Available online at www.sciencedirect.com
SCIENCE DIRECT®
Comput. Methods Appl. Mech. Engrg. 195 (2006) 3742–3759
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

Algebraic approximation of Dirichlet-to-Neumann maps for the equations of linear elasticity

Frédéric Magoulès [a,*], François-Xavier Roux [b], Laurent Series [b]

[a] Institut Elie Cartan de Nancy, Université Henri Poincaré, BP 239, 54506 Vandoeuvre-les-Nancy Cedex, France
[b] High Performance Computing Unit, ONERA, 29 av. de la Division Leclerc, BP 72, 92322 Châtillon Cedex, France

Received 20 October 2004; received in revised form 29 December 2004; accepted 9 January 2005

Abstract

The absorbing boundary conditions defined on the interface between the sub-domains are of major importance for the convergence of domain decomposition methods. In linear elasticity, optimal absorbing boundary conditions can be derived and are associated with a Dirichlet-to-Neumann map. In this paper, several original algebraic techniques of approximation of this Dirichlet-to-Neumann map are investigated. Asymptotic, spectral and numerical analysis of these techniques are successively presented for linear elasticity problems. Various numerical experiments illustrate the convergence properties of these original techniques.

Keywords: Dirichlet-to-Neumann; Absorbing boundary conditions; Domain decomposition; Linear elasticity

1. Introduction

During the last decade, a large amount of research has been carried out on domain decomposition methods with Lagrange multipliers. In these methods, the global domain is partitioned into non-overlapping sub-domains. The continuity is then enforced by using additional Lagrange multipliers [8] across the interfaces between the sub-domains. More recently [6,7], additional augmented matrices defined on the interface have been used in order to avoid possible inaccurate sub-problems. These augmented matrices are linked with the boundary conditions defined on the interface between the sub-domains. For this reason, the development of efficient absorbing boundary conditions has become a real challenge in domain decomposition.

* Corresponding author. Tel.: +33 383 684564; fax: +33 383 684534.
E-mail addresses: frederic.magoules@iecn.u-nancy.fr (F. Magoulès), francois-xavier.roux@onera.fr (F.-X. Roux), laurent.series@onera.fr (L. Series).

0045-7825/$ - see front matter
doi:10.1016/j.cma.2005.01.022

A reformulation of the problem leads to a linear system upon the Lagrange multipliers. This linear system is then solved iteratively. Without a global preconditioner, using some absorbing boundary conditions that involve a Dirichlet-to-Neumann (DtN) map of the outside of each sub-domain, leads to the optimal convergence of the domain decomposition algorithm [18,19]. This DtN map models the physical phenomena involved outside each sub-domain. It is not a local operator. A large number of very successful practical implementations of non-local boundary operators have been reported in the literature, mostly in the context of artificial boundary conditions for infinite domains, but also in the context of domain decomposition [5]. These non-local approaches often outperform their local competitors in spite of a higher cost [13,23]. On the contrary, local approximations of this DtN operator have been recently developed in order to reduce the computational cost. Several approaches have thus been investigated these last 10 years, and can be classified into two major classes. The first class consists of continuous approximations of the continuous DtN operator with Fourier analysis tools, see [1,2,9,10,16,17]. The second class consists of discrete approximations of the discrete DtN operator with algebraic techniques, see [18,20,21]. In this paper, original alternatives to these algebraic approximations are introduced. These new techniques consist of defining small and local DtN maps in order to approximate the complete and non-local DtN maps.

This paper is organized as follows. Section 2 introduces the equations of linear elasticity and the linear system of equations considered. Section 3 presents the domain decomposition method considered in this paper with two Lagrange multipliers and two augmented matrices defined on the interface between the sub-domains. Then the Dirichlet-to-Neumann map involved for the optimal convergence of this domain decomposition method is reminded. Section 4 introduces new and original techniques of approximation of this Dirichlet-to-Neumann map. These techniques are based on the computation of a 'small' Dirichlet-to-Neumann map. An asymptotic and a spectral analysis of these techniques are presented in Section 5. Numerical experiments are reported in Section 6. Finally, Section 7 contains the conclusions of this paper.

2. Equations of linear elasticity

In this paper the domain $\Omega \in \mathbb{R}^3$ denotes an elastic body, and $\partial\Omega$ denotes its boundary. A part of the boundary Γ_0 is assumed to be clamped, and the another part Γ_1 is subject to a surface force g, such that $\partial\Omega = \Gamma_0 \cup \Gamma_1$. A body force f is considered in the domain Ω. The linear elasticity problem consists of finding the displacement u, such that:

$$\begin{aligned} &-\operatorname{div}(\sigma(u)) = f, \quad \text{in } \Omega,\\ &u = 0, \quad \text{on } \Gamma_0,\\ &\sigma(u)\cdot n = g, \quad \text{on } \Gamma_1, \end{aligned}$$

where n is the outward normal vector along Γ_1. The quantity $\sigma(u)$ is the so-called stress tensor, and is linked with the strain tensor by the linear equation (1) known in elasticity as the generalized Hooke's law for isotropic materials:

$$\sigma_{ij}(u) = \lambda\epsilon_{kk}(u)\delta_{ij} + 2\mu\epsilon_{ij}, \tag{1}$$

where the summation convention applies to repeated indices. The quantities λ and μ are the Lamé coefficients of the material composing the body. They are linked with the Poisson's ratio v and the Young's modulus E by the relation:

$$\mu = \frac{E}{2(1+v)}, \quad \lambda = \frac{Ev}{(1+v)(1-2v)}.$$

In the case of small displacements and small strains, the quantity $\epsilon(u)$ is defined as

$$\epsilon(u) = \frac{1}{2}(\nabla u + \nabla u^{\mathrm{T}})$$

and this definition is not restricted to isotropic bodies. The variational formulation of the linear elasticity problem is

$$\int_{\Omega} \sigma(u)\mathrm{grad}(v) = \int_{\Omega} fv + \int_{\Gamma_1} gv.$$

The finite element discretization of the above boundary value problem leads to the following system of equations:

$$Ku = b. \tag{2}$$

The symmetric definite positive matrix K is called the stiffness matrix of the problem, and b its right-hand side vector.

3. Dirichlet-to-Neumann maps and domain decomposition

3.1. A domain decomposition method for linear elasticity

The global domain Ω is meshed and partitioned into two non-overlapping sub-domains $\Omega^{(1)}$ and $\Omega^{(2)}$ with an interface $\Gamma = \Omega^{(1)} \cap \Omega^{(2)}$, as represented in Fig. 1. Let $K^{(s)}$, and $b^{(s)}$ denote respectively the stiffness matrix, and the right-hand side vector associated with the sub-domain $\Omega^{(s)}$, $s = 1, 2$, and let $u^{(s)}$ denotes the restriction to $\Omega^{(s)}$ of the solution of problem (2). The vector $u^{(s)}$ is partitioned into two components:

$$u^{(s)} = \begin{pmatrix} u_i^{(s)} \\ u_p^{(s)} \end{pmatrix}, \tag{3}$$

where the subscripts i and p respectively designate internal and interface boundary unknowns. With these notations the local stiffness matrix and right-hand side can be written:

$$K^{(s)} = \begin{pmatrix} K_{ii}^{(s)} & K_{ip}^{(s)} \\ K_{pi}^{(s)} & K_{pp}^{(s)} \end{pmatrix}, \quad b^{(s)} = \begin{pmatrix} b_i^{(s)} \\ b_p^{(s)} \end{pmatrix}.$$

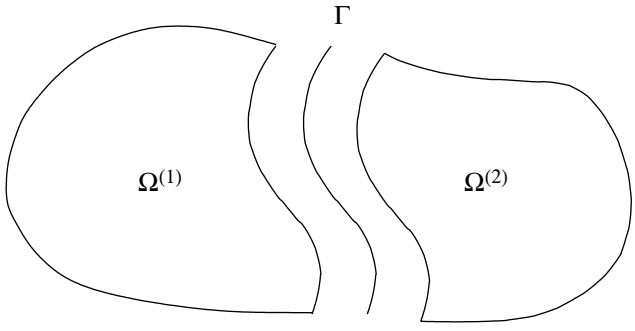

Fig. 1. Two-domains partitioning.

The global problem is a block system obtained by assembling local contribution of each sub-domain:

$$\begin{pmatrix} K_{ii}^{(1)} & 0 & K_{ip}^{(1)} \\ 0 & K_{ii}^{(2)} & K_{ip}^{(2)} \\ K_{pi}^{(1)} & K_{pi}^{(2)} & K_{pp} \end{pmatrix} \begin{pmatrix} u_i^{(1)} \\ u_i^{(2)} \\ u_p \end{pmatrix} = \begin{pmatrix} b_i^{(1)} \\ b_i^{(2)} \\ b_p \end{pmatrix}. \tag{4}$$

The matrices $K_{pp}^{(1)}$ and $K_{pp}^{(2)}$ represent the interaction matrices between the nodes on the interface obtained by integration on $\Omega^{(1)}$ and on $\Omega^{(2)}$, and $K_{pp} = K_{pp}^{(1)} + K_{pp}^{(2)}$. In a same way, the term $b_p = b_p^{(1)} + b_p^{(2)}$ is obtained by local integration of right-hand side over each sub-domain and summation on the interface.

For the previous partition, it is clear that the global problem (4) is equivalent to the local sub-problems

$$\begin{pmatrix} K_{ii}^{(1)} & K_{ip}^{(1)} \\ K_{pi}^{(1)} & K_{pp}^{(1)} \end{pmatrix} \begin{pmatrix} u_i^{(1)} \\ u_p^{(1)} \end{pmatrix} = \begin{pmatrix} b_i^{(1)} \\ b_p^{(1)} + \lambda^{(1)} \end{pmatrix}, \tag{5}$$

$$\begin{pmatrix} K_{ii}^{(2)} & K_{ip}^{(2)} \\ K_{pi}^{(2)} & K_{pp}^{(2)} \end{pmatrix} \begin{pmatrix} u_i^{(2)} \\ u_p^{(2)} \end{pmatrix} = \begin{pmatrix} b_i^{(2)} \\ b_p^{(2)} + \lambda^{(2)} \end{pmatrix}, \tag{6}$$

under the admissibility constraint

$$u_p^{(1)} = u_p^{(2)} \tag{7}$$

and the equilibrium constraint

$$0 = \lambda^{(1)} + \lambda^{(2)}. \tag{8}$$

As already discussed in [18], additional matrices defined on the interface can be introduced. This leads to the equivalence of the global problem with the local sub-problems

$$\begin{pmatrix} K_{ii}^{(1)} & K_{ip}^{(1)} \\ K_{pi}^{(1)} & K_{pp}^{(1)} + A_{pp}^{(1)} \end{pmatrix} \begin{pmatrix} u_i^{(1)} \\ u_p^{(1)} \end{pmatrix} = \begin{pmatrix} b_i^{(1)} \\ b_p^{(1)} + \lambda^{(1)} \end{pmatrix}, \tag{9}$$

$$\begin{pmatrix} K_{ii}^{(2)} & K_{ip}^{(2)} \\ K_{pi}^{(2)} & K_{pp}^{(2)} + A_{pp}^{(2)} \end{pmatrix} \begin{pmatrix} u_i^{(2)} \\ u_p^{(2)} \end{pmatrix} = \begin{pmatrix} b_i^{(2)} \\ b_p^{(2)} + \lambda^{(2)} \end{pmatrix} \tag{10}$$

under the admissibility constraint

$$u_p^{(1)} = u_p^{(2)} \tag{11}$$

and the modified equilibrium constraint

$$A_{pp}^{(1)} u_p^{(1)} + A_{pp}^{(2)} u_p^{(2)} = \lambda^{(1)} + \lambda^{(2)}, \tag{12}$$

where $A_{pp}^{(s)}$ for $s = 1, 2$ is an augmented matrix defined on the interface. A combination of Eqs. (11) and (12), as performed in [18], leads to the following continuity constraints:

$$\lambda^{(1)} + \lambda^{(2)} - A_{pp}^{(1)} u_p^{(2)} - A_{pp}^{(2)} u_p^{(2)} = 0, \tag{13}$$

$$\lambda^{(1)} + \lambda^{(2)} - A_{pp}^{(1)} u_p^{(1)} - A_{pp}^{(2)} u_p^{(1)} = 0. \tag{14}$$

The variables $u_i^{(1)}$ and $u_i^{(2)}$ can be eliminated from Eqs. (9) and (10) which leads to

$$[S_{pp}^{(1)} + A_{pp}^{(1)}] u_p^{(1)} = b_p^{(1)} - K_{pi}^{(1)} [K_{ii}^{(1)}]^{-1} b_i^{(1)} + \lambda^{(1)}, \tag{15}$$

$$[S_{pp}^{(2)} + A_{pp}^{(2)}] u_p^{(2)} = b_p^{(2)} - K_{pi}^{(2)} [K_{ii}^{(2)}]^{-1} b_i^{(2)} + \lambda^{(2)}. \tag{16}$$

The matrix $S_{pp}^{(s)} := K_{pp}^{(s)} - K_{pi}^{(s)}[K_{ii}^{(s)}]^{-1}K_{ip}^{(s)}$ is the so-called Schur complement matrix and the vector $c_p^{(s)} := b_p^{(s)} - K_{pi}^{(s)}[K_{ii}^{(s)}]^{-1}b_i^{(s)}$ is the condensed right-hand side in sub-domain $\Omega^{(s)}$, for $s = 1, 2$. Substitution of $u_p^{(1)}$ and $u_p^{(2)}$ obtained from Eqs. (15) and (16) in Eqs. (13) and (14), gives the following linear system:

$$F\lambda = d, \tag{17}$$

where the matrix F, the vector λ and the right-hand side d are equal to:

$$F := \begin{pmatrix} I & I - (A_{pp}^{(1)} + A_{pp}^{(2)})[S_{pp}^{(2)} + A_{pp}^{(2)}]^{-1} \\ I - (A_{pp}^{(1)} + A_{pp}^{(2)})[S_{pp}^{(1)} + A_{pp}^{(1)}]^{-1} & I \end{pmatrix},$$

$$\lambda := \begin{pmatrix} \lambda^{(1)} \\ \lambda^{(2)} \end{pmatrix}, \quad d := \begin{pmatrix} (A_{pp}^{(1)} + A_{pp}^{(2)})[S_{pp}^{(2)} + A_{pp}^{(2)}]^{-1}c_p^{(1)} \\ (A_{pp}^{(1)} + A_{pp}^{(2)})[S_{pp}^{(1)} + A_{pp}^{(1)}]^{-1}c_p^{(2)} \end{pmatrix},$$

with the partitioning of the unknowns introduced Eq. (3).

3.2. Interpretation in term of absorbing boundary conditions

Eqs. (9) and (10) can be interpreted as the discretization of two sub-domain linear elasticity sub-problems with the following absorbing boundary conditions on the interface:

$$\sigma^{(1)}(u^{(1)}) \cdot n^{(1)} + \mathscr{A}^{(1)}u^{(1)} = \lambda^{(1)}, \quad \text{on } \Gamma, \tag{18}$$

$$\sigma^{(2)}(u^{(2)}) \cdot n^{(2)} + \mathscr{A}^{(2)}u^{(2)} = \lambda^{(2)}, \quad \text{on } \Gamma \tag{19}$$

and therefore the two last equations (13) and (14) arise from the discretization of the following continuity conditions

$$\sigma^{(1)}(u^{(1)}) \cdot n^{(1)} + \mathscr{A}^{(1)}u^{(1)} = -\sigma^{(2)}(u^{(2)}) \cdot n^{(2)} + \mathscr{A}^{(1)}u^{(2)}, \quad \text{on } \Gamma,$$

$$\sigma^{(2)}(u^{(2)}) \cdot n^{(2)} + \mathscr{A}^{(2)}u^{(2)} = -\sigma^{(1)}(u^{(1)}) \cdot n^{(1)} + \mathscr{A}^{(2)}u^{(1)}, \quad \text{on } \Gamma,$$

where $\mathscr{A}^{(s)}$ for $s = 1, 2$ is the continuous operator associated to the matrix $A_{pp}^{(s)}$, and $n^{(s)}$ is the outward normal unitary vector of the sub-domain $\Omega^{(s)}$ along the interface Γ.

As described in [7] for the Helmholtz equation, the above continuity conditions obtained for the equations of linear elasticity are equivalent to the classical continuity conditions (20) and (21) obtained from (7) and (8):

$$u^{(1)} = u^{(2)}, \quad \text{on } \Gamma, \tag{20}$$

$$0 = \sigma^{(1)}(u^{(1)}) \cdot n^{(1)} - \sigma^{(2)}(u^{(2)}) \cdot n^{(2)}, \quad \text{on } \Gamma, \tag{21}$$

since the operator $\mathscr{A}^{(s)}$ for $s = 1, 2$ and the associated matrix $A_{pp}^{(s)}$ are invertible. It is well known that the classical continuity conditions (20) and (21) may lead to a non-well-posed linear elasticity sub-problem in one of the sub-domains [6,8]. In this case, an additional procedure for the detection of the rigid body motions of the sub-domain is required to solve this linear elasticity sub-problem [8].

The purpose of a carefully constructed matrix $A_{pp}^{(s)}$ for $s = 1, 2$ is to prevent the singularity of the sub-domain matrix. In other words, the detection of the rigid body motions is no longer required to solve the linear elasticity sub-problems, even in the case of general domain partitioning.

In this paper, some methods to build the matrices $A_{pp}^{(s)}$ are proposed for linear elasticity. These methods are based on an algebraic analysis and can thus be applied to other problems without additional difficulties. For this reason, the discretization of the absorbing boundary conditions has not been presented, but these conditions have been interpreted from the discrete equations.

3.3. Dirichlet-to-Neumann maps

The linear system (17) is usually solved with an iterative method. In the case of a partition into two domains and if a simple iterative Jacobi algorithm is used, it can been proved that choosing $A_{pp}^{(s)}$ for $s = 1, 2$ equal to the Schur complement of the neighboring sub-domain i.e. $A_{pp}^{(s)} := S_{pp}^{(q)}$ for $q = 1, 2$, $q \neq s$, leads to the convergence of the iterative algorithm into one iteration only, see [18,20]. In the case of a one-way partitioning into N_s sub-domains, choosing $A_{pp}^{(s)}$ equal to the complete outer Schur complement of the sub-domain $\Omega^{(s)}$, $\forall s = 1, \ldots, N_s$ leads to the convergence of the iterative algorithm into $N_s - 1$ iterations only, see [18,20]. Since the exchange of information between the sub-domains is reduced to their common interfaces, the convergence can never be achieved in less that $N_s - 1$ iterations for a one-way partitioning without an additional global preconditioner. For this reason, the previous choice of the augmented matrix $A_{pp}^{(s)}$ is called the optimal choice in the literature.

In the continuous analysis, this optimal selection consists of choosing as the operator $\mathscr{A}^{(s)}$ involved in the absorbing boundary conditions (18) and (19) a Steklov–Poincaré operator [11,19]. The purpose of this Steklov–Poincaré operator is to avoid possible reflection of the elastic waves on the interface between the sub-domains. For this reason, this operator is usually called transparent operator in the literature.

From a physical point of view the Steklov–Poincaré operator and the Schur complement matrix described previously are the continuous and the discrete expression of the Dirichlet-to-Neumann (DtN) map of the outside of the sub-domain on its interface boundary. This DtN map models the physical phenomena involved outside the domain $\Omega^{(s)}$. Even if the choice of the DtN map is optimal in the definition of the absorbing boundary conditions between the sub-domains, it is not a local operator. For this reason, several approximations of this non-local operator have been performed, first independently of domain decomposition, see for example [4,12,14,15] and references therein. In the particular context of domain decomposition methods, several approaches have been investigated [3], and can be classified into two major classes. The first class consists of continuous approximations of the continuous DtN operator with Fourier analysis tools, see [1,2,9,10]. The second class consists of discrete approximations of the discrete DtN operator with algebraic techniques, see [18,20,21]. In this paper, original alternatives to these algebraic approximations are introduced. These new techniques consist of defining small and local DtN maps in order to approximate the complete and non-local DtN maps.

4. Algebraic approximation of Dirichlet-to-Neumann maps

In the previous sections, it has been recalled that the best choice for the absorbing boundary conditions (18) and (19) is linked with the DtN map of the outside of each sub-domain on its interface boundary. After discretization, this choice leads to a dense augmented matrix equal to the complete outer Schur complement matrix.

4.1. Neighbor Schur complement

From a structural point of view, the complete outer Schur complement matrix models the stiffness of all the outer sub-domains. To restrict the information only to the neighboring sub-domains, implies to consider the Schur complement matrix of the neighboring sub-domains only.

This remark leads to a first approximation of the complete outer Schur complement by the neighbor Schur complement [18,20,21]. Even if this matrix is still dense, the computational cost and the exchange of data are reduced to the neighboring sub-domains only. This reduces significantly the CPU time required to compute the augmented matrix $A_{pp}^{(s)}$.

4.2. Sparse mask approximation

Since the neighbor Schur complement matrix is a dense matrix, using it as an augmented matrix increases the bandwidth of the sub-domain matrix. This implies a lot of additional operations during the factorization of the sub-domain matrix and during the forward–backward substitutions required at each iteration of the iterative algorithm. To reduce these costs, a simple sparse mask of the neighbor Schur complement matrix can be considered. This technique presents the advantage of keeping the sparsity of the sub-domain matrix after the addition of the augmented matrix. Unfortunately the computation of the neighbor Schur complement matrix cannot be avoided. This is the main drawback of this approach.

4.3. Dense approximation

In order to describe this new approach, the case of a general domain Ω split into two non-overlapping sub-domains $\Omega^{(1)}$ and $\Omega^{(2)}$, with an interface Γ is now considered, as represented in Fig. 1. The following subsets of indexes are defined for the nodes of the sub-domain $\Omega^{(s)}$, $s = 1, 2$:

$$
\begin{aligned}
V &= \{\text{indexes of nodes inside the sub-domain and on the interface}\},\\
V_\Gamma &= \{\text{indexes of nodes on the interface } \Gamma\},\\
V_d &= \{\text{indexes of nodes belonging to } V \text{ such that the minimum}\\
&\qquad \text{connectivity distance between each of these nodes and the}\\
&\qquad \text{nodes belonging to } V_\Gamma \text{ is lower than } d, d \in \mathbb{N}\}.
\end{aligned}
$$

In other words, the subset V_d corresponds to the neighboring area of depth d of the interface Γ. The subset V_d can also be interpreted as overlapping layers of the sub-domain $\Omega^{(q)}$ over the sub-domain $\Omega^{(s)}$, for $s = 1, 2$, $q = 1, 2$, $s \neq q$.

For the sake of clarity, an example is now detailed. For this purpose, a regular $\mathbb{P}_1$-finite element mesh of the domain Ω is considered with triangles, and this mesh is partitioned into two sub-domains with one interface. Fig. 2 represents the local numbering of the nodes in the sub-domain $\Omega^{(2)}$. On this example the subsets of indexes are obtained:

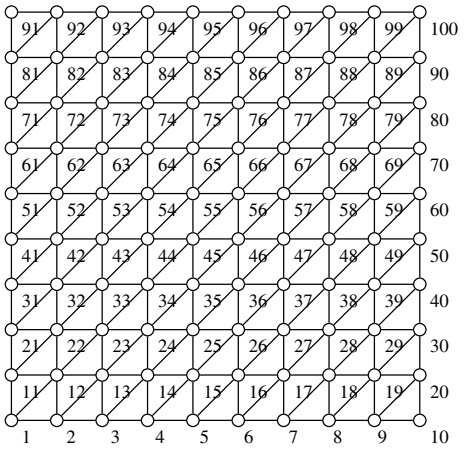

Fig. 2. Local numbering of the nodes in the sub-domain $\Omega^{(2)}$.

$$V = \{1, \ldots, 100\},$$

$$V_\Gamma = \{1, 11, 21, 31, 41, 51, 61, 71, 81, 91\},$$

$$V_1 = V_\Gamma \cup \{2, 12, 22, 32, 42, 52, 62, 72, 82, 92\},$$

$$V_2 = V_1 \cup \{3, 13, 23, 33, 43, 53, 63, 73, 83, 93\},$$

$$V_3 = V_2 \cup \{4, 14, 24, 34, 44, 54, 64, 74, 84, 94\},$$

$$V_4 = V_3 \cup \{5, 15, 25, 35, 45, 55, 65, 75, 85, 95\}$$

and are represented in Fig. 3.

Since the condensation procedure involved in the computation of the neighbor Schur complement represents the interactions of all the nodes inside the neighboring sub-domains and on the interface, it may be interesting to limit these interactions only to a part of the neighboring sub-domain. The approximation considered here consists of approximating the neighbor Schur complement $K_{pp}^{(s)} - K_{pi}^{(s)}[K_{ii}^{(s)}]^{-1}K_{ip}^{(s)}$ with the matrix $K_{pp}^{(s)} - K_{pj}^{(s)}[K_{jj}^{(s)}]^{-1}K_{jp}^{(s)}$ for $s = 1, 2$, where j denotes the nodes belonging to $V_d \backslash V_\Gamma$, $d \in \mathbb{N}$. The complete algorithm to build a dense approximation of the neighbor Schur complement matrix is formulated below.

Algorithm 4.1 (*dense approximation*). The complete algorithm to compute the augmented matrix $A_{pp}^{(q)}$ for $q = 1, 2$ can be defined as the following steps:

(1) Initialization of d
(2) Construction of the dense matrix structure of the interface matrix $A_{pp}^{(q)} \in \mathbb{R}^{\dim V_\Gamma \times \dim V_\Gamma}$.

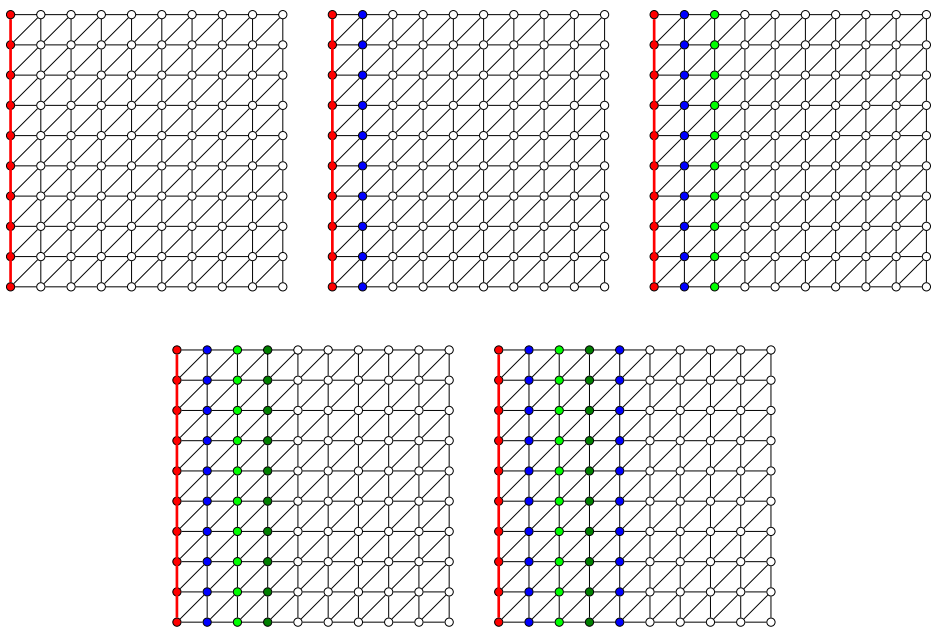

Fig. 3. Nodes of the sub-domain $\Omega^{(2)}$. From the left to the right and from the top to the bottom, the interface nodes, the interface nodes with one, two, three and four layers respectively.

(3) Construction of the sparse matrix structure of the sub-domain matrix $K^{(s)} \in \mathbb{R}^{\dim V \times \dim V}$.
(4) Assembly of the matrix $K^{(s)}$.
(5) Extraction of the coefficients $K^{(s)}_{mn}$, $(m,n) \in V_d \times V_d$, and construction of the sparse matrix $\widetilde{K}^{(s)} \in \mathbb{R}^{\dim V_d \times \dim V_d}$ with these coefficients.
(6) Computation of the matrix $\widetilde{S}^{(s)}$ by condensation of the matrix $\widetilde{K}^{(s)}$ on the interface V_Γ.
(7) Set $A^{(q)}_{pp} := \widetilde{S}^{(s)}$.

where $s = 1, 2$, $s \neq q$. The matrix $A^{(q)}_{pp}$ is then added to the sub-domain matrix $K^{(q)}$.

In this algorithm, the condensation procedure is only performed for the nodes belonging to V_d. This was not the case for the computation of the neighbor Schur complement matrix where the condensation procedure was performed for all the nodes belonging to V. This difference reduces significantly the computational cost.

Unfortunately, because the matrix $A^{(q)}_{pp}$, $\forall q$ is dense, adding it to the local matrix of the sub-domain $\Omega^{(q)}$ increases the bandwidth of the latest. This leads to an additional cost during the solution inside the sub-domains at each iteration. For this reason, an approach based on a sparse approximation is investigated in the following section.

Remark 4.1. In the previous example, $\mathbb{P}_1$-finite element have been considered for the sake of clarity. When working in linear elasticity, three degrees of freedom are associated to each node. As a consequence, the set of degree of freedom associated to the nodes should be considered in place of the set of nodes used in the Algorithm 4.1.

4.4. Sparse approximation

In the previous section, the degrees of freedom of the sub-domain belonging to a small area have been condensed on the interface. In this section, the condensation will be performed only on a small part of the interface, called patch [21]. For this purpose, new subsets of indexes are defined for the nodes of the sub-domain $\Omega^{(s)}$, $s = 1, 2$:

$$
\begin{aligned}
V &= \{\text{indexes of nodes inside the sub-domain and on the interface}\},\\
V_\Gamma &= \{\text{indexes of nodes on the interface } \Gamma\},\\
V^j_p &= \{\text{indexes of nodes belonging to } V_\Gamma \text{ such that the minimum}\\
&\qquad \text{connectivity distance between each of these nodes and the}\\
&\qquad \text{node labelled } j \text{ is lower than } p, p \in \mathbb{N}\},\\
V^j_{p,d} &= \{\text{indexes of nodes belonging to } V \text{ such that the minimum}\\
&\qquad \text{connectivity distance between each of these nodes and the}\\
&\qquad \text{nodes belonging to } V^j_p \text{ is lower than } d, d \in \mathbb{N}\}.
\end{aligned}
$$

In other words, the subset V^j_p corresponds to a patch of radius p around the node labelled j, and the subset $V^j_{p,d}$ corresponds to a neighboring area of depth d of the patch.

Considering again the example of a regular mesh with $\mathbb{P}_1$-finite elements leads for the node $j := 21$, a value of $p := 1$ and a value of $d := 1, \ldots, 4$, to the following subsets of indexes:

$$V_1^{21} = \{11, 21, 31\},$$
$$V_{1,1}^{21} = V_1^{21} \cup \{1, 22, 32, 41, 42\},$$
$$V_{1,2}^{21} = V_{1,1}^{21} \cup \{2, 13, 23, 33, 43, 51, 52, 53\},$$
$$V_{1,3}^{21} = V_{1,2}^{21} \cup \{3, 14, 24, 34, 44, 54, 61, 62, 63, 64\},$$
$$V_{1,4}^{21} = V_{1,3}^{21} \cup \{4, 15, 25, 35, 45, 55, 65, 71, 72, 73, 74, 75\},$$

which are shown in Fig. 4.

The sparse approximation consists of defining a sparse augmented matrix in order not to increase the bandwidth of the sub-domain matrix. This augmented matrix is obtained by local condensation along the interface, and extraction of some coefficients. The complete algorithm is described below.

Algorithm 4.2 (*sparse approximation*). The complete algorithm to compute the augmented matrix $A_{pp}^{(q)}$ for $q = 1, 2$ can be defined as the following steps:

(1) Initialization of p and d
(2) Construction of the sparse matrix structure of the interface matrix $A_{pp}^{(q)} \in \mathbb{R}^{\dim V_\Gamma \times \dim V_\Gamma}$.
(3) Construction of the sparse matrix structure of the sub-domain matrix $K^{(s)} \in \mathbb{R}^{\dim V \times \dim V}$.
(4) Assembly of the matrix $K^{(s)}$.
(5) do for all j in V_Γ
   (a) Extraction of the coefficients $K_{mn}^{(s)}$, $(m, n) \in V_{p,d}^j \times V_{p,d}^j$, and construction of the sparse matrix $\widetilde{K}^{(s)} \in \mathbb{R}^{\dim V_{p,d}^j \times \dim V_{p,d}^j}$ with these coefficients.
   (b) Computation of the matrix $\widetilde{S}^{(s)}$ by condensation of the matrix $\widetilde{K}^{(s)}$ on the patch V_p^j.
   (c) Extraction of the coefficients of the line associated with the node j from the matrix $\widetilde{S}^{(s)}$ and insertion inside the matrix $A_{pp}^{(q)}$ at the line associated with the node j.

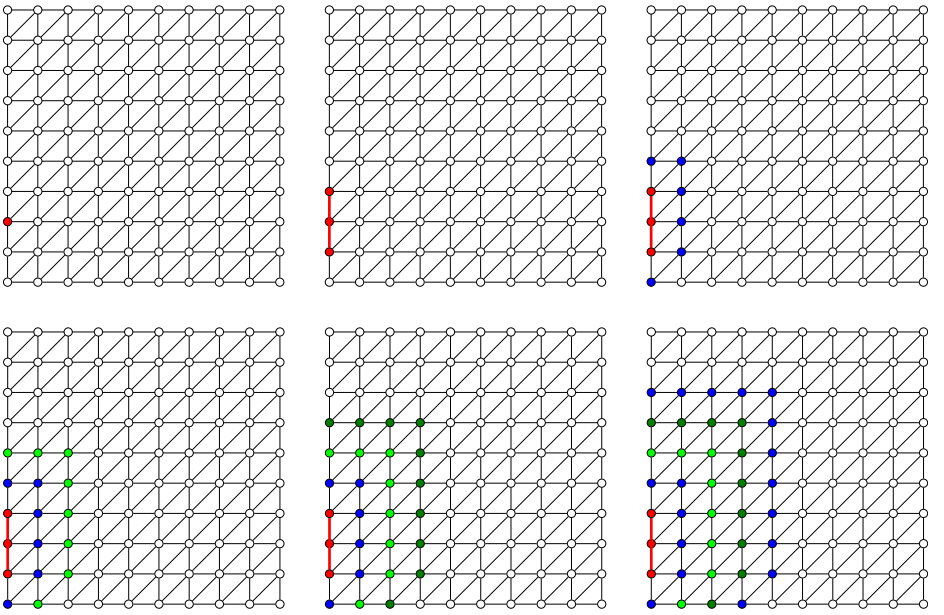

Fig. 4. Nodes of the sub-domain $\Omega^{(2)}$. From the left to the right and from the top to the bottom, one interface node, the associated patch with a radius one, this patch and a depth equal to one, two, three and four respectively.

(6) end do
(7) Construction of the symmetric matrix $A_{pp}^{(q)} = \frac{1}{2}(A_{pp}^{(q)^{\mathrm{T}}} + A_{pp}^{(q)})$,

where $s = 1, 2$, $s \neq q$. The matrix $A_{pp}^{(q)}$ is then added to the sub-domain matrix $K^{(q)}$.

Notice that the augmented matrix issue from the steps (1)–(6) of the previous algorithm is not symmetric anymore which may be an important drawback. The step (7) of the algorithm has been added in order to symmetrize the matrix $A_{pp}^{(q)}$.

4.5. Lumped approximation of the DtN operator

Following the methodology described in the previous sections, the interactions of the neighboring sub-domains are now limited to the direct contribution of the sub-domains nodes with the interface nodes [18,20]. This approach consists of approximating the neighbor Schur complement matrix $K_{pp}^{(s)} - K_{pi}^{(s)}[K_{ii}^{(s)}]^{-1}K_{ip}^{(s)}$ with its first term, i.e. the stiffness matrix $K_{pp}^{(s)}$, $s = 1, 2$. Such an approximation, presents the advantage to be very easy to implement since this matrix is already computed by the sub-domain during the assembly procedure and the integration of the contribution of the interface nodes.

5. Asymptotic and spectral analysis

5.1. Model problem

In this section, an analysis of the proposed techniques is presented. For this purpose, a two dimensional beam of length L and high l submitted to flexion is considered, as represented in Fig. 5. The Poisson's ratio v and the Young's modulus E are respectively $v := 0.3$ and $E := 2.0 \times 10^5$ N m^{-2}. Homogeneous Dirichlet boundary conditions are imposed on the left and homogeneous Neumann boundary conditions are set on the top and on the bottom. Loading, model as non-homogeneous Dirichlet boundary conditions are imposed on the right of the structure. The beam is meshed with triangular elements and discretized with $\mathbb{P}_1$-finite elements. The mesh is then split into two sub-domains ($N_s := 2$). The condensed interface problem is solved iteratively with a Krylov method namely ORTHODIR [22]. The stopping criterion is $\|r_n\|_2 < 10^{-6}\|r_0\|_2$, where r_n and r_0 are the nth and initial global residuals.

5.2. Spectral analysis

Fig. 6 represents the spectral density of the eigenvalues of the matrix F of the linear system (17) for different choice of the augmented matrices $A_{pp}^{(s)}$ for $s = 1, 2$. An augmented matrix equal to the neighbor Schur complement will lead to eigenvalues equal to one which correspond to a spectral density equal to a Dirac function. Indeed, with such a choice, the matrix F is equal to identity. It can be noticed in Fig. 6 that a dense approximation performed with a number of layers equal to four leads to a spectral density very close to a

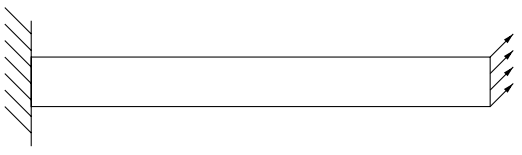

Fig. 5. Geometry of the beam.

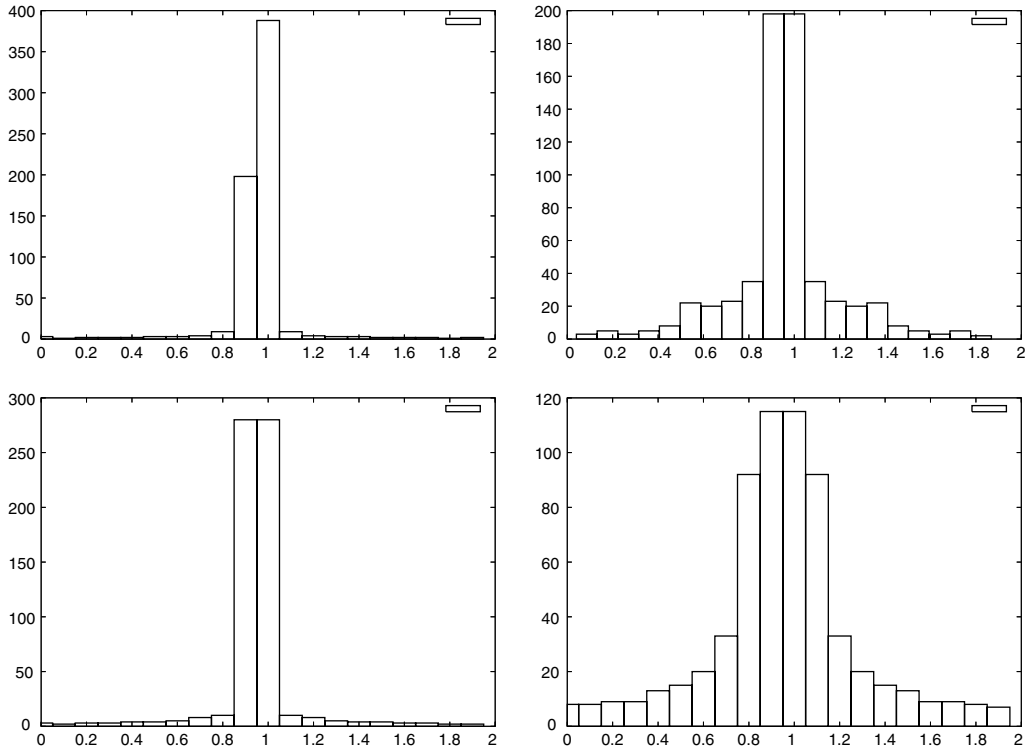

Fig. 6. Spectral density of the condensed interface problem for different augmented matrices. On the top-left a dense approximation with four layers, on the top-right a sparse mask approximation, on the bottom-left a sparse approximation with four layers, and on the bottom-right a lumped approximation. ($L := 10$, $l := 1$, $h := 1/160$, $N_s := 2$.)

Dirac function. Similar spectral density is obtained for a sparse approximation performed with a number of layers equal to four. This property can be explained by the fact that these two approximations are based on local condensation and thus have the behavior of a Dirichlet-to-Neumann operator. These dense and sparse approximations can thus be considered as 'good' approximation of the Schur complement. On the contrary, a sparse mask approximation or a lumped approximation of the Schur complement lead to a spread spectrum.

5.3. Asymptotic behavior upon the number of sub-domains

The asymptotic analysis upon the number of sub-domains is now investigated. The respective influence of the choice of the augmented matrices on the number of iterations required by the iterative algorithm are reported in Fig. 7. The quality of the different approximations can be classified as follow: the neighbor Schur complement, the dense approximation, the sparse approximation and the lumped approximation. Surprising the sparse mask approximation leads to excellent asymptotic results, even better than a dense approximation.

5.4. Asymptotic behavior upon the mesh size

The asymptotic behavior of the proposed techniques upon the mesh size is now analyzed and the results are reported in Fig. 8. Four layers are considered for the dense and for the sparse approximation. A linear

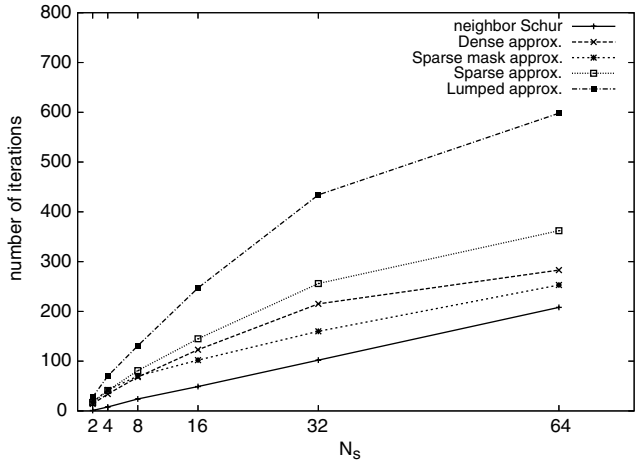

Fig. 7. Asymptotic behavior for different augmented matrices and different number of sub-domain. ($L := 64$, $l := 1$, $h := 1/20$.)

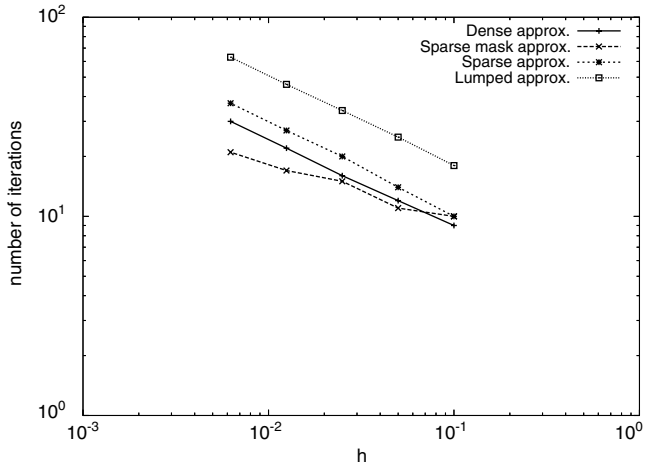

Fig. 8. Asymptotic behavior for different augmented matrices and different mesh size. ($L := 10$, $l := 1$, $N_s := 2$.)

dependance upon the mesh size can be noticed for all the methods and the respective efficiency of the approximations can be classified as follow: first the dense approximation, second the sparse approximation and finally the lumped approximation. Again the sparse mask approximation leads to good results.

Fig. 9 shows the behavior of the proposed techniques when the number of layers of the dense and of the sparse approximation increases proportionally with the mesh size. A linear dependance still occurs for the sparse approximation but the slope of the curve is lower than with a constant number of layers equal to four. A very weak dependance is obtained for the dense approximation.

5.5. Asymptotic behavior upon the number of layers

The asymptotic behavior of the dense and of the sparse approximation is analyzed for different number of layers and the results are reported in Table 1. This table shows that increasing the number of layers decreases significantly the number of iterations.

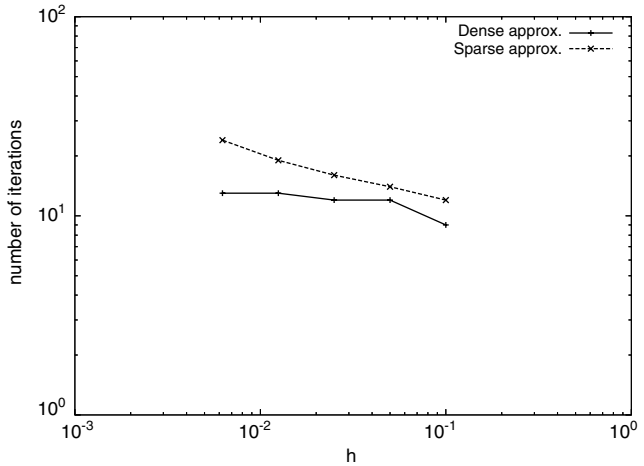

Fig. 9. Asymptotic behavior for different augmented matrices and different mesh size when the number of layers is proportional to the mesh size. ($L := 10$, $l := 1$, $N_s := 2$.)

Table 1
Number of iterations for different number of layers for the dense and for the sparse approximation for the beam problem ($L := 10$, $l := 1$, $h := 1/40$, $N_s := 2$)

Number of layers	Dense approximation	Sparse approximation
2	20	23
4	16	20
8	12	16
16	10	15
32	7	13

Since a mesh partitioning into two sub-domains is considered here, increasing the number of layers to the size of the sub-domains for the dense approximation makes the augmented matrix equal to the neighbor Schur complement matrix. The optimal convergence into one iteration is achieved, which confirms the theoretical analysis.

6. Numerical experiments

6.1. Illustration of the optimal convergence

The first example illustrates the theoretical optimal convergence property of the domain decomposition algorithm when using the complete outer Schur complement matrix as an augmented matrix. The beam of dimension $L := 10$ and $l := 1$ is meshed with a mesh size h equal to $1/20$ and the domain is decomposed into 10 square sub-domains in the length direction. The sub-domains are numbered from the right to the left with an index $s = 1, \ldots, 10$. The Jacobi algorithm is used for the solution of the interface condensed problem. Fig. 10 illustrates the iterative solution process obtained upon the iteration number with the stopping criterion $\|r_n\|_2 < 10^{-6}\|r_0\|_2$, where r_n and r_0 are the nth and initial global residuals. As expected by the theory, at each iteration step i the sub-domain $i + 1$ got a displacement field equal to the restriction of the exact solution of the discrete problem (2). This displacement field will not change anymore during the other iterations.

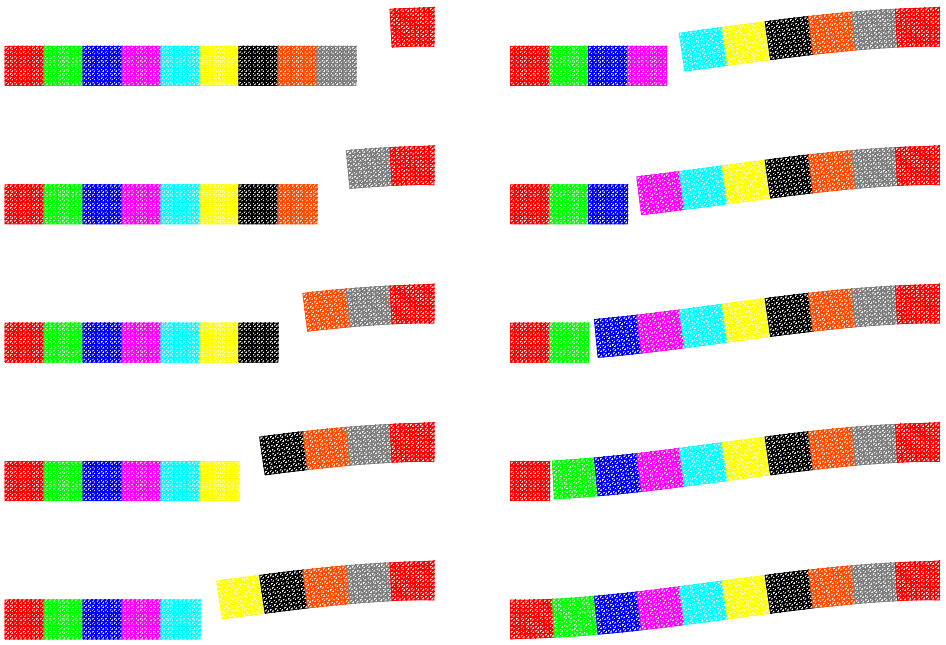

Fig. 10. Displacement field of the beam upon the iterations of the Jacobi algorithm for an augmented matrix equal to the complete outer Schur complement in the case of 10 sub-domains. From the top to the bottom and from the left to the right the displacement field at the iterations 1–10.

6.2. Dependance upon the number of layers

The dependance of the methods upon the number of layers is now analyzed. The results reported in Table 2 clearly shows that increasing the number of layers improves the quality of the approximation. Of course, increasing this number of layers will increase the CPU time to compute the augmented matrix $A_{pp}^{(s)}$ to be added to the stiffness matrix $K^{(s)}$. But this computation is performed only one time.

6.3. Dependance upon the mesh size

A number of layers equal to four is now considered and the dependance of the methods upon the mesh size is analyzed. Table 3 shows the number of iterations for different mesh size. A linear dependance can be noticed for all the methods, and once again a sparse mask approximation leads to better results than a dense approximation, than a sparse approximation, and than a lumped approximation.

Table 2
Number of iterations for different augmented matrices and different number of layers for the beam problem ($L := 10$, $l := 1$, $h := 1/40$, $N_s := 10$)

Number of layers	Dense approximation	Sparse approximation
2	77	91
4	61	76
8	46	65
16	33	56
32	29	48

Table 3
Number of iterations for different augmented matrices and different mesh size parameter for the beam problem ($L := 10$, $l := 1$, $N_s := 10$)

Mesh size	Neighbor Schur	Dense approximation	Sparse mask approximation	Sparse approximation	Lumped approximation
1/10	29	29	36	39	69
1/20	29	44	38	56	94
1/40	29	61	45	76	130
1/80	29	83	56	106	181
1/160	29	114	73	147	251

Table 4
Number of iterations for different augmented matrices and different mesh size parameter for a number of layers increasing proportionally with the mesh size for the beam problem ($L := 10$, $l := 1$, $N_s := 10$)

Mesh size	Number of layers	Dense approximation	Sparse approximation
1/10	2	40	48
1/20	4	44	56
1/40	8	46	65
1/80	16	47	79
1/160	32	43	98

If a number of layers proportional to the mesh size is considered, a weak dependance can be noticed for the dense approximation. A under-linear dependance occurs for the sparse approximation, as reported in Table 4.

6.4. Performance of a two-level preconditioner

The rigid body motions used in [8] have two goals. The first one is to determine the pseudo-inverse of the sub-domains matrices, and the second one is to accelerate the convergence of the iterative algorithm by using a global communication procedure.

Because of the absorbing boundary conditions (18) and (19) defined on the interface between the sub-domains, the rigid body motions are no longer required to ensure the well-posedness of the sub-domains matrices. In order to improve the convergence of the algorithm, a two-level preconditioner as described in [7] can be applied. Choosing as a coarse space basis the rigid body motions of the sub-domains matrices with the continuity boundary conditions (20) and (21) may be a good idea. Of course, it may seem strange to define new absorbing boundary conditions in order to avoid the rigid body motions and to compute

Table 5
Number of iterations for different augmented matrices and different mesh size for the beam problem ($L := 10$, $l := 1$, $N_s := 10$)

Mesh size	Without a two-level preconditioner			With a two-level preconditioner		
	Neighbor Schur	Dense approximation	Lumped approximation	Neighbor Schur	Dense approximation	Lumped approximation
1/10	29	29	69	2	2	7
1/20	29	44	94	2	3	9
1/40	29	61	130	2	5	12
1/80	29	83	181	2	7	16
1/160	29	114	251	2	10	20

Table 6
Number of iterations for different augmented matrices and different number of sub-domains for the beam problem ($L := 64$, $l := 1$, $h := 1/20$)

Number of sub-domains	Without a two-level preconditioner			With a two-level preconditioner		
	Neighbor Schur	Dense approximation	Lumped approximation	Neighbor Schur	Dense approximation	Lumped approximation
2	1	14	28	1	3	5
4	8	34	70	1	3	5
8	24	68	131	1	3	6
16	49	123	247	1	3	7
32	102	215	434	1	3	9
64	208	283	598	1	3	13

these rigid body motions exactly as in the past in order to add them in the coarse space basis. This approach is only shown to illustrate the performance of the two-level preconditioner. Later on, different choices of the coarse space basis will be considered, an issue under current investigation.

The key point of this two-level preconditioner is that the rigid body motions are incorporated inside the coarse space only and are not used to compute the pseudo-inverse. The implementation is much easier. The results shown in Table 5 for different mesh size and in Table 6 for different number of sub-domains, confirm the efficiency of this two-level preconditioner. A total number of four layers has been used for the dense approximation.

7. Conclusion

In this paper, several techniques for the algebraic approximation of Dirichlet-to-Neumann maps for linear elasticity have been presented. These new and original techniques are based on the local condensation of the degrees of freedom belonging to a small area—defined inside the sub-domain—on a small patch—defined on the interface. This condensation procedure can be assimilated to the computation of a small Dirichlet-to-Neumann map.

In a domain decomposition method context, the absorbing boundary conditions defined on the interface between the sub-domains are of major importance. Using the algebraic approximation of the Dirichlet-to-Neumann map presented in this paper, helps to achieve fast and robust convergence of the domain decomposition algorithm. Further improvements of the algorithm are obtained with an additional coarse space preconditioner.

An asymptotic, spectral and numerical analysis of the presented techniques are investigated and described for several configurations of a linear elasticity problem. The results presented in the numerical experiments section report impressive convergence and fantastic robustness of the domain decomposition method equipped with such absorbing boundary conditions. Furthermore, since these techniques are based on an algebraic approach, they are very easy to implement in existing software without the information of the topology of the interface.

References

[1] J.-D. Benamou, B. Després, A domain decomposition method for the Helmholtz equation and related optimal control problems, J. Comput. Phys. 136 (1997) 68–82.
[2] P. Chevalier, F. Nataf, Symmetrized method with optimized second-order conditions for the Helmholtz equation, Contemp. Math. 218 (1998) 400–407.

[3] F. Collino, S. Ghanemi, P. Joly, Domain decomposition method for harmonic wave propagation: a general presentation, Comput. Methods Appl. Mech. Engrg. 184 (2000) 171–211.
[4] B. Engquist, A. Majda, Absorbing boundary conditions for the numerical simulation of waves, Math. Comput. 31 (139) (1977) 629–651.
[5] B. Engquist, H.K. Zhao, Absorbing boundary conditions for domain decomposition, Appl. Numer. Math. 27 (1998) 341–365.
[6] C. Farhat, P.S. Chen, F.-X. Roux, The dual Schur complement method with well-posed local Neumann problems: regularization with a perturbed Lagrangian formulation, SIAM J. Sci. Stat. Comput. 14 (1993) 752–759.
[7] C. Farhat, A. Macedo, M. Lesoinne, F.-X. Roux, F. Magoulès, A. de la Bourdonnaye, Two-level domain decomposition methods with Lagrange multipliers for the fast iterative solution of acoustic scattering problems, Comput. Methods Appl. Mech. Engrg. 184 (2) (2000) 213–240.
[8] C. Farhat, F.X. Roux, Implicit parallel processing in structural mechanics, Comput. Mech. Adv. 2 (1994) 1–124.
[9] M. Gander, L. Halpern, F. Nataf, Optimal convergence for overlapping and non-overlapping Schwarz waveform relaxation, in: Proceedings of the 11th international symposium on Domain Decomposition Methods, 1999.
[10] M. Gander, F. Magoulès, F. Nataf, Optimized Schwarz methods without overlap for the Helmholtz equation, SIAM J. Scient. Comput. 24 (1) (2002) 38–60.
[11] S. Ghanemi, A domain decomposition method for Helmholtz scattering problems, in: P.E. Bjørstad, M. Espedal, D. Keyes (Eds.), Ninth International Conference on Domain Decomposition Methods, 1997, pp. 105–112.
[12] D. Givoli, High-order non-reflecting boundary conditions without high-order derivatives, J. Comput. Phys. 170 (2001) 849–870.
[13] T. Hagstrom, Radiation boundary conditions for the numerical simulation of waves, Acta Numer. 8 (1999) 47–106.
[14] L. Halpern, Absorbing boundary conditions for the discretization schemes of the one dimensional wave equation, Math. Comput. 38 (1982) 415–429.
[15] P. Joly, O. Vacus, Sur l'analyse des conditions aux limites absorbantes pour l'équation de Helmholtz, INRIA, rapport de recherche, 1996, pp. 28–50.
[16] F. Magoulès, P. Ivànyi, B.H.V. Topping, Non-overlapping Schwarz methods with optimized transmission conditions for the Helmholtz equation, Comput. Methods Appl. Mech. Engrg. 193 (45–47) (2004) 4797–4818.
[17] F. Magoulès, P. Ivànyi, B.H.V. Topping, Convergence analysis of Schwarz methods without overlap for the Helmholtz equation, Comput. Struct. 82 (2004) 1835–1847.
[18] F. Magoulès, F.-X. Roux, S. Salmon, Optimal discrete transmission conditions for a non-overlapping domain decomposition method for the Helmholtz equation, SIAM J. Scient. Comput. 25 (5) (2004) 1497–1515.
[19] F. Nataf, F. Rogier, E. de Sturler, Optimal interface conditions for domain decomposition methods, CMAP (Ecole Polytechnique), 1994.
[20] F.-X. Roux, F. Magoulès, S. Salmon, L. Series, Optimization of interface operator based on algebraic approach, in: 14th International Conference on Domain Decomposition Methods in Cocoyoc, Mexico, 2002, pp. 297–304.
[21] F.-X. Roux, F. Magoulès, L. Series, Y. Boubendir, Approximation of Optimal Interface Boundary Conditions for Two-Lagrange Multiplier FETI Method. Lecture Notes in Computational Science and Engineering, Springer-Verlag, 2004, pp. 283–290.
[22] Y. Saad, Iterative Methods for Linear Systems, PWS Publishing, Boston, 1996.
[23] S.V. Tsynkov, Numerical solution of problems on unbounded domains. A review, Appl. Numer. Math. 27 (1998) 465–532.