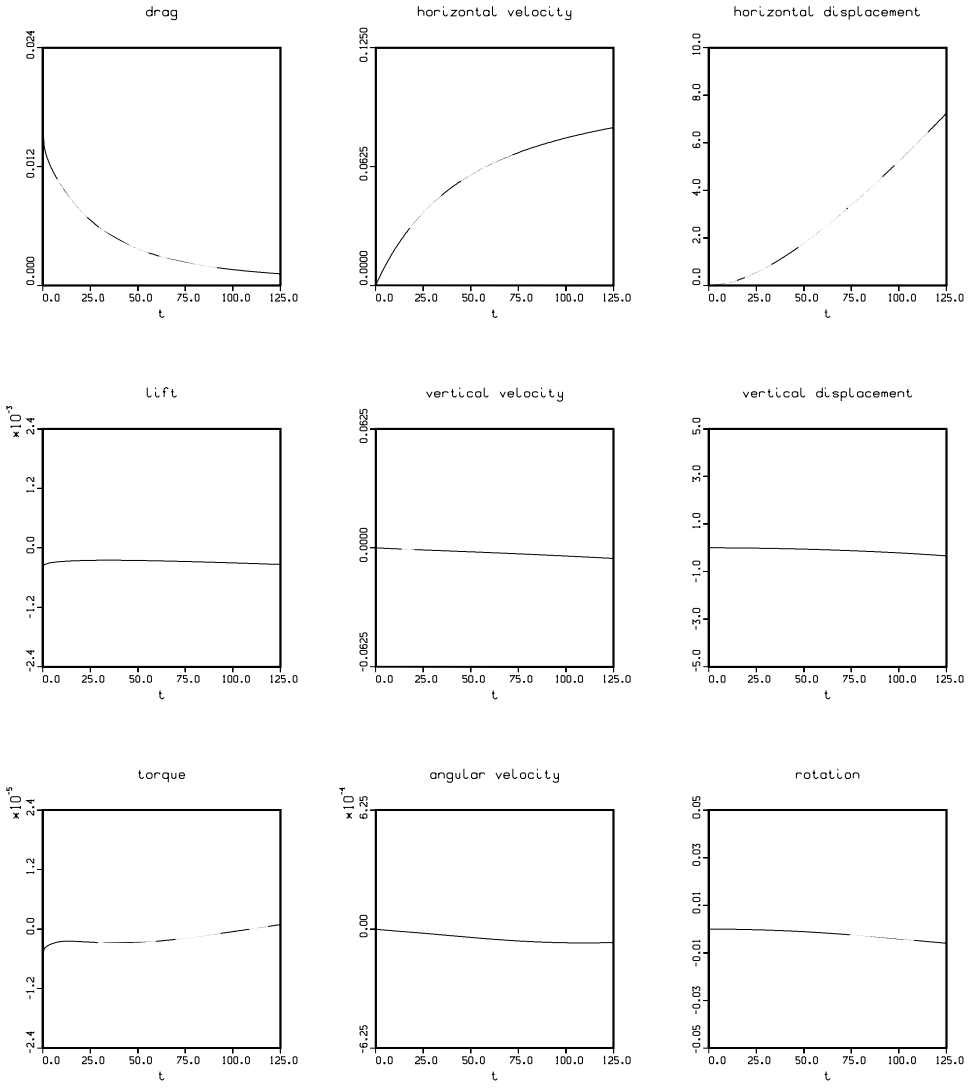

Figure 9. Cylinder drifting in shear flow: time history of the drag, lift, torque, linear and angular velocity components, displacement components, and rotation.

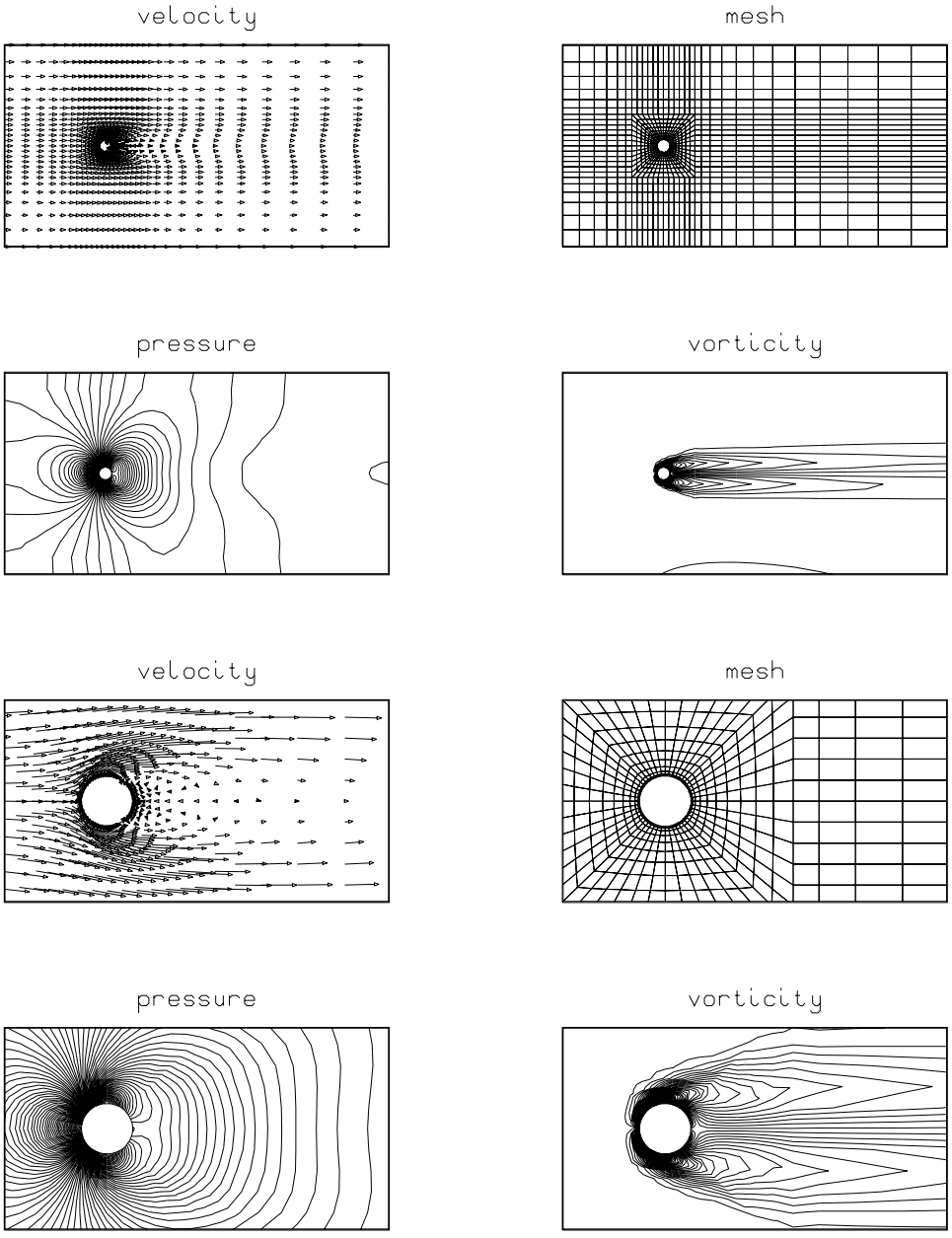

Figure 10. (a) Cylinder drifting in shear flow: the flow field and finite element mesh at $t = 0$.

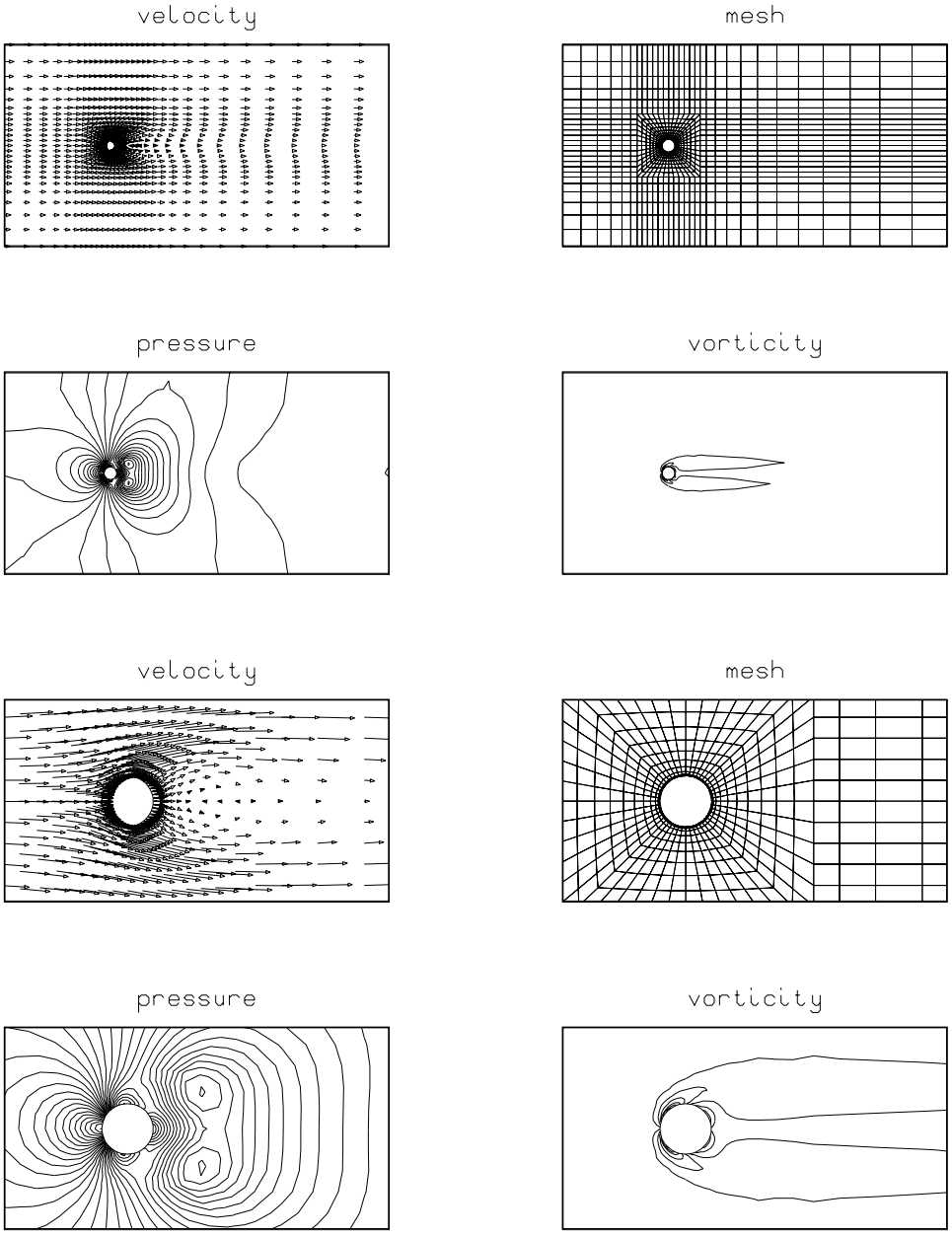

Figure 10. (b) Cylinder drifting in shear flow: the flow field and finite element mesh at $t = 31.25$.

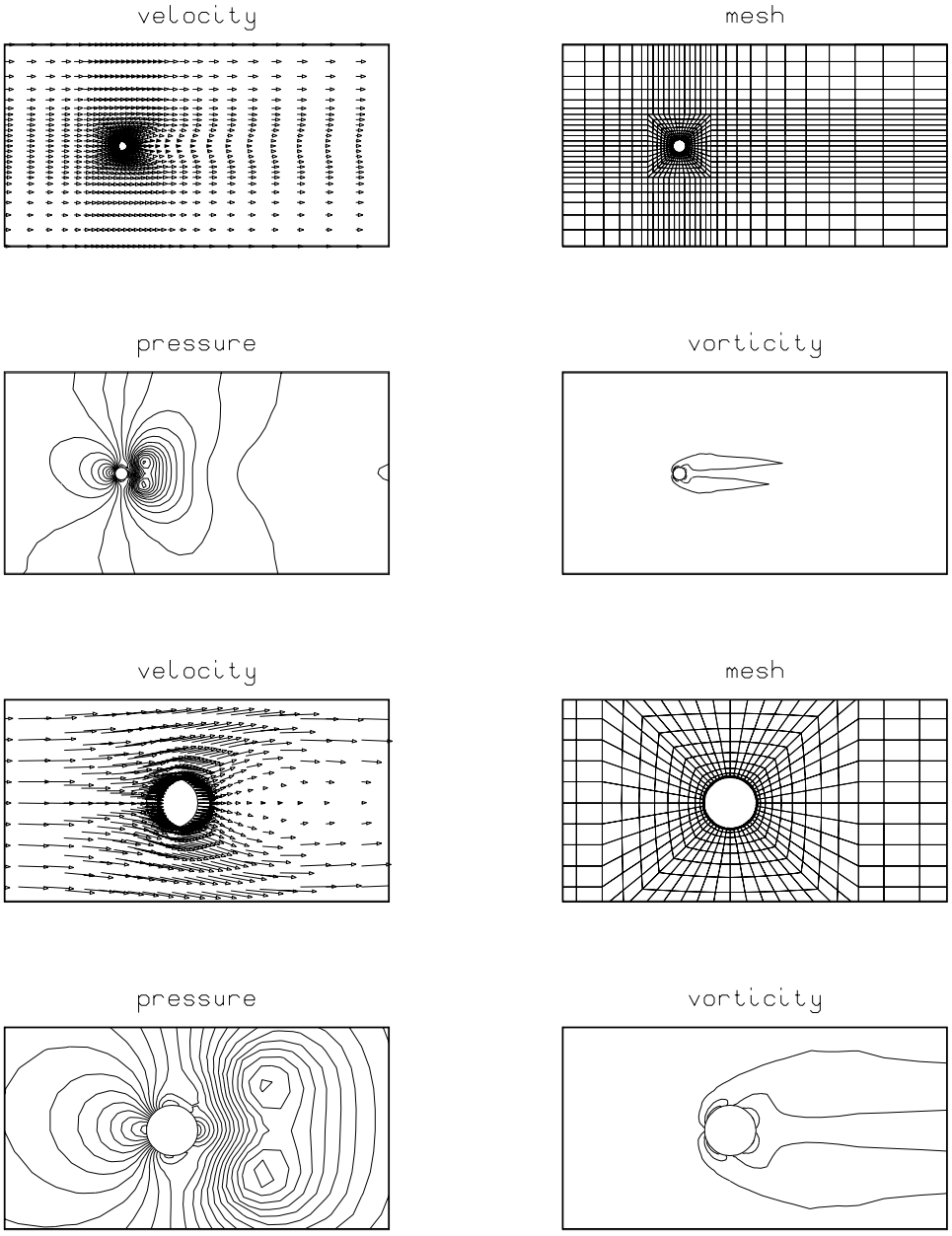

Figure 10. (c) Cylinder drifting in shear flow: the flow field and finite element mesh at $t = 62.50$.

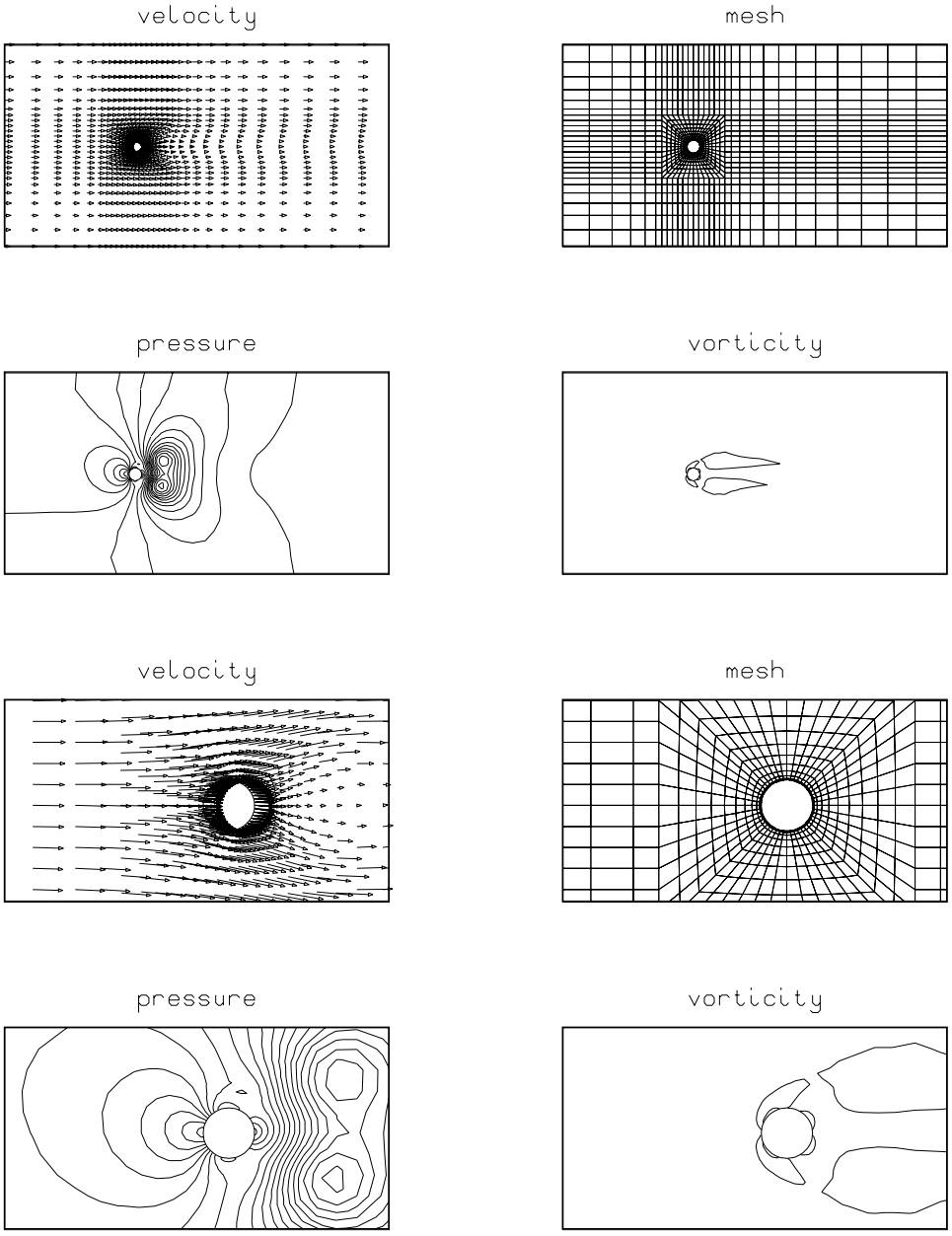

Figure 10. (d) Cylinder drifting in shear flow: the flow field and finite element mesh at $t = 93.75$.

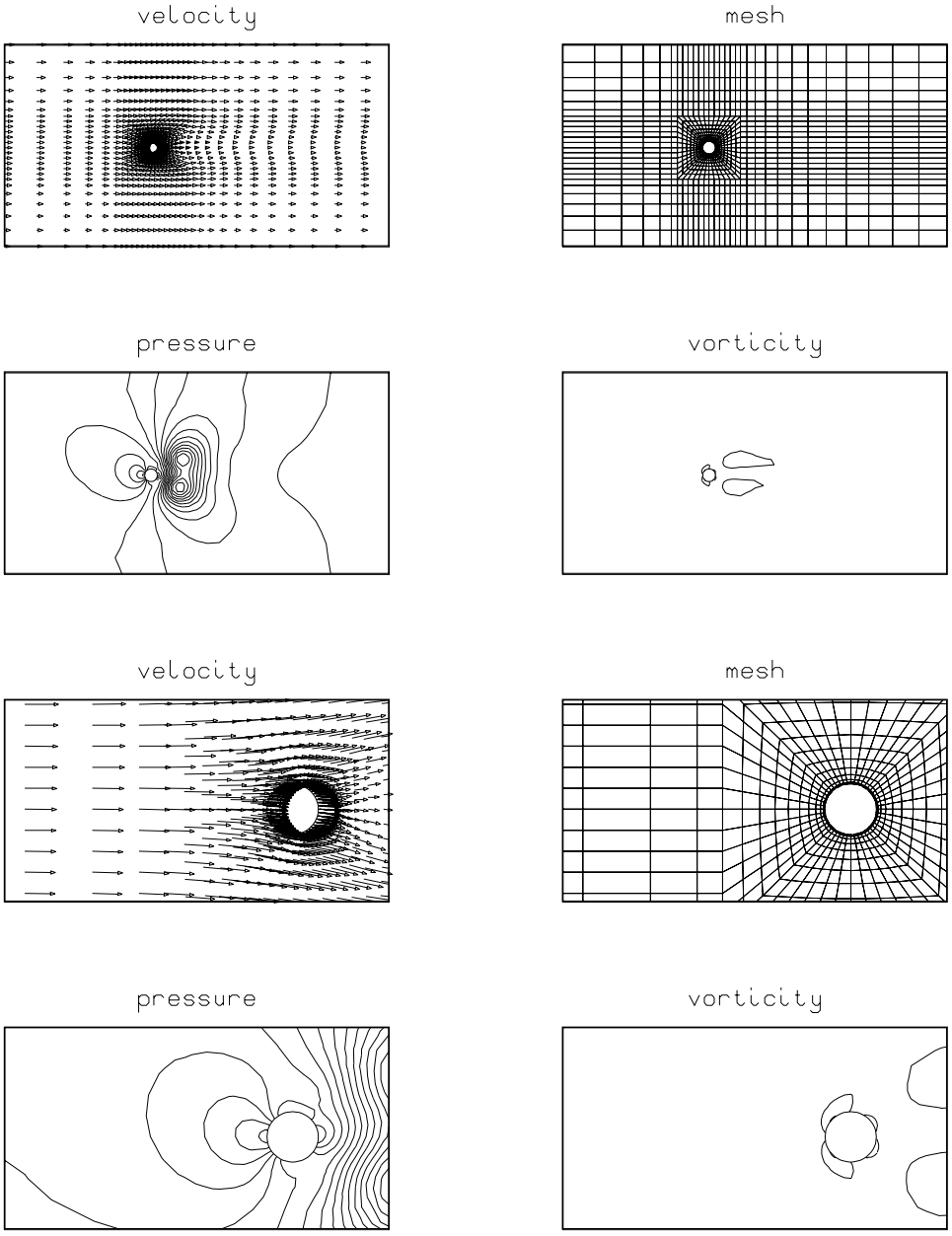

Figure 10. (e) Cylinder drifting in shear flow: the flow field and finite element mesh at $t = 125.00$.

5. Concluding Remarks

We have presented new finite element computational strategies for free-surface flows, two-liquid flows, and flows with drifting cylinders. These strategies are based on the DSD/ST procedure. With the DSD/ST approach, we are able to circumvent the difficulty involved in remeshing every time step. This way we can reduce the projection errors introduced by such frequent remeshings. We performed computations for various test problems mainly for the purpose of demonstrating the computational capability developed for this class of problems. Although we did not intend to conduct an in-depth physical investigation for any of the numerical examples, it is quite clear that the capability developed with the DSD/ST procedure provides us with the tools for such investigations. For flows involving drifting cylinders, we were able to show that the cylinder motion can be accommodated without the need for frequent remeshing.

Although for some of the numerical examples we were able to find previously published results to compare with ours, we still consider our computations to be preliminary. We showed that the DSD/ST procedure is potentially a very powerful tool; yet it would be prudent to regard it to be in its early stages and realize that more investigation needs to be carried out to use this tool best. For example, it is more economical to use "constant-in-time" interpolation functions; however, with moving meshes one has to be careful about how to interpret "constant-in-time". Furthermore, to make the space-time computations more economical, especially for three-dimensional problems, efficient iterative solution schemes need to be developed (see for example [10]).

Acknowledgements

We are thankful for the valuable comments from Professors R.L. Fosdick, T.J.R. Hughes, and O. Pironneau.

References

1. T.E. Tezduyar, M. Behr and J. Liou, 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, Comput. Meths. Appl. Mech. Engrg. 94 (1992) 339–351.

2. 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. Meths. Appl. Mech. Engrg. 63 (1987) 97–112.

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

4. T.J.R. Hughes, L.P. Franca and G.H. Hulbert, A new finite element formulation for

computational fluid dynamics: VIII. The Galerkin/least-squares method for advective-diffusive equations, Comput. Meths. Appl. Mech. Engrg. 73 (1989) 173–189.

5. F. Shakib, Finite element analysis of the compressible Euler and Navier-Stokes equations, Ph.D. thesis, Stanford University (1988).

6. P. Hansbo and A. Szepessy, A velocity-pressure streamline diffusion finite element method for the incompressible Navier-Stokes equations, Comput. Meths. Appl. Mech. Engrg. 84 (1990) 175–192.

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

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

9. A. Huerta and W.K. Liu, Viscous flow with large free surface motion, Comput. Meths. Appl. Mech. Engrg. 69 (1988) 277–324.

10. J. Liou and T.E. Tezduyar, Computation of compressible and incompressible flows with the clustered element-by-element method, University of Minnesota Supercomputer Institute Research Report, UMSI 90/215, October 1990.