Partitioned and Monolithic Algorithms for the Numerical Solution of Cardiac Fluid-Structure Interaction

Michele Bucelli[1,*], Luca Dede'[1], Alfio Quarteroni[1,2] and Christian Vergara[3]

[1] *MOX, Dipartimento di Matematica, Politecnico di Milano, P.zza Leonardo da Vinci 32, 20133 Milan, Italy.*
[2] *Mathematics Institute, EPFL, Av. Piccard, CH-1015 Lausanne, Switzerland (Professor Emeritus).*
[3] *LABS, Dipartimento di Chimica, Materiali e Ingegneria Chimica "Giulio Natta", Politecnico di Milano, P.zza Leonardo da Vinci 32, 20133 Milan, Italy.*

Abstract. We review and compare different fluid-structure interaction (FSI) numerical methods in the context of heart modeling, aiming at assessing their computational efficiency for cardiac numerical simulations and selecting the most appropriate method for heart FSI. Blood dynamics within the human heart is characterized by active muscular action, during both contraction and relaxation phases of the heartbeat. The efficient solution of the FSI problem in this context is challenging, due to the added-mass effect (caused by the comparable densities of fluid and solid, typical of biomechanics) and to the complexity, nonlinearity and anisotropy of cardiac consitutive laws. In this work, we review existing numerical coupling schemes for FSI in the two classes of strongly-coupled partitioned and monolithic schemes. The schemes are compared on numerical tests that mimic the flow regime characterizing the heartbeat in a human ventricle, during both systole and diastole. Active mechanics is treated in both the active stress and active strain frameworks. Computational costs suggest the use of a monolithic method. We employ it to simulate a full heartbeat of a human ventricle, showing how it allows to efficiently obtain physiologically meaningful results.

*Corresponding author. *Email addresses:* `michele.bucelli@polimi.it` (M. Bucelli), `luca.dede@polimi.it` (L. Dede'), `alfio.quarteroni@polimi.it` (A. Quarteroni), `christian.vergara@polimi.it` (C. Vergara)

1 Introduction

The aim of this paper is to provide for the first time a systematic review and comparison of different fluid-structure interaction (FSI) numerical coupling schemes in the context of cardiac hemodynamics. In particular, we consider partitioned fully-coupled and monolithic algorithms and we analyze their effectiveness in both systolic and diastolic phases, and with both active stress and active strain modeling frameworks for muscular contraction. We investigate the performance of the schemes during the different phases of the heartbeat, in order to assess to which extent they depend on the specific physical features of such phases, in particular the presence of active forces.

The human heart acts as a pump, driven by the electrical activation of its cells, whose purpose is to force the blood into the circulatory system, allowing the delivery of oxygen and nutrients to the whole body [66]. The feedback mechanism between blood and cardiac muscle is relevant in determining the cardiac function and its response to pathological conditions [97,98]. Numerical simulations offer a valid tool for the investigation of this mechanism [87].

A large number of computational studies model the fluid-solid feedback in the heart only in terms of a zero dimensional, lumped model for the blood flow [12, 21, 57, 67], mostly focusing on the electromechanical processes [12,43,53,92,98]. Alternatively, three-dimensional models for the blood flow are one-way coupled to mechanical models, receiving as input results of mechanical simulations to prescribe the boundary displacement of the fluid domain, but without feedback from the three-dimensional fluid model to the solid [65,106,117]. While FSI models for cardiac valves have been extensively studied [11, 30, 36, 56, 61, 73, 104], three-dimensional fluid dynamics models of blood in the cardiac cavities are seldom two-ways coupled with mechanical models for the cardiac muscle [28,75,81,99,102,111,113], due to the inherent complexity and computational cost of FSI simulations.

In the context of biomechanics, solving FSI problems poses significant challenges on the stability and efficiency of the numerical solution, mainly because of the comparable densities of fluid and solid (resulting in the added mass effect [27]) and the anisotropy and nonlinearity of the constitutive laws [51,58]. Appropriate schemes are required to enforce the fluid-solid coupling in a computationally efficient way.

The FSI numerical coupling schemes that have been proposed in the literature (see, e.g., [20, 29, 56, 59, 70]) can be roughly classified into partitioned loosely coupled (or explicit) schemes [18, 24, 25, 40, 49, 50, 52], partitioned fully coupled (or fixed-point, or implicit) schemes [14, 15, 27, 69, 70, 76, 78] and monolithic (or Newton-based) schemes [45,55,64,81,93,114,115]. The schemes differ significantly in their modularity and in the implementation effort that they require, and in terms of their performance [68,70].

The effectiveness of FSI schemes in the context of vascular hemodynamics has been widely studied, e.g., in [14,15,68,70]. However, the benchmarks and test cases under consideration were mostly related to the flow of blood within large vessels, rather than to the flow within cardiac chambers. In the heart, the flow is mostly driven by the interaction of

the blood with the active muscular action (either contraction or relaxation, changing over time according to the heartbeat phase), and possibly with cardiac valves. The resulting flow is characterized by alternating systolic and diastolic phases. The former is characterized by fast dynamics, driven by the contraction of the muscle, with the flow featuring a transition regime towards turbulence [22, 109, 110, 116]. Diastole on the other hand is characterized by slower dynamics, and the flow is determined by the interplay between the ventricle relaxation and passive mechanical properties together with inflow conditions, resulting in a balloon-type problem [15]. Moreover, the material models required to describe effectively the muscular tissue are characterized by significant nonlinearity and anisotropy [51, 58]. All these aspects play a role in the effectiveness and efficiency of the schemes used for the solution of the FSI problem.

After introducing the mathematical models used, we review several existing FSI coupling schemes. We consider benchmark cases that mimic the characteristics of the human left ventricle, although in a simplified setting from the geometric viewpoint. Different tests allow us to separately analyze the systolic ejection and diastolic filling phases of the heartbeat. Concerning in particular the systolic phase, we employ both the active stress and active strain modeling frameworks for the active contraction of the cardiac muscle, and for both we compare the effectiveness of the coupling schemes. The computational times for the different methods indicate that, for all phases and with both activation models, a monolithic approach is more suited to the strong coupling of fluid and structure in the cardiac context. Finally, we present a numerical example simulating a full heartbeat, including isovolumetric phases.

The rest of the paper is organized as follows: Section 2 introduces the mathematical models that are used in the benchmark cases, and Section 3 briefly describes the temporal and spatial discretization schemes employed. Section 4 reviews the different coupling schemes under consideration. Section 5 discusses the numerical tests, and Section 6 draws some conclusive remarks.

2   Mathematical models for cardiac fluid-structure interaction

We consider a mechanical system defined in an open, bounded, time dependent domain $\Omega^t \subset \mathbb{R}^3$ (where $t \in [0, T)$ denotes the time variable). Such domain is decomposed into two subdomains Ω_f^t and Ω_s^t, the former occupied by a Newtonian, incompressible fluid, the latter occupied by a hyperelastic structure. We denote by $\Sigma^t = \partial\Omega_f^t \cap \partial\Omega_s^t$ the moving fluid structure interface, and by $\mathbf{n}$ the unit vector normal to Σ^t, outward directed from the fluid domain, inward directed into the structure domain.

To track the motion over time of the subdomains, we introduce a fixed reference configuration, denoted by a hat (see Fig. 1 for a schematic representation): $\widehat{\Omega}$, $\widehat{\Omega}_f$, $\widehat{\Omega}_s$, $\widehat{\Sigma}$ and $\widehat{\mathbf{n}}$ represent the reference configurations for the domain Ω, the fluid and solid domains, the fluid-solid interface and the normal unit vector respectively. We introduce the following
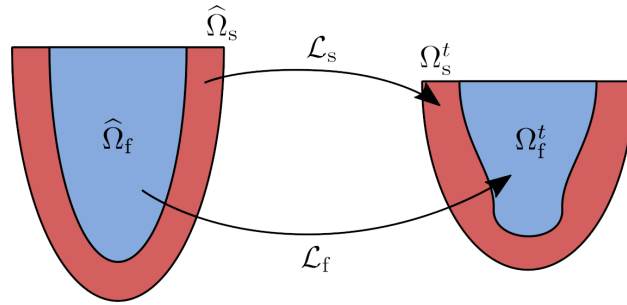
Figure 1: Schematic representation of the solid and fluid domains, in their reference (left) and current (right) configurations.

maps from the reference to the time-dependent configurations:

$$\mathcal{L}_s : \widehat{\Omega}_s \times (0,T) \to \Omega_s^t, \quad \Omega_s^t = \big\{ \mathbf{x} = \mathcal{L}_s(\widehat{\mathbf{x}},t), \widehat{\mathbf{x}} \in \widehat{\Omega}_s \big\},$$
$$\mathcal{L}_f : \widehat{\Omega}_f \times (0,T) \to \Omega_f^t, \quad \Omega_f^t = \big\{ \mathbf{x} = \mathcal{L}_f(\widehat{\mathbf{x}},t), \widehat{\mathbf{x}} \in \widehat{\Omega}_f \big\}.$$

The evolution in time of the time-dependent domains is then expressed by the time-dependence of the maps $\mathcal{L}_s$ and $\mathcal{L}_f$. The precise definition of the maps is different for the fluid and the solid domains, and is detailed later.

In the following, we will refer to the time-dependent domains as the current configuration. We will denote by $\mathbf{x} \in \Omega^t$ spatial coordinates in the current configuration, and with $\widehat{\mathbf{x}} \in \widehat{\Omega}$ spatial coordinates in the reference configuration.

2.1 Structure problem

We model the structure as a hyperelastic material in the Lagrangian setting [82]. Let ρ_s be the density of the structure, and $\widehat{\mathbf{d}}(\widehat{\mathbf{x}},t)$ be its displacement at point $\widehat{\mathbf{x}}$ in the reference configuration and time t. Then, the map $\mathcal{L}_s$ from the reference to the current configuration is given by

$$\mathbf{x}(\widehat{\mathbf{x}},t) = \mathcal{L}_s(\widehat{\mathbf{x}},t) = \widehat{\mathbf{x}} + \widehat{\mathbf{d}}(\widehat{\mathbf{x}},t), \quad \mathbf{x} \in \Omega_s, \quad \widehat{\mathbf{x}} \in \widehat{\Omega}_s, \quad t \in (0,T).$$

We will denote by $F = I + \widehat{\boldsymbol{\nabla}} \mathbf{d}$ the deformation gradient tensor, and with $J = \det F$ its determinant. The evolution of the structure displacement is described by the following partial differential equation (PDE), expressing the balance of momentum in the reference configuration [82]:

$$\rho_s \frac{\partial^2 \widehat{\mathbf{d}}}{\partial t^2} - \widehat{\nabla} \cdot P_s(\widehat{\mathbf{d}},t) = \mathbf{0} \quad \text{in} \quad \widehat{\Omega}_s \times (0,T), \tag{2.1}$$

endowed with suitable initial conditions prescribing the displacement $\widehat{\mathbf{d}}$ and its temporal derivative at time $t = 0$ and boundary conditions on $\partial \widehat{\Omega}_s \backslash \widehat{\Sigma}$ (while on $\widehat{\Sigma}$ the FSI interface conditions are imposed, as described in Section 2.4). In (2.1), $P_s(\widehat{\mathbf{d}},t)$ is the first Piola-Kirchhoff stress tensor, defined as a function of the displacement $\widehat{\mathbf{d}}$ by the constitutive

relation of the material. Since our aim is the mathematical modeling of cardiac mechanics, we incorporate active contraction in the stress tensor, either in the active stress [6,81,91, 99] or active strain [5,45,46,95] framework.

We remark that in principle active contraction in either framework is generated by coupling the mechanical model with an electrophysiology model through an active force generation model, thus resulting in an electromechanical model [13,46,53,87,88,91,95,99]. Since our focus is on the FSI numerical schemes, we prescribe the evolution of active stress or strain using an analytical law (hence the explicit dependence of P_s on the time t). In what follows, we address two models considered in the literature for the active contraction, together with the passive model.

2.1.1 Active mechanics: active stress formulation

Within the active stress framework, the tensor P_s is decomposed as the sum of a passive and an active part: $P_s = P_{pas} + P_{act}$. The passive part is defined as the derivative of a suitable strain energy function $\mathcal{W}(F)$

$$P_{pas} = \frac{\partial \mathcal{W}}{\partial F},$$

wherein $\mathcal{W}$ characterizes the passive properties of the material.

Let us introduce at each point of $\widehat{\Omega}_s$ a unit vector $\mathbf{f}$ that describes the local direction of the muscular fibers along which active force is generated [84]. The active part of the stress tensor is defined as

$$P_{act} = T_{act} \frac{F\mathbf{f}\otimes\mathbf{f}}{\sqrt{I_{4f}}},$$

where T_{act} is an active tension in the direction of fibers [6,91,99], and $I_{4f} = F\mathbf{f}\cdot F\mathbf{f}$. As stated before, T_{act} is not derived by the coupling with an electrophysiology model, rather, it is analytically prescribed.

2.1.2 Active mechanics: active strain formulation

In the active strain framework, the deformation gradient is assumed to be decomposed multiplicatively as $F = F_{pas}F_{act}$. F_{act} is the gradient of a deformation from the reference configuration to a virtual intermediate configuration only determined by the active contraction, and F_{pas} is the gradient of a deformation from this virtual configuration to the current configuration, due to the passive mechanical behavior of the material [6]. The stress tensor is defined as the derivative of the strain energy function with respect to the deformation gradient tensor. The strain energy function is expressed as function of the passive deformation gradient only, i.e.

$$P_s = \frac{\partial \mathcal{W}(F_{pas})}{\partial F}.$$

The active deformation gradient is defined as

$$F_{act} = I + \gamma_f \mathbf{f}\otimes\mathbf{f},$$

where γ_f is the strain in the direction of fibers [6,45,46,95]. For γ_f, the same comments of above regarding T_{act} hold true.

2.1.3   Passive mechanics: constitutive model

We consider the Guccione constitutive model [51,92,108], commonly used for cardiac tissue. We introduce at each point of $\widehat{\Omega}_s$ a reference system $\{\mathbf{f},\mathbf{s},\mathbf{n}\}$ describing the direction of fibers, sheetlets and cross-fibers respectively [84]. Then, the strain energy function is given by

$$\mathcal{W} = \frac{c}{2}\left(e^{Q(F)} - 1\right) + \frac{k}{2}(J-1)\log(J),$$

$$Q(F) = \sum_{\mathbf{i},\mathbf{j}\in\{\mathbf{f},\mathbf{s},\mathbf{n}\}} a_{\mathbf{i},\mathbf{j}}(E\mathbf{i}\cdot\mathbf{j}),$$

$$E = \frac{1}{2}\left(F^T F - I\right),$$

(2.2)

where c and $a_{\mathbf{i},\mathbf{j}}$, $\mathbf{i},\mathbf{j}\in\{\mathbf{f},\mathbf{s},\mathbf{n}\}$ are positive parameters and k is the bulk modulus, all tuned following [51,92]. The corresponding values are reported in Table 1.

We incorporate the quasi-incompressibility of the material by adding the term $\frac{k}{2}(J-1)\log(J)$ in the strain energy function that penalizes volume variations (i.e. penalizes values of J that are away from 1) [90,108].

For ventricular test cases, the reference system $\{\mathbf{f},\mathbf{s},\mathbf{n}\}$ is generated by means of the rule-based method proposed by Rossi et al. [95].

Table 1: Parameters used for the Guccione constitutive law [51,92].

Parameter	Value
c	$0.88\cdot 10^3\,\mathrm{Pa}$
k	$5\cdot 10^4\,\mathrm{Pa}$
$a_{\mathbf{f},\mathbf{f}}$	8
$a_{\mathbf{s},\mathbf{s}}$	6
$a_{\mathbf{n},\mathbf{n}}$	3
$a_{\mathbf{f},\mathbf{s}}$	12
$a_{\mathbf{f},\mathbf{n}}$	3
$a_{\mathbf{s},\mathbf{n}}$	3

2.2   Displacement of the fluid domain

We model the fluid dynamics in a moving domain in the Arbitrary Lagrangian Eulerian (ALE) framework [63,100]. The structure displacement at the fluid-structure interface is extended to the fluid domain by solving the following harmonic extension problem: for

every $t \in (0,T)$,

$$\begin{cases} -\widehat{\Delta}\widehat{\mathbf{d}}_f = \mathbf{0} & \text{in} \quad \widehat{\Omega}_f, \\ \widehat{\mathbf{d}}_f = \mathbf{0} & \text{on} \quad \widehat{\Gamma}_f^D, \\ (\widehat{\boldsymbol{\nabla}}\widehat{\mathbf{d}}_f)\widehat{\mathbf{n}} = \mathbf{0} & \text{on} \quad \widehat{\Gamma}_f^N, \end{cases} \tag{2.3}$$

where $\widehat{\Gamma}_f^D$ and $\widehat{\Gamma}_f^N$ are subsets of $\partial\widehat{\Omega}_f \backslash \widehat{\Sigma}$ on which Dirichlet and Neumann conditions, respectively, are prescribed. On $\widehat{\Sigma}$, interface conditions are prescribed as detailed in Section 2.4. The map $\mathcal{L}_f$ from the reference to the current configuration of the fluid domain is given by

$$\mathbf{x} = \widehat{\mathbf{x}} + \widehat{\mathbf{d}}_f(\widehat{\mathbf{x}},t), \quad \mathbf{x} \in \Omega_f, \quad \widehat{\mathbf{x}} \in \widehat{\Omega}_f, \quad t \in (0,T).$$

We define the fluid domain velocity $\mathbf{u}_f$ as the time derivative of the fluid domain displacement, i.e.

$$\widehat{\mathbf{u}}_f = \frac{\partial\widehat{\mathbf{d}}_f}{\partial t} \quad \text{in} \quad \widehat{\Omega}_f.$$

Then, we map it onto the current configuration through $\mathcal{L}_f$

$$\mathbf{u}_f = \widehat{\mathbf{u}}_f\big(\mathcal{L}_f^{-1}(\mathbf{x},t),t\big).$$

We remark that the extension of the displacement from the interface to the fluid domain is arbitrary, and other differential operators (other than the Laplacian) can be considered. In particular, a linear elasticity lifting operator can be employed to this aim [100].

2.3 Fluid dynamics problem

We model the blood as a Newtonian incompressible fluid. Denoting by ρ_f the fluid density and by $\mathbf{u}$ and p its velocity and pressure, respectively, the balance of momentum and mass conservation equations are expressed by the Navier-Stokes equations in ALE coordinates [86]

$$\begin{cases} \rho_f\left[\dfrac{\partial\mathbf{u}}{\partial t} + ((\mathbf{u}-\mathbf{u}_f)\cdot\boldsymbol{\nabla})\mathbf{u}\right] - \boldsymbol{\nabla}\cdot\sigma_f(\mathbf{u},p) = \mathbf{0} & \text{in} \quad \Omega_f^t \times (0,T), \\ \boldsymbol{\nabla}\cdot\mathbf{u} = 0 & \text{in} \quad \Omega_f^t \times (0,T), \end{cases}$$

endowed with initial conditions ($\mathbf{u}$ in Ω_f^0), and suitable boundary conditions on $\partial\Omega_f^t \backslash \Sigma^t$ (while FSI interface conditions are imposed on Σ^t, see Section 2.4). The Cauchy stress tensor $\sigma_f(\mathbf{u},p)$ is defined as

$$\sigma_f(\mathbf{u},p) = 2\mu\varepsilon(\mathbf{u}) - pI, \quad \varepsilon(\mathbf{u}) = \frac{1}{2}\big(\boldsymbol{\nabla}\mathbf{u} + \boldsymbol{\nabla}\mathbf{u}^T\big),$$

wherein μ is the dynamic viscosity of the fluid.

2.4   Fluid-structure coupling

A *geometric coupling* condition is imposed to enforce the continuity of displacements at the fluid-structure interface: $\widehat{\mathbf{d}}_f = \widehat{\mathbf{d}}$ on $\widehat{\Sigma}$. Besides, two other coupling conditions are needed. The velocity at the fluid-structure interface must be continuous, i.e. $\mathbf{u} = \frac{\partial \mathbf{d}}{\partial t}$ on Σ; this corresponds to prescribing a no-slip condition on the fluid-structure interface, and is referred to as *kinematic coupling*. Moreover, the traction at the fluid-structure interface must be continuous, i.e. $\sigma_f(\mathbf{u},p)\mathbf{n} = \sigma_s(\mathbf{d})\mathbf{n}$ on Σ^t, where $\sigma_s(\mathbf{d}) = J^{-1} F P_s(\mathbf{d})^T$ is the Cauchy stress tensor for the structure; this condition expresses Newton's third law across the fluid-structure interface, and is referred to as *dynamic coupling*.

The fully coupled FSI problem reads: find $\widehat{\mathbf{d}}, \widehat{\mathbf{d}}_f, \mathbf{u}, p$ such that

$$
\begin{cases}
\rho_s \dfrac{\partial^2 \widehat{\mathbf{d}}}{\partial t^2} - \widehat{\boldsymbol{\nabla}} \cdot P_s(\widehat{\mathbf{d}}, t) = \mathbf{0} & \text{in} \quad \widehat{\Omega}_s \times (0, T), \\[2mm]
-\widehat{\Delta} \widehat{\mathbf{d}}_f = \mathbf{0} & \text{in} \quad \widehat{\Omega}_f \times (0, T), \\[2mm]
\widehat{\mathbf{d}}_f = \widehat{\mathbf{d}} & \text{on} \quad \widehat{\Sigma} \times (0, T), \\[2mm]
(\widehat{\boldsymbol{\nabla}} \widehat{\mathbf{d}}_f) \widehat{\mathbf{n}} = \mathbf{0} & \text{on} \quad \partial \widehat{\Omega}_f \backslash \widehat{\Sigma} \times (0, T), \\[2mm]
\rho_f \left[\dfrac{\partial \mathbf{u}}{\partial t} + ((\mathbf{u} - \mathbf{u}_f) \cdot \boldsymbol{\nabla}) \mathbf{u} \right] - \boldsymbol{\nabla} \cdot \sigma_f(\mathbf{u}, p) = \mathbf{0} & \text{in} \quad \Omega_f^t \times (0, T), \\[2mm]
\boldsymbol{\nabla} \cdot \mathbf{u} = 0 & \text{in} \quad \Omega_f^t \times (0, T), \\[2mm]
\mathbf{u} = \dfrac{\partial \mathbf{d}}{\partial t} & \text{on} \quad \Sigma \times (0, T), \\[2mm]
\sigma_f(\mathbf{u}, p) \mathbf{n} = \sigma_s(\mathbf{d}) \mathbf{n} & \text{on} \quad \Sigma \times (0, T),
\end{cases}
\tag{2.4}
$$

endowed with suitable initial and boundary conditions for both the fluid and the solid.

To keep the notation light, we shall drop henceforth the hat over $\widehat{\mathbf{d}}$ and $\widehat{\mathbf{d}}_f$. The context will make clear whether we are considering quantities in reference or current configuration.

3   Time and space discretizations

We use finite differences for the time discretization of the FSI problem (2.4) [86]. Let us introduce a partition of the time domain $(0, T)$ into N_t intervals of width Δt of extremes $t^0 = 0, t^1, \ldots, t^{N_t} = T$. In the following, we will denote by a superscript on a solution variable (including the domain Ω_f and the interface Σ) the time-discrete approximation of that variable at that timestep (e.g. $\mathbf{u}^n \approx \mathbf{u}(t = t^n)$). For the Navier-Stokes momentum equation, we use the implicit Euler scheme together with a semi-implicit discretization of the advection term [86]. For the structure problem, we use a first order backward finite difference scheme. For every $n = 0, 1, \ldots, N_t - 1$, and assuming $\mathbf{d}^{-1} = \mathbf{d}^0$, the time-discrete

problem reads

$$
\begin{cases}
\rho_s \dfrac{\mathbf{d}^{n+1}-2\mathbf{d}^n+\mathbf{d}^{n-1}}{\Delta t^2} - \widehat{\boldsymbol{\nabla}} \cdot P_s(\mathbf{d}^{n+1}) = \mathbf{0} & \text{in} \quad \widehat{\Omega}_s, & \text{(3.1a)}\\[2mm]
-\widehat{\Delta}\mathbf{d}_f^{n+1} = \mathbf{0} & \text{in} \quad \widehat{\Omega}_f, & \text{(3.1b)}\\[2mm]
\mathbf{d}_f^{n+1} = \mathbf{d}^{n+1} & \text{on} \quad \widehat{\Sigma}, & \text{(3.1c)}\\[2mm]
\mathbf{d}_f^{n+1} = \mathbf{0} & \text{on} \quad \widehat{\Gamma}_f^D, & \text{(3.1d)}\\[2mm]
\left(\widehat{\boldsymbol{\nabla}}\mathbf{d}_f^{n+1}\right)\widehat{\mathbf{n}} = \mathbf{0} & \text{on} \quad \widehat{\Gamma}_f^N, & \text{(3.1e)}\\[2mm]
\mathbf{u}_f^{n+1} = \dfrac{\mathbf{d}_f^{n+1}-\mathbf{d}_f^n}{\Delta t} & \text{in} \quad \widehat{\Omega}_f, & \text{(3.1f)}\\[2mm]
\rho_f\left[\dfrac{\mathbf{u}^{n+1}-\mathbf{u}^n}{\Delta t} + \left(\left(\mathbf{u}^n-\mathbf{u}_f^{n+1}\right)\cdot\boldsymbol{\nabla}\right)\mathbf{u}^{n+1}\right] - \boldsymbol{\nabla}\cdot\sigma_f(\mathbf{u}^{n+1},p^{n+1}) = \mathbf{0} & \text{in} \quad \Omega_f^{n+1}, & \text{(3.1g)}\\[2mm]
\boldsymbol{\nabla}\cdot\mathbf{u}^{n+1} = 0 & \text{in} \quad \Omega_f^{n+1}, & \text{(3.1h)}\\[2mm]
\mathbf{u}^{n+1} = \dfrac{\mathbf{d}^{n+1}-\mathbf{d}^n}{\Delta t} & \text{on} \quad \Sigma^{n+1}, & \text{(3.1i)}\\[2mm]
\sigma_f(\mathbf{u}^{n+1},p^{n+1})\mathbf{n} = \sigma_s(\mathbf{d}^{n+1})\mathbf{n} & \text{on} \quad \Sigma^{n+1}. & \text{(3.1j)}
\end{cases}
$$

We remark that the extension to higher order finite difference schemes such as BDF2 or theta-schemes is straightforward.

Space discretization is obtained by means of the finite element method [62, 86], using hexahedral elements for both the fluid and the structure subdomains. We choose conforming meshes at the fluid-structure interface. We use Streamline Upwind Petrov-Galerkin, pressure-stabilizing Petrov-Galerkin (SUPG-PSPG) stabilization [31,86,105,116] for the Navier-Stokes equations, enabling the use of trilinear polynomials ($\mathbb{Q}_1$) for both fluid velocity and pressure, and the stabilization of the advection dominated regime. We use $\mathbb{Q}_1$ polynomials also for the structure displacement and for the fluid domain displacement. We do remark, however, that the extension to higher order polynomials is straightforward, provided that fluid velocity and solid displacement are discretized with the same finite element space.

After space and time discretization, the FSI problem is expressed as a system of nonlinear algebraic equations: for each $n=0,1,\ldots,N_t-1$,

$$
\begin{cases}
\mathbb{G}_f(\mathbf{G}^{n+1}) = \mathbf{0},\\[1mm]
\mathbb{S}_s(\mathbf{D}^{n+1}) = \mathbf{0},\\[1mm]
\mathbb{F}_f(\mathbf{U}^{n+1},\mathbf{G}^{n+1}) = \mathbf{0},\\[1mm]
\mathbb{C}_g(\mathbf{G}^{n+1},\mathbf{D}^{n+1}) = \mathbf{0},\\[1mm]
\mathbb{C}_k(\mathbf{U}^{n+1},\mathbf{D}^{n+1}) = \mathbf{0},\\[1mm]
\mathbb{C}_d(\mathbf{D}^{n+1},\mathbf{U}^{n+1},\mathbf{G}^{n+1}) = \mathbf{0}.
\end{cases}
\tag{3.2}
$$

In the former system, the vectors $\mathbf{G}, \mathbf{D}$ and $\mathbf{U}$ denote the algebraic numerical representations of the fluid domain displacement $\mathbf{d}_f$, structure displacement $\mathbf{d}$ and fluid domain variables (i.e. degrees of freedom associated to the velocity $\mathbf{u}$ and pressure p, collected in a single vector), respectively. The operators $\mathbb{G}_f$, $\mathbb{S}_s$ and $\mathbb{F}_f$ are the algebraic numerical representations of the differential operators defining the fluid domain displacement problem (3.1b), the structure problem (3.1a) and the fluid problem (3.1g) and (3.1h), each of them restricted to the degrees of freedom on the interior (not on the interface) of the respective subdomains. To keep our notation light, the dependence of the operators on solution variables at timesteps until t^n is understood. The operators $\mathbb{C}_g$, $\mathbb{C}_k$ and $\mathbb{C}_d$, are the algebraic numerical representations of the geometric (3.1c), kinematic (3.1i) and dynamic (3.1j) coupling conditions respectively. The explicit definition of each operator is given in Appendix A.

We remark that $\mathbb{G}_f$, $\mathbb{C}_g$ and $\mathbb{C}_k$ are affine, whereas the remaining operators are in general nonlinear. In particular, we remark that the dependence of $\mathbb{F}_f$ on $\mathbf{G}^{n+1}$ also accounts for the fact that the integrals involved in the weak formulation of the fluid problem are calculated on the deformed configuration (at time t^{n+1}) of the fluid domain.

4 Fluid-structure coupling schemes

The FSI problem is highly nonlinear, due to the nonlinearity of the fluid and structure problems (especially the latter) taken individually, and to the inherently nonlinear dependency of the fluid problem on the structure displacement through the fluid domain displacement. Suitable schemes need to be implemented to deal with the three subproblems (geometry, fluid, structure) and their coupling conditions.

In the context of biological tissues, the fluid and the structure have comparable densities, resulting in the so-called added-mass effect [27]. This typically yields stability issues: in particular, loosely coupled partitioned schemes (based on explicit time discretization of the whole FSI problem) may generate blowing-up solutions due to an incorrect energy balance [27]. Although in recent years stable loosely coupled partitioned schemes for hemodynamics were studied [18, 24, 25, 40, 49, 50, 52], here we focus on the two most traditional families of schemes that guarantee stability in hemodynamics:

- the *fully coupled partitioned schemes* [14, 15, 27, 29, 69, 76, 78, 99], in which at every timestep the fluid and structure subproblems are solved one independently of the other, applying suitable interface transmission conditions to each subproblem and iterating until fulfillment of the interface conditions (in the spirit of fixed-point iteration algorithms); for the sake of brevity we will refer to these schemes as partitioned, leaving it understood that we refer to fully coupled partitioned schemes.

- the *monolithic schemes* [20, 29, 45, 55, 64, 72, 80, 81, 93, 115], in which the nonlinear algebraic system arising from the discretization of the FSI problem is solved as a whole by means of Newton or inexact-Newton schemes.

In both the cases, the geometric coupling can be treated either implicitly or explicitly without hindering the time stability of the numerical scheme [78]. The explicit treatment of the geometric coupling, together with the explicit treatment of the advection term in Navier-Stokes equations, gives rise to schemes referred to as geometric-convective explicit [17, 29, 38, 77, 79].

We remark that the classes above are not exhaustive: alternative approaches involve the use of splitting schemes [38, 85] or the reduction of the FSI problem to an interface problem [32].

In the following sections, several schemes within the partitioned and monolithic families are described. In each case we will consider the fully discrete problem (3.2), and use the notation introduced in Section 3.

4.1 Partitioned (P) schemes

The appeal of partitioned schemes lies in their modularity, i.e. in the fact that they reduce to a sequence of independent calls to fluid dynamics and mechanics solvers. Therefore, one can leverage available solvers and advanced techniques for the individual subproblems, provided there exists a way to communicate interface data between them. On the other hand, these are iterative schemes, whose convergence properties are difficult to assess in the most general case, and depend heavily on the material properties and geometry of the domain [14, 27, 44].

4.1.1 Partitioned schemes with explicit geometric coupling (PE)

A fixed-point, partitioned scheme with explicit geometric coupling can be outlined as follows: for each time step $n = 1, 2, \dots, N_t$, given $\mathbf{G}^n$, $\mathbf{D}^n$ and $\mathbf{U}^n$, to obtain $\mathbf{G}^{n+1}$, $\mathbf{D}^{n+1}$ and $\mathbf{U}^{n+1}$:

1. Compute $\mathbf{G}^{n+1}$ by solving the fluid domain displacement problem, using the structure displacement at previous time step

$$\begin{cases} \mathbb{G}_{\mathrm{f}}(\mathbf{G}^{n+1}) = \mathbf{0}, \\ \mathbb{C}_{\mathrm{g}}(\mathbf{G}^{n+1}, \mathbf{D}^n) = \mathbf{0}, \end{cases}$$

and update the fluid domain according to the newly computed fluid domain displacement.

2. Set $\mathbf{D}^{n+1}_{(0)} = \mathbf{D}^n$ and iterate for $k = 0, 1, 2, \dots$ and until convergence:

   (a) Compute $\mathbf{U}^{n+1}_{(k+1)}$ by solving the fluid problem, using the structure displacement and traction at previous iteration to provide suitable boundary data at the interface

   $$\begin{cases} \mathbb{F}_{\mathrm{f}}(\mathbf{U}^{n+1}_{(k+1)}, \mathbf{G}^{n+1}) = \mathbf{0}, \\ \mathbb{C}_{\mathrm{f}}(\mathbf{D}^{n+1}_{(k)}, \mathbf{U}^{n+1}_{(k+1)}, \mathbf{G}^{n+1}) = \mathbf{0}. \end{cases}$$

(b) Compute $\widetilde{\mathbf{D}}_{(k+1)}^{n+1}$ by solving the mechanics problem, using the just computed fluid domain solution to provide suitable boundary data at the interface

$$\begin{cases} \mathbb{S}_s\left(\widetilde{\mathbf{D}}_{(k+1)}^{n+1}\right)=\mathbf{0}, \\ \mathbb{C}_s\left(\widetilde{\mathbf{D}}_{(k+1)}^{n+1},\mathbf{U}_{(k+1)}^{n+1},\mathbf{G}^{n+1}\right)=\mathbf{0}. \end{cases}$$

(c) Apply relaxation or acceleration to the structure displacement, by setting

$$\mathbf{D}_{(k+1)}^{n+1}=\mathbb{R}_{(k+1)}\left(\widetilde{\mathbf{D}}_{(k+1)}^{n+1},\mathbf{D}_{(k)}^{n+1},\mathbf{D}_{(k-1)}^{n+1},\cdots\right),$$

where $\mathbb{R}_{(k+1)}$ is a suitable relaxation or convergence acceleration operator, which can be either a relaxation with constant coefficient, Aitken acceleration or Anderson acceleration; more details are given in Appendix B.

Iterations are stopped when the norm of the residual associated to interface conditions falls below a prescribed tolerance, as described in [14].

The scheme is depicted in Fig. 2(a). The operators $\mathbb{C}_f$ and $\mathbb{C}_s$ represent suitable boundary conditions at the interface for each problem. Different algorithms are obtained by different choices of such operators. Two relevant options are the following:

- Dirichlet-Neumann (DN) scheme [14, 27]

$$\mathbb{C}_f\left(\mathbf{D}_{(k)}^{n+1},\mathbf{U}_{(k+1)}^{n+1},\mathbf{G}^{n+1}\right)=\mathbb{C}_k\left(\mathbf{U}_{(k+1)}^{n+1},\mathbf{D}_{(k)}^{n+1}\right),$$
$$\mathbb{C}_s\left(\widetilde{\mathbf{D}}_{(k+1)}^{n+1},\mathbf{U}_{(k+1)}^{n+1},\mathbf{G}^{n+1}\right)=\mathbb{C}_d\left(\widetilde{\mathbf{D}}_{(k+1)}^{n+1},\mathbf{U}_{(k+1)}^{n+1},\mathbf{G}^{n+1}\right),$$

resulting in the kinematic coupling condition being applied to the fluid and the dynamic coupling condition being applied to the structure.

- Robin-Neumann (RN) scheme [14, 15]

$$\mathbb{C}_f\left(\mathbf{D}_{(k)}^{n+1},\mathbf{U}_{(k+1)}^{n+1},\mathbf{G}^{n+1}\right)=\alpha_f\mathbb{C}_k\left(\mathbf{U}_{(k+1)}^{n+1},\mathbf{D}_{(k)}^{n+1}\right)+\mathbb{C}_d\left(\mathbf{D}_{(k)}^{n+1},\mathbf{U}_{(k+1)}^{n+1},\mathbf{G}^{n+1}\right),$$
$$\mathbb{C}_s\left(\widetilde{\mathbf{D}}_{(k+1)}^{n+1},\mathbf{U}_{(k+1)}^{n+1},\mathbf{G}^{n+1}\right)=\mathbb{C}_d\left(\widetilde{\mathbf{D}}_{(k+1)}^{n+1},\mathbf{U}_{(k+1)}^{n+1},\mathbf{G}^{n+1}\right),$$

where $\alpha_f>0$ is a suitable parameter chosen to improve convergence [44].

Other schemes can be obtained in a similar way (e.g. Robin-Robin schemes, [10,14,15, 48]), by combining the kinematic and dynamic coupling conditions. We mainly consider these two because on the one hand the DN scheme is extremely simple to formulate and implement, on the other hand the RN scheme has been shown to perform particularly well in comparison to other fixed-point schemes, most notably DN [14,15]. In any case, such schemes guarantee the fulfillment, up to a prescribed tolerance, of the kinematic and dynamic conditions.

We remark that the structure subproblem is in general nonlinear, and it must be solved e.g. by means of Newton's method at each iteration [86].
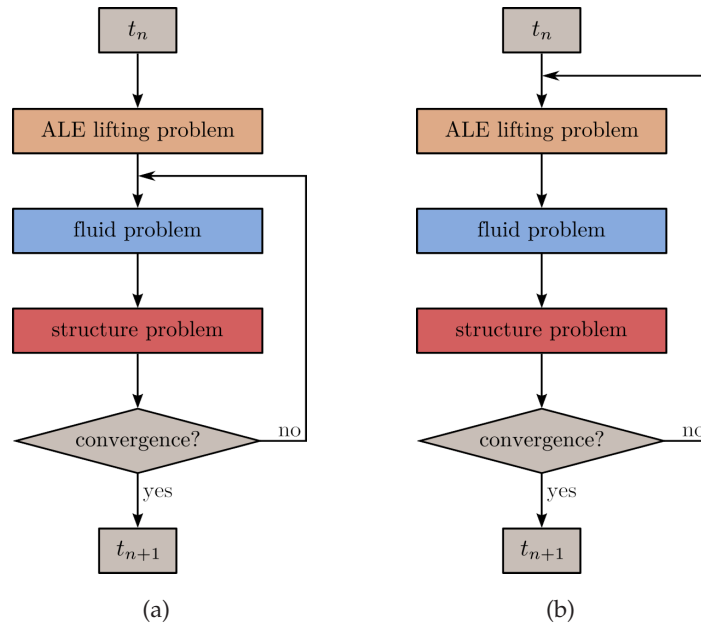
Figure 2: Partitioned schemes, with explicit geometric coupling (a) and implicit geometric coupling (b): advancement from one timestep to the following one is performed by iterating over the subproblems until convergence.

4.1.2 Partitioned schemes with implicit geometric coupling (PI)

A fixed-point scheme with implicit geometric coupling can be obtained by iterating also over the geometric interface condition, solving (2.3) within the same loop used to achieve the convergence of both the kinematic and dynamic conditions. For each $n = 1, 2, \ldots, N_t$, given $\mathbf{G}^n$, $\mathbf{D}^n$ and $\mathbf{U}^n$, in order to compute $\mathbf{G}^{n+1}$, $\mathbf{D}^{n+1}$ and $\mathbf{U}^{n+1}$, we set $\mathbf{D}^{n+1}_{(0)} = \mathbf{D}^n$, and iterate for $k = 0, 1, 2, \ldots$ until convergence:

1. Compute $\mathbf{G}^{n+1}_{(k+1)}$ by solving the fluid domain displacement problem, using the structure displacement at previous iteration

$$
\begin{cases}
\mathbb{G}_f\big(\mathbf{G}^{n+1}_{(k+1)}\big) = \mathbf{0}, \\
\mathbb{C}_g\big(\mathbf{G}^{n+1}_{(k+1)}, \mathbf{D}^{n+1}_{(k)}\big) = \mathbf{0},
\end{cases}
$$

and update the fluid domain according to the newly computed fluid domain displacement.

2. Compute $\mathbf{U}^{n+1}_{(k+1)}$ by solving the fluid problem, using the newly computed fluid domain and the structure displacement at previous iteration to provide suitable boundary data at the interface

$$
\begin{cases}
\mathbb{F}_f\big(\mathbf{U}^{n+1}_{(k+1)}, \mathbf{G}^{n+1}_{(k+1)}\big) = \mathbf{0}, \\
\mathbb{C}_f\big(\mathbf{D}^{n+1}_{(k)}, \mathbf{U}^{n+1}_{(k+1)}, \mathbf{G}^{n+1}_{(k+1)}\big) = \mathbf{0}.
\end{cases}
$$

3. Compute $\widetilde{\mathbf{D}}^{n+1}_{(k+1)}$ by solving the structure problem, using the just computed fluid domain solution to provide suitable boundary data at the interface

$$\begin{cases} \mathbb{S}_s\big(\widetilde{\mathbf{D}}^{n+1}_{(k+1)}\big)=\mathbf{0}, \\ \mathbb{C}_s\big(\widetilde{\mathbf{D}}^{n+1}_{(k+1)},\mathbf{U}^{n+1}_{(k+1)},\mathbf{G}^{n+1}_{(k+1)}\big)=\mathbf{0}. \end{cases}$$

4. Apply relaxation or acceleration to the structure displacement, by setting

$$\mathbf{D}^{n+1}_{(k+1)}=\mathbb{R}_{(k+1)}\Big(\widetilde{\mathbf{D}}^{n+1}_{(k+1)},\mathbf{D}^{n+1}_{(k)},\mathbf{D}^{n+1}_{(k-1)},\cdots\Big).$$

We use the same stopping criterion as for PE schemes.

This type of scheme is depicted in Fig. 2(b). As before, different schemes are obtained by making different choices for the interface conditions $\mathbb{C}_f$ and $\mathbb{C}_s$.

We remark that other types of schemes with implicit geometric coupling can be obtained e.g. by performing two nested loops at each timestep: in the outer loop, the geometry problem is solved and the fluid domain updated, and in the inner loop the fluid and structure problems are solved for a fixed fluid domain [78]. Moreover, an intermediate approach in this context is obtained by performing a small, fixed number of iterations on the outer loop for the domain displacement [78].

4.2 Monolithic (M) schemes

Now, we consider schemes based on solving in a single shot the algebraic nonlinear system arising from the time and space discretization of the FSI problem. We employ the Newton scheme for the linearization of such a system. With this purpose, let us rewrite (3.2) in the following more compact form:

$$\begin{cases} \mathbb{G}(\mathbf{G}^{n+1},\mathbf{D}^{n+1})=\mathbf{0}, \\ \mathbb{F}(\mathbf{G}^{n+1},\mathbf{U}^{n+1},\mathbf{D}^{n+1})=\mathbf{0}, \quad n=0,1,2,\ldots,N_t, \\ \mathbb{S}(\mathbf{G}^{n+1},\mathbf{U}^{n+1},\mathbf{D}^{n+1})=\mathbf{0}, \end{cases} \tag{4.1}$$

where $\mathbb{G}$ includes both $\mathbb{G}_f$ and the boundary conditions for the fluid domain displacement, $\mathbb{F}$ includes $\mathbb{F}_f$ and $\mathbb{C}_k$ and $\mathbb{S}$ includes $\mathbb{S}_s$ and $\mathbb{C}_d$ (see Appendix A for the explicit definition of the operators).

4.2.1 Monolithic scheme with explicit geometric coupling (ME)

Given $\mathbf{G}^n$, $\mathbf{D}^n$ and $\mathbf{U}^n$, to compute the solution at timestep $n+1$:

1. Compute $\mathbf{G}^{n+1}$ by solving the fluid domain displacement problem, using the structure displacement at previous time step

$$\begin{cases} \mathbb{G}_f(\mathbf{G}^{n+1})=\mathbf{0}, \\ \mathbb{C}_g(\mathbf{G}^{n+1},\mathbf{D}^n)=\mathbf{0}, \end{cases}$$

and update the fluid domain according to the newly computed fluid domain displacement.

2. Solve the fluid-structure coupled problem with a Newton loop: set $\mathbf{U}^{n+1}_{(0)} = \mathbf{U}^n$ and $\mathbf{D}^{n+1}_{(0)} = \mathbf{D}^n$, then for $k = 0,1,2,\dots$ and until convergence,

$$J_{\text{FS}}\left(\mathbf{G}^{n+1}, \mathbf{U}^{n+1}_{(k)}, \mathbf{D}^{n+1}_{(k)}\right) \begin{bmatrix} \delta\mathbf{U} \\ \delta\mathbf{D} \end{bmatrix} = \mathbf{R}\left(\mathbf{G}^{n+1}, \mathbf{U}^{n+1}_{(k)}, \mathbf{D}^{n+1}_{(k)}\right), \qquad (4.2)$$

$$\mathbf{U}^{n+1}_{(k+1)} = \mathbf{U}^{n+1}_{(k)} - \delta\mathbf{U},$$

$$\mathbf{D}^{n+1}_{(k+1)} = \mathbf{D}^{n+1}_{(k)} - \delta\mathbf{D},$$

where

$$J_{\text{FS}} = \begin{bmatrix} \frac{d\mathbb{F}}{d\mathbf{U}^{n+1}} & \frac{d\mathbb{F}}{d\mathbf{D}^{n+1}} \\ \frac{d\mathbb{S}}{d\mathbf{U}^{n+1}} & \frac{d\mathbb{S}}{d\mathbf{D}^{n+1}} \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} \mathbb{F} \\ \mathbb{S} \end{bmatrix}.$$

We remark that in the computation of $\mathbf{R}$ and J_{FS} the last available fluid domain displacement, $\mathbf{G}^{n+1}$ is used.

4.2.2 Monolithic scheme with implicit geometric coupling (MI)

To obtain a scheme with implicit geometric coupling, it is possible to update the fluid domain displacement within the Newton loop, resulting in a scheme in which the fluid domain displacement is treated implicitly. Given $\mathbf{G}^n$, $\mathbf{D}^n$ and $\mathbf{U}^n$, to compute the solution at timestep $n+1$ set $\mathbf{G}^{n+1}_{(0)} = \mathbf{G}^n$, $\mathbf{U}^{n+1}_{(0)} = \mathbf{U}^n$ and $\mathbf{D}^{n+1}_{(0)} = \mathbf{D}^n$, then iterate for $k = 0,1,2,\dots$ until convergence:

1. Compute $\mathbf{G}^{n+1}_{(k+1)}$ by solving the fluid domain displacement problem, using the structure displacement at previous iteration

$$\begin{cases} \mathbb{G}_{\text{f}}\left(\mathbf{G}^{n+1}_{(k+1)}\right) = \mathbf{0}, \\ \mathbb{C}_{\text{g}}\left(\mathbf{G}^{n+1}_{(k+1)}, \mathbf{D}^{n+1}_{(k)}\right) = \mathbf{0}, \end{cases}$$

and update the fluid domain according to the newly computed fluid domain displacement.

2. Compute $\mathbf{U}^{n+1}_{(k+1)}$ and $\mathbf{D}^{n+1}_{(k+1)}$ with a Newton step

$$J_{\text{FS}}\left(\mathbf{G}^{n+1}_{(k+1)}, \mathbf{U}^{n+1}_{(k)}, \mathbf{D}^{n+1}_{(k)}\right) \begin{bmatrix} \delta\mathbf{U} \\ \delta\mathbf{D} \end{bmatrix} = \mathbf{R}\left(\mathbf{G}^{n+1}_{(k+1)}, \mathbf{U}^{n+1}_{(k)}, \mathbf{D}^{n+1}_{(k)}\right),$$

$$\mathbf{U}^{n+1}_{(k+1)} = \mathbf{U}^{n+1}_{(k)} - \delta\mathbf{U},$$

$$\mathbf{D}^{n+1}_{(k+1)} = \mathbf{D}^{n+1}_{(k)} - \delta\mathbf{D}.$$

This corresponds to solving the fully coupled system (4.1) with an inexact-Newton approach, in which the Jacobian terms involving derivatives with respect to the fluid domain displacement (i.e. the shape derivatives [19, 39]) are neglected. Accounting for shape derivatives, either computed exactly [20, 35, 39, 64, 93, 94, 114] or approximated through finite differences [55, 60], is another viable option.

4.3 Solution of linear systems and preconditioning

All the numerical schemes presented above involve, in particular, the numerical solution of the ALE lifting problem. For that, we use the conjugate gradient (CG) method [96], preconditioned with algebraic multigrid (AMG) [54]. For all other linear systems we rely on the GMRES method.

When employing partitioned schemes, the linear systems arising from the fluid and solid problem are solved independently. For the fluid problem, we make use of an approximate SIMPLE preconditioner [34], which can be interpreted as an inexact block-LU factorization of the fluid matrix. The inverses of fluid velocity and pressure Schur complement blocks within SIMPLE are approximated with AMG. For the solid problem, we use instead an AMG preconditioner.

In both the ME and the MI schemes, the discretization of the problem leads to a linear system with fluid and structure unknowns, with the general block structure

$$J = \begin{bmatrix} J_{\mathrm{ff}} & J_{\mathrm{fs}} \\ J_{\mathrm{sf}} & J_{\mathrm{ss}} \end{bmatrix}. \tag{4.3}$$

As typical in the FSI literature [16, 29, 33, 55, 71], we derive a preconditioner by exploiting the block structure of the matrix J, falling back on inner black box preconditioners for the approximation of the individual blocks. Typical choices of inner preconditioners are AMG [55, 71] and geometric multigrid [60, 93] preconditioners. Our chosen preconditioner for (4.3) is a block lower triangular preconditioner

$$P = \begin{bmatrix} J_{\mathrm{ff}} & 0 \\ J_{\mathrm{sf}} & J_{\mathrm{ss}} \end{bmatrix}.$$

The inverse of P is

$$P^{-1} = \begin{bmatrix} J_{\mathrm{ff}}^{-1} & 0 \\ -J_{\mathrm{ss}}^{-1} J_{\mathrm{sf}} J_{\mathrm{ff}}^{-1} & J_{\mathrm{ss}}^{-1} \end{bmatrix}.$$

The inverse of fluid and structure blocks, required by the application of P^{-1}, is approximated by means of suitable inner preconditioners. For the fluid block (both velocity and pressure variables), we make use of the approximate SIMPLE preconditioner P_{SIMPLE}, which in turn falls back onto AMG for the approximation of the fluid velocity block and of the fluid pressure Schur complement. For the structure block, we employ an AMG

preconditioner P_s, whence

$$P_{\text{FSI}}^{-1} = \begin{bmatrix} P_{\text{SIMPLE}}^{-1} & 0 \\ -P_s^{-1} J_{\text{sf}} P_{\text{SIMPLE}}^{-1} & P_s^{-1} \end{bmatrix}.$$

The application of P_{FSI}^{-1} can be expressed in terms of one application of the SIMPLE preconditioner, one application of the AMG preconditioner for the structure, and a few matrix-vector multiplications.

Other preconditioners can be derived similarly, e.g. by resorting to a block LU or block LDU factorization of matrix (4.3) [33, 55, 71] or using a block upper triangular or block diagonal preconditioners. Among these options, we found the block lower triangular preconditioner to perform slightly better than the others in terms of overall computational cost. Alternative approaches to preconditioning can be derived starting from domain decomposition formulations, mimicking at the preconditioner level the behavior of partitioned schemes [15, 16].

For AMG preconditioners we rely on the Trilinos ML package [42].

4.4 Solver parallelization

We implemented the presented solvers in life^x [3, 4], a C++ library developed in-house, tailored to cardiac applications, based on the finite element library deal.II [2, 8, 9]. All solvers are parallel so as to exploit high-performance computing facilities, allowing for large scale simulations. As typical in finite element solvers, computational domains are partitioned across multiple processes. We do so using the mesh handling facilities exposed by deal.II [9], which fall back onto the p4est library [26]. Parallelization acts at three different levels: the assembly of linear systems, their solution, and the communication of interface data.

During the assembly phase, each process only loops over the elements of its part of the computational domain, and communication takes place at the end of the element loop to distribute to each process the matrix and vector entries it needs for the system solution. For the parallel solution of linear systems, we rely on the Trilinos library [107].

Interface data communication for partitioned solvers requires point-to-point communication between processes: indeed, the fluid and structure meshes, although conforming, are distributed across processes in an independent way, and the process that owns an interface mesh node on the fluid domain might be different from the one that owns it on the structure domain. However, the point-to-point communication pattern for interface data is precomputed at simulation startup, so that this does not yield a significant overhead.

Finally, we remark that the different subproblems of partitioned schemes are solved sequentially, while each individual subproblem is solved in parallel over multiple cores.

5   Numerical results

We present numerical results on FSI test cases. We aim at comparing the performance of the coupling schemes presented, in terms of overall computational costs and, for partitioned schemes, number of iterations required for convergence. We will consider idealized problems in the context of cardiovascular haemodynamics. In all cases we will assume the solid to have density $\rho_s = 1000 \text{kg/m}^3$, and the fluid to have density $\rho_f = 1060 \text{kg/m}^3$ and viscosity $\mu = 3.5 \cdot 10^{-3} \text{Pa} \cdot \text{s}$.

For each test, we consider the ME (Section 4.2.1) and MI (Section 4.2.2) schemes, as well as DN and RN partitioned schemes with static relaxation (B.1), Aitken acceleration (B.2) and Anderson acceleration. We sum up in Table 2 the considered schemes and the associated abbreviations.

Unless otherwise specified, the simulations ran on 48 cores from CINECA GALILEO100 [1].

We consider an idealized left ventricle shaped as a prolate ellipsoid. The corresponding fluid and structure computational domains are represented in Fig. 3, left and right respectively. On the top surface of the fluid domain, we identify two intersecting circular regions Γ_{AV}, Γ_{MV} that represent in this idealized context the aortic and mitral valve orifice, respectively, similarly to the test case considered in [103] for a computational fluid dynamics (CFD) ventricular simulation.

Table 2: Legend of the abbreviations used for the coupling schemes in numerical experiments.

Abbreviation	Meaning
PE	Partitioned, explicit geometry (Section 4.1.1)
PI	Partitioned, implicit geometry (Section 4.1.2)
ME	Monolithic, explicit geometry (Section 4.2.1)
MI	Monolithic, implicit geometry (Section 4.2.2)
DN	Dirichlet-Neumann partitioned scheme
RN	Robin-Neumann partitioned scheme
SR	Partitioned schemes with static relaxation (Appendix B)
AitA	Partitioned schemes with Aitken acceleration (Appendix B)
AndA	Partitioned schemes with Anderson acceleration (Appendix B)

5.1   Test A: idealized left ventricle in systole

In this test case, we reproduce the systolic phase during which the ventricle contracts and blood is ejected through the aortic valve. Therefore, we impose the no-slip condition $\mathbf{u} = \mathbf{0}$ on $\Gamma_{MV} \backslash \Gamma_{AV}$ and on $\Gamma_{f,b}$, and a resistance boundary condition on Γ_{AV} [45, 103]

$$\sigma_f \mathbf{n} = -\left(p_0 + R \int_{\Gamma_{AV}} \mathbf{u} \cdot \mathbf{n} d\Gamma \right) \mathbf{n} \quad \text{on} \quad \Gamma_{AV}, \tag{5.1}$$
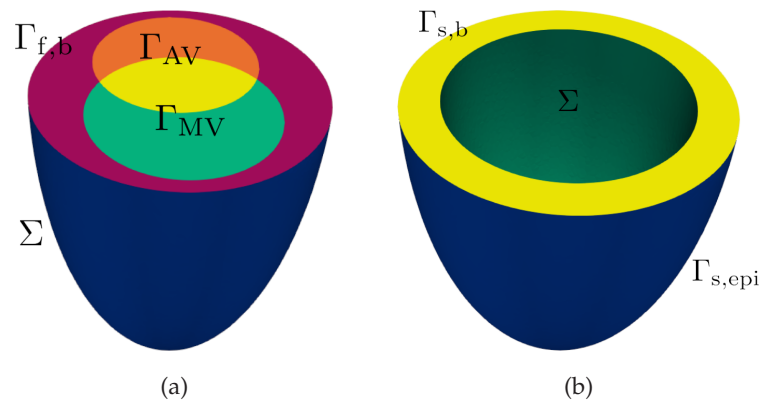
Figure 3: Fluid (a) and structure (b) meshes for the prolate ellipsoid (tests A, B and C)

.

where $p_0 = 6000$Pa represents the minimum aortic pressure and $R = 1 \cdot 10^7$ kg/(s·m^4) is a resistance parameter, calibrated in accordance with [89]. We impose a no-slip condition $\mathbf{u} = \mathbf{0}$ on $\Gamma_{f,b}$ as well.

The material model of the solid is the Guccione law (2.2). As boundary conditions for the structure problem, we use a homogeneous Dirichlet condition $\mathbf{d} = \mathbf{0}$ on the base $\Gamma_{s,b}$, and consistently keep the fluid domain base fixed, that is we impose $\mathbf{d}_f = \mathbf{0}$ on $\Gamma_{f,b} \cup \Gamma_{AV} \cup \Gamma_{MV}$. We remark that this condition is not physiological, but in this simplified test case avoids issues related to having moving inlet or outlet sections for the fluid. In a more realistic case, boundary conditions allowing for the displacement of the structure base should be employed, such as Robin boundary conditions [45] or more sophisticated conditions modeling the presence of the neglected part of the ventricle [90,91]. Finally, on the epicardium $\Gamma_{s,epi}$ of the ventricle, we impose generalized Robin boundary conditions that mimic the presence of the pericardium, a sac containing the heart [45,83,90,101]

$$P(\mathbf{d})\mathbf{n} = -(\mathbf{n} \otimes \mathbf{n})\left(K_\perp \mathbf{d} + C_\perp \frac{\partial \mathbf{d}}{\partial t}\right) - (I - \mathbf{n} \otimes \mathbf{n})\left(K_\parallel \mathbf{d} + C_\parallel \frac{\partial \mathbf{d}}{\partial t}\right), \tag{5.2}$$

in which $K_\perp = 2 \cdot 10^5$ Pa/m, $K_\parallel = 2 \cdot 10^4$ Pa/m are constant elasticity coefficients for the directions normal and parallel to the boundary respectively, and $C_\perp = 2 \cdot 10^4$ Pa·s/m and $C_\parallel = 2 \cdot 10^3$ Pa·s/m are constant viscosity coefficients for the directions normal and parallel to the boundary, tuned in accordance with [92].

We discretize the fluid domain with a mesh composed of 20980 hexahedral elements, for a total of 100412 degrees of freedom. The solid domain is discretized with a mesh of 24712 hexahedral elements, with 94485 degrees of freedom in total. This corresponds to an average mesh size of $h = 4$mm on both the fluid and solid domains.

We run two tests in this setting, prescribing active contraction in the active stress formulation in one case (Test A1) and in the active strain formulation in the other (Test A2). The active contraction of the myocardium is achieved by defining the active stress or
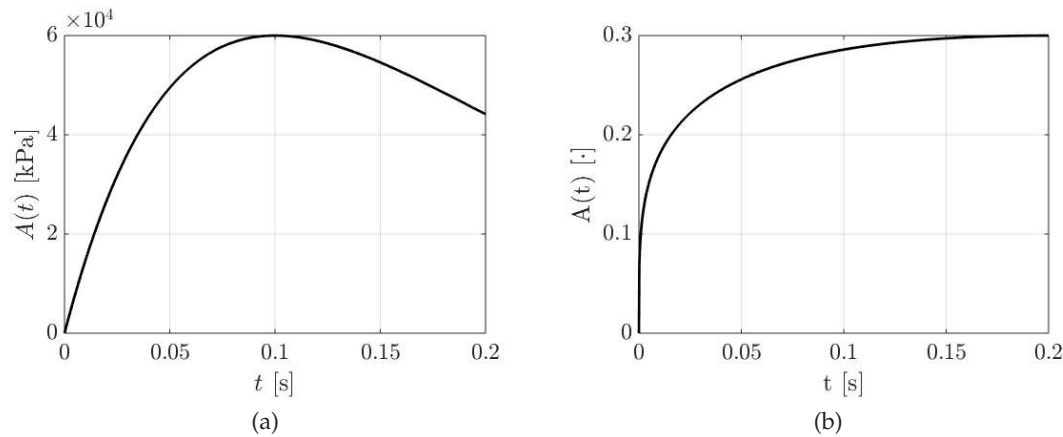
Figure 4: Analytical activation functions for the active stress (a) and active strain (b) test cases.

strain to be constant in space and variable in time according to a prescribed analytical function. Similarly to what done in [81], we choose a function of the form

$$
A(t) = \begin{cases} 0, & t < t_0, \\ A_{\max} \left(\dfrac{t - t_0}{T_{\text{peak}}} \right)^p \exp \left\{ p \left(1 - \dfrac{t - t_0}{T_{\text{peak}}} \right) \right\}, & t \geq t_0, \end{cases}
\tag{5.3}
$$

where t_0 is the time at which contraction starts, T_{peak} is the time of activation peak and p is a positive parameter. The graph of $A(t)$ is shown in Fig. 4. For the active stress test case, we choose $t_0 = 0$s, $T_{\text{peak}} = 0.1$s, $A_{\max} = 60$kPa and $p = 1$. For the active strain test case we use $t_0 = 0$s, $T_{\text{peak}} = 0.2$s, $A_{\max} = 0.3$ and $p = 0.25$. These values do not lead to physiological deformations of the idealized ventricle, but are chosen so that the resulting velocities, both of the myocardium and of the blood, have the same orders of magnitude that can be expected in a physiological simulation.

Due to the nonlinearity and anisotropy of the constitutive law of the solid structure, optimal relaxation and acceleration parameters are not available. They were tuned so as to obtain convergence, although it is worth observing that tuning parameters is itself a major issue with these schemes. Refer to Appendix B for the chosen values of acceleration and relaxation parameters. The same argument holds true for the α_f coefficient in RN schemes as well. Although optimal values of the parameter α_f have been studied for spherical geometries and linear elasticity in [47], their application here is not trivial due to the non-linearity of the structure. We selected $\alpha_f = 5000 \text{kg}/(\text{m}^2 \cdot \text{s})$ as the value that experimentally provided the best performance for our setup. More investigations are needed on this point.

For these tests, we choose a final time of $T = 0.2$s, and a time step $\Delta t = 2 \cdot 10^{-4}$s.

The solution at a few time instants, computed using the ME solver, is reported in Figs. 5 and 6, and the computational times associated to the different coupling schemes are reported in Tables 3 and 4.

Table 3: Performance of the different schemes for the systole test case with active stress formulation (Test A1). For each scheme we report: the total wall time required for the solution of the test case, as well as the average wall time per time step, the total time spent assembling linear systems and the time spent solving them; the minimum, average and maximum of the number of iterations per timestep required for convergence of the Newton method, for monolithic schemes, and of the fixed-point (FP) method, for partitioned schemes; the average number of fluid and solid assembly calls per timestep; the average number of linear solver iterations per timestep (across all Newton or partitioned iterations). Refer to Table 2 for the explanation of the abbreviations used.

Scheme	Wall time [s] (relative to best)				Newton/FP iter.			Avg. assembly calls		Avg. GMRES iter.		
	total	avg.	assembly	linear sol.	min.	avg.	max.	fluid	solid	monol.	fluid	solid
ME	6850 (1.00)	6.85	4270 (1.00)	1640 (1.0)	2	2.1	3	3.1	3.1	165.9	-	-
PE-DN-SR	Wall time > 24 h											
PE-DN-AitA	convergence failure											
PE-DN-AndA	convergence failure											
PE-RN-SR	Wall time > 24 h											
PE-RN-AitA	60120 (8.77)	60.1	28300 (6.63)	6443 (3.93)	13	15.6	17	15.6	34.1	-	808.6	643.4
PE-RN-AndA	54090 (7.89)	54.1	25500 (5.97)	5810 (3.54)	13	14.0	15	14.0	30.7	-	657.8	616.4
MI	16500 (1.00)	16.5	4262 (1.00)	3820 (1.00)	4	6.4	7	7.4	7.4	396.2	-	-
PI-DN-SR	Wall time > 24 h											
PI-DN-AitA	convergence failure											
PI-DN-AndA	convergence failure											
PI-RN-SR	Wall time > 24 h											
PI-RN-AitA	74890 (4.53)	74.9	34770 (8.16)	7368 (1.92)	14	15.9	18	15.9	34.8	-	827.8	654.9
PI-RN-AndA	58180 (3.52)	58.2	26980 (6.33)	6028 (1.57)	13	14.2	15	14.2	31.3	-	751.6	582.7

Table 4: Performance of the different schemes for the systole test case with active strain formulation (Test A2). For each scheme we report: the total wall time required for the solution of the test case, as well as the average wall time per time step, the total time spent assembling linear systems and the time spent solving them; the minimum, average and maximum of the number of iterations per timestep required for convergence of the Newton method, for monolithic schemes, and of the fixed-point (FP) method, for partitioned schemes; the average number of fluid and solid assembly calls per timestep; the average number of linear solver iterations per timestep (across all Newton or partitioned iterations). Refer to Table 2 for the explanation of the abbreviations used.

Scheme	Wall time [s] (relative to best)				Newton/FP iter.			Avg. assembly calls		Avg. GMRES iter.		
	total	avg.	assembly	linear sol.	min.	avg.	max.	fluid	solid	monol.	fluid	solid
ME	7230 (1.00)	7.23	4090 (1.00)	2100 (1.00)	2	2.0	3	3.0	3.0	201.5	-	-
PE-DN-SR	Wall time > 24 h											
PE-DN-AitA	convergence failure											
PE-DN-AndA	convergence failure											
PE-RN-SR	79110 (10.9)	79.1	36090 (8.82)	8855 (4.21)	19	20.5	25	20.5	43.5	-	951.1	1115.6
PE-RN-AitA	46920 (6.49)	46.9	21300 (5.21)	5628 (2.68)	11	12.1	21	12.1	26.4	-	612.5	685.7
PE-RN-AndA	46380 (6.41)	46.4	21000 (5.13)	5592 (2.66)	11	11.8	14	11.8	25.2	-	607.2	677.5
MI	17100 (1.00)	17.1	8790 (1.00)	5010 (1.00)	4	5.5	7	6.5	6.5	428.6	-	-
PI-DN-SR	Wall time > 24 h											
PI-DN-AitA	convergence failure											
PI-DN-AndA	convergence failure											
PI-RN-SR	79340 (4.63)	79.3	36260 (4.13)	9044 (1.81)	19	20.4	24	20.4	44.2	-	964.1	1130.1
PI-RN-AitA	50020 (2.93)	50.0	22400 (2.55)	5878 (1.17)	11	12.5	22	12.5	27.3	-	633.7	717.6
PI-RN-AndA	48760 (2.85)	48.8	21630 (2.46)	5767 (1.15)	11	12.3	14	12.3	26.0	-	632.3	706.2

(a) $t = 0$ms   (b) $t = 100$ms   (c) $t = 200$ms

velocity magnitude [m/s]

0   0.2   0.4   0.6   0.8   1

pressure [mmHg]
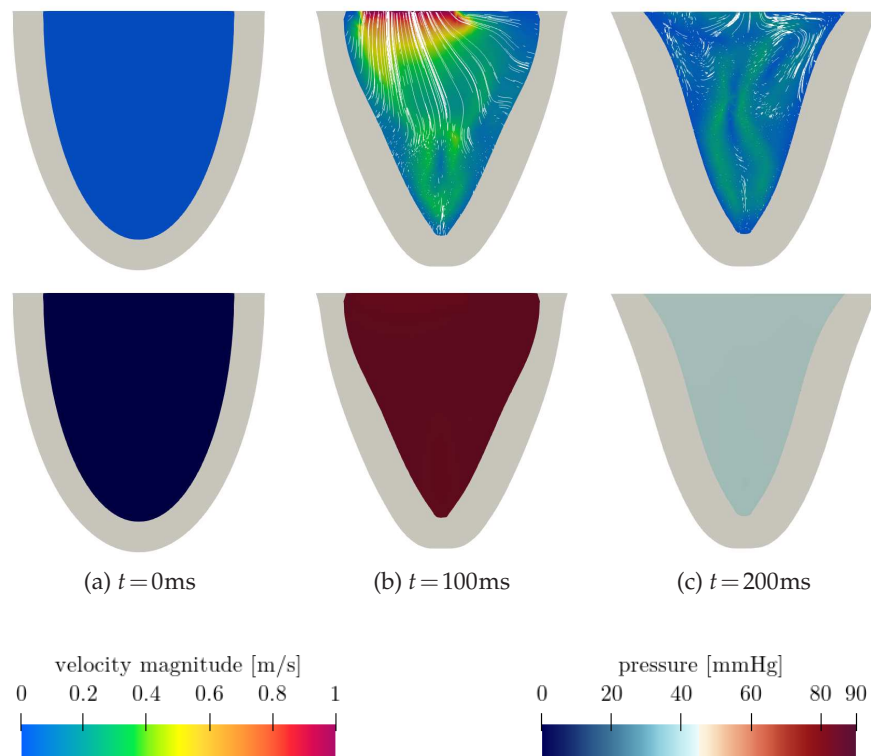
0   20   40   60   80   90

Figure 5: Snapshots of the fluid velocity (top) and pressure (bottom) of the idealized ventricle test case, in systole with the active stress formulation (Test A1).

Partitioned schemes with static relaxation require more time than 24 h to converge, except for RN in the active strain test case. The DN scheme fails to converge with all acceleration schemes, for explicit and implicit geometric coupling, and in both the active stress and active strain cases. Conversely, the RN scheme converges successfully when using Aitken and Anderson acceleration, with Anderson acceleration slightly improving over Aitken acceleration. In all cases considered the monolithic approach was several times faster than any of the partitioned schemes. The MI scheme is more costly than its explicit counterpart, while no significant increase in number of iterations is observed for RN schemes with implicit geometric coupling. No relevant difference is observed between the active stress and active strain approaches, in terms of relative efficiency of the monolithic and partitioned schemes.

5.2 Performance when varying the number of cores

We compare the performance of the ME solver with our best partitioned solver from previous section, namely the PE-RN-AndA solver, when the number of cores used in the computation varies. We consider the systolic setup of Test A1 (Section 5.1), but using
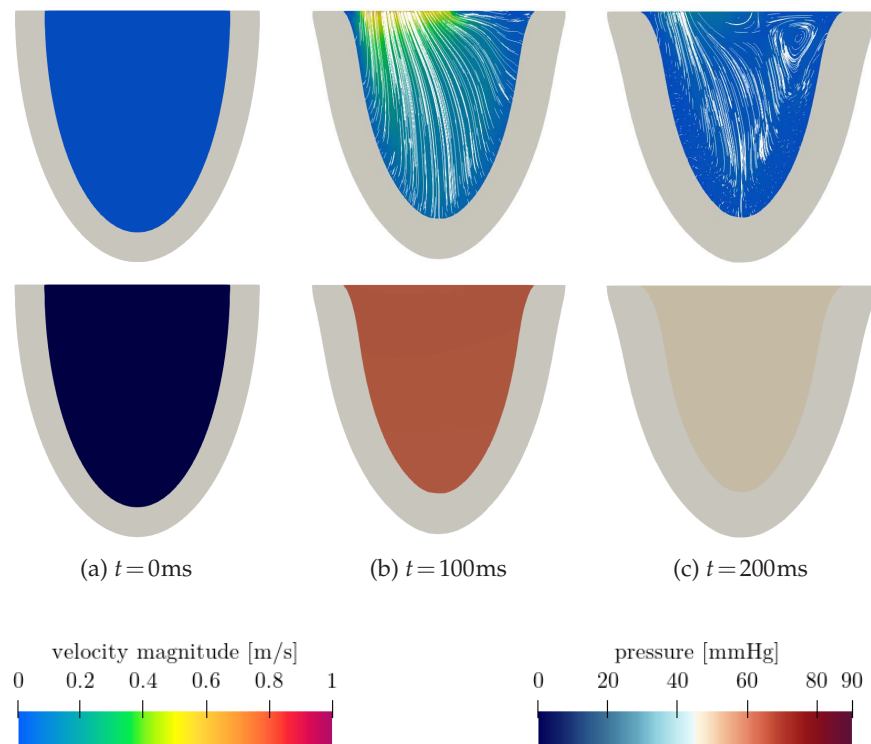
Figure 6: Snapshots of the fluid velocity (top) and pressure (bottom) of the idealized ventricle test case, in systole with the active strain formulation (Test A2).

a mesh with a higher resolution (57080 elements and 264692 degrees of freedom for the fluid, 61296 elements and 227019 degrees of freedom for the structure) and set the final time to $T = 0.01$ s. Fig. 7 reports the wall times needed for the whole simulation, as well as the ones spent during the assembly phase and during calls to the linear solvers varying the number of cores. Results suggest that, regardless of the number of processes used, the ME scheme is significantly faster. Indeed, for all the tests performed, the PE-RN-AndA scheme completed the simulation in approximately 5.5 times longer wall times.

5.3   Test B: idealized left ventricle in diastole

In this case we use the same geometry and the same mesh as described for Test A in Section 5.1. Boundary conditions on the structure and on the fluid domain are also the same. For the fluid problem, we impose an inlet condition through Γ_{MV}, to reproduce the diastolic filling of the ventricle. In particular, we set $\sigma_{\mathrm{f}}\mathbf{n} = -p_{\mathrm{MV}}\mathbf{n}$ on Γ_{MV}, with $p_{\mathrm{MV}} = 1333$ Pa. Conversely, we set $\mathbf{u} = \mathbf{0}$ on $\Gamma_{\mathrm{AV}} \backslash \Gamma_{\mathrm{MV}}$ and on $\Gamma_{\mathrm{f,b}}$. Since during diastole the ventricle is relaxing, we do not prescribe any activation, neither in the active stress nor in the active strain approach (i.e. the stress tensor in (2.1) only includes the passive contribution for the present test).
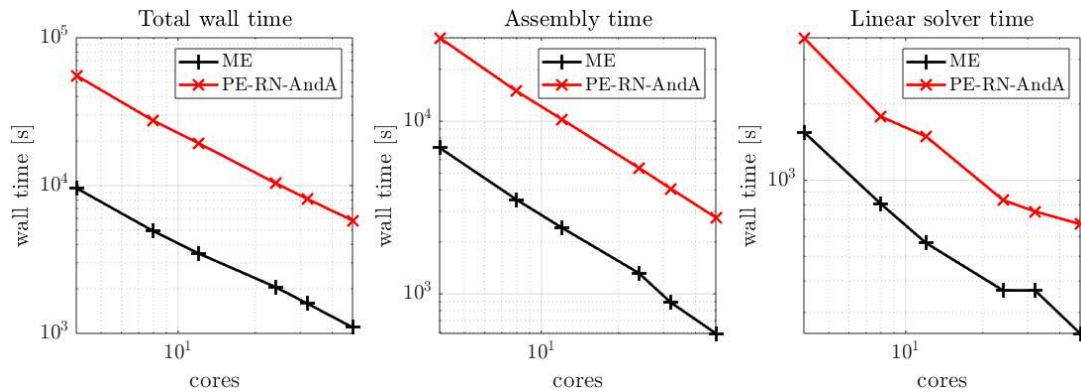
Figure 7: Comparison of ME and PE-RN-AndA schemes with varying number of cores. From left to right, we report the total wall time, the time spent for assembly and the time spent for solving linear systems. Refer to Table 2 for an explanation of the abbreviations used.

We remark that the initial conditions of this test case are not consistent with end-systolic conditions as found at the end of the previous test case. The aim of this test is to assess the performance of the algorithms in the context of a slow inflow and in absence of the active contraction, rather than providing physiologically meaningful insight.

For this test, we choose a final time of $T = 0.2$s and a time step of $\Delta t = 2 \cdot 10^{-4}$s.

A few snapshots of the solution, computed with the ME solver, are reported in Fig. 8, and computational times for the different schemes are reported in Table 5.

Table 5: Performance of the different schemes for the diastole test case (Test B). For each scheme we report: the total wall time required for the solution of the test case, as well as the average wall time per time step, the total time spent assembling linear systems and the time spent solving them; the minimum, average and maximum of the number of iterations per timestep required for convergence of the Newton method, for monolithic schemes, and of the fixed-point (FP) method, for partitioned schemes; the average number of fluid and solid assembly calls per timestep; the average number of linear solver iterations per timestep (across all Newton or partitioned iterations). Refer to Table 2 for the explanation of the abbreviations used.

Scheme	Wall time [s] (relative to best)				Newton/FP iter.			Avg. assembly calls		Avg. GMRES iter.		
	total	avg.	assembly	linear sol.	min.	avg.	max.	fluid	solid	monol.	fluid	solid
ME	6560 (1.00)	6.56	3830 (1.00)	1730 (1.00)	2	2.0	2	3.0	3.0	157.7	-	-
PE-DN-SR	Wall time > 24 h											
PE-DN-AitA	convergence failure											
PE-DN-AndA	convergence failure											
PE-RN-SR	Wall time > 24 h											
PE-RN-AitA	50050 (7.63)	50.1	22677 (5.92)	4811 (2.78)	12	14.1	15	14.1	28.3	-	575.0	426.6
PE-RN-AndA	46650 (7.11)	46.7	21235 (5.54)	4148 (2.39)	12	13.4	14	13.4	26.1	-	543.0	390.3
MI	10900 (1.00)	10.9	6150 (1.00)	2660 (1.00)	3	4.0	4	5.0	5.0	262.7	-	-
PI-DN-SR	Wall time > 24 h											
PI-DN-AitA	convergence failure											
PI-DN-AndA	convergence failure											
PI-RN-SR	Wall time > 24 h											
PI-RN-AitA	50870 (4.67)	50.9	22780 (3.70)	4773 (1.79)	12	14.0	15	14.0	28.6	-	583.6	432.6
PI-RN-AndA	49700 (4.56)	49.7	22168 (3.60)	4617 (1.73)	12	13.7	14	13.7	27.0	-	562.1	405.9

(a) $t = 0.2$ms　　　　(b) $t = 100$ms　　　　(c) $t = 200$ms
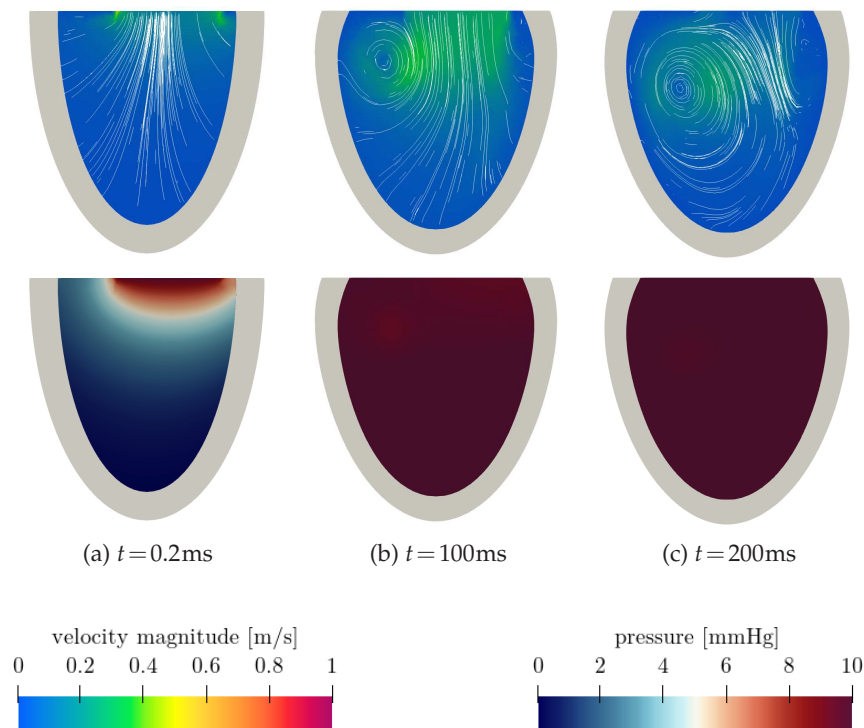
velocity magnitude [m/s]

pressure [mmHg]

Figure 8: Snapshots of the fluid velocity (top) and pressure (bottom) of the idealized ventricle test case, in diastole (Test B).

From there, it is again evident that the monolithic approach outperforms the partitioned approach, both with an explicit and with an implicit treatment of the geometry, even during the relatively slow diastolic phase.

5.4　Test C: idealized left ventricle, full heartbeat

In this test we assess the reliability of the ME scheme for the simulation of a full heartbeat on an idealized ventricle geometry. This is of particular interest since specific features arise when both systolic and diastolic are considered. In particular, during the heartbeat, the opening and closing of valves determine four distinct phases [66]:

- isovolumetric contraction; both valves are closed; the ventricle starts contracting, resulting in a rapid increase of ventricular pressure; when the pressure inside the chamber becomes larger than that of the aorta, the aortic valve opens and ejection phase begins;

- ejection; the aortic valve is open, and the mitral valve is closed; blood exits from the ventricle into the aorta. The aortic valve closes to prevent reverse flow, starting the isovolumetric relaxation phase;

- isovolumetric relaxation; both valves are closed, and the ventricle relaxes, resulting in the decrease of ventricular pressure; when the pressure falls below the atrial pressure, mitral valve opens and the filling phase starts;

- filling; the mitral valve is open, and the aortic valve is closed; blood flows from the atrium into the ventricle, increasing its volume; the mitral valve closes to prevent reverse flow, and a new isovolumetric contraction phase begins.

We reproduce these phases by switching the boundary conditions applied on Γ_{AV} and Γ_{MV} according to the ventricular pressure and to the flow through the valve orifices. Specifically, we represent open valves with the resistance condition (5.1) for the aortic valve and the Neumann condition for the mitral valve, and closed valves with the no-slip condition $\mathbf{u} = \mathbf{0}$. Valves are opened according to the average pressure $\bar{p}$ in the fluid domain: the aortic valve is opened when $\bar{p} \geq p_{AV}$, and the mitral valve when $\bar{p} \leq p_{MV}$. Conversely, valves are closed when reverse flow occurs through their section.

We treat active contraction in the active stress framework for this test, using the same analytically prescribed activation function as in Test A (Section 5.1). We consider the same mesh as the one used for Tests A and B. With respect to previous tests, we also change boundary and initial conditions. On the ventricular base of the structure, we impose a Robin-like condition as in (5.2), to allow for the base to move towards the apex as the ventricle contracts, as is the case in healthy hearts. We remark that this leads inlet and outlet boundaries for the fluid domain to move, and this can be problematic in more complex test cases with realistic geometries. Moreover, we start the simulation by inflating the structure with a pressure of 1333 Pa, roughly corresponding to the pressure at the end of the diastolic phase. This allows to start from a more realistic initial condition.

We use an active stress formulation for active mechanics. The final simulation time is $T = 0.8$ s, and we set $\Delta t = 2 \cdot 10^{-4}$ s. Overall, the total wall time needed for the simulation was 14 h.

We report ventricular volume and pressure in Fig. 9. The evolution of volume and pressure is qualitatively consistent with the behavior of a normal human heart [66]. Some snapshots of the solution are shown in Figs. 10 and 11. It can be seen how the ventricle deforms and contracts during systole by shortening and thickening, and how the original volume and shape are recovered at the end of the heartbeat, after relaxation.

It is worth remarking that this simplified model includes isovolumetric phases of the heartbeat, as can be seen from Fig. 9. Such phases are usually neglected in purely CFD simulations, in which the endocardial wall displacement is provided as data [41,103,106]. Indeed, because of the fluid incompressibility and the fact that both valves are closed, in isovolumetric phases prescribing a Dirichlet condition on the whole boundary would generally be incompatible with the divergence-free constraint. As a matter of fact, Dirichlet data should satisfy a compatibility condition [86], often not satisfied by noisy imaging[†]. Moreover, pressure would be defined only up to a constant, in that setting. Therefore, isovolumetric phases cannot be treated in a CFD model, unless special techniques

---

[†]Notice that this is true only for physiological cases, where the valves are perfectly closed. In regurgitant
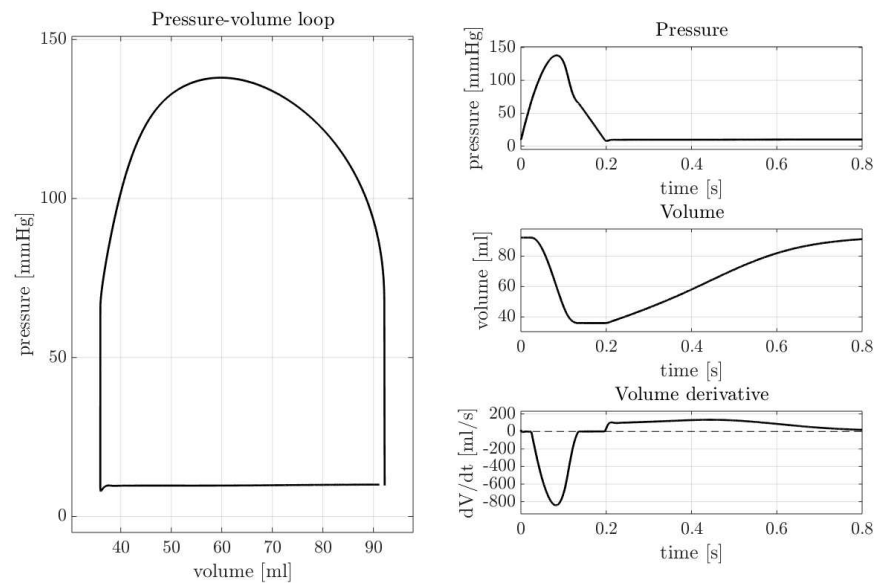
Figure 9: Pressure and volume plots of the full heartbeat test case (Test C).

are implemented to obtain a well-posed problem [106]. An FSI model is instead capable of representing isovolumetric phases [81, 106], thanks to the fact that the velocity at the endocardial wall is not known from data but is itself an unknown of the problem, and the model also accounts for the stresses exchanged at the fluid-structure interface.

However, after time and space discretizations, in our computational model the fluid volume is not exactly preserved over time. This is due mostly to two different sources of spurious volume variations:

- explicit treatment of the geometry: the use of the structure displacement at previous timestep to compute the fluid domain at current timestep leads to a mismatch between the fluid volume and the fluid velocity at the boundary;

- SUPG stabilization terms: these can be interpreted as artificial compressibility, so that at the discrete level the fluid is not strictly incompressible, and a small variation in volume is observed.

Both effects reduce as $\Delta t \to 0$. Moreover, the volume variations that they introduce are several orders of magnitude smaller than the characteristic variations in volume of the ventricle over a heartbeat: in this test case, during isovolumetric contraction, the total volume variation was of 0.07 mL, corresponding to 0.075 % of the initial volume. Overall, these spurious effects can be considered negligible.

---

cases, such as mitral valve regurgitation, a Neumann boundary would appear in correspondence of the regurgitation orifice.
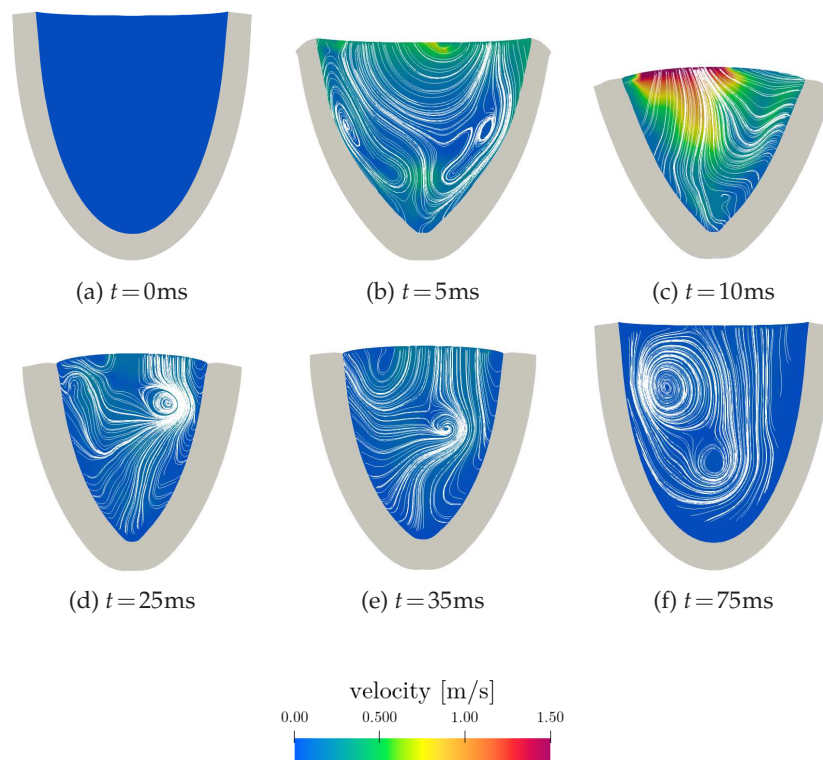
Figure 10: Snapshots of the fluid velocity of the idealized ventricle test case, for the full heartbeat (Test C).

6   Conclusions

In the context of FSI heart modeling, we compare in a systematic way fully coupled partitioned and monolithic FSI coupling schemes on benchmark cases that mimic, in an idealized setting, the flow regime characteristic of a human ventricle. For all the cases considered, the performance of the monolithic scheme was significantly better than that of partitioned iterations, resulting in a total computational time several times smaller for the whole heartbeat simulation. Partitioned schemes based on Dirichlet-Neumann interface conditions suffer from convergence issues, and have proven to be impractical for the cardiac tests. Robin-Neumann schemes performed slightly better, and benefit from acceleration methods, but they still require parameter tuning to be used effectively. For the complex geometries and material models characterizing the heart, this tuning is not trivial. Conversely, the monolithic approach is parameter-free, and is consequently more robust and flexible.

In the monolithic approach, using an implicit discretization for fluid domain displacement entails a higher computational cost than using an explicit discretization. Fully coupled partitioned schemes, on the contrary, have similar costs with both explicit and implicit geometric coupling. Nonetheless, he monolithic scheme is more efficient than

(a) $t = 0\,\text{ms}$ (b) $t = 5\,\text{ms}$ (c) $t = 10\,\text{ms}$

(d) $t = 15\,\text{ms}$ (e) $t = 35\,\text{ms}$ (f) $t = 75\,\text{ms}$

pressure
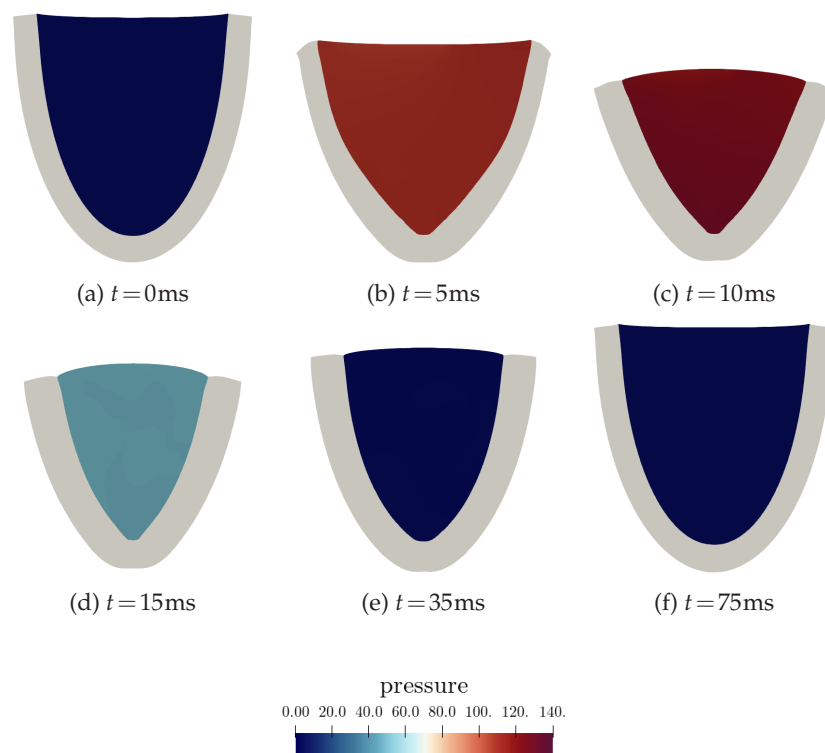
0.00 20.0 40.0 60.0 80.0 100. 120. 140.

Figure 11: Snapshots of the fluid pressure of the idealized ventricle test case, for the full heartbeat (Test C).

the partitioned ones even if an implicit geometric coupling is used. We also found that Anderson acceleration is capable of slightly improving the results of partitioned schemes. Moreover, the relative performance of the schemes does not vary depending on the choice of active mechanics formulation: indeed, with both with active stress and active strain formulations, the monolithic scheme performed significantly faster than the partitioned ones. The relative efficiency of monolithic schemes compared to the partitioned ones is verified in a significant case even as the number of parallel processes is changed.

Finally, using the monolithic scheme, we run a simulation of a full heartbeat on the idealized geometry, switching inlet and outlet boundary conditions to simulate in a simplified way the opening and closing of cardiac valves. This test case shows the ability of the computational model to replicate a full heartbeat. In particular, we reproduce the isovolumetric phases, which is a challenge in other simulation settings involving three-dimensional modeling of the cardiac blood flow.

Overall, this study indicates that, for strongly coupled FSI simulations in the cardiac context, a monolithic approach seems to be preferable to a partitioned one, thanks to its robustness and efficiency. However, more specific studies on the Robin interface parameters for the cardiac case are mandatory to improve the corresponding partitioned schemes.

Acknowledgements

Appendix A   Definition of algebraic operators

In this section we will give the explicit definition of each of the operators introduced in Section 3. We will not give details on the derivation of the operators, as it follows the classical finite elements procedure, and refer the interested reader to [62, 86].

In the following, we shall use the subscripts f, s and Σ for the vectors $\mathbf{G}, \mathbf{D}$ and $\mathbf{U}$ to denote their entries that correspond to nodes located on the interior (not on the interface) of the fluid domain, on the interior of the structure domain and on the fluid-structure interface respectively.

We shall assume that the same discretization is used for the fluid domain displacement and fluid velocity, so that both are described using the same set of basis functions. We partition the basis functions as done for the solution vectors into a subset for interior nodes and a subset for interface nodes. We will denote the basis functions as follows:

- $\boldsymbol{\varphi}_{\mathrm{s},i}^{\mathrm{s}}$, for $i \in \{1,\ldots,N_{\mathrm{s}}^{\mathrm{s}}\}$, are the basis functions for the structure displacement corresponding to the interior nodes of the structure domain (i.e. such that their support does not intersect Σ);

- $\boldsymbol{\varphi}_{\Sigma,i}^{\mathrm{s}}$, for $i \in \{1,\ldots,N_{\Sigma}^{\mathrm{s}}\}$, are the basis functions for the structure displacement corresponding to the interface nodes of the structure domain (i.e. such that their support intersects Σ);

- $\boldsymbol{\varphi}_{\mathrm{f},i}^{\mathrm{f}}$, for $i \in \{1,\ldots,N_{\mathrm{f}}^{\mathrm{f}}\}$, are the basis functions for the fluid velocity and fluid domain displacement corresponding to the interior nodes of the fluid domain;

- $\boldsymbol{\varphi}_{\Sigma,i}^{\mathrm{f}}$, for $i \in \{1,\ldots,N_{\Sigma}^{\mathrm{f}}\}$, are the basis functions for the fluid velocity and fluid domain displacement corresponding to the interface nodes of the fluid domain;

- ψ_i, for $i \in \{1, \ldots, N^{\mathrm{P}}\}$, are the basis function for the fluid pressure.

We assume that the fluid and structure meshes are conforming, so that there exists a one-to-one correspondence between interface nodes on the two subdomains. This entails $N_{\Sigma}^{\mathrm{s}} = N_{\Sigma}^{\mathrm{f}}$. We will assume that the numbering of interface basis functions on the two subdomains is the same, so that $\boldsymbol{\varphi}_{\Sigma,i}^{\mathrm{s}} = \boldsymbol{\varphi}_{\Sigma,i}^{\mathrm{f}}$ on Σ.

A.1 Structure displacement

The operator $\mathbb{S}_{\mathrm{s}}$ is defined as

$$\mathbb{S}_{\mathrm{s}}(\mathbf{D}^{n+1}) = \frac{\rho_{\mathrm{s}}}{\Delta t} M_{\mathrm{ss}}^{\mathrm{s}} \left(\mathbf{D}_{\mathrm{s}}^{n+1} - 2\mathbf{D}_{\mathrm{s}}^n + \mathbf{D}_{\mathrm{s}}^{n-1}\right) + \frac{\rho_{\mathrm{s}}}{\Delta t} M_{\mathrm{s}\Sigma}^{\mathrm{s}} \left(\mathbf{D}_{\Sigma}^{n+1} - 2\mathbf{D}_{\Sigma}^n + \mathbf{D}_{\Sigma}^{n-1}\right) + K_{\mathrm{s}}^{\mathrm{s}}(\mathbf{D}^{n+1}),$$

where

$$M_{\mathrm{kl}}^{\mathrm{s}} \in \mathbb{R}^{N_{\mathrm{k}}^{\mathrm{s}} \times N_{\mathrm{l}}^{\mathrm{s}}}, \quad \left(M_{\mathrm{kl}}^{\mathrm{s}}\right)_{i,j} = \int_{\widehat{\Omega}_{\mathrm{s}}} \boldsymbol{\varphi}_{\mathrm{k},i}^{\mathrm{s}} \cdot \boldsymbol{\varphi}_{\mathrm{l},j}^{\mathrm{s}}, \quad i \in \{1, \ldots, N_{\mathrm{k}}^{\mathrm{s}}\}, \quad j \in \{1, \ldots, N_{\mathrm{l}}^{\mathrm{s}}\}, \quad \mathrm{k,l} \in \{\mathrm{s}, \Sigma\}$$

are the structure mass matrix blocks corresponding to the different combinations of interior and interface basis functions, and

$$K_{\mathrm{k}}^{\mathrm{s}}(\mathbf{D}^{n+1}) \in \mathbb{R}^{N_{\mathrm{k}}^{\mathrm{s}}}, \quad \left(K_{\mathrm{k}}^{\mathrm{s}}\right)_i = \int_{\widehat{\Omega}_{\mathrm{s}}} P(\mathbf{d}^{n+1}) : \boldsymbol{\nabla}\boldsymbol{\varphi}_{\mathrm{k},i}^{\mathrm{s}}, \quad i \in \{1, \ldots, N_{\mathrm{k}}^{\mathrm{s}}\}, \quad \mathrm{k} \in \{\mathrm{s}, \Sigma\}$$

is the nonlinear stiffness operator, corresponding to either the interior or interface basis functions.

A.2 Fluid domain displacement

The operators $\mathbb{G}_{\mathrm{f}}$ and $\mathbb{C}_{\mathrm{g}}$ are defined as

$$\mathbb{G}_{\mathrm{f}}(\mathbf{G}^{n+1}) = K_{\mathrm{ff}}^{\mathrm{g}} \mathbf{G}_{\mathrm{f}}^{n+1} + K_{\mathrm{f}\Sigma}^{\mathrm{g}} \mathbf{G}_{\Sigma}^{n+1},$$
$$\mathbb{C}_{\mathrm{g}}(\mathbf{G}^{n+1}, \mathbf{D}^{n+1}) = \mathbf{G}_{\Sigma}^{n+1} - \mathbf{D}_{\Sigma}^{n+1},$$

where

$$K_{\mathrm{kl}}^{\mathrm{g}} \in \mathbb{R}^{N_{\mathrm{k}}^{\mathrm{f}} \times N_{\mathrm{l}}^{\mathrm{f}}}, \quad \left(K_{\mathrm{kl}}^{\mathrm{g}}\right)_{i,j} = \int_{\widehat{\Omega}_{\mathrm{f}}} \boldsymbol{\nabla}\boldsymbol{\varphi}_{\mathrm{k},i}^{\mathrm{f}} : \boldsymbol{\nabla}\boldsymbol{\varphi}_{\mathrm{l},j}^{\mathrm{f}}, \quad i \in \{1, \ldots, N_{\mathrm{k}}^{\mathrm{f}}\}, \quad j \in \{1, \ldots, N_{\mathrm{l}}^{\mathrm{f}}\}, \quad \mathrm{k,l} \in \{\mathrm{f}, \Sigma\}$$

are the stiffness matrix blocks corresponding to the different combinations of interior and interface basis functions. The operator $\mathbb{G}$ reads

$$\mathbb{G}(\mathbf{G}^{n+1}, \mathbf{D}^{n+1}) = \begin{bmatrix} \mathbb{G}(\mathbf{G}^{n+1}) \\ \mathbb{C}_{\mathrm{g}}(\mathbf{G}^{n+1}, \mathbf{D}^{n+1}) \end{bmatrix}.$$

A.3   Fluid problem

Let us denote by $\mathbf{V}$ and $\mathbf{P}$ the subset of fluid degrees of freedom $\mathbf{U}$ related to velocity and pressure respectively. The operator $\mathbb{F}_{\mathrm{f}} = \left[\mathbb{F}_{\mathrm{f}}^{\mathrm{U}}, \mathbb{F}_{\mathrm{f}}^{\mathrm{P}}\right]^{T}$ has two blocks, corresponding to the discretization of the momentum and incompressibility equations respectively. The first block reads

$$
\begin{aligned}
\mathbb{F}_{\mathrm{f}}^{\mathrm{U}}(\mathbf{U}^{n+1}, \mathbf{G}^{n+1}) = {} & \frac{\rho_{\mathrm{f}}}{\Delta t} M_{\mathrm{ff}}^{\mathrm{f}}(\mathbf{V}_{\mathrm{f}}^{n+1} - \mathbf{V}_{\mathrm{f}}^{n}) + \frac{\rho_{\mathrm{f}}}{\Delta t} M_{\mathrm{f}\Sigma}^{\mathrm{f}}(\mathbf{V}_{\Sigma}^{n+1} - \mathbf{V}_{\Sigma}^{n}) \\
& + C_{\mathrm{ff}}(\mathbf{V}^{n}, \mathbf{G}^{n+1}, \mathbf{G}^{n})\mathbf{V}_{\mathrm{f}}^{n+1} + C_{\mathrm{f}\Sigma}(\mathbf{V}^{n}, \mathbf{G}^{n+1}, \mathbf{G}^{n})\mathbf{V}_{\Sigma}^{n+1} \\
& + K_{\mathrm{ff}}^{\mathrm{f}}\mathbf{U}_{\mathrm{f}}^{n+1} + K_{\mathrm{f}\Sigma}^{\mathrm{f}}\mathbf{U}_{\Sigma}^{n+1} + B_{\mathrm{f}}^{T}\mathbf{P}^{n+1},
\end{aligned}
$$

while the second one is given by

$$
\mathbb{F}_{\mathrm{f}}^{\mathrm{P}}(\mathbf{U}^{n+1}, \mathbf{U}^{n}, \mathbf{G}^{n+1}, \mathbf{G}^{n}) = B_{\mathrm{f}}\mathbf{V}_{\mathrm{f}}^{n+1} + B\Sigma\mathbf{V}_{\Sigma}^{n+1}.
$$

In the above equations,

$$
M_{\mathrm{kl}}^{\mathrm{f}} \in \mathbb{R}^{N_{\mathrm{k}}^{\mathrm{f}} \times N_{\mathrm{l}}^{\mathrm{f}}}, \quad \left(M_{\mathrm{kl}}^{\mathrm{f}}\right)_{i,j} = \int_{\Omega_{\mathrm{f}}^{t}} \boldsymbol{\varphi}_{\mathrm{k},i}^{\mathrm{f}} \cdot \boldsymbol{\varphi}_{\mathrm{l},j}^{\mathrm{f}}, \quad i \in \{1, \dots, N_{\mathrm{k}}^{\mathrm{f}}\}, \quad j \in \{1, \dots, N_{\mathrm{l}}^{\mathrm{f}}\}, \quad \mathrm{k}, \mathrm{l} \in \{\mathrm{f}, \Sigma\}
$$

are the fluid velocity mass matrix blocks corresponding to the different combinations of interior and interface basis functions. C_{ff} and $C_{\mathrm{f}\Sigma}$ arise from the discretization of the advection term of the momentum equation, and are given by

$$
C_{\mathrm{kl}} \in \mathbb{R}^{N_{\mathrm{k}}^{\mathrm{f}} \times N_{\mathrm{l}}^{\mathrm{f}}}, \quad \left(C_{\mathrm{kl}}^{\mathrm{f}}\right)_{i,j} = \int_{\Omega_{\mathrm{f}}^{t}} (\mathbf{w} \cdot \boldsymbol{\nabla})\boldsymbol{\varphi}_{\mathrm{l},k}^{\mathrm{f}} \cdot \boldsymbol{\varphi}_{\mathrm{k},i}^{\mathrm{f}}, \quad i \in \{1, \dots, N_{\mathrm{k}}^{\mathrm{f}}\}, \quad j \in \{1, \dots, N_{\mathrm{l}}^{\mathrm{f}}\}, \quad \mathrm{k}, \mathrm{l} \in \{\mathrm{f}, \Sigma\},
$$

in which $\mathbf{w}$ is the advection velocity, computed from the discrete velocity and fluid domain displacement vectors through

$$
\mathbf{w} = \sum_{i=1}^{N_{\mathrm{f}}^{\mathrm{f}}} \left(\mathbf{U}_{\mathrm{f},i}^{n} - \frac{\mathbf{G}_{\mathrm{f},i}^{n+1} - \mathbf{G}_{\mathrm{f},i}^{n}}{\Delta t}\right)\boldsymbol{\varphi}_{\mathrm{f},i}^{\mathrm{f}} + \sum_{i=1}^{N_{\Sigma}^{\mathrm{f}}} \left(\mathbf{U}_{\Sigma,i}^{n} - \frac{\mathbf{G}_{\Sigma,i}^{n+1} - \mathbf{G}_{\Sigma,i}^{n}}{\Delta t}\right)\boldsymbol{\varphi}_{\Sigma,i}^{\mathrm{f}}.
$$

Matrices $K_{\mathrm{ff}}^{\mathrm{f}}$ and $K_{\mathrm{f}\Sigma}^{\mathrm{f}}$ are stiffness matrices for the fluid, defined by

$$
K_{\mathrm{kl}}^{\mathrm{f}} \in \mathbb{R}^{N_{\mathrm{k}}^{\mathrm{f}} \times N_{\mathrm{l}}^{\mathrm{f}}}, \quad \left(K_{\mathrm{kl}}^{\mathrm{g}}\right)_{i,j} = \int_{\widehat{\Omega}_{\mathrm{f}}} 2\mu\varepsilon(\boldsymbol{\varphi}_{\mathrm{l},j}^{\mathrm{f}}) : \boldsymbol{\nabla}\boldsymbol{\varphi}_{\mathrm{k},i}^{\mathrm{f}}, \quad i \in \{1, \dots, N_{\mathrm{k}}^{\mathrm{f}}\}, \quad j \in \{1, \dots, N_{\mathrm{l}}^{\mathrm{f}}\}, \quad \mathrm{k}, \mathrm{l} \in \{\mathrm{f}, \Sigma\},
$$

and the matrices B_{f} and B_{Σ} are defined by

$$
B_{\mathrm{k}} \in \mathbb{R}^{N^{\mathrm{P}} \times N_{\mathrm{k}}^{\mathrm{f}}}, \quad (B_{\mathrm{k}})_{i,j} = \int_{\Omega_{\mathrm{f}}^{t}} \psi_{i} \boldsymbol{\nabla} \cdot \boldsymbol{\varphi}_{\mathrm{k},j}^{\mathrm{f}}, \quad i \in \{1, \dots, N^{\mathrm{P}}\}, \quad j \in \{1, \dots, N_{\mathrm{k}}^{\mathrm{f}}\}, \quad \mathrm{k} \in \{\mathrm{f}, \Sigma\}.
$$

Terms arising from the SUPG stabilization are not reported here for the sake of brevity, but they are discussed e.g. in [31, 86, 105, 116].

A.4　Coupling conditions

The kinematic coupling operator $\mathbb{C}_k$ is defined as

$$\mathbb{C}_k(\mathbf{U}^{n+1},\mathbf{D}^{n+1}) = M_{\Sigma\Sigma}\left(\mathbf{V}_\Sigma^{n+1} - \frac{\mathbf{D}_\Sigma^{n+1} - \mathbf{D}_\Sigma^n}{\Delta t}\right).$$

We remark that in the implementation of DN scheme and of monolithic scheme, the continuity of velocity is applied essentially, i.e. we strongly impose the constraint that fluid and structure velocity interface degrees of freedom are equal. However, this formulation of the kinematic coupling operator is relevant when defining the interface conditions involved in partitioned schemes based on Robin conditions [14, 15].

Dynamic coupling is imposed in residual form [14], so that the dynamic coupling operator $\mathbb{C}_d$ reads

$$
\begin{aligned}
\mathbb{C}_d(\mathbf{D}^{n+1},\mathbf{U}^{n+1},\mathbf{G}^{n+1}) ={}& \frac{\rho_s}{\Delta t}M_{\Sigma s}^s\left(\mathbf{D}_s^{n+1} - 2\mathbf{D}_s^n + \mathbf{D}_s^{n-1}\right) + \frac{\rho_s}{\Delta t}M_{\Sigma\Sigma}^s\left(\mathbf{D}_\Sigma^{n+1} - 2\mathbf{D}_\Sigma^n + \mathbf{D}_\Sigma^{n-1}\right) \\
&+ K_\Sigma^s(\mathbf{D}^{n+1}) + \frac{\rho_f}{\Delta t}M_{\Sigma f}^f\left(\mathbf{V}_f^{n+1} - \mathbf{V}_f^n\right) + \frac{\rho_f}{\Delta t}M_{\Sigma\Sigma}^f\left(\mathbf{V}_\Sigma^{n+1} - \mathbf{V}_\Sigma^n\right) \\
&+ C_{\Sigma f}(\mathbf{V}^n,\mathbf{G}^{n+1},\mathbf{G}^n)\mathbf{V}_f^{n+1} + C_{\Sigma\Sigma}(\mathbf{V}^n,\mathbf{G}^{n+1},\mathbf{G}^n)\mathbf{V}_\Sigma^{n+1} \\
&+ K_{\Sigma f}^f\mathbf{U}_f^{n+1} + K_{\Sigma\Sigma}^f\mathbf{U}_\Sigma^{n+1} + B_\Sigma^T\mathbf{P}^{n+1}.
\end{aligned}
$$

Appendix B　Relaxation and convergence acceleration

The convergence properties of PE and PI schemes can be enhanced by using suitable relaxation or convergence acceleration schemes. We embed such schemes in the operator $\mathbb{R}_{(k+1)}(\widetilde{\mathbf{D}}_{(k+1)}^{n+1},\mathbf{D}_{(k)}^{n+1},\mathbf{D}_{(k-1)}^{n+1},\ldots)$ that, for a given guess $\widetilde{\mathbf{D}}_{(k+1)}^{n+1}$ for the structure displacement at next iteration and the structure displacements of previous iterations, computes $\mathbf{D}_{(k+1)}^{n+1}$. We consider three different relaxation operators.

B.1　Static relaxation (SR)

The relaxation operator is given by [14, 27, 69]

$$\mathbb{R}_{(k+1)}\left(\widetilde{\mathbf{D}}_{(k+1)}^{n+1},\mathbf{D}_{(k)}^{n+1},\mathbf{D}_{(k-1)}^{n+1},\ldots\right) = \lambda\widetilde{\mathbf{D}}_{(k+1)}^{n+1} + (1-\lambda)\mathbf{D}_{(k)}^{n+1}, \quad k=0,1,2,\ldots, \tag{B.1}$$

where $\lambda \in (0,1]$ is a suitable parameter whose choice is critical to the convergence of the scheme; in particular, large values of λ typically lead to a fast convergence (when convergence occurs), but too large values of λ may cause the scheme to diverge [27].

In our numerical simulations, we used a relaxation of $\lambda = 0.005$ for all DN schemes, and a relaxation $\lambda = 1.0$ for RN schemes.

B.2 Aitken acceleration (AitA)

The relaxation operator is defined as [14, 69, 76]

$$\mathbb{R}_{(k+1)}\left(\widetilde{\mathbf{D}}_{(k+1)}^{n+1}, \mathbf{D}_{(k)}^{n+1}, \mathbf{D}_{(k-1)}^{n+1}, \dots\right) = \lambda_{(k+1)}\widetilde{\mathbf{D}}_{(k+1)}^{n+1} + (1-\lambda_{(k+1)})\mathbf{D}_{(k)}^{n+1}, \quad k=0,1,2,\dots \quad (\text{B.2})$$

with $\lambda_{(k+1)}$ computed as [69]

$$\lambda_{(k+1)} = \lambda_{(k)} \frac{\widetilde{\mathbf{D}}_k^{n+1} \cdot \left(\widetilde{\mathbf{D}}_{(k+1)}^{n+1} - \mathbf{D}_{(k)}^{n+1} - \widetilde{\mathbf{D}}_k^{n+1} + -\mathbf{D}_{(k-1)}^{n+1}\right)}{\left\|\widetilde{\mathbf{D}}_{(k+1)}^{n+1} - \mathbf{D}_{(k)}^{n+1} - \widetilde{\mathbf{D}}_k^{n+1} + -\mathbf{D}_{(k-1)}^{n+1}\right\|}, \quad k=0,1,2,\dots$$

for λ_0 given. This scheme typically improves the convergence with respect to static relaxation [69]. However, the choice of the initial relaxation parameter λ_0 is still problem dependent and needs to be manually tuned [69].

In our numerical simulations, we used a relaxation of $\lambda_0 = 0.005$ for all DN schemes, and a relaxation $\lambda_0 = 1.0$ for RN schemes.

B.3 Anderson acceleration (AndA)

Anderson acceleration (AndA) [7, 37, 112], also known as Anderson mixing, can be interpreted as a multi-secant method [37] or as a nonlinear generalization of GMRES [112] and is based on computing the new iterate making use of the previous m ones, with m a suitable integer parameter. While generally used for self-consistent field iterations in the computation of electronic structures [37], it has been also been applied to domain decomposition [112] and multiphysics problems [23] and can be used to accelerate the convergence of FSI fixed-point schemes [74].

The procedure behind Anderson acceleration can be detailed as follows [112]. Consider the fixed-point iteration

$$x_{(k+1)} = g(x_k)$$

used to compute the solution of $x = g(x)$. Then, given $m \in \mathbb{N}$ and an initial guess x_0, Anderson acceleration of the sequence $x_{(k+1)}$ can be computed as follows:

1. Set $m_{(k)} = \min\{m, k\}$.

2. Set $F_{(k)} = [f_{k-m_{(k)}}, f_{k-m_{(k)}+1}, \dots, f_{(k)}]$, where $f_i = g(x_i) - x_i$.

3. Find $\alpha^{(k)} = [\alpha_0^{(k)}, \alpha_1^{(k)}, \dots, \alpha_{m_{(k)}}^{(k)}]^T = \arg\min_{\alpha \in A} \|F_{(k)}\alpha\|$, with

$$A = \left\{\alpha = \left[\alpha_0, \alpha_1, \dots, \alpha_{m_{(k)}}\right]^T \text{ such that } \sum_{i=0}^{m_{(k)}} \alpha_i = 1\right\}.$$

4. Set $x_{(k+1)} = \sum_{i=0}^{m_{(k)}} \alpha_i^{(k)} g(x_{k-m_{(k)}+i})$.

Details on how to interpret the method, as well as alternative formulations, criteria for the choice of m and implementation details can be found in [112].

In all numerical simulations involving Anderson acceleration, we used $m = 10$.

References

[1] CINECA GALILEO100 technical specifications, `https://wiki.u-gov.it/confluence/display/SCAIUS/UG3.3`

[2] Official `deal.ii` website `https://www.dealii.org/`

[3] Official `life`ˣ website `https://lifex.gitlab.io/lifex/`

[4] P. C. Africa, R. Piersanti, M. Fedele, L. Dede, and A. Quarteroni, Lifex-heart module: a high-performance simulator for the cardiac function package 1: Fiber generation, arXiv: 2201.03303, 2022.

[5] D. Ambrosi, G. Arioli, F. Nobile, and A. Quarteroni, Electromechanical coupling in cardiac dynamics: the active strain approach, *SIAM J. Appl. Math*, 71(2):605–621, 2011.

[6] D. Ambrosi and S. Pezzuto, Active stress vs. active strain in mechanobiology: constitutive issues, *J. Elast.*, 107(2):199–212, 2012.

[7] D. G. Anderson, Iterative procedures for nonlinear integral equations, *J. ACM*, 1965.

[8] D. Arndt et al., The `deal.II` library, version 9.2, *J. Numer. Math.*, 28(3):131–146, 2020.

[9] D. Arndt et al., The `deal.II` finite element library: design, features, and insights, *Comput. Math. Appl.*, 2020.

[10] M. Astorino, F. Chouly, and M. A. Fernández, Robin based semi-implicit coupling in fluid-structure interaction: Stability analysis and numerics, *SIAM J. Sci. Comput.*, 31(6):4041–4065, 2010.

[11] M. Astorino, J. F. Gerbeau, O. Pantz, and K. F. Traoré, Fluid-structure interaction and multi-body contact. Application to the aortic valves, *Comput. Methods Appl. Mech. Eng.*, 198:3603–3612, 2009.

[12] C. M. Augustin et al., Patient-specific modeling of left ventricular electromechanics as a driver for haemodynamic analysis, *EP Europace*, 18(suppl_4):iv121–iv129, 2016.

[13] C. M. Augustin et al., Anatomically accurate high resolution modeling of human whole heart electromechanics: a strongly scalable algebraic multigrid solver method for nonlinear deformation, *J. Comput. Phys.*, 305:622–646, 2016.

[14] S. Badia, F. Nobile, and C. Vergara, Fluid-structure partitioned procedures based on Robin transmission conditions, *J. Comput. Phys.*, 227:7021–7051, 2008.

[15] S. Badia, F. Nobile, and C. Vergara, Robin-Robin preconditioned Krylov methods for fluid-structure interaction problems, *Comput. Methods Appl. Mech. Eng.*, 198:2768–2784, 2009.

[16] S. Badia, A. Quaini, and A. Quarteroni, Modular vs. non-modular preconditioners for fluid-structure systems with large added-mass effect, *Comput. Methods Appl. Mech. Eng.*, 197(49-50):4216–4232, 2008.

[17] S. Badia, A. Quaini, and A. Quarteroni, Splitting methods based on algebraic factorization for fluid-structure interaction, *SIAM J. Sci. Comput.*, 30(4):1778–1805, 2008.

[18] J. W. Banks, W. D. Henshaw, and D. W. Schwendeman, An analysis of a new stable partitioned algorithm for FSI problems. Part I: Incompressible flow and elastic solids, *J. Comput. Phys*, 269:108–137, 2014.

[19] Y. Bazilevs, V. M. Calo, T. J. Hughes, and Y. Zhang, Isogeometric fluid-structure interaction: theory, algorithms, and computations, *Comput. Mech.*, 43(1):3–37, 2008.

[20] Y. Bazilevs, K. Takizawa, and T. E. Tezduyar, *Computational Fluid-Structure Interaction: Methods and Applications*, John Wiley & Sons, 2013.

[21] P. J. Blanco and R. A. Feijóo, A 3D-1D-0D computational model for the entire cardiovascular system, *Mec. Comput.*, 29(59):5887–5911, 2010.

[22] D. Bluestein and S. Einav, Transition to turbulence in pulsatile flow through heart valves-

a modified stability approach, *J. Biomech. Eng.*, 116(4):477–487, 1994.

[23] J. W. Both, N. A. Barnafi, F. A. Radu, P. Zunino, and A. Quarteroni, Iterative splitting schemes for a soft material poromechanics model, *Comput. Methods Appl. Mech. Eng.*, 388:114183, 2022.

[24] M. Bukač, S. Čanić, R. Glowinski, J. Tambača, and A. Quaini, Fluid-structure interaction in blood flow capturing non-zero longitudinal structure displacement, *J. Comput. Phys*, 235:515–541, 2013.

[25] E. Burman and M. A. Fernández, Explicit strategies for incompressible fluid-structure interaction problems: Nitsche type mortaring versus Robin–Robin coupling, *Int. J. Numer. Methods Eng.*, 97(10):739–758, 2014.

[26] C. Burstedde, L. C. Wilcox, and O. Ghattas, `p4est`: Scalable algorithms for parallel adaptive mesh refinement on forests of octrees, *SIAM J. Sci. Comput.*, 33(3):1103–1133, 2011.

[27] P. Causin, J. F. Gerbeau, and F. Nobile, Added-mass effect in the design of partitioned algorithms for fluid-structure interaction, *Comput. Methods Appl. Mech. Eng.*, 194:4506–4527, 2005.

[28] Y. Cheng, H. Oertel, and T. Schenkel, Fluid-structure coupled cfd simulation of the left ventricular flow during filling phase, *Ann. Biomed. Eng.*, 33(5):567–576, 2005.

[29] P. Crosetto, S. Deparis, G. Fourestey, A. Quarteroni, Parallel algorithms for fluid-structure interaction problems in haemodynamics, *SIAM J. Sci. Comput.*, 33(4):1598–1622, 2011.

[30] Y. Dabiri, J. Yao, K. L. Sack, G. S. Kassab, and J. M. Guccione, Tricuspid valve regurgitation decreases after mitraclip implantation: Fluid structure interaction simulation, *Mech. Res. Commun.*, 97:96–100, 2019.

[31] L. Dede', F. Menghini, and A. Quarteroni, Computational fluid dynamics of blood flow in an idealized left human heart, *Int. J. Numer. Method. Biomed. Eng.*, 2019.

[32] S. Deparis, M. Discacciati, G. Fourestey, A. Quarteroni, Fluid-structure algorithms based on Steklov-Poincarè operators, *Comput. Methods Appl. Mech. Eng.*, 195(41–43):5797–5818, 2006.

[33] S. Deparis, D. Forti, G. Grandperrin, and A. Quarteroni, FaCSI: A block parallel preconditioner for fluid-structure interaction in hemodynamics, *J. Comput. Phys*, 327:700–718, 2016.

[34] S. Deparis, G. Grandperrin, and A. Quarteroni, Parallel preconditioners for the unsteady Navier-Stokes equations and applications to hemodynamics simulations, *Comput. Fluids*, 92:253–273, 2014.

[35] W. Dettmer and D. Perić, A computational framework for fluid-structure interaction: finite element formulation and applications, *Comput. Methods Appl. Mech. Eng.*, 195(41-43):5754–5779, 2006.

[36] D. R. Einstein et al., Fluid-structure interactions of the mitral valve and left heart: comprehensive strategies, past, present and future, *Int. J. Numer. Method. Biomed. Eng.*, 26(3-4):348–380, 2010.

[37] H. R. Fang and Y. Saad, Two classes of multisecant methods for nonlinear acceleration, *Numerical Linear Algebra Appl.*, 16:197–221, 2009.

[38] M. Fernández, J. F. Gerbeau, and C. Grandmont, A projection semi-implicit scheme for the coupling of an elastic structure with an incompressible fluid, *Int. J. Numer. Methods Eng.*, 69:794–821, 2007.

[39] M. Fernández and M. Moubachir, A Newton method using exact jacobians for solving fluid-structure coupling, *Comput. Struct.*, 83(2-3):127–142, 2005.

[40] M. A. Fernández, J. Mullaert, and M. Vidrascu, Explicit Robin-Neumann schemes for the coupling of incompressible fluids with thin-walled structures, *Comput. Methods Appl. Mech. Eng.*, 267:566–593, 2013.

[41] I. Fumagalli et al., An image-based computational hemodynamics study of the Systolic Anterior Motion of the mitral valve, *Comput. Biol. Med.*, 123(May):103922, 2020.

[42] M. Gee, C. Siefert, J. Hu, R. Tuminaro, and M. Sala, Ml 5.0 smoothed aggregation user's guide, Technical Report SAND2006-2649, Sandia National Laboratories, 2006.

[43] T. Gerach et al., Electro-mechanical whole-heart digital twins: A fully coupled multi-physics approach, *Mathematics*, 9(11):1247, 2021.

[44] L. Gerardo-Giorda, F. Nobile, and C. Vergara, Analysis and optimization of Robin-Robin partitioned procedures in fluid-structure interaction problems, *SIAM J. Numer. Anal.*, 48(6):2091–2116, 2010.

[45] A. Gerbi, *Numerical approximation of cardiac electro-fluid-mechanical models: coupling strategies for large-scale simulation*, PhD Thesis, EPFL, 2018.

[46] A. Gerbi, L. Dede', and A. Quarteroni, A monolithic algorithm for the simulation of cardiac electromechanics in the human left ventricle, *Math. Eng.*, 1(1):1–37, 2018.

[47] G. Gigante, G. Sambataro, C. Vergara, Optimized Schwarz methods for spherical interfaces with application to fluid-structure interaction, *SIAM J. Sci. Comput.*, 42(2):A751–A770, 2020.

[48] G. Gigante and C. Vergara, Analysis and optimization of the generalized Schwarz method for elliptic problems with application to fluid-structure interaction, *Numer. Math. (Heidelb.)*, 131(2):369–404, 2015.

[49] G. Gigante and C. Vergara, On the choice of interface parameters in Robin-Robin loosely coupled schemes for fluid-structure interaction, *Fluids*, 6(6):213, 2021.

[50] G. Gigante and C. Vergara, On the stability of a loosely-coupled scheme based on a Robin interface condition for fluid-structure interaction, *Comput. Math. Appl.*, 96:109–119, 2021.

[51] J. M. Guccione and A. D. McCulloch, Finite element modeling of ventricular mechanics, In: *Theory of Heart*, L. Glass, P. Hunter, and A. D. McCulloch (Eds.), Springer, 1991.

[52] G. Guidoboni, R. Glowinski, N. Cavallini, and S. Canic, Stable loosely-coupled-type algorithm for fluid-structure interaction in blood flow, *J. Comput. Phys*, 228(18):6916–6937, 2009.

[53] V. Gurev, T. Lee, J. Constantino, H. Arevalo, and N. A. Trayanova, Models of cardiac electromechanics based on individual hearts imaging data: Image-based electromechanical models of the heart, *Biomech. Model. Mechanobiol*, 10(3):295–306, 2011.

[54] W. Hackbush, *Multigrid Methods and Applications*, Springer-Verlag, 1985.

[55] M. Heil, An efficient solver for the fully coupled solution of large-displacement fluid-structure interaction problems, *Comput. Methods Appl. Mech. Eng.*, 193(1-2):1–23, 2004.

[56] M. Hirschhorn, V. Tchantchaleishvili, R. Stevens, J. Rossano, and A. Throckmorton, Fluid–structure interaction modeling in cardiovascular medicine-a systematic review 2017-2019, *Med. Eng. Phys.*, 78:1–13, 2020.

[57] M. Hirschvogel, M. Bassilious, L. Jagschies, S. M. Wildhirt, and M. W. Gee, A monolithic 3D-0D coupled closed-loop model of the heart and the vascular system: experiment-based parameter estimation for patient-specific cardiac mechanics, *Int. J. Numer. Methods Biomed. Eng.*, 33(8):e2842, 2017.

[58] G. A. Holzapfel and R. W. Ogden, Constitutive modelling of passive myocardium: A structurally based framework for material characterization, *Philos. Trans. R. Soc. A*, 367(1902): 3445–3475, 2009.

[59] G. Hou, J. Wang, and A. Layton, Numerical methods for fluid-structure interaction-a review, *Comm. Comput. Phys.*, 12(2):337–377, 2012.

[60] J. Hron and S. Turek, A monolithic fem/multigrid solver for an ale formulation of fluid-structure interaction with applications in biomechanics, In: *Fluid-structure interaction*, 146–170, Springer, 2006.

[61] M. C. Hsu, D. Kamensky, Y. Bazilevs, M. S. Sacks, and T. J. R. Hughes, Fluid-structure interaction analysis of bioprosthetic heart valves: significance of arterial wall deformation, *Comput. Mech.*, 54(4):1055–1071, 2014.

[62] T. J. R. Hughes, *The Finite Element Method*, Prentice-Hall, 2000.

[63] T. J. R. Hughes, W. K. Liu, and T. K. Zimmerman, Lagrangian-Eulerian finite element formulation for incompressible viscous flows, *Comput. Methods Appl. Mech. Eng.*, 29(3):329–349, 1981.

[64] D. Jodlbauer, U. Langer, and T. Wick, Parallel block-preconditioned monolithic solvers for fluid-structure interaction problems, *Int. J. Num. Methods Eng.*, 117(6):623–643, 2019.

[65] E. Karabelas et al., Towards a computational framework for modeling the impact of aortic coarctations upon left ventricular load, *Front. Physiol.*, 9:538, 2018.

[66] A. Katz, *Physiology of the Heart*, M - Medicine Series, Wolters Kluwer Health/Lippincott Williams & Wilkins Health, 2010.

[67] R. C. P. Kerckhoffs et al., Coupling of a 3d finite element model of cardiac ventricular mechanics to lumped systems models of the systemic and pulmonic circulation, *Ann. Biomed. Eng.*, 35(1):1–18, 2007.

[68] U. Küttler, M. Gee, C. Förster, A. Comerford, and W. A. Wall, Coupling strategies for biomedical fluid-structure interaction problems, *Int. J. Numer. Method. Biomed. Eng.*, 26(3-4):305–321, 2010.

[69] U. Küttler and W. A. Wall, Fixed-point fluid-structure interaction solvers with dynamic relaxation, *Comput. Mech.*, 43(1):61–72, 2008.

[70] M. Landajuela, M. Vidrascu, D. Chapelle, and M. A. Fernández, Coupling schemes for the FSI forward prediction challenge: Comparative study and validation, *Int. J. Numer. Method. Biomed. Eng.*, 33(4):2813, 2017.

[71] U. Langer and H. Yang, Robust and efficient monolithic fluid-structure-interaction solvers, *Int. J. Num. Methods Eng.*, 108(4):303–325, 2016.

[72] J. Liu and A. L. Marsden, A unified continuum and variational multiscale formulation for fluids, solids, and fluid-structure interaction, *Comput. Methods Appl. Mech Eng.*, 337:549–597, 2018.

[73] G. Luraghi et al., Evaluation of an aortic valve prosthesis: Fluid-structure interaction or structural simulation? *J. Biomech.*, 58:45–51, 2017.

[74] M. Mehl, B. Uekermann, H. Bijl, D. Blom, B. Gatzhammer, and A. Van Zuijlen, Parallel coupling numerics for partitioned fluid-structure interaction simulations, *Comput. Math. Appl.*, 71(4):869–891, 2016.

[75] V. Meschini, R. Mittal, and R. Verzicco, Systolic anterior motion in hypertrophic cardiomyopathy: a fluid-structure interaction computational model, *Theor. Comput. Fluid Dyn.*, 35(3):381–396, 2021.

[76] D. P. Mok, W. A. Wall, and E. Ramm, Accelerated iterative substructuring schemes for instationary fluid-structure interaction, *Comp. Fluid Solid Mech.*, 2:1325–1328, 2001.

[77] C. M. Murea and S. Sy, A fast method for solving fluid-structure interaction problems numerically, *Int. J. Numer. Methods Fluids*, 60(10):1149–1172, 2009.

[78] F. Nobile, M. Pozzoli, and C. Vergara, Time accurate partitioned algorithms for the solution of fluid-structure interaction problems in haemodynamics, *Comput. Fluids*, 86:470–482, 2013.

[79] F. Nobile, M. Pozzoli, and C. Vergara, Inexact accurate partitioned algorithms for fluid–structure interaction problems with finite elasticity in haemodynamics, *J. Comput. Phys*, 273:598–617, 2014.

[80] D. Nordsletten, *Fluid-solid coupling for the simulation of left ventricular dynamics*, PhD Thesis, University of Oxford, 2009.

[81] D. Nordsletten, M. McCormick, P. J. Kilner, P. Hunter, D. Kay, and N. P. Smith, Fluid-solid coupling for the investigation of diastolic and systolic human left ventricular function, *Int. J. Numer. Method. Biomed. Eng.*, 27:1017–1039, 2011.

[82] R. W. Ogden, *Non-Linear Elastic Deformations*, Courier Corporation, 1997.

[83] M. R. Pfaller et al., The importance of the pericardium for cardiac biomechanics: from physiology to computational modeling, *Biomech. Model. Mechanobiol.*, 18(2):503–529, 2019.

[84] R. Piersanti, P. C. Africa, M. Fedele, C. Vergara, L. Dede', A. F. Corno, and A. Quarteroni, Modeling cardiac muscle fibers in ventricular and atrial electrophysiology simulations, *Comput. Methods Appl. Mech. Eng.*, 373:113468, 2021.

[85] A. Quaini and A. Quarteroni, A semi-implicit approach for fluid-structure interaction based on an algebraic fractional step, *Math. Models Methods Appl. Sci.*, 17:957–983, 2007.

[86] A. Quarteroni, *Numerical Models for Differential Problems*, Springer, 2017.

[87] A. Quarteroni, L. Dede', A. Manzoni, and C. Vergara, *Mathematical Modelling of the Human Cardiovascular System-Data, Numerical Approximation, Clinical Applications*, Cambridge University Press, 2019.

[88] A. Quarteroni, T. Lassila, S. Rossi, and R. Ruiz-Baier, Integrated Heart-Coupling multiscale and multiphysics models for the simulation of the cardiac function, *Comput. Methods Appl. Mech. Eng.*, 314:345–407, 2017.

[89] A. Quarteroni, A. Veneziani, and C. Vergara, Geometric multiscale modeling of the cardiovascular system, between theory and practice, *Comput. Methods Appl. Mech. Eng.*, 302:193–252, 2016.

[90] F. Regazzoni, L. Dede', and A. Quarteroni, Biophysically detailed mathematical models of multiscale cardiac active mechanics, *PLoS Comput. Biol.*, 16(10):1–42, 10 2020.

[91] F. Regazzoni, L. Dede', and A. Quarteroni, Machine learning of multiscale active force generation models for the efficient simulation of cardiac electromechanics, *Comput. Methods Appl. Mech. Eng.*, 370:113628, 10 2020.

[92] F. Regazzoni, M. Salvador, P. C. Africa, M. Fedele, L. Dede', and A. Quarteroni, A cardiac electromechanical model coupled with a lumped-parameter model for closed-loop blood circulation, *J. Comput. Phys.*, 457:111083, 2022.

[93] T. Richter, A monolithic geometric multigrid solver for fluid-structure interactions in ALE formulation, *Int. J. Num. Methods Eng.*, 104(5):372–390, 2015.

[94] T. Richter, *Fluid-Structure Interactions: Models, Analysis and Finite Elements*, Springer, 2017.

[95] S. Rossi, T. Lassila, R. Ruiz-Baier, A. Sequeira, and A. Quarteroni, Thermodynamically consistent orthotropic activation model capturing ventricular systolic wall thickening in cardiac electromechanics, *Eur. J. Mech. A Solids*, 48:129–142, 2014.

[96] Y. Saad, *Iterative Methods for Sparse Linear Systems*, SIAM, 2003.

[97] M. Salvador et al., Electromechanical modeling of human ventricles with ischemic cardiomyopathy: numerical simulations in sinus rhythm and under arrhythmia, *Comput. Biol. Med.*, 136:104674, 2021.

[98] M. Salvador, F. Regazzoni, S. Pagani, L. Dede', N. Trayanova, and A. Quarteroni, The role of mechano-electric feedbacks and hemodynamic coupling in scar-related ventricular tachycardia, arXiv:2110.15401, 2021.

[99] A. Santiago et al., Fully coupled fluid-electro-mechanical model of the human heart for supercomputers, *Int. J. Numer. Method. Biomed. Eng.*, 34:e3140, 2018.

[100] K. Stein, T. Tezduyar, and R. Benney, Mesh moving techniques for fluid-structure interac-

tions with large displacements, *J. Appl. Mech.*, 70(1):58–63, 2003.

[101] M. Strocchi et al., Simulating ventricular systolic motion in a four-chamber heart model with spatially varying Robin boundary conditions to model the effect of the pericardium, *J. Biomech*, 101:109645, 2020.

[102] S. Sugiura, T. Washio, A. Hatano, J. Okada, H. Watanabe, and T. Hisada, Multi-scale simulations of cardiac electrophysiology and mechanics using the university of tokyo heart simulator, *Prog. Biophys. Mol.*, 110(2-3):380–389, 2012.

[103] A. Tagliabue, L. Dede', and A. Quarteroni, Complex blood flow patterns in an idealized left ventricle: A numerical study, *Chaos*, 27(9):093939, 2017.

[104] T. Terahara, K. Takizawa, T. E. Tezduyar, Y. Bazilevs, and M. C. Hsu, Heart valve isogeometric sequentially-coupled FSI analysis with the space-time topology change method, *Comput. Mech.*, 1–21, 2020.

[105] T. Tezduyar and S. Sathe, Stabilization parameters in SUPG and PSPG formulations, *J. Comput. Appl. Mech.*, 4(1):71–88, 2003.

[106] A. This, L. Boilevin-Kayl, M. A. Fernández, and J. F. Gerbeau, Augmented resistive immersed surfaces valve model for the simulation of cardiac hemodynamics with isovolumetric phases, *Int. J. Numer. Method. Biomed. Eng.*, 36(3):1–26, 2020.

[107] T. Trilinos Project Team, *The Trilinos Project Website*, 2020 (acccessed May 22, 2020).

[108] T. Usyk, I. LeGrice, and A. D. McCulloch, Computational model of three-dimensional cardiac electromechanics, *Comput. Vis. Sci.*, 4:249–257, 2002.

[109] A. C. Verkaik, A. C. B. Bogaerds, F. Storti, and F. N. Van De Vosse, A coupled overlapping domain method for the computation of transitional flow through artificial heart valves, In: *ASME 2012 Summer Bioengineering Conference (SBC 2012)*, 217–218, ASME, 2012.

[110] I. E. Vignon-Clementel, C. A. Figueroa, K. E. Jansen, and C. A. Taylor, Outflow boundary conditions for 3D simulations of non-periodic blood flow and pressure fields in deformable arteries, *Comput. Methods Biomech. Biomed. Eng.*, 13(5):625–640, 2010.

[111] F. Viola, V. Meschini, and R. Verzicco, Fluid-Structure-Electrophysiology interaction (FSEI) in the left-heart: A multi-way coupled computational model, *Eur. J. Mech. B Fluids*, 79:212–232, 2020.

[112] H. F. Walker and P. Ni, Anderson acceleration for fixed-point iterations, *SIAM J. Numer. Anal.*, 49:1715–1735, 2011.

[113] H. Watanabe, S. Sugiura, H. Kafuku, and T. Hisada, Multiphysics simulation of left ventricular filling dynamics using fluid-structure interaction finite element method, *Biophys. J.*, 87(3):2074–2085, 2004.

[114] T. Wick, Solving monolithic fluid-structure interaction problems in arbitrary Lagrangian-Eulerian coordinates with the deal. ii library, *Arch. Num. Soft.*, 1(1):1–19, 2013.

[115] Q. Zhang and T. Hisada, Analysis of fluid-structure interaction problems with structural buckling and large domain changes by ALE finite element method, *Comput. Methods Appl. Mech. Eng.*, 190:6341–6357, 2001.

[116] A. Zingaro, L. Dede', F. Menghini, and A. Quarteroni, Hemodynamics of the heart's left atrium based on a Variational Multiscale-LES numerical method, *Eur. J. Mech. B Fluids*, 89:380–400, 2021.

[117] A. Zingaro, I. Fumagalli, L. Dede', M. Fedele, P. C. Africa, A. F. Corno, and A. Quarteroni, A geometric multiscale model for the numerical simulation of blood flow in the human left heart, *Discrete & Continuous Dynamical Systems-S*, 2022.