Computers and Fluids 141 (2016) 22–41

Contents lists available at ScienceDirect

Computers and Fluids

journal homepage: www.elsevier.com/locate/compfluid

INTERNODES: an accurate interpolation-based method for coupling the Galerkin solutions of PDEs on subdomains featuring non-conforming interfaces

Simone Deparis[a], Davide Forti[a,*], Paola Gervasio[b], Alfio Quarteroni[a,1]

[a] CMCS-MATHICSE, École Polytechnique Fédérale de Lausanne, Station 8, CH-1015 Lausanne, Switzerland
[b] DICATAM, Università degli Studi di Brescia, via Branze 38, Brescia I-25123, Italy

ARTICLE INFO

Article history:
Received 9 September 2015
Revised 15 February 2016
Accepted 28 March 2016
Available online 31 March 2016

Keywords:
Domain decomposition
Non-conforming discretization
Interpolation
Finite element method
Spectral element method
mortar method

ABSTRACT

We are interested in the approximation of partial differential equations on domains decomposed into two (or several) subdomains featuring non-conforming interfaces. The non-conformity may be due to different meshes and/or different polynomial degrees used from the two sides, or even to a geometrical mismatch. Across each interface, one subdomain is identified as master and the other as slave. We consider Galerkin methods for the discretization (such as finite element or spectral element methods) that make use of two interpolants for transferring information across the interface: one from master to slave and another one from slave to master. The former is used to ensure continuity of the primal variable (the problem solution), while the latter that of the dual variable (the normal flux). In particular, since the dual variable is expressed in weak form, we first compute a strong representation of the dual variable from the slave side, then interpolate it, transform the interpolated quantity back into weak form and eventually assign it to the master side. In case of slightly non-matching geometries, we use a radial-basis function interpolant instead of Lagrange interpolant.

The proposed method is named INTERNODES (INTERpolation for NOnconforming DEcompositionS): it can be regarded as an alternative to the mortar element method and it is simpler to implement in a numerical code. We show on two dimension al problems that by using the Lagrange interpolation we obtain at least as good convergence results as with the mortar element method with any order of polynomials. When using low order polynomials, the radial-basis interpolant leads to the same convergence properties as the Lagrange interpolant. We conclude with a comparison between INTERNODES and a standard conforming approximation in a three dimensional case.

1. Introduction

In this paper we propose a new approach for numerically solving elliptic partial differential equations by Galerkin methods on computational domains that are split into two (or several) subdomains featuring "non-conforming interfaces". By this we mean that either a priori independent grids and/or local polynomial degrees are used to discretize each subdomain. More in particular, we refer to these two cases as "grid non-conformity" and "polynomial non-conformity", respectively. A further possible case of non-conforming interfaces that our approach can cover is that of subdomains that face each other through two interfaces that geometrically do not fully agree one another, meaning that the two subdomains may either slightly overlap and/or featuring tiny holes between them (see Fig. 1). We refer to this situation of geometrical mismatch as "geometric non-conformity". It may arise when using CAD to generate the two subdomains, e.g. in fluid-structure interaction problems in hydrodynamics or aerodynamics [37,41], or else when generating the computational geometries of lumen and vessel walls from DYCOM images for arterial blood flow dynamics [29,53].

In all these cases of non-conforming interfaces, a very crucial issue is the way the subdomain solutions communicate across common interfaces. More specifically, at which extent the subdomain solutions and their normal fluxes do match. In this paper we introduce a new method, named *INTERNODES*, that we are going to describe. Across each interface, the subdomain from one side is identified as the master while that from the opposite side as

* Corresponding author. Fax: +41 216935510.
E-mail address: davide.forti@epfl.ch (D. Forti).
[1] On leave from MOX, Politecnico di Milano, via Bonardi 9, Milano 20133, Italy.

http://dx.doi.org/10.1016/j.compfluid.2016.03.033
0045-7930/

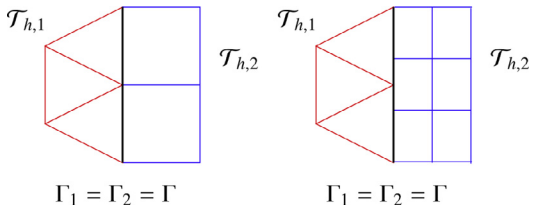


Fig. 1. A situation with non-matching interfaces Γ_1 and Γ_2 in the case $d = 2$.

the slave. Two different interpolants are then defined for transferring information across the interface: one from master to slave and another one from slave to master. The former is used to ensure continuity of the primal variable (the problem solution), while the latter for ensuring the continuity of the dual variable (the normal flux). In each subdomain we approximate the given partial differential equation by a Galerkin projection method. In this context, the dual variable in the slave domain is available in weak form: we first compute a strong representation of the dual variable from the slave side, interpolate it, transform the interpolated quantity back into weak form and eventually assign the function so obtained to the master side. This treatment of the normal flux continuity is a distinguishing feature of our method.

The interpolants characterizing the INTERNODES construction can be arbitrary: in this paper we use either Lagrangian or Radial Basis Functions (RBF) interpolants. In Section 7, we show on two dimensional problems that using the Lagrange interpolation, we can attain at least as good convergence results as using the mortar element method. This is also true when using RBF interpolants with low polynomial degrees.

The mortar element method, formerly introduced by [10,11] (see also [4,5,12,35,57]) represents nowadays a gold standard for the solution of PDEs using non-conforming discretizations. The INTERNODES method, being based on interpolation rather than on L^2 projection at interfaces, is in general simpler to implement than the mortar method: it only requires separate mass matrices and not the cross mass matrix that connects interface basis functions from both sides. It does not require any ad hoc numerical quadrature, neither special treatment of cross-points where more than two subdomains meet. These issues will be further elaborated in Section 6. INTERNODES with RBF interpolants is also very well suited to address geometric non-conformity [22].

The paper is organized as follows. After introducing the elliptic boundary value problem in Section 2, we formulate in Section 3 its Galerkin discretization based on either the Finite Element Method (FEM) or the Spectral Element Method (SEM), using non-conforming interfaces. In Section 4 we build the slave-to-master and master-to-slave interface intergrid operators (either Lagrangian or RBF based). In Section 5 we formulate the INTERNODES method: we first state it in algebraic terms, then we provide a variational interpretation as a non-conforming generalized Galerkin approximation to the original elliptic boundary-value problem. Section 6 is devoted to an analysis of similarities and differences between INTERNODES and the mortar method (both theoretical properties and algorithmic aspects are evaluated). In Section 7 we perform a systematic comparison of the numerical results that are obtained for FEM–FEM, FEM–SEM and SEM–SEM couplings, when approximating the Dirichlet problem in a 2D domain. The same problem is addressed in Section 8 for the case of geometric non-conformity. More realistic applications are considered in Sections 9 and 10 where we treat very severe (polynomial and mesh) non-conformity for the simulation of external flows in two and three dimensions. In particular, the latter simulation concerns the incompressible Navier–Stokes equations: the INTERNODES method has been easily extended to this case by simply replacing normal fluxes with normal Cauchy stresses in the new context. Conclusions are drawn in Section 11.

2. Problem setting

Let $\Omega \subset \mathbb{R}^d$, with $d = 2,\ 3$, be an open domain with Lipschitz boundary $\partial\Omega$. $\partial\Omega_N$ and $\partial\Omega_D$ are suitable disjoint subsets of $\partial\Omega$ such that $\overline{\partial\Omega_D} \cup \overline{\partial\Omega_N} = \partial\Omega$. Given $f \in L^2(\Omega)$, $g_D \in H^{1/2}(\partial\Omega_D)$, $g_N \in H^{-1/2}(\partial\Omega_N)$, $\mu, \alpha \in L^\infty(\Omega)$ such that $\exists \mu_0 > 0$, $\mu \geq \mu_0$, $\alpha \geq 0$, and $\mathbf{b} \in W^{1,\infty}(\Omega)$ s.t. $\alpha - \frac{1}{2}\nabla \cdot \mathbf{b} \geq 0$ we look for the solution u of the second order elliptic equation

$$\begin{cases} Lu \equiv -\nabla \cdot (\mu \nabla u) + \mathbf{b} \cdot \nabla u + \alpha u = f & \text{in } \Omega, \\ u = g_D & \text{on } \partial\Omega_D, \\ \mu \dfrac{\partial u}{\partial n} = g_N & \text{on } \partial\Omega_N, \end{cases} \tag{1}$$

being n the outward unit normal vector to $\partial\Omega$.

By setting $V = H^1_{\partial\Omega_D}(\Omega) = \{v \in H^1(\Omega) :\ v|_{\partial\Omega_D} = 0\}$, the weak form of problem (1) reads: find $u \in H^1(\Omega)$, with $u = g_D$ on $\partial\Omega_D$, such that

$$a(u, v) = (f, v)_\Omega + \langle g_N, v \rangle_{\partial\Omega_N} \qquad \forall v \in V, \tag{2}$$

where

$$a(u, v) = \int_\Omega (\mu \nabla u \cdot \nabla v + (\mathbf{b} \cdot \nabla u)v + \alpha u v) d\Omega, \tag{3}$$

while $(\cdot,\cdot)_\Omega$ and $\langle \cdot,\cdot \rangle_{\partial\Omega_N}$ denote the inner product in $L^2(\Omega)$ and the duality pairing between $H^{1/2}(\partial\Omega_N)$ and $H^{-1/2}(\partial\Omega_N)$, respectively.

For the sake of exposition we partition Ω into two non-overlapping subdomains Ω_1 and Ω_2 such that $\overline{\Omega} = \overline{\Omega_1 \cup \Omega_2}$; we call one *master* (say Ω_1) and the other *slave* (say Ω_2), and we set $\partial\Omega_{D,k} = \partial\Omega_D \cap \partial\Omega_k$ and $\partial\Omega_{N,k} = \partial\Omega_N \cap \partial\Omega_k$, for $k = 1, 2$.

3. Discretization

A-priori independent discretizations of either finite element type (FEM) or spectral element type (SEM) are designed in Ω_1 and Ω_2 [18,50]. SEM will be equivalently named *hp*-FEM (see [55]).

We denote by $\mathcal{T}_{h,k}$ (for $k = 1, 2$) the meshes induced by the discretization in Ω_k and we assume that they satisfy standard regularity requirements (see [50]). In both Ω_k ($k = 1, 2$) we introduce the finite elements approximation spaces

$$X^{p_k}_{h_k} = \{v \in C^0(\overline{\Omega}_k) :\ v_{|T} \in Q_{p_k},\ \forall T \in \mathcal{T}_{h,k}\}, \tag{4}$$

where $Q_{p_k} = \mathbb{P}_{p_k}$ in the simplicial case and $Q_{p_k} = \mathbb{Q}_{p_k} \circ \mathbf{F}_T^{-1}$ for quads, being $\mathbf{F}_T$ the C^1 diffeomorphism that maps the reference element $\hat{T}$ into the generic element $T \in \mathcal{T}_{h,k}$ [50]. For any $T \in \mathcal{T}_{h,k}$, we assume that $\partial T \cap \partial\Omega$ fully belongs to either $\partial\Omega_D$ or $\partial\Omega_N$.

For $k = 1, 2$, we introduce the finite dimensional subspaces $V_{k,\delta}$ and $V^0_{k,\delta}$ of $V_k = H^1_{\partial\Omega_{D,k}}(\Omega_k)$, where δ stands for discretization, more precisely

$$V_{k,\delta} = X^{p_k}_{h_k} \cap V_k, \qquad V^0_{k,\delta} = \{v \in V_{k,\delta}, v|_\Gamma = 0\}. \tag{5}$$

On the interface $\Gamma = \partial\Omega_1 \cap \partial\Omega_2$, we allow the meshes $\mathcal{T}_{h,1}$ and $\mathcal{T}_{h,2}$ to induce either *conforming grids* (like in Fig. 2, left) or *non-conforming grids* (as in Fig. 2, right). More precisely, the meshes are conforming if their restrictions to Γ coincide; otherwise they are non-conforming.

Another situation we would like to address is the one when different polynomial degrees are used on the two subdomains (called *polynomial non-conformity*), or again the case in which FEM on simplicials from one side is coupled with SEM on quads from the other side, with different polynomial degrees. For example, we reference to Fig. 2, left, using FEM in Ω_1 and SEM in Ω_2.

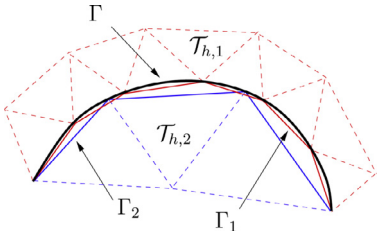


Fig. 2. Conforming (at left) and non-conforming (at right) grids at the interface when $d = 2$.

Finally, in some situations, while discretizing Γ, we could even end up with two non-matching interfaces, that is $\Gamma_1 \neq \Gamma_2$. We refer to this last situation as *non-matching interfaces* or *geometrically non-conforming* partitions. A possible instance is when Γ is a curved line that is discretized by piecewise straight segments, see Fig. 1. Another instance may occur when using isogeometric analysis (see [19]).

In order to unify our theory for both the cases of matching and non-matching interfaces, from now on we will refer to Γ_1 and Γ_2 separately, understanding that $\Gamma_1 = \Gamma_2 = \Gamma$ in the geometrically conforming case.

Generally speaking, we call *non-conforming* a situation where one (or several) of the previous cases (non-conforming grids, polynomial non-conformity, non-matching interfaces) arises.

4. Intergrid operators

For $k = 1, 2$, let us introduce the discrete trace functional spaces

$$\Lambda_{k,\delta} = \{\varphi = v|_{\Gamma_k},\ v \in X_{h_k}^{p_k}\}, \quad n_k = \dim(\Lambda_{k,\delta}), \tag{6}$$

$$\Lambda_{k,\delta}^0 = \{\varphi \in \Lambda_{k,\delta} :\ \varphi|_{\partial\Gamma} = 0\} \subseteq \Lambda_{k,\delta}, \quad n_k^0 = \dim(\Lambda_{k,\delta}^0), \tag{7}$$

and denote by $\{\lambda_j^{(k)}\}_{j=1}^{n_k}$ the Lagrange basis of $\Lambda_{k,\delta}$ associated with the nodes $\mathbf{x}_i^{(\Gamma_k)} \in \Gamma_k$ (for $i = 1, \ldots, n_k$) induced by the mesh $\mathcal{T}_{h,k}$. Notice that $\Lambda_{k,\delta}^0 = \Lambda_{k,\delta}$ when $dist(\Gamma, \partial\Omega_D) > 0$. As a matter of fact, $\Lambda_{k,\delta}^0$ does not include the Lagrange basis functions of $\Lambda_{k,\delta}$ associated with the nodes of $\Gamma_k \cap \partial\Omega_D$.

For ease of notation, we suppose that the n_k^0 basis functions of $\Lambda_{k,\delta}^0$ coincide with the first n_k^0 basis functions of $\Lambda_{k,\delta}$, thus the functions $\lambda_i^{(k)}$, for $i = n_k^0 + 1, \ldots, n_k$, are those associated with the nodes of $\Gamma_k \cap \partial\Omega_D$.

To define our method we need to introduce intergrid transfer operators, extension operators, and an interface space.

We introduce two linear operators

$$\Pi_{12} : \Lambda_{2,\delta} \to \Lambda_{1,\delta}, \qquad \Pi_{21} : \Lambda_{1,\delta} \to \Lambda_{2,\delta} \tag{8}$$

that realize the *intergrid transfer*. We consider two different instances:

1. Lagrange interpolation,
2. Radial Basis Function (RBF) interpolation [16,56], in particular RL-RBF [22].

For reader's convenience, we define here the interpolation operators.

The Lagrange interpolation operator Π_{21} is characterized as follows. Let us consider a function $\eta_{1,\delta} \in \Lambda_{1,\delta}$, then $\Pi_{21}\eta_{1,\delta}$ can be written w.r.t. the basis $\{\lambda_i^{(2)}\}$ of $\Lambda_{2,\delta}$ as

$$\left(\Pi_{21}\eta_{1,\delta}\right)(\mathbf{x}) = \sum_{i=1}^{n_2} \eta_{1,\delta}\left(\mathbf{x}_i^{(\Gamma_2)}\right)\lambda_i^{(2)}(\mathbf{x}), \quad \forall \mathbf{x} \in \Gamma_2. \tag{9}$$

By expanding $\eta_{1,\delta}$ with respect to the basis functions $\lambda_j^{(1)}$ of $\Lambda_{1,\delta}$ we have

$$\eta_{1,\delta}(\mathbf{x}) = \sum_{j=1}^{n_1} \eta_{1,\delta}(\mathbf{x}_j^{(\Gamma_1)})\lambda_j^{(1)}(\mathbf{x}) \qquad \forall \mathbf{x} \in \Gamma_1,$$

and then, for any $\mathbf{x} \in \Gamma_2$,

$$\left(\Pi_{21}\eta_{1,\delta}\right)(\mathbf{x}) = \sum_{i=1}^{n_2}\left(\sum_{j=1}^{n_1} \eta_{1,\delta}\left(\mathbf{x}_j^{(\Gamma_1)}\right)\lambda_j^{(1)}\left(\mathbf{x}_i^{(\Gamma_2)}\right)\right)\lambda_i^{(2)}(\mathbf{x}).$$

Finally, denoting by $\boldsymbol{\eta}_1$ the array in $\mathbb{R}^{n_1}$ whose components are the nodal values $\eta_{1,\delta}(\mathbf{x}_i^{(\Gamma_1)})$, for $i = 1, \ldots, n_1$, and by

$$(R_{21})_{ij} = \lambda_j^{(1)}\left(\mathbf{x}_i^{(\Gamma_2)}\right), \quad i = 1, \ldots, n_2,\ j = 1, \ldots, n_1, \tag{10}$$

the entries of the matrix associated with the operator Π_{21}, we can write

$$(\Pi_{21}\eta_{1,\delta})(\mathbf{x}_i^{(\Gamma_2)}) = (R_{21}\boldsymbol{\eta}_1)_i, \quad i = 1, \ldots, n_2.$$

By proceeding in a similar way for Π_{12}, we denote the entries of the matrix associated with the operator Π_{12} by

$$(R_{12})_{ij} = \lambda_j^{(2)}\left(\mathbf{x}_i^{(\Gamma_1)}\right), \quad i = 1, \ldots, n_1,\ j = 1, \ldots, n_2, \tag{11}$$

so that

$$(\Pi_{12}\eta_{2,\delta})(\mathbf{x}_i^{(\Gamma_1)}) = (R_{12}\boldsymbol{\eta}_{2,\delta})_i, \quad i = 1, \ldots, n_1.$$

The RL-RBF interpolation operators are defined as in [22] and they read

$$(\Pi_{21}\eta_{1,\delta})(\mathbf{x}) = \frac{\sum_{i=1}^{n_1} \gamma_i^{\eta_{1,\delta}}\phi(\|\mathbf{x} - \mathbf{x}_i^{(\Gamma_1)}\|, r_i)}{\sum_{i=1}^{n_1} \gamma_i^{1}\phi(\|\mathbf{x} - \mathbf{x}_i^{(\Gamma_1)}\|, r_i)}, \tag{12}$$

$$(\Pi_{12}\eta_{2,\delta})(\mathbf{x}) = \frac{\sum_{i=1}^{n_2} \gamma_i^{\eta_{2,\delta}}\phi(\|\mathbf{x} - \mathbf{x}_i^{(\Gamma_2)}\|, r_i)}{\sum_{i=1}^{n_2} \gamma_i^{1}\phi(\|\mathbf{x} - \mathbf{x}_i^{(\Gamma_2)}\|, r_i)}, \tag{13}$$

where ϕ is the locally supported C^2 Wendland radial basis function [56], $r_i \in \mathbb{R}$ is the local support of the basis function, and γ_i^f are the weights of the interpolant of the function f ($f \equiv 1$ denotes the constant function $f(x) = 1$) and they are determined by imposing the interpolation constraints at either the nodes $\mathbf{x}_i^{(\Gamma_2)}$ ($i = 1, \ldots, n_2$) for Π_{21}, or at $\mathbf{x}_i^{(\Gamma_1)}$ ($i = 1, \ldots, n_1$) for Π_{12}.

Then, for $k = 1,\ 2$ we define two linear and continuous *extension operators*

$$E_k : \Lambda_{k,\delta} \to X_{h_k}^{p_k}, \quad s.t. \quad (E_k\lambda^{(k)})|_{\Gamma_k} = \lambda^{(k)}, \tag{14}$$

that extend any $\lambda^{(k)} \in \Lambda_{k,\delta}$ by setting to zero the values of $E_k\lambda^{(k)}$ at all nodes of $\mathcal{T}_{h,k}$ not belonging to Γ_k.

In particular, for any Lagrange basis function $\lambda_j^{(k)}$ of $\Lambda_{k,\delta}$, $E_k\lambda_j^{(k)}$ is the Lagrange basis function of $X_{h_k}^{p_k}$ (associated with the nodes of the mesh $\mathcal{T}_{h,k}$) whose restriction on Γ_k coincides with $\lambda_j^{(k)}$. It follows that $E_k\lambda_j^{(k)} \in V_{k,\delta}$ (i.e., it satisfies homogeneous Dirichlet boundary conditions) only if $\lambda_j^{(k)} \in \Lambda_{k,\delta}^0$, or equivalently, when $j = 1, \ldots, n_k^0$.

The *interface* space is a space of functions defined in the whole $\overline{\Omega}$ as follows:

$$V_{\Gamma_1} = \{\varphi \in L^2(\Omega) : \exists\lambda_1 \in \Lambda_{1,\delta}^0 : \varphi|_{\Omega_1} = E_1\lambda_1, \ \varphi|_{\Omega_2} = E_2(\Pi_{21}\lambda_1)\}. \tag{15}$$

Note that its definition depends on the choice of the master Ω_1 and slave Ω_2 domain. We denote a basis of V_{Γ_1} as $\{\lambda_j^e\}_{j=1}^{n_1^0}$, where "$e$" stands for "extension". There is a one-to-one map between $\lambda_j^{(1)}$

(the jth basis function of $\Lambda^0_{1,\delta}$) and λ^e_j (i.e., $\lambda^e_j|_{\Gamma_1} = \lambda^{(1)}_j$), and λ^e_j satisfies $\lambda^e_j|_{\Omega_1} = E_1\lambda^{(1)}_j$ and $\lambda^e_j|_{\Omega_2} = E_2(\Pi_{21}\lambda^{(1)}_j)$.

Then, we define the subspaces of V

$$\widetilde{V}_{k,\delta} = \{v \in V : v|_{\Omega_k} \in V^0_{k,\delta} \text{ and } v|_{\Omega\setminus\Omega_k} = 0\}, \quad N^0_k = \dim(\widetilde{V}_{k,\delta}) \tag{16}$$

and we indicate their Lagrange basis as $\{\Phi^{(k)}_i\}$ for $i = 1, \ldots, N^0_k$ and for $k = 1, 2$.

Thus we set

$$V_\delta = \widetilde{V}_{1,\delta} \oplus \widetilde{V}_{2,\delta} \oplus V_{\Gamma_1}. \tag{17}$$

Notice that $V_\delta \not\subset V$ in general.

5. Formulation of the non-conforming problem

We define the bilinear forms $a_k : H^1(\Omega_k) \times H^1(\Omega_k) \to \mathbb{R}$:

$$\begin{aligned} a_1(u,v) &= \int_{\Omega_1} (\mu\nabla u \cdot \nabla v + (\mathbf{b}\cdot\nabla u)v + \alpha uv)d\Omega\,, \\ a_2(u,v) &= \int_{\Omega_2} (\mu\nabla u \cdot \nabla v + (\mathbf{b}\cdot\nabla u)v + \alpha uv)d\Omega \\ &\quad - \int_{\partial\Omega_{D,2}} \mu\frac{\partial u}{\partial n}v\; d\partial\Omega. \end{aligned} \tag{18}$$

where the last integral in the definition of $a_2(u,v)$ has to be intended as a duality. The presence of the boundary integral in a_2 will be justified later, see Remark 5.1. If we assume that $g_D \in C^0(\overline{\partial\Omega_D})$, its liftings $R_k g_D \in X^{p_k}_{h_k}$ are

$$\begin{aligned} R_1 g_D(\mathbf{x}) &= \sum_{i=1}^{N^D_1} g(\mathbf{x}^{(1)}_{i,D})\Phi^{(1)}_{i,D}(\mathbf{x}) + \sum_{i=n^0_1+1}^{n_1} g(\mathbf{x}^{(\Gamma_1)}_i)E_1\lambda^{(1)}_i(\mathbf{x}), \\ &\qquad \forall \mathbf{x} \in \overline{\Omega}_1\,, \\ R_2 g_D(\mathbf{x}) &= \sum_{i=1}^{N^D_2} g(\mathbf{x}^{(2)}_{i,D})\Phi^{(2)}_{i,D}(\mathbf{x}) + \sum_{i=n^0_1+1}^{n_1} g(\mathbf{x}^{(\Gamma_1)}_i)E_2(\Pi_{21}\lambda^{(1)}_i(\mathbf{x})), \\ &\qquad \forall \mathbf{x} \in \overline{\Omega}_2\,, \end{aligned} \tag{19}$$

where $\Phi^{(k)}_{i,D} \in X^{p_k}_{h_k}$ (for $i = 1, \ldots, N^D_k$) denote the Lagrange basis functions associated with the nodes $\mathbf{x}^{(k)}_{i,D} \in \mathcal{T}_{h,k} \cap (\partial\Omega_{D,k} \setminus \Gamma_k)$, while $\lambda^{(1)}_i$ for $i = n^0_1 + 1, \ldots, n_1$ are the Lagrange basis functions associated with the nodes in $\mathcal{T}_{h,1} \cap (\Gamma_1 \cap \partial\Omega_{D,1})$. Finally we define the linear functionals as follows

$$\mathcal{F}_k(v) = (f,v)_{\Omega_k} + \langle g_N, v\rangle_{\partial\Omega_{N,k}} - a_k(R_k g_D, v), \quad \forall v \in H^1(\Omega_k)\,. \tag{20}$$

Then, we set the matrices

$$\begin{aligned} (A_{kk})_{ij} &= a_k(\Phi^{(k)}_j, \Phi^{(k)}_i), \quad i,j = 1, \ldots, N^0_k, \\ (A_{k,\Gamma_k})_{ij} &= a_k(E_k\lambda^{(k)}_j, \Phi^{(k)}_i), \quad i = 1, \ldots, N^0_k,\ j = 1, \ldots, n_k, \\ A^0_{k,\Gamma_k} &= [A_{k,\Gamma_k}]_{i=1,\ldots,N^0_k,\ j=1,\ldots,n^0_k} \\ (A_{\Gamma_k,k})_{ij} &= a_k(\Phi^{(k)}_j, E_k\lambda^{(k)}_i), \quad i = 1, \ldots, n_k,\ j = 1, \ldots, N^0_k, \\ A^0_{\Gamma_k,k} &= [A_{\Gamma_k,k}]_{i=1,\ldots,n^0_k,\ j=1,\ldots,N^0_k} \\ (A_{\Gamma_k,\Gamma_k})_{ij} &= a_k(E_k\lambda^{(k)}_j, E_k\lambda^{(k)}_i), \quad i,j = 1, \ldots, n_k, \\ A^0_{\Gamma_k,\Gamma_k} &= [A_{\Gamma_k,\Gamma_k}]_{i,j=1,\ldots,n^0_k} \end{aligned} \tag{21}$$

the vectors

$$\begin{aligned} \mathbf{f}^{\mathbf{0}}_{\mathbf{k}} &= \mathcal{F}_{\mathbf{k}}(\mathbf{\Phi}^{(\mathbf{k})}_{\mathbf{i}}), \qquad i = 1, \ldots, N^0_k, \\ \mathbf{f}_{\Gamma_k} &= \mathcal{F}_k(E_k\lambda^{(k)}_i), \qquad i = 1, \ldots, n_k, \\ \mathbf{f}^0_{\Gamma_k} &= \left[\mathbf{f}_{\Gamma_k}\right]_{i=1,\ldots,n^0_k}, \end{aligned} \tag{22}$$

and we set

$$R_\delta g_D = \begin{cases} R_1 g_D & \text{in } \Omega_1 \\ R_2 g_D & \text{in } \Omega_2. \end{cases} \tag{23}$$

In the special case of fully conforming discretizations (that is both grid and polynomial conformity, with $\Gamma_1 = \Gamma_2 = \Gamma$ and $n_1 = n_2$), the classical conforming Galerkin approximation of (1) is defined as follows: find $u_\delta \in H^1(\Omega)$, such that $(u_\delta - R_\delta g_D) \in V_\delta \subset V$, solution of the conforming Galerkin problem

$$a(u_\delta, v_\delta) = (f, v_\delta)_\Omega + \langle g_N, v_\delta\rangle_{\partial\Omega_N} \quad \forall v_\delta \in V_\delta. \tag{24}$$

In view of both (21) and (22), the well-known algebraic domain decomposition form of (24) reads [51]

$$\begin{bmatrix} A_{1,1} & 0 & A^0_{1,\Gamma_1} \\ 0 & A_{2,2} & A^0_{2,\Gamma_2} \\ A^0_{\Gamma_1,1} & A^0_{\Gamma_2,2} & A^0_{\Gamma_1,\Gamma_1} + A^0_{\Gamma_2,\Gamma_2} \end{bmatrix} \begin{bmatrix} \mathbf{u}^{\mathbf{0}}_{\mathbf{1}} \\ \mathbf{u}^0_2 \\ \mathbf{u}^0_{\Gamma_1} \end{bmatrix} = \begin{bmatrix} \mathbf{f}^{\mathbf{0}}_{\mathbf{1}} \\ \mathbf{f}^0_2 \\ \mathbf{f}^0_{\Gamma_1} + \mathbf{f}^0_{\Gamma_2} \end{bmatrix}, \tag{25}$$

where, for $k = 1, 2$,

$$\mathbf{u}^0_k = [u_{k,\delta}(\mathbf{x}^{(k)}_j)]^{N^0_k}_{j=1}\ (k = 1,2), \quad \text{and} \quad \mathbf{u}^0_{\Gamma_1} = [u_{1,\delta}(\mathbf{x}^{(\Gamma_1)}_j)]^{n^0_1}_{j=1}, \tag{26}$$

are the array of the nodal values of $u_{k,\delta} = u_\delta|_{\Omega_k}$ at the nodes of $\overline{\Omega}_k \setminus (\Gamma_k \cup \partial\Omega_{D,k})$, and the array of the nodal values of $u_\delta|_{\Gamma_1}$ at the nodes of $\Gamma_1 \setminus \partial\Omega_D$, respectively.

Denoting by $\mathbf{g}_k$ and $\mathbf{g}_{\Gamma_k}$ the arrays of the nodal values $g_D(\mathbf{x}^{(k)}_{i,D})$ for $i = 1, \ldots, N^D_k$ and $g_D(\mathbf{x}^{(\Gamma_k)}_i)$ for $i = n^0_k + 1, \ldots, n_k$, respectively, the solutions arrays including the Dirichlet nodal values are

$$\mathbf{u}_k = \begin{bmatrix} \mathbf{u}^{\mathbf{0}}_{\mathbf{k}} \\ \mathbf{g}_k \end{bmatrix}, \qquad \mathbf{u}_{\Gamma_k} = \begin{bmatrix} \mathbf{u}^{\mathbf{0}}_{\mathbf{\Gamma_k}} \\ \mathbf{g}_{\Gamma_k} \end{bmatrix}.$$

5.1. Algebraic formulation of the INTERNODES method

In the non-conforming case we need further matrices: the local mass matrices associated with the interfaces, that is

$$(M_{\Gamma_k})_{ij} = (\lambda^{(k)}_i, \lambda^{(k)}_j)_{L^2(\Gamma_k)}, \qquad i,j = 1, \ldots, n_k, \quad k = 1,2, \tag{27}$$

and the matrices $R_{12} \in \mathbb{R}^{n_1\times n_2}$ and $R_{21} \in \mathbb{R}^{n_2\times n_1}$ defined in (10) and (11), respectively. Finally, by setting

$$Q_{21} = R_{21}, \qquad Q_{12} = M_{\Gamma_1} R_{12} M^{-1}_{\Gamma_2}, \tag{28}$$

and denoting by $Q^0_{12} \in \mathbb{R}^{n^0_1\times n_2}$ the submatrix of Q_{12} of its first n^0_1 rows, and by $Q^0_{21} \in \mathbb{R}^{n_2\times n^0_1}$ the submatrix of Q_{21} of its first n^0_1 columns, the *non-conforming* generalization of (25) reads

$$\begin{bmatrix} A_{1,1} & 0 & A^0_{1,\Gamma_1} \\ 0 & A_{2,2} & A_{2,\Gamma_2}Q^0_{21} \\ A_{\Gamma_1,1} & Q^0_{12}A_{\Gamma_2,2} & A^0_{\Gamma_1,\Gamma_1} + Q^0_{12}A_{\Gamma_2,\Gamma_2}Q^0_{21} \end{bmatrix} \begin{bmatrix} \mathbf{u}^{\mathbf{0}}_{\mathbf{1}} \\ \mathbf{u}^0_2 \\ \mathbf{u}^0_{\Gamma_1} \end{bmatrix} = \begin{bmatrix} \mathbf{f}^{\mathbf{0}}_{\mathbf{1}} \\ \mathbf{f}^0_2 \\ \mathbf{f}^0_{\Gamma_1} + Q^0_{12}\mathbf{f}_{\Gamma_2}. \end{bmatrix}. \tag{29}$$

Notice that, in the fully conforming case, Q_{12} and Q_{21} coincide with the identity matrix of size $n_1 = n_2$, thus $A_{2,\Gamma_2}Q^0_{21} = A^0_{2,\Gamma_2}$, $Q^0_{12}A_{\Gamma_2,2} = A^0_{\Gamma_2,2}$, and $Q^0_{12}A_{\Gamma_2,\Gamma_2}Q^0_{21} = A^0_{\Gamma_2,\Gamma_2}$ (and (29) returns (25)).

The sketch of the algorithm is reported for reader's convenience in Algorithm 1.

Algorithm 1 INTERNODES algorithm.

1: Build the local stiffness matrices $A_{k,k}$, A_{k,Γ_k}, and$A_{\Gamma_k,k}$ (formula (21)),
2: Buildthe right hand sides $\mathbf{f}_k$ and $\mathbf{f}_{\Gamma_k}$ for $k = 1, 2$ (formula (22)),
3: Build the local interface mass matrices M_{Γ_k}, for $k = 1, 2$,(formula (27)),
4: Build the interpolation matrices R_{21} and R_{12} (formulas(10) and (11))and Q_{21} and Q_{12} (formula (28))(only the nodes coordinates on the interfaces are needed in this step),
5: Solve system (29)

5.2. Variational formulation of the INTERNODES method

System (29) represents the algebraic counterpart of the following variational problem: find $u_{1,\delta} \in X^{p_1}_{h_1}$: $(u_{1,\delta} - R_1 g_D) \in V_{1,\delta}$ and $u_{2,\delta} \in X^{p_2}_{h_2}$: $(u_{2,\delta} - R_2 g_D) \in V_{2,\delta}$ such that

$$
\begin{aligned}
&a_1(u_{1,\delta}, v_{1,\delta}) = (f, v_{1,\delta})_{\Omega_1} + \langle g_N, v_{1,\delta}\rangle_{\partial\Omega_{N,1}} \qquad \forall v_{1,\delta} \in V^0_{1,\delta},\\
&a_2(u_{2,\delta}, v_{2,\delta}) = (f, v_{2,\delta})_{\Omega_2} + \langle g_N, v_{2,\delta}\rangle_{\partial\Omega_{N,2}} \qquad \forall v_{2,\delta} \in V^0_{2,\delta},\\
&u_{2,\delta}|_{\Gamma_2} = \Pi_{21}(u_{1,\delta}|_{\Gamma_1})\\
&\quad a_1(u_{1,\delta}, w_\delta) + a_2(u_{2,\delta}, \tilde{w}_\delta) = (f, w_\delta)_{\Omega_1} + (f, \tilde{w}_\delta)_{\Omega_2}\\
&\quad + \langle g_N, w_\delta\rangle_{\partial\Omega_{N,1}} + \langle g_N, \tilde{w}_\delta\rangle_{\partial\Omega_{N,2}}\\
&\quad \forall w_\delta \in V_{\Gamma_1}, \text{ with } \tilde{w}_\delta = E_2(\Pi^*_{12} w_\delta|_{\Gamma_1}).
\end{aligned}
\tag{30}
$$

Here $\Pi^*_{12} : \Lambda_{1,\delta} \to \Lambda_{2,\delta}$ is the adjoint operator of Π_{12} w.r.t. the L^2 product, i.e., for any $\eta_{1,\delta} \in \Lambda_{1,\delta}$ and $\eta_{2,\delta} \in \Lambda_{2,\delta}$, it satisfies

$$
(\Pi^*_{12}\eta_{1,\delta}, \eta_{2,\delta})_{L^2(\Gamma_2)} = (\eta_{1,\delta}, \Pi_{12}\eta_{2,\delta})_{L^2(\Gamma_1)}. \tag{31}
$$

Remark 5.1. Notice that, even if $\eta_{1,\delta} \in \Lambda^0_{1,\delta}$, in general $(\Pi^*_{12}\eta_{1,\delta})|_{\partial\Gamma}$ is not null and by taking $v = E_2(\Pi^*_{12}\eta_{1,\delta})$ in $(18)_2$, it holds $v|_{\partial\Omega_{D,2}} \neq 0$ and the last integral in $(18)_2$ is non-zero.

Remark 5.2. In the conforming case, by setting $\Pi_{12} = \Pi_{21} = I$, Eq. (30) returns the well known two-domain formulation associated with the Galerkin finite element method, see [51].

We prove now the equivalence between (29) and (30). Eqs. $(30)_{1,2}$ correspond to the first two equations of the system (29); $(30)_3$ follows directly by the definition of the space V_{Γ_1} and yields $\mathbf{u}_{\Gamma_2} = Q_{21}\mathbf{u}_{\Gamma_1}$. Finally, $(30)_4$ corresponds to the last equation of system (29). To prove this statement, let us choose $\eta_{2,\delta} = \lambda^{(2)}_j$ (for any $j = 1, \dots, n_2$) and $\eta_{1,\delta} = \lambda^{(1)}_i$ (for any $i = 1, \dots, n_1$) in (31), thus by (9) it holds

$$
\left(\Pi_{12}\lambda^{(2)}_j\right)(\mathbf{x}) = \sum_{\ell=1}^{n_1} \lambda^{(2)}_j\left(\mathbf{x}^{(\Gamma_1)}_\ell\right)\lambda^{(1)}_\ell(\mathbf{x}) \quad \forall \mathbf{x} \in \Gamma_1,
$$

and

$$
\begin{aligned}
\left(\Pi_{12}\lambda^{(2)}_j, \lambda^{(1)}_i\right)_{L^2(\Gamma_1)} &= \int_{\Gamma_1} \sum_{\ell=1}^{n_1} \lambda^{(2)}_j\left(\mathbf{x}^{(\Gamma_1)}_\ell\right)\lambda^{(1)}_\ell(\mathbf{x})\lambda^{(1)}_i(\mathbf{x}) d\Gamma\\
&= \sum_{\ell=1}^{n_1} \lambda^{(2)}_j\left(\mathbf{x}^{(\Gamma_1)}_\ell\right) \int_{\Gamma_1} \lambda^{(1)}_\ell(\mathbf{x})\lambda^{(1)}_i(\mathbf{x}) d\Gamma\\
&= \sum_{\ell=1}^{n_1} (R_{12})_{\ell j}\left(M_{\Gamma_1}\right)_{i\ell} = \left(M_{\Gamma_1}R_{12}\right)_{ij}.
\end{aligned}
$$

At the same time, if we expand $\Pi^*_{12}\lambda^{(1)}_i$ w.r.t. the basis function in $\Lambda_{2,\delta}$ as

$$
(\Pi^*_{12}\lambda^{(1)}_i)(\mathbf{x}) = \sum_{k=1}^{n_2} (\Pi^*_{12}\lambda^{(1)}_i)(\mathbf{x}^{(2)}_k)\lambda^{(\Gamma_2)}_k(\mathbf{x}) \quad \forall \mathbf{x} \in \Gamma_2, \tag{32}
$$

and we denote by P the associated matrix such that $P_{ji} = (\Pi^*_{12}\lambda^{(1)}_i)(\mathbf{x}^{(\Gamma_2)}_j)$, we have

$$
\begin{aligned}
(\Pi^*_{12}\lambda^{(1)}_i, \lambda^{(2)}_j)_{L^2(\Gamma_2)} &= \int_{\Gamma_2} \sum_{k=1}^{n_2} (\Pi^*_{12}\lambda^{(1)}_i)(\mathbf{x}^{(\Gamma_2)}_k)\lambda^{(2)}_k(\mathbf{x})\lambda^{(2)}_j(\mathbf{x}) d\Gamma\\
&= \sum_{k=1}^{n_2} (\Pi^*_{12}\lambda^{(1)}_i)(\mathbf{x}^{(\Gamma_2)}_k) \int_{\Gamma_2} \lambda^{(2)}_k(\mathbf{x})\lambda^{(2)}_j(\mathbf{x}) d\Gamma\\
&= \sum_{k=1}^{n_2} P_{ki}(M_{\Gamma_2})_{jk} = (M_{\Gamma_2}P)_{ji}.
\end{aligned}
$$

Then, the algebraic counterpart of (31) reads

$$
(M_{\Gamma_1}R_{12})_{ij} = (M_{\Gamma_2}P)_{ji} = (P^T M_{\Gamma_2})_{ij},
$$

for any $i = 1, \dots, n_1$ and $j = 1, \dots, n_2$, or equivalently

$$
P^T = M_{\Gamma_1}R_{12}M^{-1}_{\Gamma_2} \quad (= Q_{12} \text{ by } (28)).
$$

This means that the matrix associated with Π^*_{12} is $P = Q^T_{12}$. Now, let us write

$$
\begin{aligned}
u_{1,\delta}(\mathbf{x}) &= \sum_{j=1}^{N^0_1} u_{1,\delta}(\mathbf{x}^{(1)}_j)\Phi^{(1)}_j(\mathbf{x}) + \sum_{j=1}^{n^0_1} u_{1,\delta}(\mathbf{x}^{(\Gamma_1)}_j)E_1\lambda^{(1)}_j(\mathbf{x})\\
&\quad + R_1 g_D(\mathbf{x}) \quad \forall \mathbf{x} \in \overline{\Omega}_1,\\
u_{2,\delta}(\mathbf{x}) &= \sum_{j=1}^{N^0_2} u_{2,\delta}(\mathbf{x}^{(2)}_j)\Phi^{(2)}_j(\mathbf{x}) + \sum_{j=1}^{n^0_1} u_{1,\delta}(\mathbf{x}^{(\Gamma_1)}_j)E_2(\Pi_{21}\lambda^{(1)}_j)(\mathbf{x})\\
&\quad + R_2 g_D(\mathbf{x}) \quad \forall \mathbf{x} \in \overline{\Omega}_2,
\end{aligned}
$$

and choose $w_\delta = \lambda^e_i$, (for $i = 1, \dots, n^0_1$) in $(30)_4$.

Recalling that $\lambda^e_i|_{\Gamma_1} = \lambda^{(1)}_i$ and by (19) and (26), $(30)_4$ reads for $i = 1, \dots, n^0_1$:

$$
\begin{aligned}
&\sum_{j=1}^{N^0_1} \mathbf{u}_{1j}\, a_1(\Phi^{(1)}_j, E_1\lambda^{(1)}_i) + \sum_{j=1}^{N^0_2} \mathbf{u}_{2j}\; a_2(\Phi^{(2)}_j, E_2(\Pi^*_{12}\lambda^{(1)}_i))\\
&\quad + \sum_{j=1}^{n^0_1} \mathbf{u}_{\Gamma_1 j}[a_1(E_1\lambda^{(1)}_j, E_1\lambda^{(1)}_i) + a_2(E_2(\Pi_{21}\lambda^{(1)}_j), E_2(\Pi^*_{12}\lambda^{(1)}_i))\\
&\quad = \mathcal{F}_1(E_1\lambda^{(1)}_i) + \mathcal{F}_2(E_2(\Pi^*_{12}\lambda^{(1)}_i)),
\end{aligned}
$$

and thanks to both (21) and (32), it holds

$$
\begin{aligned}
a_2(\Phi^{(2)}_j, E_2(\Pi^*_{12}\lambda^{(1)}_i)) &= \sum_{k=1}^{n_2} (\Pi^*_{12}\lambda^{(1)}_i)(\mathbf{x}^{(\Gamma_2)}_k)\; a_2(\Phi^{(2)}_j, E_2\lambda^{(2)}_k)\\
&= \sum_{k=1}^{n_2} (Q^T_{12})_{ki}(A_{\Gamma_2,2})_{kj} = (Q_{12}A_{\Gamma_2,2})_{ij},\\
&\quad i = 1, \dots, n^0_1\\
&\quad j = 1, \dots, N^0_2,
\end{aligned}
$$

$$
\begin{aligned}
&a_2(E_2(\Pi_{21}\lambda^{(1)}_j), E_2(\Pi^*_{12}\lambda^{(1)}_i))\\
&\quad = \sum_{k=1}^{n_2} (\Pi^*_{12}\lambda^{(1)}_i)(\mathbf{x}^{(\Gamma_2)}_k) \sum_{\ell=1}^{n_2} (\Pi_{21}\lambda^{(1)}_j)(\mathbf{x}^{(\Gamma_2)}_\ell)\; a_2(E_2\lambda^{(2)}_\ell, E_2\lambda^{(2)}_k)\\
&\quad = \sum_{k=1}^{n_2}\sum_{\ell=1}^{n_2} (Q^T_{12})_{ki}(A_{\Gamma_2,\Gamma_2})_{k\ell}(Q_{21})_{\ell j} = (Q_{12}A_{\Gamma_2,\Gamma_2}Q_{21})_{ij},\\
&\quad i = 1, \dots, n^0_1\\
&\quad j = 1, \dots, n^0_1,
\end{aligned}
$$

and

$$\mathcal{F}_2(E_2(\Pi_{12}^*\lambda_i^{(1)})) = (Q_{12}\mathbf{f}_{\Gamma_2})_i, \quad i = 1, \dots, n_1^0,$$

thus $(29)_3$ is the algebraic counterpart of $(30)_4$.

Eq. $(30)_4$ (or equivalently $(29)_3$) expresses the balance of residuals in strong form. Algebraically, this becomes more evident once we reformulate $(29)_3$ as

$$-(Q_{12}\mathbf{r}_2)_i = -(M_{\Gamma_1}R_{12}M_{\Gamma_2}^{-1}\mathbf{r}_2)_i = (\mathbf{r}_1^0)_i, \qquad i = 1, \dots, n_1^0 \tag{33}$$

where

$$\mathbf{r}_1^0 = \mathbf{f}_{\Gamma_1}^0 - A_{\Gamma_1,1}^0\mathbf{u}_1^0 - A_{\Gamma_1,\Gamma_1}^0\mathbf{u}_{\Gamma_1}^0, \quad \mathbf{r}_2 = \mathbf{f}_{\Gamma_2} - A_{\Gamma_2,2}\mathbf{u}_2^0 - A_{\Gamma_2,\Gamma_2}Q_{21}^0\mathbf{u}_{\Gamma_1}^0. \tag{34}$$

In Eq. (33), we notice that $M_{\Gamma_2}^{-1}\mathbf{r}_2$ is an approximation of the strong form of the normal stresses on Γ_2; $R_{12}M_{\Gamma_2}^{-1}\mathbf{r}_2$ is an interpolation of the normal stresses on Γ_1, still in strong form, and $M_{\Gamma_1}R_{12}M_{\Gamma_2}^{-1}\mathbf{r}_2$ returns the weak form of the normal stresses but now on Γ_1. Note that the order of magnitude of the entries of $\mathbf{r}_2$ depend on the mesh size used to discretize Ω_2, that of the entries of $\mathbf{r}_1$ depend on the mesh size of Ω_1, while the order of magnitude of those of both $M_{\Gamma_2}^{-1}\mathbf{r}_2$ and $R_{12}M_{\Gamma_2}^{-1}\mathbf{r}_2$ are independent of the mesh size.

Eq. $(30)_4$ is the weak realization of the property

$$\mu\frac{\partial u_{1,\delta}}{\partial n_1} = -\Pi_{12}\left(\mu\frac{\partial u_{2,\delta}}{\partial n_2}\right) \qquad \text{on } \Gamma_1, \tag{35}$$

that enforces the discrete continuity of the normal fluxes across Γ_1.

5.3. Non-conforming Petrov–Galerkin formulation of the INTERNODES method

By defining the spaces

$$V_{\Gamma_1}^* = \{\varphi^* \in L^2(\Omega) : \exists\lambda_1 \in \Lambda_{1,\delta}^0 : \quad \varphi^*|_{\Omega_1} = E_1\lambda_1, \ \varphi^*|_{\Omega_2} = E_2(\Pi_{12}^*\lambda_1)\}. \tag{36}$$

and

$$V_\delta^* = \widetilde{V}_{1,\delta} \oplus \widetilde{V}_{2,\delta} \oplus V_{\Gamma_1}^*, \tag{37}$$

the variational statement (30) can be written in compact form as a *non-conforming Petrov-Galerkin* problem: find $u_\delta \in X_{h_1}^{p_1} \times X_{h_2}^{p_2}$ with $(u_\delta - R_\delta g_D) \in V_\delta$:

$$a_1(u_\delta, v_\delta^*) + a_2(u_\delta, v_\delta^*) = \sum_{k=1}^{2}\left[(f, v_\delta^*)_{\Omega_k} + \langle g_N, v_\delta^*\rangle_{\partial\Omega_{N,k}}\right], \quad \forall v_\delta^* \in V_\delta^*. \tag{38}$$

5.4. Non-conforming generalized Galerkin formulation of the INTERNODES method

The finite element assembly of problem (38) would be rather involved as it requires to generate a set of basis functions for V_δ^*. For this reason we reformulate (38) as a more convenient *non-conforming generalized Galerkin* problem. With this aim, for any $w_\delta \in V_\delta$ we define

$$a_{2,\delta}(w_\delta, v_\delta) = \begin{cases} a_2(w_\delta, v_\delta), & \text{if } v_\delta \in \widetilde{V}_{1,\delta} \oplus \widetilde{V}_{2,\delta} \\ a_2(w_\delta, E_2(\Pi_{12}^* v_\delta|_{\Gamma_1})), & \text{if } v_\delta \in V_{\Gamma_1} \end{cases}$$

$$(f, v_\delta)_{2,\delta} = \begin{cases} (f, v_\delta)_{\Omega_2}, & \text{if } v_\delta \in \widetilde{V}_{1,\delta} \oplus \widetilde{V}_{2,\delta} \\ (f, E_2(\Pi_{12}^* v_\delta|_{\Gamma_1}))_{\Omega_2}, & \text{if } v_\delta \in V_{\Gamma_1} \end{cases} \tag{39}$$

$$\langle g_N, v_\delta\rangle_{2,\delta} = \begin{cases} \langle g_N, v_\delta\rangle_{\Omega_2}, & \text{if } v_\delta \in \widetilde{V}_{1,\delta} \oplus \widetilde{V}_{2,\delta} \\ \langle g_N, E_2(\Pi_{12}^* v_\delta|_{\Gamma_1})\rangle_{\Omega_2}, & \text{if } v_\delta \in V_{\Gamma_1}. \end{cases}$$

We can therefore conclude that problem (30) can be equivalently reformulated as a non-conforming generalized Galerkin problem:

Find $u_\delta \in X_{h_1}^{p_1} \times X_{h_2}^{p_2}$, with $(u_\delta - R_\delta g_D) \in V_\delta$:

$$a_1(u_\delta, v_\delta) + a_{2,\delta}(u_\delta, v_\delta) = (f, v_\delta)_{\Omega_1} + (f, v_\delta)_{2,\delta} + \langle g_N, v_\delta\rangle_{\partial\Omega_{N,1}} + \langle g_N, v_\delta\rangle_{2,\delta}, \quad \forall v_\delta \in V_\delta. \tag{40}$$

6. On the mortar method and its relationship with INTERNODES

As already pointed out in the Introduction, the mortar method represents nowadays a well established approach for the solution of PDEs using non-conforming discretizations. A big deal of attention has been devoted to the theoretical analysis [4,5,11,12,14,36,38,57] as well as to the algorithmic developments [7–9,42,47,48] of this method, in connection with a broad variety of applications. Notable examples concern problems in structural mechanics [28,46,47], fluid dynamics [25], structural dynamics [26], electromagnetism [6,13,15,52], contact problems [20,27,43–45], multiphysics [1,34,40,42], etc.

Although the mortar method is a projection (rather than an interpolation-based) method, we can still represent it by the general algebraic form (29), provided we replace Q_{21}^0 with the matrix associated with the mortar projection (named Ξ in [49, Section 11.4] or P in [34, Eq. (46)]) and Q_{12}^0 with Ξ^T. Similarly, the variational formulation of the mortar method (see [11]) can be retrieved from (40) by replacing V_δ with the mortar space

$$V_\delta^M = \{v_\delta \in L^2(\Omega), v_{k,\delta} = v_\delta|_{\Omega_k} \in V_{k,\delta} \text{ for } k = 1, 2 \text{ and } v_{2,\delta} = \Pi_{21}^M v_{1,\delta} \text{ on } \Gamma\}, \tag{41}$$

where Π_{21}^M is the L^2-projection from master to slave on the interface, and by replacing Π_{12}^* with Π_{21}^M in (39). Note in particular that the mortar method requires a single intergrid operator, Π_{21}^M, rather than two operators Π_{12} and Π_{21} in INTERNODES.

On the other hand, we warn the reader that INTERNODES, even though being interpolatory, *does not* coincide with the so-called *pointwise matching* method that was presented in the seminal mortar paper [11, Eqs. (3.5)–(3.7)]. The pointwise matching is notoriously sub-optimal, as proven in [11, Section 3.2] and numerically corroborated in [3] for spectral elements discretizations.

The algorithmic and implementation aspects of both methods deserve some further consideration. Most often mortar method is formulated as a saddle point problem by introducing an extra field, the Lagrange multiplier. This yields an inf-sup compatibility condition to be fulfilled in order to ensure well-posedness. Many algorithms exist aimed at condensing the system by eliminating the Lagrange multipliers, in particular the one using dual spaces for Lagrange multipliers [25,57].

When mortar methods are formulated as a single field problem, the corresponding algebraic system (29) (with Π_{12} and Π_{21} replaced by Π_{21}^M as indicated above) is symmetric, provided the original differential operator is self-adjoint. This property is not fulfilled by INTERNODES due to the two a-priori different intergrid operators.

The INTERNODES and mortar stiffness matrices of (29) feature similar condition numbers. For example, on a piecewise linear finite element approximation of the Dirichlet problem for the Laplacian in the rectangle $(0, 2) \times (0, 1)$, we report in Fig. 3 the spectra for both INTERNODES and the mortar method relatively to structured triangulations of variable step size (precisely, we are using

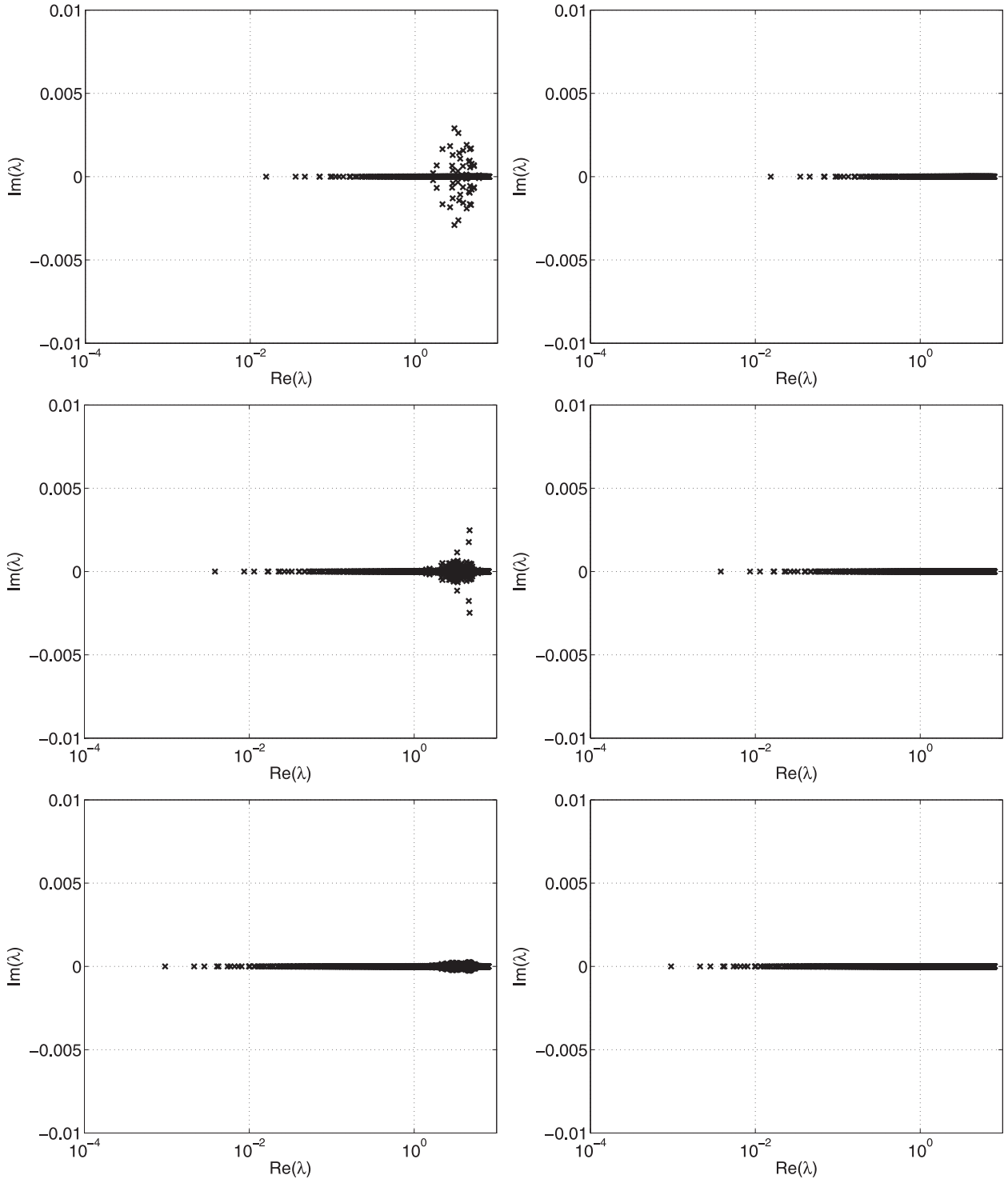


Fig. 3. The spectra of the Dirichlet stiffness matrix of system (29) for INTERNODES (left) and mortar (right), corresponding to three different structured triangulations (Set D of Table 2, $k = 16$ (top), 32 (middle), 64 (bottom)).

Set D of Table 2 for $k = \{16,\ 32,\ 64\}$). The eigenvalues of the mortar matrix are positive real whereas those of INTERNODES feature tiny imaginary parts that vanish as the step size does. The *iterative condition number* $\mathcal{K} = \max_i |\lambda_i| / \min_i |\lambda_i|$ behaves right the same way (and scales like h_1^{-2}, or equivalentely like h_2^{-2}, since $h_2 \sim h_1/2$), as reported in Table 1.

Concerning more specifically the implementation issues, INTERNODES simply requires local mass matrices at the interface and not the cross mass matrix connecting interface basis functions from both sides as in the mortar method. Moreover, it does not require numerical quadratures, neither a special treatment of cross-points where more than two subdomains meet. An in-depth analysis of these and other subtle implementation issues of mortar methods is carried out in [25,28,39,42,48,57]. The implementation of mortar method for geometric non-conforming interfaces is also far from trivial (INTERNODES instead does not feature any additional difficulty with respect to the case of geometric matching interfaces): as a matter of fact, it requires several steps such as

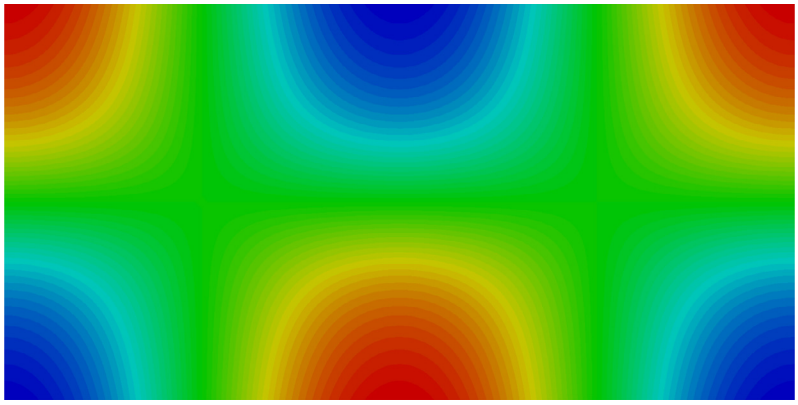


Fig. 4. Solution $u(x, y) = \arctan(4(y - 0.5))\cos(\pi x)$ of problem (42).

Table 1
Iterative condition number and extreme eigenvalues.

	INTERNODES			mortar		
k	$\mathcal{K}$	$\max_i\|\lambda_i\|$	$\min_i\|\lambda_i\|$	$\mathcal{K}$	$\max_i\|\lambda_i\|$	$\min_i\|\lambda_i\|$
16	511.65	7.98	1.56e−2	518.63	7.98	1.54e−2
32	2081.39	7.99	3.84e−3	2093.80	7.99	3.82e−3
64	8386.54	7.99	9.54e−4	8410.09	7.99	9.51e−4

Table 2
Number of grid points for the meshes used in the numerical simulations.

	Master domain	Slave domain	Aspect ratio
Set A	$(k+1)\times(k+1)$	$(k-1)\times(k-1)$	≈ 1
Set B	$(k-1)\times(k-1)$	$(k+1)\times(k+1)$	≈ 1
Set C	$(2k+1)\times(2k+1)$	$(k-1)\times(k-1)$	≈ 2
Set D	$(k-1)\times(k-1)$	$(2k+1)\times(2k+1)$	≈ 2

projection, intersection, local meshing and numerical quadrature to build up the mortar interface coupling operator. These aspects are carefully addressed in [42] (see in particular Algorithm 1, Section 3.2.3).

7. Numerical solution of an elliptic problem

In the first preliminary test, we consider the numerical solution of the Poisson problem

$$\begin{aligned} -\Delta u(x,y) &= f(x,y) \quad \text{in} \quad \Omega = (0,2)\times(0,1), \\ u(x,y) &= g_D(x,y) \quad \text{on} \quad \partial\Omega_D = \partial\Omega, \end{aligned} \tag{42}$$

and we show the orders of convergence of INTERNODES when non-conforming meshes and/or non-conforming discretizations (based on the coupling of finite element s with spectral element s) are used.

In (42) the functions $f(x, y)$ and $g(x, y)$ are chosen in such a way that $u(x, y) = \arctan(4(y - 0.5))\cos(\pi x)$ (see Fig. 4). We decompose the domain Ω in two subdomains: $\Omega_1 = (0, 1)\times(0, 1)$ and $\Omega_2 = (1, 2)\times(0, 1)$.

7.1. Coupling of non-conforming FEM–FEM discretizations

In this Section we solve problem (42) by considering non-conforming finite elements discretizations at the subdomains interface Γ. The non-conformity may come from the use of different mesh-sizes and/or different polynomial degree of the finite elements basis functions between the master and slave domains. In our numerical experiments we considered $\mathbb{P}_1$, $\mathbb{P}_2$ and $\mathbb{P}_3$ finite elements, using structured grids that feature an aspect ratio of 1 or 2 across the interface. The details of the meshes used in our simulations are reported in Table 2, wherein $k = \{8, 16, 32, 64\}$ is an index used to set up the number of grid points along each coordinate of the master and slave domains. Furthermore, the method proposed is tested using both the Lagrange and the RL-RBF interpolants as intergrid operator.

Table 3
Orders of convergence in H^1 norm obtained using non-conforming meshes and the Lagrange interpolation. In the top row we report the results obtained using Set A (left) and Set B (right) while in the bottom those with Set C (left) and Set D (right).

Master\slave	$\mathbb{P}_1$		$\mathbb{P}_2$		$\mathbb{P}_3$	
$\mathbb{P}_1$	1–1	1–1	1–2	1–2	1–1	1–1
	1–1	1–1	1–2	1–2	1–1	1–1
$\mathbb{P}_2$	2–1	2–1	2–2	2–2	2–3	2–3
	2–1	2–1	2–2	2–2	2–3	2–3
$\mathbb{P}_3$	2–1	2–1	3–2	3–2	3–3	3–3
	2–1	2–1	3–2	3–2	3–3	3–3

In Fig. 5 we show the rate of convergence obtained by INTERNODES for some of the simulations performed using the Lagrange interpolant and for set C (left column) and set D (right column). The results reported are the H^1-norms of the errors computed in each individual subdomain, i.e., $\|u - u_{1,\delta}\|_{H^1(\Omega_1)}$ and $\|u - u_{2,\delta}\|_{H^1(\Omega_2)}$.

By comparing the left and right plots in the first and fourth rows of Fig. 5 (obtained with non-conforming meshes but same polynomial degree in the master and slave domains) we observe that, as expected, the most accurate results are always obtained on the subdomain triangulated with the finer mesh, independently whether this is a master or a slave. When the master domain is discretized using a polynomial degree lower than the one of the slave (compare the left and right plots of the second row in Fig. 5) we notice that it is better, in terms of accuracy, to use the finer mesh on the master domain. In the opposite case, i.e. when the master domain is discretized using a polynomial degree that is higher than the one of the slave (compare the left and right plots of the third row in Fig. 5) we observe that more precise results are obtained using the finer mesh on the slave domain.

In Table 3 we summarize the orders of convergence obtained by INTERNODES using non-conforming meshes and the Lagrange interpolant, while in Table 4 those using matching grids but non-conforming polynomial degrees. Let us denote by p_1 and p_2 the polynomial degrees used in Ω_1 and Ω_2, respectively. As shown in Table 3, if $|p_1 - p_2| \leq 1$, the use of the Lagrange interpolant yields rates of convergence that are optimal in each individual subdomain, in fact they behave as $h_i^{p_i}$, independently of the choice of

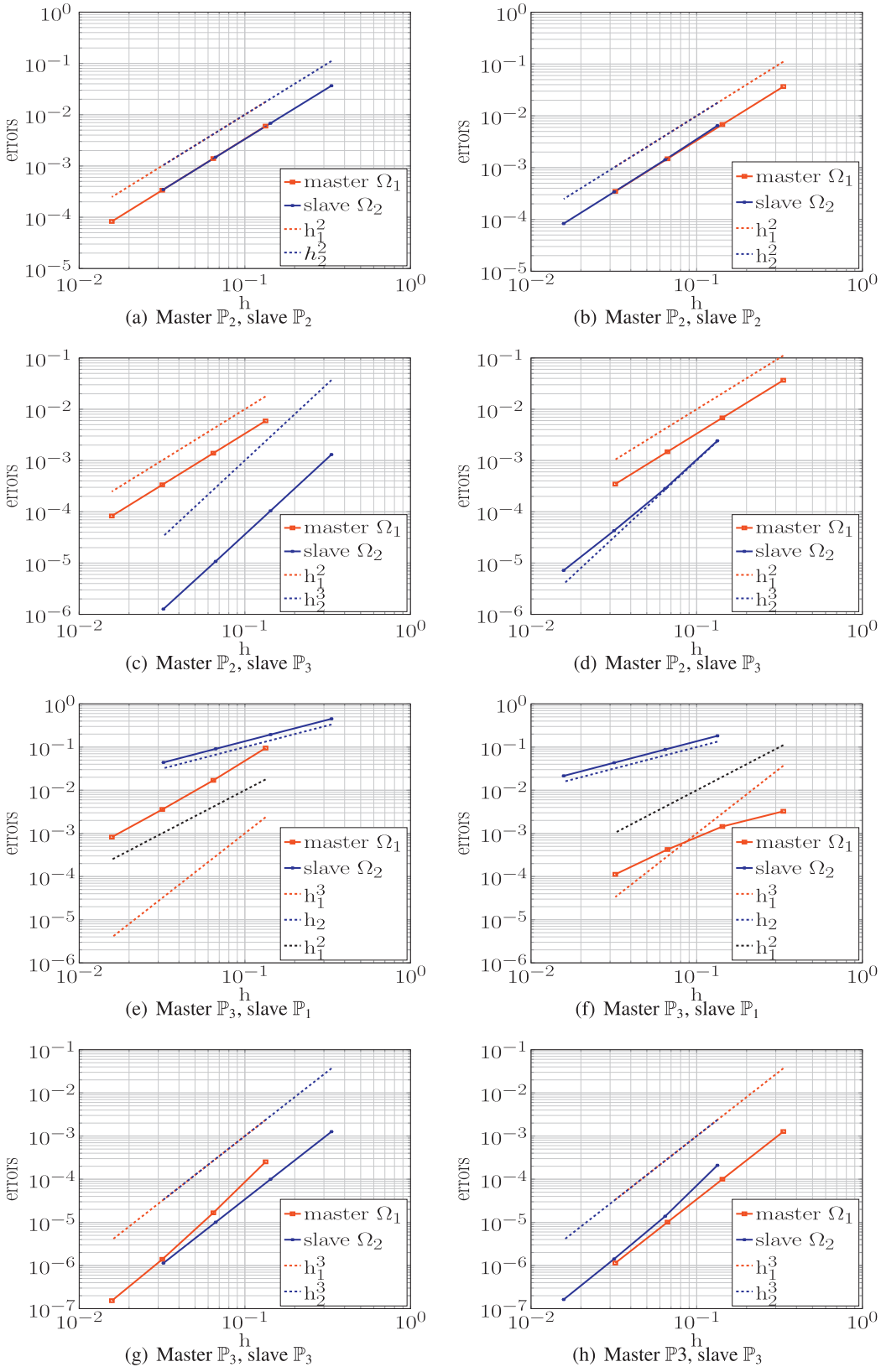


Fig. 5. FEM–FEM coupling: orders of convergence obtained using Lagrangian interpolants and non-conforming meshes with aspect ratio 2. Left column using Set C, right column using Set D (see Table 2).

Table 4
Orders of convergence in H^1 norm obtained using conforming meshes and both conforming and non-conforming polynomial degrees. In each cell of the table, on the left column we report the results computed using the Lagrange interpolation, on the right those by the RL-RBF interpolation. M stands for master while S for slave domain.

M\S	$\mathbb{P}_1$		$\mathbb{P}_2$		$\mathbb{P}_3$	
	Lagr.	RL-RBF	Lagr.	RL-RBF	Lagr.	RL-RBF
$\mathbb{P}_1$	1–1	1–1	1–2	1–2	1–2	1–2
$\mathbb{P}_2$	2–1	2–1	2–2	2–2	2–3	2–3
$\mathbb{P}_3$	2–1	2–1	3–2	3–2	3–3	3–3

Table 5
Orders of convergence in H^1 norm obtained using non-conforming meshes and the RL-RBF interpolation. The cells are organized as in Table 3.

Master\slave	$\mathbb{P}_1$		$\mathbb{P}_2$		$\mathbb{P}_3$	
$\mathbb{P}_1$	1–1	1–1	1–2	1–2	1–2	1–2
	1–1	1–1	1–2	1–2	1–2	1–2
$\mathbb{P}_2$	2–1	2–1	2–2	2–2	2–3	2–3
	2–1	2–1	2–2	2–2	2–3	2–3
$\mathbb{P}_3$	2–1	2–1	3–2	3–2	3–3	3–3
	2–1	2–1	3–2	3–2	3–3	2–2

which domain plays the role of the master or slave. In the case where $p_1 - p_2 > 1$, for instance using $\mathbb{P}_3$-$\mathbb{P}_1$ finite elements, the rate of convergence behaves like $h_1^{p_1-1}$ in Ω_1 and like $h_2^{p_2}$ in Ω_2 (or, if $p_2 - p_1 > 1$, as $h_1^{p_1}$ in Ω_1 and as $h_2^{p_1+1}$ in Ω_2).

These results can be summarized by the help of the following empirical formula, holding for Lagrange interpolants: if both $h_1, h_2 \to 0$,

$$\|u - u_{i,\delta}\|_{H^1(\Omega_i)} \le C_i(p_1, p_2) h_i^{\min(q_i, s_i - 1)} \|u\|_{H^{s_i}(\Omega_i)} \quad \text{for } i = 1, 2, \tag{43}$$

where

$$q_1 = \min(p_1, p_2 + 1),$$
$$q_2 = \begin{cases} p_2 & \text{if } p_2 \le p_1 + 1 \\ p_1 & \text{if } p_2 > p_1 + 1. \end{cases}$$

In (43), C_i are positive constants independent of h_i, while $s_i > 1$ is the order of the Sobolev regularity of the exact solution in Ω_i.

The results obtained on non-conforming meshes using the RL-RBF interpolant are reported both in Tables 4 and 5. We notice that, if $|p_1 - p_2| \le 1$, the method leads to optimal rates of convergence in all the numerical experiments performed, apart from the case of the $\mathbb{P}_3$–$\mathbb{P}_3$ discretization with grids of Set D. This may be due to the fact that, as shown in Eq. (12), the construction of the RL-RBF interpolant does not take into account the polynomial degree of the trace of the finite element basis functions at the interface since it *only requires the nodal coordinates and corresponding nodal values at the two interfaces, separately*.

However, this same feature makes RL-RBF a flexible interpolation tool to deal with problems with geometrically non-conforming interfaces (as in the case of Fig. 1). The treatment of non-matching interfaces in the mortar setting is instead more involved as it requires, in particular, an ad-hoc projection to retrieve a yet another (virtual) common interface, cf. [24,34].

In Fig. 6 we report the PDE's approximation errors in broken norm (see [49, Section 11.3]) and the pure interpolation errors for both Lagrange and RL-RBF interpolants versus the polynomial degree p when non-conforming meshes are used (Set D, $k = 64$). We notice that the approximation error and the interpolation error feature almost the same rate of decay with respect to p. In particular, for $p = 1$ the RL-RBF interpolation leads to more accurate results w.r.t. the Lagrange one; for $p = 2$ both interpolants yield the same accuracy while for $p = 3$ the Lagrange interpolation is more accurate. This justifies the sub-optimal orders of the block $\mathbb{P}_1$–$\mathbb{P}_3$ of Table 3 w.r.t. the corresponding block of Table 5, as well as those of the $\mathbb{P}_3$–$\mathbb{P}_3$ simulation with Set D in Table 5.

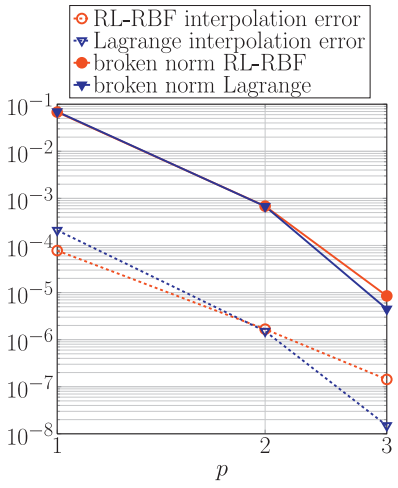


Fig. 6. Comparison between PDE approximation errors and interpolation errors for $\mathbb{P}_p$–$\mathbb{P}_p$ FEM, and non-conforming meshes (set D of Table 2 and $k = 64$).

7.2. Coupling of non-conforming SEM–SEM discretizations

In this section we consider SEM discretization in both master and slave domains and we compare the errors obtained by INTERNODES (using the Lagrange intergrid operator) with those generated by the mortar approach. As in the previous subsection, we plot the errors in H^1-norm, i.e., $\|u - u_{1,\delta}\|_{H^1(\Omega_1)}$ and $\|u - u_{2,\delta}\|_{H^1(\Omega_2)}$.

Pictures in the left column of Fig. 7 refer to INTERNODES, while those in the right column to the mortar approach.

Let p_1 and p_2 denote the polynomial degrees used in Ω_1 and Ω_2, respectively. The errors displayed by the two methods are comparable; moreover, for $i = 1, 2$, they decay as $h_i^{p_i}$ if $|p_1 - p_2| \le 1$, while the order of convergence is downgraded when $|p_1 - p_2| > 1$, as we can see in the last row of Fig. 7, where $p_1 = 5$ and $p_2 = 2$. In fact, in the latter case, the error in the master domain behaves like h_1^3 for both the methods and not as h_1^5.

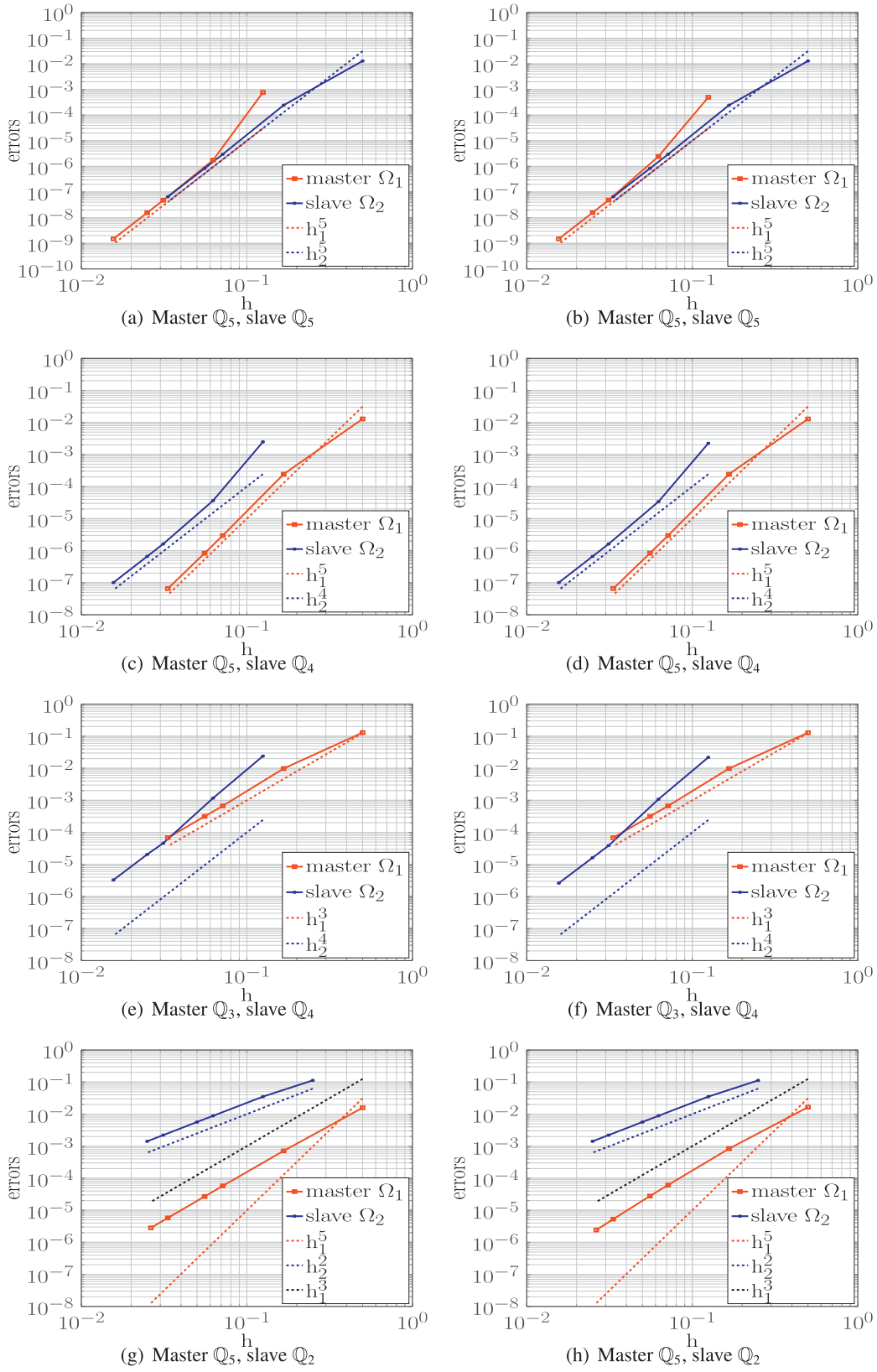
More precisely, the plot in the first row refers to a test case with polynomial conformity and mesh non-conformity; the one in the second row to a case with both polynomial and mesh non-conformity, with $p_1 = p_2 + 1$; that in the third row again to a case with both polynomial and mesh non-conformity, with $p_1 = p_2 - 1$; finally the last row to a situation with $p_1 - p_2 > 1$.

In Tables 6 and 7 we show the rates of convergence with respect to h both for INTERNODES and the mortar approach, respectively, when considering different non-conforming situations, as those in Tables 3 and 5.

First of all we notice that INTERNODES is accurate as well as the mortar method. Furthermore, the trend observed for the FEM discretization (and reported in formula (43)) holds also for SEM case.

In Table 8 the orders of convergence in H^1 norm versus the mesh sizes h_1 and h_2 are shown in the case of conforming meshes and different (but also equal) polynomial degrees.

The convergence analysis w.r.t. the polynomial degrees p_i is more involved and it is currently under investigation. See however Fig. 8 below where the error behavior of both INTERNODES and



(a) Master $\mathbb{Q}_5$, slave $\mathbb{Q}_5$ (b) Master $\mathbb{Q}_5$, slave $\mathbb{Q}_5$

(c) Master $\mathbb{Q}_5$, slave $\mathbb{Q}_4$ (d) Master $\mathbb{Q}_5$, slave $\mathbb{Q}_4$

(e) Master $\mathbb{Q}_3$, slave $\mathbb{Q}_4$ (f) Master $\mathbb{Q}_3$, slave $\mathbb{Q}_4$

(g) Master $\mathbb{Q}_5$, slave $\mathbb{Q}_2$ (h) Master $\mathbb{Q}_5$, slave $\mathbb{Q}_2$

Fig. 7. Convergence history with respect to the mesh sizes h_1 and h_2 for INTERNODES (left) and mortar (right) approaches. SEM discretization. First line: Set C; second line: Set D; third line: Set D; fourth line: Set A.

Table 6
SEM–SEM coupling: orders of convergence with respect to the mesh sizes h_1 and h_2 of INTERNODES using non-conforming meshes. M stands for master domain while S for slave domain. The cells are organized as in Table 3.

M\S	$\mathbb{Q}_2$		$\mathbb{Q}_3$		$\mathbb{Q}_4$		$\mathbb{Q}_5$	
$\mathbb{Q}_2$	2–2	2–2	2–3	2–3	2–2	2–3	2–2	2–3
	2–2	2–2	2–3	2–3	2–2	2–3	2–2	2–3
$\mathbb{Q}_3$	3–2	3–2	3–3	3–3	3–4	3–4	3–3	3–4
	3–2	3–2	3–3	3–3	3–4	3–4	3–4	3–4
$\mathbb{Q}_4$	4–2	3–2	4–3	4–3	4–4	4–4	4–5	4–5
	3–2	3–2	4–3	4–3	4–4	4–4	4–5	4–5
$\mathbb{Q}_5$	4–2	3–2	5–3	4–3	5–4	5–4	5–5	5–5
	3–2	3–2	5–3	5–3	5–4	5–4	5–5	5–5

Table 7
SEM–SEM coupling: orders of convergence with respect to the mesh sizes h_1 and h_2 of the mortar approach using non-conforming meshes. M stands for master domain while S for slave domain. The cells are organized as in Table 3.

M\S	$\mathbb{Q}_2$		$\mathbb{Q}_3$		$\mathbb{Q}_4$		$\mathbb{Q}_5$	
$\mathbb{Q}_2$	2–2	2–2	2–3	2–3	2–2	2–3	2–2	2-3
	2–2	2–2	2–3	2–3	2–2	2–3	2–2	2–3
$\mathbb{Q}_3$	3–2	3–2	3–3	3–3	3–4	3–4	3–3	3-4
	3–2	3–2	3–3	3–3	3–4	3–4	3–4	3–4
$\mathbb{Q}_4$	4–2	3–2	4–3	4–3	4–4	4–4	4–5	4–5
	3–2	3–2	4–3	4–3	4–4	4–4	4–5	4–5
$\mathbb{Q}_5$	4–2	3–2	5–3	4–3	5–4	5–4	5–5	5–5
	3–2	3–2	5–3	5–3	5–4	5–4	5–5	5–5

Table 8
SEM–SEM coupling: orders of convergence with respect to the mesh sizes h_1 and h_2, using conforming meshes. In each cell of the table, on the left and on the right we report the results computed using INTERNODES and mortar approach, respectively.

M\S	$\mathbb{Q}_2$		$\mathbb{Q}_3$		$\mathbb{Q}_4$		$\mathbb{Q}_5$	
$\mathbb{Q}_2$	2–2	2–2	2–3	2–3	2–2	2–2	2–2	2–2
$\mathbb{Q}_3$	3–2	3–2	3–3	3–3	3–4	3–4	3–4	3–4
$\mathbb{Q}_4$	4–2	4–2	4–3	4–3	4–4	4–4	4–5	4–5
$\mathbb{Q}_5$	4–2	4–2	5–3	5–3	5–4	5–4	5–5	5–5

the mortar methods versus the polynomial degree p are shown. The two curves are practically overlaid when using Lagrangian interpolation for INTERNODES. In the same figure we also plot the pure interpolation errors: as it can be appreciated, the approximation error and the interpolation error feature the same rate of decay with respect to p. Also reported are the RL-RBF interpolation errors: for moderate polynomial degrees ($p \leq 5$) the rate of decay is the same as for Lagrangian interpolation, whereas (as expected) it flattens for larger values of p. Note that the better accuracy displayed for the RL-RBF interpolation errors with respect to that of Table 5 is due to the fact that interpolation nodes now coincide with the (non-uniformly spaced) Gauss–Lobatto–Legendre nodes [17].

7.3. Coupling of FEM–SEM discretizations

We consider now the coupling of FEM–SEM discretizations. We set $\Omega_1 = (0,1)\times(0,1)$, $\Omega_2 = (1,2)\times(0,1)$ and the function $u(x,y) = \arctan(4(y-0.5))\cos(\pi(x-0.1))$ as exact solution of the problem (42).

Polynomial non-conformity: In Table 9 we show the convergence orders w.r.t. the mesh sizes h when the master domain Ω_1 is discretized by $\mathbb{P}_1$ finite elements and the slave domain Ω_2 by $\mathbb{Q}_p$ spectral elements with $p = 2, 3, 4$ and vice versa. h denotes the diameter of the structured and regular triangular mesh, that coincides with the diameter of the spectral elements. In this first test case we consider mesh conformity and interpolation by either Lagrange and RL-RBF. Finally we compare the results obtained by INTERNODES with those generated by the mortar method.

The advantage of using RL-RBF instead of Lagrange interpolation is clear when the master discrete space is the poorest one, i.e. when it is discretized by $\mathbb{P}_1$. More precisely, when using Lagrange interpolation, the first order of convergence driven by $\mathbb{P}_1$ in

Table 9
FEM–SEM coupling: orders of convergence with respect to the mesh sizes h_1 and h_2 when using conforming meshes.

Master–slave	Lagrange	RL-RBF	mortar
$\mathbb{P}_1$–$\mathbb{Q}_2$	1–1	1–2	1–1.6
$\mathbb{P}_1$–$\mathbb{Q}_3$	1–1	1–2	1–1.7
$\mathbb{P}_1$–$\mathbb{Q}_4$	1–1	1–2	1–1.7
$\mathbb{Q}_2$–$\mathbb{P}_1$	2–1	2–1	2–1
$\mathbb{Q}_3$–$\mathbb{P}_1$	2–1	2–1	2–1
$\mathbb{Q}_4$–$\mathbb{P}_1$	2–1	2–1	2-1

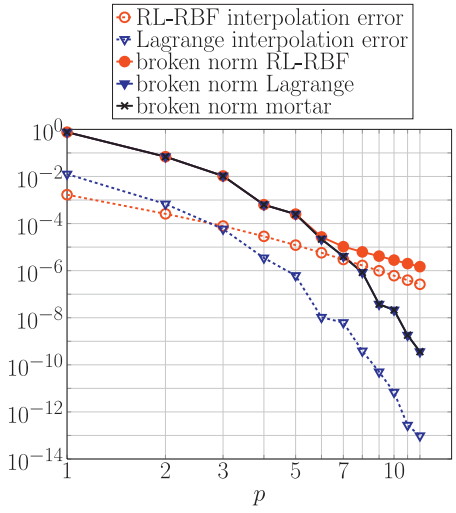


Fig. 8. Comparison between PDE approximation errors and interpolation errors for SEM–SEM couplings 8×8 and 6×6 spectral elements are used in the master and in the slave subdomains, respectively.

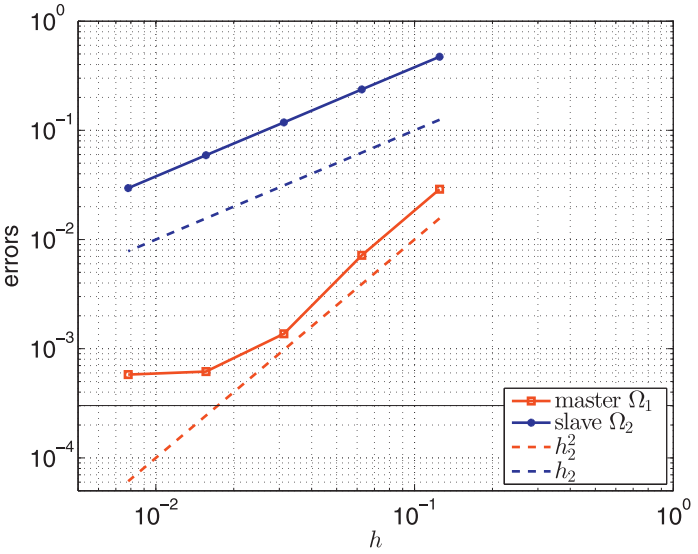


Fig. 9. SEM–FEM coupling. H^1 norm errors versus h_2 when $\mathbb{Q}_4$ are used in Ω_1 (master) with fix $h_1 = 1/6$, and $\mathbb{P}_1$ are used in Ω_2 (slave). The black line denotes the error obtained by discretizing the global domain by conforming $\mathbb{Q}_4$ SEM.

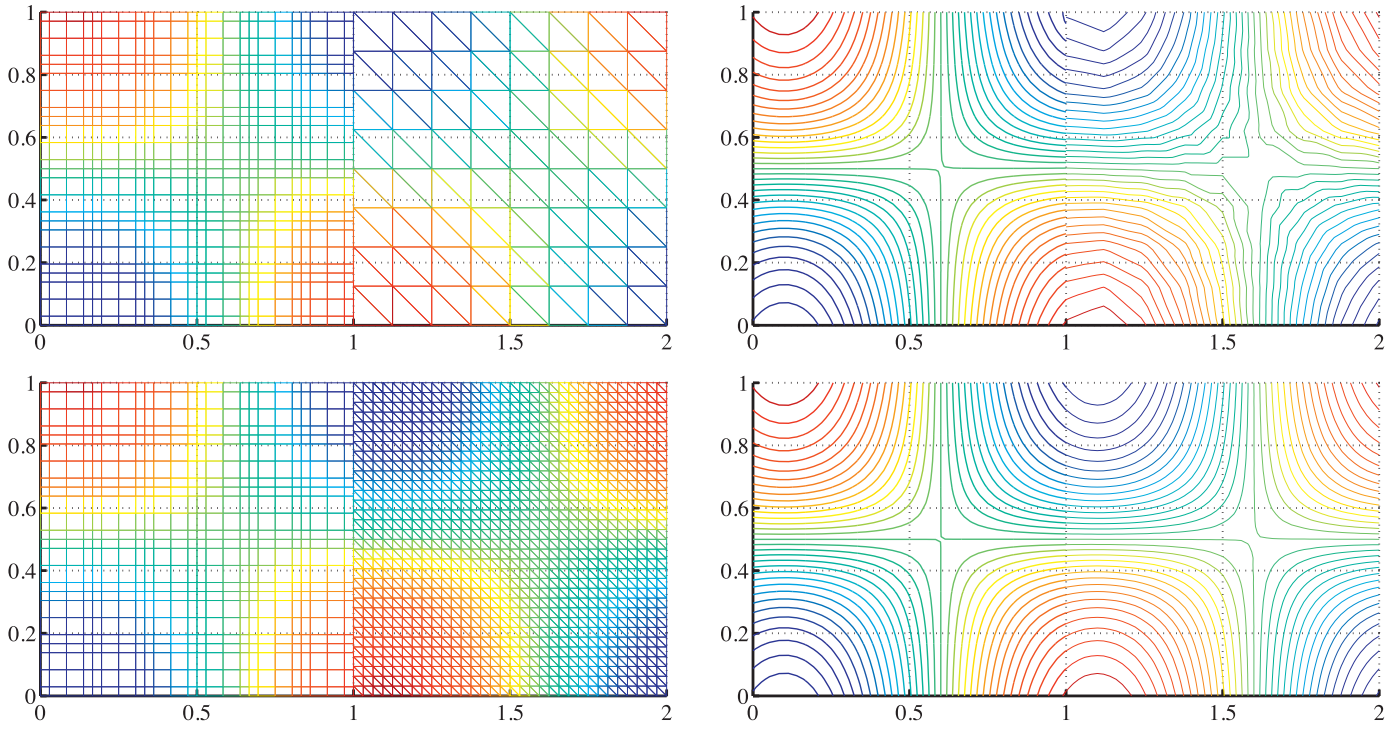


Fig. 10. SEM–FEM coupling. Meshes (at left) and solution contours (at right) obtained using $\mathbb{Q}_4$ in Ω_1 (master) and $\mathbb{P}_1$ in Ω_2 (slave). At top $h_2 = 1/8$, at bottom $h_2 = 1/32$.

Table 10
FEM–SEM coupling: orders of convergence w.r.t. the mesh-sizes h_1 in Ω_1 and h_2 in Ω_2 when using non-conforming meshes. The cells are organized as in Table 3.

Master–slave	Lagrange		RL-RBF		mortar	
$\mathbb{P}_1$–$\mathbb{Q}_2$	1–1.5	1–2	1–2	1–2	1–2	1–2
	1–2	1–1.5	1–2	1–2	1–2	1–2
$\mathbb{P}_1$–$\mathbb{Q}_3$	1–1	1–2	1–2	1–2	1–1.5	1–2
	1–1	1–1.5	1–2	1–2	1–2	1–2
$\mathbb{P}_1$–$\mathbb{Q}_4$	1–1	1–2	1–2	1–2	1–1.5	1–2
	1–1	1–1	1–2	1–2	1–2	1–2

Table 11
SEM–FEM coupling: orders of convergence w.r.t. the respective mesh-sizes h_1 in Ω_1 and h_2 in Ω_2 when using non-conforming meshes. The cells are organized as in Table 3.

Master–slave	Lagrange		RL-RBF		mortar	
$\mathbb{Q}_2$–$\mathbb{P}_1$	2–1	2–1	2–1	2–1	2–1	2–1
	2–1	2–1	2–1	2–1	2–1	2–1
$\mathbb{Q}_3$–$\mathbb{P}_1$	2–1	2–1	2–1	2–1	2–1	2–1
	2–1	2–1	2–1	2–1	2–1	2–1
$\mathbb{Q}_4$–$\mathbb{P}_1$	2–1	2–1	2–1	2–1	2–1	2–1
	2–1	2–1	2–1	2–1	2–1	2–1

Ω_1 is observed in Ω_2 as well, even if in Ω_2 a higher degree, $p \geq 2$, is used. Conversely, when using RL-RBF, the order of convergence in the slave domain is equal to 2, thus reflecting the more accurate discretization used. In any case, even when $p > 2$ the order of convergence in Ω_2 is still 2.

When the master domain is discretized more finely by $\mathbb{Q}_p$ ($p \geq 2$) and $\mathbb{P}_1$ are used in the slave, the rates of convergence are 2 and 1 in the master and the slave domain, respectively, for both the approaches.

Mesh non-conformity: We consider now non-conforming meshes and, as in the previous sections, four different situations, characterized by a varying aspect ratio between the mesh sizes h_1 and h_2, as well as by the refinements of the grids. In Tables 10 and 11 the convergence orders w.r.t. the mesh sizes h_1 and h_2 are shown. As for the mesh conforming case, when the master domain is discretized by $\mathbb{P}_1$, the Lagrange interpolation downgrades the higher approximation degree of the slave domain, while the RL-RBF interpolation always provides convergence order 1 in the master domain and 2 in the slave one. On the contrary, when the master domain is discretized by SEM, the convergence orders w.r.t. h are 2 in the master domain and 1 in the slave one for all the considered approaches.

The higher accuracy of the SEM discretization is downgraded by that of the $\mathbb{P}_1$ FEM approximation. To verify this statement,

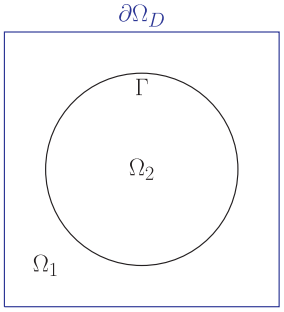


Fig. 11. Domain decomposition considered for the example with geometrically non-conforming interfaces.

we fix now the discretization in Ω_1 (master domain) by using 6 $\times$ 6 quads $\mathbb{Q}_4$, while we refine the $\mathbb{P}_1$ mesh in Ω_2 by choosing $h_2 = 1/8,\ 1/16,\ 1/32,\ 1/64,\ 1/128$. In Fig. 9, the H^1 norm of the errors with respect to the exact solution are shown versus h_2. The error in Ω_1 decays as h_2^2 until the accuracy prescribed by global $\mathbb{Q}_4$ discretization is reached, while the error in Ω_2 is $\mathcal{O}(h_2)$. In Fig. 10 we show the meshes used and contours of the computed solutions when $h_2 = 1/8$ and $h_2 = 1/32$. The rough approximation around the interface is evident when $h_2 = 1/8$.

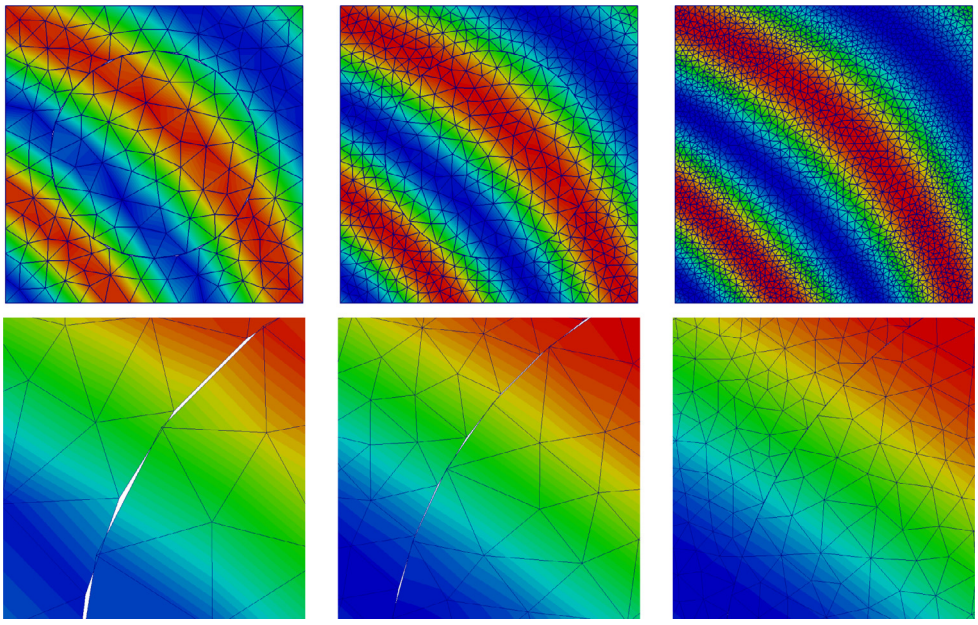
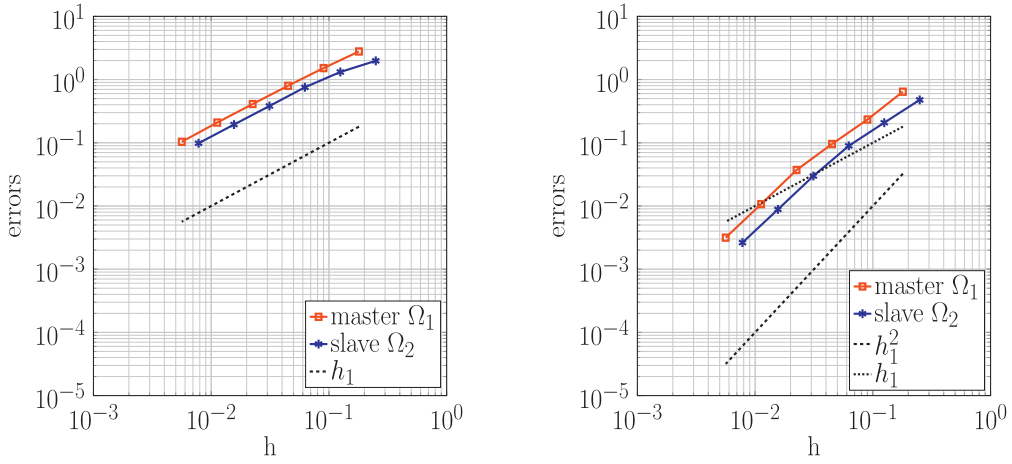


Fig. 12. Solutions obtained using $\mathbb{P}_1$–$\mathbb{P}_1$ finite elements for the master and slave domains using meshes of increasing refinement that are geometrically non-conforming. In the top row we show the numerical results on the whole domain while in the bottom row a zoom of the solution close to the interface is displayed.



(a) Master $\mathbb{P}_1$ (finer grid), slave $\mathbb{P}_1$. (b) Master $\mathbb{P}_2$ (finer grid), slave $\mathbb{P}_2$.

Fig. 13. Rates of convergence in H^1 norm using geometrically non-conforming interfaces.

8. Coupling of geometrically non-conforming subdomains

To assess the robustness of INTERNODES with respect to the geometrically non-conforming case, we solve problem (42) with exact solution $u(x, y) = \sin((x - 1.2)(y - 1.2)2\pi) + 1$ in the domain $\Omega = (-0.5, 0.5) \times (-0.5, 0.5)$, decomposed as shown in Fig. 11. The inner circle is centered at point (0, 0) and has radius $R = 0.35$.

In Fig. 12 we show the results obtained using $\mathbb{P}_1$ finite elements in both the slave and the master domains. Although many gaps and overlaps are present at the interface between the subdomain grids (see bottom row in Fig. 12), we observe that the quality of the numerical solutions obtained does not worsen. In Fig. 13 the results obtained using first or second degree polynomials in the master and slave domains are reported. From Fig. 13(a) we observe that when using linear basis functions, INTERNODES leads to first order rate of convergence in both the master and the slave subdomains. Finally, as shown in Fig. 13(b), we notice that quadratic convergence is obtained with $\mathbb{P}_2$ finite elements only for sufficiently small mesh sizes for which the gaps and overlaps between the master and slave subdomains tend to become imperceptible (see bottom-right picture of Fig. 12).

9. Diffusion of the pollutant around an industrial chimney

We consider now a 2D numerical simulation that mimics the diffusion of the pollutant concentration $u(\mathbf{x})$ in a bounded region around an industrial chimney. To this aim we solve problem (1) in $\Omega = (0, 1)^2$, with constant diffusion coefficient $\mu = 10^{-3}$, null reaction coefficient α, null convection $\mathbf{b}$ and chimney discharge $f = \chi_\omega$, being χ_ω the characteristic function of the set $\omega = \{\mathbf{x} \in \mathbb{R}^2 : |\mathbf{x} - \mathbf{x}_c| < 0.02\}$ and $\mathbf{x}_c = (0.5, 0.5)$. Homogeneous boundary conditions are set on the boundary $\partial\Omega$.

Since the largest variation of the solution occurs in a small region around the set ω, the computational domain Ω is split into the subsets $\Omega_2 = (0.45, 0.55)^2$ and $\Omega_1 = \Omega \setminus \Omega_2$, then independent meshes are designed there, with the aim to better approximate the solution in proximity of the chimney position. More precisely, 5×5 quads $\mathbb{Q}_8$ are used in Ω_2 (for a total amount of 1681

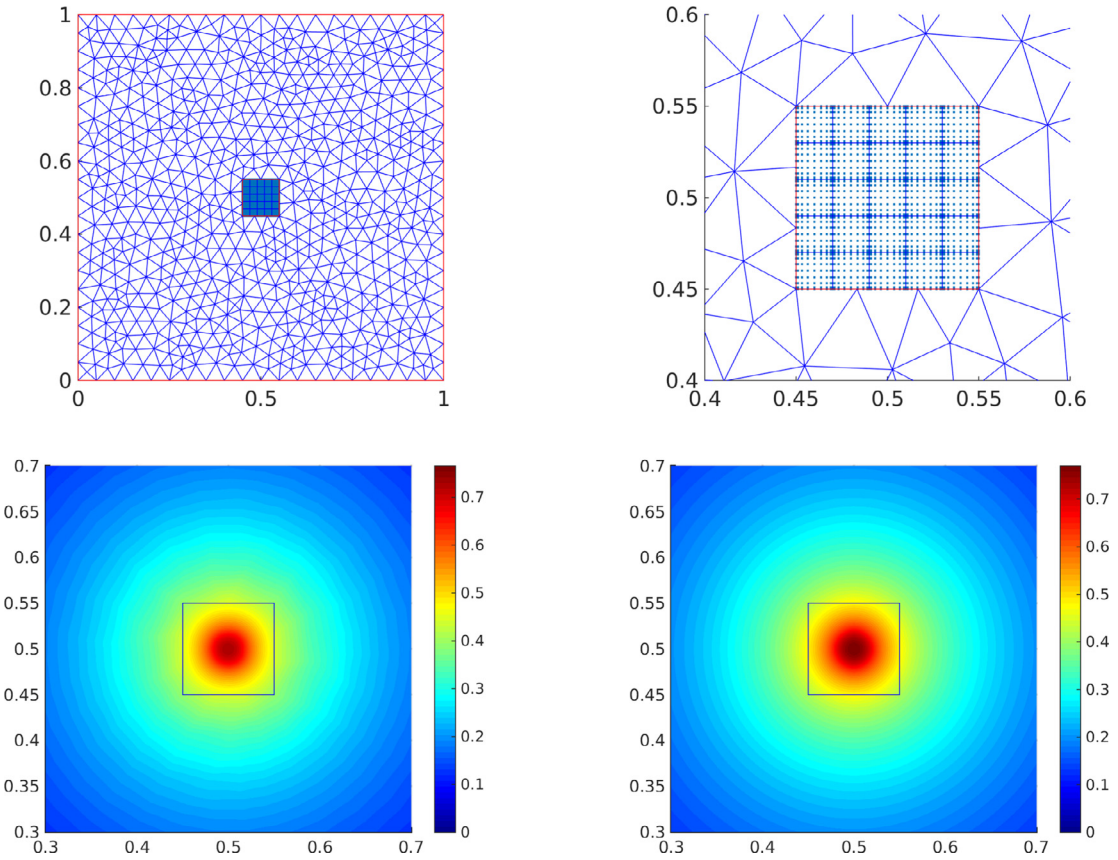


Fig. 14. At top, the mesh used in the simulation of the diffusion of pollutant concentration (the right picture is a zoom of the left one). At bottom, the numerical solution obtained by both the INTERNODES with Lagrange interpolant as intergrid operator (at left) and by monodomain conforming SEM $\mathbb{Q}_{32}$ on the whole Ω (at right).

degrees of freedoms) and $\mathbb{P}_1$ FEM with $h_{max} = 1/20$ in Ω_1 (711 d.o.f and 1331 triangles).

INTERNODES is used to compute the numerical solution, using either the Lagrange or the RL-RBF interpolants as intergrid operators. The external domain Ω_1 plays the role of master (choosing Ω_2 as master would yield a singular problem, as the reaction coefficient α is null and $\partial\Omega_2 = \Gamma_2$ is sctrictly internal to Ω).

In Fig. 14 we plot both the mesh (top) and the pollutant concentration (bottom left) obtained by solving problem (1) by INTERNODES method with Lagrange interpolation at the interfaces. The plot at bottom right of Fig. 14 refers to the numerical solution obtained by conforming SEM on a uniform mesh of 8×8 quads in Ω with polynomial degree $p = 32$. The range of the color bars used to plot the two numerical solutions is the same, the black box marks the interface Γ between the two subdomains.

In Fig. 15 we plot the traces along Γ of the numerical solutions $u_{1,\delta}$ and $u_{2,\delta}$ shown in Fig. 14 bottom-left, as well as the trace of the monodomain SEM solution u_{SEM} of Fig. 14 bottom-right. The curves are plotted versus the curvilinear abscissa s that starts from the point (0.45, 0.45) and moves counterclockwise along Γ.

This test highlights the numerical robustness of INTERNODES when addressing grid non-conformity featuring severe aspect ratio.

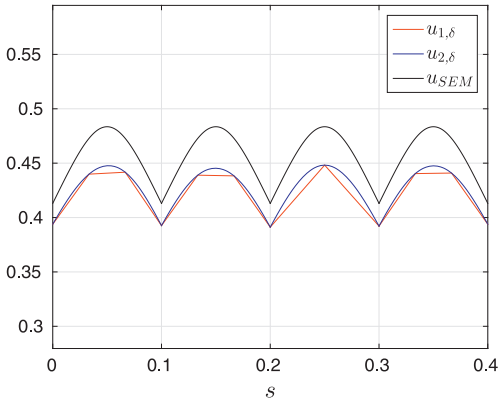


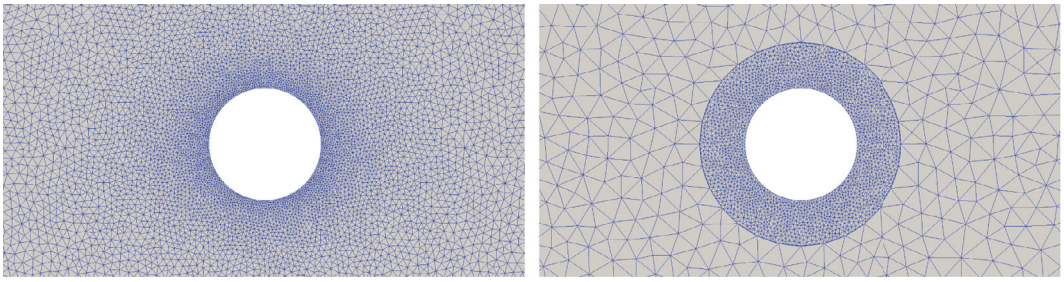
Fig. 15. Traces of the numerical solutions plotted in Fig. 14 versus the curvilinear abscissa s that starts from the point (0.45, 0.45) and moves counterclockwise along Γ.

10. Numerical solution of a fluid flow past a cylinder

In this Section we consider the numerical simulation of a fluid flow past a 3D cylindrical obstacle at two different Reynolds numbers, that are $Re = 20$ and $Re = 100$, see [54].

We model the flow dynamics by the Navier–Stokes equations for an incompressible fluid. The equations are discretized in space by means of the Finite Element method and in time by Finite Differences. More specifically, we use $\mathbb{P}_1$-$\mathbb{P}_1$ finite elements for the spatial approximation of the fluid velocity and pressure variables (stabilized by SUPG), respectively, while a second order backward differentiation formula is used for the time discretization (see, e.g. [49]). The nonlinear convective term in the fluid momentum equation is linearized by means of a second order temporal extrapolation [30,31].

In order to compare the numerical results obtained by INTERNODES with those available in the literature we compute the drag and lift coefficients of the cylinder. To this end, we introduce a



(a) Boundary layer refinement obtained by progressive reduction of the mesh size (cross section).

(b) Independent meshes for the far field and the boundary layer zones (cross section).

Fig. 16. Two different strategies to realize boundary layers mesh refinements.

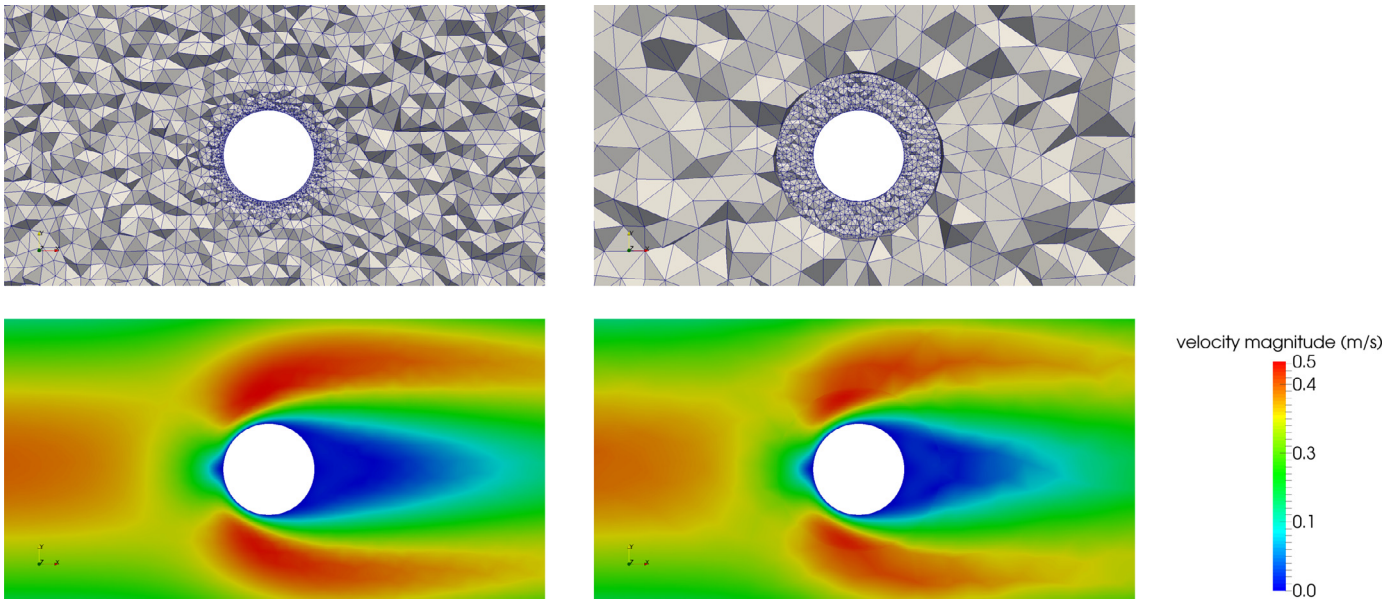


Fig. 17. Zoom of the meshes and the velocity fields in the region close to the cylinder on a cut plane at $z = 0.205$ m: in the first row we show the meshes used for the numerical simulations. In the second row we plot the velocity fields obtained.

unit vector directed as the incoming flow $\hat{\mathbf{v}}_\infty = \frac{\mathbf{v}_\infty}{\|V_\infty\|}$, and a unit vector $\hat{\mathbf{n}}_\infty$ orthogonal to $\hat{\mathbf{v}}_\infty$. The aerodynamic drag and lift coefficients for the cylinder read:

$$C_D(\mathbf{u}, p) = -\frac{1}{q_\infty S}\oint_S (\boldsymbol{\sigma}_f(\mathbf{u}, p)\,\hat{\mathbf{n}})\cdot\hat{\mathbf{v}}_\infty\, d\Gamma, \tag{44}$$

$$C_L(\mathbf{u}, p) = \frac{1}{q_\infty S}\oint_S (\boldsymbol{\sigma}_f(\mathbf{u}, p)\,\hat{\mathbf{n}})\cdot\hat{\mathbf{n}}_\infty\, d\Gamma, \tag{45}$$

where $\mathbf{u}$ and p are the velocity and pressure variables, $\boldsymbol{\sigma}_f$ is the Cauchy stress tensor of the fluid, $q_\infty = \frac{1}{2}\rho V_\infty^2$ is the dynamic pressure, ρ is the density of the fluid and S is the surface area of the cylinder.

For an accurate estimation of the aerodynamic coefficients, the use of a boundary layer refinement of the computational mesh around the cylinder is mandatory (see [32]). A possible strategy to generate such a refinement consists in gradually decreasing the mesh element size in the domain while approaching the cylinder, as shown in Fig. 16(a). An alternative strategy relies in splitting the computational fluid domain into two sub-domains with independent (non-conforming) meshes, see Fig. 16(b): the finer mesh is used to represent the boundary layer around the cylinder while the coarse one for the far field. In the latter case, after space and time discretization of the Navier–Stokes equations, the algebraic form of the linear system to be solved reads

$$\begin{bmatrix} F_1 & 0 & -I_{\Gamma_1}^T M_{\Gamma_1}\Pi_{12}M_{\Gamma_2}^{-1} \\ 0 & F_2 & I_{\Gamma_2}^T \\ -\Pi_{21}I_{\Gamma_1} & I_{\Gamma_2} & 0 \end{bmatrix} \begin{bmatrix} (\mathbf{u}_{f_1}, p_{f_1})^T \\ (\mathbf{u}_{f_2}, p_{f_2})^T \\ \boldsymbol{\lambda}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ 0 \end{bmatrix}. \tag{46}$$

In (46), we denoted by F_1 and F_2 the fully discrete form of the fluid problem in the master and slave portions of the domain, respectively. The matrices I_{Γ_1} and I_{Γ_2} restrict vectors defined on the master and slave domain to their interface, respectively, and account for the continuity of the velocities (imposed strongly). Their transposes account for the continuity of the normal stresses (imposed weakly). We notice that system (46) is written in the so-called augmented form as we introduced the auxiliary variable $\boldsymbol{\lambda}_2$ which can be regarded as a vector of Lagrange multipliers used to enforce the continuity of the velocities at the interface.

We solve (46) using the preconditioned GMRES method. The preconditioner is built up by neglecting block $-I_{\Gamma_1}^T M_{\Gamma_1}\Pi_{12}M_{\Gamma_2}^{-1}$ in

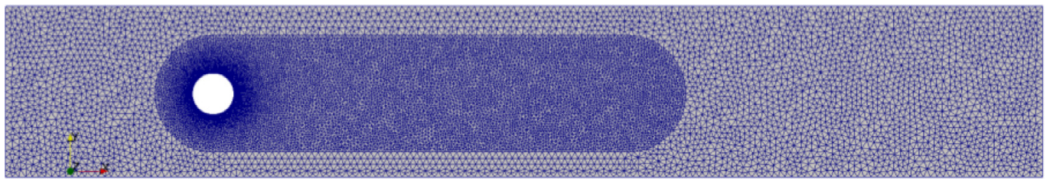


Fig. 18. Horizontal view of the meshes used for the numerical example at Reynolds 100.

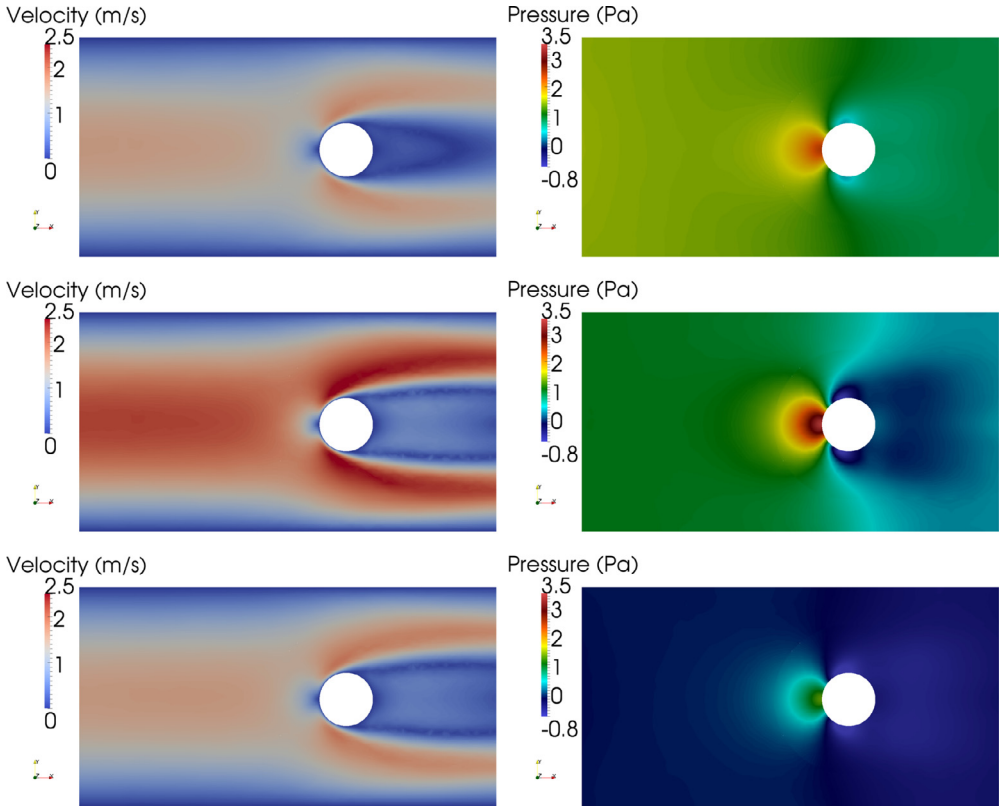


Fig. 19. Visualization of the numerical solution obtained at times $t = 2\,s$ (top row), $t = 4\,s$ (middle row) and $t = 6\,s$ (bottom row). On the left column we report the fluid velocity while on the right the fluid pressure.

(46) yielding a new matrix P that can be factorized as follows

$$P = P_1 \cdot P_2 = \begin{bmatrix} F_1 & 0 & 0 \\ 0 & I & 0 \\ -\Pi_{21} I_{\Gamma_1} & 0 & I \end{bmatrix} \begin{bmatrix} I & 0 & 0 \\ 0 & F_2 & I_{\Gamma_2}^T \\ 0 & I_{\Gamma_2} & 0 \end{bmatrix}$$
$$\approx \begin{bmatrix} \widetilde{F}_1 & 0 & 0 \\ 0 & I & 0 \\ -\Pi_{21} I_{\Gamma_1} & 0 & I \end{bmatrix} \begin{bmatrix} I & 0 & 0 \\ 0 & \widetilde{F}_2 & I_{\Gamma_2}^T \\ 0 & I_{\Gamma_2} & 0 \end{bmatrix} = \widetilde{P}_1 \cdot \widetilde{P}_2 = \widetilde{P}. \qquad (47)$$

P_1 refers solely to the master domain and P_2 to the slave. The further approximation that we have operated consists in replacing both F_1 and F_2 with a SIMPLE preconditioner [23], $\widetilde{F}_1$ in Ω_1 and $\widetilde{F}_2$ in Ω_2. Notice that F_2 encodes Dirichlet conditions on the velocity field at the interface, whereas F_1 is associated with Neumann conditions on the normal stresses, and SIMPLE changes accordingly.

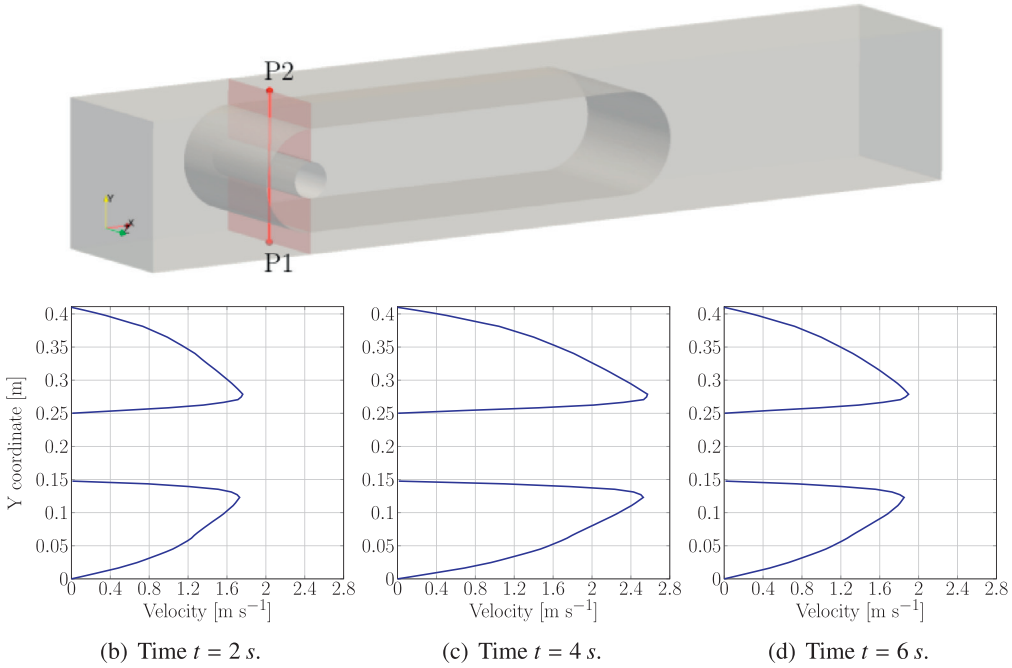
Remark 10.1. The new preconditioner $\widetilde{P}$ can in fact be formally derived from FaCSI, a preconditioner introduced in [21] for fluid-structure interaction problems, in the case in which the interface movement is dealt with explicitly in time and the block pertaining to the structure is replaced by the fluid problem in the domain Ω_1.

The values of the physical parameters for the fluid as well as the boundary conditions used in our simulations are those described in [54]. The essential boundary condition on the fluid velocity at the cylinder surface is imposed in weak form [2,33].

Table 12
Numerical results obtained for the benchmark problem at $Re = 20$. The reference drag and lift coefficients are $C_D = 6.18533$ and $C_L = 0.009401$, respectively.

Mesh level	DOF	C_D	C_L	% error C_D	% error C_L
1	149,004	6.39269	0.003331	3.35	64.56
2	250,400	6.26037	0.005431	1.21	42.22
3	531,992	6.20392	0.006838	0.31	27.26
4	1,209,060	6.17584	0.009412	0.15	0.11

In Table 12 we report the drag and lift coefficients of the cylinder obtained using different fluid meshes in which the boundary layer refinement was obtained as in Fig. 16(a). We used 373,068 degrees of freedom in the master domain and 103,768 in the slave. The coarser mesh is used to represent the far field while the finer one is around the cylinder. In this example, the mesh-size used for the coarser mesh is the one of Mesh Level 1 while the one of the finer mesh coincides with the mesh-size of Mesh level 4 (see Table 12). Furthermore, the far field domain is considered as slave domain while the one around the cylinder is the master. In this way the interface degrees of freedom of the master (finer) domain are primal unknowns of the problem (see Eq. (29)). We remark that the ratio between the mesh size of the far field and the one in the boundary layer is approximately 6.



(b) Time $t = 2\ s$. (c) Time $t = 4\ s$. (d) Time $t = 6\ s$.

Fig. 20. Plot over line, between points P1 = (0.5,0,0.205) and P2 = (0.5,0.41,0.205), of the flow velocity at different times.

The numerical simulation performed using two non-conforming meshes yields a lift coefficient $C_L = 0.009487$ and a drag coefficient $C_D = 6.19713$. The errors with respect to reference values are 0.91% and 0.19% on the estimation of the lift and the drag coefficients, respectively. To analyze the computational costs, we compare the average time to complete a single time step and the number of linear solver iterations on the fully conforming simulation with Mesh level 4 of Table 12 and the one by INTERNODES with non-conforming meshes. The approach based on the use of non-conforming meshes leads to a reduction of the 50% of the time to perform a time step (thanks to the fact that the number of degrees with non-conforming meshes is roughly half of the one of Mesh level 4). Furthermore, we observe that the average (over the time steps) number of linear solver iterations is 25, and does not differ from the one of a fully conforming case.

Finally, we report the results obtained for $Re = 100$. In Fig. 18 we show the fluid meshes considered for the region close to the obstacle and the far field. In this unsteady case, the geometry in which we use a finer mesh is extended to embed also the wake region behind the cylinder. We consider the far field domain as the slave domain, while the one near the cylinder as master. The computational meshes yield 281,393 degrees of freedom in the slave domain and 844,179 in the master. The aspect ratio between the mesh sizes of the slave and master domains at their interface is approximately 3.

The same problem of fluid flow past a cylinder has also been investigated in [25] by a dual mortar approach for non-conforming meshes tying a fine boundary layer grid in the near field with a coarser grid in the far field.

In Fig, 19 we show the numerical results computed at different times on a cut plane parallel to the z axis (located at $z = 0.205$ m): we notice that both the velocity and pressure solutions obtained in the master and slave domains are in very good agreement at their interface. In addition, we notice that the velocity field computed at time $t = 4$ s (shown in Fig. 19) is in very good agreement w.r.t. the one reported in [25], Fig. 16.

Table 13
Comparison of the aerodynamic coefficients computed with reference values available in literature [54].

	Maximum C_D	Maximum C_L	Minimum C_L
Computed	3.3017	0.0029	−0.011017
Reference	3.2978	0.0028	−0.010999

To better assess the behavior of the solution across the interface, in Fig. 20 we plot the fluid velocity along a vertical line passing through both the master and slave sub-domains.

Finally, in Table 13, we compare the aerodynamic coefficients of the cylinder computed numerically with those available in literature.

11. Conclusions

In this work we introduced INTERNODES, an accurate and easy to implement interpolation based method for coupling the solutions of PDEs on subdomains that feature non-conforming discretizations (non-conforming grids, non-conforming polynomial degrees and non-matching geometries). Individual subdomains are discretized either by the finite element method or by the spectral element method.

The non-conforming problem was formulated in variational form as a generalized Galerkin problem in which two intergrid operators are designed for transferring the solution and its fluxes across the non-conforming subdomain interfaces. These operators are based on interpolation: both Lagrangian and Radial Basis Functions interpolation were considered in this work.

We extensively investigated the convergence properties of INTERNODES by numerically solving an elliptic problem in which the subdomains were discretized by non-conforming FEM–FEM, SEM–SEM and SEM–FEM (even in the presence of geometrically non-matching interfaces).

A comparative study with the mortar method was carried out and we showed that the orders of convergence obtained by

INTERNODES are essentially the same as those generated by the mortar method.

Finally, we tested the new method by solving the benchmark problem of the fluid flow past a cylinder at different Reynolds numbers in which independent (non-conforming) meshes for the boundary layer and the far field parts of the domain were considered.

INTERNODES turns out to be very effective in treating severe grid non-conformity and polynomial non-conformity.

Acknowledgments

The research of D. Forti was supported by the Swiss National Foundation (SNF), Project no. 140184. The research of S. Deparis and A. Quarteroni was partly supported by the PASC Project "Integrative HPC Framework for Coupled Cardiac Simulations". S. Deparis, D. Forti and A. Quarteroni gratefully acknowledge the Swiss National Supercomputing Center (CSCS) for providing the CPU resources for the numerical simulations under project ID s475.

References

[1] Baaijens FPT. A fictitious domain/mortar element method for fluid–structure interaction. Int J Numer Methods Fluids 2001;35(7):743–61.
[2] Bazilevs Y, Hughes T. Weak imposition of dirichlet boundary conditions in fluid mechanics. Comput Fluids 2007;36:12–26.
[3] Bègue C, Bernardi C, Debit N, Maday Y, Karniadakis G, Mavriplis C, et al. Nonconforming spectral element-finite element approximations for partial differential equations, In: Proceedings of the eighth international conference on computing methods in applied sciences and engineering, Versailles, 1987. Comput Methods Appl Mech Eng 1989;75(1–3):109–25.
[4] Belgacem FB. The mortar finite element method with Lagrange multipliers. Numer Math 1999;84(2):173–97.
[5] Belgacem FB. The mixed mortar finite element method for the incompressible Stokes problem: convergence analysis. SIAM J Numer Anal 2000;37(4) 1085–1100(electronic).
[6] Ben Belgacem F, Buffa A, Maday Y. The mortar finite element method for 3d maxwell equations: first results. SIAM J Numer Anal 2001;39(3):880–901.
[7] Ben Belgacem F, Chilton LK, Seshaiyer P. The hp-mortar finite-element method for the mixed elasticity and stokes problems. Comput Math Appl 2003;46(1):35–55.
[8] Ben Belgacem F, Hild P, Laborde P. Extension of the mortar finite element method to a variational inequality modeling unilateral contact. Math Models Methods Appl Sci 1999;9(2):287–303.
[9] Bernardi C, Maday Y. Mesh adaptivity in finite elements using the mortar method. Rev Eur Élém Finis 2000;9(4):451–65.
[10] Bernardi C, Maday Y, Patera A. Domain decomposition by the mortar element method. In: Asymptotic and numerical methods for partial differential equations with critical parameters (Beaune, 1992). In: Volume 384 of NATO Advanced Science Institute series c: mathematical and physical sciences. Dordrecht: Kluwer Academic Publishers; 1993. p. 269–86.
[11] Bernardi C, Maday Y, Patera A. A new nonconforming approach to domain decomposition: the mortar element method. In: Nonlinear partial differential equations and their applications: Collège de France seminar volume XI (Paris, 1989–1991). In: Volume 299 of Pitman research notes in mathematics series. Harlow: Longman Sci. Tech.; 1994. p. 13–51.
[12] Bernardi C, Maday Y, Rapetti F. Discrétisations variationnelles de problèmes aux limites elliptiques. Volume 45 of mathématiques & applications. Berlin: Springer-Verlag; 2004. ISBN 3-540-21369-4.
[13] Bouillault F, Buffa A, Maday Y, Rapetti F. The mortar edge element method in three dimensions: application to magnetostatics. SIAM J Sci Comput 2003;24(4):1303–27.
[14] Brivadis E, Buffa A, Wohlmuth BI, Wunderlich L. Isogeometric mortar methods. Comput Methods Appl Mech Eng 2015;284:292–319.
[15] Buffa A, Maday Y, Rapetti F. A sliding mesh-mortar method for a two dimensional eddy currents model of electric engines. ESAIM: Math Model Numer Anal – Modél Math Anal Numér 2001;35(2):191–228.
[16] Buhmann M. Radial basis functions: theory and implementations, vol. 12. Cambridge University Press; 2003.
[17] Canuto C, Hussaini MY, Quarteroni A, Zang TA. Spectral methods. Fundamentals in single domains. Heidelberg: Springer; 2006.
[18] Canuto C, Hussaini MY, Quarteroni A, Zang TA. Spectral methods. Evolution to complex geometries and applications to fluid dynamics. Heidelberg: Springer; 2007.
[19] Cottrell JA, Hughes TJR, Bazilevs Y. Isogeometric analysis. Towards integration of CAD and FEA. Chichester: Wiley; 2009.
[20] De Lorenzis L, Wriggers P, Zavarise G. A mortar formulation for 3d large deformation contact using Nurbs-based isogeometric analysis and the augmented Lagrangian method. Comput Mech 2012;49(1):1–20.
[21] Deparis S., Forti D., Grandperrin G., Quarteroni A.. Facsi: a block parallel preconditioner for fluid–structure interaction in hemodynamics2016[submitted for publication].
[22] Deparis S, Forti D, Quarteroni A. A rescaled localized radial basis function interpolation on non-Cartesian and nonconforming grids. SIAM J Sci Comput 2014a;36:A2745–62.
[23] Deparis S, Grandperrin G, Quarteroni A. Parallel preconditioners for the unsteady Navier–Stokes equations and applications to hemodynamics simulations. Comput Fluids 2014b;92:253–73. http://dx.doi.org/10.1016/j.compfluid.2013.10.034.
[24] Dickopf T, Krause R. Efficient simulation of multi-body contact problems on complex geometries: a flexible decomposition approach using constrained minimization. Int J Numer Methods Eng 2009;77:1834–62.
[25] Ehrl A, Popp A, Gravemeier V, Wall W. A dual mortar approach for mesh tying within a variational multiscale method for incompressible flow. Int J Numer Methods Fluids 2014;76(1):1–27. doi:10.1002/fld.3920.
[26] Faucher V, Combescure A. A time and space mortar method for coupling linear modal subdomains and non-linear subdomains in explicit structural dynamics. Comput Methods Appl Mech Eng 2003;192(5):509–33.
[27] Fischer KA, Wriggers P. Frictionless 2d contact formulations for finite deformations based on the mortar method. Comput Mech 2005;36(3):226–44.
[28] Flemisch B, Puso MA, Wohlmuth BI. A new dual mortar method for curved interfaces: 2d elasticity. Int J Numer Methods Eng 2005;63(6):813–32.
[29] . Cardiovascular mathematics Modeling and simulation of the circulatory system. In: Formaggia L, Quarteroni A, Veneziani A, editors. Volume 1 of MS&A. Modeling, simulation and applications. Milan, Italia: Springer-Verlag; 2009. ISBN 978-88-470-1151-9. http://dx.doi.org/10.1007/978-88-470-1152-6.
[30] Forti D, Dedè L. Semi-implicit BDF time discretization of the Navier–Stokes equations with VMS–LES modeling in high performance computing framework. Comput Fluids 2015;117:168–82.
[31] Gervasio P, Saleri F, Veneziani A. Algebraic fractional step schemes with spectral methods for the incompressible Navier–Stokes equations. J Comput Phys 2006;214(1):347–65.
[32] Ghia U, Ghia K, Shin C. High-Re solutions for incompressible flow using the Navier–Stokes equations and a multigrid method. J Comput Phys 1982;48:387–411.
[33] Hsu MC, Akkerman I, Bazilevs Y. Wind turbine aerodynamics using ALE–VMS: validation and the role of weakly enforced boundary conditions. Comput Mech 2012;50:499–511.
[34] Klöppel T, Popp A, Küttler U, Wall WA. Fluid–structure interaction for non-conforming interfaces based on a dual mortar formulation. Comput Methods Appl Mech Eng 2011;200:3111–26.
[35] Lacour C, Ben Belgacem F. The mortar finite element method: basics, theory and implementation. Chapman and Hall/CRC; 2008.
[36] Lamichhane PB, Wohlmuth BI. Mortar finite elements for interface problems. Computing 2004;72(3):333–48. doi:10.1007/s00607-003-0062-y.
[37] Lombardi M, Parolini N, Quarteroni A, Rozza G. Numerical simulation of sailing boats: dynamics, FSI, and shape optimization. In: Buttazzo G, Frediani A, editors. Variational analysis and aerospace engineering: mathematical challenges for aerospace design. Springer; 2012. p. 339–77.
[38] Maday Y, Mavripilis C, Patera A. Nonconforming mortar element methods: application to spectral discretizations. In: Périeaux J, Chan TF, Glowinski R, Widlund OB, editors. Proceedings of the second international conference on domain decomposition methods for partial differential equations. Philadelphia: SIAM; 1988.
[39] Maday Y, Rapetti F, Wohlmuth B. The influence of quadrature formulas in 2D and 3D mortar element methods. In: Recent developments in domain decomposition methods (Zürich, 2001). In: Volume 23 of lecture notes in engineering and computer science. Springer; 2002. p. 203–21.
[40] Mayer UM, Popp A, Gerstenberger A, Wall WA. 3d fluid–structure-contact interaction based on a combined XFEM FSI and dual mortar contact approach. Computat Mech 2010;46(1):53–67. doi:10.1007/s00466-010-0486-0.
[41] Parolini N, Quarteroni A. Mathematical models and numerical simulations for the Americas Cup. Comput Methods Appl Mech Eng 2005;194(9):1001–26.
[42] Popp A. Mortar methods for computational contact mechanics and general interface problems [Ph.D. thesis]. Technische Universität München, Münich; 2012.
[43] Popp A, Gee MW, Wall WA. A finite deformation mortar contact formulation using a primal–dual active set strategy. Int J Numer Methods Eng 2009;79(11):1354–91.
[44] Popp A, Gitterle M, Gee MW, Wall WA. A dual mortar approach for 3d finite deformation contact with consistent linearization. Int J Numer Methods Eng 2010;83(11):1428–65.
[45] Popp A, Wohlmuth BI, Gee MW, Wall WA. Dual quadratic mortar finite element methods for 3d finite deformation contact. SIAM J Sci Comput 2012;34(4):B421–46.
[46] Puso MA. A 3d mortar method for solid mechanics. Int J Numer Methods Eng 2004;59(3):315–36.
[47] Puso MA, Laursen TA. A mortar segment-to-segment contact method for large deformation solid mechanics. Comput Methods Appl Mech Eng 2004;193(6):601–29.
[48] Puso MA, Laursen TA. Mesh tying on curved interfaces in 3d. Eng Comput 2003;20(3):305–19.
[49] Quarteroni A. Numerical models for differential problems. MS&A series, vol. 2. 2nd ed. Milano: Springer; 2013.

[50] Quarteroni A, Valli A. Numerical approximation of partial differential equations. Heidelberg: Springer Verlag; 1994.
[51] Quarteroni A, Valli A. Domain decomposition methods for partial differential equations. Oxford Science Publications; 1999.
[52] Rapetti F, Bouillault F, Santandrea L, Buffa A, Maday Y, Razek A. Calculation of eddy currents with edge elements on non-matching grids in moving structures. IEEE Trans Magn 2000;36(4):1351–5.
[53] Reymond P, Crosetto P, Deparis S, Quarteroni A, Stergiopulos N. Physiological simulation of blood flow in the aorta: comparison of hemodynamic indices as predicted by 3-d FSI, 3-d rigid wall and 1-d models. Med Eng Phys 2013;35(6):784–91.
[54] Schäfer M, STurek. Benchmark computations of laminar flow around a cylinder. In: Hirschel E, editor. Flow simulation with high-performance computers II. DFG priority research program results 1993–1995. Volume 52 of notes on numerical fluid mechanics. Vieweg: Vieweg, Weisbaden; 1996. p. 547–66.
[55] Sherwin SJ, Karniadakis G. A triangular spectral element method; applications to the incompressible Navier–Stokes equations. Comput Methods Appl Mech Eng 1995;123:189–229.
[56] Wendland H. Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. Adv Comput Math 1995;4(1):389–96.
[57] Wohlmuth BI. A mortar finite element method using dual spaces for the Lagrange multiplier. SIAM J Numer Anal 2000;38:989–1012.