AN ADDITIVE SCHWARZ METHOD TYPE THEORY FOR LIONS' ALGORITHM AND A SYMMETRIZED OPTIMIZED RESTRICTED ADDITIVE SCHWARZ METHOD*

R. HAFERSSAS†, P. JOLIVET‡, AND F. NATAF§

Résumé. Optimized Schwarz methods (OSM) are very popular methods which were introduced by P.L. Lions in [32] for elliptic problems and by B. Després in [8] for propagative wave phenomena. We give here a theory for Lions' algorithm that is the genuine counterpart of the theory developed over the years for the Schwarz algorithm. The first step is to introduce a symmetric variant of the ORAS (Optimized Restricted Additive Schwarz) algorithm [44] that is suitable for the analysis of a two-level method. Then we build a coarse space for which the convergence rate of the two-level method is guaranteed regardless of the regularity of the coefficients. We show scalability results for thousands of cores for nearly incompressible elasticity and the Stokes systems with a continuous discretization of the pressure.

1. Introduction. Substructuring algorithms such as BNN or FETI are defined for nonoverlapping domain decompositions but not for overlapping subdomains. Schwarz method [41] is defined only for overlapping subdomains. With the help of a coarse space correction, the two-level versions of both type of methods are weakly scalable, see [45] and references therein. The domain decomposition method introduced by P.L. Lions [32] is a third type of methods. It can be applied to both overlapping and nonoverlapping subdomains. It is based on improving Schwarz methods by replacing the Dirichlet interface conditions by Robin interface conditions. This algorithm was extended to Helmholtz problem by Després [9]. Robin interface conditions can be replaced by more general interface conditions that can be optimized (Optimized Schwarz methods, OSM) for a better convergence, see [21, 20] and references therein.

P.L. Lions proved the convergence of his algorithm in the elliptic case for a nonoverlapping domain decomposition. The proof is based on energy estimates and a summation technique. These results were extended to Helmholtz and Maxwell equations in [2, 10]. Over the last years, a lot of results have been obtained for different classes of equations and optimized algorithms based on carefully chosen parameters in the transmission conditions, have been derived, see e.g. [25, 20, 21, 11] and references therein. Most of these works are valid for nonoverlapping decomposition or for simple overlapping domain decompositions as in [28, 35] for the two-subdomain case. When the domain is decomposed into a large number of subdomains, these methods are, on a practical point of view, scalable if a second level is added to the algorithm via the introduction of a coarse space [25, 17, 7, 13, 34]. But there is no systematic procedure to build coarse spaces with a provable efficiency for general symmetric positive definite systems.

The purpose of this article is to define a general framework for building adaptive coarse space for OSM methods for decomposition into overlapping subdomains. We prove that we can achieve the same robustness that what was done for Schwarz [42] and FETI-BDD [43] domain decomposition methods with so called GenEO (Generalized

*Submitted to the editors DATE.

†CNRS, UMR 7598, Laboratoire Jacques-Louis Lions, F-75005, Paris, France (ryadh.haferssas@ljll.math.upmc.fr).

‡Toulouse Institute of Computer Science Research, France (pierre.jolivet@enseeiht.fr).

§CNRS, UMR 7598, Laboratoire Jacques-Louis Lions, F-75005, Paris, France (nataf@ljll.math.upmc.fr)

Eigenvalue in the Overlap) coarse spaces. Compared to these previous works, we have to introduce SORAS (symmetrized ORAS) a non standard symmetric variant of the ORAS method as well as two generalized eigenvalue problems. As numerical results will show in § 6.3, the method scales very well for saddle point problems such as highly heterogeneous nearly incompressible elasticity problems as well as the Stokes system. More precisely, in § 2, we give a short presentation of the current theory for the additive Schwarz method. Then, in section 3, we present algebraic variants to the P.L. Lions' domain decomposition method. In § 4, we build a coarse space so that the two-level SORAS method achieves a targeted condition number. In § 5, the method is applied to saddle point problems.

2. Short introduction to ASM theory. In order to appraise the theory developed in § 3, we first give a short presentation of the current theory for two-level additive Schwarz methods. The starting point was the original Schwarz algorithm [41] for proving the well-posedness of the Poisson problem $-\Delta u = f$ with Dirichlet boundary conditions in some domain Ω decomposed into two subdomains Ω_1 and Ω_2, $\Omega = \Omega_1 \cup \Omega_2$.

DEFINITION 2.1 (Original Schwarz algorithm). *The Schwarz algorithm is an iterative method based on solving alternatively sub-problems in domains Ω_1 and Ω_2. It updates $(u_1^n, u_2^n) \to (u_1^{n+1}, u_2^{n+1})$ by :*

(1)
$$\begin{array}{rcll} -\Delta(u_1^{n+1}) & = & f & \text{in } \Omega_1 \\ u_1^{n+1} & = & 0 & \text{on } \partial\Omega_1 \cap \partial\Omega \\ u_1^{n+1} & = & u_2^n & \text{on } \partial\Omega_1 \cap \overline{\Omega_2}. \end{array} \quad \text{then,} \quad \begin{array}{rcll} -\Delta(u_2^{n+1}) & = & f & \text{in } \Omega_2 \\ u_2^{n+1} & = & 0 & \text{on } \partial\Omega_2 \cap \partial\Omega \\ u_2^{n+1} & = & u_1^{n+1} & \text{on } \partial\Omega_2 \cap \overline{\Omega_1}. \end{array}$$

H. Schwarz proved the convergence of the algorithm and thus the well-posedness of the Poisson problem in complex geometries. A small modification of the algorithm [33] makes it suited to parallel architectures. Its convergence can be proved using the maximum principle [31].

DEFINITION 2.2 (Parallel Schwarz algorithm). *Iterative method which solves concurrently in all subdomains, $i = 1, 2$:*

$$\begin{array}{rcll} -\Delta(u_i^{n+1}) & = f & \text{in} & \Omega_i \\ u_i^{n+1} & = 0 & \text{on} & \partial\Omega_i \cap \partial\Omega \\ u_i^{n+1} & = u_{3-i}^n & \text{on} & \partial\Omega_i \cap \overline{\Omega}_{3-i}. \end{array} \tag{2}$$

The discretization of this algorithm yields a parallel algebraic method for solving the linear system $A\mathbf{U} = \mathbf{F} \in \mathbb{R}^{\#\mathcal{N}}$ ($\mathcal{N}$ is the set of degrees of freedom) arising from the discretization of the original Poisson problem set on domain Ω. Due to the duplication of the unknowns in the overlapping region $\Omega_1 \cap \Omega_2$, this direct discretization involves a matrix of size larger than that of matrix A, see e.g. [23] for more details. Actually, it is much simpler and as efficient to use the RAS preconditioner [6]

$$M_{RAS}^{-1} := \sum_{i=1}^{N} R_i^T D_i A_i^{-1} R_i \,, \tag{3}$$

where N is the number of subdomains, R_i for some $1 \le i \le N$ is the Boolean matrix that restricts a global vector to its degrees of freedom in subdomain Ω_i, matrix

$$A_i := R_i A_i R_i^T$$

is the Dirichlet matrix of subdomain Ω_i and D_i is a local diagonal matrix that yields an algebraic partition of unity on $\mathbb{R}^{\#\mathcal{N}}$:

$$I_d = \sum_{i=1}^{N} R_i^T D_i R_i \,. \tag{4}$$

Indeed, it is proved in [16] that the following fixed point algorithm

$$\mathbf{U}^{n+1} = \mathbf{U}^n + M_{RAS}^{-1}(\mathbf{F} - A\mathbf{U}^n) \tag{5}$$

yields iterates that are equivalent to that of the discretization of Algorithm (2). Note that our analysis is the same whether Di is Boolean or not.

The RAS preconditioner (3) is not symmetric. For M-matrices a general convergence result is given in [18]. In order to develop a general theory for it when used as preconditioner in a Krylov method, its symmetric variant, the ASM preconditioner :

$$M_{ASM}^{-1} = \sum_{i=1}^{N} R_i^T A_i^{-1} R_i \,, \tag{6}$$

was studied extensively, see [45] and references therein. Starting with the pioneering work [40], a lot of effort has been devoted to the design and analysis of two-level methods that are the key ingredient to scalable methods. In adaptive methods, the coarse space in the two-level method is built by solving local generalized eigenvalue problems [19, 14, 38, 42] . This way, it is possible to target a user defined condition number of the preconditioned system. Here we focus on the GenEO approach [42] where the coarse space is based on solving Generalized Eigenvalue problems for the set of degrees of freedom $\mathcal{N}_j$ of subdomain $1 \le j \le N$. Let A_j^{Neu} denote the matrix of the local Neumann problem, we have to find the eigenpairs $(V_{j,k}, \lambda_{j,k})_k$ such that : $V_{j,k} \in \mathbb{R}^{\mathcal{N}_j}$ and $\lambda_{j,k} \ge 0$:

$$D_j\, A_j D_j V_{j,k} = \; \lambda_{j,k}\, A_j^{Neu} V_{j,k} \tag{7}$$

By combining the eigenvectors corresponding to eigenvalues larger than some given threshold $\tau > 0$ into a coarse space, it is proved in [42, 12] that the eigenvalues of the hybrid Schwarz preconditioned system satisfy the following estimate

$$\frac{1}{1 + k_1\, \tau} \le \lambda(M_{HSM}^{-1}\, A) \le k_0 \,. \tag{8}$$

where k_0 is the maximum number of neighbors of a subdomain and k_1 is the maximum multiplicity of the intersections of subdomains.

To sum up, the current theory for the two-level Schwarz method is based on the following four steps :

1. Schwarz algorithm at the continuous level (1)
2. An equivalent algebraic formulation (5) with the introduction of the RAS preconditioner (3)
3. A symmetrized variant named ASM (6) of the RAS preconditioner
4. A two-level method with an adaptive coarse space with prescribed targeted convergence rate .

3. Symmetrized ORAS method. Our goal here is to develop a theory and computational framework for P.L. Lions algorithm similar to what was done for the Schwarz algorithm for a symmetric positive definite (SPD) matrix A. We follow the steps recalled above.

First we introduce the P.L. Lions' algorithm which is based on improving Schwarz methods by replacing the Dirichlet interface conditions by Robin interface conditions. Let α be a positive number, the modified algorithm reads

$$\begin{array}{rcll} -\Delta(u_1^{n+1}) & = & f & \text{in} \quad \Omega_1, \\ u_1^{n+1} & = & 0 & \text{on} \quad \partial\Omega_1 \cap \partial\Omega, \\ \left(\dfrac{\partial}{\partial \mathbf{n}_1} + \alpha\right)(u_1^{n+1}) & = & \left(\dfrac{\partial}{\partial \mathbf{n}_1} + \alpha\right)(u_2^n) & \text{on} \quad \partial\Omega_1 \cap \overline{\Omega_2}\,, \end{array} \tag{9}$$

and

$$\begin{array}{rcll} -\Delta(u_2^{n+1}) & = & f & \text{in} \quad \Omega_2, \\ u_2^{n+1} & = & 0 & \text{on} \quad \partial\Omega_2 \cap \partial\Omega \\ \left(\dfrac{\partial}{\partial \mathbf{n}_2} + \alpha\right)(u_2^{n+1}) & = & \left(\dfrac{\partial}{\partial \mathbf{n}_2} + \alpha\right)(u_1^n) & \text{on} \quad \partial\Omega_2 \cap \overline{\Omega_1} \end{array} \tag{10}$$

where $\mathbf{n}_1$ and $\mathbf{n}_2$ are the outward normals on the boundary of the subdomains.

The second step is an algebraic equivalent formulation of the P.L. Lions algorithm in the case of overlapping subdomains. It is based on the introduction of the ORAS (Optimized Restricted Additive Schwarz) [44] preconditioner :

$$M_{ORAS}^{-1} := \sum_{i=1}^{N} R_i^T D_i B_i^{-1} R_i\,, \tag{11}$$

where $(B_i)_{1\le i\le N}$ is the discretization matrix of the Robin problem in subdomain Ω_i. The following fixed point method

$$\mathbf{U}^{n+1} = \mathbf{U}^n + M_{ORAS}^{-1}(\mathbf{F} - A\mathbf{U}^n) \tag{12}$$

yields iterates that are equivalent to that of the discretization of P.L. Lions' Algorithm (9)-(10), see [44].

The third step is the introduction of a symmetric variant that allows for a comprehensive theoretical study. It seems at first glance that we should mimic what was done for the RAS algorithm and study the following symmetrized variant :

$$M_{OAS,1}^{-1} := \sum_{i=1}^{N} R_i^T B_i^{-1} R_i\,. \tag{13}$$

For reasons explained in Remark 1, we introduce another non standard variant of the ORAS preconditioner (11), the symmetrized ORAS (SORAS) algorithm :

$$M_{SORAS,1}^{-1} := \sum_{i=1}^{N} R_i^T D_i B_i^{-1} D_i R_i\,. \tag{14}$$

This variant is reminiscent of the RASH [6] algorithm. Note also that the symmetric variant of ORAS is not equivalent to Lions' algorithm exactly as neither ASM nor RASH are equivalent to RAS which is itself equivalent the Schwarz algorithm, see [15].

The missing step is the fourth one, namely to build an adaptive coarse space for a two-level SORAS method. it is done in the next section.

4. Two-level SORAS algorithm.

Before designing and analyzing the two-level SORAS method, we precise our mathematical framework.

4.1. Mathematical framework.

The problem to be solved is defined via a variational formulation on a domain $\Omega \subset \mathbb{R}^d$ for $d \in \mathbb{N}$:

$$\text{Find } u \in V \text{ such that : } a_\Omega(u, v) = l(v)\,, \quad \forall v \in V\,,$$

where V is a Hilbert space of functions from Ω with real values. The problem we consider is given through a symmetric positive definite bilinear form a_Ω that is defined in terms of an integral over any open set $\omega \subset \Omega$. Typical examples are the Darcy equation ($\mathbf{K}$ is a diffusion tensor)

$$a_\omega(u, v) := \int_\omega \mathbf{K}\,\nabla u \cdot \nabla v\, dx\,,$$

or the elasticity system ($\boldsymbol{C}$ is the fourth-order stiffness tensor and $\boldsymbol{\varepsilon}(\boldsymbol{u})$ is the strain tensor of a displacement field $\boldsymbol{u}$) :

$$a_\omega(\boldsymbol{u}, \boldsymbol{v}) := \int_\omega \boldsymbol{C} : \boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{v})\, dx\,.$$

The problem is discretized by a finite element method. Let $\mathcal{N}$ denote the set of degrees of freedom and $(\phi_k)_{k\in\mathcal{N}}$ be a finite element basis on a mesh $\mathcal{T}_h$. Let $A \in \mathbb{R}^{\#\mathcal{N}\times\#\mathcal{N}}$ be the associated finite element matrix, $A_{kl} := a_\Omega(\phi_l, \phi_k)$, $k, l \in \mathcal{N}$. For some given right hand side $\mathbf{F} \in \mathbb{R}^{\#\mathcal{N}}$, we have to solve a linear system in $\mathbf{U}$ of the form

$$A\mathbf{U} = \mathbf{F}\,.$$

Domain Ω is decomposed into N overlapping subdomains $(\Omega_i)_{1\le i\le N}$ so that all subdomains are a union of cells of the mesh $\mathcal{T}_h$. This decomposition induces a natural decomposition of the set of indices $\mathcal{N}$ into N subsets of indices $(\mathcal{N}_i)_{1\le i\le N}$:

$$\mathcal{N}_i := \{k \in \mathcal{N} \mid meas(\text{supp}(\phi_k) \cap \Omega_i) > 0\}\,,\ 1 \le i \le N. \tag{15}$$

For all $1 \le i \le N$, let R_i be the restriction matrix from $\mathbb{R}^{\#\mathcal{N}}$ to the subset $R^{\#\mathcal{N}_i}$ and D_i be a diagonal matrix of size $\#\mathcal{N}_i \times \#\mathcal{N}_i$, so that we have a partition of unity at the algebraic level,

$$\sum_{i=1}^{N} R_i^T D_i R_i = I_d\,, \tag{16}$$

where $I_d \in \mathbb{R}^{\#\mathcal{N}\times\#\mathcal{N}}$ is the identity matrix.
For all subdomains $1 \le i \le N$, let B_i be a SPD matrix of size $\#\mathcal{N}_i \times \#\mathcal{N}_i$, which comes typically from the discretization of boundary value local problems using optimized transmission conditions.

We also define for all subdomains $1 \le j \le N$, $\widetilde{A}^j$, the $\#\mathcal{N}_j \times \#\mathcal{N}_j$ matrix defined by

$$\mathbf{V}_j^T \widetilde{A}^j \mathbf{U}_j := a_{\Omega_j}\left(\sum_{l\in\mathcal{N}_j} \mathbf{U}_{jl}\phi_l,\, \sum_{l\in\mathcal{N}_j} \mathbf{V}_{jl}\phi_l\right)\,,\quad \mathbf{U}_j,\, \mathbf{V}_j \in \mathbb{R}^{\mathcal{N}_j}\,. \tag{17}$$

When the bilinear form a results from the variational solve of a Laplace problem, the previous matrix corresponds to the discretization of local Neumann boundary value problems. For this reason we will call it "Neumann" matrix even in a more general setting.

We also make use of two numbers k_0 and k_1 related to the domain decomposition. Let

$$k_0 := \max_{1 \le i \le N} \# \left\{ j \mid R_j A R_i^T \neq 0 \right\} \tag{18}$$

be the maximum multiplicity of the interaction between subdomains plus one. Let k_1 be the maximal multiplicity of subdomains intersection, i.e. the largest integer m such that there exists m different subdomains whose intersection has a non zero measure.

4.2. SORAS with GenEO-2. We now consider a two-level method based on enriching the one-level SORAS preconditioner (11) by introducing two generalized eigenvalue problems which allow us to control the spectrum of the preconditioned operator as written in Theorem 4.10.

4.2.1. Coarse Space for the lower bound. More precisely, we define the following generalized eigenvalue problem :

Definition 4.1 (Generalized Eigenvalue Problem for the lower bound). *For each subdomain $1 \le j \le N$, we introduce the generalized eigenvalue problem*

$$\begin{array}{c} \text{Find } (\mathbf{V}_{jk}, \lambda_{jk}) \in \mathbb{R}^{\#\mathcal{N}_j} \setminus \{0\} \times \mathbb{R} \text{ such that} \\ \widetilde{A}^j \mathbf{V}_{jk} = \lambda_{jk} B_j \mathbf{V}_{jk} \ . \end{array} \tag{19}$$

Let $\tau > 0$ be a user-defined threshold, we define $Z_{geneo}^{\tau} \subset \mathbb{R}^{\#\mathcal{N}}$ as the vector space spanned by the family of vectors $(R_j^T D_j \mathbf{V}_{jk})_{\lambda_{jk} < \tau, 1 \le j \le N}$ corresponding to eigenvalues smaller than τ.

Let $\tilde{\pi}_j$ be the projection from $\mathbb{R}^{\#\mathcal{N}_j}$ on $\text{Span}\{\mathbf{V}_{jk} | \lambda_{jk} < \tau\}$ parallel to $\text{Span}\{\mathbf{V}_{jk} | \lambda_{jk} \ge \tau\}$. In the present case of the SORAS-2 method, Lemma 7.6, page 167 in [12] translates into :

Lemma 4.2 (Intermediate Lemma for GenEO-SORAS-2). *For all subdomains $1 \le j \le N$ and $\mathbf{U}_j \in \mathbb{R}^{\mathcal{N}_j}$, we have :*

$$\tau \, ((I_d - \tilde{\pi}_j)\mathbf{U}_j)^T B_j (I_d - \tilde{\pi}_j)\mathbf{U}_j \le \ \mathbf{U}_j^T \widetilde{A}^j \mathbf{U}_j \, . \tag{20}$$

where by abuse of notation $I_d \in \mathbb{R}^{\#\mathcal{N}_j \times \#\mathcal{N}_j}$ is the identity matrix on $\mathbb{R}^{\mathcal{N}_j}$.

4.2.2. Coarse space for the upper bound. We introduce the following generalized eigenvalue problem :

Definition 4.3 (Generalized Eigenvalue Problem for the upper bound).

$$\begin{array}{c} \text{Find } (\mathbf{U}_{ik}, \mu_{ik}) \in \mathbb{R}^{\#\mathcal{N}_i} \setminus \{0\} \times \mathbb{R} \text{ such that} \\ D_i R_i A R_i^T D_i \mathbf{U}_{ik} = \mu_{ik} B_i \, \mathbf{U}_{ik} \ . \end{array} \tag{21}$$

Let $\gamma > 0$ be a user-defined threshold, we define $Z_{geneo}^{\gamma} \subset \mathbb{R}^{\#\mathcal{N}}$ as the vector space spanned by the family of vectors $(R_i^T D_i \mathbf{U}_{ik})_{\mu_{ik} > \gamma, 1 \le i \le N}$ corresponding to eigenvalues larger than γ.

Now, let ξ_i denote the projection from $\mathbb{R}^{\#\mathcal{N}_i}$ on Span $\{\mathbf{U}_{ik} \,|\, \gamma > \mu_{ik}\}$ parallel to Span $\{\mathbf{U}_{ik} \,|\, \gamma \le \mu_{ik}\}$. From these definitions, Lemma 7.6, page 167 in [12] leads to :

LEMMA 4.4. *For all subdomains $1 \le i \le N$ and $\mathbf{U}_i \in \mathbb{R}^{\#\mathcal{N}_i}$, we have :*

$$\left(R_i^T D_i (I_d - \xi_i)\mathbf{U}_i\right)^T A\, R_i^T D_i (I_d - \xi_i)\mathbf{U}_i) \le \gamma\; \mathbf{U}_i^T B_i \mathbf{U}_i \,. \tag{22}$$

4.3. SORAS-GENEO-2 method. We are now ready to define the SORAS two level preconditioner

DEFINITION 4.5 (Two level SORAS-GENEO-2 preconditioner). *Let P_0 denote the a-orthogonal projection on the SORAS-GENEO-2 coarse space*

$$Z_{GenEO\text{-}2} := Z^{\tau}_{geneo} \bigoplus Z^{\gamma}_{geneo} \,,$$

the two-level SORAS-GENEO-2 preconditioner is defined as follows, see [36] :

$$\boxed{M^{-1}_{SORAS,2} := P_0 A^{-1} + (I_d - P_0) \sum_{i=1}^{N} R_i^T D_i B_i^{-1} D_i R_i (I_d - P_0^T)\,.} \tag{23}$$

Let Z_0 be a matrix whose columns are a basis of $Z_{\text{GenEO-2}}$ and let denote its transpose by $R_0 := Z_0^T$. It is easily checked that

$$P_0 A^{-1} = R_0^T (R_0 A R_0^T)^{-1} R_0 \,.$$

This definition is reminiscent of the balancing domain decomposition preconditioner [36] introduced for Schur complement based methods. Note that the coarse space is now defined by two generalized eigenvalue problems instead of one in [42, 43] for ASM and FETI-BDD methods.

The proof of Theorem 4.10 is based on the Fictitious Space [39] Lemma 7.4 in [12] , page 164.

DEFINITION 4.6 (Two-level SORAS in the Fictitious Space Lemma). *Two Hilbert spaces H and H_D, two other associated bilinear forms and induced scalar products as well as the $\mathcal{R}_{SORAS,2}$ operator between them are defined as follows.*

- *Space $H := \mathbb{R}^{\#\mathcal{N}}$ endowed with the standard Euclidean scalar product. We consider another bilinear form a defined by :*

$$a : H \times H \to \mathbb{R}, \quad (\mathbf{U}, \mathbf{V}) \longmapsto a(\mathbf{U}, \mathbf{V}) := \mathbf{V}^T A \mathbf{U}. \tag{24}$$

  where A is the matrix of the problem we want to solve.
- *Space H_D, defined as the product space*

$$H_D := \mathbb{R}^{\#\mathcal{N}_0} \times \prod_{i=1}^{N} \mathbb{R}^{\#\mathcal{N}_i} \tag{25}$$

  is endowed with standard scalar Euclidean product. For $\mathcal{U} = (\mathbf{U}_i)_{1\le i\le N}$, $\mathcal{V} = (\mathbf{V}_i)_{1\le i\le N}$ with $\mathbf{U}_i, \mathbf{V}_i \in \mathbb{R}^{\#\mathcal{N}_i}$, the bilinear form b is defined by

$$\begin{aligned} b : H_D \times H_D \quad &\longrightarrow \quad \mathbb{R} \\ (\mathcal{U}, \mathcal{V}) \quad &\longmapsto \quad b(\mathcal{U}, \mathcal{V}) := (R_0^T \mathbf{V}_0)^T A\, (R_0^T \mathbf{U}_0) + \sum_{i=1}^{N} \mathbf{V}_i^T B_i \mathbf{U}_i, \end{aligned} \tag{26}$$

Let B denote the block-diagonal operator such that for all $\mathcal{U},\mathcal{V} \in H_D$, we have :

$$(B\mathcal{U},\mathcal{V}) = b(\mathcal{U},\mathcal{V})$$

— *For any $\mathcal{U} = (\mathbf{U}_i)_{0\le i\le N}$ the linear operator $\mathcal{R}_{SORAS,2}$ is defined as*

$$\mathcal{R}_{SORAS,2} : H_D \longrightarrow H,\ \mathcal{R}_{SORAS,2}(\mathcal{U}) := R_0^T\,\mathbf{U}_0 + \sum_{i=1}^N (I_d - P_0)\,R_i^T D_i\,\mathbf{U}_i. \tag{27}$$

It can easily be checked that

$$M^{-1}_{SORAS,2} = \mathcal{R}_{SORAS,2} B^{-1} \mathcal{R}^T_{SORAS,2}\,.$$

We now check the assumptions of the Fictitious Space Lemma.

LEMMA 4.7 (Surjectivity of $\mathcal{R}_{SORAS,2}$). *Operator $\mathcal{R}_{SORAS,2}$ is surjective.*

Proof. For all $\mathbf{U} \in H$, we have :

$$\mathbf{U} = P_0\,\mathbf{U} + (I_d - P_0)\,\mathbf{U} = P_0\,\mathbf{U} + \sum_{i=1}^N (I_d - P_0)\,R_i^T D_i R_i \mathbf{U}\,.$$

Since $P_0\,\mathbf{U} \in \mathrm{Span}(R_0^T)$, there exists $\mathbf{U}_0 \in R^{\#\mathcal{N}_0}$ such that $P_0\,\mathbf{U} = R_0^T\,\mathbf{U}_0$. Thus, we have

$$\mathbf{U} = R_0^T\,\mathbf{U}_0 + \sum_{i=1}^N (I_d - P_0)\,R_i^T D_i (R_i \mathbf{U})\,,$$

or, in other words

$$\mathcal{R}_{SORAS,2}(\mathbf{U}_0, (R_i\mathbf{U})_{1\le i\le N}) = \mathbf{U}\,,$$

which proves the surjectivity. □

We now prove

LEMMA 4.8 (Continuity of $\mathcal{R}_{SORAS,2}$). *Let $\mathcal{U} = (\mathbf{U}_i)_{0\le i\le N} \in H_D$. We have the following continuity estimate*

$$a(\mathcal{R}_{SORAS,2}(\mathcal{U}), \mathcal{R}_{SORAS,2}(\mathcal{U})) \le \max(1, k_0\,\gamma)\,b(\mathcal{U},\mathcal{U})\,.$$

Proof. Since P_0 and $I_d - P_0$ are a-orthogonal projections, we have by a-orthogonality :

$$\begin{aligned} a(\mathcal{R}_{SORAS,2}(\mathcal{U}), \mathcal{R}_{SORAS,2}(\mathcal{U})) \quad = \quad & a\left(P_0\,R_0^T\,\mathbf{U}_0, P_0\,R_0^T\,\mathbf{U}_0\right) \\ & + a\left((I_d - P_0)\sum_{i=1}^N R_i^T D_i\,\mathbf{U}_i, (I_d - P_0)\sum_{i=1}^N R_i^T D_i\,\mathbf{U}_i\right) \end{aligned}$$

Since P_0 is the a-orthogonal projection on $Z_{\text{GenEO-2}}$ and that

$$\sum_{i=1}^N R_i^T D_i \xi_i \mathbf{U}_i \in Z^{\gamma}_{geneo} \subset Z_{\text{GenEO-2}}\,,$$

we have

$$(I_d - P_0)\sum_{i=1}^N R_i^T D_i \xi_i \mathbf{U}_i = 0\,,$$

and thus

$$
\begin{aligned}
& a\left((I_d - P_0)\sum_{i=1}^N R_i^T D_i\, \mathbf{U}_i, (I_d - P_0)\sum_{i=1}^N R_i^T D_i\, \mathbf{U}_i\right) \\
& \quad = a\left((I_d - P_0)\sum_{i=1}^N R_i^T D_i\,(I_d - \xi_i)\mathbf{U}_i, (I_d - P_0)\sum_{i=1}^N R_i^T D_i\,(I_d - \xi_i)\mathbf{U}_i\right).
\end{aligned}
$$

Finally, using k_0 defined as in in Lemma 7.11, page 174 in [12] , we have

$$
\begin{aligned}
a(\mathcal{R}_{SORAS,2}(\mathcal{U}), \mathcal{R}_{SORAS,2}(\mathcal{U})) \quad & \leq \quad a\left(R_0^T\,\mathbf{U}_0,\, R_0^T\,\mathbf{U}_0\right) \\
& \quad\; + a\left(\sum_{i=1}^N R_i^T D_i\,(I_d - \xi_i)\mathbf{U}_i, \sum_{i=1}^N R_i^T D_i\,(I_d - \xi_i)\mathbf{U}_i\right) \\
& \leq \quad a\left(R_0^T\,\mathbf{U}_0,\, R_0^T\,\mathbf{U}_0\right) \\
& \quad\; + k_0 \sum_{i=1}^N a\left(R_i^T D_i\,(I_d - \xi_i)\mathbf{U}_i, R_i^T D_i\,(I_d - \xi_i)\mathbf{U}_i\right).
\end{aligned}
$$

Then, using estimate (22), we have :

$$
\begin{aligned}
a(\mathcal{R}_{SORAS,2}(\mathcal{U}), \mathcal{R}_{SORAS,2}(\mathcal{U})) \quad & \leq \quad a(\,R_0^T\,\mathbf{U}_0,\, R_0^T\,\mathbf{U}_0) + k_0\,\gamma \sum_{i=1}^N (B_i\,\mathbf{U}_i,\,\mathbf{U}_i) \\
& \leq \quad \max(1, k_0\,\gamma)\, b(\mathcal{U}, \mathcal{U}).
\end{aligned}
$$

which concludes the estimate of the continuity of $\mathcal{R}_{SORAS,2}$. ∎

LEMMA 4.9 (Stable decomposition with $\mathcal{R}_{SORAS,2}$). *Let $\mathbf{U}$ be a vector in H. We define :*

$$
\mathbf{U}_j := (I_d - \widetilde{\pi}_j)\, R_j\, \mathbf{U}
$$

and $\mathbf{U}_0 \in \mathbb{R}^{\#\mathcal{N}_0}$ such that :

$$
R_0^T\,\mathbf{U}_0 = P_0\,\mathbf{U}.
$$

We define $\mathcal{U} := (\mathbf{U}_i)_{0 \leq i \leq N}$.
Then, the stable decomposition property is verified with a constant $(1 + k_1\,\tau^{-1})^{-1}$, since we have :

$$
\mathcal{R}_{SORAS,2}(\mathcal{U}) = \mathbf{U},
$$

$$
\frac{1}{(1 + k_1\,\tau^{-1})} b(\mathcal{U}, \mathcal{U}) \leq\, a(\mathbf{U}, \mathbf{U}).
$$

Proof. We first check that we have indeed a decomposition $\mathcal{R}_{SORAS,2}(\mathcal{U}) = \mathbf{U}$. Note that for all $1 \leq j \leq N$ we have

$$
R_j^T\, D_j\, \widetilde{\pi}_j\, R_j\, \mathbf{U} \in Z_{geneo}^{\tau} \subset Z_{\text{GenEO-2}} \Rightarrow (I_d - P_0) R_j^T\, D_j\, \widetilde{\pi}_j\, R_j\, \mathbf{U} = 0\,.
$$

We have :

$$
\begin{aligned}
\mathbf{U} \quad & = \quad P_0\mathbf{U} + (I_d - P_0)\mathbf{U} = P_0\mathbf{U} + (I_d - P_0)\sum_{j=1}^N R_j^T D_j\, R_j\, \mathbf{U} \\
& = \quad P_0 R_0^T\, \mathbf{U}_0 + (I_d - P_0)\sum_{j=1}^N R_j^T D_j\,(I_d - \widetilde{\pi}_j)\, R_j\, \mathbf{U} = \mathcal{R}_{SORAS,2}(\mathcal{U}).
\end{aligned}
$$

The last thing to do is to check the stability of this decomposition. Using (20) and then Lemma 7.13, page 175 in [12] . , we have

$$\begin{aligned} b(\mathcal{U},\mathcal{U}) &= a(R_0^T\mathbf{U}_0, R_0^T\mathbf{U}_0) \\ &\quad + \sum_{j=1}^{N} \left((I_d - \widetilde{\pi}_j)\, R_j\, \mathbf{U} \right)^T B_j \left((I_d - \widetilde{\pi}_j)\, R_j\, \mathbf{U} \right) \\ &\le a(P_0\mathbf{U}, P_0\mathbf{U}) + \tau^{-1} \sum_{j=1}^{N} (\widetilde{R}_j \mathbf{U})^T \widetilde{A}^j (R_j \mathbf{U}) \\ &\le a(\mathbf{U},\mathbf{U}) + k_1\, \tau^{-1} a(\mathbf{U},\mathbf{U}) \le (1 + k_1\, \tau^{-1})\, a(\mathbf{U},\mathbf{U}). \end{aligned}$$ ∎

The assumptions of the Fictitious Space Lemma are verified and thus we have just proved the following

THEOREM 4.10 (Spectral estimate for the two level SORAS-GenEO-2). *Let γ be a chosen threshold in Definition 4.3, τ be a chosen threshold in Definition (4.1) of the GenEO-2 coarse space and the two-level SORAS-GenEO-2 preconditioner defined by* (23). *Then, the eigenvalues of the two-level SORAS-GenEO-2 preconditioned system satisfy the following estimate*

$$\boxed{\frac{1}{1 + \frac{k_1}{\tau}} \le \lambda(M^{-1}_{SORAS,2}\, A) \le \max(1, k_0\, \gamma)}$$

We have the

Remark 1. *An analysis of a two-level version of the preconditioner M^{-1}_{OAS}* (13) *following the same path yields the following two generalized eigenvalue problems :*

$$\begin{array}{c} \text{Find } (\mathbf{U}_{jk}, \mu_{jk}) \in \mathbb{R}^{\#\mathcal{N}_i} \setminus \{0\} \times \mathbb{R} \text{ such that} \\ A^i \mathbf{U}_{ik} = \mu_{ik} B_i \mathbf{U}_{ik}\ , \end{array}$$

and

$$\begin{array}{c} \text{Find } (\mathbf{V}_{jk}, \lambda_{jk}) \in \mathbb{R}^{\#\mathcal{N}_i} \setminus \{0\} \times \mathbb{R} \text{ such that} \\ \widetilde{A}^i \mathbf{V}_{ik} = \lambda_{ik} D_i B_i D_i \mathbf{V}_{ik}\ . \end{array} .$$

In the general case for $1 \le i \le N$, matrices D_i may have zero entries for boundary degrees of freedom since they are related to a partition of unity. Moreover very often matrices B_i and A_i differ only by the interface conditions that is for entries corresponding to boundary degrees of freedom. Therefore, matrix $D_i B_i D_i$ on the right hand side of the last generalized eigenvalue problem is not impacted by the choice of the interface conditions of the one level optimized Schwarz method. This cannot lead to efficient adaptive coarse spaces.

5. Saddle point problems.

Many applications in science and engineering require solving large linear algebraic systems in saddle point form ; see [3] for an extensive survey. Although our theory does not apply in a straightforward manner to saddle point problems, we use it for these difficult problems for which it is not always possible to preserve both symmetry and positivity of the problem, see [30]. Note that generalized eigenvalue problems (21) and (19) still make sense if A is the matrix of a saddle point problem and local matrices A_i, B_i and $\widetilde{A}_i$, $1 \le i \le N$, are based on a partition of unity and on variational formulations.

We start by the global problem defined via variational formulation see for instance § 6.1 for the systems of almost incompressible elasticity and of Stokes. As in § 4.1, these formulations are written in terms of integrals of differential quantities (gradient, divergence, ...) over some domain $\Omega \subset \mathbb{R}^d$ for $d \in \mathbb{N}$:
Find $(\boldsymbol{u}, p) \in V \times \Lambda$ such that :

$$\begin{aligned} a_\Omega(\boldsymbol{u},\, \boldsymbol{v}) + b_\Omega(\boldsymbol{v},\, p) &= l_1(\boldsymbol{v})\,, \quad \forall \boldsymbol{v} \in V\,, \\ b_\Omega(\boldsymbol{u},\, q) - c_\Omega(p,\, q) &= l_2(q)\,, \quad \forall q \in \Lambda\,, \end{aligned}$$

where V and Λ are Hilbert spaces of functions from Ω with real values, a_Ω, b_Ω and c_Ω are bilinear forms, a_Ω and b_Ω being symmetric. Discretization by a finite element method yields a saddle point system of the form :

$$A := \begin{bmatrix} H & B^T \\ B & -C \end{bmatrix} \begin{bmatrix} \boldsymbol{u} \\ p \end{bmatrix} = \begin{bmatrix} \boldsymbol{f} \\ g \end{bmatrix}, \tag{28}$$

where $H = H^T$ is positive definite, $C = C^T$ is positive semidefinite. The set of degrees of freedom is decomposed into subsets $(\mathcal{N}_i)_{1 \le i \le N}$. The matrices involved in the partition of unity (16) have a block diagonal form

$$D_i := \begin{bmatrix} D_i^u & 0 \\ 0 & D_i^p \end{bmatrix} \text{ and } R_i := \begin{bmatrix} R_i^u & 0 \\ 0 & R_i^p \end{bmatrix}.$$

The local "Dirichlet" matrices have the following block form :

$$A_i := R_i A R_i^T = \begin{bmatrix} H_i & B_i^T \\ B_i & -C_i \end{bmatrix}$$

where

$$H_i := R_i^u H R_i^{u\,T}\,, \quad C_i := R_i^p C R_i^{p\,T} \text{ and } B_i := R_i^p B R_i^{u\,T}\,.$$

The local "Neumann" problems arise from the variational formulation restricted the finite element space of a subdomain as in (17). We use the following block notation

$$\widetilde{A}_i := \begin{bmatrix} \widetilde{H}_i & \widetilde{B}_i^T \\ \widetilde{B}_i & -\widetilde{C}_i \end{bmatrix}.$$

For each subdomain $1 \le i \le N$, the "Robin" matrix is

$$B_i = \widetilde{A}_i + Z_i$$

where $Z_i = Z_i^T$ is positive semidefinite and is such that matrix B_i is symmetric positive definite. For sake of simplicity the "Robin" boundary condition will only apply to the $\boldsymbol{u}$ term, that is :

$$Z_i = \begin{bmatrix} Z_i^u & 0 \\ 0 & 0 \end{bmatrix}.$$

5.1. GenEO eigenvalue problem for saddle point problems. Eigenvalue problem for saddle point problem has been considered by various authors, see [4] and references therein. We cannot use directly their results since we consider generalized eigenvalue problems where both left and right matrices have saddle point structures. In order to prove that the GenEO eigenvalues are real and non negative, we need the following assumption :

ASSUMPTION 1.

$$(\widetilde{H}_i \boldsymbol{u},\, \boldsymbol{u}) + (Z_i^u \boldsymbol{u},\, \boldsymbol{u}) + (\widetilde{C}_i p, p) = 0 \Rightarrow \boldsymbol{u} = 0 \text{ and } p = 0. \tag{29}$$

This assumption is satisfied for the two applications we consider below in § 6. For instance, in the case of nearly incompressible elasticity, matrix $\widetilde{C}_i$ is the mass matrix of subdomain Ω_i weighted by the inverse of the first Lamé coefficient (λ) which is SPD. As for $\widetilde{H}_i + Z_i^u$ it is the sum of the stiffness matrix of subdomain Ω_i and of a positive boundary term on the interface. This matrix is thus SPD as well.

Consider the generalized eigenvalue problem that controls the lower bound of the spectrum of the preconditioned system :

$$\begin{bmatrix} \widetilde{H}_i & \widetilde{B}_i^T \\ \widetilde{B}_i & -\widetilde{C}_i \end{bmatrix} \begin{bmatrix} \boldsymbol{u} \\ p \end{bmatrix} = \lambda \begin{bmatrix} \widetilde{H}_i + Z_i^u & \widetilde{B}_i^T \\ \widetilde{B}_i & -\widetilde{C}_i \end{bmatrix} \begin{bmatrix} \boldsymbol{u} \\ p \end{bmatrix}. \tag{30}$$

By Assumption 1, it is clear that the matrix in the right part of the equality is invertible. In order to prove it, it suffices to take an element $\begin{bmatrix} \boldsymbol{u} \\ p \end{bmatrix}$ in the kernel and take the scalar product with $\begin{bmatrix} \boldsymbol{u} & -p \end{bmatrix}^T$ to prove that the kernel reduces to the null vector. Thus, left multiplying problem (30) by the inverse of this matrix reduces it to a standard eigenvalue problem.

We now take the scalar product of (30) with $\begin{bmatrix} \boldsymbol{u} & -p \end{bmatrix}^T$. The cross product terms $(\widetilde{B}_i^T \boldsymbol{u},\, p)$ cancel and we get :

$$(\widetilde{H}_i \boldsymbol{u},\, \boldsymbol{u}) + (\widetilde{C}_i p,\, p) = \lambda \left[(\widetilde{H}_i \boldsymbol{u},\, \boldsymbol{u}) + (Z_i^u \boldsymbol{u},\, \boldsymbol{u}) + (\widetilde{C}_i p, p) \right]. \tag{31}$$

All terms above are non negative. From Assumption 1, the right term cannot be zero. Therefore, $\lambda \in [0, 1]$.

Consider now the eigenvalue problem that controls the upper bound of the spectrum of the preconditioned system :

$$\begin{bmatrix} D_i^u H_i D_i^u & D_i^u B_i^T D_i^p \\ D_i^p B_i D_i^u & -D_i^p C_i\, D_i^p \end{bmatrix} \begin{bmatrix} \boldsymbol{u} \\ p \end{bmatrix} = \mu \begin{bmatrix} \widetilde{H}_i + Z_i^u & \widetilde{B}_i^T \\ \widetilde{B}_i & -\widetilde{C}_i \end{bmatrix} \begin{bmatrix} \boldsymbol{u} \\ p \end{bmatrix}. \tag{32}$$

We take the scalar product of (32) with $\begin{bmatrix} \boldsymbol{u} & -p \end{bmatrix}^T$ and we get :

$$(H_i D_i^u \boldsymbol{u},\, D_i^u \boldsymbol{u}) + (C_i D_i^p p,\, D_i^p p) = \mu \left[(\widetilde{H}_i \boldsymbol{u},\, \boldsymbol{u}) + (Z_i^u \boldsymbol{u},\, \boldsymbol{u}) + (\widetilde{C}_i p, p) \right] \tag{33}$$

All terms above are non negative. From Assumption 1, the right term cannot be zero. Therefore, $\mu \geq 0$.

6. Application to the systems of Stokes and of Nearly Incompressible elasticity. Mixed finite elements are often used to solve incompressible Stokes and nearly incompressible elasticity problems. Continuous pressures have been used in many mixed finite elements. However, most domain decomposition methods require that the pressure be discontinuous when they are used to solve the indefinite linear systems arising from such mixed finite element discretizations. Several domain decomposition algorithms allow one to use continuous pressures, see [46] and references therein. To our knowledge, our method is the first one to exhibit scalability for a highly heterogeneous nearly incompressible elasticity problems with continuous pressures.

6.1. Variational formulations. The mechanical properties of a solid can be characterized by its Young modulus E and Poisson ratio ν or alternatively by its Lamé coefficients λ and μ. These coefficients relate to each other by the following formulas :

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)} \text{ and } \mu = \frac{E}{2(1+\nu)}. \tag{34}$$

The variational problem consists in finding $(\boldsymbol{u}_h, p_h) \in \mathcal{V}_h := \mathbb{P}_2^d \cap H_0^1(\Omega) \times \mathbb{P}_1$ such that for all $(\boldsymbol{v}_h, q_h) \in \mathcal{V}_h$

$$\begin{cases} \int_\Omega 2\mu \underline{\underline{\varepsilon}}(\boldsymbol{u}_h) : \underline{\underline{\varepsilon}}(\boldsymbol{v}_h) dx & - \int_\Omega p_h \operatorname{div}(\boldsymbol{v}_h) dx = \int_\Omega \boldsymbol{f} \boldsymbol{v}_h dx \\ - \int_\Omega \operatorname{div}(\boldsymbol{u}_h) q_h dx & - \int_\Omega \frac{1}{\lambda} p_h q_h = 0 \end{cases} \tag{35}$$

Let $\boldsymbol{u}$ denote the degrees of freedom of $\boldsymbol{u}_h$ and p that of p_h, they satisfy a linear system denoted as follows :

$$A\mathbf{U} = \begin{bmatrix} H & B^T \\ B & -C \end{bmatrix} \begin{bmatrix} \boldsymbol{u} \\ p \end{bmatrix} = \begin{bmatrix} \mathrm{f} \\ 0 \end{bmatrix} = \mathbf{F}. \tag{36}$$

Matrix $\widetilde{A}_i$ arises from the variational formulation (35) where the integration over domain Ω is replaced by the integration over subdomain Ω_i and finite element space $\mathcal{V}_h$ is restricted to subdomain Ω_i. Matrix B_i corresponds to a Robin problem and is the sum of matrix $\widetilde{A}_i$ and of the matrix of the following variational formulation restricted to the same finite element space :

$$\int_{\partial\Omega_i \setminus \partial\Omega} \frac{2\alpha\mu(2\mu+\lambda)}{\lambda+3\mu} \boldsymbol{u}_h \cdot \boldsymbol{v}_h \text{ with } \alpha = 10 \text{ in our test.} \tag{37}$$

In the next section, we explain the origin of the term (37).

6.2. Interface conditions. We touch here another peculiarity of the P.L. Lions algorithm. In some situations, it is possible to choose the interface condition in order to have convergence in a number of steps equal to the number of subdomains, see [37]. In our case, let the global domain Ω be the whole plane $\mathbb{R}^2$ decomposed into two half planes $\Omega_1 := (-\infty, \delta) \times \mathbb{R}$ and $\Omega_2 := (0, \infty) \times \mathbb{R}$ where $\delta \geq 0$ is the width of the overlap, k denote the Fourier transform in the y direction, the following interface condition yields to a convergence in two iterations :

$$\underline{\underline{\boldsymbol{\sigma}}} \cdot \underline{\boldsymbol{n}} + \mathcal{F}^{-1} \left(\begin{bmatrix} \frac{2|k|\mu(2\mu+\lambda)}{\lambda+3\mu} & \frac{2ik\mu^2}{\lambda+3\mu} \\ \frac{-2ik\mu^2}{\lambda+3\mu} & \frac{2|k|\mu(2\mu+\lambda)}{\lambda+3\mu} \end{bmatrix} \begin{bmatrix} \mathcal{F}(u_x) \\ \mathcal{F}(u_y) \end{bmatrix} \right) \tag{38}$$

where $\underline{\underline{\boldsymbol{\sigma}}} \cdot \underline{\boldsymbol{n}}$ is the normal component of the stress tensor, the velocity is decomposed into its normal u_x and tangential component u_y $\boldsymbol{u} = [u_x, u_y]^T$ and $\mathcal{F}$ denotes the Fourier transform in the y direction. Due to the absolute value $|k|$ this interface condition is non local in space and also difficult to apply in the general domain decompositions and has to be approximated, see [21]. For sake of simplicity, we drop the extra diagonal terms which correspond to tangential derivative in the physical space.

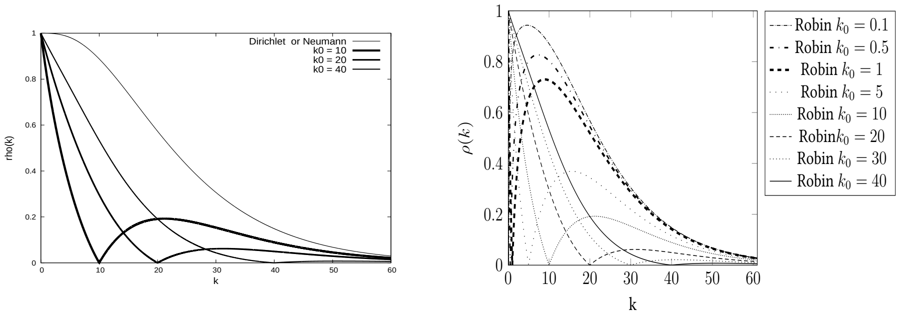

FIGURE 1. *Convergence rate vs Fourier number k for various interface conditions – Poisson ratio $\nu = 0.4999$ – overlap $\delta = 0.1$.*

As for the diagonal terms, we approximate them at some frequency k_0. Finally, the optimal interface condition (38) is approximated as follows :

$$\underline{\underline{\sigma}} \cdot \underline{n} + \mathcal{F}^{-1}\left(\begin{bmatrix} \frac{2|k_0|\mu(2\mu+\lambda)}{\lambda+3\mu} & 0 \\ 0 & \frac{2|k_0|\mu(2\mu+\lambda)}{\lambda+3\mu} \end{bmatrix} \begin{bmatrix} \mathcal{F}(u_x) \\ \mathcal{F}(u_y) \end{bmatrix}\right)$$

which simplifies in :

$$\underline{\underline{\sigma}} \cdot \underline{n} + |k_0| \frac{2\mu(2\mu+\lambda)}{\lambda+3\mu} \boldsymbol{u}\,. \tag{39}$$

This approximation has an impact on the convergence rate of the P.L. Lions' algorithm. Using similar arguments to that of [21] or [12] chapter 2, it is possible to derive a formula for the convergence rate as a function of the Fourier mode in the y direction. Since we have a system of partial differential equations, the formula is quite complex and was obtained with the help of Maple software. It can be found in [22]. On Figure 1, we plot the convergence as a function of the Fourier mode in the y direction for various values of the parameter k_0 for an almost incompressible elasticity system $\nu = 0.4999$ and an overlap of size $\delta = 0.1$. Although the Robin interface condition (39) is never exact even for $k = k_0$, the convergence rate is quite close to zero (of the order of 10^{-4}) for $k = k_0$. We see on Figure 1 that taking $k_0 = 10$ yields small convergence rate except for k very close to 1 and thus was chosen in our numerical tests below. Note that Dirichlet (cf. $k_0 \gg 1$) or Neumann (cf. $k_0 = 0$) (stress free) interface conditions yield the same bad convergence rates. For small Fourier numbers, the convergence rate is very close to 1 which is bad. Overall, Robin interface conditions perform much better than simple Dirichlet or Neumann interface conditions.

As for Stokes system, it can be seen as the limit as λ tends to infinity of the elasticity system. As a result, the interface condition we take reads :

$$\underline{\underline{\sigma}} \cdot \underline{n} + |k_0| 2\mu \boldsymbol{u}\,.$$

The interface condition (39) can be used for arbitrary domain decompositions since its variational formulation is the one of a stress free BVP to which we add the variational formulation of (37) where $\alpha := |k_0|$ for some chosen Fourier number k_0. Thus although the Fourier analysis has a limited domain of validity, the interface condition (39) can be used for arbitrary domain decompositions.

FIGURE 2. *2D Elasticity : coefficient distribution of steel and rubber.*

		AS	SORAS	AS+ZEM		SORAS +ZEM		AS-GenEO		SORAS GenEO2	
d.o.f.	N	iter	iter	iter	*dim*	iter	*dim*	iter	*dim*	iter	*dim*
35841	8	150	184	117	*24*	74	*24*	110	*184*	13	*145*
70590	16	276	337	170	*48*	136	*48*	153	*400*	17	*303*
141375	32	497	>1000	261	*96*	199	*96*	171	*800*	22	*561*
279561	64	>1000	>1000	333	*192*	329	*192*	496	*1600*	24	*855*
561531	128	>1000	>1000	329	*384*	325	*384*	>1000	*2304*	29	*1220*
1077141	256	>1000	>1000	330	*768*	321	*768*	>1000	*3840*	36	*1971*

TABLE 1
2D Elasticity. GMRES iteration counts for a solid made of steel and rubber.

6.3. Numerical results. The new coarse space was tested quite successfully on nearly incompressible elasticity and Stokes problems with a discretization based on saddle point formulations in order to avoid locking phenomena.

6.3.1. Tests against other algorithms. We first report 2D results for a heterogeneous beam of eight layers of steel $(E_1, \nu_1) = (210 \cdot 10^9, 0.3)$ and rubber $(E_2, \nu_2) = (0.1 \cdot 10^9, 0.4999)$, see Figure 2. The beam is clamped on its left and right sides. Simulations were made with FreeFem++ [24]. Iteration counts for various domain decomposition methods for a relative tolerance of 10^{-6} are given in Table 1. We compare the one level Additive Schwarz (AS) and SORAS methods, the two level AS and SORAS methods with a coarse space consisting of rigid body motions which are zero energy modes (ZEM) and finally AS with a GenEO coarse space as defined in [42] and SORAS with the GenEO-2 coarse space defined in Definition 4.1 with $\tau = 0.4$ and $\gamma = 10^3$. Columns *dim* refer to the total size of the coarse space of a two-level method. Eigenvalue problem (19) accounts for roughly 90% of the GenEO-2 coarse space size. We see that only the last method scales well with respect to the number of subdomains denoted by N. We also considered the dependence on the optimized interface condition. We found that for SORAS+ZEM, the method is very sensitive to the choice of k_0. Fortunately, SORAS+GenE02 yielded iteration counts that were very similar for k_0 ranging from 4 to 60.

6.3.2. 3D and 2D highly heterogeneous linear elasticity equations. Throughout this section we look at a linear elasticity problem with highly heterogeneous Lamé coefficients corresponding to steel and rubber materials. In the case of rubber which is nearly incompressible material the Poisson ratio ν approaches 1/2 and $\lambda/\mu = 2\nu/(1-2\nu)$ approaches infinity. In order to avoid the resulting locking phenomena with finite element discretization, the pure displacement problem is replaced by a mixed formulation as proposed in [5]. We performed a large 2D and 3D simulations, on an heterogeneous beam, where the Lamé (E, ν) vary discontinuously over the domain in eight alternating layers of steel material with $(E_1, \nu_1) = (210 \times 10^9, 0.3)$ and rubber material with $(E_2, \nu_2) = (0.1 \times 10^9, 0.4999)$ submitted to an external forces, see Figure 3. The system is discretized using a Taylor-Hood mixed finite element discretization which are inf-sup stable. P_3/P_2 for the 2D case and P_2/P_1 for the 3D case. The problem is solved with a minimal geometric overlap of one mesh element and a preconditioned GMRES is used to solve the resulting linear system where the stopping criteria for the relative residual norm is fixed to 10^{-6}. All the test cases were

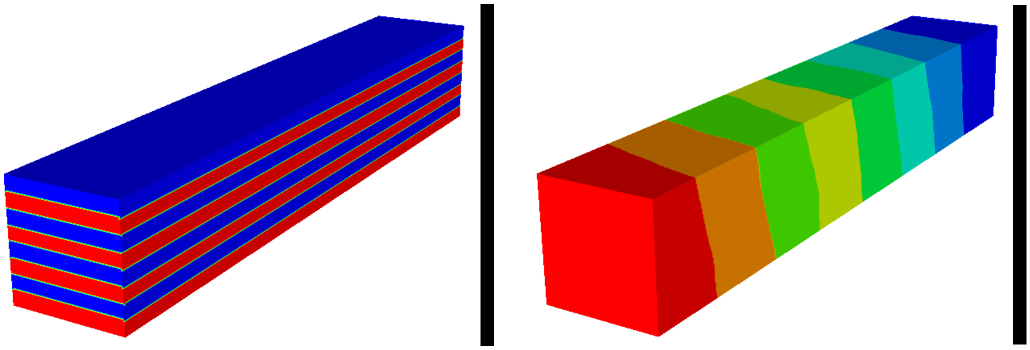

FIGURE 3. *Material coefficient, alternating layers of steel and rubber (left) and domain decomposition into 8 subdomains with a graph partitioner (right)*

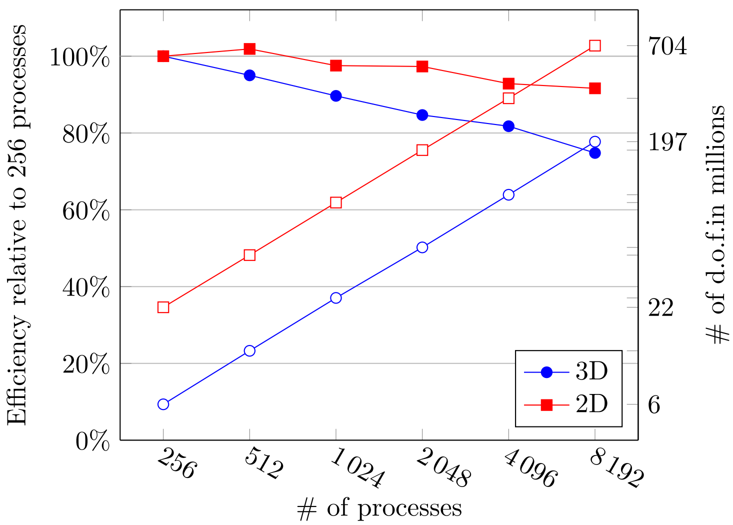

FIGURE 4. *Weak scaling experiments.*

performed inside FreeFem++ code interfaced with the domain decomposition library HPDDM [26, 27]. The factorizations are computed for each local problem and also for the global coarse problem using MUMPS [1]. Generalized eigenvalue problems to generate the GenEO space are solved using ARPACK [29]. The coarse space is formed only with the generalized eigenvalue problem (19) since we noticed that the second one (21) has only a little effect on the convergence. All the results of this section were obtained on Turing machine which is an IBM/Blue Gene/Q machine composed of 1024 compute nodes where each one is made of 16 cores PowerPC A2 clocked at 1.6 GHz.

These computations, see Figure 4, assess the weak scalability of the algorithm with respect to the problem size and the number of subdomains. All times are wall clock times. The domain is decomposed automatically into subdomains with a graph partitioner, ranging from 256 subdomains to 8192. and the problem size is increased by mesh refinement. In 3D the initial problem is about 6 millions d.o.f decomposed into 256 subdomains and solved in 145.2s and the final problem is about 197 millions of d.o.f decomposed into 8192 subdomains and solved in 196s which gives an efficiency

	N	Factorization	Deflation	Solution	# of it.	Total	# of d.o.f.
3D	256	25.2 s	76.0 s	37.2 s	46	145.2 s	$6.1 \cdot 10^6$
	512	26.5 s	81.1 s	39.8 s	47	155.1 s	$12.4 \cdot 10^6$
	1 024	29.2 s	82.6 s	41.7 s	45	165.5 s	$25.0 \cdot 10^6$
	2 048	26.9 s	83.5 s	46.3 s	47	171.0 s	$48.8 \cdot 10^6$
	4 096	28.3 s	88.8 s	54.5 s	53	177.7 s	$97.9 \cdot 10^6$
	8 192	29.0 s	78.3 s	79.8 s	60	196.1 s	$197.6 \cdot 10^6$
2D	256	4.8 s	72.9 s	39.9 s	46	123.9 s	$22.1 \cdot 10^6$
	512	4.7 s	65.9 s	45.0 s	51	121.3 s	$44.0 \cdot 10^6$
	1 024	4.8 s	70.0 s	46.1 s	51	127.0 s	$88.3 \cdot 10^6$
	2 048	4.8 s	69.0 s	46.5 s	51	127.4 s	$176.8 \cdot 10^6$
	4 096	4.8 s	65.8 s	52.8 s	56	132.6 s	$351.0 \cdot 10^6$
	8 192	4.8 s	65.4 s	53.0 s	54	134.8 s	$704.1 \cdot 10^6$

FIGURE 5. *Weak scaling experiments elasticity timings tab .*

near to 75%. For the 2D case, the initial problem is approximately of size 22 millions unknowns (d.o.f) decomposed into 256 subdomains and solved in 123.9s and we end up with a bigger problem about 704 millions unknowns (d.o.f) decomposed into 8192 subdomains and solved in 134s. The efficiency is close to 90%. In figure table 5, we report the number of GMRES iterations. They increase very slowly as the mesh is refined which shows the scalability of the preconditioner. We report in the same table all the timings concerning the algorithm, column "Factorization" concerns the local subdomains, the assembling and the factorization of the coarse operator are in column "Deflation" and in column "Solution" we display the time spent by GMRES.

6.4. 3D and 2D incompressible Stokes system. Using the same libraries, we also performed a strong scaling test for an incompressible Stokes system of equations for a driven cavity problem :

Find $(\boldsymbol{u}, \boldsymbol{p}) \in H(\Omega)^{d=2,3} \times L_0(\Omega)$ such that

$$- \operatorname{div} \underline{\underline{\sigma}}_F(\boldsymbol{u}, \boldsymbol{p}) = 0, \quad \text{and } \operatorname{div}(\boldsymbol{u}) = 0 \quad \text{in } \Omega, \tag{40}$$

with

$$\begin{cases} \underline{\underline{\sigma}}_F(\boldsymbol{u}, \boldsymbol{p}) = -pI + 2\mu \underline{\underline{\varepsilon}}(\boldsymbol{u}), \\ \underline{\underline{\varepsilon}}(\boldsymbol{u}) = \frac{1}{2}(\underline{\underline{\nabla}}\, \boldsymbol{u} + (\underline{\underline{\nabla}}\, \boldsymbol{u})^T) \text{ and } \varepsilon_{i,j} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right). \end{cases} \tag{41}$$

As a boundary conditions, we consider a continuous velocity on the upper face and zero Dirichlet otherwise. The computations are done in both two and three dimensions on a domain $\Omega = [0, 1]^2$ and $\Omega = [0, 1]^3$, respectively. Once more the problems are discretized via Taylor-Hood finite element P_2/P_1 with a continuous pressure.

We assess here the strong scalability of the algorithm. For this, we make the number of subdomains vary for a fixed global system size. In our test case the system size is fixed to 50 millions unknowns (d.o.f) in 3D and to 100 millions unknowns (d.o.f) in 2D, as we can show in figure 6, from 1024 subdomains to 8192 subdomains we get a quite good speed up. In the three dimensional case, we pass from 387.5s

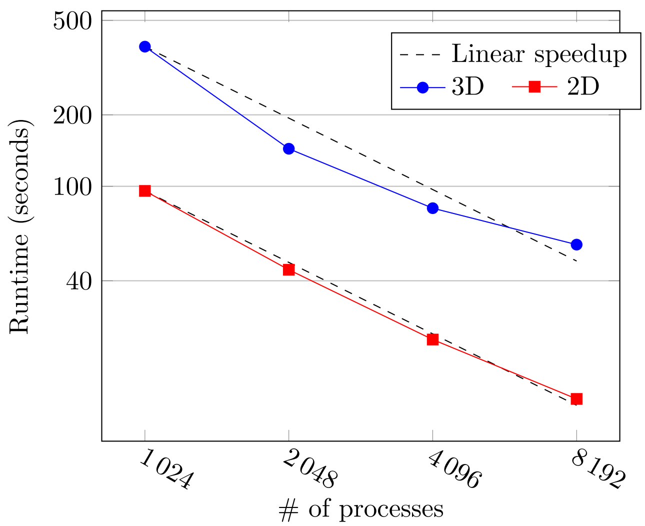

FIGURE 6. *Timings of various simulations Stokes.*

	N	Factorization	Deflation	Solution	# of it.	Total	# of d.o.f.
3D	1 024	79.2 s	229.0 s	76.3 s	45	387.5 s	$50.63 \cdot 10^6$
	2 048	29.5 s	76.5 s	34.8 s	42	143.9 s	
	4 096	11.1 s	45.8 s	19.8 s	42	80.9 s	
	8 192	4.7 s	26.1 s	14.9 s	41	56.8 s	
2D	1 024	5.2 s	37.9 s	51.5 s	51	95.6 s	$100.13 \cdot 10^6$
	2 048	2.4 s	19.3 s	22.1 s	42	44.5 s	
	4 096	1.1 s	10.4 s	10.2 s	35	22.6 s	
	8 192	0.5 s	4.6 s	6.9 s	38	12.7 s	

FIGURE 7. *Strong scaling experiments Stokes.*

using 1024 subdomains to 56.8s when using 8192 subdomains. In figure table 7 we display all timings relative to this test, column "Factorization" gives the time spent in the factorization of the local submatrices, column "Deflation" corresponds to local eigenvalue solvers and the coarse space correction construction, column "Solution" is the time taken by the GMRES solve of the global linear system by the domain decomposition algorithm.

7. Conclusion. We developed a theory for the overlapping P.L. Lions' algorithm similar to the existing one for the Schwarz algorithm in that we show how to build adaptively a coarse space so that the two-level preconditioner achieves a targeted condition number. The theory is based on the introduction of the SORAS (14) algorithm which is a new symmetric variant of the ORAS preconditioner. The two-level method is implemented in the HPDDM library that is interfaced with finite element solvers such as FreeFem++ and Feel++.

Note that for a given targeted condition number, the size of the coarse space depends on the interface condition. A small coarse space is important in order to achieve good scalability results. Thus, it might be interesting to optimize this condition with respect to the coarse space size.

Références

[1] P. R. Amestoy, I. S. Duff, J.-Y. L'Excellent, and J. Koster, *A fully asynchronous multifrontal solver using distributed dynamic scheduling*, SIAM J. Matrix Analysis and Applications, 23 (2001), pp. 15–41.

[2] J.-D. Benamou and B. Desprès, *A domain decomposition method for the Helmholtz equation and related optimal control problems*, J. Comput. Phys., 136 (1997), pp. 68–82.

[3] M. Benzi, G. H. Golub, and J. Liesen, *Numerical solution of saddle point problems*, Acta Numer., 14 (2005), pp. 1–137.

[4] M. Benzi and V. Simoncini, *On the eigenvalues of a class of saddle point matrices*, Numer. Math., 103 (2006), pp. 173–196.

[5] S. C. Brenner and L. R. Scott, *The Mathematical Theory of Finite Element Methods*, Springer, New York, Applied Mathematics ed., 2008.

[6] X.-C. Cai and M. Sarkis, *A restricted additive Schwarz preconditioner for general sparse linear systems*, SIAM Journal on Scientific Computing, 21 (1999), pp. 239–247.

[7] L. Conen, V. Dolean, R. Krause, and F. Nataf, *A coarse space for heterogeneous Helmholtz problems based on the Dirichlet-to-Neumann operator*, J. Comput. Appl. Math., 271 (2014), pp. 83–99.

[8] B. Després, *Décomposition de domaine et problème de Helmholtz*, C.R. Acad. Sci. Paris, 1 (1990), pp. 313–316.

[9] ———, *Domain decomposition method and the Helmholtz problem.II*, in Second International Conference on Mathematical and Numerical Aspects of Wave Propagation (Newark, DE, 1993), Philadelphia, PA, 1993, SIAM, pp. 197–206.

[10] B. Després, P. Joly, and J. E. Roberts, *A domain decomposition method for the harmonic Maxwell equations*, in Iterative methods in linear algebra (Brussels, 1991), Amsterdam, 1992, North-Holland, pp. 475–484.

[11] V. Dolean, L. G. Giorda, and M. J. Gander, *Optimized Schwarz methods for Maxwell equations*, SIAM J. Scient. Comp., 31 (2009), pp. 2193–2213.

[12] V. Dolean, P. Jolivet, and F. Nataf, *An Introduction to Domain Decomposition Methods : algorithms, theory and parallel implementation*, SIAM, 2015.

[13] O. Dubois, M. J. Gander, S. Loisel, A. St-Cyr, and D. B. Szyld, *The optimized Schwarz method with a coarse grid correction*, SIAM J. Sci. Comput., 34 (2012), pp. A421–A458.

[14] Y. Efendiev, J. Galvis, R. Lazarov, and J. Willems, *Robust domain decomposition preconditioners for abstract symmetric positive definite bilinear forms*, ESAIM Math. Model. Numer. Anal., 46 (2012), pp. 1175–1199.

[15] E. Efstathiou and M. J. Gander, *RAS : Understanding restricted additive Schwarz*, Tech. Rep. 06, McGill University, 2002.

[16] E. Efstathiou and M. J. Gander, *Why restricted additive Schwarz converges faster than additive Schwarz*, BIT, 43 (2003), pp. 945–959.

[17] C. Farhat, A. Macedo, and M. Lesoinne, *A two-level domain decomposition method for the iterative solution of high-frequency exterior Helmholtz problems*, Numer. Math., 85 (2000), pp. 283–303.

[18] A. Frommer and D. B. Szyld, *An algebraic convergence theory for restricted additive Schwarz methods using weighted max norms*, SIAM J. Numer. Anal., 39 (2001), pp. 463–479 (electronic).

[19] J. Galvis and Y. Efendiev, *Domain decomposition preconditioners for multiscale flows in high contrast media : reduced dimension coarse spaces*, Multiscale Model. Simul., 8 (2010), pp. 1621–1644.

[20] M. J. Gander, *Optimized Schwarz methods*, SIAM J. Numer. Anal., 44 (2006), pp. 699–731.

[21] M. J. Gander, F. Magoulès, and F. Nataf, *Optimized Schwarz methods without overlap for the Helmholtz equation*, SIAM J. Sci. Comput., 24 (2002), pp. 38–60.

[22] R. Haferssas, *Espaces grossiers pour les méthodes de décomposition de domaine avec conditions d'interface optimisées*, PhD thesis, UPMC, 2016.

[23] P. Havé, R. Masson, F. Nataf, M. Szydlarski, H. Xiang, and T. Zhao, *Algebraic domain decomposition methods for highly heterogeneous problems*, SIAM J. Sci. Comput., 35 (2013), pp. C284–C302.

[24] F. Hecht, *New development in freefem++*, J. Numer. Math., 20 (2012), pp. 251–265.

[25] C. Japhet, F. Nataf, and F.-X. Roux, *The Optimized Order 2 Method with a coarse grid preconditioner. application to convection-diffusion problems*, in Ninth International Conference on Domain Decompositon Methods in Science and Engineering, P. Bjorstad, M. Espedal, and D. Keyes, eds., John Wiley & Sons, 1998, pp. 382–389.

[26] P. Jolivet, F. Hecht, F. Nataf, and C. Prud'homme, *Scalable domain decomposition preconditioners for heterogeneous elliptic problems*, in Proceedings of the 2013 ACM/IEEE conference on Supercomputing, SC13, ACM, 2013, pp. 80 :1–80 :11. Best paper finalist.

[27] P. Jolivet and F. Nataf, *Hpddm : High-Performance Unified framework for Domain Decomposition methods, MPI-C++ library.* https ://github.com/hpddm/hpddm, 2014.

[28] J.-H. Kimn, *A convergence theory for an overlapping Schwarz algorithm using discontinuous iterates*, Numer. Math., 100 (2005), pp. 117–139.

[29] R. B. Lehoucq, D. C. Sorensen, and C. Yang, *ARPACK users' guide : solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods*, vol. 6, SIAM, 1998.

[30] J. Liesen and B. N. Parlett, *On nonsymmetric saddle point matrices that allow conjugate gradient iterations*, Numer. Math., 108 (2008), pp. 605–624.

[31] P.-L. Lions, *On the Schwarz alternating method. II.*, in Domain Decomposition Methods, T. Chan, R. Glowinski, J. Périaux, and O. Widlund, eds., Philadelphia, PA, 1989, SIAM, pp. 47–70.

[32] ———, *On the Schwarz alternating method. III : a variant for nonoverlapping subdomains*, in Third International Symposium on Domain Decomposition Methods for Partial Differential Equations , held in Houston, Texas, March 20-22, 1989, T. F. Chan, R. Glowinski, J. Périaux, and O. Widlund, eds., Philadelphia, PA, 1990, SIAM.

[33] ———, *On the Schwarz alternating method. III : a variant for nonoverlapping subdomains*, in First International Symposium on Domain Decomposition Methods for Partial Differential Equations, T. F. Chan, R. Glowinski, J. Périaux, and O. Widlund, eds., Philadelphia, PA, 1990, SIAM.

[34] S. Loisel, H. Nguyen, and S. R., *Optimized schwarz and 2-lagrange methods for multiscale pdes*, tech. rep., Bath University, Mathematical Sciences Dept., 2014.

[35] S. Loisel and D. B. Szyld, *On the geometric convergence of optimized Schwarz methods with applications to elliptic problems*, Numer. Math., 114 (2010), pp. 697–728.

[36] J. Mandel, *Balancing domain decomposition*, Comm. on Applied Numerical Methods, 9 (1992), pp. 233–241.

[37] F. Nataf, F. Rogier, and E. de Sturler, *Optimal interface conditions for domain decomposition methods*, Tech. Rep. 301, CMAP (Ecole Polytechnique), 1994.

[38] F. Nataf, H. Xiang, V. Dolean, and N. Spillane, *A coarse space construction based on local Dirichlet to Neumann maps*, SIAM J. Sci Comput., 33 (2011), pp. 1623–1642.

[39] S. V. Nepomnyaschikh, *Mesh theorems of traces, normalizations of function traces and their inversions*, Sov. J. Numer. Anal. Math. Modeling, 6 (1991), pp. 1–25.

[40] R. A. Nicolaides, *Deflation of conjugate gradients with applications to boundary value problems*, SIAM J. Numer. Anal., 24 (1987), pp. 355–365.

[41] H. A. Schwarz, *Über einen Grenzübergang durch alternierendes Verfahren*, Vierteljahrsschrift der Naturforschenden Gesellschaft in Zürich, 15 (1870), pp. 272–286.

[42] N. Spillane, V. Dolean, P. Hauret, F. Nataf, C. Pechstein, and R. Scheichl, *Abstract robust coarse spaces for systems of PDEs via generalized eigenproblems in the overlaps*, Numer. Math., 126 (2014), pp. 741–770.

[43] N. Spillane, V. Dolean, P. Hauret, F. Nataf, and D. Rixen, *Solving generalized eigenvalue problems on the interfaces to build a robust two-level FETI method*, C. R. Math. Acad. Sci. Paris, 351 (2013), pp. 197–201.

[44] A. St-Cyr, M. J. Gander, and S. J. Thomas, *Optimized Multiplicative, Additive, and Restricted Additive Schwarz Preconditioning*, SIAM J. Sci. Comput., 29 (2007), pp. 2402–2425 (electronic).

[45] A. Toselli and O. Widlund, *Domain Decomposition Methods - Algorithms and Theory*, vol. 34 of Springer Series in Computational Mathematics, Springer, 2005.

[46] X. Tu and J. Li, *A FETI-DP type domain decomposition algorithm for three-dimensional incompressible Stokes equations*, SIAM J. Numer. Anal., 53 (2015), pp. 720–742.