MOX–Report No. 32/2010

Algorithms for the partitioned solution of weakly coupled fluid models for cardiovascular flows

A. Cristiano I. Malossi, Pablo J. Blanco,
Simone Deparis, Alfio Quarteroni

MOX, Dipartimento di Matematica "F. Brioschi"
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

mox@mate.polimi.it http://mox.polimi.it

Algorithms for the partitioned solution of weakly coupled fluid models for cardiovascular flows*

A. Cristiano I. Malossi[♯], Pablo J. Blanco[†,§], Simone Deparis[♯], Alfio Quarteroni[♯,‡]

September 13, 2010

[♯] CMCS, Chair of Modelling and Scientific Computing, MATHICSE, Mathematics Institute of Computational Science and Engineering, EPFL, École Polytechnique Fédérale de Lausanne, Station 8, CH-1015, Lausanne, Switzerland

[†] LNCC, Laboratório Nacional de Computação Científica, Av. Getúlio Vargas 333, Quitandinha, 25651-075, Petrópolis, Brazil

[§] INCT-MACC, Instituto Nacional de Ciência e Tecnologia em Medicina Assistida por Computação Científica, Petrópolis, Brazil

[‡] MOX, Modeling and Scientific Computing, Department of Mathematics, Politecnico di Milano, Via Bonardi 9, Milan, Italy

Keywords: Domain decomposition; Parallel algorithms; Navier–Stokes equations; Geometrical multiscale; Newton method; Hemodynamics; Preconditioning

Abstract

The main goal of the present work is to devise robust iterative strategies to partition the solution of the Navier–Stokes equations in a three-dimensional (3D) computational domain, into non overlapping 3D subdomains, which communicate through the exchange of integrated quantities across the interfaces. The novel aspect of the present approach is that at coupling boundaries the conservation of flow rates and of the associated dual variables is imposed, entailing a weak physical coupling. For the solution of the non-linear problem, written in terms of interfaces variables, two strategies are compared: relaxed fixed point iterations and Newton iterations. The algorithm is tested in several configurations for problems which involve more than two components at each coupling interface. In such cases it is shown that relaxed fixed point methods are not convergent, whereas the Newton method leads in all the tested cases to convergent schemes. One of the appealing aspects of the strategy proposed here is the flexibility in the setting of boundary conditions at branching points, where no hierarchy is established a priori, unlike classical Gauss–Seidel methods.

*This work has been supported by the ERC Grant: Mathcard, *Mathematical Modelling and Simulation of the Cardiovascular System*, ERC-2008-AdG 227058.

Such an approach can be applied in two other different contexts: *(i)* when coupling dimensionally-heterogeneous models, just by replacing some of the 3D models by one-dimensional (or zero-dimensional) condensed ones, and *(ii)* as a preconditioner method for domain decomposition methods for the Navier–Stokes equations. These two issues are also addressed in the present work. Finally, several examples of application are presented, ranging from academic examples to some related to the computational hemodynamics field.

1 Introduction

In the last years there have been increasing efforts towards performing the coupling of models that are heterogeneous regarding some of their features. In specific applications, this has been evinced by the increasing use of dimensionally-heterogeneous coupled models with the aim of bringing together the phenomena pertaining to different geometrical scales within the problem.

In the field of computational hemodynamics, dimensionally-heterogeneity is mandatory to correctly model the global and local circulation. There, the use of coupled three-dimensional (3D) or two-dimensional (2D) detailed models of the Navier–Stokes equations with one-dimensional (1D) and/or zero-dimensional (0D) models to account for the surrounding part of the geometrically isolated vessel has become common practice [1, 2, 3, 4, 5, 6, 7, 8, 9]. Either by imposing a defective boundary condition [10, 11], by setting kinematically-incompatible variational problem [1, 12], or by imposing a lumped impedance boundary condition [5, 9] leads to imposing constraints over the flow rate across the inlets/outlets of the computational domain modelled by the Navier–Stokes equations.

Nonetheless, the continuity equations couple a large number of degrees of freedom, posing some practical difficulties at the discrete level. This motivates the development of iterative strategies in order to deal, in a segregated manner, with such coupled models. It is worth remarking that the continuity conditions enforced between these heterogeneous models are written in terms of integrated quantities that somehow belong to the lower-dimensional 0D/1D models, as showed in the examples cited in the previous paragraph.

In the computational hemodynamics community, the efforts made so far to decouple this problem have addressed very specific situations, without attempting to develop an abstract setting (see approaches in [1, 3, 7, 10]). In a slightly different context, some developments led to more robust schemes, like the ones presented in [11, 13] with the purpose of imposing a given flow rate boundary condition to the 3D Navier–Stokes equations. More recently, in [14] the authors proposed dealing with this kind of coupling by rewriting it as an interface problem for which any matrix-free method for linear systems can be applied (in the case of linear problems). This interface problem is characterized by having a very small number of degrees of freedom, since the interface unknowns are the 0D/1D quantities. In addition, another interesting aspect of that approach is

that two unknowns per coupling interface are kept in the formulation, namely the primal and the dual variables of the problem. This makes the decomposition into subproblems an easier task since it is possible to set up in an independent manner the boundary conditions for the models sharing a common coupling interface. This has also been successfully applied in [15] to the decomposition of 1D networks in order to simulate wave propagation phenomena in compliant vessels, addressing the simulation of the whole systemic circulation.

Exploting the ideas developed in [14] and further extended in [15], the goal of the present work is to devise robust iterative strategies for the coupling of dimensionally-homogeneous flow models through integrated coupling quantities; these quantities are defined according to lower-dimensional fluid models and therefore provide a weak coupling in the case of 3D Navier–Stokes equations. The aim is to partition a 3D computational domain into several complementary non-overlapping subdomains, imposing, at each interface, the conservation of flow rate and the continuity of the associated dual variable, i.e., the normal component of the traction vector. On the reduced problem involving the unknown interface variables we apply the Newton method, which requires the exact evaluation of the Jacobian matrix, as well as a relaxed fixed point method based on the Aitken acceleration. Furthermore, the formulation and iterative solution of interface problems involving more than two branches merging at a coupling point is addressed. This is the situation that arises, for instance, at bifurcations or trifurcations. In such cases it is shown that neither the classical Aitken method nor its variants converge, whereas the Newton method leads, in all the tested cases, to convergent schemes. Another appealing aspect of our approach is the flexibility in the setting of the boundary conditions at the coupling interfaces, where no hierarchy should be established a priori (see [14]). Indeed, Gauss–Seidel methods define a sequential exchange of information at the interfaces, thus yielding an undesirable hierarchical interaction between the models.

All the features of the coupling strategy proposed here are shown through several examples, ranging from academic examples to situations in computational hemodynamics. It is also worth pointing out that these partitioned strategies can be applied to the coupling of dimensionally-heterogeneous models just by replacing some of the 3D models by 1D (or 0D) condensed models or as preconditioners when solving the classical domain decomposition problem involving the Navier–Stokes equations.

This paper is organized as follows: in Section 2 we introduce the general approach for the weak coupling of fluid flow models, which is addressed through two different strategies. Then, in Section 3 we solve the coupled problem using two different approaches: the Aitken method and the Newton method. These techniques are applied to three numerical examples presented in Section 4, while in Section 5 we discuss other extensions and fields of application of the ideas developed here. Finally, in Section 6 we conclude briefly summarizing the main results.

2 Description of the problem

In this section we propose a general approach for the weak coupling of fluid flow models. The main field of application of the methodology devised here is the modelling of complex cardiovascular networks, but it can also be employed to deal with other problems, such as the study of fluid/gas flows in pipe networks (see for instance [16, 17]) and structural frames in solid mechanics (see [18, 19] and references therein), among others. The driving motivation of the present approach is to simplify the modelling of a problem, which will be otherwise too complicated and not affordable from the computational viewpoint. Our aim is to develop a geometrical multiscale framework, able to assemble a network made by components of different geometrical dimension (i.e., 0D, 1D, and 3D) and governed by Partial Differential Equations (PDEs) of various nature. One of the main issues of this problem is how to couple together different models, through heterogeneous interfaces, without losing generality. Our framework has to be flexible enough to assemble a network of heterogeneous (or homogeneous) elements using a general setting. This objective is achieved by imposing the conservation of integrated (averaged) quantities over the interfaces, which do not depend on the geometrical nature or the mathematical formulation of the models.

The target application of the present work is the weak coupling between 3D fluid flow models governed by Navier–Stokes equations. This choice may be motivated by the study of the impact of the weak coupling approximation in solving an originally homogeneous 3D fluid flow problem, by means of weakly coupled homogeneous fluid flow subproblems. However, in Section 5.2 we briefly discuss the extension of the methodology proposed here to cover the case of a truly geometrical multiscale problem.

2.1 Domain decomposition approach

Let us consider a bounded domain Ω with Lipschitz continuous boundary $\partial\Omega$, that may represent a set of branching pipes. Using a domain decomposition approach, we can divide Ω into n_Ω subdomains Ω_j, each one representing a specific part of the original domain. This procedure requires the imposition of continuity conditions on the fictitious boundaries $\partial\Omega_j^{\mathrm{DD}}$ (the interfaces), with $j = 1, \ldots, n_\Omega$. In particular we have

$$\partial\Omega_j^{\mathrm{BC}} = \partial\Omega \cap \partial\Omega_j, \quad \partial\Omega_j^{\mathrm{DD}} = \partial\Omega_j \backslash \partial\Omega_j^{\mathrm{BC}} \qquad \forall j = 1, \ldots, n_\Omega,$$

and

$$\partial\Omega_j^{\mathrm{DD}} = \bigcup_{f=1}^{n_{\Gamma_j}} \Gamma_{j,f},$$

with n_{Γ_j} denoting the number of additional boundary faces $\Gamma_{j,f}$ generated by the domain decomposition on the j-subdomain (see Figure 1). In the following we refer to the continuity conditions on these faces as *coupling conditions*.

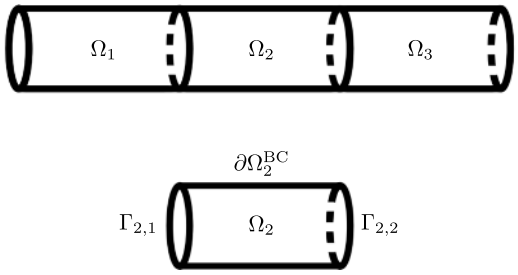


Figure 1: Decomposition of a pipe into three parts: subdomain Ω_2 is bounded by $\partial\Omega_2^{\mathrm{BC}}$ plus two additional fictitious boundaries $\Gamma_{2,1}$ and $\Gamma_{2,2}$, such that $\partial\Omega_2^{\mathrm{DD}} = \Gamma_{2,1} \cup \Gamma_{2,2}$.

2.2 Geometrical considerations

In a very general setting, like the geometrical multiscale method addressed in the introduction (see [1, 3, 9]), different fluid flow models can be adopted in different subdomains. Consequently, different kind of coupling strategies could be envisaged across subdomain boundaries. From the geometrical point of view, we have two different scenarios:

1. **the models belong to the same geometrical space:** coupling conditions consist in imposing directly the pointwise continuity of the unknowns in a functional space (e.g., pointwise velocity and normal stresses for the 3D Navier–Stokes equations), which is the classical approach in domain decomposition;

2. **the models belong to different geometrical spaces:** coupling conditions consist in the conservation of integrated quantities, which do not depend on the geometrical dimension of the models, i.e., these are real numbers.

Even if the two cases do not preclude each other, in this paper we only focus on the second one, which is the natural choice in presence of geometrical multiscale couplings. Nonetheless, in Section 5 we briefly discuss some possible applications of the present approach to problems arising in the first scenario.

Remark 1. We underline that with this approach the resulting model is not equivalent to the original global model. In particular, with respect to classic domain decomposition methods, the coupling conditions are relaxed by averaging the quantities over the interfaces. However, this choice allows the coupling of more than two models at the same coupling node, which is an appealing feature in presence of branching elements.

2.3 Fluid flow model

As said before, the coupling strategies developed in the forthcoming sections are general and can be employed with different physical and geometrical models.

However, to continue our discussion with the aid of a specific example, we select the 3D fluid flows modelled by the Navier–Stokes equations:

$$\begin{cases} \rho\left(\dfrac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u}\cdot\boldsymbol{\nabla})\boldsymbol{u}\right) = -\boldsymbol{\nabla}p + \boldsymbol{\nabla}\cdot(2\mu\boldsymbol{\epsilon}(\boldsymbol{u})) + \boldsymbol{f} & \text{in } \Omega_j \times (0,T], \\ \boldsymbol{\nabla}\cdot\boldsymbol{u} = 0 & \text{in } \Omega_j \times (0,T], \\ \boldsymbol{u} = \boldsymbol{u}_0 & \text{in } \Omega_j \times \{0\}, \\ \boldsymbol{u} = \boldsymbol{0} & \text{on } \partial\Omega_j^{\mathrm{BC}} \times (0,T], \\ (\boldsymbol{\sigma}\cdot\boldsymbol{n})\cdot\boldsymbol{\tau}_1 = \boldsymbol{0}, \quad (\boldsymbol{\sigma}\cdot\boldsymbol{n})\cdot\boldsymbol{\tau}_2 = \boldsymbol{0} & \text{on } \partial\Omega_j^{\mathrm{DD}} \times (0,T], \\ + \text{ Coupling condition(s)} & \text{on } \partial\Omega_j^{\mathrm{DD}} \times (0,T], \end{cases} \tag{1}$$

where $\Omega_j \subset \mathbb{R}^3$ is the fluid domain, $(0,T]$ is the time interval, $\boldsymbol{u}$ is the velocity vector, p is the hydrostatic pressure, μ is the dynamic viscosity, ρ is the density, $\boldsymbol{\epsilon}(\boldsymbol{u}) = (\boldsymbol{\nabla}\boldsymbol{u} + (\boldsymbol{\nabla}\boldsymbol{u})^T)/2$ is the strain rate tensor, $\boldsymbol{\sigma} = -p\boldsymbol{I} + 2\mu\boldsymbol{\epsilon}(\boldsymbol{u})$ is the Cauchy stress tensor (with $\boldsymbol{I}$ as the identity), $(\boldsymbol{n}, \boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ are the normal and tangential directions, and $\boldsymbol{f}$ represents body forces.

Remark 2. From the modelling viewpoint we can replace $(\boldsymbol{\sigma}\cdot\boldsymbol{n})\cdot\boldsymbol{\tau} = \boldsymbol{0}$ on $\partial\Omega_j^{\mathrm{DD}}$ by $\boldsymbol{u}\cdot\boldsymbol{\tau} = \boldsymbol{0}$. This choice is compatible with the imposition of the continuity of the normal stress and of the flow rate, described in equation (3). In fact, from the physical point of view, our domain decomposition approach introduces cutting-sections in the domain Ω, which are supposed to be used only where the flow is almost fully-developed.

2.4 Coupling quantities

Under the hypothesis that all the coupling interfaces are flat and equipped with the outgoing normal $\boldsymbol{n}$, we consider, for a generic coupling interface Γ, the following choice for the coupling quantities in our fluid flow problem

$$Q = \int_\Gamma \boldsymbol{u}\cdot\boldsymbol{n}\,\mathrm{d}\Gamma \quad \text{and} \quad \Sigma = (\boldsymbol{\sigma}\cdot\boldsymbol{n})\cdot\boldsymbol{n} \quad \text{on } \Gamma, \tag{2}$$

where Q is the volumetric flow rate and Σ is the normal component of the traction vector, hereafter referred to as the *coupling stress*. This choice leads to the following equations for the problem at the coupling interfaces:

$$\forall c = 1,\dots,\mathcal{C}\ : \quad \begin{cases} \displaystyle\sum_{m=1}^{\mathcal{M}_c} Q_{c,m} = 0, \\ \Sigma_{c,1} = \Sigma_{c,m}, \qquad \forall m = 2,\dots,\mathcal{M}_c, \end{cases} \tag{3}$$

where $\mathcal{C}$ is the total number of coupling interfaces in the general framework and $\mathcal{M}_c$ is the number of models coupled by the c-coupling interface (see Figure 2). A similar set of coupling equations is introduced in [20] for the coupling of 1D models through the conservation of the flow rate and the continuity of the total pressure.

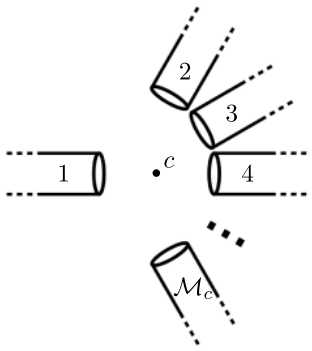


Figure 2: General configuration for the c-coupling between $\mathcal{M}_c$ models.

2.5 Coupling strategies

To satisfy the set of equations (3) we can use different coupling strategies, corresponding to the imposition of different quantities on the boundaries. In other words, we can set up each subproblem with different combinations of boundary data over the coupling interfaces. Let us introduce the two strategies that we employed in the present work for setting such conditions.

Strategy $\mathcal{A}$: the first strategy is obtained by imposing the flow rate boundary data on model 1 and the coupling stress boundary data on models $2, \dots, \mathcal{M}_c$ (see Figure 2). This can be done by rearranging the equations in the following form

$$\forall c = 1, \dots, \mathcal{C} \; : \quad \begin{cases} Q_{c,1} = -\displaystyle\sum_{m=2}^{\mathcal{M}_c} Q_{c,m}(\Sigma_{c,m}), \\ \\ \Sigma_{c,m} = \Sigma_{c,1}(Q_{c,1}), \qquad \forall m = 2, \dots, \mathcal{M}_c, \end{cases} \tag{4}$$

where we have explicitly expressed the dependence of $Q_{c,m}$ on $\Sigma_{c,m}$, and of $\Sigma_{c,1}$ on $Q_{c,1}$. Thanks to $(4)_2$, $\Sigma_{c,2} = \Sigma_{c,3} = \ldots = \Sigma_{c,\mathcal{M}_c}$ and the problem can be rewritten as

$$\forall c = 1, \dots, \mathcal{C} \; : \quad \begin{cases} Q_c = -\displaystyle\sum_{m=2}^{\mathcal{M}_c} Q_{c,m}(\Sigma_c), \\ \\ \Sigma_c = \Sigma_{c,1}(Q_c), \end{cases}$$

or, equivalently,

$$\forall c = 1, \dots, \mathcal{C} \; : \quad \begin{pmatrix} Q_c \\ \Sigma_c \end{pmatrix} = \mathcal{F}_c \begin{pmatrix} Q_c \\ \Sigma_c \end{pmatrix}, \tag{5}$$

where we highlight the presence of two coupling quantities: Q_c and Σ_c. Moreover, the number of coupling equations is independent of $\mathcal{M}_c$.

Strategy $\mathcal{B}$: a second coupling strategy is devised by imposing a coupling stress boundary condition on all models. However, this approach requires an implicit imposition of the coupling equations through a residual formulation

$$\forall c = 1, \dots, \mathcal{C} \; : \quad \boldsymbol{\mathcal{R}}_c(\Sigma_c, Q_{c,2}, \dots, Q_{c,\mathcal{M}_c}) = \mathbf{0}, \tag{6}$$

where

$$\boldsymbol{\mathcal{R}}_c = \begin{pmatrix} Q_{c,1}(\Sigma_c) & + & \displaystyle\sum_{m=2}^{\mathcal{M}_c} Q_{c,m} \\ Q_{c,2}(\Sigma_c) & - & Q_{c,2} \\ Q_{c,3}(\Sigma_c) & - & Q_{c,3} \\ & \vdots & \\ Q_{c,\mathcal{M}_c}(\Sigma_c) & - & Q_{c,\mathcal{M}_c} \end{pmatrix}.$$

In this case, the number of coupling equations grows as a linear function of $\mathcal{M}_c$; in particular, for $\mathcal{M}_c > 2$ the number of coupling equations is greater than the one of the previous case.

Finally we could have a further case, say strategy $\mathcal{C}$, involving the imposition of the flow rate on all models. Nevertheless, this kind of strategy cannot be used systematically, as in some situations the imposition of the flow rate on all boundaries may lead to ill-posed subproblems.

Remark 3. Even if strategies $\mathcal{A}$ and $\mathcal{B}$ involve different coupling conditions, they are equivalent as they have been derived from the same set of equations and coupling conditions, without any approximation. Therefore both systems will deliver the same solution.

Remark 4. For the sake of simplicity, in (5) and (6) we have put in evidence only the dependence on quantities defined at the c-coupling interface. Nevertheless, in a general network, boundary data at the c-coupling could depend also on quantities defined at other interfaces.

3 Numerical approaches

To solve the problem described in the previous section in a segregated manner, we can use different iterative techniques. Let $\boldsymbol{\lambda} = \{\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \dots, \boldsymbol{\lambda}_{\mathcal{C}}\}$ be the vector containing the coupling variables. For the two coupling strategies seen in the previous section this becomes

$$\text{either} \quad \boldsymbol{\lambda}_c^{\mathcal{A}} = \begin{pmatrix} Q_c \\ \Sigma_c \end{pmatrix} \quad \text{or} \quad \boldsymbol{\lambda}_c^{\mathcal{B}} = \begin{pmatrix} \Sigma_c \\ Q_{c,2} \\ \vdots \\ Q_{c,\mathcal{M}_c} \end{pmatrix} \quad \forall c = 1, \dots, \mathcal{C},$$

where we use $\mathcal{A}$ and $\mathcal{B}$ to refer to strategies (5) and (6), respectively.

Remark 5. The global vector of coupling variables $\boldsymbol{\lambda}$ may consist in a combination of different types of local coupling strategies, leading to a quite arbitrary assignments of the boundary conditions on the subdomains. Such flexibility, which is another appealing feature of the present approach, holds as long as the local problems are well-posed; for instance, we cannot impose flow rate boundary conditions on all the boundaries $\Gamma_{j,f}$, $f = 1, \ldots, n_{\Gamma_j}$ of the same subdomain Ω_j, $j = 1, \ldots, n_\Omega$.

3.1 Aitken method

The size and nature of the problem suggest to start with a cheap method. A pure fixed point algorithm cannot be employed, since the convergence is not guaranteed. The first method that we propose is generalized Aitken's (see [21, 22]), based on the following update procedure

$$\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k + \omega^k \boldsymbol{\mathcal{R}}(\boldsymbol{\lambda}^k), \tag{7}$$

where the value of the relaxation parameter ω^k is computed using one of the following formulas

$$\text{Direct relaxation}: \quad \omega^k = \frac{(\boldsymbol{\mathcal{R}}(\boldsymbol{\lambda}^k) - \boldsymbol{\mathcal{R}}(\boldsymbol{\lambda}^{k-1})) \cdot (\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k-1})}{||\boldsymbol{\mathcal{R}}(\boldsymbol{\lambda}^k) - \boldsymbol{\mathcal{R}}(\boldsymbol{\lambda}^{k-1}))||^2},$$

$$\text{Inverse relaxation}: \quad \omega^k = \frac{||\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k-1}||^2}{(\boldsymbol{\mathcal{R}}(\boldsymbol{\lambda}^k) - \boldsymbol{\mathcal{R}}(\boldsymbol{\lambda}^{k-1})) \cdot (\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k-1})}.$$

Note that if $\omega^k = 1$, (7) is a fixed point method. The residual in strategy $\boldsymbol{\mathcal{A}}$ is defined as $\boldsymbol{\mathcal{R}}(\boldsymbol{\lambda}^k) = \mathcal{F}(\boldsymbol{\lambda}^k) - \boldsymbol{\lambda}^k$, while for strategy $\boldsymbol{\mathcal{B}}$ it is embedded in its formulation. Unfortunately, the results provided by the Aitken method are not satisfactory and convergence is guaranteed only in few simple cases, with strong restrictions on the values of $\mathcal{M}_c$ (see Table 1).

Table 1: Qualitative convergence results of the Aitken method.

Strategy	$\omega = 1$	Direct relaxation	Inverse relaxation
$\boldsymbol{\mathcal{A}}$	slow convergence only for $\max(\mathcal{M}_c) = 2$	not converging	convergence only for $\max(\mathcal{M}_c) = 2$
$\boldsymbol{\mathcal{B}}$	slow convergence only for $\max(\mathcal{M}_c) = 2$	not converging	not converging

Indeed, consider the systematic coupling of 3D fluid flow models connected in series as shown in Figure 3.

We set up a stationary simulation of a Poiseuille flow with strategies $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$ using (7). Even in the best case (strategy $\boldsymbol{\mathcal{A}}$ with inverse relaxation) the number

Figure 3: Serial network of n_Ω cylinders connected by $\mathcal{C} = n_\Omega - 1$ couplings. On the left we impose a unitary flow rate, while on the right a no-stress Neumann boundary condition is applied.

of iterations grows quickly with the number of elements inside the serial network, as shown in Table 2.

Table 2: Convergence results of the Aitken method, for a series of elements connected one by one, such that $\max(\mathcal{M}_c) = 2$, as shown in Figure 3.

$\mathcal{C}$	2	3	4	5	6
iterations	4	17	21	24	26

Such unsatisfactory results motivate the use of more sophisticated approaches, like the one presented in the next section.

3.2 Newton method

In order to devise a convergent methodology even in the most general case, we make use of the Newton method

$$\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k + \delta\boldsymbol{\lambda}^k,$$

where $\delta\boldsymbol{\lambda}^k$ is computed solving the following linear system

$$\mathcal{J}(\boldsymbol{\lambda}^k)\delta\boldsymbol{\lambda}^k = -\boldsymbol{\mathcal{R}}(\boldsymbol{\lambda}^k), \tag{8}$$

which requires the computation of the Jacobian matrix $\mathcal{J}(\boldsymbol{\lambda}^k)$. A general detailed description of that matrix is not available, as it depends on the structure of the graph that represents the network of couplings of the specific problem. Nevertheless, each sub-block of the matrix depends only on the type of coupling strategy, as we show in the following section.

This approach is more robust then the Aitken method: the solution is reached in only one iteration in case of the linearized Navier–Stokes equations, while in the non-linear case the flow regime and the network size may affect the number of iterations.

3.2.1 Assembling the Jacobian matrix

Let $\mathcal{J}_{c_1c_2}(\boldsymbol{\lambda}_{c_1}, \boldsymbol{\lambda}_{c_2})$ be a generic sub-block of the Jacobian matrix, where $c_1, c_2 = 1, \ldots, \mathcal{C}$ are two coupling nodes inside the network; $\mathcal{J}_{c_1c_2}(\boldsymbol{\lambda}_{c_1}, \boldsymbol{\lambda}_{c_2})$ expresses the connectivity between coupling nodes c_1 and c_2 through one of the following cases:

1. $c_1 \equiv c_2$: the perturbation $\delta\boldsymbol{\lambda}_{c_2}$ produces a variation on its coupling conditions through a block diagonal element of the Jacobian matrix (c_2 is connected with itself);

2. $c_1 \neq c_2$ and these nodes are connected through a certain model: the perturbation $\delta\boldsymbol{\lambda}_{c_2}$ produces a variation on the coupling conditions in c_1 through an extra diagonal block element of the Jacobian matrix;

3. $c_1 \neq c_2$ and they are not connected through any model: the perturbation $\delta\boldsymbol{\lambda}_{c_2}$ does not affect coupling conditions in c_1.

The block structure of the Jacobian matrix is

$$\mathcal{J}(\boldsymbol{\lambda}) = \begin{bmatrix} \mathcal{J}_{11}(\boldsymbol{\lambda}_{c_1}, \boldsymbol{\lambda}_{c_1}) & \mathcal{J}_{12}(\boldsymbol{\lambda}_{c_1}, \boldsymbol{\lambda}_{c_2}) & \dots & \mathcal{J}_{1\mathcal{C}}(\boldsymbol{\lambda}_{c_1}, \boldsymbol{\lambda}_{\mathcal{C}}) \\ \mathcal{J}_{21}(\boldsymbol{\lambda}_{c_2}, \boldsymbol{\lambda}_{c_1}) & \mathcal{J}_{22}(\boldsymbol{\lambda}_{c_2}, \boldsymbol{\lambda}_{c_2}) & \dots & \mathcal{J}_{2\mathcal{C}}(\boldsymbol{\lambda}_{c_2}, \boldsymbol{\lambda}_{\mathcal{C}}) \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{J}_{\mathcal{C}1}(\boldsymbol{\lambda}_{\mathcal{C}}, \boldsymbol{\lambda}_{c_1}) & \mathcal{J}_{\mathcal{C}2}(\boldsymbol{\lambda}_{\mathcal{C}}, \boldsymbol{\lambda}_{c_2}) & \dots & \mathcal{J}_{\mathcal{C}\mathcal{C}}(\boldsymbol{\lambda}_{\mathcal{C}}, \boldsymbol{\lambda}_{\mathcal{C}}) \end{bmatrix}. \tag{9}$$

In particular, for strategies $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$ the sub-blocks are computed as follows.

Strategy $\boldsymbol{\mathcal{A}}$: the block diagonal elements are

$$\mathcal{J}_{c_1c_1}(\boldsymbol{\lambda}^{\boldsymbol{\mathcal{A}}}) = \begin{bmatrix} -1 & -\displaystyle\sum_{m=2}^{\mathcal{M}_{c_1}} \frac{\partial Q_{c_1,m}}{\partial \Sigma_{c_1}} \\ \dfrac{\partial \Sigma_{c_1}}{\partial Q_{c_1}} & -1 \end{bmatrix},$$

while the pattern of the 2×2 extra diagonal blocks depends on the type of boundary conditions imposed on c_1 and c_2. More precisely:

- $\mathcal{J}_{c_1c_2}(\boldsymbol{\lambda}^{\boldsymbol{\mathcal{A}}}) = \begin{bmatrix} 0 & 0 \\ \dfrac{\partial \Sigma_{c_1}}{\partial Q_{c_2}} & 0 \end{bmatrix}$, if a flow rate is imposed on both c_1 and c_2;
- $\mathcal{J}_{c_1c_2}(\boldsymbol{\lambda}^{\boldsymbol{\mathcal{A}}}) = \begin{bmatrix} -\dfrac{\partial Q_{c_1}}{\partial Q_{c_2}} & 0 \\ 0 & 0 \end{bmatrix}$, if a coupling stress is imposed on c_1 and a flow rate on c_2;
- $\mathcal{J}_{c_1c_2}(\boldsymbol{\lambda}^{\boldsymbol{\mathcal{A}}}) = \begin{bmatrix} 0 & -\dfrac{\partial Q_{c_1}}{\partial \Sigma_{c_2}} \\ 0 & 0 \end{bmatrix}$, if a coupling stress is imposed on both c_1 and c_2;
- $\mathcal{J}_{c_1c_2}(\boldsymbol{\lambda}^{\boldsymbol{\mathcal{A}}}) = \begin{bmatrix} 0 & 0 \\ 0 & \dfrac{\partial \Sigma_{c_1}}{\partial \Sigma_{c_2}} \end{bmatrix}$, if a flow rate is imposed on c_1 and a coupling stress on c_2.

Strategy $\mathcal{B}$: the block diagonal elements are

$$\mathcal{J}_{c_1c_1}(\boldsymbol{\lambda}^{\mathcal{B}}) = \begin{bmatrix} \dfrac{\partial Q_{c_1,1}}{\partial \Sigma_{c_1}} & 1 & 1 & \dots & 1 \\ \dfrac{\partial Q_{c_1,2}}{\partial \Sigma_{c_1}} & -1 & 0 & \dots & 0 \\ \dfrac{\partial Q_{c_1,3}}{\partial \Sigma_{c_1}} & 0 & -1 & & \vdots \\ \vdots & \vdots & & \ddots & \vdots \\ \dfrac{\partial Q_{c_1,\mathcal{M}_{c_1}}}{\partial \Sigma_{c_1}} & 0 & \dots & \dots & -1 \end{bmatrix},$$

while all the extra diagonal blocks assume the following form

$$\mathcal{J}_{c_1c_2}(\boldsymbol{\lambda}^{\mathcal{B}}) = \begin{bmatrix} 0 & 0 & \dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & 0 \\ \dfrac{\partial Q_{c_1,v}}{\partial \Sigma_{c_2}} & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix},$$

where $1 \leq v \leq \mathcal{V}_c$ and $\mathcal{V}_c$ is the local number of coupling variables, i.e., the length of $\boldsymbol{\lambda}_c$. The size of the blocks of the matrix is $\mathcal{M}_{c_1} \times \mathcal{M}_{c_1}$ for the diagonal elements and $\mathcal{M}_{c_1} \times \mathcal{M}_{c_2}$ for the extra diagonal ones.

To assemble the Jacobian matrix (9) we can use different approaches. An instance is provided in Algorithm 1, which builds the Jacobian by blocks.

Algorithm 1 – Jacobian matrix assembling procedure.

```
for c_1 = 1, ..., C
    impose δλ_{c_1} = 1
    compute J_{c_1c_1}(λ_{c_1})
    for c_2 = 1, ..., C
        if c_1 is connected to c_2
            if c_1 ≠ c_2
                compute J_{c_1c_2}(λ_{c_1}, λ_{c_2})
            end
        else
            J_{c_1c_2} = 0
        end
    end
end
```

This approach is quite naive and not very efficient, as in general it leads to perform multiple perturbations of the same coupling variable $\lambda_{c,v}$ and hence to perform multiple solutions of the same linear system. To minimize the computational cost it is necessary to assemble the matrix column by column; in other words, for each perturbation $\delta\lambda_{c,v}$ we solve the linear system associated to each problem only once, computing all the coefficients in the same column.

A practical example of the resulting matrices is presented in Section 4.1, while the computation of the Jacobian entries is outlined in the following section.

3.2.2 Computation of the Jacobian coefficients

The computation of the coefficients of the Jacobian matrix depends on the equations used within each subdomain. For the case of the 3D Navier–Stokes equations (1), the computation of the derivatives can be achieved through the solution of the tangent problem formulated on the subdomain Ω_j. Given $\boldsymbol{u} \in \boldsymbol{\mathcal{U}}(\Omega_j)$, δQ_f, $f \in \mathcal{L}_j^Q$, and $\delta\Sigma_f$, $f \in \mathcal{L}_j^\Sigma$, find $\delta\boldsymbol{u} \in \boldsymbol{\mathcal{U}}(\Omega_j)$, $\delta p \in L^2(\Omega_j)$, and $\delta\Lambda_f \in \mathbb{R}$, $f \in \mathcal{L}_j^Q$, such that

$$\begin{cases} \displaystyle\int_{\Omega_j} \rho \frac{\partial \delta\boldsymbol{u}}{\partial t} \cdot \boldsymbol{v} \mathrm{d}\Omega + \int_{\Omega_j} \rho(\delta\boldsymbol{u} \cdot \boldsymbol{\nabla})\boldsymbol{u} \cdot \boldsymbol{v} \mathrm{d}\Omega + \int_{\Omega_j} \rho(\boldsymbol{u} \cdot \boldsymbol{\nabla})\delta\boldsymbol{u} \cdot \boldsymbol{v} \mathrm{d}\Omega \\ \displaystyle -\int_{\Omega_j} \delta p \boldsymbol{\nabla} \cdot \boldsymbol{v} \mathrm{d}\Omega + \int_{\Omega_j} \mu \boldsymbol{\nabla}\delta\boldsymbol{u} \cdot \boldsymbol{\nabla}\boldsymbol{v} \mathrm{d}\Omega + \sum_{f \in \mathcal{L}_j^Q} \int_{\Gamma_{j,f}} \delta\Lambda_f \boldsymbol{v} \cdot \boldsymbol{n} \mathrm{d}\Gamma = \\ \displaystyle \qquad\qquad - \sum_{f \in \mathcal{L}_j^\Sigma} \int_{\Gamma_{j,f}} \delta\Sigma_f \boldsymbol{v} \cdot \boldsymbol{n} \mathrm{d}\Gamma & \forall \boldsymbol{v} \in \boldsymbol{\mathcal{V}}(\Omega_j), \\ \displaystyle\int_{\Omega_j} q \boldsymbol{\nabla} \cdot \delta\boldsymbol{u} \mathrm{d}\Omega = 0 & \forall q \in L^2(\Omega_j), \\ \displaystyle\int_{\Gamma_{j,f}} \delta\boldsymbol{u} \cdot \boldsymbol{n} \mathrm{d}\Gamma = \delta Q_f & f \in \mathcal{L}_j^Q, \end{cases} \tag{10}$$

where in $\boldsymbol{\mathcal{U}}(\Omega_j)$ we consider the essential boundary conditions as given in (1) and $\boldsymbol{\mathcal{V}}(\Omega_j)$ is the associated linear space. $\mathcal{L}_j^Q$ and $\mathcal{L}_j^\Sigma$ are the lists of boundaries of Ω_j where flow rate and coupling stress conditions are applied, respectively. With this set of equations we compute the coefficients of the block matrices $\mathcal{J}_{c_1,c_2}$, with $c_1, c_2 = 1, \dots, \mathcal{C}$, that are used in the assembling procedure of the Jacobian matrix. First of all, we have to find which are the boundaries Γ_{j,f_1} and Γ_{j,f_2} associated with the couplings c_1 and c_2, respectively. Then, we compute the coefficients in the following way:

- $\dfrac{\partial Q_{c_1}}{\partial Q_{c_2}}$ and $\dfrac{\partial \Sigma_{c_1}}{\partial Q_{c_2}}$ are computed as the resulting flow rate and coupling stress from the imposition to (10), of a variation $\delta Q_{c_2} = 1$ on Γ_{j,f_2} and homogeneous boundary conditions of the corresponding type on all the other boundaries, i.e., $\delta Q_f = 0$, $\forall f \in \mathcal{L}_j^Q$, $f \neq f_2$ and $\delta\Sigma_f = 0$, $\forall f \in \mathcal{L}_j^\Sigma$ (see Figures 4(a) and 4(b));

- $\dfrac{\partial \Sigma_{c_1}}{\partial \Sigma_{c_2}}$ and $\dfrac{\partial Q_{c_1}}{\partial \Sigma_{c_2}}$ are computed as the resulting coupling stress and flow rate from the imposition to (10) of a variation $\delta\Sigma_{c_2} = 1$ on Γ_{j,f_2} and homogeneous boundary conditions of the corresponding type on all the other boundaries, i.e., $\delta Q_f = 0,\ \forall f \in \mathcal{L}_j^Q$ and $\delta\Sigma_f = 0,\ \forall f \in \mathcal{L}_j^\Sigma,\ f \neq f_2$ (see Figures 4(c) and 4(d)).

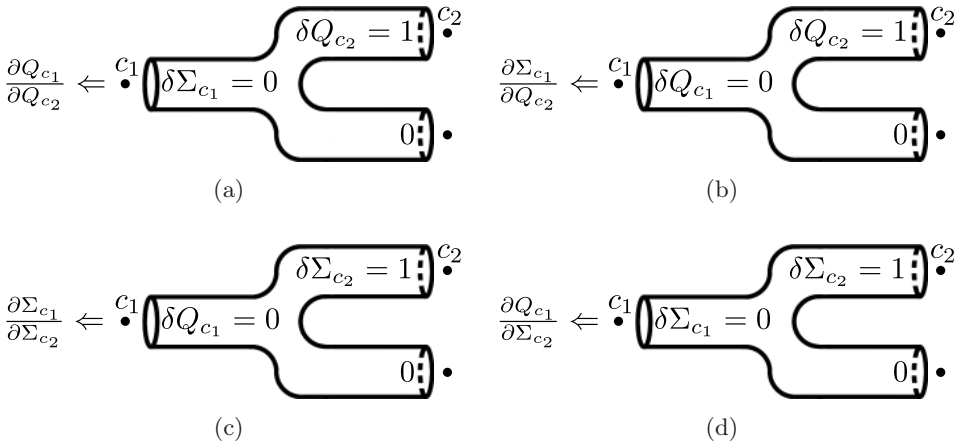


Figure 4: Example of computation of the Jacobian coefficients on a bifurcation. In the sketches, the zero on the lower branch of the bifurcation stands for an homogeneous value of the corresponding boundary condition (Q or Σ, depending on the nature of the original boundary data). (a) Computation of $\dfrac{\partial Q_{c_1}}{\partial Q_{c_2}}$. (b) Computation of $\dfrac{\partial \Sigma_{c_1}}{\partial Q_{c_2}}$. (c) Computation of $\dfrac{\partial \Sigma_{c_1}}{\partial \Sigma_{c_2}}$. (d) Computation of $\dfrac{\partial Q_{c_1}}{\partial \Sigma_{c_2}}$.

It is possible to use matrix-free iterative methods to solve the Jacobian system (8), but due to the particular nature and the small size of the Jacobian, this usually leads to an increased number of solutions of problem (10).

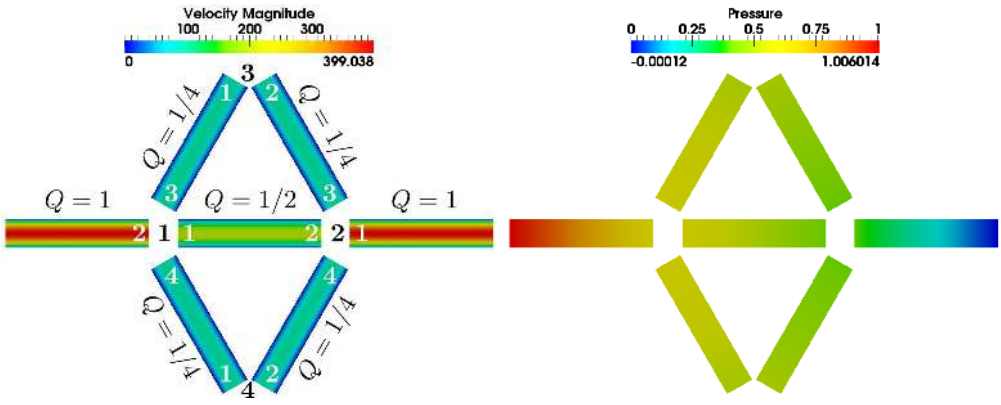
4 Results

In this section we show some numerical results obtained using the coupling strategies and the numerical algorithms presented in the previous sections. In particular, we present three examples: the first one is a steady Stokes test case, which helps us on giving more details about the main features of the presented methods. In the second example, we make use of the coupling strategies to solve the Womersley flow (see [23]) and present a quantitative comparison between this numerical solution and the exact velocity profile. Finally, in the third case we simulate the blood flow in the carotid bifurcation under physiological regimes. All the numerical solutions presented in the following sections have been obtained

using a $\mathbb{P}1$–$\mathbb{P}1$ Finite Element (FE) method, stabilized through interior penalty [24].

4.1 Seven cylinders in steady Stokes

The first example we propose is a 3D steady Stokes problem, consisting in a set of seven cylinders connected by four coupling interfaces, as shown in Figure 5(a). All the cylinders have the same dimensions (radius 0.08; length 0.4). A unitary flow rate is imposed on the leftmost side, while an homogeneous Neumann boundary condition is applied on the outflow, on the right. The physical quantities (density and viscosity) are chosen in order to obtain a unitary pressure drop along the network. The solution obtained with the strategies presented in Section 2.5 is shown in Figure 5.



(a) (b)

Figure 5: Configuration and solution of seven cylinders in a steady Stokes simulation. (a) Velocity magnitude; black numbers indicate the coupling interfaces, while white numbers indicate the local numeration of the boundaries. (b) Pressure field: the drop pressure between the inflow and the outflow is equal to one.

For this specific problem, the vectors of coupling variables and the residual

equations for the two coupling strategies are

$$\boldsymbol{\lambda}^{\boldsymbol{\mathcal{A}},k} = \left(\begin{array}{c} Q_1^k \\ \Sigma_1^k \\ \hline Q_2^k \\ \Sigma_2^k \\ \hline Q_3^k \\ \Sigma_3^k \\ \hline Q_4^k \\ \Sigma_4^k \end{array}\right), \qquad \boldsymbol{\mathcal{R}}(\boldsymbol{\lambda}^{\boldsymbol{\mathcal{A}},k}) = \left(\begin{array}{l} -Q_{1,2}\left(\Sigma_1^k\right) - Q_{1,3}\left(\Sigma_1^k, Q_3^k\right) - Q_{1,4}\left(\Sigma_1^k, Q_4^k\right) \quad - Q_1^k \\ \Sigma_{1,1}\left(Q_1^k, \Sigma_2^k\right) \quad - \Sigma_1^k \\ \hline -Q_{2,2}\left(Q_1^k, \Sigma_2^k\right) - Q_{2,3}\left(\Sigma_2^k, \Sigma_3^k\right) - Q_{2,4}\left(\Sigma_2^k, \Sigma_4^k\right) - Q_2^k \\ \Sigma_{2,1}\left(Q_2^k\right) \quad - \Sigma_2^k \\ \hline -Q_{3,2}\left(\Sigma_2^k, \Sigma_3^k\right) \quad - Q_3^k \\ \Sigma_{3,1}\left(\Sigma_1^k, Q_3^k\right) \quad - \Sigma_3^k \\ \hline -Q_{4,2}\left(\Sigma_2^k, \Sigma_4^k\right) \quad - Q_4^k \\ \Sigma_{4,1}\left(\Sigma_1^k, Q_4^k\right) \quad - \Sigma_4^k \end{array}\right),$$

$$\boldsymbol{\lambda}^{\boldsymbol{\mathcal{B}},k} = \begin{pmatrix} \Sigma_1^k \\ Q_{1,2}^k \\ Q_{1,3}^k \\ Q_{1,4}^k \\ \hline \Sigma_2^k \\ Q_{2,2}^k \\ Q_{2,3}^k \\ Q_{2,4}^k \\ \hline \Sigma_3^k \\ Q_{3,2}^k \\ \hline \Sigma_4^k \\ Q_{4,2}^k \end{pmatrix}, \qquad \boldsymbol{\mathcal{R}}(\boldsymbol{\lambda}^{\boldsymbol{\mathcal{B}},k}) = \begin{pmatrix} Q_{1,1}\left(\Sigma_1^k, \Sigma_2^k\right) + Q_{1,2}^k + Q_{1,3}^k + Q_{1,4}^k \\ Q_{1,2}\left(\Sigma_1^k\right) - Q_{1,2}^k \\ Q_{1,3}\left(\Sigma_1^k, \Sigma_3^k\right) - Q_{1,3}^k \\ Q_{1,4}\left(\Sigma_1^k, \Sigma_4^k\right) - Q_{1,4}^k \\ \hline Q_{2,1}\left(\Sigma_2^k\right) + Q_{2,2}^k + Q_{2,3}^k + Q_{2,4}^k \\ Q_{2,2}\left(\Sigma_1^k, \Sigma_2^k\right) - Q_{2,2}^k \\ Q_{2,3}\left(\Sigma_2^k, \Sigma_3^k\right) - Q_{2,3}^k \\ Q_{2,4}\left(\Sigma_2^k, \Sigma_4^k\right) - Q_{2,4}^k \\ \hline Q_{3,1}\left(\Sigma_1^k, \Sigma_3^k\right) + Q_{3,2}^k \\ Q_{3,2}\left(\Sigma_2^k, \Sigma_3^k\right) - Q_{3,2}^k \\ \hline Q_{4,1}\left(\Sigma_1^k, \Sigma_4^k\right) + Q_{4,2}^k \\ Q_{4,2}\left(\Sigma_2^k, \Sigma_4^k\right) - Q_{4,2}^k \end{pmatrix},$$

where the numeration follows the scheme of Figure 5(a). The Aitken method does not converge because of the complex pattern of the network, while the Newton method converges in one iteration, since the fluid flow problem defined inside each cylinder is linear. The Jacobian matrices for the two coupling strategies are

$$\mathcal{J}(\boldsymbol{\lambda}^{\boldsymbol{\mathcal{A}}}) = \left[\begin{array}{cc|cc|cc|cc} -1 & -\sum_{m=2}^{4} \partial_{\Sigma_1} Q_{1,m} & & & -\partial_{Q_3} Q_{1,3} & & -\partial_{Q_4} Q_{1,4} & \\ \partial_{Q_1}\Sigma_{1,1} & -1 & & & & & & \\ \hline -\partial_{Q_1} Q_{2,2} & & -1 & -\sum_{m=2}^{4} \partial_{\Sigma_2} Q_{2,m} & & -\partial_{\Sigma_3} Q_{2,3} & & -\partial_{\Sigma_4} Q_{2,4} \\ & & \partial_{Q_2}\Sigma_{2,1} & -1 & & & & \\ \hline & & & -\partial_{\Sigma_2} Q_{3,2} & -1 & -\partial_{\Sigma_3} Q_{3,2} & & \\ & \partial_{\Sigma_1}\Sigma_{3,1} & & & \partial_{Q_3}\Sigma_{3,1} & -1 & & \\ \hline & & & -\partial_{\Sigma_2} Q_{4,2} & & & -1 & -\partial_{\Sigma_4} Q_{4,2} \\ & \partial_{\Sigma_1}\Sigma_{4,1} & & & & & \partial_{Q_4}\Sigma_{4,1} & -1 \end{array}\right],$$

$$\mathcal{J}(\boldsymbol{\lambda}^{\boldsymbol{\mathcal{B}}}) = \left[\begin{array}{cccc|cccc|cc|cc} \partial_{\Sigma_1} Q_{1,1} & 1 & 1 & 1 & \partial_{\Sigma_2} Q_{1,1} & & & & & & & \\ \partial_{\Sigma_1} Q_{1,2} & -1 & & & & & & & & & & \\ \partial_{\Sigma_1} Q_{1,3} & & -1 & & & & & & \partial_{\Sigma_3} Q_{1,3} & & & \\ \partial_{\Sigma_1} Q_{1,4} & & & -1 & & & & & & & \partial_{\Sigma_4} Q_{1,4} & \\ \hline & & & & \partial_{\Sigma_2} Q_{2,1} & 1 & 1 & 1 & & & & \\ \partial_{\Sigma_1} Q_{2,2} & & & & \partial_{\Sigma_2} Q_{2,2} & -1 & & & & & & \\ & & & & \partial_{\Sigma_2} Q_{2,3} & & -1 & & \partial_{\Sigma_3} Q_{2,3} & & & \\ & & & & \partial_{\Sigma_2} Q_{2,4} & & & -1 & & & \partial_{\Sigma_4} Q_{2,4} & \\ \hline \partial_{\Sigma_1} Q_{3,1} & & & & & & & & \partial_{\Sigma_3} Q_{3,1} & 1 & & \\ & & & & \partial_{\Sigma_2} Q_{3,2} & & & & \partial_{\Sigma_3} Q_{3,2} & -1 & & \\ \hline \partial_{\Sigma_1} Q_{4,1} & & & & & & & & & & \partial_{\Sigma_4} Q_{4,1} & 1 \\ & & & & \partial_{\Sigma_2} Q_{4,2} & & & & & & \partial_{\Sigma_4} Q_{4,2} & -1 \end{array}\right],$$

where we use the compact notation ∂_Q and ∂_Σ to express the partial derivative of a quantity, with respect to the flow rate and the coupling stress, respectively; the white spaces indicate null entries. To compute all the coefficients of the Jacobian matrix we perform twelve evaluations of problem (10). This number is directly related to the implementation of the assembling algorithm, which in our case is optimal with respect to the number of evaluations of the linear problems associated to the Jacobian. For this problem, in which we have the flow rate and the coupling stress as the unknowns, this number is equal to the number of artificial boundary surfaces created by the decomposition of the original domain Ω into subdomains $\Omega_j, j = 1, \ldots, n_\Omega$ (twelve in this example). Therefore, at each subiteration of the Newton method we need to perform $\sum_{j=1}^{n_\Omega} n_{\Gamma_j}$ evaluations of problem (10), plus n_Ω evaluations of problem (1) to evaluate the residual. These operations can be done in a complete parallel fashion, as we will discuss in Section 5.1.

The results obtained with strategies $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$ are equivalent. The computed flow rate is exact up to the imposed tolerance for the Navier–Stokes solver (i.e., 10^{-13}). The convergence of the coupling stress with respect to h (the spatial discretization of the whole domain Ω) is shown in Figure 6.

4.2 Five cylinders in a Womersley regime

The second example we propose is a Womersley fluid flow (see [23]) in a cylindrical domain Ω composed by five subdomains Ω_j of equal size and shape discretized by the same unstructured mesh (with $h/D = 0.15415$, where D is the diameter). On the leftmost inflow we impose a sinusoidal flow rate $Q = -A\sin(2\pi t/T)$, while an homogeneous Neumann boundary condition is applied on the rightmost

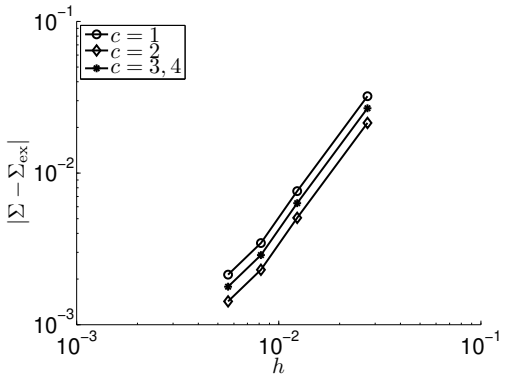


Figure 6: Coupling stress error for the four coupling interfaces in the example of Figure 5(a). Strategies $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$ give the same result.

outflow. The amplitude A, the period T, and all the physical quantities have been chosen in order to obtain a Womersley number of 5 and a Reynolds number of 600, which correspond to physiological quantities from the hemodynamic point of view. The time step is given by $T/256$.

For this problem, each subiteration of the Newton method requires 8 evaluations of problem (10) to assemble the Jacobian matrix, plus 5 evaluations of problem (1) to compute the solution in the 3D domains once the linear system (8) was solved. This approach ensures that the solution satisfies, at each time step, the continuity equations (3) at the coupling interfaces among the five subdomains. As for the example described in Section 4.1, the solutions computed using strategies $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$ are the same. Moreover, also the velocity profiles in the different subdomains coincide. In Figure 7, we show a comparison between the magnitude of the computed velocity profiles and the analytical solution, at selected times. From these graphs, we observe that the velocity profiles follow the shape of the exact solution.

4.3 Carotid bifurcation

In this last example, we use the methodology devised in this paper to compare the solution obtained using a Physiological Carotid Bifurcation (PCB), with the one given by a Synthetic Carotid Bifurcation (SCB). The geometries and the arranges of the two schemes are described in Figure 8.

For the physical quantities we use physiological values ($\rho = 1\text{g/cm}^3$, $\mu = 0.035\text{g/cm/s}$), imposing a heart systole-diastole flow rate cycle (corresponding to 75 beats per minute) at the inlet of Ω_1 while an homogeneous Neumann boundary condition is applied on the outflow of Ω_2 and Ω_3. It is worth pointing out that the two configurations feature a different number of coupling interfaces. Therefore for the PCB each subiteration of the Newton method requires 6 evaluations of problem (10), plus 4 evaluations of problem (1), while for the SCB it requires 9

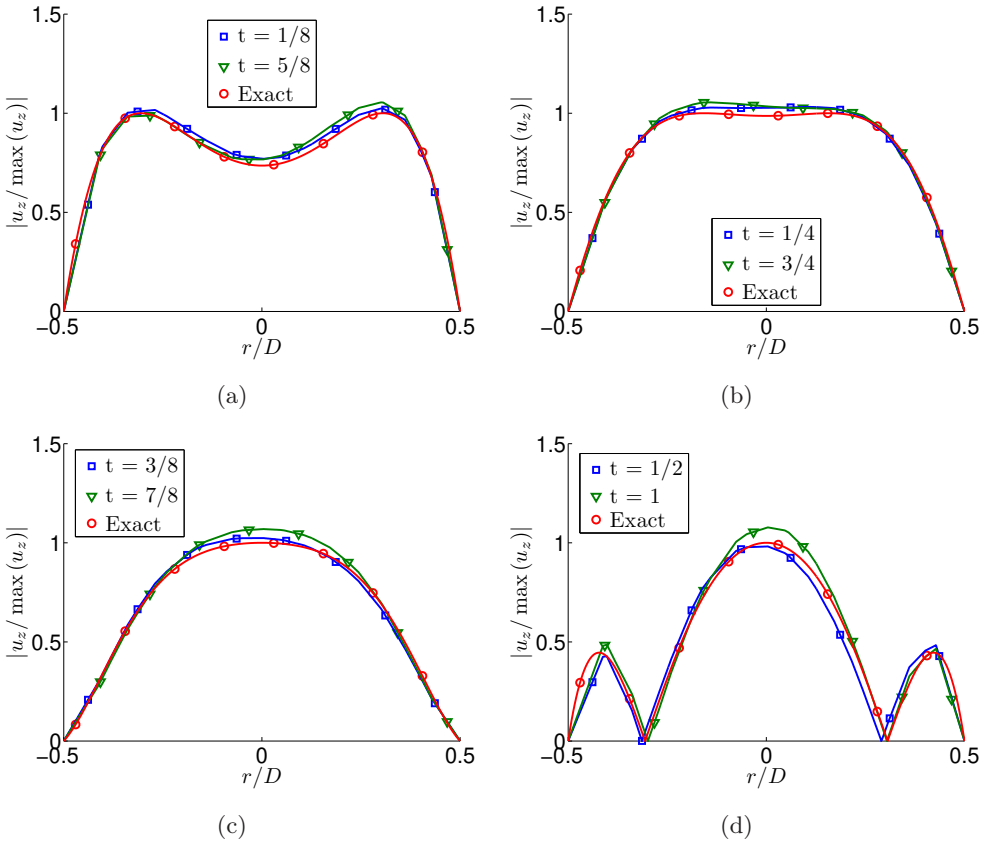


Figure 7: Womersley velocity profiles: on the x-axis we show the ratio between the radial coordinate and the diameter, while on the y-axis we display the modulus of the axial velocity, normalized on the maximum value of the exact solution.

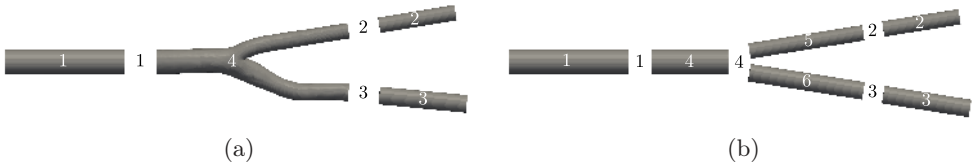
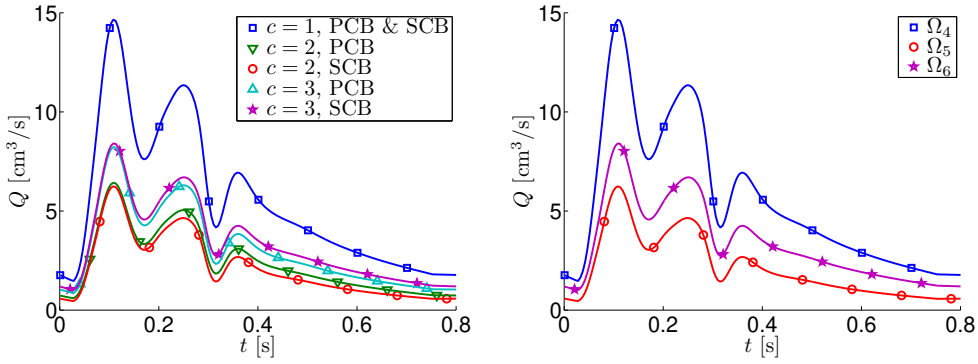


Figure 8: Carotid bifurcation schemes: subdomains Ω_1, Ω_2, and Ω_3 have been scaled in order to have the same area of the corresponding interfaces of the carotid bifurcation. (a) Physiological carotid: Ω_4. (b) Synthetic carotid: Ω_4, Ω_5, and Ω_6.

evaluations of problem (10), plus 6 evaluations of problem (1).

For the two configurations we compare the flow division and the coupling stress at the different coupling interfaces. In addition, we compare also the velocity profiles inside the two carotids. In Figure 9, the evolution of the flow

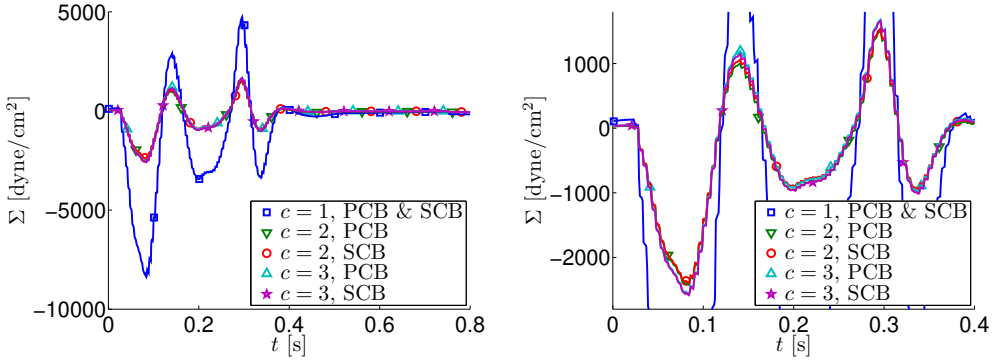
rate during one heart beat is shown. In particular, in Figure 9(a), the small difference between the flow rates of the PCB with respect to the ones of the SCB (at $c = 2$ and $c = 3$), is due to the different approximation used at the bifurcation, which in the case of the SCB neglects the effects of the 3D geometry. Nevertheless, we obtain a good separation of the flows in the SCB when compared to the PCB.



(a) (b)

Figure 9: (a) Flow rate at the different coupling interfaces of the PCB and the SCB. (b) Flow rate splitting at $c = 4$ of the SCB.

On the other hand, for the coupling stress (Figure 10), we observe that there are no significant differences between the PCB and the SCB, as this quantity mainly depends on the global length of the geometrical model, rather than on the 3D geometry.



(a) (b)

Figure 10: (a) Coupling stress at different coupling interfaces for the PCB and the SCB. (b) Zoomed view of the left graph.

Finally, in Figure 11, we show a qualitative comparison between the velocity

profiles of the two carotids, at selected times. As expected, in the SCB the velocity assumes a Womersley-like profile, which is not the case of the PCB, where the shape of the 3D bifurcation produces asymmetries in the solution. Nevertheless, the magnitude of the velocity in the two cases is nearly the same. In addition, we observe that the velocity profiles at the coupling interfaces agree well in both cases.

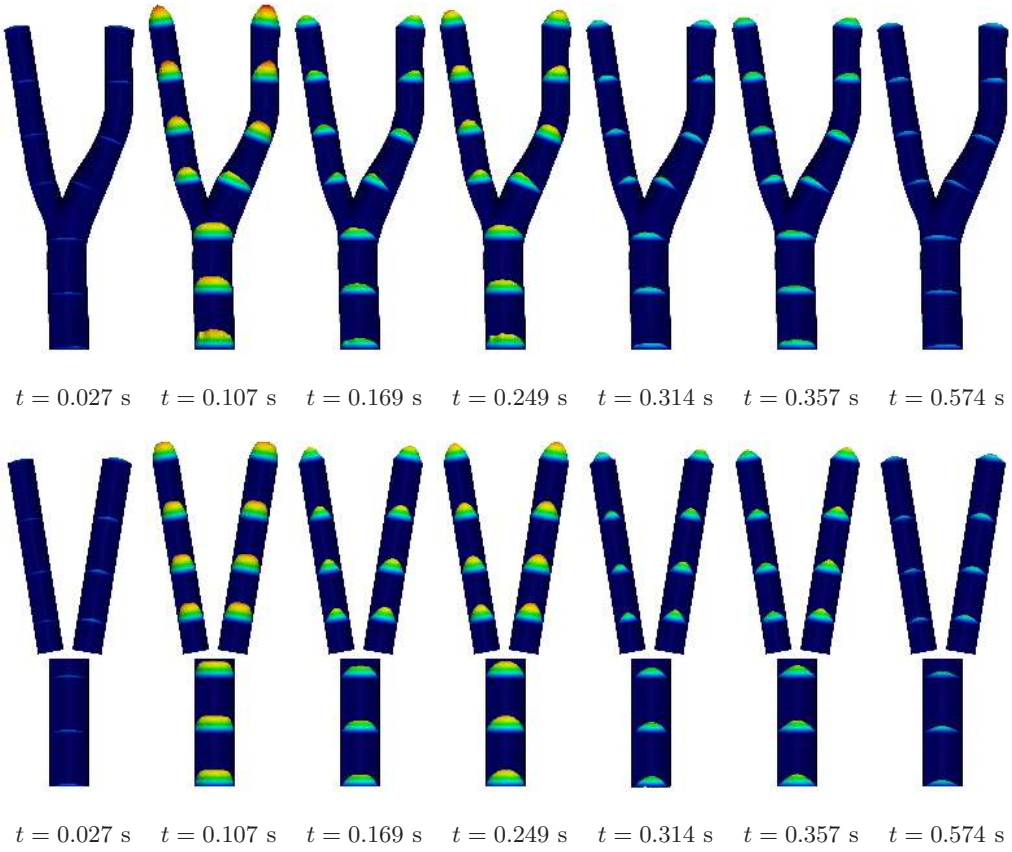


Figure 11: Comparison between the velocity profiles of the PCB (first line), with the ones obtained for the SCB (second line). Colors: blue = 0.0 cm/s, red = 79.6 cm/s.

5 Extensions and applications

In this section, we comment about the implementation of our methodology in a parallel framework, then we address two other possible fields of application, where the main ideas proposed in this paper can be effectively used.

5.1 Implementation in a parallel framework

By the domain decomposition approach introduced in Section 2.1 the original problem is partitioned into n_Ω smaller subproblems defined in Ω_j, with $j = 1, \ldots, n_\Omega$, where each of them exchanges information with the others only through the coupling conditions imposed on their artificial boundaries. Unlike a Gauss–Seidel method, which implies a sequential exchange of information at the boundaries, the methods proposed in this paper are perfectly suited for a parallelization of the code at the higher level (i.e., the level of the management of the different subdomains). Let us consider Algorithm 2, which synthetically illustrates the main steps required for the parallel solution of a general problem at each time step.

Algorithm 2 – Time advancing with Newton method.

1: initialize $\boldsymbol{\lambda} = \mathbf{0}$
2: **for** $t = 0, \ldots, T$
3: solve $\Omega_j,\ \forall j = 1, \ldots, n_\Omega$ and compute $\boldsymbol{\mathcal{R}}(\boldsymbol{\lambda})$ [parallel step]
4: **while** $\boldsymbol{\mathcal{R}}(\boldsymbol{\lambda}) \geq$ tolerance
5: compute $\mathcal{J}(\boldsymbol{\lambda})$ [parallel step]
6: solve $\mathcal{J}(\boldsymbol{\lambda})\delta\boldsymbol{\lambda} = -\boldsymbol{\mathcal{R}}(\boldsymbol{\lambda})$ and update $\boldsymbol{\lambda}$
7: solve $\Omega_j,\ \forall j = 1, \ldots, n_\Omega$ and get $\boldsymbol{\mathcal{R}}(\boldsymbol{\lambda})$ [parallel step]
8: **end**
9: **end**

From the computational point of view, the most expensive steps are the solution of the subproblems and the computation of the coefficients of the Jacobian matrix. Nevertheless, all these phases can be easily parallelized. In particular, the solution of the subproblems, performed in steps 3 and 7, is done for a fixed value of $\boldsymbol{\lambda}$. Hence, each subproblem can be solved on a different machine (where also a lower level parallelization can be performed), without exchanging any information with the others. The same happens for the computation of the coefficients of the Jacobian matrix, performed in step 5. In fact, we can solve each problem, together with the corresponding perturbation on the associated boundary interface, in a completely independent manner. From this analysis we conclude that the strategies and the algorithms developed in the previous sections can be easily ported to a parallel framework.

5.2 Geometrical multiscale modelling

The set of equations (3) is written in terms of integrated quantities and therefore it can be employed to couple models of different type in a geometrical multiscale framework. Let us consider a problem of wave propagation in a deformable pipe, which can be modelled by the coupling between a 3D Fluid-Structure Interaction (FSI) model (e.g., see [25]) and a non-linear 1D model of fluid flow in compliant

vessels (e.g., see [26]), as shown in Figure 12.

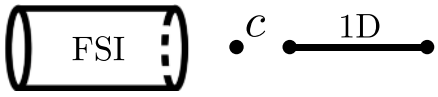


Figure 12: Geometrical multiscale coupling between a FSI model and a 1D model.

In this specific case, strategy $\boldsymbol{\mathcal{A}}$ assumes one of the following forms depending on the assignment of the coupling conditions on the coupling interfaces

$$\boldsymbol{\mathcal{A}}_1: \begin{cases} Q_c = -Q_c^{\mathrm{FSI}}(\Sigma_c), \\ \Sigma_c = \Sigma_c^{\mathrm{1D}}(Q_c), \end{cases} \qquad \boldsymbol{\mathcal{A}}_2: \begin{cases} Q_c = -Q_c^{\mathrm{1D}}(\Sigma_c), \\ \Sigma_c = \Sigma_c^{\mathrm{FSI}}(Q_c), \end{cases}$$

where Q_c^{FSI}, Q_c^{1D}, Σ_c^{FSI}, and Σ_c^{1D} are non-linear operators hiding the complexity of the set of equations of the 1D and FSI models. In particular, strategy $\boldsymbol{\mathcal{A}}_1$ corresponds to the imposition of a coupling stress Σ_c on the FSI model through the operator Q_c^{FSI} and of a flow rate Q_c on the 1D model through the operator Σ_c^{1D}, while strategy $\boldsymbol{\mathcal{A}}_2$ is the inverse situation. For strategy $\boldsymbol{\mathcal{B}}$ we have

$$\boldsymbol{\mathcal{B}}: \begin{cases} Q_c^{\mathrm{FSI}}(\Sigma_c) + Q_c = 0, \\ Q_c^{\mathrm{1D}}(\Sigma_c) - Q_c = 0. \end{cases}$$

This example allows us to highlight an additional aspect of the coupling strategies described in this paper. From the coupling viewpoint, the models can be seen as *black boxes*, where the coupling boundaries are their unique external interfaces. In addition, from the numerical point of view, all the intricacies related to the nonlinearities of the models are embodied inside these boxes. Naturally, according to the strength of the nonlinearities we have a different behaviour of the iterative solution method.

5.3 Preconditioning strategies based on weak coupling

Suppose that we want to solve a fluid problem in a continuous 3D domain Ω partitioned into n_Ω smaller subdomains Ω_j, with $j = 1, \ldots, n_\Omega$. Now, using the classical domain decomposition approach described in Section 2.4 (see item 1), we impose the pointwise continuity of the unknowns (velocity and stress) on the coupling interfaces, as shown in Figure 13.

The methodology developed in this work, provides a systematic and simple way to build a preconditioner for this kind of problem. Let us consider a condensation of the coupling quantities through equation (2) in order to perform a weak coupling on the boundaries, as shown in Figure 14.

This approximation of the original strongly coupled domain decomposition problem (like in Figure 13) is, indeed, the problem which is addressed throughout

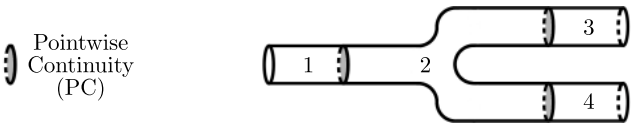


Figure 13: An example of classical domain decomposition, imposing pointwise continuity of the unknowns on the interfaces (in grey). This problem is referred to as A_{PC}.

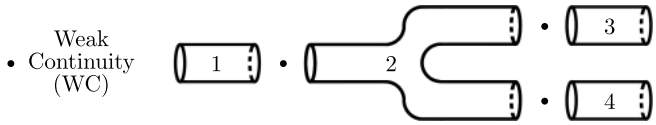


Figure 14: Condensation of the coupling quantities and weak domain decomposition for the problem of Figure 13. This problem is referred to as A_{WC}.

this work and that is solved using coupling strategies $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$, described in Section 2.5. From the mathematical viewpoint, we can define R as the operator that maps the solution computed for the original problem A_{PC}, to the solution for the condensed problem A_{WC}. Hence, we can write

$$A_{\mathrm{PC}} = R^T A_{\mathrm{WC}}.$$

Based on this remark, we can use the solution of problem A_{WC}, which is easier to obtain and whose scalability with respect to the number of subdomains is higher, as a preconditioner for problem A_{PC}. Since all the n_Ω subdomains in Figures 13 and 14 are the same, then there exists a trivial map R from the solution of A_{PC} onto the solution of A_{WC}.

Notice that problem A_{WC} takes into account the long range interactions between the subdomains through a simpler problem. In other words, A_{WC} can play the role of a coarse correction for A_{PC}. Moreover, we highlight that the same idea can be employed to introduce a further preconditioner for problem A_{WC}, leading to a two-level preconditioner strategy. This preconditioner, may be obtained from a zero-dimensional representation of the problem (see [4, 6]), whose solution gives the approximate values of the coupling quantities on the coupling interfaces.

6 Conclusions

In this work, two strategies for the coupling of multiple 3D flow models have been presented. These strategies are based on a different choice for the boundary conditions to be applied at the interfaces of the subdomains. Both of them allow to partition the solution of the 3D Navier–Stokes equations by a network of non overlapping subdomains, which communicate only through the exchange of integrated quantities over the interfaces. The coupling equations at the interfaces

are based just on the conservation of the flow rate and on the continuity of the coupling stress (the dual variable). This choice leads to an approximation of the original problem for the case of 3D Navier–Stokes equations. Moreover it is perfectly suitable for a geometrical multiscale framework, where most of the coupling involves reduced models, such as 0D and 1D models. The numerical schemes employed in this work were the Newton and the Aitken methods. The analysis of the numerical examples conducted here showed that relaxed fixed point methods behave badly when applied to problems arising in multi-branched systems. In contrast, the Newton method is quite robust for the flow regimes studied here and more generally for closed loop network configurations. This dictates the use of Newton-like algorithms in order to build convergent schemes.

One of the most appealing aspects of the methodology devised in this paper is the fact that it allows to solve each subdomain separately, leaving all the interactions at the level of the boundary conditions; hence, the implementation in a parallel framework is straightforward and efficient. In addition, no hierarchy has to be established a priori in the setting of the boundary conditions at the branching points. Finally, we also showed how the same methodology could be employed to build, in a systematic way, a scalable preconditioner for the original strongly coupled domain decomposition problem.

Acknowledgements

We acknowledge the European Research Council Advanced Grant "Mathcard, Mathematical Modelling and Simulation of the Cardiovascular System", Project ERC-2008-AdG 227058. The second author acknowledge the support of the Brazilian agencies CNPq and FAPERJ. All the numerical results presented in this paper have been computed using the LifeV library (`www.lifev.org`). The entire LifeV community is kindly acknowledged.

References

[1] Blanco P, Feijóo R, Urquiza S. A unified variational approach for coupling 3D–1D models and its blood flow applications. *Comp. Meth. Appl. Mech. Engrg.* 2007; **196**(41–44):4391–4410.

[2] Blanco P, Pivello M, Urquiza S, Feijóo R. On the potentialities of 3D–1D coupled models in hemodynamics simulations. *J. Biomech.* 2009; **42**(7):919–930.

[3] Formaggia L, Gerbeau J, Nobile F, Quarteroni A. On the coupling of 3D and 1D Navier–Stokes equations for flow problems in compliant vessels. *Comp. Meth. Appl. Mech. Engrg.* 2001; **191**(6–7):561–582.

[4] Formaggia L, Quarteroni A, Veneziani A. *Cardiovascular Mathematics, Modeling and simulation of the circulatory system*, vol. 1. Springer, 2009.

[5] Kim H, Vignon-Clementel I, Figueroa C, LaDisa J, Jansen K, Feinstein J, Taylor C. On coupling a lumped parameter heart model and a three-dimensional finite element aorta model. *Ann. Biomed. Eng.* 2009; **37**(11):2153–2169.

[6] Migliavacca F, Balossino R, Pennati G, Dubini G, Hsia T, de Leval M, Bove E. Multiscale modelling in biofluidynamics: Application to reconstructive paediatric cardiac surgery. *J. Biomech.* 2006; **39**(6):1010–1020.

[7] Quarteroni A, Veneziani A. Analysis of a geometrical multiscale model based on the coupling of PDE's and ODE's for blood flow simulations. *SIAM J. Multiscale Model. Simul.* 2003; **1**(2):173–195.

[8] Urquiza S, Blanco P, Vénere M, Feijóo R. Multidimensional modelling for the carotid artery blood flow. *Comp. Meth. Appl. Mech. Engrg.* 2006; **195**(33–36):4002–4017.

[9] Vignon-Clementel I, Figueroa C, Jansen K, Taylor C. Outflow boundary conditions for three–dimensional finite element modeling of blood flow and pressure in arteries. *Comp. Meth. Appl. Mech. Engrg.* 2006; **195**(29–32):3776–3796.

[10] Formaggia L, Gerbeau J, Nobile F, Quarteroni A. Numerical treatment of defective boundary conditions for the Navier–Stokes equations. *SIAM J. Numer. Anal.* 2002; **40**(1):376–401.

[11] Veneziani A, Vergara C. Flow rate defective boundary conditions in haemodynamics simulations. *Int. J. Num. Meth. Fluids* 2005; **47**(8–9):803–816.

[12] Blanco P, Feijóo R, Urquiza S. A variational approach for coupling kinematically incompatible structural models. *Comp. Meth. Appl. Mech. Engrg.* 2008; **197**(17–18):1577–1602.

[13] Veneziani A, Vergara C. An approximate method for solving incompressible Navier–Stokes problems with flow rate conditions. *Comp. Meth. Appl. Mech. Engrg.* 2007; **196**(9–12):1685–1700.

[14] Leiva J, Blanco P, Buscaglia G. Iterative strong coupling of dimensionally–heterogeneous models. *Int. J. Num. Meth. Engng.* 2010; **81**(12):1558–1580.

[15] Blanco P, Leiva J, Buscaglia G. Black-box decomposition approach for computational hemodynamics: One-dimensional models. *Submitted to Comp. Meth. Appl. Mech. Engrg.* 2010; .

[16] Ivanova K, Bournaski E. Combined distributed and lumped parameters model for transient flow analysis in complex pipe networks. *Comp. Meth. Appl. Mech. Engrg.* 1996; **130**(1–2):47–56.

[17] Kiuchi T. An implicit method for transient gas flows in pipe networks. *Int. J. Heat Fluid Fl.* 1994; **15**(5):378–383.

[18] Marin-Artieda C, Dargush G. Approximate limit load evaluation of structural frames using linear elastic analysis. *Eng. Struct.* 2007; **29**(3):296–304.

[19] Yang Y, Kuob S, Wu Y. Incrementally small-deformation theory for nonlinear analysis of structural frames. *Eng. Struct.* 2002; **24**(6):783–798.

[20] Formaggia L, Lamponi D, Quarteroni A. One-dimensional models for blood flow in arteries. *J. Eng. Math.* 2003; **47**(3–4):251–276.

[21] Deparis S. Numerical analysis of axisymmetric flows and methods for fluid-structure interaction arising in blood flow simulation. PhD Thesis, EPFL 2004.

[22] Quarteroni A, Sacco R, Saleri F. *Numerical Mathematics*, vol. 37. Springer-Verlag, New York, 2000.

[23] Womersley J. The mathematical analysis of the arterial circulation in a state of oscillatory motion. *Technical Report WADC-TR-56-614*, Wright Air Dev. Center 1957.

[24] Burman E, Fernández M, Hansbo P. Continuous interior penalty finite element method for Oseen's equations. *SIAM J. Numer. Anal.* 2006; **44**(3):1248–1274.

[25] Deparis S, Discacciati M, Quarteroni A. *A domain decomposition framework for fluid-structure interaction problems*. Computational Fluid Dynamics, Springer Berlin Heidelberg, 2004.

[26] Hughes T, Lubliner J. On the one-dimensional theory of blood flow in the larger vessels. *Math. Biosci.* 1973; **18**(1-2):161–170.

MOX Technical Reports, last issues

Dipartimento di Matematica "F. Brioschi",
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

32/2010 A. Cristiano I. Malossi, Pablo J. Blanco, Simone Deparis, Alfio Quarteroni:
Algorithms for the partitioned solution of weakly coupled fluid models for cardiovascular flows

31/2010 Andrea Manzoni, Alfio Quarteroni, Gianluigi Rozza:
Shape optimization for viscous flows by reduced basis methods and free-form deformation

30/2010 Pablo J. Blanco, Marco Discacciati, Alfio Quarteroni:
Modeling dimensionally-heterogeneous problems: analysis, approximation and applications

29/2010 Matteo Lesinigo, Carlo D'Angelo, Alfio Quarteroni:
A multiscale Darcy-Brinkman model for fluid flow in fractured porous media

28/2010 Paolo Crosetto, Philippe Reymond, Simone Deparis, Dimitrios Kontaxakis, Nikolaos Stergiopulos, Alfio Quarteroni:
Fluid Structure Interaction Simulations of Physiological Blood Flow in the Aorta

27/2010 Matteo Bruggi, Marco Verani:
An adaptive algorithm for topology optimization with goal-oriented error control

26/2010 Francesca Ieva, Anna Maria Paganoni:
Designing and mining a multicenter observational clinical registry concerning patients with Acute Coronary Syndromes

25/2010 G. Pena, C. Prud'homme, Alfio Quarteroni:
High Order Methods for the Approximation of the Incompressible Navier-Stokes Equations in a Moving Domain

24/2010 Lorenzo Tamellini, Luca Formaggia, Edie Miglio, Anna Scotti:
An Uzawa iterative scheme for the simulation of floating boats

23/2010 Joakim Baeck, Fabio Nobile,
Lorenzo Tamellini, Raul Tempone:
Stochastic Spectral Galerkin and collocation methods for PDEs with random coefficients: a numerical comparison