Available online at www.sciencedirect.com
SCIENCE DIRECT®
Comput. Methods Appl. Mech. Engrg. 195 (2006) 5743–5753
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

Solution techniques for the fully discretized equations in computation of fluid–structure interactions with the space–time formulations

Tayfun E. Tezduyar [a,*], Sunil Sathe [a], Keith Stein [b]

[a] Mechanical Engineering, Rice University—MS 321 6100 Main Street, Houston, Texas 77005, USA
[b] Department of Physics, Bethel University, 3900 Bethel Drive, St. Paul, MN 55112, USA

Received 4 March 2005; received in revised form 26 August 2005; accepted 26 August 2005

Abstract

We provide an overview of the solution techniques we have developed for the fully discretized equations encountered at every time step in computation of fluid–structure interactions with the space–time techniques. These coupled, nonlinear equations are generated from the finite element discretization of the governing equations for the fluid mechanics, structural mechanics and the motion of the fluid mechanics mesh. The fluid mechanics equations are discretized with the deforming-spatial-domain/stabilized space–time formulation. The mesh motion is governed by the equations of elasticity, with the smaller elements stiffened in the finite element formulation. The coupled, fully discretized equations are solved with the block-iterative, quasi-direct and direct coupling methods. We present numerical examples with incompressible flows and membrane and cable structures.

Keywords: Space–time finite elements; Fluid–structure interactions; FSI; Iterative and direct coupling techniques; Light structures

1. Introduction

Numerical modeling of fluid–structure interactions (FSI) offers a variety of challenges in computational engineering. The spatial domain occupied by the fluid changes in time as the fluid–structure interface moves, and the mathematical model has to be able to handle that. Accurate representation of the flow field near the fluid–structure interface requires that the mesh be updated to track the interface, and this requires special attention in 3D problems with complex geometries. Solution of the coupled fluid and structural mechanics and mesh moving equations offers additional challenges, depending on the type of the problem solved. Different aspects of the challenges involved in FSI computations attracted much attention, especially in recent years, and a good number of papers were published addressing these challenges. As examples of the papers published in recent years on FSI and related and supporting topics, see [1–11]. Additional examples can be found in [12].

In this paper, we focus on the solution techniques we developed for the coupled, nonlinear equations generated from the finite element discretization of the governing equations for the fluid mechanics, structural mechanics and the motion of the fluid mechanics mesh. These fully discretized equations need to be solved at every time step in computation of a fluid–structure interaction problem with the space–time techniques. We emphasize solution techniques for light structures.

In our FSI computations the fluid mechanics equations are discretized with the deforming-spatial-domain/stabilized space–time (DSD/SST) formulation [13–15]. This formulation was originally introduced as a general-purpose interface-tracking technique for flow problems with moving boundaries and interfaces. In an interface-tracking technique the mesh

* Corresponding author. Tel.: +1 713 348 6051; fax: +1 713 348 5423.
E-mail addresses: tezduyar@rice.edu (T.E. Tezduyar), sathe@rice.edu (S. Sathe), k-stein@bethel.edu (K. Stein).

0045-7825/$ - see front matter
doi:10.1016/j.cma.2005.08.023

moves to track the fluid–solid and fluid–fluid interfaces. The DSD/SST method is based on stabilized finite element formulations, which are written over the space–time domain of a fluid mechanics problem. The stabilized formulations are the Streamline-Upwind/Petrov–Galerkin (SUPG) [16–19] and Pressure-Stabilizing/Petrov–Galerkin (PSPG) [13,20] formulations. The SUPG formulation prevent numerical instabilities that might be encountered when we have high Reynolds or Mach numbers and shocks and strong boundary layers. With the PSPG formulation, we can use, without numerical instabilities, equal-order interpolation functions for velocity and pressure. An earlier version of the pressure stabilization, for Stokes flows, was introduced in [21]. The stabilized space–time formulations were used earlier by other researchers to solve problems with fixed spatial domains (see for example [22]). The space–time computations are carried out for one space–time "slab" at a time, where the "slab" is the slice of the space–time domain between the time levels n and $n+1$. With this approach, a 3D computational problem is spared from becoming a 4D problem including the time dimension. Some additional special features are exploited in the Special DSD/SST (S-DSD/SST) formulation [23,24] to make the calculation of the element-level vectors and matrices more efficient.

How the mesh is updated depends on several factors, including the complexity of the fluid–structure interface and overall geometry, how unsteady the interface is, and how the starting mesh was generated. The mesh update in our FSI computations is accomplished with an automatic mesh moving method [25,26], where the motion of the nodes is governed by the equations of elasticity. As the boundary condition, the motion of the nodes at the interfaces is specified to match the normal velocity of the fluid. In [25] the mesh deformation is dealt with selectively based on the sizes of the elements. Mesh moving techniques with comparable features were later introduced in [27]. In the technique introduced in [25], selective treatment based on element sizes is attained by altering the way we account for the Jacobian of the transformation from the element domain to the physical domain. The objective is to stiffen the smaller elements, which are typically placed near solid surfaces, more than the larger ones. When the mesh becomes too distorted, a full or partial remeshing takes place. We introduced a number of enhancements to this general-purpose mesh moving technique, including the Solid-Extension Mesh Moving Technique (SEMMT) [28,29]. The SEMMT was introduced to address challenges involved in moving a mesh with very thin fluid elements near the solid surfaces.

In this paper, we describe the three techniques we developed for solving the fully discretized equations of fluid and structural mechanics and mesh motion, which need to be solved in their coupled form. They are: block-iterative coupling, which we have widely used in our earlier computations (see [30–33]); quasi-direct coupling [23,24]; and direct coupling [23,24]. The direct coupling approach is based on the mixed analytical/numerical element-vector-based (AEVB/NEVB) computation technique that was introduced in [26,34]. We focus on FSI problems where the fluid mechanics is governed by the Navier–Stokes equations of incompressible flows and the structural mechanics, where the structures are light, is governed by the membrane and cable equations.

These governing equations are reviewed in Section 2. The finite element formulations, including the DSD/SST formulation, are described in Section 3. In Section 4 we describe the block-iterative, quasi-direct, and direct coupling techniques. Numerical examples are presented in Section 5.

2. Governing equations

2.1. Fluid mechanics

Let $\Omega_t \subset \mathbb{R}^{n_{sd}}$ be the spatial domain with boundary Γ_t at time $t \in (0,T)$. The subscript t indicates the time-dependence of the domain. The Navier–Stokes equations of incompressible flows are written on Ω_t and $\forall t \in (0,T)$ as

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u} - \mathbf{f}\right) - \nabla\cdot\boldsymbol{\sigma} = \mathbf{0}, \tag{1}$$

$$\nabla\cdot\mathbf{u} = 0, \tag{2}$$

where ρ, $\mathbf{u}$ and $\mathbf{f}$ are the density, velocity and the external force, respectively. The stress tensor $\boldsymbol{\sigma}$ is defined as $\boldsymbol{\sigma}(p,\mathbf{u}) = -p\mathbf{I} + 2\mu\boldsymbol{\varepsilon}(\mathbf{u})$, with $\boldsymbol{\varepsilon}(\mathbf{u}) = ((\nabla\mathbf{u}) + (\nabla\mathbf{u})^{\mathrm{T}})/2$. Here p is the pressure, $\mathbf{I}$ is the identity tensor, $\mu = \rho\nu$ is the viscosity, ν is the kinematic viscosity, and $\boldsymbol{\varepsilon}(\mathbf{u})$ is the strain-rate tensor. The essential and natural boundary conditions for Eq. (1) are represented as $\mathbf{u} = \mathbf{g}$ on $(\Gamma_t)_g$ and $\mathbf{n}\cdot\boldsymbol{\sigma} = \mathbf{h}$ on $(\Gamma_t)_h$, where $(\Gamma_t)_g$ and $(\Gamma_t)_h$ are complementary subsets of the boundary Γ_t, $\mathbf{n}$ is the unit normal vector, and $\mathbf{g}$ and $\mathbf{h}$ are given functions. A divergence-free velocity field $\mathbf{u}_0(\mathbf{x})$ is specified as the initial condition.

2.2. Structural mechanics

Let $\Omega_t^{\mathrm{s}} \subset \mathbb{R}^{n_{xd}}$ be the spatial domain with boundary Γ_t^{s}, where $n_{xd} = 2$ for membranes and $n_{xd} = 1$ for cables. The parts of Γ_t^{s} corresponding to the essential and natural boundary conditions are represented by $(\Gamma_t^{\mathrm{s}})_g$ and $(\Gamma_t^{\mathrm{s}})_h$. The superscript "s" indicates the structure. The equations of motion are written as

$$\rho^{\mathrm{s}}\left(\frac{\mathrm{d}^2\mathbf{y}}{\mathrm{d}t^2}+\eta\frac{\mathrm{d}\mathbf{y}}{\mathrm{d}t}-\mathbf{f}^{\mathrm{s}}\right)-\nabla\cdot\boldsymbol{\sigma}^{\mathrm{s}}=\mathbf{0}, \tag{3}$$

where ρ^{s}, $\mathbf{y}$, $\mathbf{f}^{\mathrm{s}}$ and $\boldsymbol{\sigma}^{\mathrm{s}}$ are the material density, structural displacement, external force and the Cauchy stress tensor [35,36], respectively. Here η is the mass-proportional damping coefficient. The damping provides additional stability and can be used where time-accuracy is not required, such as in determining the deformed shape of the structure for specified fluid mechanics forces acting on it. The stresses are expressed in terms of the 2nd Piola–Kirchoff stress tensor $\mathbf{S}$, which is related to the Cauchy stress tensor through a kinematic transformation. Under the assumption of large displacements and rotations, small strains, and no material damping, the membranes and cables are treated as Hookean materials with linear elastic properties. For membranes, under the assumption of plane stress, $\mathbf{S}$ becomes (see [37]).

$$S^{ij}=(\bar{\lambda}_m G^{ij}G^{kl}+\mu_m[G^{il}G^{jk}+G^{ik}G^{jl}])E_{kl}, \tag{4}$$

where for the case of isotropic plane stress $\bar{\lambda}_m=2\lambda_m\mu_m/(\lambda_m+2\mu_m)$. Here, E_{kl} are the components of the Cauchy–Green strain tensor, G^{ij} are the components of the contravariant metric tensor in the original configuration, and λ_m and μ_m are the Lamé constants. For cables, under the assumption of uniaxial tension, $\mathbf{S}$ becomes $S^{11}=E_{\mathrm{c}}G^{11}G^{11}E_{11}$, where E_{c} is the Young's modulus for the cable. To account for stiffness-proportional material damping, the Hookean stress–strain relationships defined by Eq. (4) and its version for cables are modified, and E_{kl} is replaced by $\widehat{E}_{kl}$, where $\widehat{E}_{kl}=E_{kl}+\zeta\dot{E}_{kl}$. Here ζ is the stiffness-proportional damping coefficient and $\dot{E}_{kl}$ is the time derivative of E_{kl}.

3. Finite element formulations

3.1. DSD/SST formulation of fluid mechanics

In the DSD/SST method [13–15], the finite element formulation is written over a sequence of N space–time slabs Q_n, where Q_n is the slice of the space–time domain between the time levels t_n and t_{n+1}. At each time step, the integrations are performed over Q_n. The space–time finite element interpolation functions are continuous within a space–time slab, but discontinuous from one space–time slab to another. The notation $(\cdot)_n^-$ and $(\cdot)_n^+$ denotes the function values at t_n as approached from below and above. Each Q_n is decomposed into elements Q_n^e, where $e=1,2,\ldots,(n_{el})_n$. The subscript n used with n_{el} is for the general case in which the number of space–time elements may change from one space–time slab to another. The essential and natural boundary conditions are imposed over $(P_n)_{\mathrm{g}}$ and $(P_n)_{\mathrm{h}}$, the complementary subsets of the lateral boundary of the space–time slab. The finite element trial function spaces $(\mathscr{S}_{\mathbf{u}}^h)_n$ for velocity and $(\mathscr{S}_p^h)_n$ for pressure, and the test function spaces $(\mathscr{V}_{\mathbf{u}}^h)_n$ and $(\mathscr{V}_p^h)_n$ $(=(\mathscr{S}_p^h)_n)$ are defined by using, over Q_n, first-order polynomials in space and time. The DSD/SST formulation is written as follows: given $(\mathbf{u}^h)_n^-$, find $\mathbf{u}^h\in(\mathscr{S}_{\mathbf{u}}^h)_n$ and $p^h\in(\mathscr{S}_p^h)_n$ such that $\forall\mathbf{w}^h\in(\mathscr{V}_{\mathbf{u}}^h)_n$ and $\forall q^h\in(\mathscr{V}_p^h)_n$

$$\begin{aligned}
&\int_{Q_n}\mathbf{w}^h\cdot\rho\left(\frac{\partial\mathbf{u}^h}{\partial t}+\mathbf{u}^h\cdot\nabla\mathbf{u}^h-\mathbf{f}^h\right)\mathrm{d}Q+\int_{Q_n}\boldsymbol{\varepsilon}(\mathbf{w}^h):\boldsymbol{\sigma}(p^h,\mathbf{u}^h)\mathrm{d}Q-\int_{(P_n)_{\mathrm{h}}}\mathbf{w}^h\cdot\mathbf{h}^h\,\mathrm{d}P+\int_{Q_n}q^h\nabla\cdot\mathbf{u}^h\mathrm{d}Q\\
&+\int_{\Omega_n}(\mathbf{w}^h)_n^+\cdot\rho((\mathbf{u}^h)_n^+-(\mathbf{u}^h)_n^-)\,\mathrm{d}\Omega+\sum_{e=1}^{(n_{el})_n}\int_{Q_n^e}\frac{1}{\rho}\left[\tau_{\mathrm{SUPG}}\rho\left(\frac{\partial\mathbf{w}^h}{\partial t}+\mathbf{u}^h\cdot\nabla\mathbf{w}^h\right)+\tau_{\mathrm{PSPG}}\nabla q^h\right]\cdot[\mathrm{Ł}(p^h,\mathbf{u}^h)-\rho\mathbf{f}^h]\,\mathrm{d}Q\\
&+\sum_{e=1}^{(n_{el})_n}\int_{Q_n^e}\nu_{\mathrm{LSIC}}\nabla\cdot\mathbf{w}^h\rho\nabla\cdot\mathbf{u}^h\,\mathrm{d}Q=0,
\end{aligned} \tag{5}$$

where

$$\mathrm{Ł}(q^h,\mathbf{w}^h)=\rho\left(\frac{\partial\mathbf{w}^h}{\partial t}+\mathbf{u}^h\cdot\nabla\mathbf{w}^h\right)-\nabla\cdot\boldsymbol{\sigma}(q^h,\mathbf{w}^h). \tag{6}$$

Here τ_{SUPG}, τ_{PSPG} and ν_{LSIC} are the SUPG, PSPG and LSIC (least-squares on incompressibility constraint) stabilization parameters. For ways of calculating τ_{SUPG}, τ_{PSPG} and ν_{LSIC} (see [38,39,34]). This formulation is applied to all space–time slabs $Q_0,Q_1,Q_2,\ldots,Q_{N-1}$, starting with $(\mathbf{u}^h)_0^-=\mathbf{u}_0$. For an earlier, detailed reference on the formulation see [13–15].

There are various ways of defining the stabilization parameters. The definitions given in [39] are used in the computations reported in this paper

$$\tau_{\mathrm{SUPG}} = \left(\frac{1}{\tau_{\mathrm{SUGN12}}^2} + \frac{1}{\tau_{\mathrm{SUGN3}}^2}\right)^{-\frac{1}{2}}, \tag{7}$$

$$\tau_{\mathrm{SUGN12}} = \left(\sum_{a=1}^{n_{en}} \left|\frac{\partial N_a}{\partial t} + \mathbf{u}^h \cdot \nabla N_a\right|\right)^{-1}, \quad \tau_{\mathrm{SUGN3}} = \frac{h_{\mathrm{RGN}}^2}{4\nu}, \tag{8}$$

$$h_{\mathrm{RGN}} = 2\left(\sum_{a=1}^{n_{en}} |\mathbf{r} \cdot \nabla N_a|\right)^{-1}, \quad \mathbf{r} = \frac{\nabla \|\mathbf{u}^h\|}{\|\nabla \|\mathbf{u}^h\|\|}, \tag{9}$$

$$\tau_{\mathrm{PSPG}} = \tau_{\mathrm{SUPG}}, \tag{10}$$

$$\nu_{\mathrm{LSIC}} = \tau_{\mathrm{SUPG}} \|\mathbf{u}^h\|^2. \tag{11}$$

For more ways of calculating τ_{SUPG}, τ_{PSPG} and ν_{LSIC} (see [38,39,34]).

3.2. Semi-discrete formulation and time-integration of structural mechanics

We see no compelling reason to use a space–time formulation for the structural mechanics equations. With $\mathbf{y}^h$ and $\mathbf{w}^h$ coming from appropriately defined trial and test function spaces, respectively, the semi-discrete finite element formulation of the structural mechanics equations are written as

$$\int_{\Omega_0^{\mathrm{s}}} \mathbf{w}^h \cdot \rho^{\mathrm{s}} \frac{\mathrm{d}^2 \mathbf{y}^h}{\mathrm{d}t^2}\, \mathrm{d}\Omega^{\mathrm{s}} + \int_{\Omega_0^{\mathrm{s}}} \mathbf{w}^h \cdot \eta \rho^{\mathrm{s}} \frac{\mathrm{d}\mathbf{y}^h}{\mathrm{d}t}\, \mathrm{d}\Omega^{\mathrm{s}} + \int_{\Omega_0^{\mathrm{s}}} \delta \mathbf{E}^h : \mathbf{S}^h\, \mathrm{d}\Omega^{\mathrm{s}} = \int_{\Omega_t^{\mathrm{s}}} \mathbf{w}^h \cdot (\mathbf{t}^h + \rho^{\mathrm{s}} \mathbf{f}^{\mathrm{s}})\, \mathrm{d}\Omega^{\mathrm{s}}. \tag{12}$$

The fluid mechanics forces acting on the structure are represented by vector $\mathbf{t}^h$. This force term is geometrically nonlinear and thus increases the overall nonlinearity of the formulation. The left-hand-side terms of Eq. (12) are referred to in the original configuration and the right-hand-side terms in the deformed configuration at time t. From this formulation at each time step we obtain a nonlinear system of equations. In solving that nonlinear system with an iterative method, we use the following incremental form:

$$\left[\frac{\mathbf{M}}{\beta \Delta t^2} + \frac{(1-\alpha)\gamma \mathbf{C}}{\beta \Delta t} + (1-\alpha)\mathbf{K}\right] \Delta \mathbf{d}^i = \mathbf{R}^i, \tag{13}$$

where $\mathbf{C} = \eta \mathbf{M} + \zeta \mathbf{K}$. Here $\mathbf{M}$ is the mass matrix, $\mathbf{K}$ is the consistent tangent stiffness matrix associated with the internal elastic forces, $\mathbf{C}$ is a damping matrix, $\mathbf{R}^i$ is the residual vector at the ith iteration, and $\Delta \mathbf{d}^i$ is the ith increment in the nodal displacements vector $\mathbf{d}$. The damping matrix $\mathbf{C}$ is used only in stand-alone structural mechanics computations with specified fluid mechanics forces while establishing a starting shape for the FSI computations. In Eq. (13), all of the terms known from the previous iteration are lumped into the residual vector $\mathbf{R}^i$. The parameters α, β, γ are part of the Hilber–Hughes–Taylor [40] scheme, which is used here for time-integration.

4. Solution of fully discretized equations

Full discretizations of the formulations described in Sections 3.1 and 3.2 lead to coupled, nonlinear equation systems that need to be solved at every time step. In a form that is partitioned with respect to the models represented, these nonlinear equations can be written as follows:

$$\mathbf{N}_1(\mathbf{d}_1, \mathbf{d}_2) = \mathbf{F}_1, \tag{14}$$

$$\mathbf{N}_2(\mathbf{d}_1, \mathbf{d}_2) = \mathbf{F}_2, \tag{15}$$

where $\mathbf{d}_1$ and $\mathbf{d}_2$ are the vectors of nodal unknowns corresponding to unknown functions $\mathbf{u}_1$ and $\mathbf{u}_2$, respectively. In the context of a coupled FSI problem, $\mathbf{u}_1$ and $\mathbf{u}_2$ represent the fluid and structure unknowns, respectively. For the space–time formulation of the fluid mechanics problem, $\mathbf{d}_1$ represents unknowns associated with the finite element formulation written for the space–time slab between the time levels n to $n+1$ (see [13–15]). Solution of these equations with the Newton–Raphson method would necessitate at every Newton–Raphson step solution of the following linear equation system:

$$\mathbf{A}_{11}\mathbf{x}_1 + \mathbf{A}_{12}\mathbf{x}_2 = \mathbf{b}_1, \tag{16}$$

$$\mathbf{A}_{21}\mathbf{x}_1 + \mathbf{A}_{22}\mathbf{x}_2 = \mathbf{b}_2, \tag{17}$$

where $\mathbf{b}_1$ and $\mathbf{b}_2$ are the residuals of the nonlinear equation system, $\mathbf{x}_1$ and $\mathbf{x}_2$ are the correction increments for $\mathbf{d}_1$ and $\mathbf{d}_2$, and $\mathbf{A}_{\beta\gamma} = \partial \mathbf{N}_\beta / \partial \mathbf{d}_\gamma$. Keeping the coupling matrices $\mathbf{A}_{12}$ and $\mathbf{A}_{21}$ in the picture would require taking into account the depen-

dence of Eq. (14) on the mesh motion. In Sections 4.1–4.3 we describe different ways of handling the coupling between Eqs. (14) and (15).

4.1. Block-iterative coupling

In the block-iterative coupling, we do not keep the coupling matrices $\mathbf{A}_{12}$ and $\mathbf{A}_{21}$ in the picture. In an iteration step taking us from iterative solution i to $i+1$, we solve the following set of equations [23,24]:

$$\left.\frac{\partial \mathbf{N}_1}{\partial \mathbf{d}_1}\right|_{(\mathbf{d}_1^i,\mathbf{d}_2^i)} (\Delta \mathbf{d}_1^i) = \mathbf{F}_1 - \mathbf{N}_1(\mathbf{d}_1^i, \mathbf{d}_2^i), \tag{18}$$

$$\left.\frac{\partial \mathbf{N}_2}{\partial \mathbf{d}_2}\right|_{(\mathbf{d}_1^{i+1},\mathbf{d}_2^i)} (\Delta \mathbf{d}_2^i) = \mathbf{F}_2 - \mathbf{N}_2(\mathbf{d}_1^{i+1}, \mathbf{d}_2^i). \tag{19}$$

The linear equation systems given by Eqs. (18) and (19) are also solved iteratively, using the GMRES search technique [41].

Because the matrix blocks representing the coupling between the fluid and structural mechanics systems are not in the picture, in computations where the structure is light, structural response becomes very sensitive to small changes in the fluid mechanics forces and convergence becomes difficult to achieve. In Sections 4.2 and 4.3 we describe ways of keeping the coupling matrix blocks in the picture. In the absence of keeping the coupling matrices $\mathbf{A}_{12}$ and $\mathbf{A}_{21}$, we proposed in [34,42] a shortcut approach for improving the convergence of the block iterations. In this approach, to reduce "over-correcting" (i.e. "over-incrementing") the structural displacements during the block iterations, we increase the mass matrix contribution to $\mathbf{A}_{22}$. This is achieved without altering $\mathbf{b}_1$ or $\mathbf{b}_2$ (i.e. $\mathbf{F}_1 - \mathbf{N}_1(\mathbf{d}_1, \mathbf{d}_2)$ or $\mathbf{F}_2 - \mathbf{N}_2(\mathbf{d}_1, \mathbf{d}_2)$), and therefore when the block iterations converge, they converge to the solution of the problem with the correct structural mass.

4.2. Quasi-direct coupling

In the quasi-direct coupling approach [23,24], we keep the coupling matrices $\mathbf{A}_{12}$ and $\mathbf{A}_{21}$ in the picture partially, without taking into account the dependence of $\mathbf{A}_{12}$ on the mesh motion. In describing this approach, we re-write the finite element formulations given by Eqs. (5) and (12), with a slight change of notation, and with a clarification of how the fluid–structure interface conditions are handled

$$\begin{aligned}
&\int_{Q_n} \mathbf{w}_{1\mathrm{E}}^h \cdot \rho\left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{u}^h - \mathbf{f}^h\right) \mathrm{d}Q + \int_{Q_n} \boldsymbol{\varepsilon}(\mathbf{w}_{1\mathrm{E}}^h) : \boldsymbol{\sigma}(p^h, \mathbf{u}^h)\, \mathrm{d}Q - \int_{(P_n)_{\mathrm{h}}} \mathbf{w}_{1\mathrm{E}}^h \cdot \mathbf{h}_{1\mathrm{E}}^h\, \mathrm{d}P + \int_{Q_n} q_{1\mathrm{E}}^h \nabla \cdot \mathbf{u}^h\, \mathrm{d}Q \\
&\quad + \int_{\Omega_n} (\mathbf{w}_{1\mathrm{E}}^h)_n^+ \cdot \rho((\mathbf{u}^h)_n^+ - (\mathbf{u}^h)_n^-)\, \mathrm{d}\Omega + \sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \frac{1}{\rho}\left[\tau_{\mathrm{SUPG}} \rho\left(\frac{\partial \mathbf{w}_{1\mathrm{E}}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{w}_{1\mathrm{E}}^h\right) + \tau_{\mathrm{PSPG}} \nabla q_{1\mathrm{E}}^h\right] \cdot [\mathrm{Ł}(p^h, \mathbf{u}^h) - \rho \mathbf{f}^h]\, \mathrm{d}Q \\
&\quad + \sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \nu_{\mathrm{LSIC}} \nabla \cdot \mathbf{w}_{1\mathrm{E}}^h \rho \nabla \cdot \mathbf{u}^h\, \mathrm{d}Q = 0,
\end{aligned} \tag{20}$$

$$\int_{Q_n} q_{1\mathrm{I}}^h \nabla \cdot \mathbf{u}^h\, \mathrm{d}Q + \sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \frac{1}{\rho}[\tau_{\mathrm{PSPG}} \nabla q_{1\mathrm{I}}^h] \cdot [\mathrm{Ł}(p^h, \mathbf{u}^h) - \rho \mathbf{f}^h]\, \mathrm{d}Q = 0, \tag{21}$$

$$\int_{\Gamma_{1\mathrm{I}}} (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^- \cdot ((\mathbf{u}_{1\mathrm{I}}^h)_{n+1}^- - \mathbf{u}_{2\mathrm{I}}^h)\, \mathrm{d}\Gamma = 0, \tag{22}$$

$$\begin{aligned}
&\int_{Q_n} (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^- \cdot \rho\left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{u}^h - \mathbf{f}^h\right) \mathrm{d}Q + \int_{Q_n} \boldsymbol{\varepsilon}((\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^-) : \boldsymbol{\sigma}(p^h, \mathbf{u}^h)\, \mathrm{d}Q - \int_{(P_n)_{\mathrm{h}}} (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^- \cdot \mathbf{h}_{1\mathrm{I}}^h\, \mathrm{d}P \\
&\quad + \sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \frac{1}{\rho}\left[\tau_{\mathrm{SUPG}} \rho\left(\frac{\partial (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^-}{\partial t} + \mathbf{u}^h \cdot \nabla (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^-\right)\right] \cdot [\mathrm{Ł}(p^h, \mathbf{u}^h) - \rho \mathbf{f}^h]\, \mathrm{d}Q \\
&\quad + \sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \nu_{\mathrm{LSIC}} \nabla \cdot (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^- \rho \nabla \cdot \mathbf{u}^h\, \mathrm{d}Q = 0,
\end{aligned} \tag{23}$$

$$\begin{aligned}
&\int_{(\Omega_2)_0} \mathbf{w}_2^h \cdot \rho_2 \frac{\mathrm{d}^2 \mathbf{y}^h}{\mathrm{d}t^2}\, \mathrm{d}\Omega + \int_{(\Omega_2)_0} \mathbf{w}_2^h \cdot \eta \rho_2 \frac{\mathrm{d}\mathbf{y}^h}{\mathrm{d}t}\, \mathrm{d}\Omega + \int_{(\Omega_2)_0} \delta \mathbf{E}^h : \mathbf{S}^h\, \mathrm{d}\Omega \\
&\quad = \int_{\Omega_2} \mathbf{w}_2^h \cdot \rho_2 \mathbf{f}_2^h\, \mathrm{d}\Omega + \int_{\Omega_{2\mathrm{E}}} \mathbf{w}_{2\mathrm{E}}^h \cdot \mathbf{h}_{2\mathrm{E}}^h\, \mathrm{d}\Omega - \int_{\Omega_{2\mathrm{I}}} \mathbf{w}_{2\mathrm{I}}^h \cdot \mathbf{h}_{1\mathrm{I}}^h\, \mathrm{d}\Omega.
\end{aligned} \tag{24}$$

While the subscript "I" refers to the fluid–structure interface, the subscript "E" refers to "elsewhere" in the fluid and structure domains or boundaries. In reconciling the slightly modified notation used here with the notation we used in Eqs. (5) and (12), we note that $\rho_2 = \rho^s$, $\mathbf{f}_2^h = \mathbf{f}^s$, $(\Omega_2)_0 = \Omega_0^s$, $\Omega_2 = \Omega_t^s$, and $\Omega_{2\mathrm{I}}$ and $\Omega_{2\mathrm{E}}$ indicate the partitions of Ω_2 corresponding to the interface and "elsewhere". We also note that $\mathbf{h}_{1I}^h = -\mathbf{t}^h$, and $\mathbf{h}_{2\mathrm{E}}^h$ denotes the prescribed external forces acting on the structure in $\Omega_{2\mathrm{E}}$, which is separate from $\mathbf{f}_2^h$. In this formulation, $(\mathbf{u}_{1\mathrm{I}}^h)_{n+1}^-$ and $\mathbf{h}_{1\mathrm{I}}^h$ (the fluid velocity and stress at the interface) are treated as separate unknowns, and Eqs. (22) and (23) can be seen as equations corresponding to these two unknowns, respectively. The structural displacement rate at the interface, $\mathbf{u}_{2\mathrm{I}}^h$, is derived from $\mathbf{y}^h$.

The formulation above is based on allowing for cases when the fluid and structure meshes at the interface are not identical. If they are identical, the same formulation can still be used, but one can also use its reduced version where Eq. (22) is no longer needed and $\mathbf{h}_{1\mathrm{I}}^h$ is no longer treated as a separate unknown. If the structure is represented by a 3D continuum model instead of a membrane model, the formulation above would still be applicable if the domain integrations over $\Omega_{2\mathrm{E}}$ and $\Omega_{2\mathrm{I}}$ in the last two terms of Eq. (24) are converted to boundary integrations over $\Gamma_{2\mathrm{E}}$ and $\Gamma_{2\mathrm{I}}$. In such cases, $\mathbf{h}_{2\mathrm{E}}^h$ would represent the prescribed forces acting "elsewhere" on the surface of the structure.

In a slightly altered version of the formulation given by Eqs. (20)–(24), we suppress the SUPG and LSIC stabilizations in Eq. (23), and use the following equation:

$$\int_{Q_n} (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^- \cdot \rho\left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{u}^h - \mathbf{f}^h\right) \mathrm{d}Q + \int_{Q_n} \boldsymbol{\varepsilon}((\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^-) : \boldsymbol{\sigma}(p^h, \mathbf{u}^h)\, \mathrm{d}Q - \int_{(P_n)_{\mathrm{h}}} (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^- \cdot \mathbf{h}_{1\mathrm{I}}^h \, \mathrm{d}P = 0. \tag{25}$$

Our more recent experiences indicate that Eq. (25) leads to a more robust solution algorithm. Although this may not be so surprising when Eq. (25) is considered in conjunction with Eq. (22), it is something that we plan to investigate further.

4.3. Direct coupling

The mixed analytical/numerical element-vector-based (AEVB/NEVB) computation technique introduced in [26,34] can be employed to keep the coupling matrices in the picture fully by taking into account their dependence on the mesh motion. In describing the mixed AEVB/NEVB technique, we first write the iterative solution of the equation system given by Eqs. (16) and (17) as follows:

$$\mathbf{P}_{11}\Delta\mathbf{y}_1 + \mathbf{P}_{12}\Delta\mathbf{y}_2 = \mathbf{b}_1 - (\mathbf{A}_{11}\mathbf{x}_1 + \mathbf{A}_{12}\mathbf{x}_2), \tag{26}$$

$$\mathbf{P}_{21}\Delta\mathbf{y}_1 + \mathbf{P}_{22}\Delta\mathbf{y}_2 = \mathbf{b}_2 - (\mathbf{A}_{21}\mathbf{x}_1 + \mathbf{A}_{22}\mathbf{x}_2), \tag{27}$$

where $\Delta\mathbf{y}_1$ and $\Delta\mathbf{y}_2$ represent the candidate corrections to $\mathbf{x}_1$ and $\mathbf{x}_2$, and $\mathbf{P}_{\beta\gamma}$'s represent the blocks of the preconditioning matrix $\mathbf{P}$. Here we focus our attention on computation of the residual vectors on the right-hand side, and explore ways for evaluating the matrix–vector products.

Let us suppose that we are able to compute, without major difficulty, the element-level matrices $\mathbf{A}_{11}^e$ and $\mathbf{A}_{22}^e$ associated with the global matrices $\mathbf{A}_{11}$ and $\mathbf{A}_{22}$, and that we prefer to evaluate $\mathbf{A}_{11}\mathbf{x}_1$ and $\mathbf{A}_{22}\mathbf{x}_2$ by using those element-level matrices. Let us also suppose that calculation of $\mathbf{A}_{12}^e$ and $\mathbf{A}_{21}^e$ is exceedingly difficult. Then the computations can be carried out by using a mixed element-matrix-based (EMB)/element-vector-based (EVB) technique [26,34]

$$(\mathbf{A}_{11}\mathbf{x}_1 + \mathbf{A}_{12}\mathbf{x}_2) = \mathop{\mathsf{A}}_{e=1}^{n_{el}} (\mathbf{A}_{11}^e \mathbf{x}_1) + \mathop{\mathsf{A}}_{e=1}^{n_{el}} \lim_{\epsilon_1 \to 0} \left[\frac{\mathbf{N}_1^e(\mathbf{d}_1, \mathbf{d}_2 + \epsilon_1 \mathbf{x}_2) - \mathbf{N}_1^e(\mathbf{d}_1, \mathbf{d}_2)}{\epsilon_1}\right], \tag{28}$$

$$(\mathbf{A}_{21}\mathbf{x}_1 + \mathbf{A}_{22}\mathbf{x}_2) = \mathop{\mathsf{A}}_{e=1}^{n_{el}} (\mathbf{A}_{22}^e \mathbf{x}_2) + \mathop{\mathsf{A}}_{e=1}^{n_{el}} \lim_{\epsilon_2 \to 0} \left[\frac{\mathbf{N}_2^e(\mathbf{d}_1 + \epsilon_2 \mathbf{x}_1, \mathbf{d}_2) - \mathbf{N}_2^e(\mathbf{d}_1, \mathbf{d}_2)}{\epsilon_2}\right], \tag{29}$$

where ϵ_1 and ϵ_2 are small parameters used in numerical evaluation of the directional derivatives. Here, $\mathbf{A}_{11}\mathbf{x}_1$ and $\mathbf{A}_{22}\mathbf{x}_2$ are evaluated with an EMB technique and $\mathbf{A}_{12}\mathbf{x}_2$ and $\mathbf{A}_{21}\mathbf{x}_1$ with an EVB technique.

In extending the mixed EMB/EVB technique to a more general framework, evaluation of a matrix–vector product $\mathbf{A}_{\beta\gamma}\mathbf{x}_\gamma$ (for $\beta\gamma = 1,2,\ldots,N$ and no sum) appearing in a residual vector can be formulated as choice between the following EMB and EVB techniques:

$$\mathbf{A}_{\beta\gamma}\mathbf{x}_\gamma = \mathop{\mathsf{A}}_{e=1}^{n_{el}} (\mathbf{A}_{\beta\gamma}^e \mathbf{x}_\gamma), \tag{30}$$

$$\mathbf{A}_{\beta\gamma}\mathbf{x}_\gamma = \mathop{\mathsf{A}}_{e=1}^{n_{el}} \lim_{\epsilon_\beta \to 0} \left[\frac{\mathbf{N}_\beta^e(\ldots, \mathbf{d}_\gamma + \epsilon_\beta \mathbf{x}_\gamma, \ldots) - \mathbf{N}_\beta^e(\ldots, \mathbf{d}_\gamma, \ldots)}{\epsilon_\beta}\right]. \tag{31}$$

Sometimes computation of $\mathbf{A}_{\beta\gamma}^e$ might not be exceedingly difficult, but we might still prefer to evaluate $\mathbf{A}_{\beta\gamma}\mathbf{x}_\gamma$ with an EVB technique, typically for the purpose of reducing the memory requirements. In such cases, instead of an EVB technique requiring numerical evaluation of directional derivatives, we might want to use the EVB technique described below.

Let us suppose that the nonlinear vector function $\mathbf{N}_\beta$ corresponds to a finite element integral form $\mathbf{B}_\beta(\mathbf{W}_\beta,\mathbf{u}_1,\ldots,\mathbf{u}_N)$. Here $\mathbf{W}_\beta$ represents the vector of nodal values associated with the weighting function $\mathbf{w}_\beta$, which generates the nonlinear equation block β. Let us also suppose that we are able to, without major difficulty, derive the first-order terms in the expansion of $\mathbf{B}_\beta(\mathbf{W}_\beta,\mathbf{u}_1,\ldots,\mathbf{u}_N)$ in $\mathbf{u}_\gamma$. Let the finite element integral form $\mathbf{G}_{\beta\gamma}(\mathbf{W}_\beta,\mathbf{u}_1,\ldots,\mathbf{u}_N,\Delta\mathbf{u}_\gamma)$ represent those first-order terms in $\Delta\mathbf{u}_\gamma$. We note that this integral form will generate $\frac{\partial \mathbf{N}_\beta}{\partial \mathbf{d}_\gamma}$. Consequently, the product $\mathbf{A}_{\beta\gamma}\mathbf{x}_\gamma$ can be evaluated as [26,34]

$$\mathbf{A}_{\beta\gamma}\mathbf{x}_\gamma = \frac{\partial \mathbf{N}_\beta}{\partial \mathbf{d}_\gamma}\mathbf{x}_\gamma = \mathop{\mathsf{A}}_{e=1}^{n_{el}} \mathbf{G}_{\beta\gamma}(\mathbf{W}_\beta,\mathbf{u}_1,\ldots,\mathbf{u}_N,\mathbf{v}_\gamma), \tag{32}$$

where $\mathbf{v}_\gamma$ is a function interpolated from $\mathbf{x}_\gamma$ in the same way $\mathbf{u}_\gamma$ is interpolated from $\mathbf{d}_\gamma$. This EVB technique allows us to evaluate matrix–vector products without dealing with numerical evaluation of directional derivatives. To differentiate between the EVB techniques defined by Eqs. (31) and (32), we call them, respectively, numerical EVB (NEVB) and analytical EVB (AEVB) techniques.

We proposed in [23,24] two ways of using the mixed AEVB/NEVB computation technique to take into account the dependence of the coupling matrices on the mesh motion. In the first way, we propose to use the NEVB technique to compute $\mathbf{A}_{12}\mathbf{x}_2$ while including the dependence of $\mathbf{A}_{12}$ on the mesh motion. This would be done as seen in Eq. (28). In the second way we propose, we limit the use of the NEVB technique to evaluation of the matrix–vector products involving coupling matrices representing the dependence on the mesh motion. This would be done by considering a three-block version of the nonlinear equation system given by Eqs. (14) and (15), where $\mathbf{d}_3$ is the vector of nodal unknowns representing the mesh motion, and the third block of equations represents the mesh-moving equations. The three-block version of Eqs. (16) and (17) can be solved iteratively with the three-block version of Eqs. (26) and (27), which is written as follows:

$$\mathbf{P}_{11}\Delta\mathbf{y}_1 + \mathbf{P}_{12}\Delta\mathbf{y}_2 + \mathbf{P}_{13}\Delta\mathbf{y}_3 = \mathbf{b}_1 - (\mathbf{A}_{11}\mathbf{x}_1 + \mathbf{A}_{12}\mathbf{x}_2 + \mathbf{A}_{13}\mathbf{x}_3), \tag{33}$$

$$\mathbf{P}_{21}\Delta\mathbf{y}_1 + \mathbf{P}_{22}\Delta\mathbf{y}_2 + \mathbf{P}_{23}\Delta\mathbf{y}_3 = \mathbf{b}_2 - (\mathbf{A}_{21}\mathbf{x}_1 + \mathbf{A}_{22}\mathbf{x}_2 + \mathbf{A}_{23}\mathbf{x}_3), \tag{34}$$

$$\mathbf{P}_{31}\Delta\mathbf{y}_1 + \mathbf{P}_{32}\Delta\mathbf{y}_2 + \mathbf{P}_{33}\Delta\mathbf{y}_3 = \mathbf{b}_3 - (\mathbf{A}_{31}\mathbf{x}_1 + \mathbf{A}_{32}\mathbf{x}_2 + \mathbf{A}_{33}\mathbf{x}_3). \tag{35}$$

The NEVB technique can be used for computing $\mathbf{A}_{13}\mathbf{x}_3$ as follows [23,24]:

$$\mathbf{A}_{13}\mathbf{x}_3 = \mathop{\mathsf{A}}_{e=1}^{n_{el}} \lim_{\epsilon_1\to 0}\left[\frac{\mathbf{N}_1^e(\mathbf{d}_1,\mathbf{d}_2,\mathbf{d}_3+\epsilon_1\mathbf{x}_3) - \mathbf{N}_1^e(\mathbf{d}_1,\mathbf{d}_2,\mathbf{d}_3)}{\epsilon_1}\right]. \tag{36}$$

4.4. Remarks on relative efficiencies of the three coupling techniques

Depending on which coupling technique we select, the three linear equation systems corresponding to the fluid and structural mechanics and mesh motion are solved together or separately at each nonlinear iteration. If the linear equations are solved iteratively by using EVB computations, then the memory requirements of the block-iterative (BI), quasi-direct (QD) and direct (D) coupling techniques are not significantly different. However, the computer time increases significantly as we move from QD to D. This is because the combined linear system for the fluid and structural mechanics and mesh motion is substantially larger than when the linear system for the mesh motion is treated separately. Another factor is how the fluid and structure meshes at the interface are stored on distributed processors. Depending on the inter-processor communication speed, there could be a significant increase in computer time as we move from BI to QD. If the interface meshes are stored on one of the processors and the related projection operations are carried out on that single processor, then this bottleneck worsens in QD. This is because while the projection operations are carried out only at every nonlinear iteration in BI, they are carried out at every inner GMRES iteration in QD.

5. Numerical examples

5.1. Flow past a "flag"

In this test problem, we compute the FSI involved in the flapping of a "flag". Results from a very similar problem were presented in [43]. We use the quasi-direct coupling approach, as described by Eqs. (20)–(22) and (24) and (25). In calculation of the stabilization parameters, the $\frac{\partial N_a}{\partial t}$ term in Eq. (8) has been dropped. The dimensions of the flag are 1.5 m in the flow direction and 1.0 m in the span-wise direction. The fluid velocity, density and kinematic viscosity are 2 m/s, 1 kg/m^3 and 0.008 m^2/s, respectively. The flag is modeled as a membrane with density 1000 kg/m^3, thickness 0.2 mm, and Young's modulus 40,000 N/m^2. The leading edge of the flag is held fixed and the lateral edges of the flag are constrained to move only in a normal plane. The number of fluid and structural mechanics nodes are approximately 90,000 and 2800. The time-step size, scaled with the inflow velocity and the length of the flag in the flow direction, is 0.0333. The number of nonlinear iterations and the outer and inner GMRES iterations are 5, 1, and 25, respectively. The FSI computations are

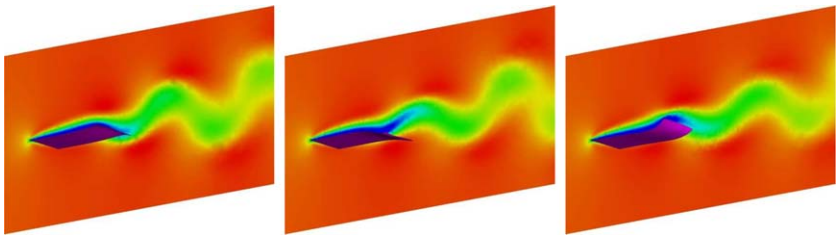

Fig. 1. Time history (left to right) of the flag motion and the horizontal velocity on a normal plane.

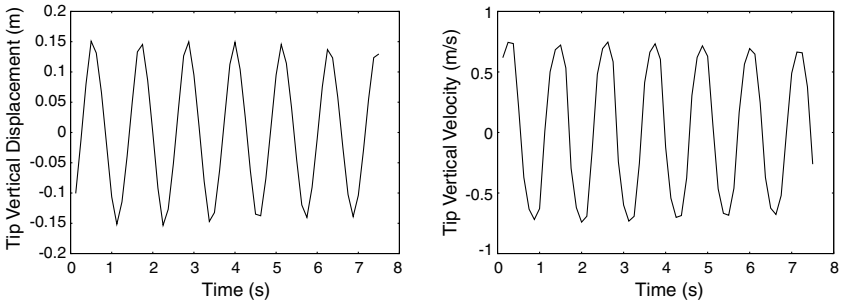

Fig. 2. Vertical displacement (left) and velocity (right) for the midpoint of the free edge of the flag.

carried out until a nearly cyclic pattern of flapping is reached. Fig. 1 shows a sequence of snapshots of the flag and the horizontal velocity on a normal plane. Fig. 2 shows the displacement and velocity for the midpoint of the free edge of the flag. We note that the results are very similar to those reported in [23,24], using the quasi-direct coupling technique described by Eqs. (20)–(24).

5.2. Soft landing of a G-12 parachute

In this test problem, we compute the soft-landing of a G-12 parachute using a direct coupling technique based on Eqs. (20)–(24) and (36). This is a 64-ft diameter parachute with 64 suspension lines, each about 51 ft long, and 4 risers, where each riser is connected to 16 suspension lines. The risers meet at a single confluence point, which is connected to the payload with 4 pneumatic muscle actuators (PMAs), each about 15 ft long. The payload is 2200 lb. The soft-landing is accomplished by the retraction of these PMAs prior to landing. In this test case the PMAs are retracted 7.1 ft in 0.23 s while the

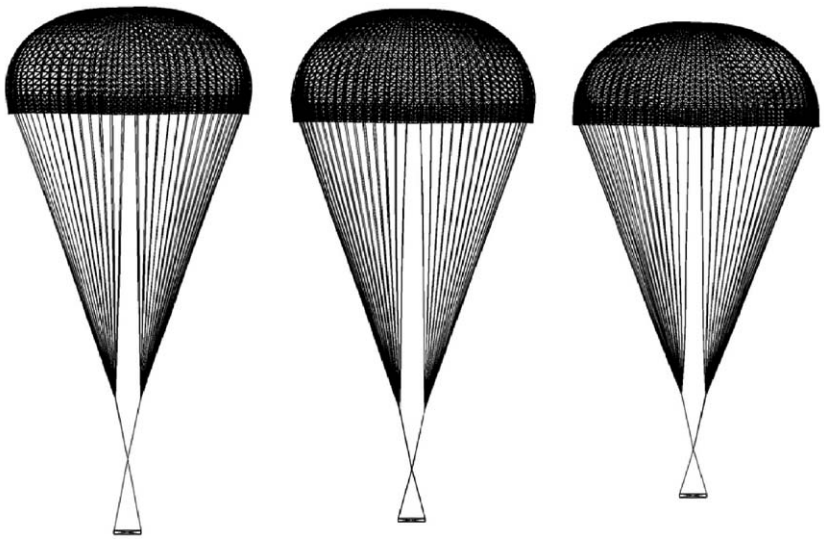

Fig. 3. Soft landing of a G-12 parachute. Dynamics of the parachute, represented by images at three instants during and after the retraction. For more on this computation (see [44]).

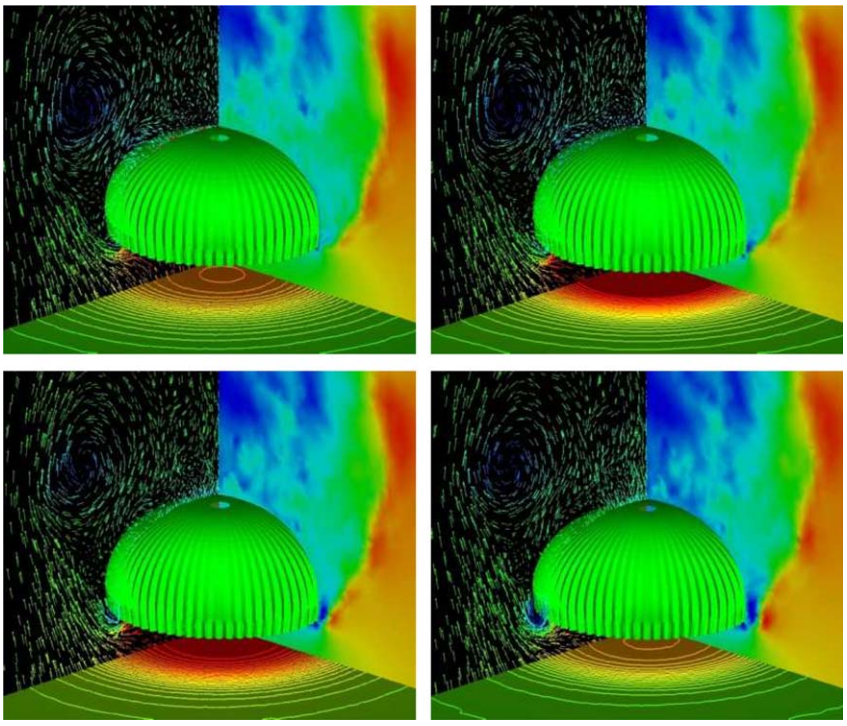

Fig. 4. Soft landing of a G-12 parachute. Flow field at four instants during and after the retraction (left to right and top to bottom). Left plane: velocity vectors; right plane: magnitude of vertical velocity; bottom plane: pressure. For more on this computation (see [44]).

parachute is descending at 28 ft/s. In the computation the $\frac{\partial \mathbf{w}^h}{\partial t}$ term in Eqs. (20) and (23) has been dropped. The number of fluid and structural mechanics nodes are approximately 138,000 and 5,200. The time-step size, scaled with the descent speed and the parachute diameter, is 0.005. The value of ϵ_1 in Eq. (36) is set to 10^{-4}. The number of nonlinear iterations and the outer and inner GMRES iterations are 5, 1, and 30, respectively. Fig. 3 shows the dynamics of the parachute during and after the retraction. The payload descent speed becomes as low as 10 ft/s, and we see a large increase in drag. Fig. 4 shows the flow field during and after the retraction. For more details on this computation (see [44]).

6. Concluding remarks

We provided an overview of the techniques we developed for solving the coupled, nonlinear equations involved in FSI computations with the space–time formulations. These fully discretized equations are generated at every time step from the finite element discretization of the governing equations for the fluid mechanics, structural mechanics and the motion of the fluid mechanics mesh. We focused on the class of problems where the fluid mechanics is governed by the Navier–Stokes equations of incompressible flows, and the structure is governed by the equations of motion for membranes and cables. The mesh motion is governed by the equations of elasticity. The DSD/SST formulation is used for the full discretization of the Navier–Stokes equations. In the finite element formulation of the mesh-moving equations, the smaller elements, which are typically placed near solid surfaces, are stiffened more than the larger ones.

What technique to use for the solution of the fully discretized, coupled fluid and structural mechanics and mesh-moving equations should, to a certain extent, depend on the nature of the application problem. Keeping that in mind, we have developed block-iterative, quasi-direct, and direct coupling techniques. The block-iterative technique gives us more flexibility in terms of algorithmic modularity and independence of the fluid and structural mechanics solvers. The quasi-direct and direct coupling techniques give us more robust algorithms for FSI computations where the structure is light. Because the block-iterative coupling technique was used widely in computations we reported in earlier publications, in this paper we focused on numerical examples computed with the quasi-direct and direct coupling techniques. We believe that the coupling techniques we described in this article can meet the requirements of a diverse class of applications in FSI computations.

Acknowledgements

This work was supported by the US Army Natick Soldier Center (Contract No. DAAD16-03-C-0051), NSF (Grant No. EIA-0116289), and NASA Johnson Space Center (Grant No. NAG9-1435).

References

[1] R. Ohayon, Reduced symmetric models for modal analysis of internal structural-acoustic and hydroelastic-sloshing systems, Comput. Methods Appl. Mech. Engrg. 190 (2001) 3009–3019.
[2] W. Wall, Fluid–Structure Interaction with Stabilized Finite Elements, Ph.D. thesis, University of Stuttgart, 1999.
[3] A. Sameh, V. Sarin, Hybrid parallel linear solvers, Int. J. Comput. Fluid Dyn. 12 (1999) 213–223.
[4] A. Sameh, V. Sarin, Parallel algorithms for indefinite linear systems, Parallel Comput. 28 (2002) 285–299.
[5] C. Michler, E.H. van Brummelen, S.J. Hulshoff, R. de Borst, The relevance of conservation for stability and accuracy of numerical methods for fluid–structure interaction, Comput. Methods Appl. Mech. Engrg. 192 (2003) 4195–4215.
[6] J.-F. Gerbeau, M. Vidrascu, A quasi-Newton algorithm based on a reduced model for fluid–structure interaction problems in blood flows, Math. Model. Numer. Anal. 37 (2003) 663–680.
[7] B. Hubner, E. Walhorn, D. Dinkler, A monolithic approach to fluid–structure interaction using space–time finite elements, Comput. Methods Appl. Mech. Engrg. 193 (2004) 2087–2104.
[8] M. Heil, An efficient solver for the fully coupled solution of large-displacement fluid–structure interaction problems, Comput. Methods Appl. Mech. Engrg. 193 (2004) 1–23.
[9] W. Dettmer, Finite Element Modeling of Fluid Flow with Moving Free Surfaces and Interfaces Including Fluid–Solid Interaction, Ph.D. thesis, University of Wales, Swansea, 2004.
[10] W. Dettmer, D. Peric, A computational framework for fluid–rigid body interaction: finite element formulation and applications, Comput. Methods Appl. Mech. Engrg. 195 (2006) 1633–1666.
[11] T. Washio, T. Hisada, H. Watanabe, T.E. Tezduyar, A robust preconditioner for fluid–structure interaction problems, Comput. Methods Appl. Mech. Engrg. 194 (2005) 4027–4047.
[12] R. Ohayon, C. Felippa, (Eds.), Advances in computational methods for fluid–structure interaction and coupled problems, Comput. Methods Appl. Mech. Engrg. 190 (2001) 2977–2978.
[13] T.E. Tezduyar, Stabilized finite element formulations for incompressible flow computations, Adv. Appl. Mech. 28 (1992) 1–44.
[14] T.E. Tezduyar, M. Behr, 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. Methods Appl. Mech. Engrg. 94 (1992) 339–351.
[15] T.E. Tezduyar, M. Behr, S. Mittal, J. Liou, 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, Comput. Methods Appl. Mech. Engrg. 94 (1992) 353–371.
[16] T.J.R. Hughes, 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.34, ASME, New York, 1979, pp. 19–35.
[17] A.N. Brooks, 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.
[18] T.E. Tezduyar, T.J.R. Hughes, Finite element formulations for convection dominated flows with particular emphasis on the compressible Euler equations, in: Proceedings of AIAA 21st Aerospace Sciences Meeting, AIAA Paper 83-0125, Reno, Nevada, 1983.
[19] T.J.R. Hughes, T.E. Tezduyar, Finite element methods for first-order hyperbolic systems with particular emphasis on the compressible Euler equations, Comput. Methods Appl. Mech. Engrg. 45 (1984) 217–284.
[20] T.E. Tezduyar, S. Mittal, S.E. Ray, R. Shih, Incompressible flow computations with stabilized bilinear and linear equal-order-interpolation velocity-pressure elements, Comput. Methods Appl. Mech. Engrg. 95 (1992) 221–242.
[21] T.J.R. Hughes, L.P. Franca, 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.
[22] T.J.R. Hughes, G.M. Hulbert, Space–time finite element methods for elastodynamics: formulations and error estimates, Comput. Methods Appl. Mech. Engrg. 66 (1988) 339–363.
[23] T.E. Tezduyar, S. Sathe, R. Keedy, K. Stein, Space-time techniques for finite element computation of flows with moving boundaries and interfaces, in: S. Gallegos, I. Herrera, S. Botello, F. Zarate, G. Ayala (Eds.), Proceedings of the III International Congress on Numerical Methods in Engineering and Applied Science, CD-ROM, 2004.
[24] T.E. Tezduyar, S. Sathe, R. Keedy, K. Stein, Space-time finite element techniques for computation of fluid–structure interactions, Comput. Methods Appl. Mech. Engr., ~~in press, doi:10.1016/j.cma.2004.09.014.~~ **195** (2006) 2002-2027.
[25] T.E. Tezduyar, M. Behr, S. Mittal, A.A. Johnson, Computation of unsteady incompressible flows with the finite element methods—space–time formulations, iterative strategies and massively parallel implementations, in: New Methods in Transient Analysis, PVP-vol. 246/AMD-vol. 143, ASME, New York, 1992, pp. 7–24.
[26] T.E. Tezduyar, Finite element methods for flow problems with moving boundaries and interfaces, Arch. Comput. Methods Engrg. 8 (2001) 83–130.
[27] A. Masud, T.J.R. Hughes, A space–time Galerkin/least-squares finite element formulation of the Navier–Stokes equations for moving domain problems, Comput. Methods Appl. Mech. Engrg. 146 (1997) 91–126.
[28] T. Tezduyar, Finite element interface-tracking and interface-capturing techniques for flows with moving boundaries and interfaces, in: Proceedings of the ASME Symposium on Fluid-Physics and Heat Transfer for Macro- and Micro-Scale Gas–Liquid and Phase-Change Flows (CD-ROM), ASME Paper IMECE2001/HTD-24206, ASME, New York, 2001.
[29] T.E. Tezduyar, Stabilized finite element formulations and interface-tracking and interface-capturing techniques for incompressible flows, in: M.M. Hafez (Ed.), Numerical Simulations of Incompressible Flows, World Scientific, New Jersey, 2003, pp. 221–239.
[30] K. Stein, R. Benney, V. Kalro, T.E. Tezduyar, J. Leonard, M. Accorsi, Parachute fluid–structure interactions: 3-D computation, Comput. Methods Appl. Mech. Engrg. 190 (2000) 373–386.
[31] K. Stein, R. Benney, T. Tezduyar, J. Potvin, Fluid–structure interactions of a cross parachute: numerical simulation, Comput. Methods Appl. Mech. Engrg. 191 (2001) 673–687.
[32] K.R. Stein, R.J. Benney, T.E. Tezduyar, J.W. Leonard, M.L. Accorsi, Fluid–structure interactions of a round parachute: modeling and simulation techniques, J. Aircraft 38 (2001) 800–808.

[33] K. Stein, T. Tezduyar, V. Kumar, S. Sathe, R. Benney, E. Thornburg, C. Kyle, T. Nonoshita, Aerodynamic interactions between parachute canopies, J. Appl. Mech. 70 (2003) 50–57.
[34] T.E. Tezduyar, Finite element methods for fluid dynamics with moving boundaries and interfaces, in: E. Stein, R. De Borst, T.J.R. Hughes (Eds.), Encyclopedia of Computational Mechanics, Fluids, vol. 3, John Wiley & Sons, 2004, Chapter 17.
[35] A.E. Green, J.E. Adkins, Large Elastic Deformations, Oxford Clarendon Press, Amen House, London, U.K., 1960.
[36] A.E. Green, W. Zerna, Theoretical Elasticity, Oxford Clarendon Press, Ely House, London, UK, 1968.
[37] M.L. Accorsi, J.W. Leonard, R. Benney, K. Stein, Structural modeling of parachute dynamics, AIAA J. 38 (2000) 139–146.
[38] T.E. Tezduyar, Y. Osawa, Finite element stabilization parameters computed from element matrices and vectors, Comput. Methods Appl. Mech. Engrg. 190 (2000) 411–430.
[39] T.E. Tezduyar, Computation of moving boundaries and interfaces and stabilization parameters, Int. J. Numer. Methods Fluids 43 (2003) 555–575.
[40] H.M. Hilber, T.J.R. Hughes, R.L. Taylor, Improved numerical dissipation for time integration algorithms in structural dynamics, Earthquake Engrg. Struct. Dyn. 5 (1977) 283–292.
[41] Y. Saad, M. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Statist. Comput. 7 (1986) 856–869.
[42] T.E. Tezduyar, Stabilized finite element methods for computation of flows with moving boundaries and interfaces, in: Lecture Notes on Finite Element Simulation of Flow Problems (Basic—Advanced Course), Japan Society of Computational Engineering and Sciences, Tokyo, Japan, 2003.
[43] T. Hisada, H. Watanabe, S. Sugiura. Fluid–structure interaction analysis of human heart by ALE finite element method, 2003, Presentation at the Seventh US National Congress on Computational Mechanics, Albuquerque, New Mexico.
[44] S. Sathe, Enhanced-discretization and solution techniques in flow simulations and parachute fluid–structure interactions, Ph.D. thesis, Rice University, 2004.