ELSEVIER

Comput. Methods Appl. Mech. Engrg. 146 (1997) 91–126

Computer methods in applied mechanics and engineering

A space–time Galerkin/least-squares finite element formulation of the Navier–Stokes equations for moving domain problems

Arif Masud[a,*,1], Thomas J.R. Hughes[b,2]

[a]*Department of Civil and Materials Engineering, University of Illinois at Chicago, Chicago, IL 60607, USA*
[b]*Division of Applied Mechanics, Durand Building, Stanford University, Stanford, CA 94305, USA*

Received 24 June 1996

Abstract

A space–time Galerkin/least-squares finite element formulation of the Navier–Stokes equations is presented for the analysis of free surface flows, moving spatial configurations and deforming fluid-structure interfaces. The variational equation is based on the time discontinuous Galerkin method employing the physical entropy variables. The space–time elements are oriented in time to accommodate the spatial deformations. If the elements are oriented along the particle paths, the formulation is Lagrangian and if they are fixed in time, it is Eulerian. Consequently this formulation is analogous to the arbitrary Lagrangian–Eulerian (ALE) technique. A novel mesh rezoning strategy is presented to orient the elements in time and adapt the fluid mesh to the changing spatial configuration. Numerical results are presented to show the performance of the method.

1. Introduction

Fluid flow problems that involve moving and deforming spatial configurations have been an area of active interest. Over the years both Lagrangian and Eulerian viewpoints have been formulated, each however with serious limitations. The Lagrangian description provides a precise definition of moving boundaries and is devoid of convective effects. Its major drawback is that it does not handle the material distortions very well and can lead to highly contorted mesh configurations. The Eulerian viewpoint on the other hand allows strong distortions in the fluid motion. It, however, suffers from two important drawbacks, namely, the presence of convective effects that arise due to relative movement between nodal points and fluid particles, and secondly, complex mathematical mappings between stationary and moving boundaries are required. In order to deal with time dependent fluid flow problems with changing spatial configurations in a generalized setting arbitrary Lagrangian–Eulerian (ALE) finite element techniques have so far been used (for background see [5,9–12,28,43] and references therein).

In a relatively recent development, space–time Galerkin/least-squares finite element formulations with fixed spatial domains have been developed by Hughes et al. [24,26,29,38,42] and Johnson et al. [18,31,32] for fluid and solid mechanics problems. The variational foundations of these formulations employ the time-discontinuous Galerkin method and include least-squares and discontinuity-capturing

* Corresponding author.
[1] Assistant Professor of Mechanics and Materials. Formerly, Graduate Research Assistant, Stanford University.
[2] Professor of Mechanical Engineering.

PII S0045-7825(96)01222-4

operators. The conceptual framework of the underlying Galerkin/least-squares method has evolved as a generalization of the streamline-upwind/Petrov–Galerkin method (SUPG), developed earlier by Hughes et al. [7,19,21] for convective transport problems. The added terms involve *residuals of the Euler–Lagrange equations* evaluated over element interiors. Consequently, these formulations preserve the consistency of the Galerkin method which is an important ingredient in obtaining improved convergence rates with higher-order interpolation, and also respect the regularity requirements on the functions employed. In the Galerkin/least-squares framework, general combinations of the interpolation functions (which otherwise violate the critical stability conditions in the Galerkin framework) become convergent [22,23] either by circumventing the stability conditions or by satisfying them [14].

In this paper we have extended the idea of the space–time Galerkin/least-squares finite element formulation of the Navier–Stokes equations to computations that involve changing spatial configurations. The basis of our formulation is a time-discontinuous Galerkin method. Due to the discontinuity of finite element functions at the space–time slab interfaces the spatial discretization can be changed from one slab to another. This provides a natural mechanism for incorporating adaptive remeshing in the formulation. In similar efforts based on the space–time finite element concept, Tezduyar et al. [2,3,4,39,44–47] have developed a deforming domain strategy and Hansbo [17] has developed a characteristic streamline diffusion method for the compressible and the incompressible Navier–Stokes equations.

An outline of the paper is as follows. Section 2 presents a modified equation of state which incorporates the bulk elastic response of a liquid and thus yields a slightly compressible form of the Navier–Stokes equations. The strong form of the initial/boundary value problem is presented in Section 3. Section 4 discusses the kinematics of the moving space–time slabs and exhibits the relationship with the classical arbitrary Lagrangian–Eulerian technique. The discretized weighted residual formulation emanating from the strong form of the problem is presented next. Section 6 presents a linear-in-time approximation of the variational equation which is appropriate for the class of problems considered here. Section 7 presents an arbitrary mesh rezoning strategy which successfully exploits the idea of arbitrary orientation in time of the space–time elements, thereby accommodating the spatial deformations. The issue of boundary and interface conditions is addressed in Section 8. Various numerical examples are presented in Section 9, and conclusions are drawn in Section 10.

2. A slightly compressible form of the Navier–Stokes equations

In terms of conservation variables, the compressible Navier–Stokes equations can be written as

$$\boldsymbol{U}_{,t} + \boldsymbol{F}_{i,i} = \boldsymbol{F}^{d}_{i,i} + \mathscr{F} \tag{2.1}$$

where $\boldsymbol{U}$ is the vector of conservation variables, $\boldsymbol{F}_i$ and $\boldsymbol{F}^d_i$ are, respectively, the vectors of advective and diffusive fluxes in the ith direction, $i = 1, \ldots, n_{sd}$, n_{sd} is the number of space dimensions, and $\mathscr{F}$ is the source vector. In three dimensions these can be explicitly expressed as

$$\boldsymbol{U} = \rho \begin{Bmatrix} 1 \\ \boldsymbol{v} \\ e^{\text{tot}} \end{Bmatrix} \qquad \text{(conservation variables)} \tag{2.2}$$

$$\boldsymbol{F}_i = \rho v_i \begin{Bmatrix} 1 \\ \boldsymbol{v} \\ e^{\text{tot}} \end{Bmatrix} + p \begin{Bmatrix} 0 \\ \boldsymbol{\delta}_i \\ v_i \end{Bmatrix} \qquad \text{(Euler flux)} \tag{2.3}$$

$$\boldsymbol{F}^{d}_i = \begin{Bmatrix} 0 \\ \boldsymbol{\tau}_i \\ \tau_{ij} v_j \end{Bmatrix} + \begin{Bmatrix} 0 \\ \boldsymbol{0} \\ -q_i \end{Bmatrix} \qquad \text{(diffusive flux)} \tag{2.4}$$

$$\mathscr{F} = \rho \begin{Bmatrix} 0 \\ \boldsymbol{b} \\ b_i v_i + r \end{Bmatrix} \qquad \text{(source vector)} \tag{2.5}$$

The five equations in (2.1) represent the conservation of mass, momentum and energy. In these relations the following notations have been used: ρ is the density; $\boldsymbol{v} = \{v_i\}$ is the velocity vector; e^{tot} is the total energy density; p is the thermodynamic pressure; $\boldsymbol{\delta} = [\delta_{ij}]$ is the Kronecker delta; $\boldsymbol{\tau} = [\tau_{ij}]$ is the viscous-stress tensor; $\boldsymbol{\delta}_i = \boldsymbol{\delta}\boldsymbol{e}_i$ and $\boldsymbol{\tau}_i = \boldsymbol{\tau}\boldsymbol{e}_i$ where $\boldsymbol{e}_i$ is the unit basis vector in the ith direction; $\boldsymbol{q} = \{q_i\}$ is the heat-flux vector; $\boldsymbol{b} = \{b_i\}$ is the body force vector per unit mass; r is the heat supply per unit mass; and the summation convention is assumed throughout. The total energy density is the sum of the internal energy density, e, and the kinetic energy density, i.e.

$$e^{\text{tot}} = e + \frac{1}{2}|\boldsymbol{v}|^2 \tag{2.6}$$

2.1. The equation of state

The equation of state for a fluid typically relates pressure to density and temperature. For the class of problems considered in this work we assume temperature changes to be negligible. Accordingly, we expand p as follows:

$$p = p_{\text{ref}} + \left(\frac{\partial p}{\partial \rho}\right)_{\rho_{\text{ref}}}(\rho - \rho_{\text{ref}}) + \frac{1}{2}\left(\frac{\partial^2 p}{\partial \rho^2}\right)_{\rho_{\text{ref}}}(\rho - \rho_{\text{ref}})^2 + \mathrm{O}((\rho - \rho_{\text{ref}})^3) \tag{2.7}$$

where p_{ref} and ρ_{ref} are constant reference values of pressure and density. If the changes in density are small, which is assumed to be the case, higher-order terms can be dropped. Therefore

$$p - p_{\text{ref}} \approx \left(\frac{\partial p}{\partial \rho}\right)_{\rho_{\text{ref}}}(\rho - \rho_{\text{ref}}) \tag{2.8}$$

This leads to an equation of state for a slightly compressible fluid in terms of the bulk modulus β, defined as $\beta = \rho_{\text{ref}}(\partial p/\partial \rho)_{\rho_{\text{ref}}}$, viz.

$$p - p_{\text{ref}} = \frac{\beta}{\rho_{\text{ref}}}(\rho - \rho_{\text{ref}}) \tag{2.9}$$

The additional constitutive relations are

$$\tau_{ij} = \lambda u_{k,k}\delta_{ij} + \mu(u_{i,j} + u_{j,i}) \tag{2.10}$$

$$q_i = -\kappa T_{,i} \tag{2.11}$$

$$e = c_{\text{v}} T \tag{2.12}$$

where λ and μ are the viscosity coefficients, κ is the thermal conductivity, c_{v} is the specific heat at constant volume and T is the absolute temperature. Eq. (2.10) defines the viscous stress components and (2.11) is *Fourier's law* of heat conduction.

REMARK. The ideal gas relation can be written in a differential form as

$$\mathrm{d}p = RT\,\mathrm{d}\rho + R\rho\,\mathrm{d}T \tag{2.13}$$

Assuming that the temperature gradients are insignificantly small, the second term on the right-hand side of (2.13) can be dropped and the resulting equation can be written as

$$(p - p_{\text{ref}}) = RT_{\text{ref}}(\rho - \rho_{\text{ref}}) \tag{2.14}$$

where T_{ref} is the reference temperature. Comparing (2.9) and (2.14) we obtain a modified gas constant R which is defined as

$$R = \frac{\beta}{T_{\text{ref}}\rho_{\text{ref}}} \tag{2.15}$$

The specific heat at constant volume c_{v}, and the specific heat at constant pressure c_{p} are defined as

$$c_v = \frac{R}{(\gamma - 1)}, \qquad c_p = \frac{\gamma R}{(\gamma - 1)} \tag{2.16}$$

where γ is the ratio of specific heats, i.e. $\gamma = c_p/c_v$. Substituting (2.15) in the above expressions we get the modified values of c_v and c_p. Similarly, the Prandtl number is defined as $P_r = (\mu c_p)/\kappa$ which yields a modified coefficient of thermal conductivity

$$\kappa = \frac{\mu}{P_r} \frac{\gamma R}{(\gamma - 1)} \tag{2.17}$$

Substituting (2.16) and (2.17) in the corresponding constitutive relations introduces a factor which is proportional to the bulk modulus of water in the equations.

We can write the system of equations in a quasi-linear form

$$\boldsymbol{U}_{,t} + \boldsymbol{A}_i \boldsymbol{U}_{,i} = (\boldsymbol{K}_{ij} \boldsymbol{U}_{,j})_{,i} + \mathscr{F}(\boldsymbol{U}) \tag{2.18}$$

where $\boldsymbol{A}_i = \boldsymbol{F}_{i,U}$ is the ith Euler Jacobian matrix; and $\boldsymbol{K} = [\boldsymbol{K}_{ij}]$ is the diffusivity matrix, with the $\boldsymbol{K}_{ij}$'s satisfying $\boldsymbol{K}_{ij} \boldsymbol{U}_{,j} = \boldsymbol{F}_i^d$. Eq. (2.18) generates a non-symmetric system of partial differential equations, i.e. $\boldsymbol{A}_i$ is not symmetric and $\boldsymbol{K}$ is neither symmetric nor positive-definite. The Galerkin formulation of this form of the Navier–Stokes equations lacks certain properties which are needed to establish stability proofs and convergence analyses. In terms of entropy variables $\boldsymbol{V}$, as discussed in detail in [25,36,42,8], Eq. (2.1) is symmetrized via a change of variables to the following form:

$$\tilde{\boldsymbol{A}}_0 \boldsymbol{V}_{,t} + \tilde{\boldsymbol{A}}_i \boldsymbol{V}_{,i} = (\tilde{\boldsymbol{K}}_{ij} \boldsymbol{V}_{,j})_{,i} + \tilde{\mathscr{F}}(\boldsymbol{V}) \tag{2.19}$$

where $\boldsymbol{V}$ is referred to as the vector of *physical entropy variables*, and

$$\tilde{\boldsymbol{A}}_0 = \boldsymbol{U}_{,V} \tag{2.20}$$

$$\tilde{\boldsymbol{A}}_i = \boldsymbol{A}_i \tilde{\boldsymbol{A}}_0 \tag{2.21}$$

$$\tilde{\boldsymbol{K}}_{ij} = \boldsymbol{K}_{ij} \tilde{\boldsymbol{A}}_0 \tag{2.22}$$

Eqs. (2.20)–(2.22) represent the Riemannian metric tensor $\tilde{\boldsymbol{A}}_0$, the Jacobians of the Euler fluxes $\tilde{\boldsymbol{A}}_i$ and the diffusivity-coefficient matrices $\tilde{\boldsymbol{K}}_{ij}$. Explicit definition of the flux vectors and coefficient matrices in terms of $\boldsymbol{V}$ variables can be found in [25].

3. Strong form of the initial/boundary-value problem

Let Ω_t be an open, bounded domain in $R^{n_{sd}}$, where n_{sd} is the number of space dimensions. The closure of Ω_t is $\bar{\Omega}_t$ and the boundary of Ω_t is denoted by Γ_t. It is important to note that in the present case the geometry of the spatial domain and the spatial boundary are also time-dependent. The unit outward normal vector to Γ_t is denoted by $\boldsymbol{n} = \{n_i\}$. We also assume that Γ_t admits the following decomposition:

$$\overline{\Gamma_{g_t} \cup \Gamma_{h_t}} = \Gamma_t \tag{3.1}$$

and

$$\Gamma_{g_t} \cap \Gamma_{h_t} = \emptyset \tag{3.2}$$

where Γ_{g_t} and Γ_{h_t} decompose the boundary Γ_t into regions where essential and natural boundary conditions, respectively, are prescribed. All data are assumed to be functions of space ($\boldsymbol{x} \in \bar{\Omega}_t$) and time ($t \in]0, T[$).

Let $\mathscr{L}$ be the differential operator for the Navier–Stokes equations

$$\mathscr{L} = \tilde{\boldsymbol{A}}_0 \frac{\partial}{\partial t} + \tilde{\boldsymbol{A}}_i \frac{\partial}{\partial x_i} - \frac{\partial}{\partial x_i}\left(\tilde{\boldsymbol{K}}_{ij} \frac{\partial}{\partial x_j}\right) \tag{3.3}$$

The formal statement of the initial/boundary-value problem under consideration is: Given $\boldsymbol{V}_0 : \Omega_0 \to R^{n_{\text{dof}}}$, $\boldsymbol{g} : \Gamma_g \times]0, T[\to R^{n_{\text{dof}}}$ and $\boldsymbol{h} : \Gamma_h \times]0, T[\to R^{n_{\text{dof}}}$, find $\boldsymbol{V} : \Omega \times [0, T] \to R^{n_{\text{dof}}}$ such that $\forall \boldsymbol{x} \in \Omega_t, t \in]0, T[$

$$\mathscr{L}\boldsymbol{V} - \tilde{\mathscr{F}} = \boldsymbol{0} \qquad \text{on } \Omega \times]0, T[\tag{3.4}$$

$$\boldsymbol{V}(\boldsymbol{x}, 0) = \boldsymbol{V}_0(\boldsymbol{x}) \qquad \forall \boldsymbol{x} \in \Omega_0 \tag{3.5}$$

$$\boldsymbol{q}(\boldsymbol{V}) = \boldsymbol{g} \qquad \text{on } \Gamma_g \times]0, T[\tag{3.6}$$

$$(-\boldsymbol{F}_i + \boldsymbol{F}_i^d) n_i = \boldsymbol{h} \quad \text{on } \Gamma_h \times]0, T[\tag{3.7}$$

where $n_{\text{dof}} = n_{\text{sd}} + 2$ is the number of degrees of freedom, $\boldsymbol{q}$ is a nonlinear boundary condition function, and $\boldsymbol{g}$ and $\boldsymbol{h}$ are the vectors of prescribed essential and natural boundary conditions, respectively.

In practice, it is convenient to implement the boundary and initial conditions in terms of primitive variables. Transformation of these prescribed conditions to the physical entropy variables results in nonlinear relations in $\boldsymbol{V}$ as evidenced by the function $\boldsymbol{q}(\boldsymbol{V})$. Various techniques to handle these nonlinearities and a consistent method to calculate fluxes at the boundaries are discussed in [42].

4. A space–time description of the moving domains

Let $I = [0, T[$ be an open time interval partitioned by an ordered series of time levels $0 = t_0 < t_1 < \cdots < t_N = T$. Denoting the nth time interval $I_n =]t_n, t_{n+1}[$, we have $I = \cup_{n=0}^{N-1} I_n \cup \{t_1, t_2, \ldots, t_{N-1}\}$.

Let Ω_n and Γ_n be the approximations to the n_{sd}-dimensional spatial domain Ω with boundary Γ at time level t_n. Similarly Ω_{n+1} and Γ_{n+1} are the approximations at time level t_{n+1}, respectively (Fig. 1). A *space–time 'slab'* Q_n in the context of moving domains is then defined as the region enclosed between Ω_n, Ω_{n+1} and P_n, where P_n is the surface described by the boundary Γ_t as t traverses I_n. P_n is also assumed to admit the following decomposition:

$$\overline{P_{g_n} \cup P_{h_n}} = P_n \tag{4.1}$$

and

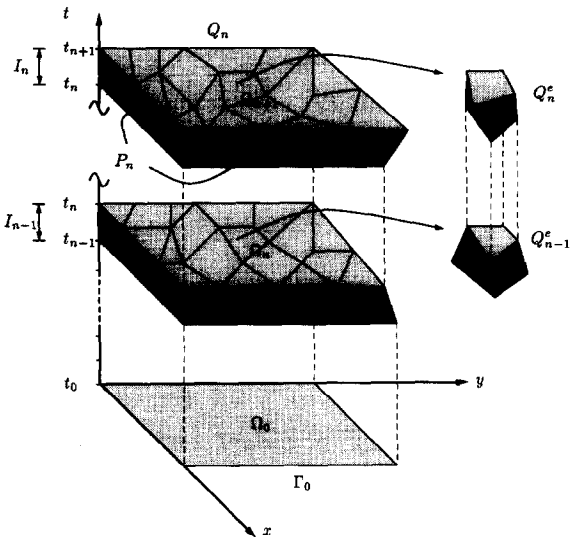
$$P_{g_n} \cap P_{h_n} = \emptyset \tag{4.2}$$

For the nth space–time slab, let the spatial domain be subdivided into $(n_{\text{el}})_n$ elements Ω_n^e, $e = 1, \ldots, (n_{el})_n$. Then, for the nth slab, we define a space–time element domain Q_n^e as the region enclosed between Ω_n^e, Ω_{n+1}^e and the space–time element 'lateral' boundary P_n^e which may or may not be a part of the slab boundary P_n.

Fig. 1 shows motion of the mesh during two consecutive time intervals. The displacement field of the spatial mesh is assumed continuous in time. Consequently, a space–time slab deforms during a time-step and the initial configuration of the spatial domain in a slab is identical to the final configuration in the previous slab. This precludes the need for a projection of the solution from the previous mesh onto the new mesh, which would otherwise be required if we were using non-matching meshes at slab interfaces.

We now introduce a *reference space–time domain* $\tilde{Q}_n \subset R^{n_{\text{sd}}} \times]0, T[$ and a mapping $\Phi_n : \tilde{Q}_n \to Q_n$ from the reference space–time element in the $(\boldsymbol{\xi}, t)$ coordinate system on to the deformed physical element in the $(\boldsymbol{x}, t)$ coordinate system (Fig. 2) [17]. We assume the mapping Φ_n is sufficiently smooth, orientation preserving and invertible. The motion of $\tilde{Q}_n$ is a time-dependent family of configurations, written as

$$(\boldsymbol{x}, t) = \Phi_n(\boldsymbol{\xi}, t) \tag{4.3}$$



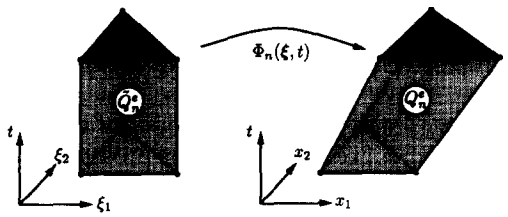


Fig. 1. Two adjacent space–time slabs for the moving domain.

Fig. 2. Transformation between reference and spatial domains.

Within each space–time element, the trial solutions and the weighting functions are approximated by kth-order interpolatory polynomials, $\mathscr{P}_k$. These functions are assumed to be H^1 functions and C^0 continuous within each space–time slab, but are discontinuous across the interfaces of the slabs, namely at times $t_1, t_2, \ldots, t_{N-1}$. The finite element spaces for the trial and weighting functions for the space–time domain Q_n are defined as follows:

Trial functions in $(\boldsymbol{x}, t)$ space:

$$\mathscr{S}_n^h = \left\{ \boldsymbol{V}^h \mid \boldsymbol{V}^h \in (C^0(Q_n))^{n_{\text{dof}}}, \boldsymbol{V}^h|_{Q_n^e} \in (\mathscr{P}_k(Q_n^e))^{n_{\text{dof}}}, \boldsymbol{q}(\boldsymbol{V}^h) = \boldsymbol{g}(t) \text{ on } P_{g_n} \right\} \tag{4.4}$$

Weighting functions in $(\boldsymbol{x}, t)$ space:

$$\mathscr{V}_n^h = \left\{ \boldsymbol{W}^h \mid \boldsymbol{W}^h \in (C^0(Q_n))^{n_{\text{dof}}}, \boldsymbol{W}^h|_{Q_n^e} \in (\mathscr{P}_k(Q_n^e))^{n_{\text{dof}}}, (\partial \boldsymbol{q}/\partial \boldsymbol{V}^h)(\boldsymbol{V}^h)\boldsymbol{W}^h = \boldsymbol{0} \text{ on } P_{g_n} \right\} \tag{4.5}$$

See [42] for further details.

The functions $\boldsymbol{V}^h(\boldsymbol{x}, t)$ and $\boldsymbol{W}^h(\boldsymbol{x}, t)$ are related to functions $\tilde{\boldsymbol{V}}^h(\boldsymbol{\xi}, t)$ and $\tilde{\boldsymbol{W}}^h(\boldsymbol{\xi}, t)$ by means of (4.3), thus defining the trial and weighting function spaces in the reference space–time domain $\tilde{Q}_n$ in the $(\boldsymbol{\xi}, t)$ coordinate system.

Trial functions in $(\boldsymbol{\xi}, t)$ space:

$$\tilde{\mathscr{S}}_n^h = \left\{ \tilde{\boldsymbol{V}}^h \mid \tilde{\boldsymbol{V}}^h(\boldsymbol{\xi}, t) = \boldsymbol{V}^h(\boldsymbol{x}, t), \boldsymbol{V}^h \in \mathscr{S}_h^h, (\boldsymbol{x}, t) = \boldsymbol{\Phi}_n(\boldsymbol{\xi}, t) \right\} \tag{4.6}$$

Weighting functions in $(\boldsymbol{\xi}, t)$ space:

$$\tilde{\mathscr{V}}_n^h = \left\{ \tilde{\boldsymbol{W}}^h \mid \tilde{\boldsymbol{W}}^h(\boldsymbol{\xi}, t) = \boldsymbol{W}^h(\boldsymbol{x}, t), \boldsymbol{W}^h \in \mathscr{V}_n^h, (\boldsymbol{x}, t) = \boldsymbol{\Phi}_n(\boldsymbol{\xi}, t) \right\} \tag{4.7}$$

where $\tilde{\boldsymbol{V}}^h(\boldsymbol{\xi}, t)$ and $\tilde{\boldsymbol{W}}^h(\boldsymbol{\xi}, t)$ inherit the essential boundary conditions by way of (4.4) and (4.5).

4.1. Kinematics of the space–time slabs

Our objective in this section is to show that the conservation equations written in the *spatial space–time domain* Q_n yield the classical arbitrary Lagrangian–Eulerian form when mapped on to the

reference space–time domain $\tilde{Q}_n$. We introduced the mapping $\Phi_n: \tilde{Q}_n \to Q_n$ in (4.3). For the space–time domains where time is an additional dimension, the deformation gradient generates a $(n_{sd}+1)\times(n_{sd}+1)$ matrix of partial derivatives of the transformation Φ_n. For clarity of presentation we refer to this matrix as the space–time deformation gradient f_n, as opposed to the classical spatial deformation gradient $\boldsymbol{F}$.

The space–time deformation gradient at time t_n can be written as

$$f_n = \begin{bmatrix} \boldsymbol{F} & \boldsymbol{v}_n^{\mathrm{m}} \\ \boldsymbol{0}^{\mathrm{T}} & 1 \end{bmatrix} \tag{4.8}$$

where $\boldsymbol{v}_n^{\mathrm{m}}$ denotes the fluid mesh velocity field in slab n. The mapping is assumed to be sufficiently smooth and invertible, therefore f_n^{-1} exists and is unique

$$f_n^{-1} = \begin{bmatrix} \boldsymbol{F}^{-1} & -\boldsymbol{F}^{-1}\boldsymbol{v}_n^{\mathrm{m}} \\ \boldsymbol{0}^{\mathrm{T}} & 1 \end{bmatrix} \tag{4.9}$$

The Jacobian determinant of the linear transformation is denoted by $\tilde{J}(\boldsymbol{\xi}, t)$. Since Φ_n is assumed to preserve orientation, $\tilde{J} > 0$. It is important to note that

$$\tilde{J} \overset{\text{def.}}{=} \det(f_n) = \det(\boldsymbol{F}) \overset{\text{def.}}{=} J$$

The time rate of change of Jacobian (see [37, p. 86]) is given by

$$\frac{\partial \tilde{J}(\boldsymbol{\xi}, t)}{\partial t} = \tilde{J}\, \mathrm{div}_{\boldsymbol{x}}\, \boldsymbol{v}_n^{\mathrm{m}}\Big|_{(\boldsymbol{x},t)=\Phi_n(\boldsymbol{\xi},t)} \tag{4.10}$$

Let $G(\boldsymbol{x}, t)$ be a conserved quantity in the spatial frame (physically deforming frame). The local form of a conservation law in the spatial description can be written as

$$\frac{\partial}{\partial t}\Big|_{\boldsymbol{x}} G + \mathrm{div}_{\boldsymbol{x}}(G\boldsymbol{v}_n) = 0 \tag{4.11}$$

where $\boldsymbol{v}_n$ is the fluid particle velocity field. We can write this equation as the application of a space–time divergence operator defined as

$$0 = \widetilde{\mathrm{div}}\ \tilde{\boldsymbol{w}} = \begin{Bmatrix} \mathrm{div}_{\boldsymbol{x}} \\ \dfrac{\partial}{\partial t}\Big|_{\boldsymbol{x}} \end{Bmatrix} \cdot \begin{Bmatrix} G\boldsymbol{v}_n \\ G \end{Bmatrix} \tag{4.12}$$

Since $\tilde{\boldsymbol{w}}$ is a vector function of $(\boldsymbol{x}, t)$, its Piola transform $\tilde{\boldsymbol{W}}(\boldsymbol{\xi}, t)$ is

$$\begin{aligned} \tilde{\boldsymbol{W}}(\boldsymbol{\xi}, t) &= \tilde{J} f_n^{-1} \tilde{\boldsymbol{w}}(\boldsymbol{x}, t) \\ &= \tilde{J} \begin{bmatrix} \boldsymbol{F}^{-1} & -\boldsymbol{F}^{-1}\boldsymbol{v}_n^{\mathrm{m}} \\ \boldsymbol{0}^{\mathrm{T}} & 1 \end{bmatrix} \begin{Bmatrix} G\boldsymbol{v}_n \\ G \end{Bmatrix} \\ &= \tilde{J} \begin{Bmatrix} G\boldsymbol{F}^{-1}(\boldsymbol{v}_n - \boldsymbol{v}_n^{\mathrm{m}}) \\ G \end{Bmatrix} \end{aligned} \tag{4.13}$$

Corresponding to the operator $\widetilde{\mathrm{div}}$ in the spatial domain, we define $\widetilde{\mathrm{DIV}}$ in the reference space–time domain as

$$\widetilde{\mathrm{DIV}} = \begin{Bmatrix} \mathrm{DIV}_{\boldsymbol{\xi}} \\ \dfrac{\partial}{\partial t}\Big|_{\boldsymbol{\xi}} \end{Bmatrix} \tag{4.14}$$

Applying (4.14) to (4.13) we get

$$\begin{aligned}
\widetilde{\mathrm{DIV}}\ \tilde{\boldsymbol{W}}(\boldsymbol{\xi},t) &= \left.\frac{\partial}{\partial t}\right|_{\boldsymbol{\xi}} (\tilde{J}G) + \mathrm{DIV}_{\boldsymbol{\xi}}(\tilde{J}\boldsymbol{F}^{-1}G(\boldsymbol{v}_n - \boldsymbol{v}_n^{\mathrm{m}})) \\
&= \left.\frac{\partial}{\partial t}\right|_{\boldsymbol{\xi}} (\tilde{J}G) + \tilde{J}\ \mathrm{tr}(\boldsymbol{F}^{-1}\ \mathrm{GRAD}_{\boldsymbol{\xi}}(G(\boldsymbol{v}_n - \boldsymbol{v}_n^{\mathrm{m}}))) \\
&= \left.\frac{\partial}{\partial t}\right|_{\boldsymbol{\xi}} (\tilde{J}G) + \tilde{J}\ \mathrm{div}_{\boldsymbol{x}}(G(\boldsymbol{v}_n - \boldsymbol{v}_n^{\mathrm{m}})) \\
&= \tilde{J}\left.\frac{\partial}{\partial t}\right|_{\boldsymbol{\xi}} G + \tilde{J}G\ \mathrm{div}_{\boldsymbol{x}}\,\boldsymbol{v}_n^{\mathrm{m}} + \tilde{J}\ \mathrm{div}_{\boldsymbol{x}}(G(\boldsymbol{v}_n - \boldsymbol{v}_n^{\mathrm{m}})) \\
&= \tilde{J}\left\{\left.\frac{\partial}{\partial t}\right|_{\boldsymbol{\xi}} G + G\ \mathrm{div}_{\boldsymbol{x}}\,\boldsymbol{v}_n^{\mathrm{m}} + \mathrm{div}_{\boldsymbol{x}}(G(\boldsymbol{v}_n - \boldsymbol{v}_n^{\mathrm{m}}))\right\}
\end{aligned} \tag{4.15}$$

where we have used the Piola identity $\mathrm{DIV}_{\boldsymbol{\xi}}(\tilde{J}\boldsymbol{F}^{-1}) = 0$ in the second equation and (4.10) in the fourth equation. Eq. (4.15) represents the conservation law written in an arbitrary Lagrangian–Eulerian framework over the moving reference domain $\tilde{Q}_n$. Consequently, the space–time technique and classical ALE procedure are analogous to each other.

REMARKS.

(1) The present approach can be considered as a mapping of the reference configuration of the continuum onto the current configuration (spatial configuration) of the reference domain.

(2) It is important to note that the mapping (4.3) is not restricted to the linear-in-time case, but rather it allows for higher order approximations in space and in time.

(3) An example of a linear-in-time mapping, also discussed in [17], is

$$\Phi_n(\boldsymbol{\xi},t) \equiv (\boldsymbol{\xi} + (t - t_n)\boldsymbol{v}_n^{\mathrm{m}}(\boldsymbol{\xi}), t) \tag{4.16}$$

where $\boldsymbol{v}_n^{\mathrm{m}}$ is the nodal velocity vector of the spatial mesh in space–time slab n.

If the mesh velocity $\boldsymbol{v}_n^{\mathrm{m}} = \boldsymbol{0}$, then from (4.3) and (4.15) it can be seen that we recover the Eulerian form of the equations. However, if the mesh velocity is equal to the fluid particle velocity (i.e. $\boldsymbol{v}_n^{\mathrm{m}} = \boldsymbol{v}_n$), the path of the particles is approximated by the motion of the mesh. It can be seen from (4.15) that the convective derivative vanishes in this case and we recover the Lagrangian formulation.

In situations where the mesh is moved in order to accommodate moving free surfaces, deforming interfaces, or translating objects in the fluid domain, the mesh velocity $\boldsymbol{v}^{\mathrm{m}}$ may not be equal to the fluid particle velocity. In such circumstances, the present scheme is analogous to the arbitrary Lagrangian–Eulerian technique. However, the advantage of the present technique is that by relaxing the continuity-in-time requirements on the finite element functions, a new spatial mesh can be constructed under circumstances of very severe deformations and the solution can be projected from the old mesh onto the new mesh in a variationally consistent manner.

5. Galerkin/least-squares weighted residual formulation

This section presents a statement of the finite element weighted residual formulation which is written on the current configuration, i.e. the spatial configuration of the space–time slab Q_n. The formal statement is that within each space–time slab Q_n, $n = 0, \ldots, N-1$, find $\boldsymbol{V}^h \in \mathcal{S}_n^h$ such that for all $\boldsymbol{W}^h \in \mathcal{V}_n^h$ the following variational equation is satisfied

$$\begin{aligned}
&\int_{Q_n} (-\boldsymbol{W}_{,t}^h \cdot \boldsymbol{U}(\boldsymbol{V}^h) - \boldsymbol{W}_{,i}^h \cdot \boldsymbol{F}_i(\boldsymbol{V}^h) + \boldsymbol{W}_{,i}^h \cdot \tilde{\boldsymbol{K}}_{ij}\boldsymbol{V}_{,j}^h - \boldsymbol{W}^h \cdot \tilde{\boldsymbol{\mathcal{F}}})\,\mathrm{d}Q \\
&\quad + \int_{\Omega_{n+1}} (\boldsymbol{W}^h(t_{n+1}^-) \cdot \boldsymbol{U}(\boldsymbol{V}^h(t_{n+1}^-)))\,\mathrm{d}\Omega - \int_{\Omega_n} (\boldsymbol{W}^h(t_n^+) \cdot \boldsymbol{U}(\boldsymbol{V}^h(t_n^-)))\,\mathrm{d}\Omega \\
&\quad + \sum_{e=1}^{(n_{\mathrm{el}})_n} \int_{Q_n^e} (\mathcal{L}\boldsymbol{W}^h) \cdot \boldsymbol{\tau}(\mathcal{L}\boldsymbol{V}^h - \tilde{\boldsymbol{\mathcal{F}}})\,\mathrm{d}Q
\end{aligned}$$

$$= \int_{P_n} \boldsymbol{W}^h \cdot (-\boldsymbol{F}_i(\boldsymbol{V}^h) + \boldsymbol{F}_i^d(\boldsymbol{V}^h)) n_i \, \mathrm{d}P \tag{5.1}$$

Eq. (5.1) is linear in $\boldsymbol{W}^h$, but nonlinear in $\boldsymbol{V}^h$. The first three integrals on the left-hand side, together with the last integral on the right-hand side, constitute the time-discontinuous Galerkin formulation. Integration-by-parts of the time flux term gives rise to the *jump term*

$$\cdots + \int_{\Omega_n} \boldsymbol{W}^h(t_n^+) \cdot (\boldsymbol{U}(\boldsymbol{V}^h(t_n^+)) - \boldsymbol{U}(\boldsymbol{V}^h(t_n^-))) \, \mathrm{d}\Omega + \cdots \tag{5.2}$$

This term provides a mechanism by which information is propagated from one space–time slab to the next.

The fourth integral in (5.1) is the *least-squares operator*, which is only defined on element interiors. $\mathscr{L}$ is the differential operator defined in (3.3), $\mathscr{L}\boldsymbol{V}^h - \tilde{\boldsymbol{F}}$ is the residual of the Navier–Stokes equations, and $\boldsymbol{\tau}$ is a $n_{\mathrm{dof}} \times n_{\mathrm{dof}}$ symmetric positive semidefinite matrix of intrinsic time scales. Its presence in the formulation enhances the stability of the numerical solution [27].

The boundary integrals on the right-hand side of (5.1) give rise to natural boundary conditions. A comprehensive account of the treatment of boundary conditions and the boundary integral is presented in [42]. This space–time weighted residual formulation is consistent since the exact solution of the Navier–Stokes equations, $\boldsymbol{V}$, also satisfies the variational equation.

6. Linear-in-time approximation

The time finite element space can be selected constant-in-time, or linear-in-time, etc., resulting in a first-order, third-order, etc., accurate system of equations, respectively. In this work we are interested in transient problems with changing spatial configurations (see Fig. 3) that requires accuracy higher than first order in time. Consequently, we will focus on the linear-in-time finite element approximation.

The finite element trial solution within the nth space–time slab is defined as

$$\boldsymbol{V}^h(x,t) = \sum_{A=1}^{(n_{np})_{(n)}} [N_A^{(n)}(x)(\pi_n(t)\boldsymbol{V}_{A;(n+1)}) + \tilde{N}_A^{(n)}(x)(\tilde{\pi}_n(t)\tilde{\boldsymbol{V}}_{A;(n)})] \quad \text{for } (x,t) \in Q_n \tag{6.1}$$

where $N_A(x)$ and $\tilde{N}_A(x)$ are the spatial shape functions of node A at time levels t_{n+1}^- and t_n^+, respectively. The primary variable $\boldsymbol{V}_{A;(n+1)}$ and the secondary variable $\tilde{\boldsymbol{V}}_{A;(n)}$ are the $n_{\mathrm{dof}} \times 1$ vectors of unknowns $\boldsymbol{V}^h$

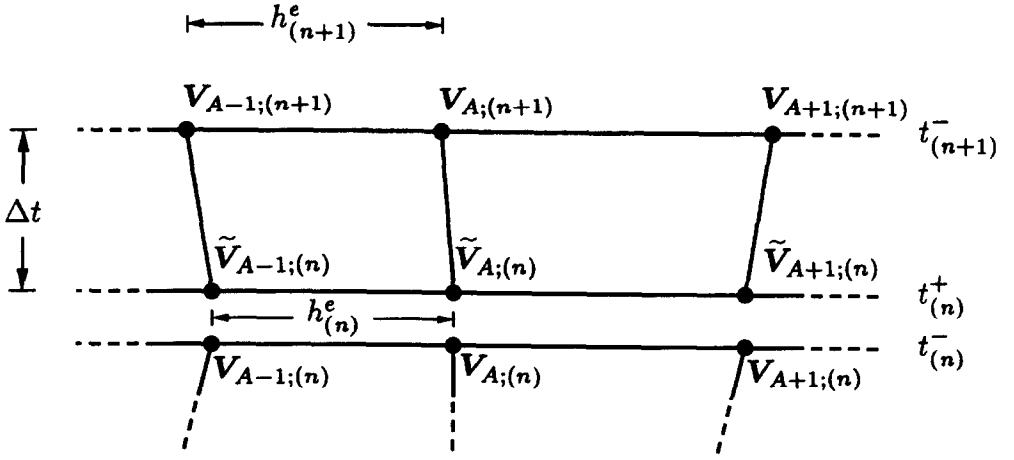


Fig. 3. A schematic diagram of the nodal configuration for linear-in-time space–time formulation of a one-dimensional moving domain.

at node A and times t_{n+1}^{-} and t_n^{+}, respectively. $\pi_n(t)$ and $\tilde{\pi}_n(t)$ are the temporal shape-functions in space–time slab n and are defined as

$$\pi_n(t) = \frac{t - t_n}{\Delta t} \tag{6.2}$$

$$\tilde{\pi}_n(t) = \frac{t_{n+1} - t}{\Delta t} \tag{6.3}$$

Similarly, we define the finite element weighting function for the nth space–time slab as

$$\boldsymbol{W}^h(\boldsymbol{x}, t) = \sum_{A=1}^{(n_{np})_{(n)}} [N_A^{(n)}(\boldsymbol{x})(\pi_n(t)\boldsymbol{W}_{A;(n+1)}) + \tilde{N}_A^{(n)}(\boldsymbol{x})(\tilde{\pi}_n(t)\tilde{\boldsymbol{W}}_{A;(n)})] \quad \text{for } (\boldsymbol{x}, t) \in Q_n \tag{6.4}$$

where $\boldsymbol{W}_{A;(n+1)}$ and $\tilde{\boldsymbol{W}}_{A;(n)}$ are the nodal values of the weighting functions (see [42] for further details).

7. An adaptive mesh rezoning strategy

A major challenge in the class of problems considered here lies in the development of a mesh rezoning technique to adapt the fluid mesh to the changing spatial configuration (for background concerning mesh movement schemes see [6,9,28,33,43,48] and references therein). In this section we present an adaptive mesh rezoning technique which is appropriate for arbitrarily shaped domains. The arbitrary orientation in time of the space–time elements provides an opportunity to integrate such a mesh moving technique in a consistent manner in the overall solution strategy.

Let $\Omega \subset R^{n_{sd}}$ be a bounded open set with piecewise smooth boundary Γ; $n_{sd} \geq 2$ denotes the number of spatial dimensions. We assume that Γ admits the decomposition

$$\Gamma = \Gamma_m \cup \Gamma_f \tag{7.1}$$

and

$$\emptyset = \Gamma_m \cap \Gamma_f \tag{7.2}$$

where Γ_m and Γ_f are the moving and the fixed portions of the boundary, respectively.

The formal statement of the boundary-value problem is: Given $\boldsymbol{g}$, the prescribed displacements at the moving boundary, find the mesh displacement field $\boldsymbol{u}: \Omega \to R^{n_{sd}}$, such that

$$\nabla \cdot ([1 + \tau_m]\nabla)\boldsymbol{u} = \boldsymbol{0} \quad \text{in } \Omega \tag{7.3}$$

$$\boldsymbol{u} = \boldsymbol{g} \quad \text{on } \Gamma_m \tag{7.4}$$

$$\boldsymbol{u} = \boldsymbol{0} \quad \text{on } \Gamma_f \tag{7.5}$$

Eqs. (7.3)–(7.5) are, respectively, the governing equation, the moving and the fixed boundary conditions. τ_m is a bounded, non-dimensional function which is designed to prevent the inversion of small elements in the high resolution regions of the fluid mesh.

In coupled interaction problems the motion, $\boldsymbol{g}$, of the interface boundaries is a function of the overall response of the system and not known a priori. Consequently, it is imperative to seek solution to the problem in an unconstrained space of functions in which the boundary conditions are not embedded from the outset. Accordingly, we propose an augmented Lagrangian formulation for the problem of mesh rezoning.

The functional for the augmented Lagrangian formulation is

$$\Pi(\boldsymbol{u}, \boldsymbol{p}) = \frac{1}{2}([1 + \tau_m]\nabla\boldsymbol{u}, \nabla\boldsymbol{u}) + \langle \boldsymbol{p}, \boldsymbol{u} - \boldsymbol{g} \rangle + \frac{\varepsilon}{2}|\boldsymbol{u} - \boldsymbol{g}|^2 \tag{7.6}$$

where ε is a user specified penalty parameter, $(\cdot,\cdot)$ denotes the L_2-norm on Ω, $|\cdot|$ denotes the L_2-norm on Γ_m, $\langle \cdot,\cdot \rangle$ is the L_2-inner product on Γ_m, and $\boldsymbol{p}$ is the boundary Lagrange multiplier. The variational equation emanating from (7.6) is

$$0 = \left(\frac{\mathrm{d}}{\mathrm{d}\delta} \Pi(\boldsymbol{u} + \delta \boldsymbol{w}, \boldsymbol{p} + \delta \boldsymbol{q})\right)\Big|_{\delta=0}$$
$$= ([1 + \tau_{\mathrm{m}}]\nabla \boldsymbol{u}, \nabla \boldsymbol{w}) + \langle \boldsymbol{q}, \boldsymbol{u} - \boldsymbol{g} \rangle + \langle \boldsymbol{p}, \boldsymbol{w} \rangle + \varepsilon \langle \boldsymbol{u} - \boldsymbol{g}, \boldsymbol{w} \rangle \tag{7.7}$$

REMARKS.

(1) This formulation can be viewed as a combination of the penalty function and the Lagrange multiplier method. It combines the two concepts to eliminate many of the disadvantages associated with each method alone (see e.g. [15]).

(2) For $\boldsymbol{p} = \boldsymbol{0}$, we have

$$\Pi(\boldsymbol{u}, \boldsymbol{0}) = \frac{1}{2}([1 + \tau_{\mathrm{m}}]\nabla \boldsymbol{u}, \nabla \boldsymbol{u}) + \frac{\varepsilon}{2}|\boldsymbol{u} - \boldsymbol{g}|^2 \tag{7.8}$$

This is the classical penalty function formulation for the constraint $\boldsymbol{u} = \boldsymbol{g}$. The advantage of the augmented Lagrangian formulation is that due to the term $\langle \boldsymbol{p}, \boldsymbol{u} - \boldsymbol{g} \rangle$, the exact solution of the problem (7.3)–(7.5) can be determined without making ε tend to infinity, which, using ordinary penalty methods, would have the effect of causing a deterioration in the conditioning of the system to be solved.

The variational form (7.7) can be expressed as: Find $\{\boldsymbol{u}, \boldsymbol{p}\} \in \mathcal{V} \times \mathcal{W}$, where $\mathcal{V} = H^1(\Omega)^{n_{\mathrm{sd}}}$ and $\mathcal{W} = H^{1/2}(\Omega)^{n_{\mathrm{sd}}}$, such that

$$B(\boldsymbol{u}, \boldsymbol{p}; \boldsymbol{w}, \boldsymbol{q}) = L(\{\boldsymbol{w}, \boldsymbol{q}\}) \quad \forall \{\boldsymbol{w}, \boldsymbol{q}\} \in \mathcal{V} \times \mathcal{W} \tag{7.9}$$

where

$$\begin{aligned} B(\boldsymbol{u}, \boldsymbol{p}; \boldsymbol{w}, \boldsymbol{q}) &= ([1 + \tau_{\mathrm{m}}]\nabla \boldsymbol{u}, \nabla \boldsymbol{w}) + \varepsilon \langle \boldsymbol{u}, \boldsymbol{w} \rangle + \langle \boldsymbol{p}, \boldsymbol{w} \rangle + \langle \boldsymbol{q}, \boldsymbol{u} \rangle \\ L(\{\boldsymbol{w}, \boldsymbol{q}\}) &= \varepsilon \langle \boldsymbol{w}, \boldsymbol{g} \rangle + \langle \boldsymbol{q}, \boldsymbol{g} \rangle \end{aligned} \tag{7.10}$$

Let $\mathcal{V}^h$ and $\mathcal{W}^h$ be finite-dimensional subspaces of $\mathcal{V}$ and $\mathcal{W}$, respectively. The discrete form of (7.9) can be expressed as: Find $\{\boldsymbol{u}^h, \boldsymbol{p}^h\} \in \mathcal{V}^h \times \mathcal{W}^h$ such that

$$B(\boldsymbol{u}^h, \boldsymbol{p}^h; \boldsymbol{w}^h, \boldsymbol{q}^h) = L(\{\boldsymbol{w}^h, \boldsymbol{q}^h\}) \quad \forall \{\boldsymbol{w}^h, \boldsymbol{q}^h\} \in \mathcal{V}^h \times \mathcal{W}^h \tag{7.11}$$

where

$$B(\boldsymbol{u}^h, \boldsymbol{p}^h; \boldsymbol{w}^h, \boldsymbol{q}^h) = (\nabla \boldsymbol{u}^h, \nabla \boldsymbol{w}^h) + \varepsilon \langle \boldsymbol{u}^h, \boldsymbol{w}^h \rangle + \langle \boldsymbol{p}^h, \boldsymbol{w}^h \rangle + \langle \boldsymbol{q}^h, \boldsymbol{u}^h \rangle + \sum_{e=1}^{n_{\mathrm{el}}} \tau_{\mathrm{m}}^e (\nabla \boldsymbol{u}^h, \nabla \boldsymbol{w}^h)_{\Omega^e} \tag{7.12}$$

REMARKS.

(1) We may use a conjugate gradient algorithm with diagonal preconditioning to solve the resulting system of equations. For a detailed account of conjugate gradient algorithms (see e.g. [13,15] and references therein).

(2) In order to have an automatic check on element distortion, a simple check on element Jacobians can be made, i.e.

$$\delta_1 J_{\mathrm{ref}} \not< J_{\mathrm{cur}} \not< \delta_2 J_{\mathrm{ref}}$$

where δ_1 and δ_2 are the user specified tolerance parameters for the change in the area of elements in the current mesh, J_{cur} and J_{ref} denote the Jacobians of elements in the current and reference configurations, respectively.

7.1. *Design of the weight function for mesh movement*

As mentioned previously, τ_{m} is a positive, nondimensional and bounded function which is designed to prevent the inversion of smaller elements in the mesh. Our objective is to bound the absolute value of the relative displacement by the absolute value of the distance between points.

$$|\boldsymbol{u}_i - \boldsymbol{u}_j| \leq \alpha |\boldsymbol{x}_i - \boldsymbol{x}_j| \tag{7.13}$$

$$\Rightarrow |\nabla \boldsymbol{u}| \leq \alpha \tag{7.14}$$

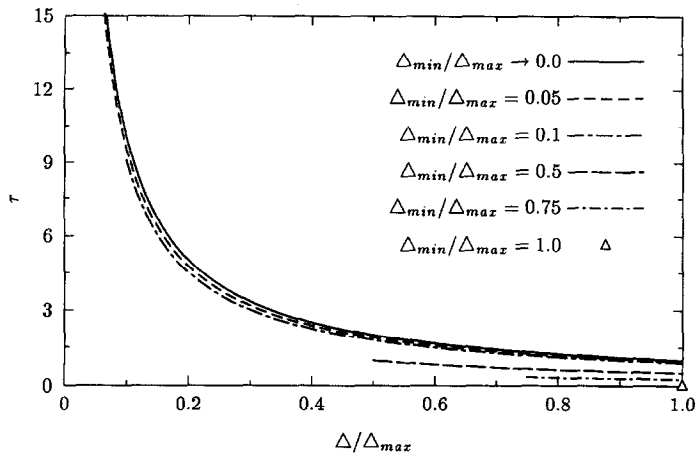


Fig. 4. Weight function for various mesh refinements.

where $\alpha \in [0, 1)$ is the tolerance parameter for domain distortion. The case of least distortion in the smaller elements can be attained in the limit as $\alpha \to 0$, namely

$$\nabla \boldsymbol{u} = 0 \tag{7.15}$$

τ_{m} is designed to apply this condition strongly over the smaller elements, thereby preserving their shapes and preventing mesh inversion. In this way, the smaller elements close to the moving interfaces translate with the least amount of distortion, and the larger elements in the far field deform more to absorb the motion. Consequently, shape of the elements is maintained in the boundary layer regions, resulting in a well-behaved mesh.

A simple example of τ_{m} follows. Let $\Delta_{\max}$ and $\Delta_{\min}$ represent the area of the largest and the smallest element in the mesh, respectively, and let Δ^e represent the area of the element under current consideration. We define τ_{m} for an element as

$$\tau_{\mathrm{m}}^e = \frac{1 - \Delta_{\min}/\Delta_{\max}}{\Delta^e/\Delta_{\max}} \qquad e = 1, 2, \ldots, n_{\mathrm{el}} \tag{7.16}$$

The graphical representation of the behavior of τ_{m} is presented in Fig. 4. The values of τ_{m} employed for the numerical simulations mimic this behavior.

REMARK. For the degenerate case of a uniform mesh where $\Delta_{\max} = \Delta_e = \Delta_{\min}$, the Laplace equation works well. Under such circumstances $\tau_{\mathrm{m}} = 0$ (see Fig. 4) and the additional term over the element interiors in (7.12) vanishes. However, as soon as any movement of the mesh takes place, it becomes graded, and a non-zero τ activates the additional term in (7.12) for successive rezonings.

8. Boundary conditions for fluid–structure interaction

Fig. 5 presents various types of fluid–solid interface conditions encountered in practice. Continuity of displacement, velocity and traction fields must be satisfied for all times at all the points on the boundary Γ.

8.1. Coupling of solid and fluid-mesh displacement fields

In transient fluid–structure interaction problems, a Lagrangian mesh for the structure deforms with the structure and maintains a sharp definition of the moving boundary. The appropriate interface condition for the fluid mesh is

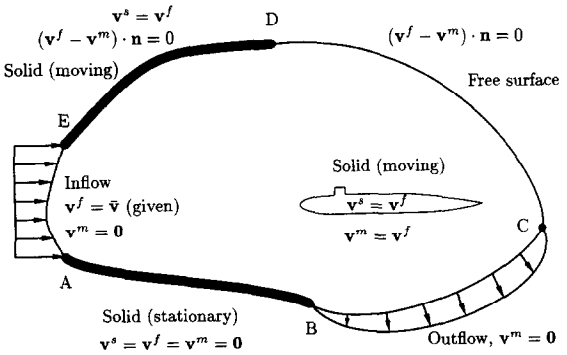


Fig. 5. A schematic diagram of the various types of kinematic boundary conditions frequently encountered in multi-media interaction problems.

$$(\boldsymbol{u}^s - \boldsymbol{u}^m)\cdot\boldsymbol{n} = 0 \quad \text{on } \Gamma_m \tag{8.1}$$

where $\boldsymbol{n}$ is the unit outward normal to the interface (from the structure into the fluid), Γ_m denotes the interface, and $\boldsymbol{u}^s$ and $\boldsymbol{u}^m$ represent the displacement fields of the solid and the fluid mesh at Γ_m, respectively. By imposing (8.1) we keep the fluid nodes belonging to Γ_m on the moving interface while permitting relative slip between the solid and the fluid meshes in the tangential direction. This condition is particularly useful in the analysis of free surface waves breaking against a solid wall or the transient response analysis of liquid-filled tanks, where we need to let the fluid mesh slide along the structure.

In certain situations it is advantageous to move the fluid mesh together with the solid even in the tangential direction. In such a situation we need also apply

$$(\boldsymbol{u}^s - \boldsymbol{u}^m)\cdot\boldsymbol{t} = 0 \quad \text{on } \Gamma_m \tag{8.2}$$

in addition to (8.1), where $\boldsymbol{t}$ is the unit tangent vector to the interface.

8.2. *No-slip boundary conditions for fluid–structure system*

Newtonian fluids add the restriction that the fluid particles adhere, without slipping, to the boundary, i.e.

$$(\boldsymbol{v}^s - \boldsymbol{v}^f) = \boldsymbol{0} \quad \text{on } \Gamma \tag{8.3}$$

where $\boldsymbol{v}^s$ and $\boldsymbol{v}^f$ represent the velocity fields of the solid and the fluid particles, respectively.

In addition to the kinematic boundary conditions we need to impose the continuity of traction at the fluid–solid interfaces, i.e.

$$(\boldsymbol{\sigma}^s + \boldsymbol{\sigma}^f)\cdot\boldsymbol{n} = \boldsymbol{0} \quad \text{on } \Gamma \tag{8.4}$$

where $\boldsymbol{\sigma}^s$ and $\boldsymbol{\sigma}^f$ are the stress tensors in the solid and fluid, respectively.

8.3. *Slip boundary conditions for fluid–structure system*

In the interaction of inviscid fluids with structures, only normal compatibility is required, i.e. only the normal component of the velocity field is continuous across the fluid–structure interface, viz.

$$(\boldsymbol{v}^s - \boldsymbol{v}^f)\cdot\boldsymbol{n} = 0 \quad \text{on } \Gamma \tag{8.5}$$

Furthermore, in such simulations only the normal stress at the surface of the solid is equated to the mean pressure in the fluid while no restriction is imposed on the tangential stresses, viz.

$$\boldsymbol{n}\cdot(\boldsymbol{\sigma}^s + \boldsymbol{\sigma}^f)\boldsymbol{n} = 0 \quad \text{on } \Gamma \tag{8.6}$$

$$\boldsymbol{t}_i \cdot \boldsymbol{\sigma}^s \boldsymbol{n} = 0 \qquad \text{on } \Gamma \tag{8.7}$$

$$\boldsymbol{t}_i \cdot \boldsymbol{\sigma}^f \boldsymbol{n} = 0 \qquad \text{on } \Gamma \tag{8.8}$$

where $\boldsymbol{t}_i$, $i = 1, 2$, are the unit vectors which span the tangent plane to Γ.

9. Numerical examples

This section presents some numerical simulations to demonstrate the performance and the range of applicability of the proposed methodology. To solve the linear matrix systems, we have used the GMRES iterative solver and Gauss–Siedel EBE preconditioner [41]. A global time-increment strategy was used for all test cases.

9.1. Moving cylinder: Steady state solution

In this problem we test the algorithm on a rigidly moving mesh and compare it with an essentially identical problem in which the mesh is stationary but subjected to a uniform flow. Adjusting the solutions by the uniform superposed flow results in identical solutions, and thus this comparison is a good initial test of the veracity of the approach.

In this test case a circular cylinder moving at Mach 0.01 and Reynolds number 40 is introduced in a stationary fluid domain at time t_0 (Fig. 6). The Reynolds number is based on the mean stream density of the fluid and the velocity and diameter of the cylinder. The computational domain covers an area $-6 \leq x \leq 20$ and $-6 \leq y \leq 6$, with a unit diameter cylinder centered at $x = 0$ and $y = 0$. An unstructured mesh comprising 4936 bilinear quadrilaterals with 5063 nodes is generated to solve this problem (Fig. 7). Free stream values of ρ, v_1 and v_2 are prescribed (i.e. $v_1 = v_2 = 0$) on the inflow, top and bottom boundaries. Zero heat and viscous flux conditions are imposed on the top, bottom and outflow boundaries. The no-slip condition $v_1 = -1$ and $v_2 = 0$ is prescribed on the surface of the cylinder. In this simulation the spatial mesh also translates rigidly with the cylinder.

The computed steady flow-field, employing an implicit, first-order predictor corrector algorithm with one corrector pass is show in the accompanying figures. The maximum dimension of the Krylov space is 5. Fig. 8 presents the pressure coefficient $C_p = (p - p_\infty)/\frac{1}{2}\rho_\infty v_c^2$, and the friction coefficient $C_f = \tau_{wall}/\frac{1}{2}\rho_\infty v_c^2$, ($v_c$ is the velocity of the cylinder), developed on the upper surface of the moving cylinder. The solution is then compared with the steady flow-field generated by a uniform steady flow past a stationary cylinder at the same Reynolds number [38]. Also shown are the experimental values of C_p by

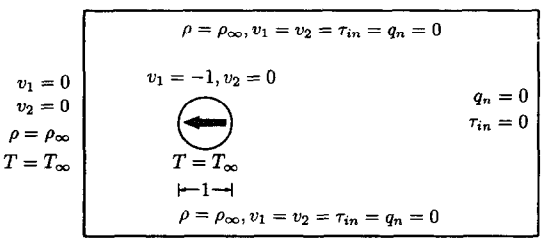


Fig. 6. Schematic diagram of a circular cylinder moving in a stationary flow field.

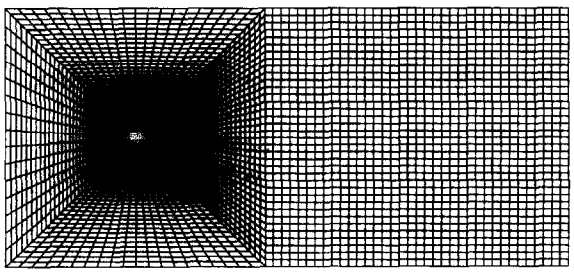


Fig. 7. A view of the entire mesh.

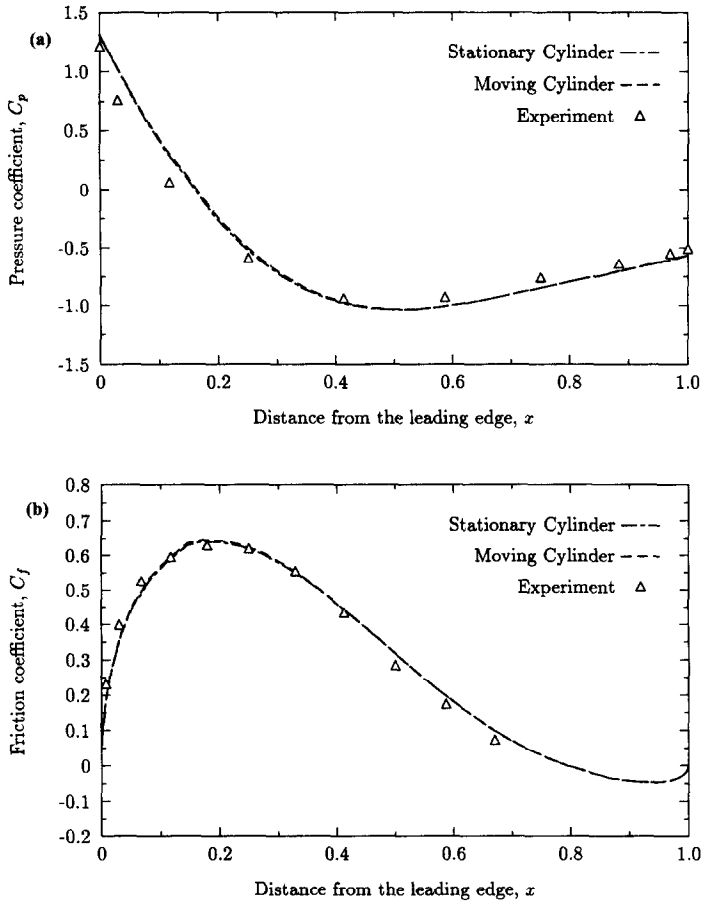


Fig. 8. Comparison of wall quantities for stationary and moving cylinders. Pressure and skin-friction coefficients.

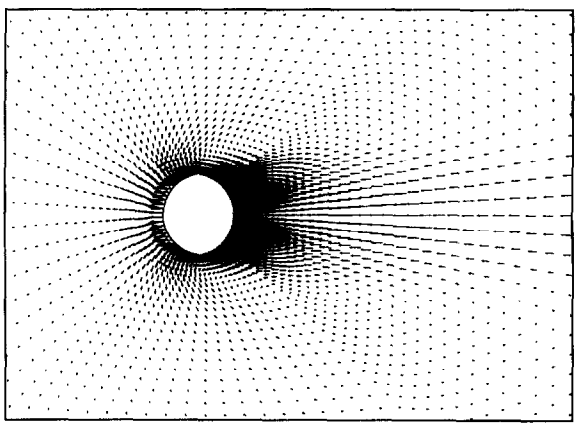


Fig. 9. Absolute velocity vector field.

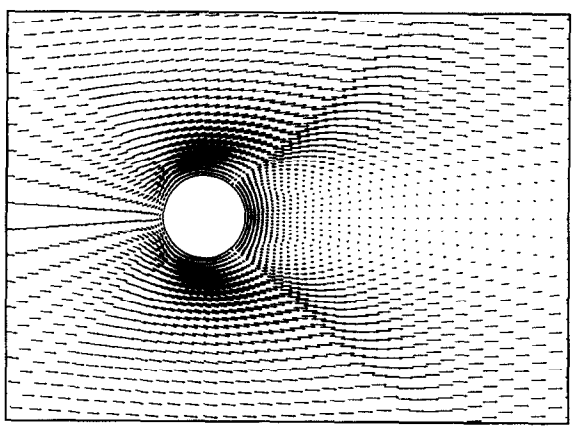
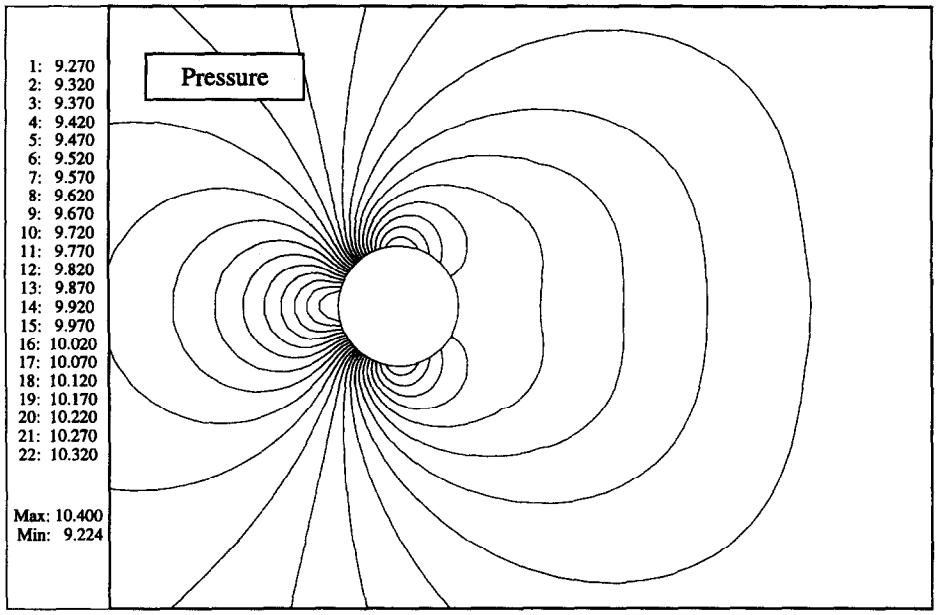
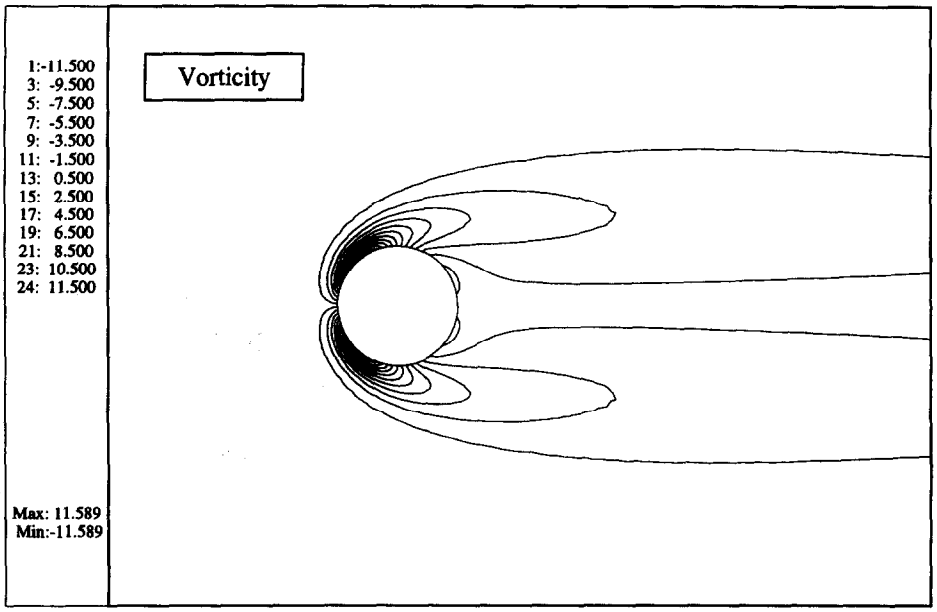


Fig. 10. Relative velocity vector field.

Grove et al. [16, p. 66], and C_f by Acrivos et al. [1, p. 37] which are in good agreement with the computed results. Figs. 9 and 10 present the absolute and the relative velocity vector fields developed around the moving cylinder while Fig. 11 presents the pressure and vorticity fields. A comparison of the flow fields around the moving and stationary cylinders revealed that the relative velocity, density, pressure and vorticity fields were virtually identical [38].



(a)



(b)

Fig. 11. Pressure and vorticity fields around the moving cylinder.

9.2. Solitary wave propagation

The second test case simulates the propagation of a solitary wave. Fig. 12 shows the geometry of the computational domain. H and d represent the maximum wave height and the still water depth, respectively, with $H/d = 0.2$. R denotes the run-up height of fluid against a rigid wall. Laitone's analytical approximations have been used for a comparative study where $\eta(x, t)$ defines the initial crest [40, p. 414].

Fig. 13 shows a complete view of the initial mesh which is composed of 6169 linear triangular elements. In this simulation the density is assumed to be constant and gravity $g = 1$. The fluid is assumed to be inviscid and slip boundary conditions are imposed on all interfaces. The fluid mesh is moved proportional to the normal component of the fluid velocity at the free surface. The time step increment is 0.02. In this simulation we have used an implicit, third-order predictor multicorrector algorithm with 3 corrector passes. The maximum dimension of the Krylov space is 10.

Fig. 14 shows a zoomed view of the initial displacement and velocity fields. At time $t = 7.7$, the wave reaches the right wall and Fig. 15 shows the corresponding displacement and velocity fields. The wave reflects and at time $t = 15$, reaches the center of the channel as shown in Fig. 16.

Table 1 presents the time history of the normalized run-up heights at the right and left walls and the normalized maximum height at the center of the channel for two complete cycles.

9.3. Missile launch from a submarine

This simulation presents a viscous flow-field generated around a moving submerged missile. The viscosity of the fluid is assumed constant and the effects of gravity have been neglected. In this simulation, the Mach number is 0.01 and Reynolds number based on the velocity and length of the missile and mean flow density of the fluid is 1000.

Fig. 17 shows a detailed account of the boundary conditions. On all the outer boundaries, temperature T and velocity components v_1 and v_2 are prescribed. On the body of the submarine, the no-slip condition and the temperature T are prescribed to account for the adhesion of the viscous fluid to the surface of the submarine and to simulate an isothermal process, respectively. On the surface of the missile the velocity of the fluid particles is the same as the missile velocity, i.e. $v_1 = 0$, $v_2 = 1$.

The computational domain is $-7 \leq x \leq 7$ and $0 \leq y \leq 6$, with a unit length missile placed in the launch tube at $x = 0$ and $y = -1$. Fig. 18 shows the complete initial mesh. The missile is moved via the mesh rezoning technique until the shapes of the elements in the mesh start deteriorating. At this instant, a new mesh is constructed, and information is projected from the old mesh onto the new mesh.

The entire simulation requires five remeshings. We will first show the process of evolution of the spatial mesh and a zoomed view of the tip and tail sections around the missile in the final configuration for a given mesh. For example, Fig. 19 presents the initial configuration, an intermediate configuration,

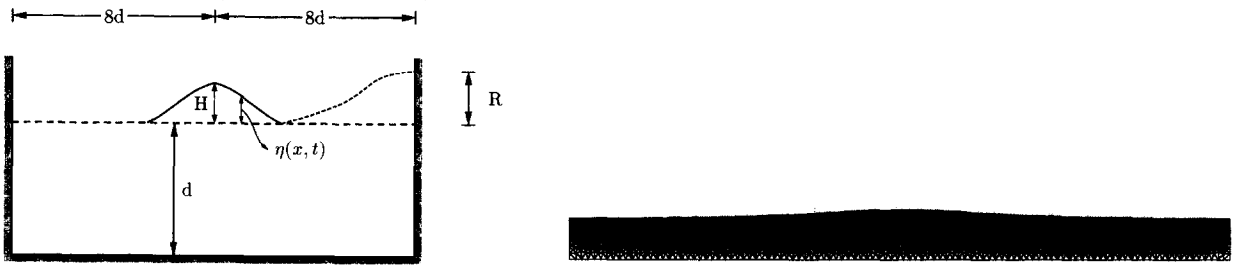
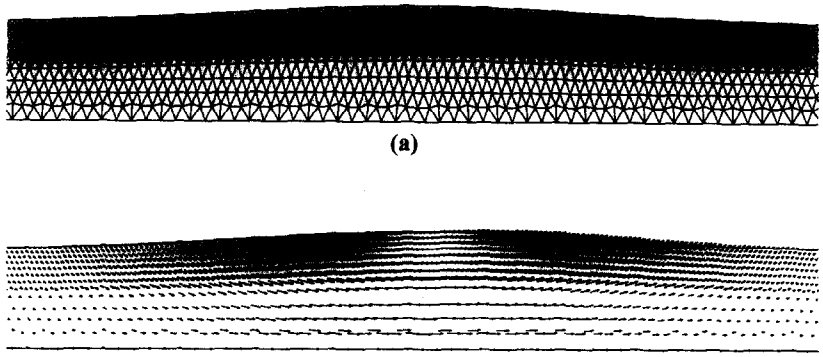


Fig. 12. Details of the problem description.

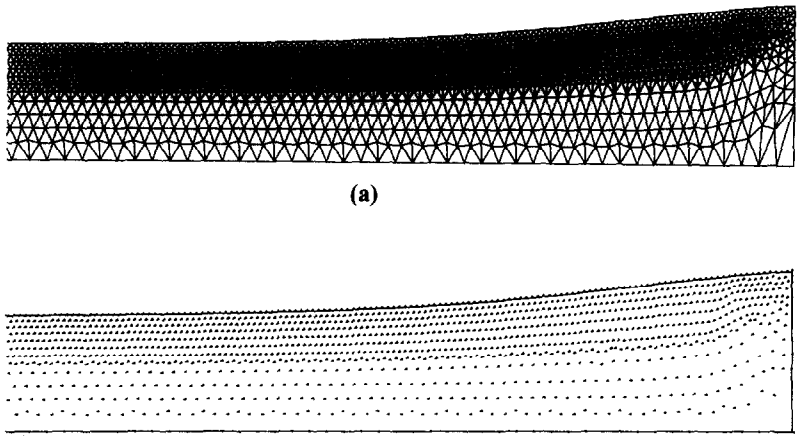
Fig. 13. A view of the entire mesh.



(a)

(b)

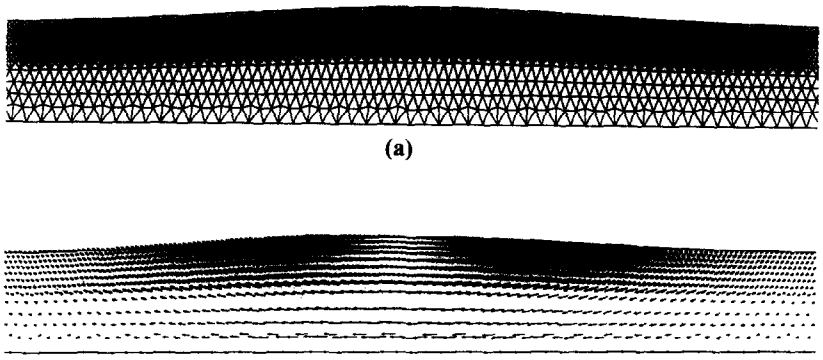
Fig. 14. Initial displacement and velocity fields.



(a)

(b)

Fig. 15. Displacement and velocity field at $t = 7.7$.



(a)

(b)

Fig. 16. Displacement and velocity field at $t = 15$.

Table 1
Numerical solution normalized by Laitone's analytical solution [40, p. 414].

Time (s)	Left wall	Center of channel	Right wall
0.0	–	1.0	–
7.7	–	–	1.026
15.0	–	0.998	–
22.7	1.021	–	–
30.0	–	0.993	–
37.7	–	–	1.002
45.0	–	0.990	–
52.7	0.998	–	–
60.0	–	0.989	–

and the final configuration of the first mesh. At this stage a new mesh is generated around the current position of the missile (see Fig. 22). It can be seen that the proposed mesh rezoning strategy prevents the inversion of the elements in the boundary layer regions even under very severe mesh distortions.

In order to prevent temporal oscillations due to start-up, we move the missile from rest with a constant acceleration till it achieves a constant velocity $u_2 = 1$. Figs. 28–32 present the pressure

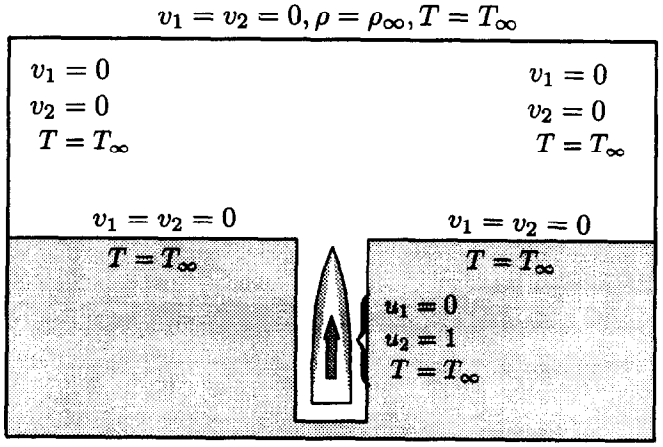


Fig. 17. Details of the problem description.

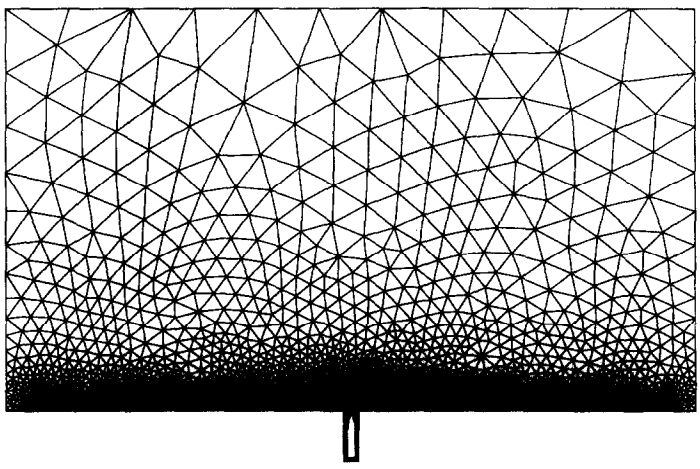


Fig. 18. A complete view of the initial mesh.

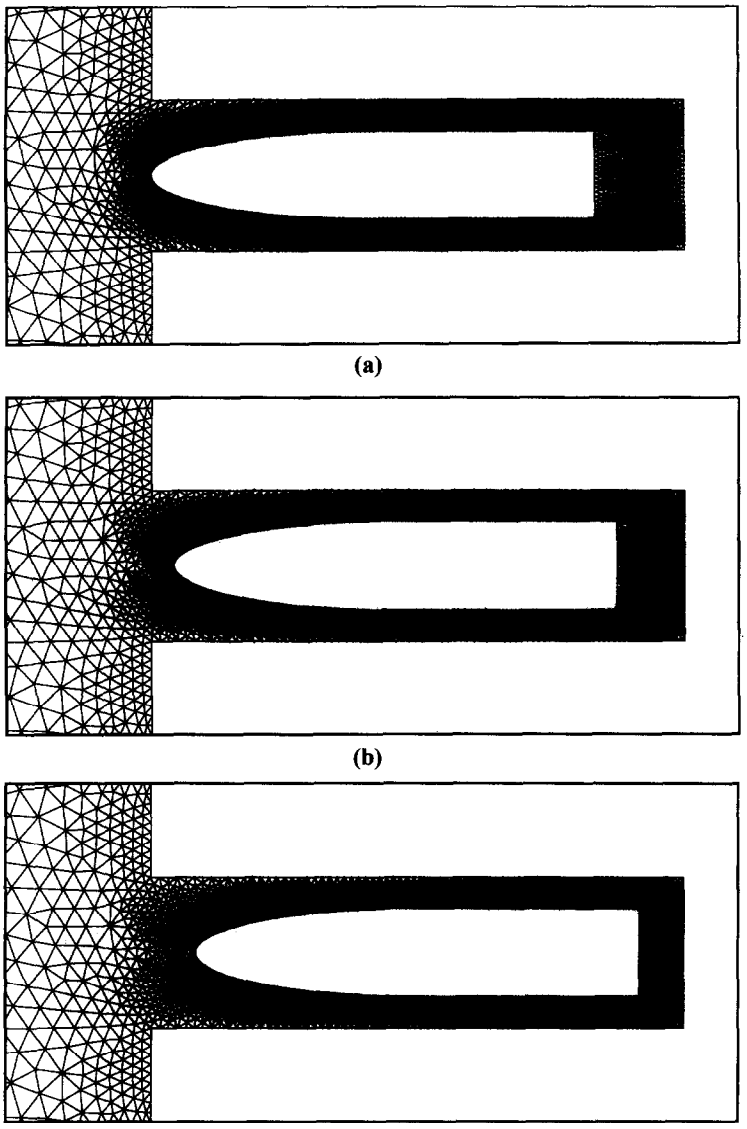


Fig. 19. Spatial configuration of the first mesh at various instants.

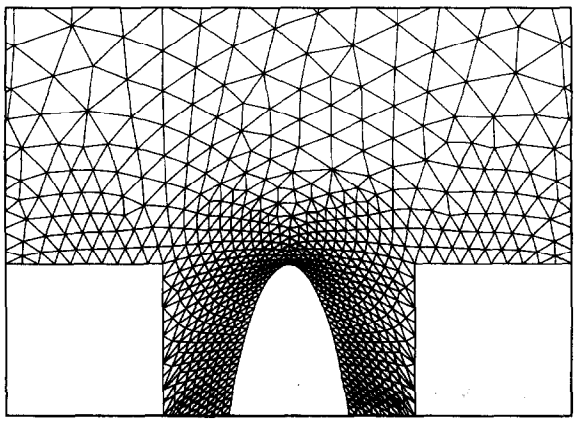


Fig. 20. Tip of the final configuration of first mesh.

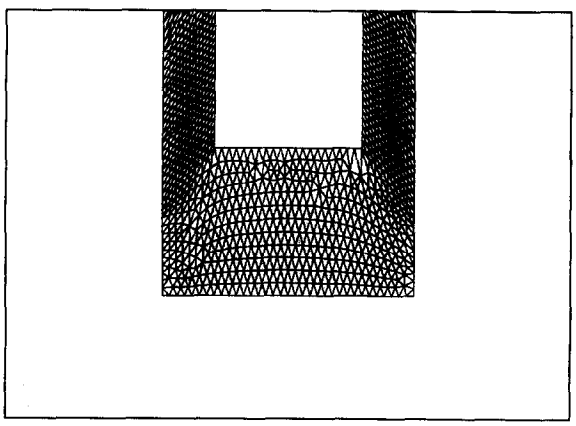


Fig. 21. Tail of the final configuration of first mesh.

contours and streamlines around the moving missile at various instants. The slight nonsymmetry in pressure contours in Fig. 32 is due to the fact that we are using an unstructured mesh which is not symmetric about the symmetry line, thereby rendering the ensuing discrete problem non-symmetric.

We used an implicit, third-order accurate predictor multicorrector algorithm with 5 corrector passes. A global time step increment $\Delta t = 5 \times 10^{-3}$ was used and the maximum dimension of the Krylov space was 10. The average number of elements in the various meshes generated to solve this problem was 12 300 with an average number of 6526 nodes in each mesh. Approximately 5.8% of the total CPU time was consumed in mesh rezonings.

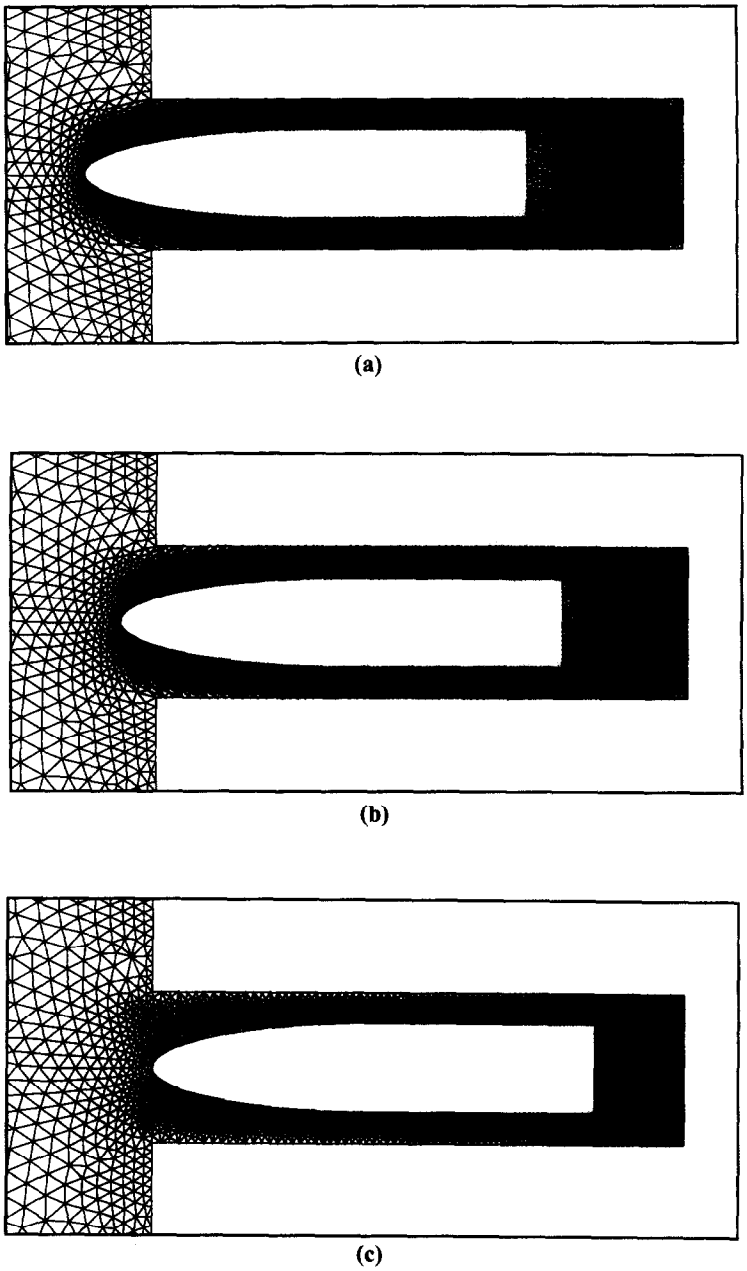


Fig. 22. Spatial configuration of the second mesh at various instants.

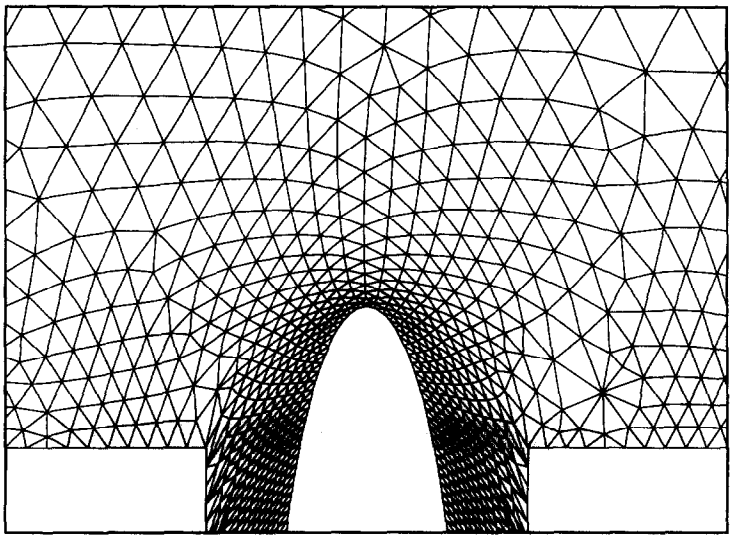


Fig. 23. Tip of the final configuration of second mesh.

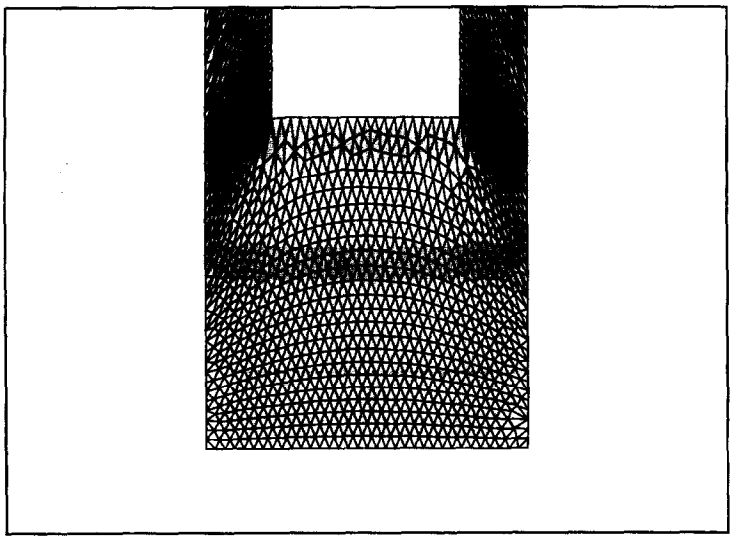


Fig. 24. Tail of the final configuration of second mesh.

9.4. *Coupling of compliant coating with viscous fluid*

This test case presents the transient response of a viscous fluid coupled to a linear elastic solid. Fig. 33 shows the details of the problem description. The effect of gravity has been neglected. The elastic

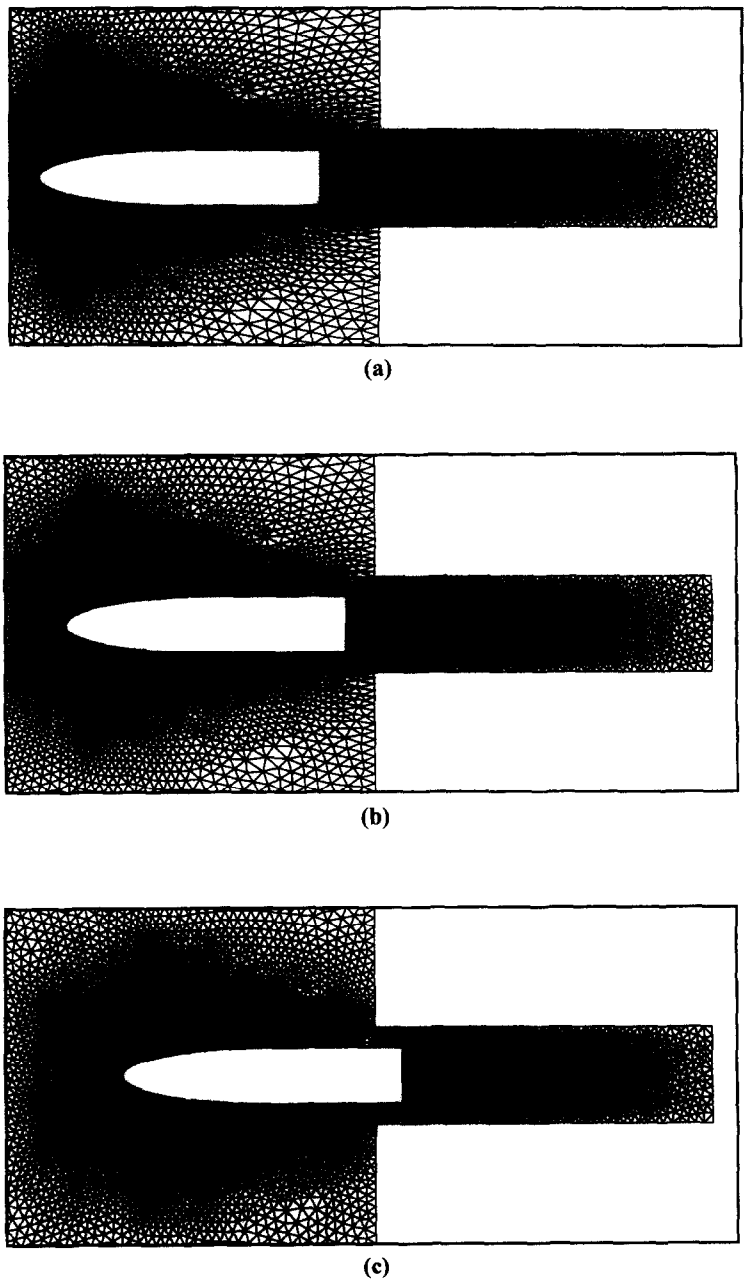


Fig. 25. Spatial configuration of the fifth mesh at various instants.

structure is modeled with a finite element formulation described in [38]. No-slip conditions are prescribed at the interface. The data employed in this analysis are as follows: Young's modulus of solid, $E = 2. \times 10^{10}$ Pa, Poisson's ratio, $\nu = 0.3$, density of solid, $\rho_s = 1500\,\mathrm{kg/m^3}$, bulk modulus of water, $\beta = 2.28 \times 10^9$ Pa, density of fluid, $\rho_f = 1026\,\mathrm{kg/m^3}$, Reynolds number Re = 1000, and viscosity of the fluid is assumed to be constant.

Fig. 34 shows a complete view of the initial mesh. It is composed of an unstructured mesh of 5018

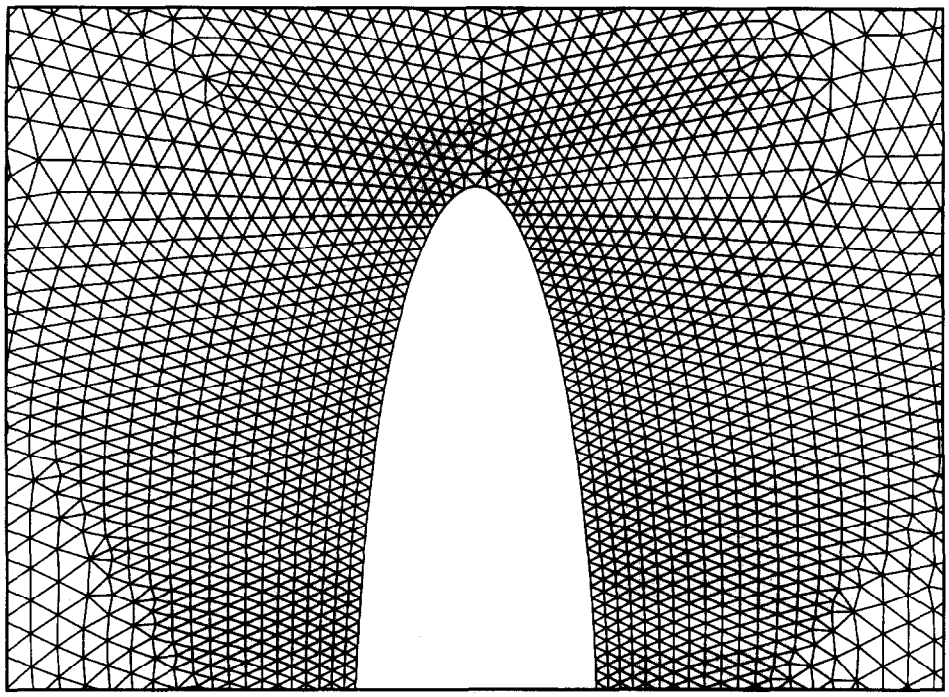


Fig. 26. Tip of the final configuration of fifth mesh.

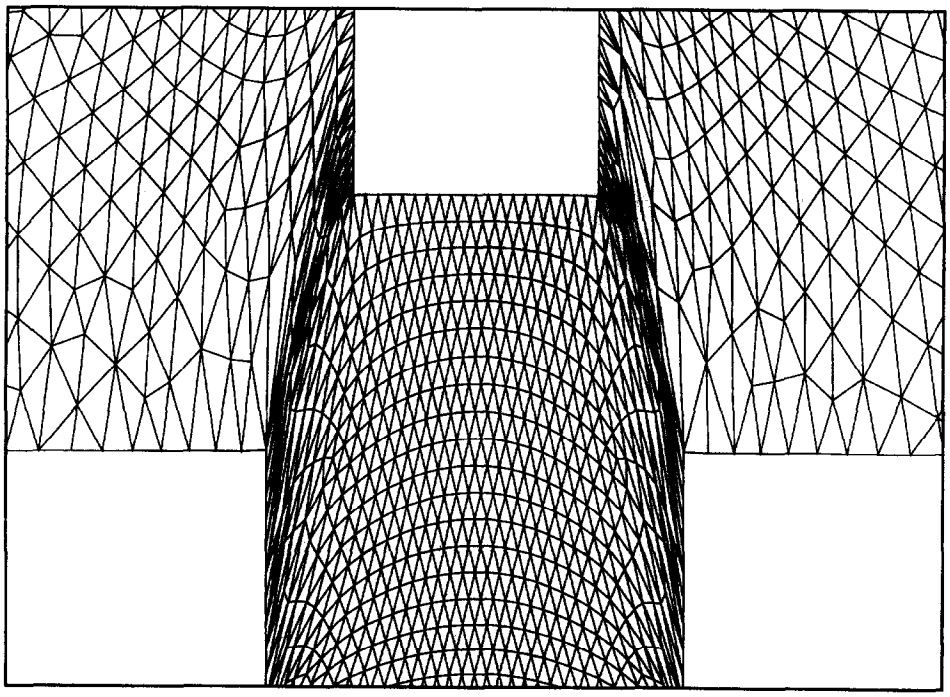
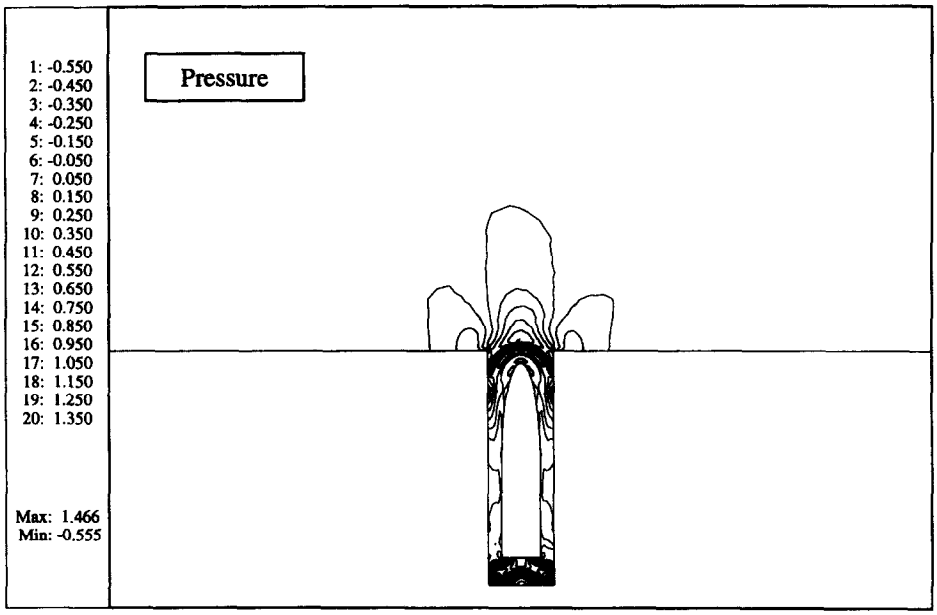
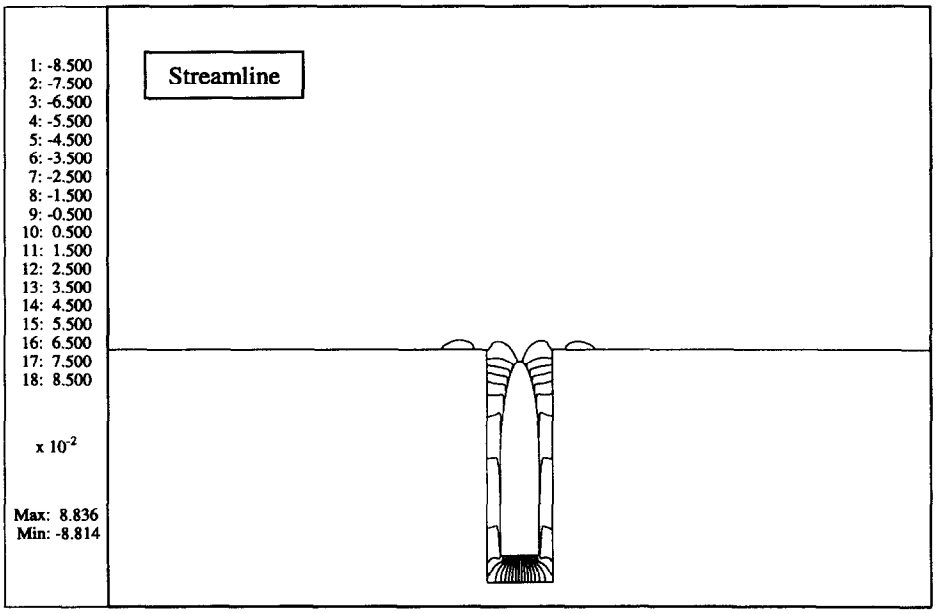


Fig. 27. Tail of the final configuration of fifth mesh.



(a)

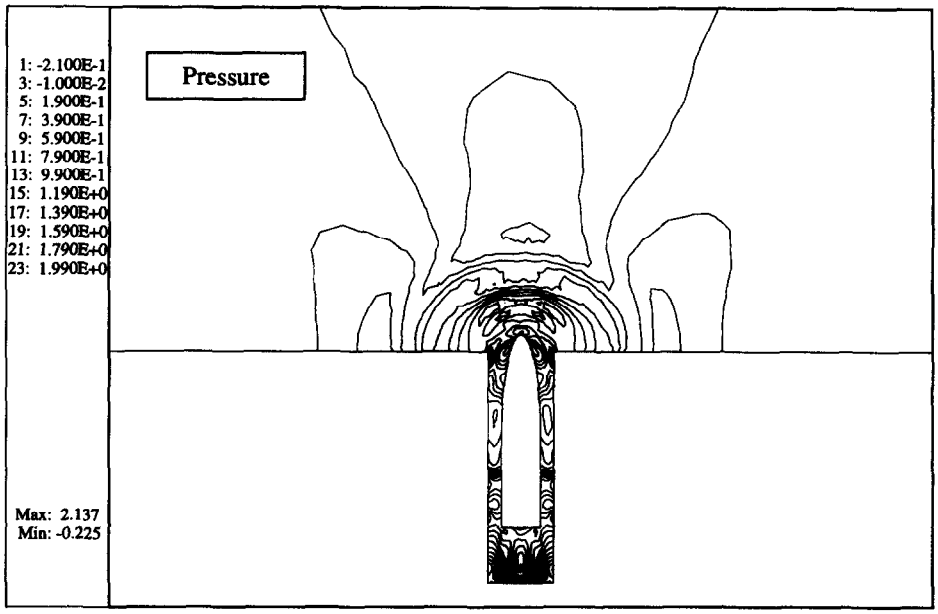


(b)

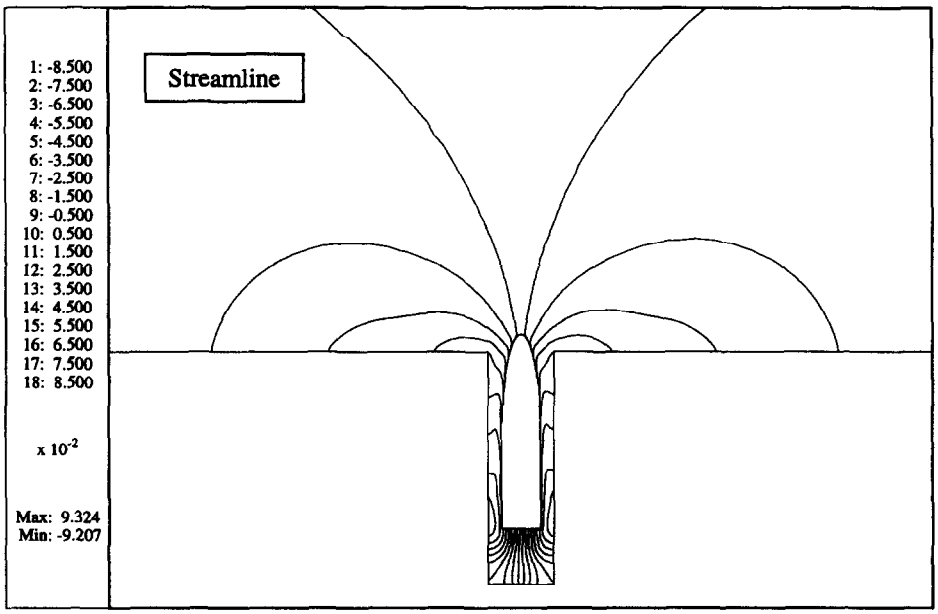
Fig. 28. Flow field around moving missile (9517 elements).

triangles with 2624 nodes representing the fluid subdomain and a structured mesh of 400 quadrilaterals with 451 nodes representing the solid subdomain. An important feature of the mesh is that at the fluid–structure interface, the fluid mesh has a much higher resolution than the solid mesh, and the fluid nodes do not necessarily coincide with the solid nodes.

We employed an implicit, third-order accurate predictor multicorrector algorithm for the fluid subdomain. For the structural subdomain we used the HHT-α method with $\alpha = -0.1$ which results in



(a)

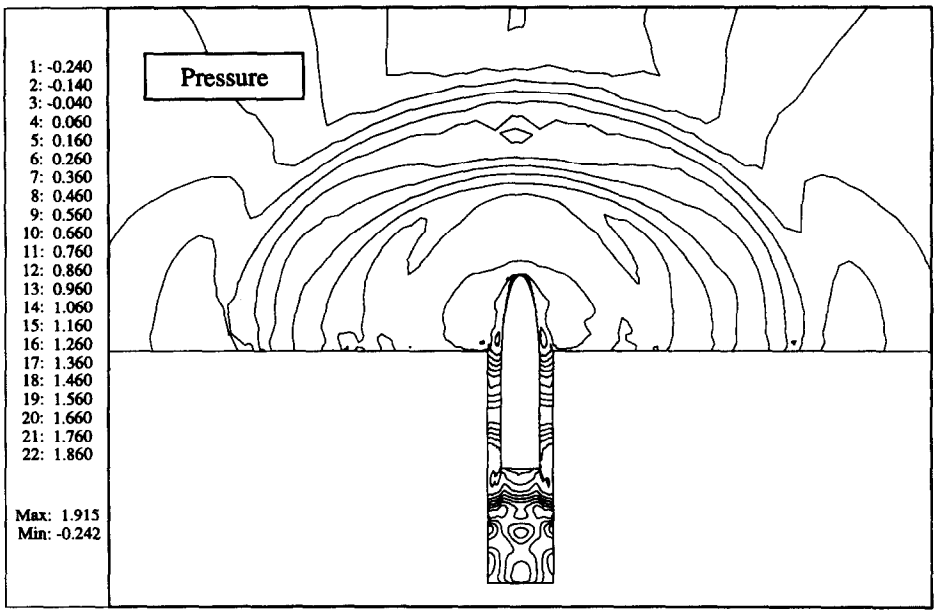


(b)

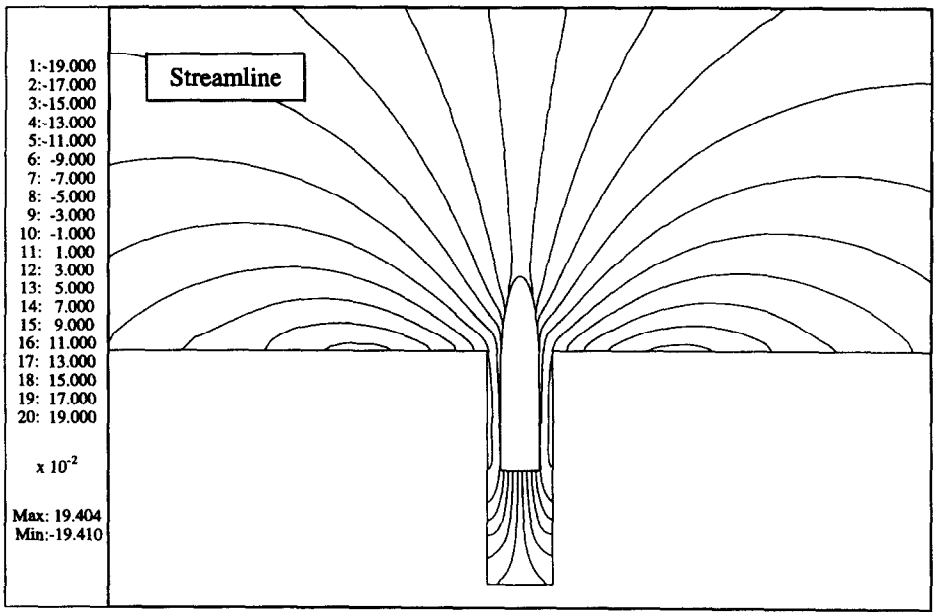
Fig. 29. Flow field around moving missile (10 139 elements).

an unconditionally stable, second-order accurate scheme (see [20, p. 532]). The staggered solution procedure in which the two integrators are interfaced through the coupling terms and the integration process is carried out in alternating stages can be summarized as follows.

Step 1. Solve the structural problem subject to fluid pressure along the interface and obtain the displacement, velocity and stress fields.



(a)



(b)

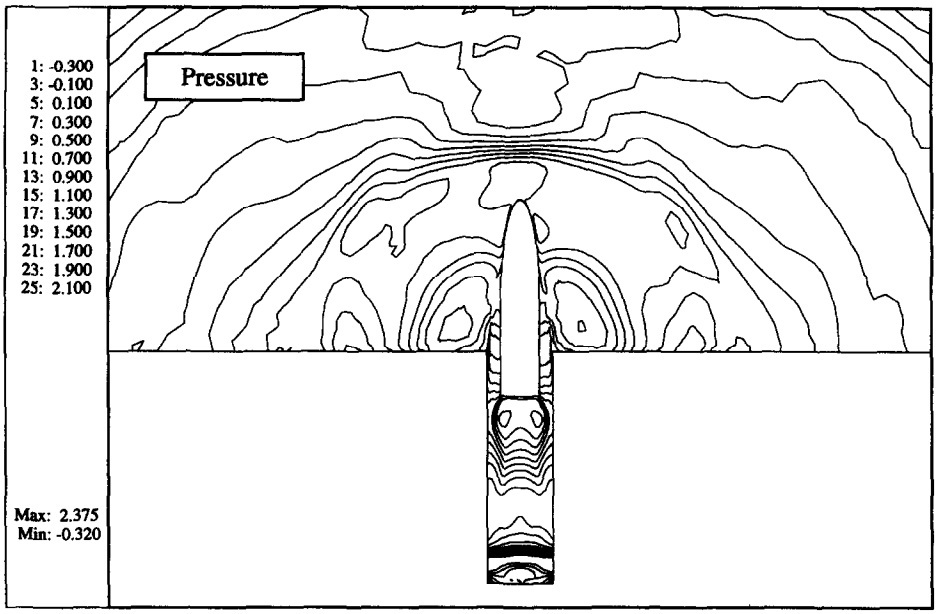
Fig. 30. Flow field around moving missile (11 776 elements).

Step 2. Use the mesh rezoning technique to update the fluid mesh.

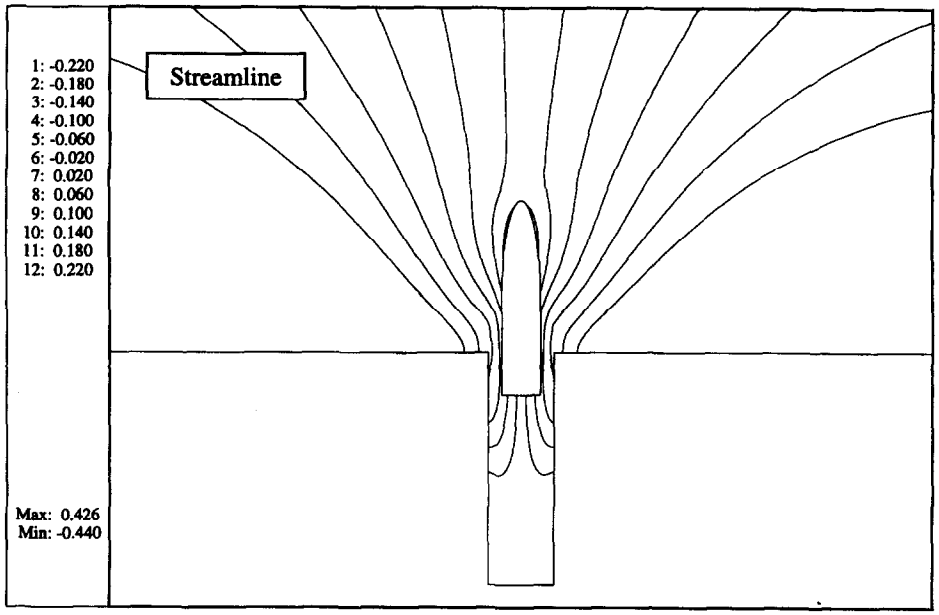
Step 3. Generate the space–time slab by using the updated configuration of the spatial mesh at time level t_{n+1} and the previous configuration at level t_n.

Step 4. Impose the boundary conditions and solve for the unknown field variables in the fluid subdomain.

Step 5. Update the pressure field along the interface, increment time t and go to Step 1.



(a)

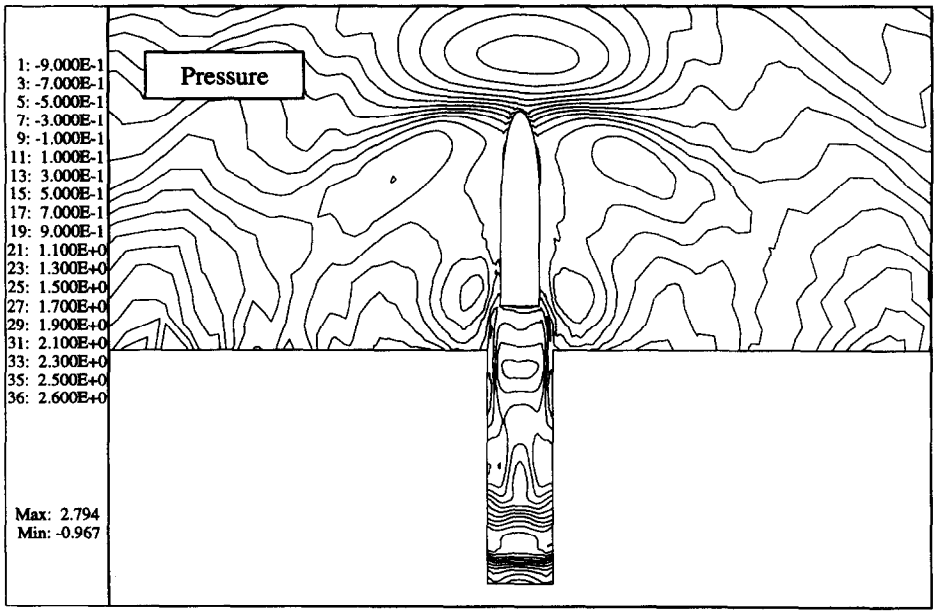


(b)

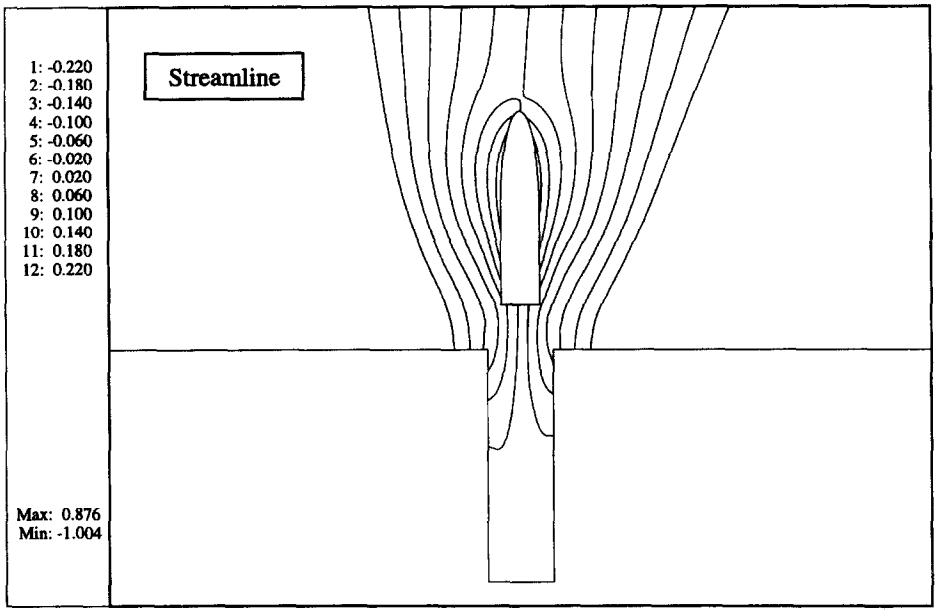
Fig. 31. Flow field around moving missile (14 532 elements).

We divided this simulation into three stages. In the initial stage the structure is stationary and fluid flows over it, developing a boundary layer. In the second stage a wave is introduced at the lower surface of the structure via a sinusoidal driving force

$$f(x, t) = F(x) \sin(\omega t) \tag{9.1}$$



(a)



(b)

Fig. 32. Flow field around moving missile (15 588 elements).

where ω is the angular frequency. The wavelength is spread over sixteen elements. This periodic force produces a transverse wave at the fluid–structure interface with displacements perpendicular to the propagation direction.

Fig. 35 shows the complete view of the spatial configurations one half cycle apart. Fig. 36 shows a

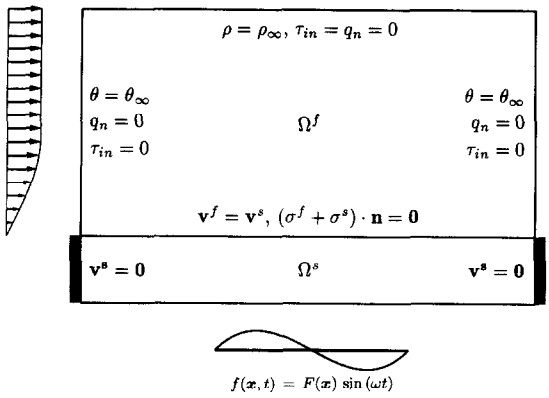


Fig. 33. Details of problem description.

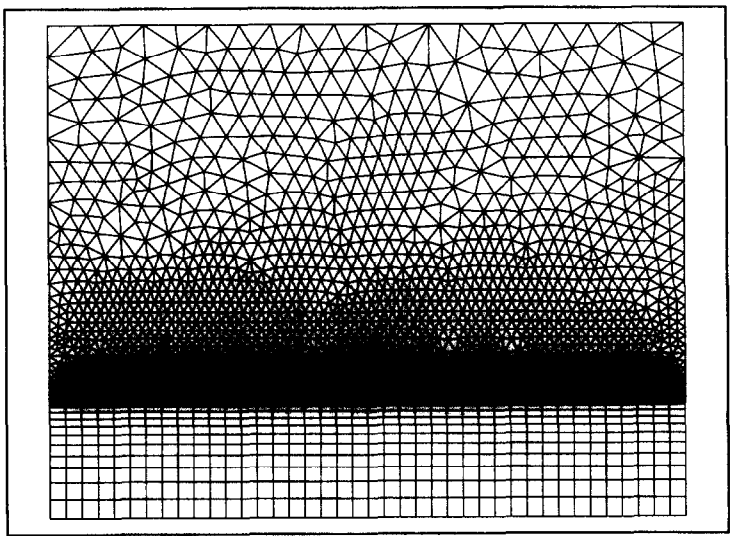
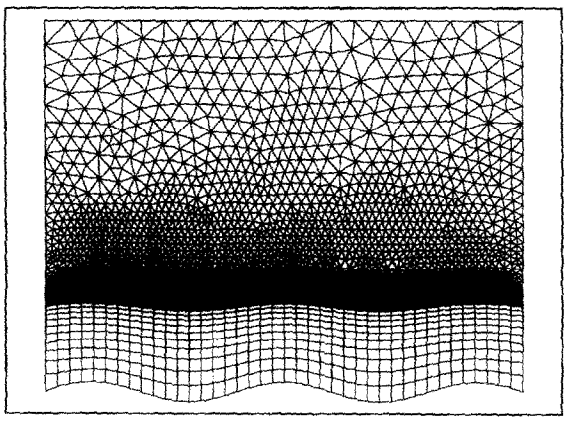
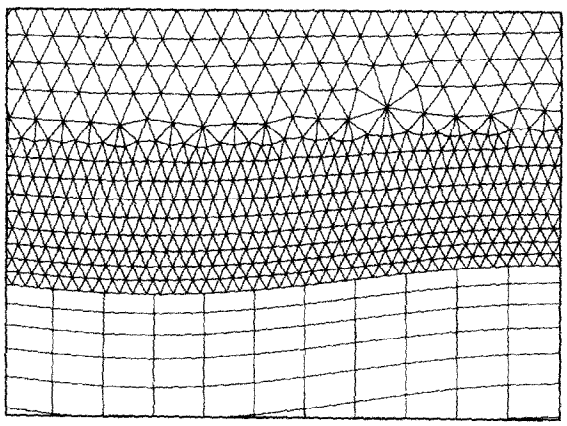


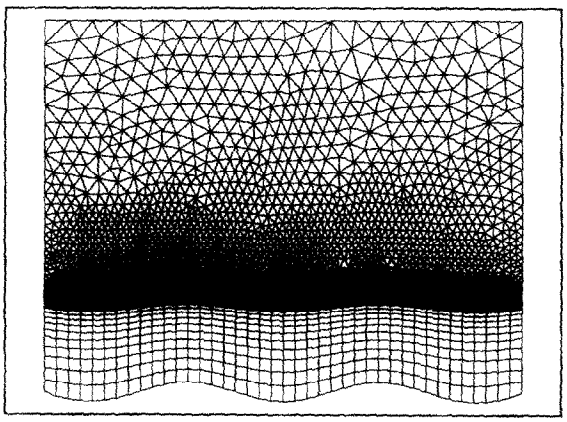
Fig. 34. Complete view of the initial mesh.

zoomed view of the coupled displacement field where the surface response of the solid is equated to the nodal response of the fluid via the mesh movement technique. It can be seen that the smaller fluid elements in the boundary layer region maintain their shape throughout the simulation. Figs. 37 presents the close-up of the coupled velocity field half cycle apart and Fig. 38 shows the streamlines in the fluid domain in the presence of the mean background flow.

In the third stage, the fluid inflow is turned off while the structure is still driven by $f(x, t)$. After some time the mean flow becomes insignificantly small. Fig. 39 shows the coupled velocity field generated by the moving interface. The amplitude of the transmitted wave decays with depth in the fluid medium, Fig. 39, until the mean flow almost becomes stationary in the far field. Fig. 40 shows the streamlines generated by the wall motion in a zero background flow.







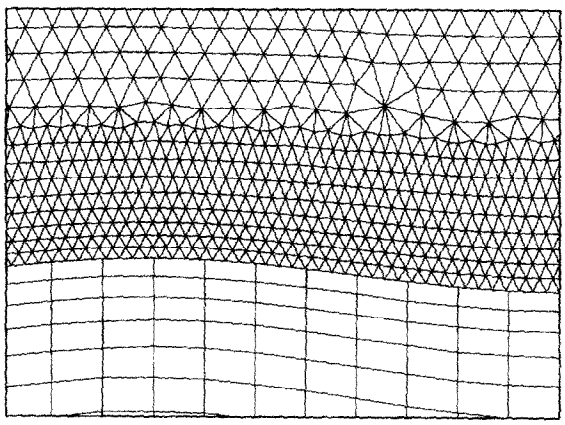


Fig. 35. Displacement field half cycle apart.

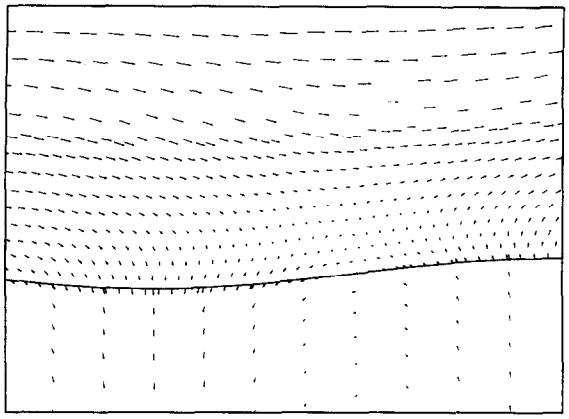
Fig. 36. Close-up of displacement field half cycle apart.

The energy transmitted to the fluid domain gives rise to a surface pressure wave shown in Fig. 41. The solid horizontal line represents the mean pressure in the absence of the wall motion. Also plotted are the surface pressures at the end of a typical cycle for the two flow cases.

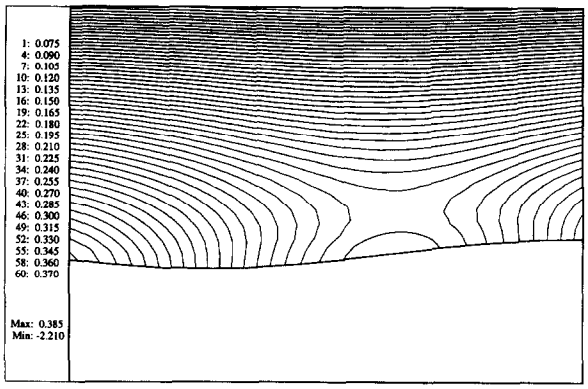
9.5. Seismic response of liquid-filled tank

This test case represents long-time response of a fluid-tank system with small structural displacements. Fig. 42 shows the finite element model and the dimensions of the 2D anchored system. The fluid is assumed to be homogeneous, isotropic and inviscid. The density of the fluid is 1×10^3 kg/m^3 and its bulk modulus is 1×10^9 Pa. Gravity is 9.8 m/s. Slip boundary conditions are enforced at the interface. The tank walls are modeled with elasticity elements with in-plane rotational degrees of freedom [38]. These elements are stacked in five layers through the thickness to improve the bending response of the shell wall and plane strain conditions are assumed enforced.

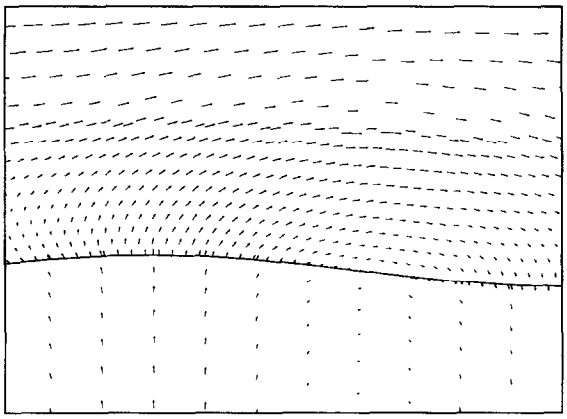
A 10 second duration modified North–South component of the 1940 El-Centro acceleration time



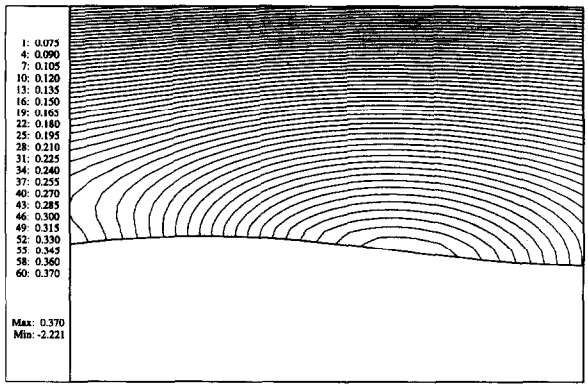
(a)



(a)



(b)



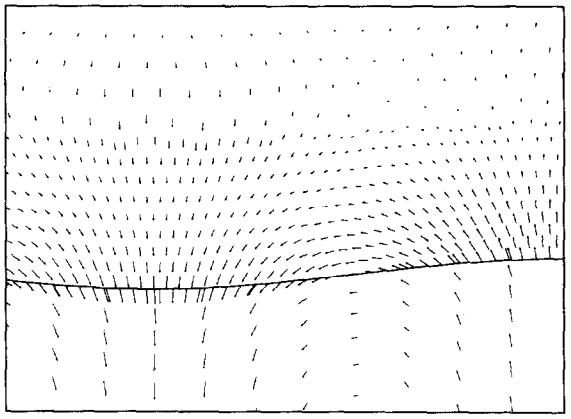
(b)

Fig. 37. Coupled velocity field in the presence of mean background flow.

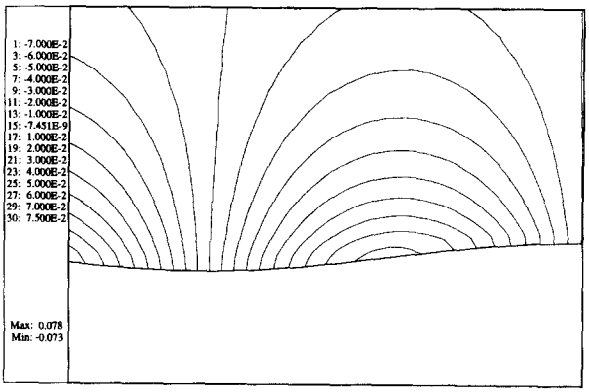
Fig. 38. Streamlines in the presence of mean background flow.

history (Fig. 43) is applied at the base of the tank in the horizontal direction. The time step increment, also used in [35], is $\Delta t = 0.02$ s.

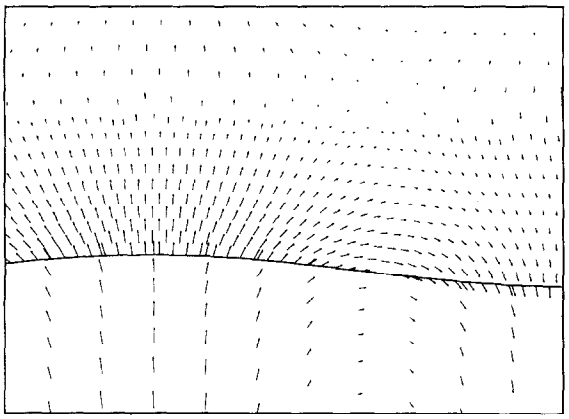
The staggered solution algorithm described in the previous numerical simulation is employed here. The maximum dimension of the Krylov space is 10 with 3 corrector passes in the predictor corrector algorithm. Two numerical simulations have been performed. In the first simulation the convective effects are ignored to get a comparison with the results in the literature [35]. A plot of the sloshing motion at the free surface is presented in Fig. 44. Node 57 is the fluid node belonging to the free-surface which also lies on the fluid–solid interface. The results compare well with those of Liu et al. [35]. Node 59 is an interior node on the free surface. The sloshing frequency of the coupled system is obtained by counting the number of oscillations per second in the sloshing wave plots. The second simulation is shown in Fig. 45, where the convective effects have been accounted for. The ratio of the free surface oscillation to the total depth of the tank is about 8% which makes the convective effects important close to the free surface.



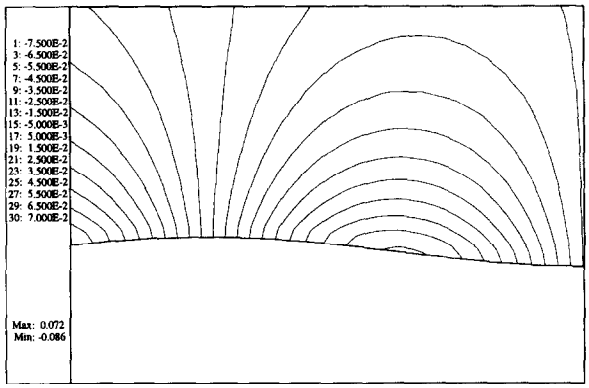
(a)



(a)



(b)



(b)

Fig. 39. Coupled velocity field in zero background flow.

Fig. 40. Streamlines in zero background flow.

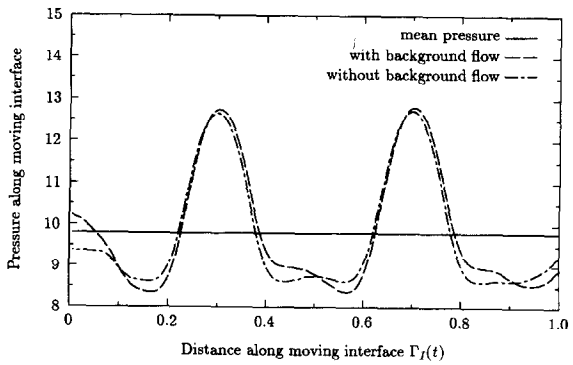
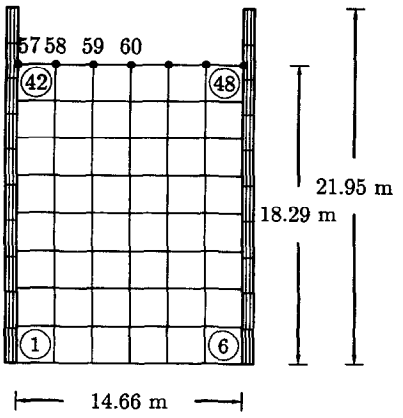


Fig. 41. Pressure wave generation along the moving wall.



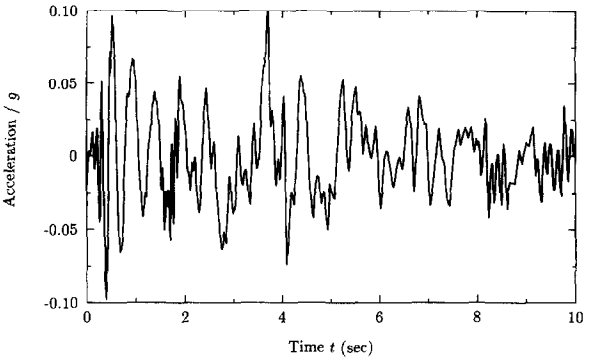
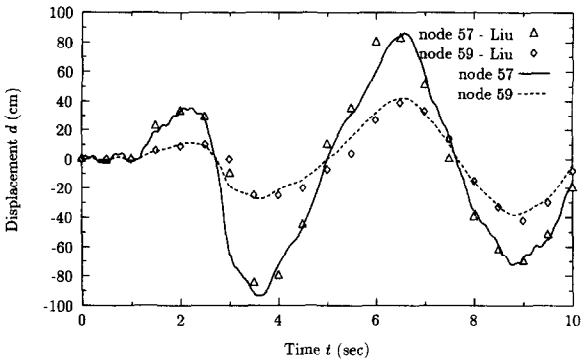


Fig. 42. 2D fluid-tank model.

Fig. 43. Modified N–S component of El-Centro earthquake.



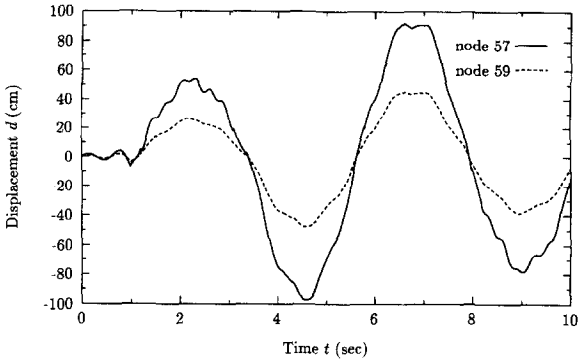


Fig. 44. Sloshing wave height plot. Convective effects not included ($f = 5$ Hz).

Fig. 45. Sloshing wave height plot. Convective effects included ($f = 5$ Hz).

10. Conclusions

In this paper we have presented a space–time Galerkin/least-squares finite element formulation of the Navier–Stokes equations, suitable for fluid flow problems that involve significant changes in the spatial configuration. It is stable for a wide range in Reynolds numbers and allows equal-order interpolation for velocity and pressure. The formulation is shown to be analogous to the classical arbitrary Lagrangian–Eulerian techniques. However, the advantage here is that by using the discontinuous Galerkin method in time, a new mesh can be constructed under circumstances of severe mesh deformations and the solution can be projected onto the new mesh in a variationally consistent manner.

We have also presented an adaptive mesh rezoning technique which is appropriate for arbitrarily shaped domains and accommodates triangular and quadrilateral elements. This strategy is particularly well-suited for implementation in a parallel processing environment because it maintains the connectivity of the mesh. Consequently, the mapping of the data to the processing nodes of the parallel computers remains fixed throughout the computations.

In this work we have adopted a continuum based approach for the coupling of fluid and structural subdomains. The approach is general in the sense that it allows for different discretization and solution strategies in the two subdomains. Efficient solution of the resulting equations of motion is achieved by a modular computer implementation in which separate fluid and structural analyzers are interfaced through coupling terms. This strategy offers the important advantage of preserving the program modularity and thus the general-purpose applicability and overall flexibility of the finite element method. Various numerical simulations are presented to show that the proposed formulation is suitable for a range of coupled transient problems that involve considerable changes in the spatial configuration.

Acknowledgments

This research was supported by the U.S. Office of Naval Research under Contract N00014-88-K-0446. The authors gratefully acknowledge Prof. T.E. Tezduyar and Dr. Farzin Shakib for their helpful comments, Dr. Zdeněk Johan for numerous discussions about the finite element program ENSA, Dr. Frédéric Chalot for the use of his adaptive mesh generator and Dr. Kenneth Jansen for his projection algorithm.

References

[1] A. Acrivos, LG. Leal, D.S. Snowden and F. Pan, Further experiments on steady separated flows past bluff objects, J. Fluid Mech. 34 (1968) 25–48.
[2] S.K. Aliabadi and T.E. Tezduyar, Space–time finite element computation of compressible flows involving moving boundaries and interfaces, Comput. Methods Appl. Mech. Engrg. 107 (1993) 209–223.
[3] M. Behr, A.A. Johnson, J. Kennedy, S. Mittal and T.E. Tezduyar, Computation of incompressible flows with implicit finite element implementations on the connection machine, Comput. Methods Appl. Mech. Engrg. 108 (1993) 99–118.
[4] M. Behr and T.E. Tezduyar, Finite element solution strategies for large-scale flow simulations, Comput. Methods Appl. Mech. Engrg. 112 (1994) 3–24.
[5] T. Belytschko, J.M. Kennedy and D.F. Schoeberie, Quasi-Eulerian finite element formulation for fluid–structure interaction, ASME J. Pressure Vessel Technol. 102 (1980) 62–69.
[6] J.U. Brackbill and J.S. Saltzman, Adaptive zoning for singular problems in two dimensions, J. Comput. Phys. 46 (1982) 342–368.
[7] A.N. Brooks and T.J.R. Hughes, Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations, Comput. Methods Appl. Mech. Engrg. 32 (1982) 199–259.
[8] F. Chalot, T.J.R. Hughes and F. Shakib, Symmetrization of conservation laws with entropy for high-temperature hypersonic computations, Comput. Syst. Engrg. 1 (1990) 495–521.
[9] J.S. Chen, W.K. Liu and T. Belytschko, Arbitrary Lagrangian–Eulerian methods for materials with memory and friction, in: T.E. Tezduyar and T.J.R. Hughes, eds., Recent Developments in Computational Fluid Dynamics, AMD-Vol. 95 (1988).
[10] J. Donea, Arbitrary Lagrangian–Eulerian finite element methods, in: T. Belytschko and T.J.R. Hughes, eds., Computational Methods for Transient Analysis (North-Holland, Amsterdam, 1983) 473–516.
[11] J. Donea, P. Fasoli-Stella and S. Giuliani, Lagrangian and Eulerian finite element techniques for transient fluid–structure interaction problems, in: Transactions of the 4th International Conference on Structural Mechanics in Reactor Technology, Paper B1/2, 1977.
[12] J. Donea, S. Giuliani and J.P. Halleux, An arbitrary Lagrangian–Eulerian finite element method for transient dynamic fluid–structure interactions, Comput. Methods Appl. Mech. Engrg. 33 (1982) 689–723.
[13] R.M. Ferencz and T.J.R. Hughes, Iterative Finite Element Solutions in Nonlinear Solid Mechanics, in: P.G. Ciarlet and J.L. Lions, eds., Handbook of Numerical Analysis, Vol. V: Numerical Methods for Solids (Elsevier, Amsterdam), in press.
[14] L.P. Franca and T.J.R. Hughes, Two classes of mixed finite element methods, Comput. Methods Appl. Mech. Engrg. 69 (1988) 89–129.
[15] R. Glowinski and P.L. Tallec, Augmented Lagrangian and Operator-Splitting Methods in Nonlinear Mechanics, SIAM Studies in Applied Mathematics, Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania, 1989.
[16] A.S. Grove, F.H. Shair, E.E. Petersen and A. Acrivos, An experimental investigation of the steady separated flow past a circular cylinder, J. Fluid Mech. 19 (1964) 60–80.
[17] P. Hansbo, The characteristic streamline diffusion method for the time-independent incompressible Navier–Stokes equations, Comput. Methods Appl. Mech. Engrg. 99 (1992) 171–186.
[18] 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.
[19] T.J.R. Hughes, Recent progress in the development and understanding of SUPG methods with special reference to the compressible Euler and Navier–Stokes equations, Int. J. Numer. Methods Engrg. 7 (1987) 1261–1275.
[20] T.J.R. Hughes, The Finite Element Method: Linear Static and Dynamic Finite Element Analysis (Prentice-Hall, Englewood Cliffs, NJ, 1987).
[21] T.J.R. Hughes and A.N. Brooks, A multi-dimensional upwind scheme with no crosswind diffusion, in: T.J.R. Hughes, ed., Finite Element Methods for Convection Dominated Flows, AMD-Vol. 95 (ASME, New York, 1979) 19–35.
[22] T.J.R. Hughes and L.P. Franca, A new finite element formulation for computational fluid dynamics: VII. The Stokes problem with various well-posed boundary conditions: symmetric formulations that converge for all velocity/pressure spaces, Comput. Methods Appl. Mech. Engrg. 65 (1987) 85–96.
[23] T.J.R. Hughes, L.P. Franca and M. Balestra, A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuška–Brezzi condition: a stable Petrov–Galerkin formulation of the Stokes problem accommodating equal-order interpolations, Comput. Methods Appl. Mech. Engrg. 59 (1986) 85–99.

[24] 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.
[25] T.J.R. Hughes, L.P. Franca and M. Mallet, A new finite element formulation for computational fluid dynamics: I. Symmetric forms of the compressible Euler and Navier–Stokes equations and the second law of thermodynamics, Comput. Methods Appl. Mech. Engrg. 54 (1986) 223–234.
[26] 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.
[27] T.J.R. Hughes and M. Mallet, A new finite element formulation for computational fluid dynamics: III. The generalized streamline operator for multidimensional advective–diffusive systems, Comput. Methods Appl. Mech. Engrg. 58 (1986) 305–328.
[28] T.J.R. Hughes, W.K. Liu and T.K. Zimmerman, Lagrangian–Eulerian finte element formulation for incompressible viscous flows, Comput. Methods Appl. Mech. Engrg. 29 (1984) 329–349.
[29] G.M. Hulbert and T.J.R. Hughes, Space–time finite element methods for second-order hyperbolic equations, Comput. Methods Appl. Mech. Engrg. 84 (1990) 327–348.
[30] K. Jansen, F. Shakib and T.J.R. Hughes, Fast projection algorithm for unstructured meshes, in: S.N. Atluri, ed., Computational Nonlinear Mechanics in Aerospace Engineering (AIAA, Washington D.C., 1992).
[31] C. Johnson and J. Saranen, Streamline diffusion methods for the incompressible Euler and Navier–Stokes equations, Math. Comput. 47 (1986) 1–18.
[32] C. Johnson, A. Szepessy and P. Hansbo, On the convergence of shock-capturing streamline diffusion finite element methods for hyperbolic conservation laws, Math. Comput. 54 (1990) 107–129.
[33] A.A. Johnson and T.E. Tezduyar, Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces, Comput. Methods Appl. Mech. Engrg. 119 (1994) 73–94.
[34] W.K. Liu, H. Chang, J. Chen and T. Belytschko, Arbitrary Lagrangian–Eulerian Petrov–Galerkin finite elements for nonlinear continua, Comput. Methods Appl. Mech. Engrg. 68 (1988) 259–310.
[35] W.K. Liu and D.C. Ma, Coupling effect between liquid sloshing and flexible fluid-filled systems, Nucl. Engrg. Des. 72 (1982) 345–357.
[36] M. Mallet, A finite element method for computational fluid dynamics, Ph.D. Thesis, Stanford University, August 1985.
[37] J.E. Marsden and T.J.R. Hughes, Mathematical Foundations of Elasticity (Prentice-Hall, Englewood Cliffs, NJ, 1983).
[38] A. Masud, A space–time finite element method for fluid–structure interaction, Ph.D. Thesis, Stanford University, April 1993.
[39] S. Mittal and T.E. Tezduyar, Massively parallel finite element computation of incompressible flows involving fluid-body interactions, Comput. Methods Appl. Mech. Engrg. 112 (1994) 253–282.
[40] B. Ramaswamy and M. Kawahara, Arbitrary Lagrangian–Eulerian finite element method for unsteady, convective, incompressible viscous free surface fluid flow, Int. J. Numer. Methods Engrg. 7 (1987) 1053–1075.
[41] F. Shakib, T.J.R. Hughes and Z. Johan, A multi-element group preconditioned GMRES algorithm for nonsymmetric systems arising in finite element analysis, Comput. Methods Appl. Mech. Engrg. 75 (1989) 415–465.
[42] F. Shakib, T.J.R. Hughes and Z. Johan, A new finite element formulation for computational fluid dynamics: X. The compressible Euler and Navier–Stokes equations, Comput. Methods Appl. Mech. Engrg. 89 (1991) 141–219.
[43] A. Soulaimani, M. Fortin, G. Dhatt and Y. Ouellet, Finite element simulation of two- and three-dimensional free surface flows, Comput. Methods Appl. Mech. Engrg. 86 (1991) 265–296.
[44] T.E. Tezduyar, S.K. Aliabadi, M. Behr and S. Mittal, Massively parallel finite element simulation of compressible and incompressible flows, Comput. Methods Appl. Mech. Engrg. 119 (1994) 157–177.
[45] T.E. Tezduyar, M. Behr, S. Mittal and A.A. Johnson, Computation of unsteady incompressible flows with the stabilized finite element methods: Space–time formulations, iterative strategies and massively parallel implementation, New Methods in Transient Analysis, AMD-Vol. 143 (ASME, 1992).
[46] T.E. Tezduyar, J. Liou and M. Behr, A new strategy for finite element computations involving moving boundaries and interfaces—the DSD/ST procedure: I. The concept and the preliminary numerical tests, Comput. Methods Appl. Mech. Engrg. 94 (1992) 339–351.
[47] T.E. Tezduyar, J. Liou and M. Behr, A new strategy for finite element computations involving moving boundaries and interfaces—the DSD/ST procedure: II. Computation of free-surface flows, two-liquid flows, and flows with drifting cylinders, Comput. Methods Appl. Mech. Engrg. 94 (1992) 353–371.
[48] A.M. Winslow, Equipotential zoning of two-dimensional meshes, University of California, Lawrence Radiation Laboratory Report, UCRL-7312, 1963.