



HAL
open science

A Coarse Space Construction Based on Local DtN Maps

Frédéric Nataf, Hua Xiang, Victorita Dolean

► **To cite this version:**

Frédéric Nataf, Hua Xiang, Victorita Dolean. A Coarse Space Construction Based on Local DtN Maps. 2010. hal-00491919v1

HAL Id: hal-00491919

<https://hal.science/hal-00491919v1>

Preprint submitted on 14 Jun 2010 (v1), last revised 9 Feb 2011 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Coarse Grid Space Construction Based on Local DtN Maps

Frédéric Nataf* Hua Xiang† Victorita Dolean‡

May 24, 2010

Abstract

Coarse grid correction is a key ingredient in order to have scalable domain decomposition methods. In this work we construct the coarse grid space using the low frequency modes of the subdomain DtN maps, and apply the obtained two-level preconditioner (the additive Schwarz method together with the new coarse grid) to the extended or the original linear system arising from an overlapping domain decomposition. The patch method is also added to further improve the convergence. Our method is suitable for the parallel implementation and its efficiency is demonstrated by numerical examples on problems with high heterogeneities for both manual and automatic partitionings.

Some notations and definitions

A	coefficient matrix of the linear system $Ax = b$
Z, Y	full rank matrices which span the coarse grid subspaces.
E	$E = Y^T AZ$, Galerkin matrix or coarse-grid matrix
Ξ	$\Xi = ZE^{-1}Y^T$, coarse-grid correction matrix in MG and DDM
P_D	$P_D = I - A\Xi = I - AZ(Y^T AZ)^{-1}Y^T$
Q_D	$Q_D = I - \Xi A = I - Z(Y^T AZ)^{-1}Y^T A$
P_{BNN}	$P_{BNN} = Q_D M^{-1} P_D + ZE^{-1}Y^T$
P_{ADEF2}	$P_{ADEF2} = Q_D M^{-1} + ZE^{-1}Z^T$

1 Introduction

We consider the solution of the linear system $Ax = b \in \mathbb{R}^p$ arising from the discretization of an elliptic boundary value problem of the type

$$\eta u - \operatorname{div}(\kappa \nabla u) = f \quad (1)$$

where κ is the diffusion tensor which can be discontinuous. When using an iterative method in a one-level domain decomposition framework, we can meet

*Laboratoire J.L. Lions, CNRS UMR 7598, Université Pierre et Marie Curie, 175 rue du Chevaleret, 75013 Paris, France.

†School of Mathematics and Statistics, Wuhan University, Wuhan 430072, P. R. China.

‡Laboratoire J.-A. Dieudonné, CNRS UMR 6621, Université de Nice-Sophia Antipolis, Parc Valrose, 06108 Nice Cedex 02, France

a long stagnation or a slow convergence, especially when the number of subdomains is large. Even when $\kappa = 1$, the convergence of a one-level domain decomposition method presents a long plateau of stagnation. Its length is related to the number of subdomains of the decomposition in one direction. For example, we know that for the problem divided into N subdomains in a one-way partitioning, the convergence can never be achieved in less than $N - 1$ iterations since the exchange of information between the subdomains is reduced to their common interfaces. Thus, the global information transfers only from one subdomain to its neighbors [15, 18]. One needs a two-level method to have a scalable algorithm, i.e., an algorithm whose convergence rate is weakly dependent on the number of subdomains, see [26] and references therein.

Two-level domain decomposition methods are closely related to multigrid methods and deflation corrections, see [24] and references therein. These methods are defined by two ingredients: a full rank matrix $Z \in \mathbb{R}^{p \times m}$ with $m \ll p$ and an algebraic formulation of the correction. These techniques imply solving a reduced size problem of order $m \times m$ called a coarse grid problem. The space spanned by the columns of Z should ideally contain the vectors responsible for the stagnation of the one level method. We will come back later to the choice of Z in the framework of domain decomposition methods and focus now on the various algebraic ways to improve the convergence by using the coarse grid.

According to [24], for a domain decomposition method (DDM), one well-known coarse grid correction preconditioner is of the form $I + ZE^{-1}Z^T$, where $E = Z^T AZ$ is the coarse grid correction used to speed up the convergence. The abstract additive coarse grid correction proposed in [20] is $M^{-1} + ZE^{-1}Z^T$, where M^{-1} is the additive Schwarz preconditioner, a sum of local solvers in each subdomain, which can be implemented in parallel. The first term is a fine grid solver, and the second term represents a coarse solver. Hence it is called the two-level additive Schwarz preconditioner. The BPS preconditioner introduced by Bramble, Paschiak and Schatz [1] is of this type. In the context of domain decomposition methods, we mention the balancing Neumann-Neumann preconditioner and the FETI algorithm. They have been extensively investigated, see [26] and references therein. For symmetric systems the balancing preconditioner was proposed by Mandel [16]. The abstract balancing preconditioner [16] for nonsymmetric systems reads [8],

$$P_{BNN} = Q_D M^{-1} P_D + Z E^{-1} Z^T. \quad (2)$$

For the preconditioner P_{BNN} , if we choose the initial approximation $x_0 = \Xi b$, then the action of P_D is not required in practice, see [26, p.48]. Note that the multigrid (MG) V(1,1)-cycle preconditioner P_{MG} is closely related to P_{BNN} . Choosing the proper smoother M in P_{BNN} , we can ensure that P_{MG} and P_{BNN} are SPD, and P_{MGA} and $P_{BNN}A$ have the same spectrum [25].

For a SPD system, by choosing $Y = Z$, the authors in [24] define

$$P_{ADEF2} = Q_D M^{-1} + Z E^{-1} Z^T, \quad (3)$$

which is as robust as P_{BNN} but usually less expensive [24]. The two-level hybrid Schwarz preconditioner in [23, p.48] has the same form as P_{ADEF2} . In this paper we will mainly use P_{ADEF2} , but for the nonsymmetric cases.

Given the coarse grid subspace, we can construct the two-level preconditioners above. An effective two-level preconditioner is highly dependent on the

choice of coarse grid subspace. We focus now on the choice of the coarse space Z in the context of domain decomposition methods for problems of type (1) with heterogeneous coefficients. In this case, the coarse space is made of vectors with support in one subdomain or one subdomain and its neighbors. When the jumps in the coefficients are inside the subdomains (and not on the interface) or across the interface separating the subregions, the use of a fixed number of vectors per subdomain in Z gives good results, see [6], [17], [21], [22], [4] and [5]. When the discontinuities are along the interfaces between the subdomains, results are not so good.

Here, we propose to construct a coarse subspace such that the two-level method is robust with respect to heterogeneous coefficients for an arbitrary domain decomposition. Even if the discontinuities are along the interface like for instance in with layered coefficients and a one-way decomposition in the orthogonal direction (see § 4.1, the iteration counts are very stable with respect to jumps of the coefficients. Our method is based on the low-frequency modes associated with the Dirichlet-to-Neumann (DtN) map on each subdomain. After obtaining the eigenvectors associated with the small eigenvalues of DtN, we use the harmonic extension to the whole subdomain. It is quite efficient even for the problem with large discontinuities in the coefficients. Moreover, it is suitable for the parallel implementation. We apply such a two-level preconditioner to the original linear system (4) and to the extended one (5) arising from the domain decomposition method.

Using the interface conditions is another strategy to accelerate the convergence [18]. One can tune the interface conditions for a given problem to improve the convergence speed of the iterative solution method. In this paper we use the patch method, see [15].

The paper is organized as follows. In Section 2, we introduce the two-level preconditioners using the additive Schwarz (AS) or the restricted additive Schwarz (RAS) with the coarse grid correction. The construction of coarse grid spaces is presented in Section 3. In Section 4 numerical tests on the model problem demonstrate the efficiency of our method; in § 4.3 the patch method is used to further improve the convergence. Some conclusion remarks are given in Section 5.

2 Algebraic Domain Decomposition Methods

Without loss of generality, we consider here a decomposition of a domain Ω into two overlapping subdomains Ω_1 and Ω_2 . The overlap is denoted by $O := \Omega_1 \cap \Omega_2$. This yields a partition of the domain: $\bar{\Omega} = \bar{\Omega}_I^{(1)} \cup \bar{O} \cup \bar{\Omega}_I^{(2)}$ where $\Omega_I^{(i)} := \Omega_i \setminus \bar{O}$, $i = 1, 2$. At the algebraic level this corresponds to a partition of the set of indices \mathcal{N} into three sets: $\mathcal{N}_I^{(1)}$, O and $\mathcal{N}_I^{(2)}$.

After the discretization of the BVP (1) defined in domain Ω , we obtain a linear system of the following form

$$A u := \begin{bmatrix} A_{II}^{(1)} & A_{IO}^{(1)} & \\ A_{OI}^{(1)} & A_{OO} & A_{OI}^{(2)} \\ & A_{IO}^{(2)} & A_{II}^{(2)} \end{bmatrix} \begin{bmatrix} u_I^{(1)} \\ u_O \\ u_I^{(2)} \end{bmatrix} = \begin{bmatrix} f_I^{(1)} \\ f_O \\ f_I^{(2)} \end{bmatrix}. \quad (4)$$

We can also define the extended linear system by considering twice the variables

is a need to choose a priori the coarse space. For instance in [19], Nicolaides defines the deflation subspace Z as follows

$$(z_j)_i = \begin{cases} 1, & \text{if } i \in \Omega_j, \\ 0, & \text{if } i \notin \Omega_j. \end{cases} \quad (8)$$

In [27, 28], the projection vectors z_i are chosen in a similar but more complicated way. Definition (8) is also used as the aggregation-based coarse level operator in AMG [9]. Originally (8) is used for disjoint sets, not for the overlapping case. In the following, we use it in the overlapping case as well. This coarse space performs well in the constant coefficient case. But when there are jumps in the coefficients, it cannot prevent stagnation in the convergence.

We propose now a construction of the coarse space that will be suitable for parallel implementation and efficient for accelerating the convergence for the problem with highly heterogeneous coefficients and arbitrary domain decompositions. We still choose Z such that it has the form

$$Z = \begin{bmatrix} w_1 & 0 & \cdots & 0 \\ \vdots & w_2 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & w_d \end{bmatrix}, \quad (9)$$

where d is the number of subdomains. The vectors w_i are based on the smallest eigenvectors associated with the DtN map in each subdomain Ω_i .

More precisely, let us consider first at the continuous level the Dirichlet to Neumann map DtN_{Ω_i} . Let $u_{\Gamma_i} : \Gamma_i \mapsto \mathbb{R}$,

$$\text{DtN}_{\Omega_i}(u_{\Gamma_i}) = \frac{\partial v}{\partial n_i} \Big|_{\Gamma_i},$$

where v satisfies

$$\begin{cases} \mathcal{L}(v) := (\eta - \text{div}(k\nabla))v = 0, & \text{in } \Omega_i, \\ v = u_{\Gamma_i}, & \text{on } \Gamma_i, \end{cases} \quad (10)$$

where Γ_i is the interface boundary. If the subdomain is not a floating one (i.e. $\partial\Omega_i \cap \partial\Omega \neq \emptyset$), it is necessary to add on this part of the boundary, the boundary condition coming from the original problem. To construct the coarse grid subspace, we use the low frequency modes associated with the DtN operator:

$$\text{DtN}_{\Omega_i}(w_i) = \lambda w_i$$

with λ small.

In order to motivate this choice of the coarse space, we write the original Schwarz method at the continuous level, where the domain Ω is decomposed in a one-way partitioning.

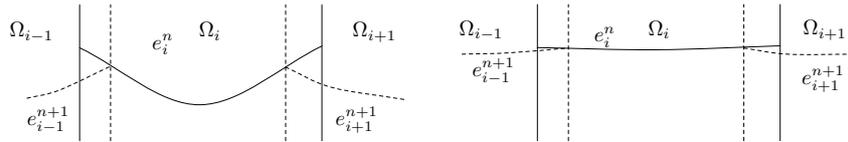


Figure 3: Fast or slow convergence of the Schwarz algorithm.

The error $e_i^n := u_i^n - u|_{\Omega_i}$ in subdomain Ω_i at step n of the algorithm satisfies:

$$\begin{aligned} \mathcal{L}(e_i^{n+1}) &= 0 && \text{in } \Omega_i, \\ e_i^{n+1} &= \sum_{j \neq i} \xi_i^j e_j^n && \text{on } \bar{\Omega}_i \cap \partial\Omega_j, \end{aligned}$$

where ξ_i^j is such that

$$\sum_{j \neq i} \xi_i^j = 1_{\partial\Omega_i}.$$

On the 1D example sketched in Figure 3, we see that the rate of convergence of the algorithm is related to the decay of the harmonic functions e_i^n in the vicinity of $\partial\Omega_i$. A fast decay corresponds to a large eigenvalue of the DtN map whereas a slow decay corresponds to small eigenvalues of this map. Thus the small eigenvalues of the DtN map are responsible for the slow convergence of the algorithm and it is natural to incorporate them in the coarse grid space.

To obtain the discrete form of the DtN map, we consider the variational form of (10). Let's define the bilinear form $a : H^1(\Omega_i) \times H^1(\Omega_i) \rightarrow \mathbb{R}$,

$$a(w, v) := \int_{\Omega_i} \eta w v + \kappa \nabla w \cdot \nabla v.$$

With a finite element basis $\{\phi_k\}$, the coefficient matrix of a Neumann boundary value problem in domain Ω_i is

$$A_{kl}^{(i)} = \int_{\Omega_i} \eta \phi_k \phi_l + \kappa \nabla \phi_k \cdot \nabla \phi_l.$$

Note that for the whole domain Ω , the coefficient matrix is given by

$$A_{kl} = \int_{\Omega} \eta \phi_k \phi_l + \kappa \nabla \phi_k \cdot \nabla \phi_l.$$

In problem (10), we use Dirichlet boundary condition: v is given on Γ_i by u_Γ . Let v_I denote the value of interior points:

$$A_{II} v_I + A_{I\Gamma} u_\Gamma = 0.$$

Hence, $v_I = -A_{II}^{-1} A_{I\Gamma} u_\Gamma$. The discrete counterpart of the normal derivative is given by

$$A_{\Gamma I} v_I + A_{\Gamma\Gamma}^{(i)} u_\Gamma.$$

Hence we get the discrete version of DtN in subdomain Ω_i ,

$$A_{\Gamma\Gamma}^{(i)} - A_{\Gamma I} A_{II}^{-1} A_{I\Gamma}.$$

Let t be an eigenvector of the DtN map corresponding to a small eigenvalue, we use the harmonic extension

$$w_i = \begin{pmatrix} -A_{II}^{-1} A_{I\Gamma} t \\ t \end{pmatrix}$$

to construct the coarse grid space (9). If the spectrum of the DtN map has several isolated small eigenvalues we take all of them in the coarse space. We denote by $N_{sm eig}$ the number of eigenvectors per subdomain we incorporate in

the coarse space. We call this procedure the Z_{D2N} method. We also use Z_{D2N} to denote the coarse grid space constructed by this method. Its construction is fully parallel. Similarly we call Z_{Nico} the method of Nicolaides or the corresponding coarse grid space. Let us remark that when $\eta = 0$ and the subdomain is a floating one, the lowest eigenvalue of the DtN map is zero and the corresponding eigenvector is a constant vector. Thus, if $N_{sm eig}=1$, Z_{Nico} and Z_{D2N} coincide. As we shall see in the next section, when a subdomain has several jumps of the coefficient, taking Z_{Nico} is not efficient and it is necessary to take Z_{D2N} with more than one vector per subdomain.

4 Numerical Results

In this section, we first consider in § 4.1 a one-way decomposition of the domain, then in § 4.2 an arbitrary decomposition and in § 4.3 the effect of changing the interface conditions.

4.1 One-way domain decomposition

In order to illustrate numerically the behavior of the coarse space, we first consider the following model problem:

$$\mathcal{L}u := \left(-\frac{\partial}{\partial x} c(y) \frac{\partial}{\partial x} - \frac{\partial}{\partial y} d(y) \frac{\partial}{\partial y} + \eta(y) \right) u(x, y) = f(x, y) \text{ in } (0, 1)^2.$$

We use Neumann BC on the top and the bottom, Dirichlet BC on the left

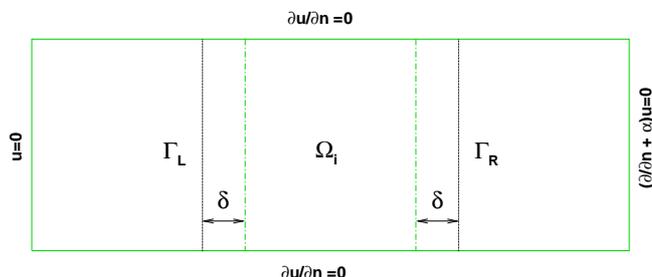


Figure 4: Subdomains and boundary conditions.

and $(\partial/\partial n + \alpha)u = 0, \alpha \ll 1$ on the right boundary, see figure 4. In our tests, we first consider the case with constant coefficients, $c(y) = d(y) = 1.0$. It is the case CONST in Table 2. We also test difficult cases with discontinuities in the coefficients. We consider the coefficient with discontinuity $c(y) = d(y) = \text{val}[10 * y]$, where val is an array to be defined and that depends on two parameters a and b . In our tests we take $a = 100000, b = 1$, and $\eta = 10^{-7}$. For instance, in the case HPL3 in Table 2, $\text{val} = [a \ a \ b \ b \ a \ a \ b \ b \ a \ a]$. So there are three high-permeability layers in the case HPL3.

In addition, we use the following parameters in our tests. n_{domain} represents the number of subdomains. $n_{overlap}$ stands for the overlap in x -direction; correspondingly the width of the overlap is $\delta = (n_{overlap} + 1)h$. Here $(n_{ypt} + n_{overlap})$

Case	Remarks
CONST	$c(y) = d(y) = 1.0$, constant coefficients
HPL1	$c(y) = d(y) = \text{val}[10 * y]$, $\text{val} = [b \ b \ b \ a \ a \ a \ a \ b \ b \ b]$
HPL2	$c(y) = d(y) = \text{val}[10 * y]$, $\text{val} = [b \ b \ a \ a \ b \ b \ a \ a \ b \ b]$
HPL3	$c(y) = d(y) = \text{val}[10 * y]$, $\text{val} = [a \ a \ b \ b \ a \ a \ b \ b \ a \ a]$
HPL4	$c(y) = d(y) = \text{val}[10 * y]$, $\text{val} = [a \ b \ b \ a \ b \ b \ a \ b \ b \ a]$
HPL5	$c(y) = d(y) = \text{val}[10 * y]$, $\text{val} = [b \ a \ b \ a \ b \ a \ b \ a \ b \ a]$

Table 2: Test cases. $b = 1, a = 100000$.

gives the number of grid points in x -direction in one subdomain, and n_{ypt} is the number of grid points in the y -direction. $N_{sm eig}$ stands for the number of eigenvectors associated small eigenvalues in each subdomain, which are used in constructing coarse spaces. These numerical tests are performed in MATLAB. We use the full GMRES as the linear solver, together with the preconditioner P_{ADEF2} , which is constructed by taking Z equal to Z_{Nico} or to Z_{D2N} .

4.1.1 The extended and the original systems

We compare the effect of the coarse grid correction when applied to the original system (4) (RAS method) or to the extended system (5) (AS method). We begin with the Nicolaidis coarse grid space. Note that the coarse subspace Z for RAS is chosen as

$$Z = \begin{bmatrix} 1_{\Omega_1^*} & 0 & 0 \\ 0 & 1_{\Omega_2^*} & 0 \\ 0 & 0 & 1_{\Omega_3^*} \end{bmatrix},$$

where $1_{\Omega_i^*}$ is a vector of all ones, and its length equals to the number of unknowns in Ω_i^* . Recall that the subdomains Ω_i^* are nonoverlapping as shown in Figure 2.

As expected when there is no coarse grid correction, RAS and AS perform similarly and have a plateau in the convergence curve. For the constant case in Figure 5, the piecewise coarse grid space of Nicolaidis is quite good for both the extended system and the original system. It works better on the extended system (5) (AS method) than on the original system (4) using RAS (21 vs. 32 iterations). When we use our method Z_{D2N} , the convergence is further improved. Since we used $N_{sm eig}=3$, the coarse space is then larger and this is no surprise. But Z_{Nico} is not good for the case with discontinuities. For this kind of problem, Z_{D2N} gives a much faster convergence, see Figure 6.

We can see that our Z_{D2N} method works well on both the extended system and the original system. It gives better convergence on the extended system. In the following, we focus on the extended system.

4.1.2 The extended system and Z_{D2N}

In this subsection, we further investigate our Z_{D2N} method applied to the extended system since it is the configuration that gives the best results. We test the robustness of the approach with respect to the various parameters of the problem: jumps of the coefficients, number of subdomains, mesh size, size of the overlap and anisotropy of the mesh.

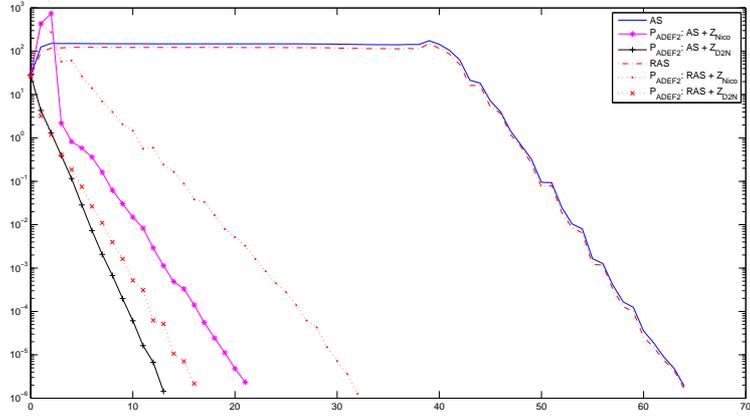


Figure 5: Case CONST, AS and RAS using coarse grids. $n_{domain}=16$, $n_{overlap}=1$, $n_{ypt}=8$, $n_{ypt}=16$, $N_{smeig}=3$. Both Z_{Nico} and Z_{D2N} work well for the constant case.

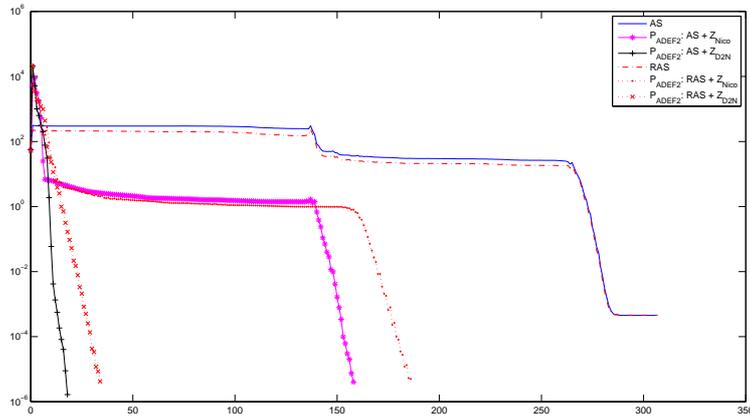


Figure 6: Case HPL2, AS and RAS with coarse grid correction. $n_{domain}=64$, $n_{overlap}=1$, $n_{ypt}=8$, $n_{ypt}=16$, $N_{smeig}=2$. Z_{Nico} does not work well, while Z_{D2N} gives a fast convergence.

Figure 7 shows that when using a coarse space the iteration numbers are almost constant as the number of subdomains increases. The three convergence curves with Z_{D2N} are difficult to distinguish.

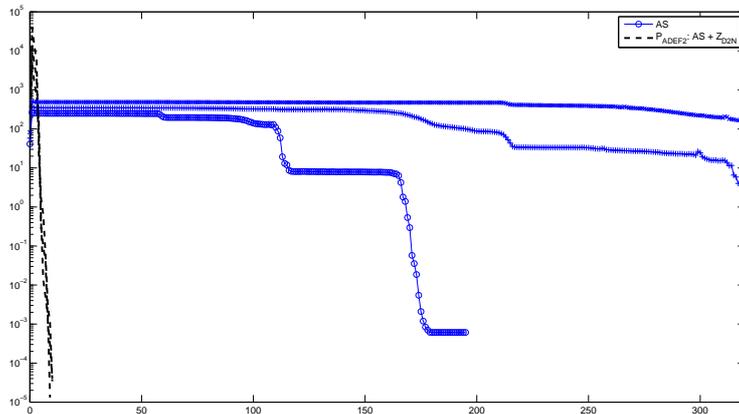


Figure 7: HPL3($a=100000$) and various number of subdomains. $N_{smeig}=3$, $n_{domain}=32, 64, 128$, $n_{overlap}=3$, $n_{ypt}=8$, $n_{ypt}=16$. Z_{D2N} : dotted lines, AS : solid lines.

We also consider the robustness with respect to the size of the jumps. We take as an example in table 3 the case HPL2 with $a = 1, 10^3, 10^5$ and 10^6 with the Z_{D2N} coarse space. We have 32 subdomains and $N_{smeig}=2$. We see that the iteration counts increase only slightly as the size of the jump in the coefficients increases by six orders of magnitude.

Jump	1	10^3	10^5	10^6
Iteration counts	10	11	14	18

Table 3: HPL2 with the Z_{D2N} coarse space for jumps in the coefficients ranging from 1 to 10^6 . $N_{smeig}=2$, $n_{domain}=32$, $n_{overlap}=3$, $n_{ypt}=8$, $n_{ypt}=16$.

In Figure 8, we plot the convergence curves for three successive refinements of the mesh. We see an increase in the number of iterations that can be related to the fact that the convergence rate of the Schwarz method depends on the physical size of the overlap. Here, we have three meshes of overlap and thus the physical size of the overlap decreases as the mesh is refined.

In Figure 9, we consider the test HPL1 with various sizes of overlap $2h$, $3h$ and $4h$ solved with the Z_{D2N} coarse space or without any coarse space (AS). For both methods, the iteration counts are improved as the overlap is increased. This is in agreement with the theory of Schwarz method.

We also consider the test HPL3 with $a = 100000$ and three mesh ratios h_x/h_y equals to 2., 1. and 1/2. In order to have a good convergence, we have to increase the size of the coarse space from N_{smeig} equals to three up to five. Another possibility is to apply a diagonal scaling to the matrix before building

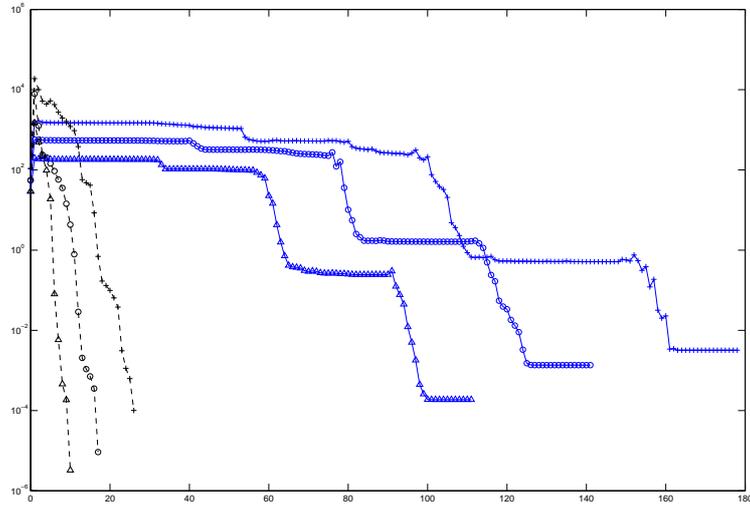


Figure 8: HPL3 with various mesh sizes: $1/64$ (cross), $1/32$ (circle) and $1/16$ (triangle), Z_{D2N} : dotted lines, AS solid lines. $N_{sm eig}=3$, $n_{domain}=32$.

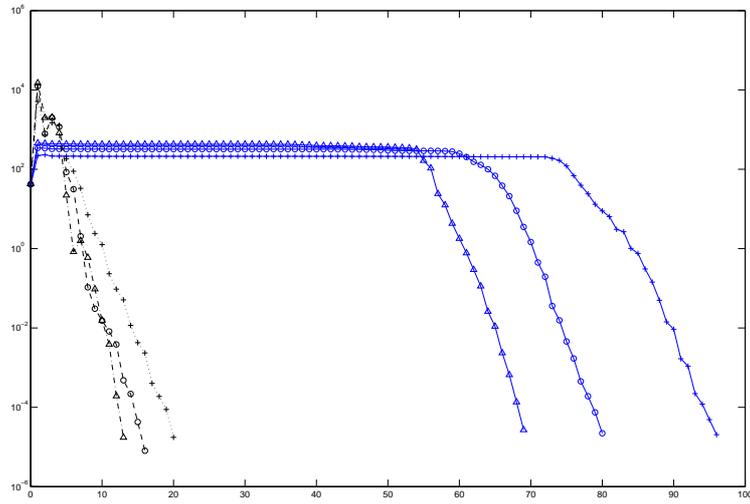


Figure 9: HPL1 with various overlaps: $2h$ (cross), $3h$ (circle) and $4h$ (triangle), Z_{D2N} : dotted lines, AS solid lines. $N_{sm eig}=1$, $n_{domain}=32$, $n_{ypt}=8$, $n_{ypt}=16$.

the domain decomposition method. This enables us to have a good convergence while keeping the number of small eigenvalues (N_{smeig}) equal to three which is the number of high permeability layers in the test HPL3.

For problems with jumps in the coefficients, N_{smeig} is important. If N_{smeig} is small, the coarse grid space is not large enough to capture the global information, and it still converges slowly. We can increase N_{smeig} , i.e., the size of the coarse grid space, to improve it (see Figure 10). When using a diagonal scaling the best choice is N_{smeig} equals to the number of layers, see [27].

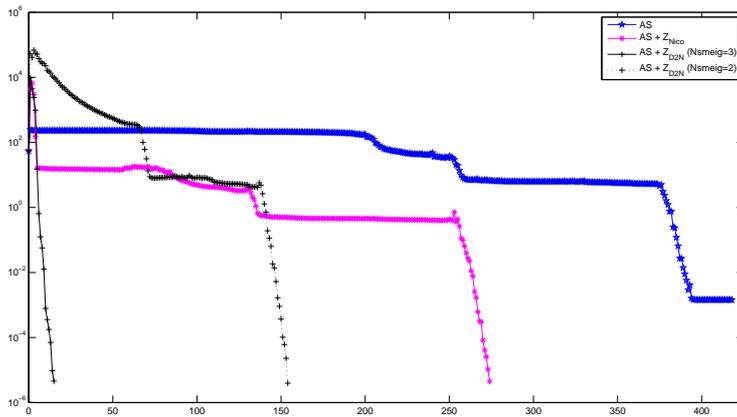


Figure 10: HPL3($a=100000$), the case with 3 high-permeability layers. $n_{domain}=64$, $n_{overlap}=1$, $n_{ypt}=8$, $n_{ypt}=16$. Increasing N_{smeig} can improve the convergence. We use P_{ADEF2} , constructed by Z_{Nico} , Z_{D2N} .

From the residual curves, we see that our methods are very efficient and robust. Next we examine the spectrum. The spectrum of preconditioned matrices has two characteristics: The spectrum is more clustered, and the smallest eigenvalue is well separated from the origin. We check the spectrum of the preconditioned system using P_{ADEF2} , which are constructed by Z_{D2N} . In Figure 11 and 12, we display the spectra of CONST case and HPL2 case respectively. Since the small eigenvalues near the origin influence the fast convergence, we check the minimum real part of eigenvalues of six cases (see Table 4).

Case	CONST	HPL1	HPL2	HPL3	HPL4	HPL5
N_{smeig}	1	1	2	3	7	9
$M^{-1}\tilde{A}$	2.67e-6	2.67e-6	2.67e-6	2.67e-6	2.67e-6	2.67e-6
$P_{ADEF2}\tilde{A}$	0.23	0.19	0.40	0.4	0.4	0.098

Table 4: The minimum real part of eigenvalues. \tilde{A} is the coefficient matrix of the extended system (5) and M is the Schwarz preconditioner. $n_{domain}=16$, $n_{overlap}=1$, $n_{ypt}=8$, $n_{ypt}=16$. The test cases are defined in Table 2.

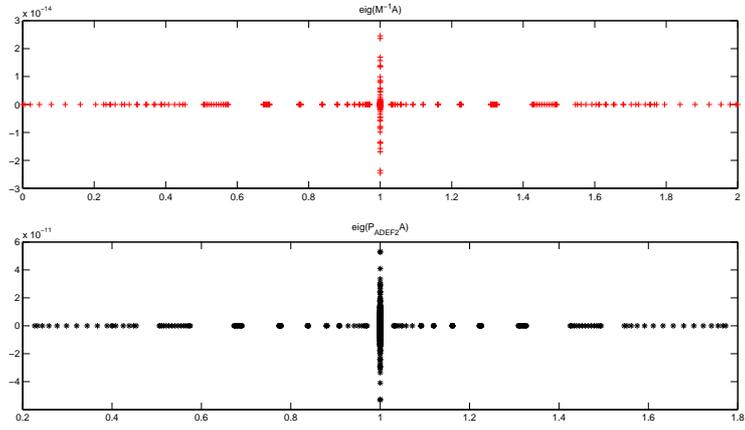


Figure 11: Spectra of CONST case. $n_{domain}=16$, $n_{overlap}=1$, $n_{ypt}=8$, $n_{ypt}=16$. The 2nd is the result of Z_{D2N} method with $N_{smeig}=1$.

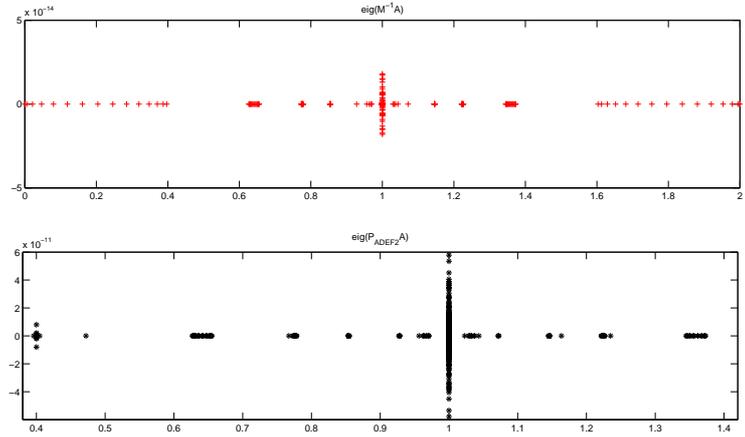


Figure 12: Spectra of HPL2 case. $n_{domain}=16$, $n_{overlap}=1$, $n_{ypt}=8$, $n_{ypt}=16$. The 2nd is the result of Z_{D2N} method with $N_{smeig}=2$.

4.2 General decompositions

We solve now the model problem (1) on the domain $\Omega = [0, 1]^2$ discretized by a P_1 finite element method. The diffusion κ is a function of x and y . The corresponding discretizations and data structures were obtained by using the software FreeFem++ [10] in connection with Metis partitioner [13]. In the following we will compare the RAS preconditioner with and without Nicolaides coarse space to the new preconditioner based on the harmonic extension of the eigenvectors of the local DtN operators. We consider here two situations:

- homogeneous viscosity: $\kappa = 1$ in the whole domain.
- highly heterogeneous viscosity: skyscraper and alternating case, (see Figure 15), defined as follows.

skyscraper viscosity: for x and y such that for $[10x] \equiv 0(\text{mod } 2)$ or $[10y] \equiv 0(\text{mod } 2)$, $\kappa = 10^3 \cdot ([10y] + 1)$, and $\kappa = 1$ elsewhere.

alternating viscosity: for y such that for $[10y] \equiv 0(\text{mod } 2)$, $\kappa = 10^3 \cdot ([10y] + 1)$, and $\kappa = 1$ elsewhere.

In all three cases, we first test the methods on overlapping decompositions into rectangular $N \times N$ domains with $N = 2, 4$ and 8 and then on decompositions into irregular domains obtained via Metis. These overlapping decompositions are build by adding layers to non-overlapping ones. The general non-overlapping decompositions can be generated for example, by using the Metis partitioner (see Figure 13 for such a decomposition into 64 subdomains). The number of discretization points for the uniform decompositions is 14×14 in every subdomain and an overlap of $\delta = 4$ mesh cells. The number of unknowns increase as N^2 .

Decomposition into 64 domains using Metis

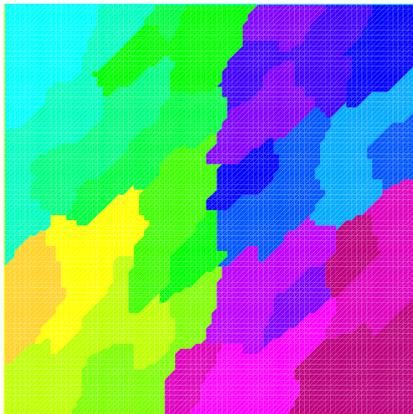


Figure 13: Decomposition into 64 subdomains using Metis.

Homogeneous viscosity As in the case of a one-way decomposition, the two preconditioning methods based on Nicolaides coarse grid and DtN eigenvectors, behave in quite a similar way when the viscosity is constant in the whole domain. If we take $N_{smeig}=1$ (as in this case the DtN operator has one small

eigenvalue in each subdomain) the iteration number for the two preconditioners is very close. In order to get a significant advantage (half of the number of iterations), one needs to consider at least $N_{smeig}=4$. Moreover, the results are stable in both cases when one varies the number of subdomains in each direction (see Table 5) and the convergence history shows no significant differences for uniform or automatic decompositions (see Figure 14).

N	uniform $N \times N$ decomposition			$N \times N$ decomp. using Metis		
	RAS	P_{BNN} : RAS+ Z_{Nico}	P_{BNN} : RAS+ Z_{D2N}	RAS	P_{BNN} : RAS+ Z_{Nico}	P_{BNN} : RAS+ Z_{D2N}
2	18	17	10	27	23	12
4	41	23	12	54	32	15
8	78	25	12	111	37	16

Table 5: The homogeneous case, $\kappa = 1$, $N_{smeig}=4$.

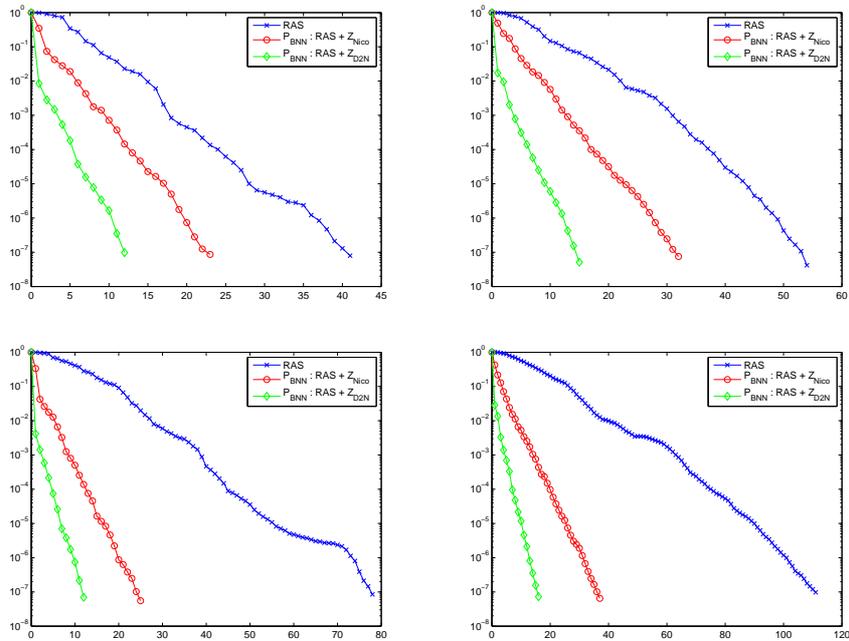


Figure 14: Convergence curves for the homogeneous case. $N_{smeig}=4$, Decomposition into N^2 subdomains ($N = 4$ top, $N = 8$ bottom) and a regular decomposition (left) vs. Metis decomposition (right pictures).

Highly heterogeneous viscosity In this case we will perform the same tests as before on two different configurations of highly heterogeneous viscosity, as defined above and shown in Figure 15. Here the number of eigenvectors considered in the construction of the coarse space is much more important than in the homogeneous case (otherwise the global information cannot be correctly captured) and it may vary as a function of the number of subdomains. Nevertheless, even with a very low number of eigenvectors $N_{smeig}=4$ (as in the homogeneous

case) show a clear advantage of the DtN preconditioner over Nicolaidis one: see Table 6 and Figure 16 for the skyscraper configuration and Table 7 and Figure 17 for the alternating configuration.

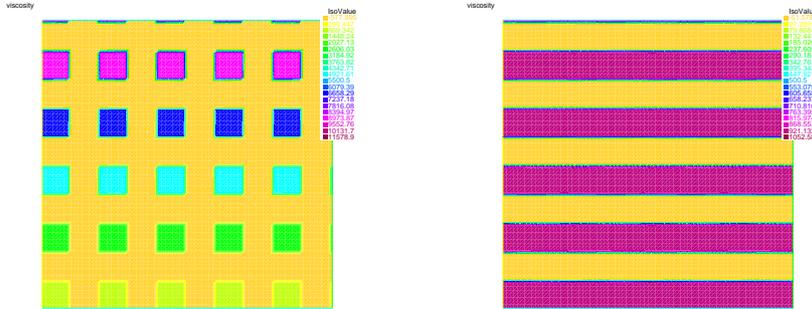


Figure 15: Heterogeneous viscosity: skyscraper and alternating cases.

N	uniform $N \times N$ decomposition			$N \times N$ decomp. using Metis		
	RAS	P_{BNN} : RAS+ Z_{Nico}	P_{BNN} : RAS+ Z_{D2N}	RAS	P_{BNN} : RAS+ Z_{Nico}	P_{BNN} : RAS+ Z_{D2N}
2	112	97	24	229	203	29
4	>400	299	18	>400	>400	32
8	>400	294	18	>400	>400	23

Table 6: The skyscraper, $N_{smeig}=4$.

N	uniform $N \times N$ decomposition			$N \times N$ decomp. using Metis		
	RAS	P_{BNN} : RAS+ Z_{Nico}	P_{BNN} : RAS+ Z_{D2N}	RAS	P_{BNN} : RAS+ Z_{Nico}	P_{BNN} : RAS+ Z_{D2N}
2	30	29	11	46	41	19
4	80	58	26	116	84	25
8	180	61	31	220	110	31

Table 7: The alternating viscosity, $N_{smeig}=4$.

A better strategy would consist in adapting N_{smeig} in function of the decomposition into subdomains. In Table 8, N_{smeig} is increased as the number of subdomains increases. We see that the results are further improved and are very similar to the homogeneous case.

4.3 The patch method

As we have seen before, using the coarse grid space enables the capture of the global information and greatly accelerates the convergence, for the constant case as well as for the cases with discontinuities. Without the coarse grid space, the information between two successive subdomains is exchanged through the boundary. In the previous section, we used Dirichlet interface conditions. Since the seminal work of P.L. Lions [14], it is well known that other interface conditions yield better convergence rates, see [3, 11]. In this section, we test the patch interface conditions, see [15]. They have the advantage to be defined at the algebraic level and they are easy to use for operators with arbitrary coefficients.

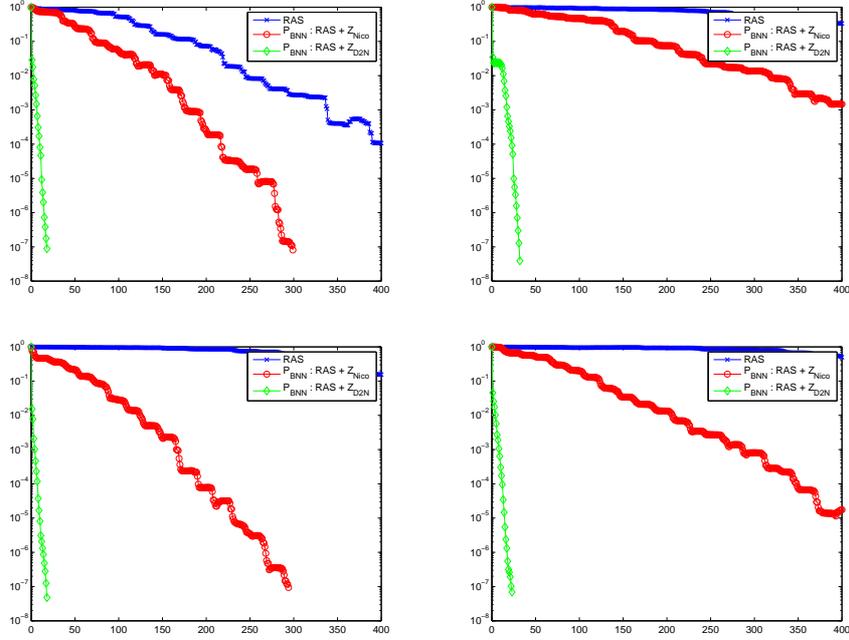


Figure 16: Convergence curves for the skyscraper case. $N_{smeig}=4$, Decomposition into N^2 subdomains ($N = 4$ top, $N = 8$ bottom) and a regular decomposition (left) vs. Metis decomposition (right pictures).

	uniform $N \times N$ decomposition		$N \times N$ decomp. using Metis	
N	$P_{BNN}: \text{RAS} + Z_{Nico}$	$P_{BNN}: \text{RAS} + Z_{D2N}(N_{smeig})$	$P_{BNN}: \text{RAS} + Z_{Nico}$	$P_{BNN}: \text{RAS} + Z_{D2N}(N_{smeig})$
2	97	12(6)	203	12(6)
4	299	11(8)	>400	12(8)
8	294	13(10)	>400	14(10)

Table 8: The skyscraper, varying N_{smeig} .

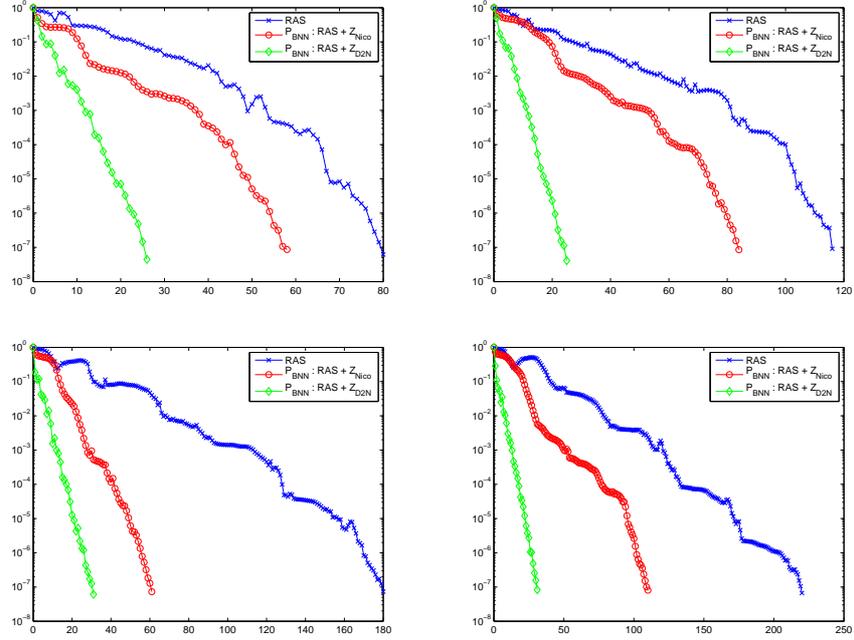


Figure 17: Convergence curves for the alternating viscosity case. $N_{smeig}=4$, Decomposition into N^2 subdomains ($N = 4$ top, $N = 8$ bottom) and a regular decomposition (left) vs. Metis decomposition (right pictures).

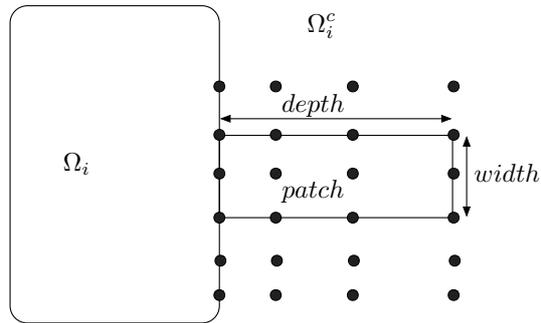


Figure 18: Patch method.

The principle is to approximate the Schur complement of the exterior of each subdomain (discrete counterpart of the DtN map) by a sparse matrix obtained by a “patch” method. At each node on the interface, is associated a patch region which is a small part of the exterior to the subdomain, see Figure 18. The small local Schur complement of the patch is computed and contributes locally to the approximation of the global Schur complement. This procedure is repeated for each node on the interface. Our tests are performed on the one-way decomposition of § 4.1. The width of the patch is 5 nodes and the depth is the size of the neighbor subdomain. Using the patch method can decrease the iteration number. Only half of the iteration number is needed for the constant case (see Figure 19). For the case with discontinuities, such as HPL3, the patch method can also improve the convergence (see Figure 20).

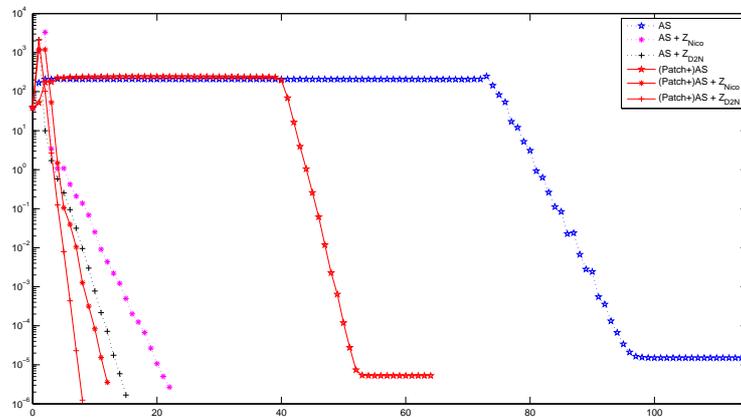


Figure 19: Convergence of CONST case with or without patch. $N_{smeig}=2$, $n_{domain}=32$. The preconditioner used here is P_{ADEF2} .

5 Conclusions

We have considered the extended (5) and the original (4) linear systems arising from the domain decomposition method with overlapping. We applied the two-level preconditioner using the Schwarz algorithm and the coarse grid correction. The coarse grid space is based on the low frequency modes of the local DtN map. Its size can be adapted to the difficulty of the problem. With this coarse space, we can obtain fast convergence for problems with large discontinuities (even along the interface) and arbitrary domain decompositions.

References

- [1] J. H. BRAMBLE, J. E. PASCIAK, A. H. SCHATZ, *The construction of preconditioners for elliptic problems by substructuring, I*, Math. Comput., 47(1986), pp. 103-134.

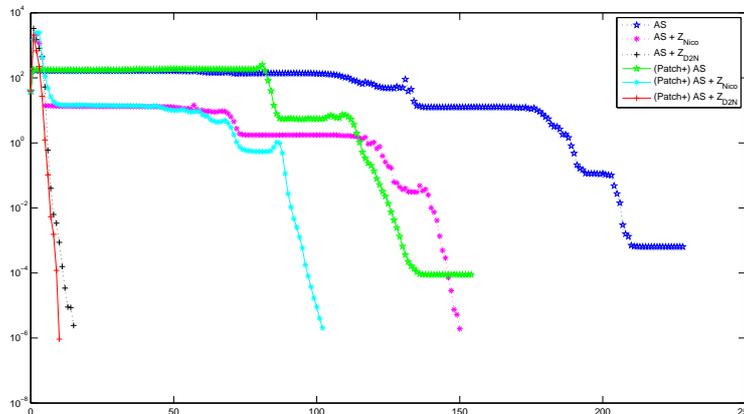


Figure 20: Convergence of HPL3 case with or without patch. $N_{smeig}=3$, $n_{domain}=32$. The preconditioner is P_{ADEF2} . When using Z_{D2N} and patch, it needs 10 iterations; Z_{D2N} without patch needs 15 iterations.

- [2] X. -C. CAI, M. SARKIS, *A restricted additive Schwarz preconditioner for general sparse linear systems*, SIAM J. Sci. Comput., 21(1999), pp. 239-247.
- [3] B. DESPRÉS, *Décomposition de domaine et problème de Helmholtz*, C.R. Acad. Sci. Paris, 1(6):313–316, 1990.
- [4] DOHRMANN, CLARK R. ; WIDLUND, OLOF B. *An overlapping Schwarz algorithm for almost incompressible elasticity*. SIAM J. Numer. Anal. 47 (2009), no. 4, 2897–2923.
- [5] CLARK R. DOHRMANN AND OLOF B. WIDLUND. *Hybrid Domain Decomposition Algorithms for Compressible and Almost Incompressible Elasticity* Internat. J. Numer. Methods Engrg., Vol. 82. 2010, 157–183.
- [6] DRYJA, MAKSYMILIAN ; SARKIS, MARCUS V. ; WIDLUND, OLOF B. *Multilevel Schwarz methods for elliptic problems with discontinuous coefficients in three dimensions.*, Numer. Math. 72 (1996), no. 3, 313–348.
- [7] E. EFSTATHIOU, M. GANDER, *Why restricted additive Schwarz converges faster than additive Schwarz*, BIT Numer. Math., 43(2003), pp. 945-959.
- [8] Y. A. ERLANGGA, R. NABBEN, *Deflation and balancing preconditioners for Krylov subspace methods applied to nonsymmetric matrices*, SIAM J. Matrix Anal. Appl., 30(2008), pp. 684-699.
- [9] Y. A. ERLANGGA, R. NABBEN, *Algebraic multilevel Krylov methods*, SIAM J. Sci. Comput., 31 (2009), pp. 3417-3437.
- [10] FREEFEM++, Laboratoire J.L. Lions, CNRS UMR 7598, <http://www.freefem.org/ff++/>

- [11] M. J. GANDER, F. MAGOULÉS, F. NATAF, *Optimized Schwarz methods without overlap for the Helmholtz equation*, SIAM J. Sci. Comput., 24(2002), pp. 38-60.
- [12] G. H. Golub, C. F. Van Loan, *Matrix Computations*, 3rd Edition, John Hopkins University Press, Baltimore, MD, 1996.
- [13] G. KARYPIS, V. KUMAR, METIS - Unstructured Graph Partitioning and Sparse Matrix Ordering System.
- [14] P.-L. LIONS, *On the Schwarz alternating method. III: a variant for nonoverlapping subdomains*, In T. F. Chan, R. Glowinski, J. Périaux, and O. Widlund, editors, *Third International Symposium on Domain Decomposition Methods for Partial Differential Equations , held in Houston, Texas, March 20-22, 1989*, Philadelphia, PA, 1990. SIAM.
- [15] F. MAGOULÈS, F. -X. ROUX, L. SERIES, *Algebraic approximation of Dirichlet-to-Neumann maps for the equations of linear elasticity*, Comput. Methods Appl. Mech. Engrg., 195(2006), pp. 3742-3759.
- [16] J. MANDEL, *Balancing domain decomposition*, Communications in Applied and Numerical Methods, 9(1993), pp. 233-241.
- [17] MANDEL, JAN ; BREZINA, MARIAN, *Balancing domain decomposition for problems with large jumps in coefficients*. Math. Comp. 65 (1996), no. 216, 1387–1401.
- [18] F. NATAF, F. ROGIER, E. DE STURLER, *Optimal interface conditions for domain decomposition methods*, CMAP (Ecole Polytechnique), Internal Report 301, 1994. <http://www.ann.jussieu.fr/nataf/cvfiniMod.pdf>
- [19] R. A. NICOLAIDES, *Deflation of conjugate gradients with applications to boundary value problems*, SIAM J. Numer. Anal., 24(1987), pp. 355-365.
- [20] A. PADIY, O. AXELSSON, B. POLMAN, *Generalized augmented matrix preconditioning approach and its application to iterative solution of ill-conditioned algebraic systems*, SIAM J. Matrix Anal. Appl., 22(2000), pp. 793-818.
- [21] PECHSTEIN, CLEMENS ; SCHEICHL, ROBERT . *Scaling up through domain decomposition*. Appl. Anal. 88 (2009), no. 10-11, 1589–1608.
- [22] PECHSTEIN, CLEMENS ; SCHEICHL, ROBERT . *Analysis of FETI methods for multiscale PDEs*. Numer. Math. 111 (2008), no. 2, 293–333.
- [23] B. E. SMITH, P. E. BJØSTAD, *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*, Cambridge University Press, 1996.
- [24] J. M. TANG, R. NABBEN, C. VUIK, Y. A. ERLANGGA, *Comparison of Two-Level Preconditioners Derived from Deflation, Domain Decomposition and Multigrid Methods*, J. Sci. Comput., 39(2009), pp. 340-370.

- [25] J. M. TANG, S. P. MACLACHLAN, R. NABBEN, C. VUIK, *A comparison of two-level preconditioners based on multigrid and deflation*, SIAM. J. Matrix Anal. Appl., 31(2010), pp. 1715-1739.
- [26] A. TOSELLI, O. WIDLUND, *Domain decomposition methods: algorithms and theory*, Springer, 2005.
- [27] C. VUIK, A. SEGAL, J. A. MEIJERINK, *An efficient preconditioned CG method for the solution of a class of layered problems with extreme contrasts in the coefficients*, Journal of Computational Physics, 152(1999), pp. 385-403.
- [28] C. VUIK, A. SEGAL, J. A. MEIJERINK, G. T. WIJMA, *The construction of projection vectors for a deflated ICCG method applied to problems with extreme contrasts in the coefficients*, Journal of Computational Physics, 172(2001), pp. 426-450.