

# Methodology for Comparing Coupling Algorithms for Fluid-Structure Interaction Problems

Jason P. Sheldon\*, Scott T. Miller, Jonathan S. Pitt

Applied Research Laboratory, The Pennsylvania State University, State College, USA  
Email: [\\*Jason.P.Sheldon@psu.edu](mailto:Jason.P.Sheldon@psu.edu), [scott.miller@psu.edu](mailto:scott.miller@psu.edu), [jonathan.pitt@psu.edu](mailto:jonathan.pitt@psu.edu)

Received November 23, 2013; revised December 18, 2013; accepted January 17, 2014

Copyright © 2014 Jason P. Sheldon *et al.* This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. In accordance of the Creative Commons Attribution License all Copyrights © 2014 are reserved for SCIRP and the owner of the intellectual property Jason P. Sheldon *et al.* All Copyright © 2014 are guarded by law and by SCIRP as a guardian.

## ABSTRACT

The multi-physics simulation of coupled fluid-structure interaction problems, with disjoint fluid and solid domains, requires one to choose a method for enforcing the fluid-structure coupling at the interface between solid and fluid. While it is common knowledge that the choice of coupling technique can be very problem dependent, there exists no satisfactory coupling comparison methodology that allows for conclusions to be drawn with respect to the comparison of computational cost and solution accuracy for a given scenario. In this work, we develop a computational framework where all aspects of the computation can be held constant, save for the method in which the coupled nature of the fluid-structure equations is enforced. To enable a fair comparison of coupling methods, all simulations presented in this work are implemented within a single numerical framework within the deal.ii [1] finite element library. We have chosen the two-dimensional benchmark test problem of Turek and Hron [2] as an example to examine the relative accuracy of the coupling methods studied; however, the comparison technique is equally applicable to more complex problems. We show that for the specific case considered herein the monolithic approach outperforms partitioned and quasi-direct methods; however, this result is problem dependent and we discuss computational and modeling aspects which may affect other comparison studies.

## KEYWORDS

Fluid-Structure Interaction; FSI; Finite Element Method; Monolithic Coupling; Partitioned Coupling; Dirichlet-Neumann Coupling; Multi-Physics

## 1. Introduction

In this work, we focus on constructing a numerical framework in which algorithms for solving the class of engineering problems commonly referred to as Fluid-Structure Interaction (FSI) problems can be benchmarked and compared in a consistent and quantitative manner. The primary challenge in finding a numerical solution to a FSI problem is not in solving the field equations over either the solid or fluid components of the domain, but rather handling the compatibility conditions that must be enforced on the interface between these components.<sup>1</sup> Recently, two primary strategies for enforcing this coupl-

ing in the Arbitrary Lagrangian-Eulerian (ALE) framework have emerged: solving the solid and fluid components of the domain separately, while iteratively passing information between them, and a simultaneous solution of all field equations at once. The former method is referred to in this work as “partitioned”, and the latter as “monolithic”.

The choice of coupling strategy is an important decision to make from the start of any endeavor to implement a numerical ALE framework for the simulation of fluid-structure interaction, as it places inherent constraints on the entire effort moving forward. On one hand, an author must be sure to choose an approach that allows for all of the relevant physics to be captured, but must also consider the performance of the resulting computer code, the ease of implementation, and its ability to address problems of interest. If one turns to the work of other authors

\*Corresponding author.

<sup>1</sup>Mesh motion [3] is another challenging research area for ALE-based methods. For this work, we assume that a mesh motion strategy is chosen and fixed for our coupling types, and as such studies of optimal mesh motion strategies are outside of the current scope.

for guidance, conflicting statements can be found regarding the performance of various coupling approaches. In addition to this, there is seemingly conventional wisdom in the community today that partitioned approaches are faster than their monolithic counterparts due to the optimized nature of the individual component solvers and smaller systems inverted by the linear solvers [4-7]; however, these notions have been challenged in [8]. There is agreement within the FSI community that, in general, the choice of coupling type is problem dependent, although no specific guidance exists on exactly when a coupling strategy should be preferred.

Michler *et al.* [9] state that in a simple one dimensional FSI problem “the computational cost of the monolithic procedure is three to four times the one of the partitioned procedure;” however, he does not hold each approach to the same convergence tolerance. In fact, he requires more strict convergence of the monolithic solver, which contributes to its longer run time. In contrast, he later hypothesizes that for more complex problems the monolithic approach would be superior in a comparison of computational cost for a desired accuracy against his partitioned method.

Degroote *et al.* [6] utilize ADINA [10], a commercial finite element software package, to compare partitioned and monolithic FSI. They claim that ADINA converges both methods to the same tolerances, for the norms of displacement and force on the interface; however, it is unclear exactly what convergence criterion is used as no equations or methods are presented. They find that for certain cases their partitioned method requires half the computational time of their monolithic method, while for other cases it requires as much as four times the computational time of the monolithic method, and in some cases, the partitioned method does not converge at all.

On the other hand, Heil *et al.* [8] use the open source finite element library OOMP-LIB [11] to compare partitioned and monolithic methods. In their tests, they found that, in terms of computation time required, monolithic methods outperform partitioned methods when FSI effects were significant and that this behavior was amplified with proper pre-conditioners. Again, it is unclear exactly what parameters they use to signify convergence. They claim that the convergence criterion is either the maximum global residual, or the maximum change in position of nodes between iterations; however, in all of their cases they do not require the partitioned method to converge to the same tolerances as the monolithic method.

As can be seen by these examples, although there has been past work in comparing coupling techniques, there is no clear agreement on which method is optimal for a given problem. To aid in providing clarity to this situation, we have developed an ALE-based framework that

can hold constant many of the confounding factors and can focus solely on the fluid-structure coupling methodology. This includes performing all calculations on the same mesh, using the same discretization scheme for all approaches, and using the same benchmark problem for all approaches. Most importantly, we use the same convergence criterion and linear solvers for all coupling methods, ensuring that we are eliminating as many confounding factors as possible. We believe that this type of framework is necessary in order to provide a fair comparison between FSI coupling types, and has clearly been lacking in prior work.

We note that we are interested in algorithms appropriate for fully-coupled FSI problems, *i.e.*, where the structural and fluid responses are tightly intertwined and require a truly simultaneous solution. Problems requiring only one-way or weak coupling are outside the scope of this study. We begin this presentation by deriving the governing equations necessary for FSI in Sections 2 - 3. The discretization procedure is presented in Section 4, the three coupling strategies we are comparing are discussed in Section 5, and formal verification of the implemented algorithm is demonstrated in Section 6. Finally, results of the study and our conclusions are discussed in Sections 7 - 8.

## 2. Kinematics

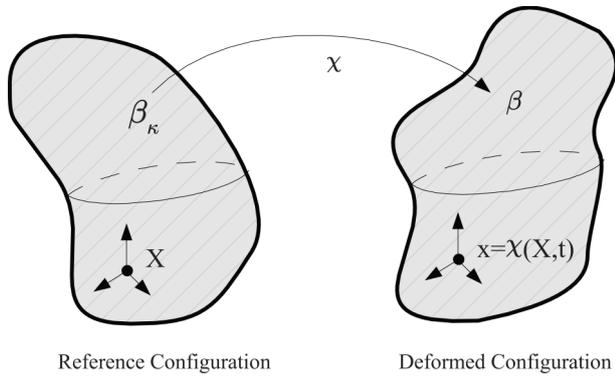
Fluid-structure interactions are modeled as initial boundary value problems (IBVPs). This section presents an overview of the mathematical background necessary to describe these problems. This material is presented from a continuum mechanics perspective, see Gurtin [12], Bowen [13], Chadwick [14], or Spencer [15] for a more detailed discussion.

Kinematics is the geometrical description of motion. It provides descriptions for the movement of bodies (a collection of material points) and changes of reference frames. We denote a body  $\beta \in \mathbb{E}^3$  in the reference configuration  $\kappa^3$  as  $\beta_\kappa$ , where the reference configuration is chosen to be the configuration of the body initially before deformation. The same body  $\beta$  can be defined in subsequent configurations through a deformation function  $\chi$ . The position of a material point in  $\kappa$  is  $X(X, t)$ , while the position of a material point in the deformed configuration, known as the spatial point, is  $x$ . The deformation function  $\chi(X, t)$  maps a material point  $X$  to the spatial point  $x$  at the instant of time  $t$ , as shown in [Figure 1](#).

Considering a smooth sequence of configurations ordered in time, a motion is defined as

$$\chi(X, t) = x(X, t) = X + \mathbf{u}, \quad (1)$$

where  $\mathbf{u}$  is the displacement vector, given by



**Figure 1.** Configurations of a body before and after deformation.

$$\mathbf{u} = \mathbf{x}(\mathbf{X}, t) - \mathbf{X}. \quad (2)$$

The domain  $\Omega$  of an FSI problem, as shown in **Figure 2**, is defined as the union of the disjoint solid and fluid subdomains:

$$\Omega = \Omega_S \cup \Omega_F,$$

where  $\Omega_S$  is the solid subdomain and  $\Omega_F$  is the fluid subdomain. Similarly, the domain's boundary  $\partial\Omega$  is split into four parts: the Dirichlet boundaries for the fluid and solid subdomain  $\Gamma_{D_F}$  and  $\Gamma_{D_S}$  and the Neumann boundaries for the fluid and solid subdomain  $\Gamma_{N_F}$  and  $\Gamma_{N_S}$ , where  $\partial\Omega = \Gamma_{D_S} \cup \Gamma_{D_F} \cup \Gamma_{N_S} \cup \Gamma_{N_F}$ . These boundaries can be subdivided further to suit a given problem. The interface between the solid and fluid subdomains is  $\Gamma_{FS} = \Omega_S \cap \Omega_F$ .

Furthermore, we find it useful to think of this boundary as  $\Gamma_{FS} = \Gamma_{FS_D} \cup \Gamma_{FS_N}$ , denoting the Dirichlet and Neumann parts with the subscripts. The usefulness of this arises when defining proper finite-dimensional function spaces and setting boundary conditions for our discrete problems.

The displacement vector is piecewise defined over the solid and fluid subregions as

$$\mathbf{u} = \begin{cases} \mathbf{u}_s \forall \mathbf{X} \in \Omega_S, \\ \mathbf{u}_m \forall \mathbf{X} \in \Omega_F, \end{cases} \quad (3)$$

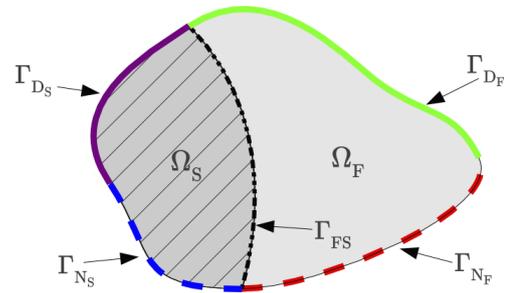
where  $\mathbf{u}_s$  is the solid displacement and  $\mathbf{u}_m$  is the (arbitrary) displacement of the fluid subdomain. Continuity of displacement on the fluid-solid interface requires

$$\mathbf{u}_s - \mathbf{u}_m = \mathbf{0} \quad \forall \mathbf{X} \in \Gamma_{FS}. \quad (4)$$

The deformation gradient is the spatial gradient of the motion, defined as

$$\mathbf{F}(\mathbf{X}, t) = \text{Grad} \chi = \mathbf{I} + \text{Grad} \mathbf{u}, \quad (5)$$

where  $\mathbf{I}$  is the identity tensor. In the next section, we will use  $\mathbf{F}$  for push-forward and pull-back operations in the Arbitrary-Lagrangian-Eulerian (ALE) description



**Figure 2.** Domain  $\Omega$  of an arbitrary two-dimensional fluid-structure interaction problem. The solid and fluid subdomains, Dirichlet and Neumann boundaries, and the fluid-solid interface are labeled.

of the fluid region. Operators with capitalized first letters (e.g. Grad) denote differential operators with respect to  $\mathbf{X}$ , while those with lower-case first letters (e.g. grad) denote differential operators with respect to  $\mathbf{x}$ . The determinant of  $\mathbf{F}$  is

$$J = \det(\mathbf{F}). \quad (6)$$

### 3. Governing Equations

Governing equations are generated by complementing balance laws with constitutive relations for specific materials and kinematic relations. We consider solid materials that are modeled as St. Venant-Kirchhoff hyperelastic isotropic solids. Fluids considered are all incompressible and linearly viscous (Newtonian). We choose to use the ALE description for the fluid to address the disparity on the fluid-solid interface between the reference frames naturally used for the fluid (Eulerian) and solid (Lagrangian). Other approaches, such as space-time techniques [16,17], or immersed boundary/body methods [18,19], could also have been implemented, but we chose to use the ALE description for the purposes of our study. The ALE fluid deformation is described with the equations of linearized elasticity.

#### 3.1. Elasticity

The governing differential equations for an elastic solid are cast in the Lagrangian (referential) frame. The balance of linear momentum is

$$\rho_s \frac{\partial \mathbf{v}_s}{\partial t} - \text{Div} \mathbf{T}_R = \rho_s \mathbf{b}_s \quad \forall \mathbf{X} \in \Omega_S, \quad (7)$$

where  $\mathbf{v}_s$  is the solid's velocity field,  $\rho_s = \rho_s(\mathbf{X})$  is the referential mass density,  $\mathbf{b}_s = \mathbf{b}_s(\mathbf{X}, t)$  is the body force per unit mass, and  $\mathbf{T}_R(\mathbf{X}, t)$  is the first Piola-Kirchhoff stress. The first Piola-Kirchhoff stress is related to the second Piola-Kirchhoff stress  $\mathbf{S}(\mathbf{X}, t)$  via

$$\mathbf{T}_R = \mathbf{F}\mathbf{S}. \quad (8)$$

For an isotropic material,

$$\mathbf{S} = \lambda_s \text{tr}(\mathbf{E})\mathbf{I} + 2\mu_s \mathbf{E}, \quad (9)$$

where  $\lambda$  and  $\mu$  are Lamé's first and second parameters and the strain tensor  $\mathbf{E}(\mathbf{X}, t) = 1/2(\mathbf{F}^T \mathbf{F} - \mathbf{I})$ . Lamé's parameters relate to Poisson's ratio  $\nu$  and Young's modulus  $E$  as  $\nu = \lambda / (2(\lambda + \mu))$  and  $E = \mu(3\lambda + 2\mu) / (\lambda + \mu)$ . The strain tensor expressed as a function of displacement is

$$\mathbf{E} = \frac{1}{2}(\text{Grad} \mathbf{u}_s + (\text{Grad} \mathbf{u}_s)^T + (\text{Grad} \mathbf{u}_s)^T \text{Grad} \mathbf{u}_s). \quad (10)$$

The separate displacement and velocity fields necessitate the use of the kinematic compatibility condition

$$\frac{\partial \mathbf{u}_s}{\partial t} - \mathbf{v}_s = \mathbf{0} \quad \forall \mathbf{X} \in \Omega_s. \quad (11)$$

Strong enforcement of (11) yields the familiar governing differential equation for elastic solids solely in terms of  $\mathbf{u}_s$ :

$$\rho_s \frac{\partial^2 \mathbf{u}_s}{\partial t^2} - \text{Div} \mathbf{T}_R = \rho_s \mathbf{b}_s \quad \forall \mathbf{X} \in \Omega_s. \quad (12)$$

In this work, (7) and (11) are referred to as the two-field  $(\mathbf{u}_s, \mathbf{v}_s)$  formulation of elastodynamics, while (12) is referred to as the one-field  $(\mathbf{u}_s)$  formulation of elastodynamics. We choose to use the two-field formulation in our FSI formulation, as it is very convenient to have the velocity as a primitive variable on the fluid-solid interface (see Section 3.4).

### 3.2. Incompressible Navier-Stokes

The Eulerian frame is the natural setting for the mathematical description of fluid behavior. In the context of FSI descriptions, however, we need to unify the mathematical description of the fluid and solid domains (on the interface, at least). The ALE description of fluid behavior utilizes a smooth map between the referential and spatial domains. We begin by providing the incompressible Navier-Stokes equations in the Eulerian frame, and then transform those equations into the ALE description.

#### 3.2.1. Eulerian Description

The balance of linear momentum in the Eulerian frame is

$$\rho \left( \frac{\partial \mathbf{v}}{\partial t} + \text{div} \mathbf{v} \otimes \mathbf{v} \right) = \text{div} \mathbf{T} + \rho \mathbf{b}_f \quad \forall \mathbf{x} \in \Omega_F(t), \quad (13)$$

where  $\Omega_F(t)$  is the current deformed fluid domain,  $\rho$  is the mass density, and  $\mathbf{T}$  is the Cauchy stress tensor. We denote the velocity field in the fluid domain as  $\mathbf{v} = \mathbf{v}_f$ . Mass balance for an incompressible flow reduces to the divergence free velocity condition

$$\text{div} \mathbf{v} = 0 \quad \forall \mathbf{x} \in \Omega_F(t). \quad (14)$$

The Cauchy stress for a linearly viscous, incompressible fluid, is given as

$$\mathbf{T} = -p\mathbf{I} + \mu(\text{grad} \mathbf{v} + (\text{grad} \mathbf{v})^T), \quad (15)$$

where  $p$  is the fluid pressure and  $\mu$  is the fluid dynamic viscosity. Equations (13), (14), and (15) are collectively known as the Navier-Stokes equations for an incompressible flow.

#### 3.2.2. Arbitrary Lagrangian-Eulerian Description

The ALE formulation is used to provide a unified reference frame for fluid-structure interaction problems. This requires transforming the fluid governing differential Equations (13), (14), and (15), into a form defined on the reference configuration  $\Omega_r(0)$ . The resulting equations are termed the 'incompressible Navier-Stokes equations in ALE form.'

The following relations are necessary to construct the transformation:

$$\text{grad} \mathbf{v} = \text{Grad} \mathbf{v}_\kappa \mathbf{F}^{-1}, \quad (16)$$

and

$$\frac{\partial \mathbf{v}}{\partial t} = \frac{\partial \mathbf{v}_\kappa}{\partial t} - (\text{Grad} \mathbf{v}_\kappa \mathbf{F}^{-1})[\mathbf{v}_m], \quad (17)$$

where  $\mathbf{v}_m$  is the fluid domain velocity, defined as  $\mathbf{v}_m = \partial \chi / \partial t$ . The  $\kappa$  subscript is used to denote quantities transformed from spatial coordinates into the ALE reference frame. Using these relations, the ALE form of the incompressible Navier-Stokes equations<sup>2</sup> are

$$\begin{aligned} & \rho \frac{\partial \mathbf{v}_\kappa}{\partial t} + \rho (\text{Grad} \mathbf{v}_\kappa \mathbf{F}^{-1})[\mathbf{v}_\kappa - \mathbf{v}_m] \\ & + \text{Div}(\mathbf{T}_\kappa \mathbf{F}^{-T}) = \rho \mathbf{b}_\kappa \quad \forall \mathbf{X} \in \Omega_F(0), \end{aligned} \quad (18)$$

$$\mathbf{F}^{-1} : \text{Grad} \mathbf{v}_\kappa^T = 0 \quad \forall \mathbf{X} \in \Omega_F(0), \quad (19)$$

and

$$\mathbf{T}_\kappa = -p_\kappa \mathbf{I} + \mu_\kappa (\text{Grad} \mathbf{v}_\kappa \mathbf{F}^{-1} + \mathbf{F}^{-T} (\text{Grad} \mathbf{v}_\kappa)^T). \quad (20)$$

### 3.3. Mesh Motion

The computational mesh is a non-physical object which can be constructed in any manner we see as beneficial; this is the "arbitrary" component of the ALE formulation. We choose to treat the mesh as a linearly elastic material governed by an elastostatic equation (derived from the balance of linear momentum (7)):

$$\text{Div} \mathbf{T}_R = 0 \quad \forall \mathbf{X} \in \Omega_F(0). \quad (21)$$

For linearly elastic (infinitesimal deformation) materials, which require that  $\|\text{Grad} \mathbf{u}_m\| \ll 1$ , the first Piola-

<sup>2</sup>Hron and Turek [20], Richter and Wick [21], and perhaps others, present a different relation for the balance of mass in an ALE frame than the one we present in (19). It can be shown that (19) is a simplified but identical, version of these through application of equation (2.2.30) from Bowen [13].

Kirchhoff stress, when linearized with respect to  $\text{Grad}\mathbf{u}_m$ , is given as

$$\mathbf{T}_R = \lambda_m \text{tr}(\mathbf{e})\mathbf{I} + 2\mu_m \mathbf{e}, \quad (22)$$

where  $\lambda_m$  and  $\mu_m$  are arbitrarily defined mesh material properties and  $\mathbf{e}$  is the infinitesimal strain tensor. The infinitesimal strain tensor

$$\mathbf{e} = \frac{1}{2} \left( \text{Grad}\mathbf{u}_m + (\text{Grad}\mathbf{u}_m)^\top \right). \quad (23)$$

results from linearizing  $\mathbf{E}$  with respect to  $\text{Grad}\mathbf{u}_m$ .

### 3.4. Boundary and Initial Conditions

In addition to the governing differential equations, there are also boundary conditions, fluid-structure interfacial conditions, and initial conditions necessary to fully define a FSI problem. A boundary condition that prescribes a value for a field variable is a Dirichlet boundary condition, while a boundary condition involving spatial derivatives of a field variable is a Neumann boundary condition. Initial conditions prescribe a value for a field variable over the entire domain (or a subdomain) at the initial time. Specific boundary and initial conditions are problem dependent, but the general form can be defined.

The primitive variables can be prescribed on the Dirichlet boundaries as

$$\mathbf{u}_s|_{\Gamma_{\text{DS}}} = \bar{\mathbf{u}}_s(\mathbf{X}, t), \quad (24)$$

$$\mathbf{v}_s|_{\Gamma_{\text{DS}}} = \dot{\bar{\mathbf{u}}}_s(\mathbf{X}, t), \quad (25)$$

$$\mathbf{v}_\kappa|_{\Gamma_{\text{DF}}} = \bar{\mathbf{v}}_\kappa(\mathbf{X}, t), \quad (26)$$

where time differentiation is denoted with a superposed dot as  $\dot{\theta} = \partial\theta/\partial t$ . The tractions can be prescribed on the Neumann boundaries as

$$\mathbf{T}_s[\mathbf{n}_s]|_{\Gamma_{\text{NS}}} = \bar{\mathbf{t}}_s(\mathbf{x}, t), \quad (27)$$

$$\mathbf{T}_\kappa[\mathbf{n}_\kappa]|_{\Gamma_{\text{NF}}} = \bar{\mathbf{t}}_\kappa(\mathbf{x}, t). \quad (28)$$

For the mesh, a constraint is placed on the entire boundary that there be no normal mesh displacement, to maintain a conforming geometry:

$$(\mathbf{u}_m \cdot \mathbf{n}_f)|_{\Gamma_{\text{DF}} \cup \Gamma_{\text{NF}}} = 0. \quad (29)$$

The initial conditions for the primitive variables can be prescribed over their respective domains at  $t=0$  as

$$\mathbf{u}_s(\mathbf{X}, t=0) = \mathbf{u}_{0s}(\mathbf{X}), \quad (30)$$

$$\mathbf{v}_s(\mathbf{X}, t=0) = \mathbf{v}_{0s}(\mathbf{X}), \quad (31)$$

$$\mathbf{v}_f(\mathbf{X}, t=0) = \mathbf{v}_{0f}(\mathbf{X}). \quad (32)$$

FSI coupling imposes three additional conditions on

the fluid-solid interface. The solid displacement governs the mesh displacement on the interface:

$$(\mathbf{u}_m - \mathbf{u}_s)|_{\Gamma_{\text{FSD}}} = \mathbf{0}, \quad (33)$$

the velocity fields are continuous across the interface:

$$(\mathbf{v}_\kappa - \mathbf{v}_s)|_{\Gamma_{\text{FSD}}} = \mathbf{0}, \quad (34)$$

and the tractions are continuous across the interface:

$$(\mathbf{T}_s[\mathbf{n}_s] + \mathbf{T}_\kappa[\mathbf{n}_\kappa])|_{\Gamma_{\text{FSN}}} = (\mathbf{T}_s - \mathbf{T}_\kappa)[\mathbf{n}_s]|_{\Gamma_{\text{FSN}}} = \mathbf{0}, \quad (35)$$

where the orientation of the interface requires  $\mathbf{n}_s = -\mathbf{n}_\kappa$ .

## 4. Discretization via Finite Elements

The finite element method (FEM) is used to solve an IBVP by transforming its governing partial differential equations (PDEs) into a system of algebraic equations that approximately solve the original problem. The PDEs are weighted by an arbitrary function, and integrated over their respective domains. Integration by parts yields the weak form of the equations. In this work, the notation is used that  $(\mathbf{a}, \mathbf{b})_\Omega \equiv \int_\Omega \mathbf{a} \cdot \mathbf{b} d\Omega$ . The ALE form of the fluid equations requires us to compute the spatial integrals over a reference configuration; as such, the following transformation from the Eulerian to the ALE frame is necessary:

$$\int_{\Omega_{\text{F}(t)}} f(\mathbf{x}) d\mathbf{x} = \int_{\Omega_{\text{F}(0)}} f_\kappa(\mathbf{X}) J d\mathbf{X}. \quad (36)$$

In all future equations  $\Omega_{\text{F}} \equiv \Omega_{\text{F}}(0)$  for brevity. A sufficiently regular computational mesh is used to discretize the domain  $\Omega$ , and we ensure that the fluid-structure interface  $\Gamma_{\text{FS}}$  is coincident with element edges.

The approximate solutions are denoted by the superscript  $^h$ . We define the following standard finite-dimensional trial solution spaces on our mesh:

$$\mathcal{U}_s^h := \left\{ \mathbf{u}_s^h | \mathbf{u}_s^h \in H^1(\Omega_s), \mathbf{u}_s^h|_{\Gamma_{\text{DS}}} = \bar{\mathbf{u}}_s \right\}, \quad (37)$$

$$\mathcal{V}_s^h := \left\{ \mathbf{v}_s^h | \mathbf{v}_s^h \in L^2(\Omega_s), \mathbf{v}_s^h|_{\Gamma_{\text{DS}}} = \dot{\bar{\mathbf{u}}}_s \right\}, \quad (38)$$

$$\mathcal{V}_\kappa^h := \left\{ \mathbf{v}_\kappa^h | \mathbf{v}_\kappa^h \in H^1(\Omega_f), \mathbf{v}_\kappa^h|_{\Gamma_{\text{DF}}} = \bar{\mathbf{v}}_\kappa \right\}, \quad (39)$$

$$\mathcal{P}_\kappa^h := \left\{ p_\kappa^h | p_\kappa^h \in L^2(\Omega_f) \right\}, \quad (40)$$

$$\mathcal{U}_m^h := \left\{ \mathbf{u}_m^h | \mathbf{u}_m^h \in H^1(\Omega_f), (\mathbf{u}_m^h \cdot \mathbf{n}_\kappa)|_{\Gamma_{\text{DF}} \cup \Gamma_{\text{NF}}} = 0 \right\}. \quad (41)$$

Our weighting function spaces are

$$\mathcal{U}_s := \left\{ \hat{\mathbf{u}}_s | \hat{\mathbf{u}}_s \in H^1(\Omega_s), \hat{\mathbf{u}}_s|_{\Gamma_{\text{DS}}} = \mathbf{0} \right\}, \quad (42)$$

$$\mathcal{V}_s := \left\{ \hat{\mathbf{v}}_s | \hat{\mathbf{v}}_s \in L^2(\Omega_s), \hat{\mathbf{v}}_s|_{\Gamma_{\text{DS}}} = \mathbf{0} \right\}, \quad (43)$$

$$\mathcal{V}_\kappa := \left\{ \hat{\mathbf{v}}_\kappa \mid \hat{\mathbf{v}}_\kappa \in H^1(\Omega_F), \hat{\mathbf{v}}_\kappa|_{\Gamma_{Df}} = \mathbf{0} \right\}, \quad (44)$$

$$\mathcal{P}_\kappa := \left\{ \hat{p}_\kappa \mid \hat{p}_\kappa \in L^2(\Omega_F) \right\}, \quad (45)$$

$$\mathcal{Z}_m := \left\{ \hat{\mathbf{u}}_m \mid \hat{\mathbf{u}}_m \in H^1(\Omega_F), \hat{\mathbf{u}}_m|_{\Gamma_{Df} \cup \Gamma_{Nf}} = \mathbf{0} \right\}. \quad (46)$$

With these definitions, it is clear that our finite element method is of the standard (Bubnov-) Galerkin weak formulation, in which weighting function spaces and trial solution spaces are taken to be the same except on the Dirichlet boundaries [22].

The temporal discretization is handled with the implicit, second-order accurate Crank-Nicolson method. The choice of time integration, while important for accuracy and efficiency of a method, is not the focus of our current work. We are interested in investigating different strategies for solving the fully-coupled FSI system. As such, in the remaining mathematical formulation of the FSI problem, we shall omit the time discretization aspect of the problem for the sake of brevity. The weak formulation of the problem is understood to be integrated over a time step with a suitable approximation of the time derivatives.

## Weak Formulation

An FSI problem consists of three distinct sub-problems which are coupled through boundary conditions on the fluid-structure interface  $\Gamma_{FS}$ . We introduce the Galerkin formulation for each sub-problem separately, followed by the statement of the global FSI problem.

### Problem 1 (Solid sub-problem)

Find  $\{\mathbf{u}_s^h, \mathbf{v}_s^h\} \in \mathcal{Z}_s^h \times \mathcal{V}_s^h$  such that

$$\begin{aligned} & \left( \dot{\mathbf{u}}_s, k_{\Omega_s} \hat{\mathbf{u}}_s \right)_{\Omega_s} - \left( \mathbf{v}_s, k_{\Omega_s} \hat{\mathbf{u}}_s \right)_{\Omega_s} + \left( \rho_s \dot{\mathbf{v}}_s, \hat{\mathbf{v}}_s \right)_{\Omega_s} \\ & + \left( \mathbf{T}_R, \text{Grad} \hat{\mathbf{v}}_s \right)_{\Omega_s} - \left( \rho_s \mathbf{b}_s, \hat{\mathbf{v}}_s \right)_{\Omega_s} + \left( (\mathbf{v}_s - \mathbf{v}_\kappa), \hat{\mathbf{v}}_s \right)_{\Gamma_{FSD}} \\ & - \left( \bar{\mathbf{t}}_s, \hat{\mathbf{v}}_s \right)_{\Gamma_{FSN}} - \left( \bar{\mathbf{t}}_s, \hat{\mathbf{v}}_s \right)_{\Gamma_{NS}} = 0 \quad \forall \{\hat{\mathbf{u}}_s, \hat{\mathbf{v}}_s\} \in \mathcal{Z}_s \times \mathcal{V}_s. \end{aligned} \quad (47)$$

where  $k_{\Omega_s}$  is a constant over the solid domain with dimensions of mass per length cubed per time cubed to give our equations dimensional consistency. For the solids considered in this work, the first Piola-Kirchhoff stress  $\mathbf{T}_R$  uses the St. Venant-Kirchhoff strain tensor given in (10).

### Problem 2 (Fluid sub-problem)

Find  $\{\mathbf{v}_\kappa^h, p_\kappa^h\} \in \mathcal{V}_\kappa^h \times \mathcal{P}_\kappa^h$  such that

$$\begin{aligned} & \left( J \rho_\kappa \left( \dot{\mathbf{v}}_\kappa - (\text{Grad} \mathbf{v}_\kappa \mathbf{F}^{-1})[\mathbf{v}_m] \right), \hat{\mathbf{v}}_\kappa \right)_{\Omega_F} \\ & + \left( J \rho_\kappa \left( \text{Grad} \mathbf{v}_\kappa \mathbf{F}^{-1} \right)[\mathbf{v}_\kappa], \hat{\mathbf{v}}_\kappa \right)_{\Omega_F} - \left( \rho \mathbf{b}_\kappa, \hat{\mathbf{v}}_\kappa \right)_{\Omega_F} \\ & + \left( \mathbf{J} \mathbf{T}_\kappa \mathbf{F}^{-T}, \text{Grad} \hat{\mathbf{v}}_\kappa \right)_{\Omega_F} + \left( \mathbf{J} \mathbf{F}^{-1} : \text{Grad} \mathbf{v}_\kappa^T, \hat{p}_\kappa \right)_{\Omega_F} \\ & - \left( \bar{\mathbf{t}}_\kappa, \hat{\mathbf{v}}_\kappa \right)_{\Gamma_{NF}} - \left( \mathbf{t}_\kappa, \hat{\mathbf{v}}_\kappa \right)_{\Gamma_{FSN}} + \left( (\mathbf{v}_s - \mathbf{v}_\kappa), \hat{\mathbf{v}}_\kappa \right)_{\Gamma_{FSD}} \\ & = 0 \quad \forall \{\hat{\mathbf{v}}_\kappa, \hat{p}_\kappa\} \in \mathcal{V}_\kappa \times \mathcal{P}_\kappa \end{aligned} \quad (48)$$

where  $\mathbf{F}$  and  $\mathbf{J}$  are as given in (5) and (6), and, for the linearly viscous, incompressible fluids considered in this work, the Cauchy stress  $\mathbf{T}_\kappa$  is as given in (20).

### Problem 3 (Mesh sub-problem)

Find  $\mathbf{u}_m^h \in \mathcal{Z}_m^h$  such that

$$\begin{aligned} & \left( \lambda_m \text{Div} \mathbf{u}_m, \text{Div} \hat{\mathbf{u}}_m \right)_{\Omega_F} + \left( (\mathbf{u}_m - \mathbf{u}_s), \hat{\mathbf{u}}_m \right)_{\Gamma_{FSD}} \\ & + \left( \mu_m \left( \text{Grad} \mathbf{u}_m + \text{Grad} \mathbf{u}_m^T \right), \text{Grad} \hat{\mathbf{u}}_m \right)_{\Omega_F} \\ & = 0 \quad \forall \hat{\mathbf{u}}_m \in \mathcal{Z}_m. \end{aligned} \quad (49)$$

Solving (49) using spatially constant material properties  $\lambda_m$  and  $\mu_m$  can result in poor quality meshes if the solid displacement is large enough<sup>3</sup>. To prevent this, an artificial stiffness is imposed on the mesh, using  $\lambda_m = \lambda_m(\mathbf{X})$  and  $\mu_m = \mu_m(\mathbf{X})$ , in a manner such that it is stiffer near the interface and less stiff away from the interface. This propagates the distortion caused by the solid displacement throughout the entire mesh, instead of it being localized at the interface.

The global FSI problem is a direct sum of Problems 1, 2, and 3.

### Problem 4 (Global FSI problem)

Find  $\{\mathbf{u}_s^h, \mathbf{v}_s^h, \mathbf{v}_\kappa^h, p_\kappa^h, \mathbf{u}_m^h\} \in \mathcal{Z}_s^h \times \mathcal{V}_s^h \times \mathcal{V}_\kappa^h \times \mathcal{P}_\kappa^h \times \mathcal{Z}_m^h$

such that (47), (48), and (49) are simultaneously satisfied.

The individual sub-problems are coupled solely via the boundary conditions on the FSI interface  $\Gamma_{FS}$ . The globally defined system in Problem 4 is a fully coupled non-linear system of equations. We have employed the Newton-Raphson method to solve these equations, including all non-linear sub-problems that arise. We present some of the details in Appendix 9 since the calculations can be daunting, especially for the monolithic method.

## 5. FSI Algorithms

The numerical solution of Problem 4 is the focus of our study. We examine three different types of solution procedures: partitioned, monolithic, and quasi-direct. These are often referred to as FSI coupling methods, although they can also be thought of as a family of algebraic solution methods to solve the fully coupled Problem 4.

One of the key features of our study is that all three solution algorithms are implemented within the same software package (deal.ii). This feature allows us to use the same finite element assembly routines for all FSI coupling types, ensuring that computational cost differences between the methods studied is not due to different implementations. Furthermore, the convergence criterion chosen are applied in exactly the same manner for each FSI algorithm, as described in Section 7.2.

<sup>3</sup>A poor quality mesh satisfies its boundary conditions but contains cells that are extremely distorted, such as cells with nearly parallel adjacent sides.

The global solution procedure of the three coupling types is the same. After a setup procedure and application of initial conditions, a time loop advances the solution with a prescribed time step size until the final simulation time is reached. For each time step, the three FSI coupling algorithms solve Problem 4 in different manners. We refer to the non-linear solution procedure within a single time step as *FSI iterations*.

### 5.1. Partitioned Coupling

The partitioned coupling method couples Problems 1, 2, and 3 in a sequential fashion to solve the global FSI problem. This coupling method is also referred to as Dirichlet-Neumann coupling, due to the way constraints are imposed on the FSI interface. There are other types of partitioned coupling, but we choose Dirichlet-Neumann coupling as it is by far the most common type used in practice: used by Turek *et al.* [2], Degroote *et al.* [6], Küttler *et al.* [23], and Campbell *et al.* [24], amongst many others. The FSI iterations for the partitioned method involve solving the fluid, solving the solid, and solving the mesh; this process is iterated until convergence is attained. The partitioned algorithm is detailed in **Figure 3(a)**.

We solve (48) with prescribed velocities on  $\Gamma_{\text{FS}_D}$ . As such, we enforce Dirichlet conditions on  $\Gamma_{\text{FS}_D}$  and define  $\Gamma_{\text{FS}_N} = \emptyset$ . The corresponding change to our finite dimensional vector spaces requires us to replace (39) and (44) with

$$\mathcal{V}_\kappa^h := \left\{ \mathbf{v}_\kappa^h \mid \mathbf{v}_\kappa^h \in H^1(\Omega_F), \bar{\mathbf{v}}_\kappa^h \Big|_{\Gamma_{\text{Df}}} = \mathbf{v}_\kappa, \mathbf{v}_\kappa^h \Big|_{\Gamma_{\text{FS}_D}} = \mathbf{v}_s \right\} \quad (50)$$

$$\mathcal{V}_\kappa := \left\{ \hat{\mathbf{v}}_\kappa \mid \hat{\mathbf{v}}_\kappa \in H^1(\Omega_F), \hat{\mathbf{v}}_\kappa \Big|_{\Gamma_{\text{Df}} \cup \Gamma_{\text{FS}_D}} = \mathbf{0} \right\}, \quad (51)$$

respectively. The strong enforcement of the velocity coupling is achieved in our implementation through the use of deal.ii's constraint tools [1]. Newton-Raphson iterations are used to solve the non-linear fluid sub-problem.

Once a fluid solution is obtained, Problem 1 is solved over the solid domain using the fluid traction as the driving force on the FSI interface. That is,  $\Gamma_{\text{FS}_D} = \emptyset$  and the applied traction on  $\Gamma_{\text{FS}_N}$  is computed as

$$\begin{aligned} \mathbf{t}_s &= -\mathbf{t}_\kappa = -\mathbf{T}_\kappa[\mathbf{n}_\kappa] \\ &= \left( p_\kappa \mathbf{I} - \mu_\kappa \left( \text{Grad} \mathbf{v}_\kappa \mathbf{F}^{-1} + \mathbf{F}^{-\text{T}} \text{Grad} \mathbf{v}_\kappa^{\text{T}} \right) \right) \left[ \mathbf{J} \mathbf{F}^{-\text{T}} \mathbf{n}_f \right], \end{aligned} \quad (52)$$

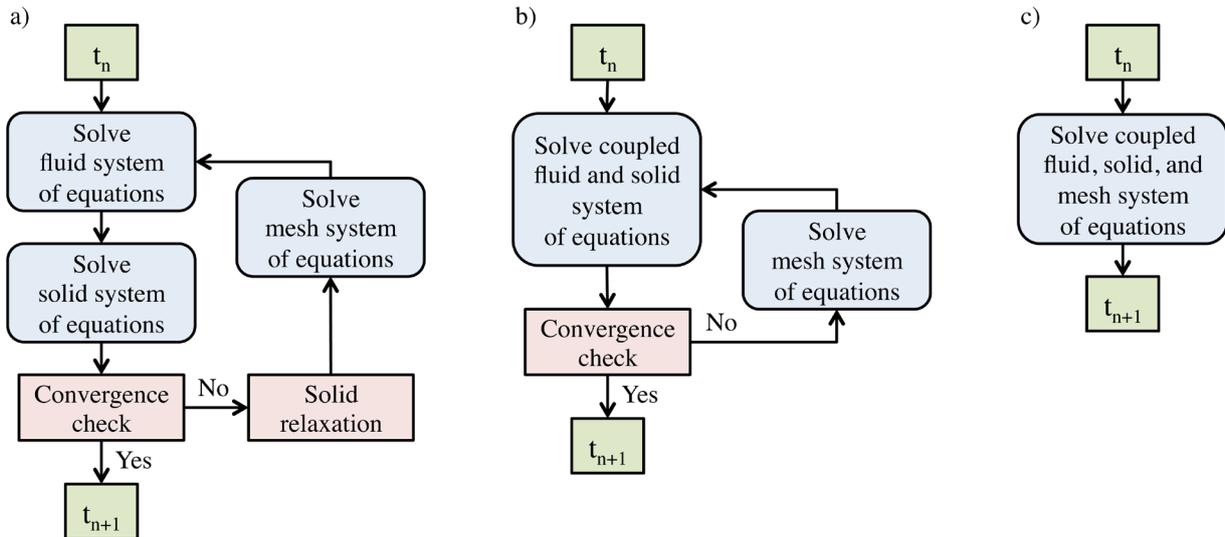
which comes from (35) and (20). The function space definitions on  $\Omega_s$  do not need modification for the partitioned method. Due to the geometric non-linearities in Problem 1, a Newton-Raphson method is used to solve for the solid displacements and velocities.

If the change in position of  $\Gamma_{\text{FS}}$  is non-trivial (see Section 7.2 for discussion of this criterion), we deem it necessary to update the mesh position by solving Problem 3. Before solving for the mesh displacement, it is well known [23] that the interfacial displacements should be under-relaxed to ensure convergence. The solid displacements and velocities are relaxed as

$$\mathbf{u}_{s,i+1}^h = \mathbf{u}_{s,i}^h + \omega_{i+1} \left( \tilde{\mathbf{u}}_{s,i+1}^h - \mathbf{u}_{s,i}^h \right), \quad (53)$$

$$\mathbf{v}_{s,i+1}^h = \mathbf{v}_{s,i}^h + \omega_{i+1} \left( \tilde{\mathbf{v}}_{s,i+1}^h - \mathbf{v}_{s,i}^h \right), \quad (54)$$

where the subscript  $i$  denotes the iteration count and the superposed tilde denotes the non-relaxed solution from solving Problem 1. We find it beneficial (or, at a minimum, non-detrimental) to relax the entire solid solution, rather than just the degrees of freedom whose support is on  $\Gamma_{\text{FS}}$ .



**Figure 3.** Flowcharts demonstrating the partitioned (a); quasi-direct (b); and monolithic (c) FSI algorithms.

The scalar relaxation parameter  $\omega_{i+1}$  can be chosen to be constant, but it has been shown [23] that allowing it to change between iterations can accelerate convergence. Aitken's  $\delta^2$  convergence acceleration method seems to be the most popular algorithm for choosing  $\omega$ , and we have used it in this work. The position of the fluid-solid interface is viewed as a set of fixed points for the partitioned FSI iterations, and the secant method is used to extrapolate the expected position of the interface. The result is the recursive formula for computing the relaxation factor

$$\omega_{i+1} = -\omega_i \frac{\mathbf{r}_i \cdot (\mathbf{r}_{i+1} - \mathbf{r}_i)}{\|\mathbf{r}_{i+1} - \mathbf{r}_i\|^2}, \quad \mathbf{r}_{i+1} = \tilde{\mathbf{d}}_{s,i+1} - \mathbf{d}_{s,i}, \quad (55)$$

and the discrete vector  $\mathbf{d}_s$  consists of the nodal values from  $\mathbf{u}_s^h|_{\Gamma_{FS}}$ . The Euclidean inner product and norm are used for the discrete vector. For the first FSI iteration of each timestep a relaxation factor of 0.75 is used.

Once the solid solution has been relaxed, we update the mesh displacements by solving Problem 3. The function space definitions (41) and (46) are augmented as

$$\mathcal{Z}_m^h := \left\{ \mathbf{u}_m^h \mid \mathbf{u}_m^h \in H^1(\Omega_F), (\mathbf{u}_m^h \cdot \mathbf{n}_\kappa)_{\Gamma_{D_f} \cup \Gamma_{N_f}} = 0, \mathbf{u}_m^h|_{\Gamma_{FS}} = \mathbf{u}_s^h|_{\Gamma_{FS}} \right\}, \quad (56)$$

$$\mathcal{Z}_m := \left\{ \hat{\mathbf{u}}_m \mid \hat{\mathbf{u}}_m \in H^1(\Omega_F), \hat{\mathbf{u}}_m|_{\Gamma_{D_f} \cup \Gamma_{N_f} \cup \Gamma_{FS}} = \mathbf{0} \right\}, \quad (57)$$

respectively, to account for enforcing a Dirichlet condition on the fluid-structure interface. Unlike the fluid and solid sub-problems, the mesh motion is governed by a linear system of equations and only requires a single algebraic solve.

## 5.2. Quasi-Direct Coupling

The second algorithm of our study for quasi-direct coupling, as first performed by Tezduyar [25], is detailed

$$\mathcal{V}^h := \left\{ \mathbf{v}^h \mid \mathbf{v}^h \in H^1(\Omega), \mathbf{v}^h|_{\Gamma_D} = \bar{\mathbf{v}}, \mathbf{v}^h|_{\Omega_S} = \mathbf{v}_s^h, \mathbf{v}^h|_{\Omega_F} = \mathbf{v}_\kappa^h \right\}, \quad (58)$$

$$\mathcal{V} := \left\{ \hat{\mathbf{v}} \mid \hat{\mathbf{v}} \in H^1(\Omega), \hat{\mathbf{v}}|_{\Gamma_D} = \mathbf{0}, \hat{\mathbf{v}}|_{\Omega_S} = \hat{\mathbf{v}}_s, \hat{\mathbf{v}}|_{\Omega_F} = \hat{\mathbf{v}}_\kappa \right\}. \quad (59)$$

The consequence of this choice is that (34) is strongly satisfied, which implies that

$$-(\mathbf{t}_s, \hat{\mathbf{v}}_s)_{\Gamma_{FSN}} - (\mathbf{t}_\kappa, \hat{\mathbf{v}}_\kappa)_{\Gamma_{FSN}} = (\mathbf{t}_R, \hat{\mathbf{v}}_\kappa - \hat{\mathbf{v}}_s)_{\Gamma_{FSN}} = 0, \quad (60)$$

after enforcing (35) and recognizing that the velocity weighting functions are chosen from the same function space. Another way to recognize that the tractions on  $\Gamma_{FS}$  do not contribute to the weak form is that  $\Gamma_{FS_D} = \Gamma_{FS}$  and  $\Gamma_{FS_N} = \emptyset$  by construction of a conforming (continuous) velocity space. Therefore, the traction continuity is implicitly satisfied and does not enter the quasi-direct (or monolithic) equations [26].

The solution from the fluid-solid does not need to be relaxed as it was for the partitioned algorithm. The relaxation procedure is only used for iterative Dirichlet-Neumann coupling schemes for stability. Our only requirements on the mesh motion are the same as for the partitioned case. Namely,  $\mathbf{u}_m^h$  is computed using the function spaces given in (56) and (57).

in **Figure 3(b)**. The algorithm solves Problems 1 and 2 monolithically, and an iterative procedure is used to couple with Problem 3 to fully solve the complete FSI Problem 4. For each time step, we solve the coupled fluid-solid system, update the mesh displacements, and iterate on these two solves until convergence is reached.

The quasi-direct algorithm does not use the Dirichlet-Neumann coupling scheme that the partitioned method uses. Rather, we construct a conforming velocity field over the entire reference domain  $\Omega$  as

## 5.3. Monolithic Coupling

The full monolithically coupled solution procedure is the third algorithm of our study and is detailed in **Figure 3(c)**. Problem 4 is solved as a single non-linear system, with no FSI iterations necessary. We use the conforming velocity space given in (58) and (59) to strongly couple the fluid and solid domains.

The displacement fields  $\mathbf{u}_s^h$  and  $\mathbf{u}_m^h$  can be combined into a single conforming field, but doing so raises some computational challenges. Unlike the fluid-solid domains which are two-way coupled (they affect each other mutually), the solid-mesh displacement fields are one-way coupled. That is, solving Problem 3 for the mesh displacements should not directly influence the displacement field of the solid, while the solution of Problem 1 provides the driving boundary condition for the mesh solution. Thus, constructing a typical conforming finite element space over the entire domain has the detrimental affect of allowing the artificial mesh 'stiffness' to limit the motion of the solid structure.

We have used a simple and efficient method to strongly enforce (33) while avoiding the artificial stiffness phenomena. We solve Problem 4 using function spaces defined by (37), (40), (41), (42), (45), (46), (58), (59). The Newton-Raphson method is used to compute the solution, and we provide details of the linearization in the Appendix. Within each Newton-Raphson iteration, we weakly enforce the displacement continuity constraint within Problem 3. At the end of the iteration, the nodal mesh displacements on the interface are set to be equal to the solid displacement. In our numerical computations, we have found that weakly enforcing continuity yields mesh values that are relatively close to those computed for the solid. Also, modifying the mesh solution at the end of each Newton-Raphson iteration does not seem to affect the convergence characteristics of the method.

## 6. Code Verification

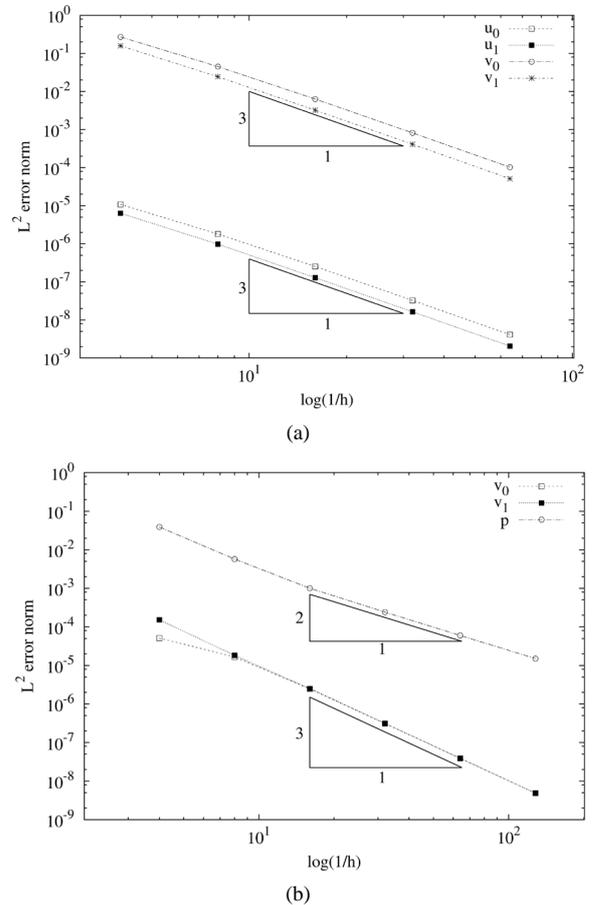
Verification of the discretized equations of motion, and the software in which it is implemented, is performed via the method of manufactured solutions (see, e.g., [27,28]). We perform the following sequence of analysis for the solid and fluid solvers separately: 1) choose an exact solution for the governing equations, 2) substitute this solution in the equations to obtain analytical expressions for the corresponding forcing functions and boundary conditions, 3) implement these into the code, and 4) compute an approximate solution. The error between the exact and approximate solutions is then computed according to some norm, and the process is repeated for a sequence of uniformly refined meshes.

Figure 4 shows the results of this process for several iterations of mesh refinement. The Figure 4(a) shows the individual components of the elastic displacement and velocity error declining as mesh size decreases, with a linear rate of  $p + 1 = 3$  under the  $L^2$  norm when using bi-quadratic  $Q^2$  tensor-product elements for all fields. The Figure 4(b) shows the analogous situation for the velocity and pressure fields in the fluid solver, where the difference in convergence rate is due to the Taylor-Hood  $Q^2 - Q^1$  tensor product elements used for the velocity and pressure, respectively.

Overall, this simple verification exercise demonstrates that the individual components of the FSI solver are functioning correctly. We will evaluate the correctness of the coupling implementation in the next section; more details regarding verification of the solvers can be found in Sheldon [29].

## 7. Model Validation

An FSI benchmark was proposed by Turek and Hron [2] for the purpose of facilitating comparison between different FSI modeling approaches and algorithms, regardless of coupling types or discretization schemes. This

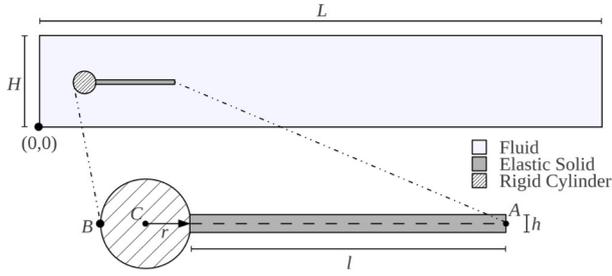


**Figure 4. Error convergence plots for the two-field elasticity formulation (a) and Navier-Stokes formulation (b). In each case, the expected rates of error convergence are achieved, with respect to the  $L^2$ -norm. This demonstrates that each component of the FSI solver is functioning correctly.**

benchmark consists of a numerical experiment involving laminar incompressible channel flow around a bluff body, which subsequently sheds vortical structures that excite a flexible tail until it reaches a state-state flow induced oscillation. Data were provided by Turek and Hron for the tip displacement of the elastic structure, hereafter called the flag (see Figure 5), and the drag and lift about the combined boundary of the flag and the cylinder to which the flag is attached. We used this benchmark as a means to validate the FSI model developed in this work, via direct comparison of our model's predictions to that of Turek and Hron, and as a baseline for comparing the coupling strategies to one another.

### 7.1. Numerical Experiment Definition

The problem domain is depicted in Figure 5, where the channel, rigid cylinder, and elastic flag are noted. Flow is from left to right, with the cylinder slightly offset in the channel. Relevant dimensions are reproduced for the reader in Table 1. We will present our analysis of repro-



**Figure 5.** Turek and Hron [2] FSI benchmark domain with expanded view of cylinder and flag.

**Table 1.** Turek and Hron [2] FSI benchmark domain dimensions and material properties.

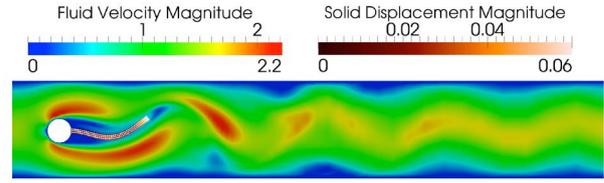
Parameter	Symbol	Value [m]	Property	Value
Channel length	$L$	2.5	$\rho_s \left[ \frac{\text{kg}}{\text{m}^3} \right]$	10,000
Channel height	$H$	0.41	$\nu_s$	0.4
Cylinder center	$C$	(0.2, 0.2)	$E_s \left[ \text{e6} \frac{\text{kg}}{\text{ms}^2} \right]$	5.6
Cylinder radius	$r$	0.05	$\rho_c \left[ \frac{\text{kg}}{\text{m}^3} \right]$	1000
Flag length	$l$	0.35	$\mu_c \left[ \frac{\text{kg}}{\text{ms}} \right]$	1
Flag height	$h$	0.02	$\bar{U} \left[ \frac{\text{m}}{\text{s}} \right]$	1
Reference point	$A$	(0.6, 0.2)		
Reference point	$B$	(0.15, 0.2)		

ducing the “FSI2” case of Turek and Hron, as it involves the largest structural deformations, and is therefore the most intense test for our solver. The material properties for this simulation are also reproduced for the reader in **Table 1**, where  $\rho$  is the density,  $\nu$  is the Poisson’s ratio,  $\mu$  is the viscosity,  $E$  is the Young’s modulus, and  $\bar{U}$  is the mean inflow velocity. As before, the  $s$  and  $f$  subscripts indicate solid and fluid properties, respectively. **Figure 6** shows an example of the flow field and solid deformation, using results from FSI2 after 16.5 simulation seconds, to help in visualizing the benchmark.

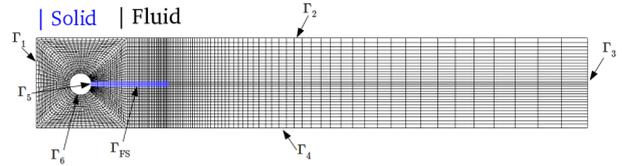
The boundaries of the computational domain are labeled in **Figure 7**, which also shows the mesh we used for our calculations. The boundary conditions are as follows: a parabolic fluid velocity inflow on  $\Gamma_1$ :

$$\mathbf{v}^*(y) \Big|_{\Gamma_1} = \left\{ \begin{array}{l} 1.5\bar{U} \frac{y(H-y)}{(H/2)^2} \\ 0 \end{array} \right\}, \quad (61)$$

a no-slip fluid velocity on  $\Gamma_2$ ,  $\Gamma_4$ , and  $\Gamma_6$ :



**Figure 6.** Visualization of the velocity magnitude and flag deformation at  $t = 16:5$  [s]. Solid displacement and fluid velocity are shown with their own color scales. The flag is shown as a wireframe to distinguish it from the fluid and the rigid cylinder.



**Figure 7.** Turek and Hron [2] FSI benchmark mesh with labeled boundaries and domains.

$$\mathbf{v}_\kappa \Big|_{\Gamma_{2,4,6}} = 0, \quad (62)$$

a constant fluid pressure on  $\Gamma_3$ , which is prescribed to be zero:

$$p_\kappa \Big|_{\Gamma_3} = 0, \quad (63)$$

a fixed solid displacement and velocity on  $\Gamma_5$ :

$$\mathbf{u}_s \Big|_{\Gamma_5} = \mathbf{v}_s \Big|_{\Gamma_5} = 0, \quad (64)$$

the interface conditions given in (33) and (34) on  $\Gamma_{FS}$ , and the mesh displacement condition given in (29) on  $\Gamma_1$ ,  $\Gamma_2$ ,  $\Gamma_3$ ,  $\Gamma_4$ , and  $\Gamma_6$ .

All fields are initially zero. For the first two seconds the fluid inlet velocity is subject to a smooth increase defined by

$$\mathbf{v}_\kappa(y, t) \Big|_{\Gamma_1} = \left\{ \begin{array}{l} \mathbf{v}^* \Big|_{\Gamma_1} \frac{1 - \text{Cos}\left(\frac{\pi}{2}t\right)}{2} \quad t < 2.0 \\ \mathbf{v}^* \Big|_{\Gamma_1} \quad t \geq 2.0 \end{array} \right. \quad (65)$$

A timestep size  $\Delta t = 0.0025$  [s] was used in all cases to satisfy the Courant-Friedrichs-Lewy (CFL) condition [30], computed from a maximum flow velocity and minimum cell diameter of approximately

$v_{\max} = 2.2$  [m/s] and  $\Delta x_{\min} = 0.011$  [m], respectively.

As in Section 6, all subsequent results in the section were computed with  $Q^2$  elements for the solid displacement and velocity, Taylor-Hood  $Q^2 - Q^1$  elements for the fluid velocity/pressure,  $Q^2$  elements for the mesh displacement, and the Crank-Nicholson method for the time discretization. For consistency in comparison of the timing results, all simulations were performed on the same computer. Linear algebraic solves were done

with the sparse direct solver UMFPACK [31].

## 7.2. Convergence Criterion

The FSI coupling strategies that we have employed in this work all require some sort of sub-iteration at each time step in the simulation process. For the partitioned schemes, we make fixed-point iterations between the solver components to converge the solution, while for the monolithic approaches we use Newton-Raphson iterations. Consistency in defining convergence criteria for the global coupled FSI solution, that is applicable to both the partitioned and monolithic approaches, is a key point of this work. If this metric is not chosen consistently, we will not be able to say that the solutions are similarly converged to the same level of error, and therefore be unable to make comparisons regarding the wall clock time.

For the global FSI solution, the convergence criterion [23,32] is given by

$$\frac{\|\mathbf{u}_s^{i+1} - \mathbf{u}_s^i\|_{L^2(\Gamma_{FS})}}{\|\mathbf{1}\|_{L^2(\Gamma_{FS})}} < \varepsilon_{FSI}, \quad (66)$$

where  $\mathbf{u}_s^{i+1}$  and  $\mathbf{u}_s^i$  refer to the solid displacement at the current and previous FSI iteration, whether that iteration is fixed-point (partitioned) or Newton-Raphson (monolithic). Here,  $\|\cdot\|_{L^2(\Gamma_{FS})}$  is the standard  $L^2$  norm over the fluid-solid interface, and  $\varepsilon_{FSI}$  is some convergence tolerance, which we typically defined to be  $1\text{E-}10$ , for all coupling methods. For the partitioned method, we perform this solution check after every instance where the solid displacement is solved, while for the monolithic approaches, we perform this check after each Newton Iteration. In all coupling approaches, once this criterion is satisfied, we exit the sub-iteration loop and move on to the next time step.

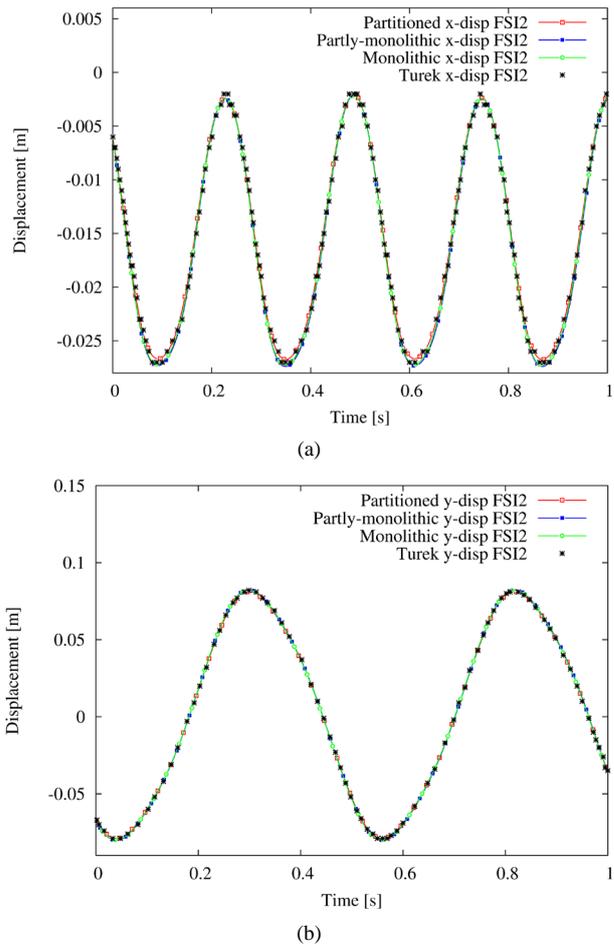
## 7.3. Computations

In Turek and Hron's FSI2 simulation benchmark, the system starts at rest and the speed of the inflow grows until it levels off to some steady value. After this initial transient period, which is approximately the first twelve seconds of simulation time, the solution reaches a steady state of oscillation. All results over a given periodic interval after this time are interchangeable, and as such we have normalized the comparison time interval on all forthcoming images.

We were able to reproduce the results given by Turek and Hron, for the x- and y-displacement of point A, at the mid-point of the tip of the tail, and the lift and drag generated about the combined surface of the flag and cylinder. Our calculations are nearly indistinguishable whether

using the partitioned, quasi-direct, or monolithic coupling, as seen in **Figures 8** and **9**. **Table 2** provides numerical values for the maximum and minimum of the displacement, drag, and lift from both Turek and Hron's work and our own, along with the percent error between our solutions and the validation data. The maximum deviation from the validation data for any quantity is 4%, with the three coupling methods each being the most accurate for some quantities and the least for others. Overall, the visual and numerical results demonstrate the coupling methods are all within an acceptable deviation from the validation data, and can be confirmed to functioning correctly. Even more so, we have demonstrated that using the same convergence criterion for the global FSI problem has resulted in three solutions that are quite similar to each other.

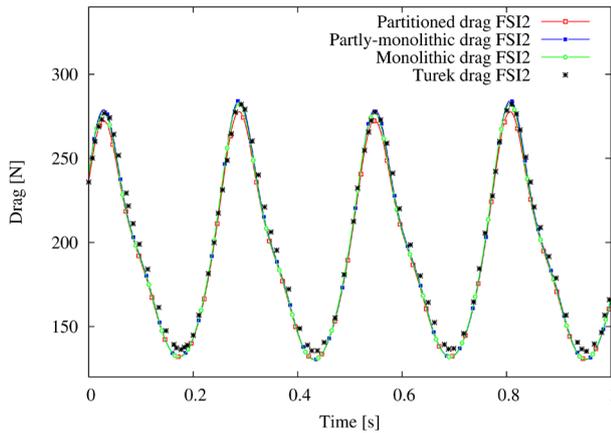
Finally, **Figure 10** plots the wall clock runtime of the three coupling techniques over 20 seconds of simulation time. The noticeable change in slopes at approximately 7 simulation seconds aligns with the time at which the flag



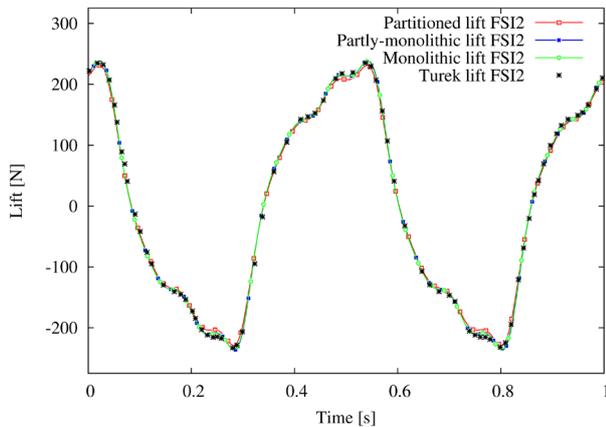
**Figure 8.** The x- and y-displacement of point A on the flag tip. Our monolithic and quasi-direct schemes show nearly identical results to Turek and Hron [2], while our partitioned scheme is slightly different.

**Table 2.** Maximum and minimum values for the x- and y-displacements of point A, and the lift and drag generated about the combined surface of the flag and cylinder. The results from Turek and Hron [2] are presented alongside ours, and our percent error from theirs is calculated.

Quantity	Turek & Hron	Partitioned	% error	Quasi-Direct	% error	Monolithic	% error
max x-disp [E-3 m]	-2.14	-2.05	4.00%	-2.12	1.05%	-2.05	2.23%
min x-disp [E-3 m]	-27.02	-26.75	0.99%	-27.40	1.41%	-27.26	0.89%
max y-disp [E-3 m]	81.83	81.21	0.76%	82.40	0.69%	82.18	0.43%
min y-disp [E-3 m]	-79.37	-78.97	0.50%	-80.15	0.98%	-79.93	0.71%
max drag [N]	282.58	277.92	1.65%	284.58	0.71%	282.71	0.05%
min drag [N]	135.08	130.56	3.35%	130.25	3.58%	130.25	3.58%
max lift [N]	235.08	231.67	1.45%	239.43	1.85%	231.81	1.59%
min lift [N]	-233.32	-229.20	1.77%	-237.39	1.74%	-236.75	1.39%

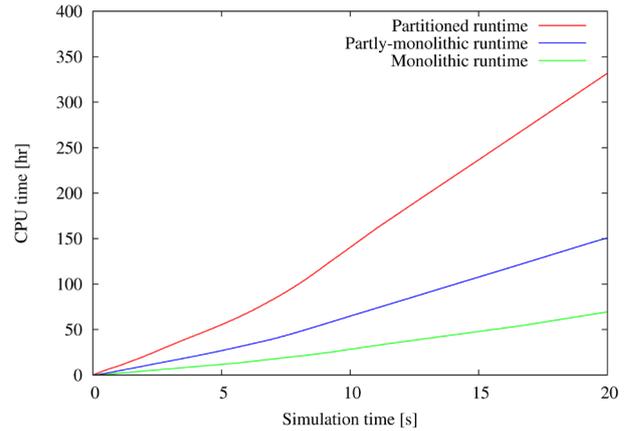


(a)



(b)

**Figure 9.** The drag and lift generated about the combined surface of the flag and cylinder. All three of our schemes show larger minimum values than Turek and Hron [2] do when calculating the drag, with otherwise good agreement. Our monolithic and quasi-direct schemes capture the complex shape of of Turek and Hron’s lift results [2], while our partitioned scheme matches the global maximum and minimum values but not all local the minimums and maximums in-between.



**Figure 10.** Time study for partitioned, quasi-direct, and monolithic coupling over 20 simulation seconds (8000 time-steps). Our partitioned coupling scheme required over twice the amount of CPU time per time step compared to our quasi-direct coupling scheme, which in turn required approximately twice that of our monolithic coupling scheme.

oscillations began to be significant, thus requiring more sub-iterations of the FSI algorithm. As can be seen, for this benchmark case the partitioned scheme requires over twice the amount of CPU time to solve any given time-step compared to the quasi-direct scheme, and the monolithic scheme requires even less CPU time than that. Given that we have held each solver to the same convergence criterion, and given that all other aspects of the simulation are held constant, we can state that for this problem, the monolithic scheme is certainly more efficient.

## 8. Summary and Conclusions

A methodology has been presented for comparing FSI coupling algorithms in the most fair and unbiased manner we could conceive. We developed a numerical ALE framework that allows for different FSI coupling algo-

gorithms to be evaluated for accuracy and efficiency in a quantitative manner. By holding constant many of the usually confounding factors, such as the grid, mesh motion, discretization approach, and solver tolerances, we were able to compare three different coupling strategies. These include a Dirichlet-Neumann partitioned coupling method, a monolithic coupling method, and quasi-direct coupling method. Each of these is tightly-coupled and uses the same convergence criterion to define when the global FSI solution is converged during a given time step. All three coupling methods use the same software infrastructure, system of equations assembly code, algebraic solver, and parameters. Other than their inherent differences in methodology, they are identical.

The benchmark proposed by Turek and Hron [2] was used to test the FSI coupling strategies developed in this work. We were able to achieve results which accurately reproduce those provided by Turek and Hron to within four percent at worst, generally within two percent. For this case, our results show that monolithic coupling and quasi-direct coupling are slightly more accurate, with respect to the validation data, than partitioned coupling and slightly better at capturing fine details along the interface. We also found that monolithic coupling and quasi-direct coupling are significantly faster than partitioned coupling when held to tight coupling tolerances. This substantiates Michler's hypothesis discussed earlier, that monolithic schemes can outperform partitioned schemes when held to the same tolerances [9], at least for the specific problem consider herein.

While this work aims to present a methodology for proper comparison between ALE FSI coupling algorithms, we acknowledge that the benchmark example considered is neither exhaustive nor addresses many of the other practical simulation issues. In the weak-coupling limit, we might expect that a partitioned or quasi-direct approach would be more efficient than a monolithic approach. Large three-dimensional problems may also change the comparison results: as the number of degrees of freedom grows quickly, the cost of solving a large linear system becomes the main computational cost and scalability needs to be addressed. The choice of linear algebraic solvers could also greatly sway the study, as extremely efficient methods for solving each subproblem exist, but cannot be applied to the monolithic system of equations. The governing equations and the choice of discretization scheme will also undoubtedly affect any sort of computational cost comparisons. Finally, there exist numerous other ALE partitioned coupling methods (e.g. Robin-Robin coupling rather than Dirichlet-Neumann) that may be more accurate and efficient than the methods we investigated. These aspects of the ALE FSI solution can be studied within our numerical framework, and we leave them as future research efforts.

## Acknowledgements

J. Sheldon gratefully acknowledges financial support from Naval Sea Systems Command, Advanced Submarine Systems Development Office (SEA 073R) with Ms. Diane Segelhorst and Mr. Patrick Tyler as program managers. S. Miller and J. Pitt acknowledge internal support for this research from the Applied Research Laboratory at The Pennsylvania State University.

## REFERENCES

- [1] W. Bangerth, T. Heister and G. Kanschat, "Deal. II Differential Equations Analysis Library," Technical Reference. <http://www.dealii.org>
- [2] S. Turek and J. Hron, "Proposal for Numerical Benchmarking of Fluid-Structure Interaction between an Elastic Object and Laminar Incompressible Flow," *Fluid-Structure Interaction*, 2006, pp. 371-385.
- [3] T. Wick, "Fluid-Structure Interactions Using Different Mesh Motion Techniques," *Computers & Structures*, Vol. 89, No. 13, 2011, pp. 1456-1467. <http://dx.doi.org/10.1016/j.compstruc.2011.02.019>
- [4] H. G. Matthies and J. Steindorf, "Partitioned but Strongly Coupled Iteration Schemes for Nonlinear Fluid-Structure Interaction," *Computers & Structures*, Vol. 80, No. 27, 2002, pp. 1991-1999. [http://dx.doi.org/10.1016/S0045-7949\(02\)00259-6](http://dx.doi.org/10.1016/S0045-7949(02)00259-6)
- [5] C. Förster, W. A. Wall and E. Ramm, "Artificial Added Mass Instabilities in Sequential Staggered Coupling of Nonlinear Structures and Incompressible Viscous Flows," *Computer Methods in Applied Mechanics and Engineering*, Vol. 196, No. 7, 2007, pp. 1278-1293. <http://dx.doi.org/10.1016/j.cma.2006.09.002>
- [6] J. Degroote, K. Bathe and J. Vierendeels, "Performance of a New Partitioned Procedure versus a Monolithic Procedure in Fluid-Structure Interaction," *Computers & Structures*, Vol. 87, No. 11, 2009, pp. 793-801. <http://dx.doi.org/10.1016/j.compstruc.2008.11.013>
- [7] G. Hou, J. Wang and A. Layton, "Numerical Methods for Fluid-Structure Interaction—A Review," *Communications in Computational Physics*, Vol. 12, No. 2, 2012, pp. 337-377.
- [8] M. Heil, "An Efficient Solver for the Fully Coupled Solution of Large-Displacement Fluid-Structure Interaction Problems," *Computer Methods in Applied Mechanics and Engineering*, Vol. 193, No. 1-2, 2004, pp. 1-23. <http://dx.doi.org/10.1016/j.cma.2003.09.006>
- [9] C. Michler, "A Monolithic Approach to Fluid-Structure Interaction," *Computers & Fluids*, Vol. 33, No. 5-6, 2004, pp. 839-848. <http://dx.doi.org/10.1016/j.compfluid.2003.06.006>
- [10] ADINA, "Adina fsi," Cited 08-23-2012.
- [11] M. Heil and A. Hazel, "Oomph-lib," Cited 09-12-2012.
- [12] M. Gurtin, E. Fried and L. Anand, "The mechanics and Thermodynamics of Continua," Cambridge University Press, 2009.
- [13] R. Bowen, "Introduction to Continuum Mechanics for

- Engineers,” Plenum Press, New York, 1989.  
<http://dx.doi.org/10.1007/978-1-4684-7761-0>
- [14] P. Chadwick, *Continuum Mechanics: Concise Theory and problems*. Dover Publications, 1999.
- [15] A. Spencer, “Continuum mechanics,” Dover Publications, 2004.
- [16] T. Tezduyar, K. Takizawa, C. Moorman, S. Wright, and J. Christopher, “Space-Time Finite Element Computation of Complex Fluid-Structure Interactions,” *International Journal for Numerical Methods in Fluids*, Vol. 64, No. 10-12, 2010, pp. 1201-1218.  
<http://dx.doi.org/10.1002/flid.2221>
- [17] Y. Bazilevs, K. Takizawa and T. E. Tezduyar, “Computational Fluid-Structure Interaction: Methods and Applications,” Wiley, New York, 2012.
- [18] C. Peskin, “The Immersed Boundary Method,” *Acta Numerica*, Vol. 11, 2002, pp. 479-517.
- [19] M. Uhlmann, “An Immersed Boundary Method with Direct Forcing for the Simulation of Particulate Flows,” *Journal of Computational Physics*, Vol. 209, No. 2, 2005, pp. 448-476. <http://dx.doi.org/10.1016/j.jcp.2005.03.017>
- [20] J. Hron and S. Turek, “A Monolithic Fem/Multigrid Solver for an Ale Formulation of Fluid-Structure Interaction with Applications in Biomechanics,” *Fluid-Structure Interaction*, Vol. 53, 2006, pp. 146-170.
- [21] T. Richter and T. Wick, “Finite Elements for Fluid-Structure Interaction in Ale and Fully Eulerian Coordinates,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 199, No. 41, 2010, pp. 2633-2642.  
<http://dx.doi.org/10.1016/j.cma.2010.04.016>
- [22] M. P. Robert Cook, David Malkus and R. Witt, “Concepts and Applications of Finite Element Analysis,” JohnWiley & Sons, 4 Edition, 2001.
- [23] U. Kuttler and W. A. Wall, “Fixed-Point Fluid-Structure Interaction Solvers with Dynamic Relaxation,” *Computational Mechanics*, Vol. 43, No. 1, 2008, pp. 61-72.  
<http://dx.doi.org/10.1007/s00466-008-0255-5>
- [24] R. Campbell and E. Paterson, “Fluid-Structure Interaction Analysis of Flexible Turbomachinery,” Ph.D. Thesis, The Pennsylvania State University, 2011.
- [25] T. E. Tezduyar, S. Sathe, R. Keedy and K. Stein, “Space-Time Finite Element Techniques for Computation of Fluid-Structure Interactions,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 195, No. 17, 2006, pp. 2002-2027.  
<http://dx.doi.org/10.1016/j.cma.2004.09.014>
- [26] S. Kang, H. Choi and J. Yoo, “Investigation of Fluid-Structure Interactions Using a Velocity-Linked p2/p1 Finite Element Method and the Generalized-A Method,” *International Journal for Numerical Methods in Engineering*, Vol. 90, No. 12, 2012, p. 1529.
- [27] C. Roy, C. Nelson, T. Smith and C. Ober, “Verification of Euler/Navier-Stokes Codes Using the Method of Manufactured Solutions,” *International Journal for Numerical Methods in Fluids*, Vol. 44, No. 6, 2004, pp. 599-620.  
<http://dx.doi.org/10.1002/flid.660>
- [28] P. J. Roach, “Code Verification by the Method of Manufactured Solutions,” *Journal of Fluids Engineering-Transactions of the ASME*, Vol. 124, No. 1, 2002, pp. 4-10.
- [29] J. Sheldon, “A Comparison of Fluid-Structure Interaction Coupling Algorithms Using the Finite Element Method,” Master’s Thesis, The Pennsylvania State University, 2012.
- [30] S. Osher and R. Fedkiw, “Level Set Methods and Dynamic Implicit Surfaces,” Springer, New York, 2003.
- [31] T. Davis, “Algorithm 832: Umfpack v4. 3-An unsymmetric-pattern multifrontal method,” *ACM Transactions on Mathematical Software (TOMS)*, Vol. 30, No. 2, 2004, pp. 196-199. <http://dx.doi.org/10.1145/992200.992206>
- [32] S. Deparis, M. A. Fern’andez and L. Formaggia, “Acceleration of a Fixed Point Algorithm for Fluid-Structure Interaction Using Transpiration Conditions,” *ESAIM: Mathematical Modelling and Numerical Analysis*, Vol. 37, No. 4, 2003, pp. 601-616.

## Appendix

### Linearization

Problems 1, 2, and 4 are non-linear systems of equations which we have chosen to solve numerically via the Newton-Raphson (NR) method. Problem 3 is linear, but we formulate the solution via the NR method for consistency. For the partitioned and quasi-direct algorithms, the mesh solution is obtained with a single NR iteration. Given an initial solution estimate (computed from the previous time step or initial conditions) iterate over the following problem until convergence is reached:

#### Problem 5 (Newton-Raphson iterate)

Given a solution vector  $\mathbf{U}^i$ , compute  $\mathbf{U}^{i+1}$  via

$$\mathbf{U}^{i+1} = \mathbf{U}^i - \delta\mathbf{U}^i, \quad [\mathbf{J}(\mathbf{U}^i)]\delta\mathbf{U}^i = \mathbf{R}(\mathbf{U}^i), \quad (67)$$

where the vector  $\mathbf{R}$  is the *residual vector*,  $\mathbf{J}$  is the *Jacobian matrix*, and  $\delta\mathbf{U}^i$  is the *solution update vector*.

The solution and residual vectors are

$$\mathbf{U} = \begin{bmatrix} \mathbf{U}_s \\ \mathbf{V}_s \\ \mathbf{V}_f \\ \mathbf{P}_f \\ \mathbf{U}_m \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} \mathbf{R}_{u_s} \\ \mathbf{R}_{v_s} \\ \mathbf{R}_{v_f} \\ \mathbf{R}_{p_f} \\ \mathbf{R}_{u_m} \end{bmatrix} \quad (68)$$

The sub-vectors within  $\mathbf{U}$  correspond to the nodal values of the solid, fluid, and mesh component blocks of the solution. The individual components of the residual

vector are

$$\mathbf{R}_{u_s} = (\dot{\mathbf{u}}_s, k_{\Omega_S} \hat{\mathbf{u}}_s)_{\Omega_S} - (\mathbf{v}_s, k_{\Omega_S} \hat{\mathbf{u}}_s)_{\Omega_S}, \quad (69)$$

$$\begin{aligned} \mathbf{R}_{v_s} = & (\rho_s \dot{\mathbf{v}}_s, \hat{\mathbf{v}}_s)_{\Omega_S} + (\mathbf{T}_R, \text{Grad} \hat{\mathbf{v}}_s)_{\Omega_S} \\ & - (\rho_s \mathbf{b}_s, \hat{\mathbf{v}}_s)_{\Omega_S} - (\bar{\mathbf{t}}_s, \hat{\mathbf{v}}_s)_{\Gamma_{NS}} \\ & - (\mathbf{t}_s, \hat{\mathbf{v}}_s)_{\Gamma_{FSN}} + ((\mathbf{v}_s - \mathbf{v}_\kappa), \hat{\mathbf{v}}_s)_{\Gamma_{FSD}}, \end{aligned} \quad (70)$$

$$\begin{aligned} \mathbf{R}_{v_f} = & (J \rho_\kappa (\dot{\mathbf{v}}_\kappa - (\text{Grad} \mathbf{v}_\kappa \mathbf{F}^{-1})[\mathbf{v}_m]), \hat{\mathbf{v}}_\kappa)_{\Omega_F} \\ & + (J \rho_\kappa (\text{Grad} \mathbf{v}_\kappa \mathbf{F}^{-1})[\mathbf{v}_m], \hat{\mathbf{v}}_\kappa)_{\Omega_F} \\ & + (\mathbf{J} \mathbf{T}_\kappa \mathbf{F}^{-T}, \text{Grad} \hat{\mathbf{v}}_\kappa)_{\Omega_F} - (\bar{\mathbf{t}}_\kappa, \hat{\mathbf{v}}_\kappa)_{\Gamma_{NF}} \\ & + ((\mathbf{v}_s - \mathbf{v}_\kappa), \hat{\mathbf{v}}_\kappa)_{\Gamma_{FSD}} - (\mathbf{t}_\kappa, \hat{\mathbf{v}}_\kappa)_{\Gamma_{FSN}}, \end{aligned} \quad (71)$$

$$\mathbf{R}_{p_f} = (\mathbf{J} \mathbf{F}^{-1} : \text{Grad} \mathbf{v}_\kappa^T, \hat{p}_\kappa)_{\Omega_F}, \quad (72)$$

$$\begin{aligned} \mathbf{R}_{u_m} = & (\mu_m (\text{Grad} \mathbf{u}_m + \text{Grad} \mathbf{u}_m^T), \text{Grad} \hat{\mathbf{u}}_m)_{\Omega_F} \\ & + (\lambda_m (\text{Div} \mathbf{u}_m), \text{Div} \hat{\mathbf{u}}_m)_{\Omega_F} + ((\mathbf{u}_m - \mathbf{u}_s), \hat{\mathbf{u}}_m)_{\Gamma_{FSD}}. \end{aligned} \quad (73)$$

The Jacobian matrix is formed by taking the the derivative of the residual vector with respect to the nodal unknowns:

$$\mathbf{J} = \frac{\partial \mathbf{R}}{\partial \mathbf{U}} \quad (74)$$

In the most general form, the discrete equations for the monolithic system are written as

$$\begin{bmatrix} \mathbf{J}_{u_s u_s} & \mathbf{J}_{u_s v_s} & \mathbf{J}_{u_s v_f} & \mathbf{J}_{u_s p_f} & \mathbf{J}_{u_s u_m} \\ \mathbf{J}_{v_s u_s} & \mathbf{J}_{v_s v_s} & \mathbf{J}_{v_s v_f} & \mathbf{J}_{v_s p_f} & \mathbf{J}_{v_s u_m} \\ \mathbf{J}_{v_f u_s} & \mathbf{J}_{v_f v_s} & \mathbf{J}_{v_f v_f} & \mathbf{J}_{v_f p_f} & \mathbf{J}_{v_f u_m} \\ \mathbf{J}_{p_f u_s} & \mathbf{J}_{p_f v_s} & \mathbf{J}_{p_f v_f} & \mathbf{J}_{p_f p_f} & \mathbf{J}_{p_f u_m} \\ \mathbf{J}_{u_m u_s} & \mathbf{J}_{u_m v_s} & \mathbf{J}_{u_m v_f} & \mathbf{J}_{u_m p_f} & \mathbf{J}_{u_m u_m} \end{bmatrix} \begin{bmatrix} \mathbf{U}_s \\ \mathbf{V}_s \\ \mathbf{V}_f \\ \mathbf{P}_f \\ \mathbf{U}_m \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{u_s} \\ \mathbf{R}_{v_s} \\ \mathbf{R}_{v_f} \\ \mathbf{R}_{p_f} \\ \mathbf{R}_{u_m} \end{bmatrix}. \quad (75)$$

Although this system seems daunting, application of the residual vector components, from (69), (70), (71), (72), and (73), to the Jacobian definition in (74) shows

that many of the terms are zero for all coupling algorithms, reducing the system to

$$\begin{bmatrix} \mathbf{J}_{u_s u_s} & \mathbf{J}_{u_s v_s} & 0 & 0 & 0 \\ \mathbf{J}_{v_s u_s} & \mathbf{J}_{v_s v_s} & \mathbf{J}_{v_s v_f} & \mathbf{J}_{v_s p_f} & \mathbf{J}_{v_s u_m} \\ 0 & \mathbf{J}_{v_f v_s} & \mathbf{J}_{v_f v_f} & \mathbf{J}_{v_f p_f} & \mathbf{J}_{v_f u_m} \\ 0 & 0 & \mathbf{J}_{p_f v_f} & 0 & \mathbf{J}_{p_f u_m} \\ \mathbf{J}_{u_m u_s} & 0 & 0 & 0 & \mathbf{J}_{u_m u_m} \end{bmatrix} \begin{bmatrix} \mathbf{U}_s \\ \mathbf{V}_s \\ \mathbf{V}_f \\ \mathbf{P}_f \\ \mathbf{U}_m \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{u_s} \\ \mathbf{R}_{v_s} \\ \mathbf{R}_{v_f} \\ \mathbf{R}_{p_f} \\ \mathbf{R}_{u_m} \end{bmatrix}. \quad (76)$$

### Discrete Monolithic Equations

When discretized monolithically, (76) is further simplified due to the conforming velocity fields given in (58) and (59). As shown in (60), these velocity fields

lead to the tractions on the fluid-solid interface not contributing to the monolithic (or quasi-direct) equations. This eliminates  $J_{v_s v_f}$ ,  $J_{v_s p_f}$ ,  $J_{v_f v_s}$ , and  $J_{v_s u_m}$  from (76), resulting in

$$\begin{bmatrix} J_{u_s u_s} & J_{u_s v_s} & 0 & 0 & 0 \\ J_{v_s u_s} & J_{v_s v_s} & C_{v_s v_f} & 0 & 0 \\ 0 & C_{v_f v_s} & J_{v_f v_f} & J_{v_f p_f} & J_{v_f u_m} \\ 0 & 0 & J_{p_f v_f} & 0 & J_{p_f u_m} \\ J_{u_m u_s} & 0 & 0 & 0 & J_{u_m u_m} \end{bmatrix} \begin{bmatrix} U_s \\ V_s \\ V_f \\ P_f \\ U_m \end{bmatrix} = \begin{bmatrix} R_{u_s} \\ R_{v_s} \\ R_{v_f} \\ R_{p_f} \\ R_{u_m} \end{bmatrix}, \quad (77)$$

where the terms marked by  $C$  arise due to the conforming velocity space in (58).

### Discrete Quasi-Direct Equations

The quasi-direct discretization of (76) is similar to the monolithic discretization in that the tractions do not contribute on the interface. This results in the same terms being eliminated as were in the monolithic case. Furthermore, as the quasi-direct approach only solves Problems 1 and 2 monolithically, and iterates their solution with that of Problem 3, (76) can be separated into two smaller systems which reflect the coupled fluid-solid and separate mesh:

$$\begin{bmatrix} J_{u_s u_s} & J_{u_s v_s} & 0 & 0 \\ J_{v_s u_s} & J_{v_s v_s} & C_{v_s v_f} & 0 \\ 0 & C_{v_f v_s} & J_{v_f v_f} & J_{v_f p_f} \\ 0 & 0 & J_{p_f v_f} & 0 \end{bmatrix} \begin{bmatrix} U_s \\ V_s \\ V_f \\ P_f \end{bmatrix} = \begin{bmatrix} R_{u_s} \\ R_{v_s} \\ R_{v_f} \\ R_{p_f} \end{bmatrix}, \quad (78)$$

$$\begin{bmatrix} J_{u_m u_m} \end{bmatrix} \begin{bmatrix} U_m \end{bmatrix} = \begin{bmatrix} R_{u_m} \end{bmatrix}. \quad (79)$$

### Discrete Partitioned Equations

With partitioned coupling, Problems 1, 2, and 3 are all solved sequentially. This allows for (76) to be simplified into three independent smaller systems for the solid, fluid, and mesh:

$$\begin{bmatrix} J_{u_s u_s} & J_{u_s v_s} \\ J_{v_s u_s} & J_{v_s v_s} \end{bmatrix} \begin{bmatrix} U_s \\ V_s \end{bmatrix} = \begin{bmatrix} R_{u_s} \\ R_{v_s} \end{bmatrix}, \quad (80)$$

$$\begin{bmatrix} J_{v_f v_f} & J_{v_f p_f} \\ J_{p_f v_f} & 0 \end{bmatrix} \begin{bmatrix} V_f \\ P_f \end{bmatrix} = \begin{bmatrix} R_{v_f} \\ R_{p_f} \end{bmatrix}, \quad (81)$$

$$\begin{bmatrix} J_{u_m u_m} \end{bmatrix} \begin{bmatrix} U_m \end{bmatrix} = \begin{bmatrix} R_{u_m} \end{bmatrix}. \quad (82)$$

### Jacobian Component Calculations

The actual Jacobian calculations are simply derivatives of the residuals presented previously, and so they have been omitted. When evaluating the Jacobians, the following tensor calculus identities from Gurtin [12] are useful:

$$J' = J \operatorname{tr}(\mathbf{F} \mathbf{F}^{-1}), \quad (83)$$

$$(\mathbf{F}^{-T}) = \left( (\mathbf{F}^{-1})' \right)^T, \quad (84)$$

$$(\mathbf{F}^{-1}) = -\mathbf{F}^{-1} \mathbf{F}' \mathbf{F}^{-1}. \quad (85)$$

In our calculations, we approximate the mesh (fluid domain) velocity within each time step as

$$\mathbf{v}_m \approx \frac{\mathbf{u}_m^h(\mathbf{X}, t_{n+1}) - \mathbf{u}_m^h(\mathbf{X}, t_n)}{t_{n+1} - t_n} \Rightarrow \mathbf{v}'_m = \frac{\hat{\mathbf{u}}_m}{\Delta t}, \quad (86)$$

Because it is not one of our primitive variables.