Heterogeneous Domain Decomposition Methods for Fluid-Structure Interaction Problems

S. Deparis[1], M. Discacciati[2], G. Fourestey[2], and A. Quarteroni[2,3]

[1] Mechanical Engineering Department MIT, 77 Mass Ave, Cambridge MA 02139, USA, deparis@mit.edu
[2] IACS - Chair of Modeling and Scientific Computing, EPFL, CH-1015 Lausanne, Switzerland, marco.discacciati@oeaw.ac.at, gilles.fourestey@epfl.ch
[3] MOX, Politecnico di Milano, P.zza Leonardo da Vinci 32, 20133 Milano, Italy alfio.quarteroni@epfl.ch

Summary. In this note, we propose Steklov-Poincaré iterative algorithms (mutuated from the analogy with heterogeneous domain decomposition) to solve fluid-structure interaction problems. Although our framework is very general, the driving application is concerned with the interaction of blood flow and vessel wall in large arteries.

1 Introduction

Mathematical modeling of real-life problems may lead to different kind of boundary value problems in different subregions of the original computational domain. The reason may be twofold.

Often, in order to reduce the computational cost of the simulation, a very detailed model can be used only in a region of specific interest while recurring to a simplified version of the same model sufficiently far away from where the most relevant physical phenomena occur. This is, e.g., the strategy adopted when one considers the coupling of advection-diffusion equations with advection equations, after neglecting the diffusive effects in a certain subregion (see, e.g., [11]), or when the full Navier-Stokes equations are coupled with Oseen, Stokes or even velocity potential models, the latter being adopted where the nonlinear convective effects are negligible (see, e.g., [7; 8]).

In a second circumstance, one may be obliged to consider truly different models to account for the presence of distinct physical problems within the same global domain. This case is usually indicated as multi-physics or multi-field problem.

Typical examples are given by filtration processes such as in biomechanics or in environmental applications where a fluid (e.g. blood or water) can filtrate

through a porous medium (e.g. the arterial wall or the soil), so that the Navier-Stokes equations must be coupled with Darcy's (or more complicated models, e.g., Forchheimer or Brinkmann equations) to describe the underlying physics (see, e.g., [5; 10; 16; 24]).

All these problems may be casted into the same common framework provided by the so-called heterogeneous domain decomposition method, which extends the classical domain decomposition theory whenever two (or more) kinds of boundary value problems, say $L_i u_i = f_i$, hold in subregions Ω_i of the computational domain Ω.

A major role is played by the compatibility conditions that the unknowns u_i must satisfy across the interface which separates the subdomains. In fact, the setting of proper coupling conditions is a crucial issue to model as closely as possible the real physical phenomena. For example, when coupling the Navier-Stokes and the Oseen equations, the compatibility conditions require the continuity of the velocities and of the normal stresses across the interface. However, it is worth mentioning that they might be much less intuitive and easy to handle than in the case we have just mentioned (see, e.g., [5; 25]).

In this paper, we will apply the heterogeneous domain decomposition paradigm to a fluid-structure interaction problem arising in hemodynamics for modeling blood flows in large arteries. To preserve stability one should solve exactly the fluid-structure coupling, e.g. by Newton methods [9; 13] or fixed-point algorithms [2]. A Newton method with exact Jacobian has been investigated both mathematically and numerically in [9]. Segregated solvers yielding a single fluid-structure interaction per time step do not preserve stability and may produce blowing up when the density of the structure stays below a critical threshold. On the other hand, to relax the computational complexity of fixed-point or Newton methods several inexact solution strategies can be adopted.

The Jacobian matrix can be simplified by dropping the cross block expressing the sensitivity of the fluid state to solid motion, or by replacing it by a simpler term that models the so-called added-mass effect (see [1; 12]). Alternative inexact solvers exploit the analogy of the fluid structure coupled problem with heterogeneous domain decomposition problems.

This approach was firstly presented in [17; 20] for a Stokes-linearized shell coupling and later studied also in [19], where the whole problem was first reformulated as an interface equation. In this paper we further pursue this approach. Iterative substructuring methods, typical of the domain decomposition approach, are used to solve the interface problem, exploiting the classical Dirichlet-Neumann, the Neumann-Neumann, or more sophisticated scaling (preconditioning) techniques.

After describing a precise setting of the problem (Sect. 2), we shall define the associated interface equation (Sect. 3) and illustrate possible iterative methods to solve it (Sect. 4). Finally, some numerical results will be presented (Sect. 5).

2 Problem setting

To describe the evolution of the fluid and the structure domains in time, we adopt the ALE (Arbitrary Lagrangian Eulerian) formulation for the fluid (see [6; 14]) and a purely Lagrangian framework for the structure. We denote by $\Omega(t)$ the moving domain composed of the deformable structure $\Omega^{\mathrm{s}}(t)$ and the fluid subdomain $\Omega^{\mathrm{f}}(t)$. If we denote by $\boldsymbol{d}^{\mathrm{s}}(x_0,t)$ the displacement of the solid

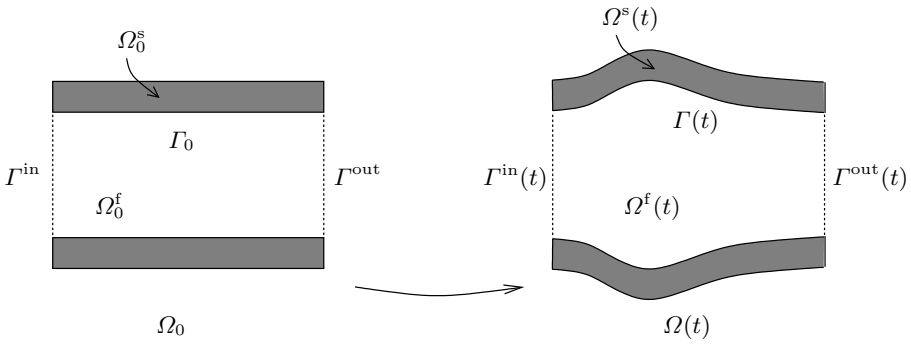

Fig. 1. ALE mapping

at a time t, we can define the following mapping: $\forall t$, $\Omega_0^{\mathrm{s}} \to \Omega^{\mathrm{s}}(t)$,

$$x_0 \to \boldsymbol{x}_t^{\mathrm{s}}(x_0) = x_0 + \boldsymbol{d}^{\mathrm{s}}(x_0,t), \quad x_0 \in \Omega_0^{\mathrm{s}}. \tag{1}$$

Likewise, for the fluid domain: $\forall t$, $\Omega_0^{\mathrm{f}} \to \Omega^{\mathrm{f}}(t)$,

$$x_0 \to \boldsymbol{x}_t^{\mathrm{f}}(x_0) = x_0 + \boldsymbol{d}^{\mathrm{f}}(x_0,t), \quad x_0 \in \Omega_0^{\mathrm{f}}. \tag{2}$$

The fluid domain displacement $\boldsymbol{d}^{\mathrm{f}}$ can be defined as a suitable extension of the solid interface displacement $\boldsymbol{d}^{\mathrm{s}}_{|\Gamma_0}$: $\boldsymbol{d}^{\mathrm{f}} = Ext(\boldsymbol{d}^{\mathrm{s}}_{|\Gamma_0})$ (see, e.g., [21]).

We assume the fluid to be Newtonian, viscous and incompressible, so that its behavior is described by the following fluid state problem: given the boundary data $\boldsymbol{u}_{\mathrm{in}}$, $\boldsymbol{g}_{\mathrm{f}}$, and the forcing term $\boldsymbol{f}_{\mathrm{f}}$, and denoting $\boldsymbol{w}^{\mathrm{f}} = \partial_t \boldsymbol{d}^{\mathrm{f}}$ the rate of change of the fluid domain, the velocity field $\boldsymbol{u}$ and the pressure p satisfy the momentum and continuity equations:

$$\begin{aligned} \rho_{\mathrm{f}} \left(\left.\frac{\partial \boldsymbol{u}}{\partial t}\right|_{x_0} + (\boldsymbol{u} - \boldsymbol{w}^{\mathrm{f}}) \cdot \boldsymbol{\nabla} \boldsymbol{u} \right) - \mathrm{div}[\boldsymbol{\sigma}_{\mathrm{f}}(\boldsymbol{u},p)] &= \boldsymbol{f}_{\mathrm{f}} \ \text{ in } \Omega^{\mathrm{f}}(t), \\ \mathrm{div}\, \boldsymbol{u} &= 0 \quad \text{in } \Omega^{\mathrm{f}}(t), \\ \boldsymbol{u} = \boldsymbol{u}_{\mathrm{in}} \ \text{ on } \Gamma^{\mathrm{in}}(t), \quad \boldsymbol{\sigma}_{\mathrm{f}}(\boldsymbol{u},p) \cdot \boldsymbol{n}_{\mathrm{f}} &= \boldsymbol{g}_{\mathrm{f}} \ \text{ on } \Gamma^{\mathrm{out}}(t). \end{aligned} \tag{3}$$

We denote by ρ_{f} the fluid density, μ the fluid viscosity, $\boldsymbol{\sigma}_{\mathrm{f}}(\boldsymbol{u},p) = -pId + 2\mu\boldsymbol{\epsilon}(\boldsymbol{u})$ the Cauchy stress tensor, Id is the identity matrix, $\boldsymbol{\epsilon}(\boldsymbol{u}) = (\boldsymbol{\nabla}\boldsymbol{u} +$

$(\boldsymbol{\nabla u})^T)/2$ the strain rate tensor. Note that (3) does not define univocally a solution $(\boldsymbol{u}, p)$ as no boundary data are prescribed on the interface $\Gamma(t)$.

Similarly, for given vector functions $\boldsymbol{g}_\mathrm{s}$, $\boldsymbol{f}_\mathrm{s}$, we consider the following structure problem whose solution is $\boldsymbol{d}^\mathrm{s}$:

$$\begin{aligned} \rho_\mathrm{s}\frac{\partial^2 \boldsymbol{d}^\mathrm{s}}{\partial t^2} - \mathrm{div}_{|x_0}(\boldsymbol{\sigma}_\mathrm{s}(\boldsymbol{d}^\mathrm{s})) &= \boldsymbol{f}_\mathrm{s} \ \text{ in } \Omega_0^\mathrm{s}, \\ \boldsymbol{\sigma}_\mathrm{s}(\boldsymbol{d}^\mathrm{s})\cdot\boldsymbol{n}_\mathrm{s} &= \boldsymbol{g}_\mathrm{s} \ \text{ on } \partial\Omega_0^\mathrm{s}\setminus\Gamma_0, \end{aligned} \tag{4}$$

where $\boldsymbol{\sigma}_\mathrm{s}(\boldsymbol{d}^\mathrm{s})$ is the first Piola–Kirchoff stress tensor. Remark that boundary values on Γ_0 for (4) are missing.

When coupling the two problems together, the "missing" boundary conditions are indeed supplemented by suitable matching conditions on the reference interface Γ_0. If $\lambda = \lambda(t)$ denotes the displacement of the interface, at any time t the coupling conditions on the reference interface Γ_0 are

$$\begin{aligned} \boldsymbol{x}_t^\mathrm{s} = x_0 + \lambda = \boldsymbol{x}_t^\mathrm{f}, \qquad \boldsymbol{u}\circ\boldsymbol{x}_t^\mathrm{f} = \frac{\partial\lambda}{\partial t}, \\ (\boldsymbol{\sigma}_\mathrm{f}(\boldsymbol{u},p)\cdot\boldsymbol{n}_\mathrm{f})\circ\boldsymbol{x}_t^\mathrm{f} = -\boldsymbol{\sigma}_\mathrm{s}(\boldsymbol{d}^\mathrm{s})\cdot\boldsymbol{n}_\mathrm{s}, \end{aligned} \tag{5}$$

imposing the matching of the interface displacements from the fluid and solid subdomains, the continuity of the velocities and of the normal stresses.

3 The interface equations associated to problem (3)-(5)

We consider the coupled problem at a given time $t = t^{n+1} = (n+1)\delta t$, being δt the discrete time-step.

According to the interface conditions (5), we can envisage two possible natural choices for the interface variable: either we consider the displacement λ of the fluid-structure interface, or the normal stress exerted on it. In the following, we shall focus our attention on the case of the interface variable as the displacement; the "dual" approach using the normal stress was presented in [4] on a simple linear problem.

Thus, we define the fluid and structure interface operators as follows.

S_f is the *Dirichlet-to-Neumann map* in $\Omega^\mathrm{f}(t)$:

$$S_\mathrm{f} : H^{1/2}(\Gamma_0) \to H^{-1/2}(\Gamma_0), \qquad \lambda \to \sigma_\mathrm{f}(\lambda),$$

that operates between the trace space of displacements on the interface Γ_0 and the dual space of the normal stresses exerted on Γ_0 by the fluid. Computing $S_\mathrm{f}(\lambda)$ involves the extension of the interface displacement to the whole fluid domain (in order to compute the ALE velocity), the solution of a Navier-Stokes problem in $\Omega^\mathrm{f}(t)$ with the Dirichlet boundary condition on the interface $\boldsymbol{u}_{|\Gamma(t)}\circ\boldsymbol{x}_t^\mathrm{f} = (\lambda - \boldsymbol{d}_{|\Gamma_0}^{\mathrm{s},n})/\delta t$, and then to recover the normal stress $\sigma_\mathrm{f} = (\boldsymbol{\sigma}_\mathrm{f}(\boldsymbol{u},p)\cdot\boldsymbol{n}_\mathrm{f})_{|\Gamma(t)}\circ\boldsymbol{x}_t^\mathrm{f}$ as a residual of the Navier-Stokes equations on the interface.

Moreover, we consider the *Dirichlet-to-Neumann map* S_s in Ω_0^s:

$$S_\mathrm{s} : H^{1/2}(\Gamma_0) \to H^{-1/2}(\Gamma_0), \qquad \lambda \to \sigma_\mathrm{s}(\lambda),$$

that operates between the space of displacements on the interface Γ_0 and the space of the normal stresses exerted by the structure on Γ_0. Computing $S_\mathrm{s}(\lambda)$ corresponds to solve a structure problem in Ω_0^s with Dirichlet boundary condition $\boldsymbol{d}^\mathrm{s}_{|\Gamma_0} = \lambda$ on Γ_0, and then to recover the normal stress $\sigma_\mathrm{s} = \boldsymbol{\sigma}_\mathrm{s}(\boldsymbol{d}^\mathrm{s})\cdot\boldsymbol{n}_\mathrm{s}$ on the interface, again as a residual.

The definitions of S_f and S_s involve also the boundary and forcing terms, because of the nonlinearity of the problem at hand.

Then, the coupled fluid-structure problem can be expressed in terms of the solution λ of the following nonlinear Steklov-Poincaré interface problem:

$$\text{find } \lambda \in H^{1/2}(\Gamma_0): \quad S_\mathrm{f}(\lambda) + S_\mathrm{s}(\lambda) = 0. \tag{6}$$

Remark 1. In the case of a linear coupled Stokes-shell model, Mouro [20] has given a precise characterization of these interface operators and shown that they are selfadjoint and positive.

The inverse operator S_s^{-1} is a Neumann-to-Dirichlet map that at any given normal stress σ on Γ_0 associates the interface displacement $\lambda(t^{n+1}) = \boldsymbol{d}^{\mathrm{s},n+1}$ by solving a structure problem with the Neumann boundary condition $\boldsymbol{\sigma}_\mathrm{s}(\boldsymbol{d}^s)\cdot \boldsymbol{n}_\mathrm{s} = \sigma$ on Γ_0 and then computing the restriction on Γ_0 of the displacement of the structure domain.

For nonlinear structural models (i.e. $\boldsymbol{\sigma}_\mathrm{s}(\boldsymbol{d}^s)$ is a nonlinear constitutive law in (4), see, e.g., [18]), we will need the *tangent* operator S_s'

$$S_\mathrm{s}'(\bar\lambda)\delta\lambda = \lim_{h\to 0} \frac{S_\mathrm{s}(\bar\lambda + h\delta\lambda) - S_\mathrm{s}(\bar\lambda)}{h}, \qquad \forall \bar\lambda, \delta\lambda \in H^{1/2}(\Gamma_0).$$

Its inverse $(S_\mathrm{s}')^{-1}$ is a Neumann-to-Dirichlet map that to any given variation of the normal stress $\delta\sigma$ on Γ_0 associates the corresponding variation of the displacement $\delta\lambda$ of the interface by solving a linearized structure problem with boundary condition $\boldsymbol{\sigma}_\mathrm{s}(\boldsymbol{d}^\mathrm{s})\cdot\boldsymbol{n}_\mathrm{s} = \delta\sigma$ on Γ_0. Similarly, we define S_f' by

$$S_\mathrm{f}'(\bar\lambda)\delta\lambda = \lim_{h\to 0} \frac{S_\mathrm{f}(\bar\lambda + h\delta\lambda) - S_\mathrm{f}(\bar\lambda)}{h}, \qquad \forall \bar\lambda, \delta\lambda \in H^{1/2}(\Gamma_0).$$

This is a Dirichlet-to-Neumann map that for any variation of the interface displacement $\delta\lambda$ computes the corresponding variation of the normal stress $\delta\sigma$ on Γ_0 through the solution of linearized Navier-Stokes equations. To compute $S_\mathrm{f}'(\lambda)\delta\lambda$ see, e.g, [9].

The computation of the inverse operator $S_\mathrm{f}'(\lambda)^{-1}$ can be simplified by neglecting the shape derivatives. We then obtain the Oseen equations in the fixed configuration defined by λ that we computed while evaluating $S_\mathrm{f}(\lambda)$. $S_\mathrm{f}'(\lambda)^{-1}$ is a Neumann-to-Dirichlet map that for any given variation of the

normal stress $\delta\sigma$ on Γ_0 computes the corresponding displacement $\delta\lambda$ of the interface through the solution of linearized Navier-Stokes equations with the boundary condition $(\boldsymbol{\sigma}_{\mathrm{f}}(\boldsymbol{u},p)\cdot\boldsymbol{n}_{\mathrm{f}})\circ\boldsymbol{x}^{\mathrm{f}}=\sigma$ on Γ_0.

Other possible formulations for the interface equation can be given:

$$\text{find } \lambda \text{ such that } S_{\mathrm{s}}^{-1}(-S_{\mathrm{f}}(\lambda))=\lambda \text{ on } \Gamma_0, \tag{7}$$

$$\text{or, equivalently, find } \lambda \text{ such that } S_{\mathrm{s}}^{-1}(-S_{\mathrm{f}}(\lambda))-\lambda=0 \text{ on } \Gamma_0. \tag{8}$$

These are common formulations in fluid-structure interaction problems, but it is worth pointing out that here the unknown λ is the displacement of the sole interface, whereas classically the displacement of the whole solid domain is considered (see, e.g., [21; 9]).

4 Iterative methods for problems (6)-(8)

We consider the preconditioned Richardson method to solve the Steklov-Poincaré interface problem (6): given λ^0, for $k\geq 0$, solve

$$P_k\left(\lambda^{k+1}-\lambda^k\right)=\omega^k\left(-S_{\mathrm{f}}(\lambda^k)-S_{\mathrm{s}}(\lambda^k)\right). \tag{9}$$

The scaling operator P_k maps the space $H^{1/2}(\Gamma_0)$ of the interface variable onto the space $H^{-1/2}(\Gamma_0)$ of normal stresses, and may depend on the iterate λ^k or, more generally, on the iteration step k. The acceleration parameter ω^k can be computed via the Aitken technique (see [4]) or by line search (see [23]).

At each step k, (9) requires to solve separately the fluid and the structure problems and then to apply a scaling operator. Precisely,

1. apply S_{f} to λ^k, i.e., compute the extension of λ^k to the entire fluid domain to obtain the ALE velocity, and solve the fluid problem in $\Omega^{\mathrm{f}}(t)$ with boundary condition $\boldsymbol{u}_{|\Gamma(t)}\circ\boldsymbol{x}_t^{\mathrm{f}}=(\lambda-\boldsymbol{d}^{\mathrm{s},n}_{|\Gamma_0})/\delta t$ on Γ_0; then, recover the normal stress σ_{f}^k on the interface;
2. apply S_{s} to λ^k, i.e., solve the structure problem with boundary condition $\boldsymbol{d}^{\mathrm{s},k}_{|\Gamma(t)}=\lambda^k$ on $\Gamma(t)$ and compute the normal stress σ_{s}^k;
3. apply P_k^{-1} to the total stress $\sigma^k=\sigma_{\mathrm{f}}^k+\sigma_{\mathrm{s}}^k$ on the interface.

Note that steps 1. and 2. can be performed in parallel. The crucial issue is how to choose the scaling operator (more precisely, a preconditioner in the finite dimensional case) in order for the iterative method to converge as quickly as possible.

We define a generic linear operator (more precisely, its inverse):

$$P_k^{-1}=\alpha_{\mathrm{f}}^k\,S_{\mathrm{f}}'(\lambda^k)^{-1}+\alpha_{\mathrm{s}}^k\,S_{\mathrm{s}}'(\lambda^k)^{-1}, \tag{10}$$

for two given scalars α_{f}^k and α_{s}^k, and we retrieve the following operators:

Dirichlet-Neumann (DN): $P_k = P_{DN} = S'_{\rm s}(\lambda^k)$, for $\alpha_{\rm f}^k = 0$, $\alpha_{\rm s}^k = 1$, (11)

Neumann-Dirichlet (ND): $P_k = P_{ND} = S'_{\rm f}(\lambda^k)$, for $\alpha_{\rm f}^k = 1$, $\alpha_{\rm s}^k = 0$, (12)

Neumann-Neumann (NN): $P_k = P_{NN}$ with $\alpha_{\rm f}^k + \alpha_{\rm s}^k = 1$, $\alpha_{\rm f}^k, \alpha_{\rm s}^k \neq 0$. (13)

If the structure is linear, in the DN case the computational effort of a Richardson step may be reduced to the solution of only one fluid Dirichlet problem and one structure Neumann problem.

The parameters $\alpha_{\rm f}^k$, $\alpha_{\rm s}^k$ and ω^k can be chosen dynamically using a generalized Aitken technique (see [3; 4]).

Should we consider the scaling operator

$$P_k = S'_{\rm f}(\lambda^k) + S'_{\rm s}(\lambda^k), \tag{14}$$

then, we would retrieve the genuine Newton algorithm applied to the Steklov-Poincaré problem (6). Note that in order to perform the scaling step 3. in the Richardson algorithm, one must use a (preconditioned) iterative method (e.g., GMRES) and may approximate the tangent problems to accelerate the computations. Thus, using the scaling operator (14) we obtain a *domain decomposition-Newton (DD-Newton)* method; more precisely, given a solid state displacement λ^k, for $k \geq 0$, the algorithm reads

1. solve the fluid and the structure subproblems separately, as for the Richardson method, to get σ^k;
2. solve the following linear system via GMRES to compute μ^k:

$$\left[S'_{\rm f}(\lambda^k) + S'_{\rm s}(\lambda^k)\right] \mu^k = -(S_{\rm f}(\lambda^k) + S_{\rm s}(\lambda^k)) \tag{15}$$

3. update the displacement: $\lambda^{k+1} = \lambda^k + \omega^k \mu^k$.

The GMRES solver should in its turn be preconditioned in order to accelerate its convergence rate. To this aim, one can use one of the previously defined scaling operators. In our numerical tests, we have considered the DN operator $S'_{\rm s}(\lambda)$, so that the preconditioned matrix of the GMRES method becomes:

$$[S'_{\rm s}(\lambda^k)]^{-1} \cdot [S'_{\rm f}(\lambda_k) + S'_{\rm s}(\lambda_k)]. \tag{16}$$

Let us briefly recall the Newton method for problem (8) in order to compare it with the previous domain decomposition approach. For a more complete discussion we refer to [4].

Let $J(\lambda)$ denote the Jacobian of $S_{\rm s}^{-1}(-S_{\rm f}(\lambda))$ in λ. Given λ^0, for $k \geq 0$:

$$\begin{array}{ll} \text{solve} & (J(\lambda^k) - Id)\mu^k = -(S_{\rm s}^{-1}(-S_{\rm f}(\lambda^k)) - \lambda^k), \\ \text{update} & \lambda^{k+1} = \lambda^k + \omega^k \mu^k. \end{array} \tag{17}$$

The parameter ω^k can be computed, e.g., by a line search technique (see [23]). Note that the Jacobian in λ^k has the following expression:

$$J(\lambda^k) = -\left[S'_{\rm s}\left(S_{\rm s}^{-1}(-S_{\rm f}(\lambda^k))\right)\right]^{-1} \cdot S'_{\rm f}(\lambda^k) = -\left[S'_{\rm s}\left(\bar{\lambda}^k\right)\right]^{-1} \cdot S'_{\rm f}(\lambda^k). \tag{18}$$

The solution of the linear system (17) can be obtained using an iterative matrix-free method such as GMRES.

In general, the Newton method applied to (8) and to the Steklov-Poincaré formulation (6) are not equivalent. However, in the case of a linear structure, they actually are (to see this, left multiply by S_{s}^{-1} both hand sides in (15), exploit $S_{\mathrm{s}}'(\lambda^k) = S_{\mathrm{s}}$ and compare (16) and (17)).

We remark that while the computation of $\left[S_{\mathrm{s}}'\left(\bar{\lambda}^k\right)\right]^{-1} \cdot \delta\sigma$ (for any given $\delta\sigma$) does only require the derivative with respect to the state variable at the interface, the computation of $S_{\mathrm{f}}'(\lambda^k)\cdot\delta\lambda$ is nontrivial since it implies also shape derivatives, as a variation in λ determines a variation of the fluid domain.

Finally, remark that in the classical Newton method, the fluid and structure problems must be solved separately and sequentially, while the domain decomposition formulation allows us to set up parallel algorithms to solve the Steklov-Poincaré equation (6).

5 Numerical results

In this section, we present some numerical results which compare the domain decomposition methods to the classical fixed point and Newton ones, and illustrate their behavior with respect to the grid size h and the time step δt.

For the domain decomposition algorithms, we consider the DN preconditioner (11), and the NN preconditioner (13) in which S_{f}' is linearized by neglecting the shape derivatives.

Finally, we consider the DD-Newton method (14). The fluid tangent problem is considered as in [9] in its exact form. To solve (15), we apply the GMRES method possibly preconditioned by the operator DN (11).

Both problems (3) and (4) are discretized, and we adopt $\mathbb{P}_1$-bubble/$\mathbb{P}_1$ finite elements for the fluid and $\mathbb{P}_1$ elements for the structure. The simulations are performed on a dual 2.8 Ghz Pentium 4 Xeon with 3 GB of RAM.

We simulate a pressure wave in a straight cylinder of length 5 cm and radius 5 mm at rest. The structure of thickness 0.5 mm is linear and clamped at both the inlet and the outlet. The fluid viscosity is set to $\mu = 0.03$ $poise$, the densities to $\rho_{\mathrm{f}} = 1\ g/cm^3$ and $\rho_{\mathrm{s}} = 1.2\ g/cm^3$. We impose zero body forces and homogeneous Dirichlet boundary conditions on $\partial\Omega_0^{\mathrm{s}} \setminus \Gamma_0$. The fluid and the structure are initially at rest and a pressure (a normal stress, actually) of $1.3332 \cdot 10^4\ dynes/cm^2$ is imposed on the inlet for $3 \cdot 10^{-3}$ s. We consider two computational meshes: a coarse one with 1050 nodes (4680 elements) for the fluid and 1260 nodes (4800 elements) for the solid, and a finer mesh with 2860 nodes (14100 elements) for the fluid and 2340 nodes (9000 elements) for the solid.

A comparison between the fixed point iterations on problem (7) and Richardson iterations (9) (with DN and NN preconditioners) on problem (6) is shown in table 1 for two time steps and for the coarse and the fine mesh. In

this table, "FS eval" stands for the average number of evaluations per time step of either (7) or (9), while "FS' eval" represents the average number of evaluations of the corresponding linearized system per time step (that is (10) for DN, ND or NN preconditioners, (16) for the DD-Newton method (15), and (18) for the classical Newton method (17)). We can see that, using the preconditioned Richardson method (9), fewer FS evaluations than with the classical fixed point algorithm are needed. However, the computational time of the domain decomposition formulation is slightly higher than that of the fixed point formulation. The reason is that the domain decomposition formulation requires to solve, at each iteration, the fluid and the structure subproblems, as well as the associated tangent problems, while the latter are indeed skipped by the fixed point procedure. Furthermore, since the operator for the structure is linear, the two approaches are very similar and since our research code is sequential, the parallel structure of the Steklov-Poincaré formulation (6) is not capitalized.

Moreover, we notice that using the NN preconditioner the number of iterations required for the convergence with respect to both parameters h and δt, does not vary sensibly.

Table 1. Comparison of the number of sub-iterations and computational time for the fixed point, and domain decomposition based algorithms for the coarse mesh (left) and fine mesh (right)

$\delta t = 0.001$			
Method	FS eval	FS' eval	CPU time
Fixed point	19.8	0	1h16'
DN	19.8	19.8	1h17'
NN	17.9	17.9	1h42'
Newton	3	12	0h56'
DD-Newton	3	24	1h30'
DD-Newton DN	3	12	0h58'
$\delta t = 0.0005$			
Method	FS eval	FS' eval	CPU time
Fixed point	32.1	0	3h27'
DN	29.2	29.2	3h50'
NN	22	22	4h20'
Newton	3	17	1h55'
DD-Newton	3	29	3h30'
DD-Newton DN	3	17	2h10'
$\delta t = 0.0001$			
Method	FS eval	FS' eval	CPU time
Newton	3	19	11h41'
DD-Newton	3	35	16h21'
DD-Newton DN	3	19	12h39'

$\delta t = 0.001$			
Method	FS eval	FS' eval	CPU time
Fixed point	19.9	0	4h28'
DN	19.5	19.5	4h40'
NN	17.7	17.7	6h12'
Newton	3	12	3h39'
DD-Newton	3	30	4h56'
DD-Newton DN	3	12	3h45'
$\delta t = 0.0005$			
Method	FS eval	FS' eval	CPU time
Fixed point	33	0	12h40'
DN	29.6	29.6	12h50'
NN	22.1	22.1	15h44'
Newton	3	14	8h31'
DD-Newton	3	35	10h50'
DD-Newton DN	3	14	8h40'
$\delta t = 0.0001$			
Method	FS eval	FS' eval	CPU time
Newton	3	19	26h40'
DD-Newton	3	37	40h26'
DD-Newton DN	3	19	27h01'

The same table shows also the results obtained using the Newton and DD-Newton methods. The Jacobian matrices (14) and (18) have been computed exactly (see [9]) and inverted by GMRES. The number of iterations between Newton and DD-Newton is equivalent, but the inversion of the Jacobian in DD-Newton ("FS' eval") needs more GMRES iterations, whose number depends on h and δt. However, preconditioning GMRES by DN reduces these iterations to the same number as in Newton, and the CPU time is then equivalent. As before, the reasons reside in the linearity of the structure model and in the fact that our code is sequential.

Further improvements may be obtained recurring to more sophisticated preconditioners for the Jacobian system, derived either from the classical domain decomposition theory or from lower dimensional models (in a multiscale approach, see [22]).

Now, we simulate a pressure wave in the carotid bifurcation using the same fluid and structure characteristics as before. We solve the coupling using our DD-Newton algorithm with DN preconditioner for the GMRES inner iterations. The mesh that we have used was computed using an original realistic geometry first proposed in [15].

The fluid and the structure are initially at rest and a pressure of $1.3332 \cdot 10^4$ $dynes/cm^2$ is set on the inlet for a time of $3 \cdot 10^{-3}$ s. The average inflow diameter is 0.67 cm, the time step used is $\delta t = 1e-04$ and the total number of iterations is 200. Figure 2 displays the solution computed at two different time steps. Table 2 shows the comparison between the classical Newton algorithm and our DD-Newton algorithm preconditioned by DN. Like in the previous test, "FS eval" and "FS' eval" represent respectively the average number of fluid/structure evaluations and the average number of linearized fluid/structure evaluations. As expected, both methods behave in the same way with respect to the number of operators evaluations. The total computation times are also in very good agreement for the two largest time step.

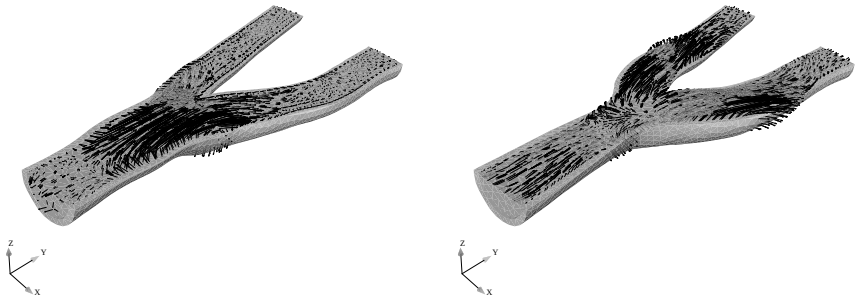

Fig. 2. Structure deformation and fluid velocity at $t = 0.005$ s (left) and $t = 0.008$ s (right)

Table 2. Convergence comparison of the computational time for the exact Newton and DD-Newton methods (case of carotid bifurcation)

	$\delta t = 0.001$			$\delta t = 0.0005$			$\delta t = 0.0001$		
Method	FS eval	FS' eval	CPU time	FS eval	FS' eval	CPU time	FS eval	FS' eval	CPU time
Newton	3	7.5	8h51'	3	10	19h41'	3	19	125h20'
DD-Newton DN	3	7.5	8h12'	3	10	19h33'	3	19	131h08'

Acknowledgement. This research has been supported by the Swiss National Science Foundation (project 20-101-800) and by the INDAM project "Integrazione di sistemi complessi in biomedicina: modelli, simulazioni, rappresentazioni".

References

[1] P. Causin, J.F. Gerbeau, and F. Nobile. Added-mass effect in the design of partitioned algorithms for fluid-structure problems. Technical Report 5084, INRIA, 2004. Accepted on Comp. Meth. Appl. Mech. Engrg.

[2] M. Cervera, R. Codina, and M. Galindo. On the computational efficiency and implementation of block-iterative algorithms for nonlinear coupled problems. *Engrg. Comput.*, 13(6):4–30, 1996.

[3] S. Deparis. *Numerical Analysis of Axisymmetric Flows and Methods for Fluid-Structure Interaction Arising in Blood Flow Simulation.* PhD thesis, EPFL, 2004.

[4] S. Deparis, M. Discacciati, and A. Quarteroni. A domain decomposition framework for fluid-structure interaction problems. In *Proceedings of the Third International Conference on Computational Fluid Dynamics (ICCFD3), Toronto*, 2004. Submitted.

[5] M. Discacciati. *Domain Decomposition Methods for the Coupling of Surface and Groundwater Flows.* PhD thesis, EPFL, 2004.

[6] J. Donea. Arbitrary Lagrangian Eulerian methods. In *Computational Methods for Transient Analysis*, volume 1 of *Computational Methods in Mechanics*. North-Holland, Elsevier, 1983.

[7] L. Fatone, P. Gervasio, and A. Quarteroni. Multimodels for incompressible flows. *J. Math. Fluid Mech.*, 2(2):126–150, 2000.

[8] L. Fatone, P. Gervasio, and A. Quarteroni. Multimodels for incompressible flows: iterative solutions for the Navier-Stokes/Oseen coupling. *M2AN Math. Model. Numer. Anal.*, 35(3):549–574, 2001.

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

[10] J.C. Galvis and M. Sarkis. Inf-sup for coupling Stokes-Darcy. In *Proceedings of the XXV Iberian Latin American Congress in Computational Methods in Engineering*, 2004.

[11] F. Gastaldi, A. Quarteroni, and G. Sacchi Landriani. On the coupling of two dimensional hyperbolic and elliptic equations: analytical and numerical approach. In T.F. Chan et al., editor, *Third International Symposium on Domain Decomposition Methods for Partial Differential Equations*, pages 22–63. SIAM, Philadephia, 1990.

[12] J.F. Gerbeau and M. Vidrascu. A quasi-Newton algorithm based on a reduced model for fluid-structure interaction problems in blood flows. *M2AN Math. Model. Numer. Anal.*, 37(4):663–680, 2003.

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

[14] T.J.R. Hughes, W.K. Liu, and T.K. Zimmermann. Lagrangian-Eulerian finite element formulation for incompressible flows. *Comput. Methods Appl. Mech. Engrg.*, 29:329–349, 1981.

[15] G. Karner, K. Perktold, M. Hofer, and D. Liepsch. Flow characteristics in an anatomically realistic compliant carotid artery bifurcation model. *Comput. Methods Biomech. and Biomedical Engrg.*, 2:171–185, 1999.

[16] W.L. Layton, F. Schieweck, and I. Yotov. Coupling fluid flow with porous media flow. *SIAM J. Num. Anal.*, 40:2195–2218, 2003.

[17] P. Le Tallec and J. Mouro. Fluid structure interaction with large structural displacements. *Comput. Methods Appl. Mech. Engrg.*, 190:3039–3067, 2001.

[18] J.E. Marsden and T.J.R. Hughes. *Mathematical Foundations of Elasticity*. Dover Publications Inc., New York, 1994. Reprint.

[19] D. P. Mok and W. A. Wall. Partitioned analysis schemes for the transient interaction of incompressible flows and nonlinear flexible structures. In K. Schweizerhof and W.A. Wall, editors, *Trends in Computational Structural Mechanics*. K.U. Bletzinger, CIMNE, Barcelona, 2001.

[20] J. Mouro. *Interactions Fluide Structure en Grands Déplacements. Résolution Numérique et Application aux Composants Hydrauliques Automobiles*. PhD thesis, École Polytechnique, Paris, 1996.

[21] F. Nobile. *Numerical Approximation of Fluid-Structure Interaction Problems with Application to Haemodynamics*. PhD thesis, EPFL, 2001.

[22] A. Quarteroni and L. Formaggia. Mathematical Modelling and Numerical Simulation of the Cardiovascular System. In *Handbook of Numerical Analysis. Vol. XII*, pages 3–127. North-Holland, Amsterdam, 2004.

[23] A. Quarteroni, R. Sacco, and F. Saleri. *Numerical Mathematics*. Springer, New York, 2000.

[24] A. Quarteroni, A. Veneziani, and P. Zunino. A domain decomposition method for advection-diffusion processes with application to blood solutes. *SIAM J. Sci. Comput.*, 23(6):1959–1980, 2002.

[25] P. Zunino. Iterative substructuring methods for advection-diffusion problems in heterogeneous media. In *Challenges in Scientific Computing—CISC 2002*, volume 35 of *Lect. Notes Comput. Sci. Eng.*, pages 184–210. Springer, Berlin, 2003.