

Interface Connections in Domain Decomposition Methods

NATO Advanced Study Institute

Modern Methods in Scientific Computing and Applications

Département de Mathématiques et de statistique

Université de Montréal

F. Nataf*

9-20 Juillet 2001

Abstract

Parallel computers are increasingly used in scientific computing. They enable one to perform large scale computations. New algorithms which are well suited to such architectures have to be designed. Domain decomposition methods are a very natural way to exploit the possibilities of multiprocessor computers, but such algorithms are very useful when used on monoprocessor computers as well.

The idea is to decompose the computational domain into smaller subdomains. The equations are solved on each subdomain. In order to enforce the matching of the local solutions, interface conditions have to be written on the boundary between subdomains. These conditions are imposed iteratively. The convergence rate is very sensitive to these interface conditions. Theoretical and numerical results are given.

1 Introduction

1.1 Why domain decomposition methods?

Domain decomposition is a tool introduced artificially to ease large scale computations or that is natural in some situations. These methods are well adapted to parallel computers and are very popular in parallel computing. They are also very efficient when used on sequential computers. This is due to the nonlinear cost of a simulation. Since the 50's, and long before the advent of parallel computers, these methods have been (and are still) used. They enable one to perform robust large scale computations even on a monoprocessor computer. In some sense, for domain decomposition methods, parallel computing is a secondary issue. Also, in some situations, the domain decomposition is natural from the physics of the problem: different physics in different subdomains, moving domains, strongly heterogeneous media.

Three dimensional numerical simulations are very demanding in CPU or/and in memory. When explicit schemes of the general following form

$$U(t + \Delta t, \cdot) = U(t, \cdot) + \Delta t F(U(t, \cdot))$$

*CMAP, CNRS UMR7641, Ecole Polytechnique, 91128 Palaiseau, France. nataf@cmapx.polytechnique.fr, www.cmap.polytechnique.fr/~nataf

or Monte Carlo methods are used, the limitation comes mostly from the CPU time. The parallelization is clear. The correct balancing of tasks is then a crucial issue in order to use parallel computers efficiently. The use of a parallel computer is very profitable. But very often, numerical simulations involve the solving very large linear systems arising from Poisson or Helmholtz equations or from the Jacobian matrix in a Newton’s method. These computations are very demanding both in CPU time and in memory. Generally speaking, direct solvers are too costly and iterative solvers are not robust enough especially for problems with strongly heterogeneous media and ill conditioned matrices, see e.g. [Nep91]. There is a need for hybrid iterative/direct solvers: these are domain decomposition methods. Roughly speaking, the computational domain is decomposed into smaller subdomains. The equations in the subdomains are solved by a direct method and the matching of the solutions is imposed iteratively. Since the cost of a simulation is nonlinear w.r.t. the number of unknowns, breaking the initial problem into a set of smaller subproblems is profitable. Also, in some situations, the domain decomposition is natural from the physics of the problem: different physics in different subdomains (e.g. fluid/structure interaction), moving domains (e.g. rotor and stator in an electric motor), strongly heterogeneous media: sliding blocks along faults in subsurface modeling, see Fig. 1).

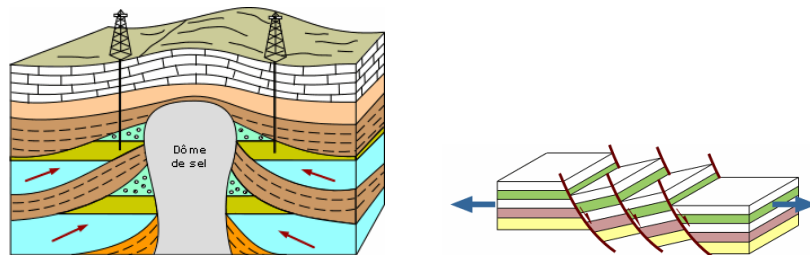


Figure 1: Subsurface modeling of a geological basin — extract from <http://www.ggl.ulaval.ca>

In a slightly different context, the mesh generation is a complex and time-consuming task both for the user and the computer. In order to speed up this task, an increasingly popular possibility is to first generate a decomposition of the domain into large subdomains and then to mesh each subdomain concurrently. The mesh generation becomes a parallel task. The resulting mesh is of course non-conforming on the interfaces between the subdomains. Therefore new tools for handling non-conforming mesh are needed. In this case, “Domain connection” would be a more appropriate term, see Fig. 2 and references [BMP93, BD98, CDS99, AMW99, AK95, AKP95, Woh99].

1.2 The original Schwarz method (1870)

The first domain decomposition method was developed at the end of the 19th century by the mathematician H. A. Schwarz. His goal was to study the Laplace operator. At that time, the main tool for this purpose was Fourier analysis and more generally the use of special functions. Geometries of the domain were essentially restricted to simple configurations: rectangles and disks, see Fig. 3. His idea was to study the case of a domain that is the union

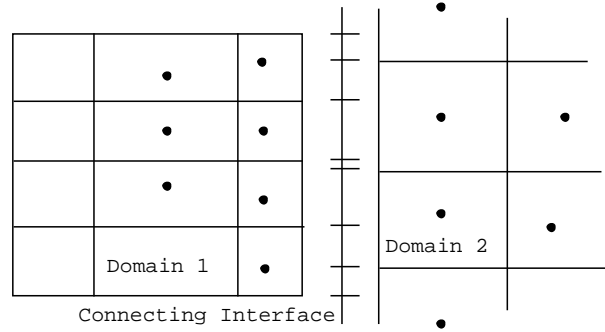


Figure 2: Non conforming mesh

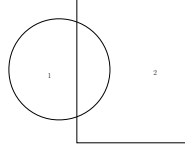


Figure 3: Overlapping domain decomposition

of simple domains. For example, let $\Omega = \Omega_1 \cup \Omega_2$ with $\Omega_1 \cap \Omega_2 \neq \emptyset$. We want to solve

$$\begin{aligned} -\Delta(u) &= f & \text{in } \Omega \\ u &= 0 & \text{on } \partial\Omega. \end{aligned} \tag{1.1}$$

Schwarz proposed the following algorithm (Multiplicative Schwarz Method, **MSM**):

Let (u_1^n, u_2^n) be an approximation to $(u|_{\Omega_1}, u|_{\Omega_2})$ at step n of the algorithm, (u_1^{n+1}, u_2^{n+1}) is defined by

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

Problem in domain Ω_1 has to be solved before problem in domain Ω_2 . This algorithm is sequential.

A slight modification of the algorithm is the additive Schwarz method (**ASM**)

$$\begin{aligned} -\Delta(u_1^{n+1}) &= f & \text{in } \Omega_1 & & -\Delta(u_2^{n+1}) &= f & \text{in } \Omega_2 \\ u_1^{n+1} &= 0 & \text{on } \partial\Omega_1 \cap \partial\Omega & & u_2^{n+1} &= 0 & \text{on } \partial\Omega_2 \cap \partial\Omega \\ u_1^{n+1} &= u_2^n & \text{on } \partial\Omega_1 \cap \overline{\Omega_2} & & u_2^{n+1} &= u_1^n & \text{on } \partial\Omega_2 \cap \overline{\Omega_1}. \end{aligned} \tag{1.2}$$

Problems in domains Ω_1 and Ω_2 may be solved concurrently. The ASM is a parallel algorithm and is adapted to parallel computers. Schwarz proved the linear convergence of (u_1^n, u_2^n) to $(u|_{\Omega_1}, u|_{\Omega_2})$ as n tends to infinity.

The benefit of these algorithms is the saving in memory requirements. Indeed, if the problems are solved by direct methods, the cost of the storage is non-linear with respect to

the number of unknowns. By dividing the original problem into smaller pieces the amount of storage can be significantly reduced. As far as CPU is concerned, the original algorithms ASM and MSM are very slow. Another weakness is the need of overlapping subdomains. Indeed, only the continuity of the solution is imposed and nothing is imposed on the matching of the fluxes. When there is no overlap convergence is thus impossible.

The slowness of the method and the need for overlapping subdomains are linked. Indeed, it can be proved that the convergence rate of the Schwarz method is a continuous function of the size of the overlap denoted δ . For small overlaps the convergence rate is close to one. Actually it can be proved that for small overlaps the convergence rate varies as $1 - C^t \delta$.

1.3 Towards faster methods: two families of methods

In order to remedy the drawbacks of the original Schwarz method, two families of methods have been developed. They both work in the non-overlapping case and consist of introducing the normal derivative of the solution, but in two very different ways:

- write a substructured formulation of the domain decomposition problem where the matching of the solution and of its normal derivative along the interface are imposed explicitly.
- Modify the original Schwarz method by replacing the Dirichlet interface conditions on $\partial\Omega_i \setminus \partial\Omega$, $i = 1, 2$, by Robin interface conditions $(\partial_{n_i} + \alpha)$, where n is the outward normal to subdomain Ω_i , see [Lio90].

The first approach is explained in section 2 and the second in section 3.

More generally, a complete overview of various domain decomposition methods may be found in a few books [CM94, SBG96, QV99] or in the proceedings of various conferences on domain decomposition methods, see e.g. [CGPW89, PEBK97, LBCW98] and references therein.

2 Substructured formulation

We explain the basic ideas for a two domain decomposition first at the continuous level and then at the matrix level. Then we give some references on the general case.

2.1 The symmetric positive definite continuous case

2.1.1 Substructured formulation

We still consider equation (1.1). The domain Ω is decomposed into two non-overlapping subdomains Ω into Ω_1 and Ω_2 . The interface $\partial\Omega_1 \cap \partial\Omega_2$ is denoted by Γ . A first formulation of (1.1) as a domain decomposition method consists of looking for $u_i = u|_{\Omega_i}$, $i = 1, 2$, which

must satisfy

$$\begin{aligned} -\Delta(u_i) &= f \quad \text{in } \Omega_i, \quad i = 1, 2 \\ u_i &= 0 \quad \text{on } \partial\Omega_i \setminus \Gamma, \quad i = 1, 2 \\ u_1 &= u_2 \quad \text{on } \Gamma, \\ \left(\frac{\partial u_1}{\partial n_1} + \frac{\partial u_2}{\partial n_2} \right) &= 0 \quad \text{on } \Gamma. \end{aligned}$$

The substructured formulation consists in formulating the problem in terms of the common value of u_1 and u_2 on Γ , denoted by $u|_\Gamma$. This is obtained by eliminating the internal unknowns via the solving of local subproblems. We introduce a Dirichlet BVP in each subdomain with $u|_\Gamma$ as a Dirichlet data:

$$\begin{aligned} -\Delta(u_i) &= f \quad \text{in } \Omega_i, \\ u_i &= u|_\Gamma \quad \text{on } \Gamma, \quad u_i = 0 \quad \text{on } \partial\Omega_i \setminus \Gamma. \end{aligned}$$

The jump of the normal derivative across the interface is a function of f and $u|_\Gamma$,

$$\mathcal{S}(f, u|_\Gamma) = \left(\frac{\partial u_1}{\partial n_1} + \frac{\partial u_2}{\partial n_2} \right) \Big|_\Gamma. \quad (2.1)$$

The substructured interface problem reads: Find $u|_\Gamma$ such that

$$\mathcal{S}(0, u|_\Gamma) = -\mathcal{S}(f, 0). \quad (2.2)$$

The corresponding discretized operator \mathcal{S}_h (the Schur complement of the matrix) is a matrix whose coefficients can be computed by solving a Dirichlet boundary value problem for each interface unknown (or d.o.f.). This task is long. Moreover, this matrix is full and the solving of the substructured problem by a direct method would be very costly. For these reasons, the discretized problem corresponding to (2.2) is solved by a Krylov type method such as CG, GMRES, BICGSTAB, QMR, . . . , as only matrix-vector products are needed. In our case, it corresponds to solving a Dirichlet boundary value problem in each subdomain. Therefore, there is no need to build explicitly the substructured matrix \mathcal{S}_h . Moreover, this is a parallel task.

The convergence is enhanced when compared to a Krylov type method applied to the original problem. Indeed, the condition number of the discretized operator $\kappa(\mathcal{S}_h)$ is $O(1/h)$ whereas the condition number of a finite difference, finite volume or finite element discretization of the Laplace operator is $O(1/h^2)$. The size of the substructured problem is much smaller than the size of the overall problem. Moreover, as we shall see next, there are very good preconditioners \mathcal{T}_h available for \mathcal{S}_h which are almost optimal ($\kappa(\mathcal{T}_h \mathcal{S}_h) \simeq O(1)$). These remarks are even more relevant for problems with large jumps in the coefficients. In this case, there are no robust preconditioners for the original problem, whereas they exist for the substructured formulation, see [DSW96, LT94].

2.1.2 The basis for the Neumann-Neumann preconditioner

A very popular preconditioner for the operator $\mathcal{S}(0, \cdot)$ has been proposed in [BGLTV]. The basis for the preconditioner comes from the special case where the domain Ω is decomposed along a symmetry axis, e.g. a rectangle decomposed into two half rectangles or the whole plane decomposed into two half planes. In that case we have

- the operator $\mathcal{S}(0, \cdot)$ is naturally split into the sum of two operators \mathcal{S}_1 and \mathcal{S}_2 which are defined for each subdomain and such that $\mathcal{S}_1 = \mathcal{S}_2$.
- The inverses \mathcal{S}_1^{-1} and \mathcal{S}_2^{-1} are obtained via the solving of boundary value problems in each subdomain.

Then

$$\mathcal{T} := \frac{1}{4}(\mathcal{S}_1^{-1} + \mathcal{S}_2^{-1}) \quad (2.3)$$

is an exact preconditioner: $\mathcal{T}\mathcal{S}(0, \cdot) = \text{Id}$.

For simplicity, we explicitly give these operators for the following model problem:

Find $u : \mathbf{R}^2 \rightarrow \mathbf{R}$ such that

$$(\eta - \Delta)(u) = f,$$

where η is positive and f is a given function. The domain is decomposed into two non-overlapping half-planes $\Omega_1 = (-\infty, 0) \times \mathbf{R}$ and $\Omega_2 = (0, \infty) \times \mathbf{R}$, and the interface is $\Gamma = \{0\} \times \mathbf{R}$. We define the operators \mathcal{S}_i , $i = 1, 2$, which maps a function living on Γ to a function living on Γ via the solving of a Dirichlet boundary value problem

$$\mathcal{S}_i : v_\Gamma \rightarrow \frac{\partial v}{\partial n_i} \Big|_\Gamma,$$

where n_i is the outward normal to the domain Ω_i , and v_i is the solution to the following BVP:

$$\begin{cases} (\eta - \Delta)(v_i) = 0 & \text{in } \Omega_i, \\ v_i = v_\Gamma & \text{on } \Gamma. \end{cases}$$

The operator \mathcal{S}_i is the Steklov-Poincaré a.k.a. the Dirichlet to Neumann (DtN) operator of domain Ω_i . By symmetry of the operator and of the decomposition, we have $\mathcal{S}_1 = \mathcal{S}_2$. It is clear that $\mathcal{S}(0, \cdot) = \mathcal{S}_1 + \mathcal{S}_2$.

The inverse of \mathcal{S}_i corresponds to the solving of a Neumann problem

$$\mathcal{S}_i^{-1} : g_\Gamma \rightarrow v_i|_\Gamma$$

with v_i the solution to the following BVP

$$\begin{cases} (\eta - \Delta)(v_i) = 0 & \text{in } \Omega_i, \\ \frac{\partial v_i}{\partial n_i} = g_\Gamma & \text{on } \Gamma. \end{cases}$$

The name ‘‘Neumann-Neumann’’ preconditioner (2.3) comes from the fact that applying it amounts to solving a Neumann BVP in each subdomain.

Only the symmetry of the operator and of the domain decomposition and the well-posedness of Dirichlet and Neumann BVP’s are used. Therefore, this preconditioner can naturally be extended to elasticity or the Stokes system. It is also easily defined at the discrete level which eases its implementation.

2.1.3 The FETI or dual method

It is possible to consider the Neumann-Neumann preconditioner the other way round. The unknown is the flux at the interface between the subdomains. Let g be a function that lives on the interface Γ , and consider the Neumann BVP's ($i = 1, 2$):

$$\begin{cases} -\Delta v_i = f & \text{in } \Omega_i, \\ \frac{\partial v_i}{\partial n_i} = (-1)^i g_\Gamma & \text{on } \Gamma. \end{cases}$$

The jump of the solutions across the interface is a function of f and g :

$$\mathcal{T}(f, g_\Gamma) := v_2 - v_1.$$

The substructured formulation is

Find g_Γ such that

$$\mathcal{T}(0, g_\Gamma) := -\mathcal{T}(f, 0).$$

A good preconditioner is

$$\frac{1}{4}\mathcal{S}(0, \cdot)$$

which involves the solving in parallel of a Dirichlet problem in each subdomain. This approach has been developed in [FR91, CMW95, KW01].

2.2 At the matrix level

When the problem (1.1) is discretized by a finite element method for instance, it yields a linear system of the form $AU = F$, where F is a given right-hand side and U is the set of unknowns. Corresponding to the domain decomposition, the set of unknowns U is decomposed into interior nodes of the subdomains U_1 and U_2 , and to unknowns, U_Γ , associated to the interface Γ . This leads to a block decomposition of the linear system

$$\begin{pmatrix} A_{11} & A_{1\Gamma} & 0 \\ A_{\Gamma 1} & A_{\Gamma\Gamma} & A_{\Gamma 2} \\ 0 & A_{2\Gamma} & A_{22} \end{pmatrix} \begin{pmatrix} U_1 \\ U_\Gamma \\ U_2 \end{pmatrix} = \begin{pmatrix} F_1 \\ F_\Gamma \\ F_2 \end{pmatrix}. \quad (2.4)$$

The substructuring of the linear system corresponds to the Gauss elimination of the unknowns U_1 using the first line of (2.4), and of U_2 using the last line. These eliminations correspond to solving discretized Dirichlet boundary value problems. The resulting linear system reads:

Find U_Γ such that

$$\mathcal{S}_h(0, U_\Gamma) := (A_{\Gamma\Gamma} - A_{\Gamma 1} A_{11}^{-1} A_{1\Gamma} - A_{\Gamma 2} A_{22}^{-1} A_{2\Gamma})(U_\Gamma) = F_\Gamma - A_{\Gamma 1} A_{11}^{-1} F_1 - A_{\Gamma 2} A_{22}^{-1} F_2$$

The matrix $\mathcal{S}_h(0, \cdot)$ (the Schur complement of the original matrix) is full, due to A_{ii}^{-1} in its expression.

We now describe the Neumann-Neumann preconditioner at the matrix level. In order to split the matrix $\mathcal{S}_h(0, \cdot)$ into two matrices, we use the natural decomposition of $A_{\Gamma\Gamma}$ into its contribution coming from each subdomain $A_{\Gamma\Gamma} = A_{\Gamma\Gamma}^1 + A_{\Gamma\Gamma}^2$. More precisely, for the Laplace operator, let k, l be the indices of two degrees of freedom on the interface associated with

two basis functions ϕ^k and ϕ^l . The corresponding entry $a_{\Gamma\Gamma}^{kl}$ can be decomposed into a sum $a_{\Gamma\Gamma}^{kl} = a_{\Gamma\Gamma}^{1,kl} + a_{\Gamma\Gamma}^{2,kl}$, where

$$a_{\Gamma\Gamma}^{i,kl} = \int_{\Omega_i} \nabla\phi^k \nabla\phi^l, \quad i = 1, 2.$$

Then we define the discrete counterparts of \mathcal{S}_i ($i = 1, 2$):

$$\mathcal{S}_{i,h} : U_\Gamma \rightarrow A_{\Gamma\Gamma}^i U_\Gamma + A_{\Gamma i}^i U_i,$$

where $A_{11}U_1 = -A_{1\Gamma}U_\Gamma$ (Dirichlet problem). Its inverse reads

$$\mathcal{S}_{i,h}^{-1} : G \rightarrow V_\Gamma,$$

where

$$\begin{pmatrix} A_{ii} & A_{i\Gamma} \\ A_{i\Gamma} & A_{\Gamma\Gamma}^i \end{pmatrix} \begin{pmatrix} V_i \\ V_\Gamma \end{pmatrix} = \begin{pmatrix} 0 \\ G \end{pmatrix} \quad (\text{Neumann problem}).$$

We have $\mathcal{S}_{i,h}(0, \cdot) = \mathcal{S}_{1,h} + \mathcal{S}_{2,h}$ and the Neumann-Neumann preconditioner is

$$\mathcal{T}_h = \frac{1}{4}(\mathcal{S}_{1,h}^{-1} + \mathcal{S}_{2,h}^{-1}).$$

For a symmetric decomposition of the domain, it is exact.

For a definition of the preconditioner for an arbitrary decomposition, see e.g. [LT94].

2.3 The non-symmetric continuous case: The Robin-Robin preconditioner

We shall see that the substructured formulation works as in the SPD case but that the Neumann-Neumann preconditioner has to be modified.

We consider the convection-diffusion equation arising from the time discretization by a backward Euler scheme of the time-dependent equation

$$\mathcal{L} := \frac{1}{\Delta t} + \mathbf{a} \cdot \nabla - \nu \Delta,$$

where Δt is the time step, \mathbf{a} is a given vector field and ν is the viscosity. The domain on which the equations are posed is the plane \mathbf{R}^2 decomposed into two half-planes as in section 2.1.2. By replacing $-\Delta$ by \mathcal{L} in the definition of \mathcal{S} (2.1), the substructured formulation is still given by (2.2). The Neumann-Neumann preconditioner is still given by

$$\mathcal{T} := \frac{1}{4}(\mathcal{S}_1^{-1} + \mathcal{S}_2^{-1}), \quad (2.5)$$

where

$$\mathcal{S}_i : v_\Gamma \rightarrow \left. \frac{\partial v_i}{\partial n_i} \right|_\Gamma \quad (2.6)$$

and

$$\begin{cases} \mathcal{L}(v_i) = 0 & \text{in } \Omega_i, \\ v_i = v_\Gamma & \text{on } \Gamma. \end{cases}$$

It is clear that $\mathcal{S}(0, \cdot) = \mathcal{S}_1 + \mathcal{S}_2$. But, due to the non symmetry of the operator \mathcal{L} , the operators \mathcal{S}_1 and \mathcal{S}_2 are different in general. Thus, the Neumann-Neumann preconditioner \mathcal{T} is no longer exact, $\mathcal{T}\mathcal{S}(0, \cdot) \neq \text{Id}$. In order to see exactly how \mathcal{S}_1 and \mathcal{S}_2 differ, we perform a Fourier analysis. It will lead us to a new preconditioner adapted to the non-symmetric case: the Robin-Robin preconditioner.

2.3.1 A Fourier analysis

By using Fourier transform, we give an explicit form for the operators \mathcal{S}_i , $i = 1, 2$, see above (2.6). We denote the partial Fourier transform of $f(x, y) : \mathbf{R}^2 \rightarrow \mathbf{R}$ in the y variable by

$$\hat{f}(x, k) = \mathcal{F}_y(f)(x, k) := \int_{-\infty}^{\infty} e^{-Iky} f(x, y) dy$$

($I^2 = -1$), and the inverse Fourier transform of $\hat{f}(x, k)$ by

$$f(x, y) = \mathcal{F}_y^{-1}(\hat{f})(x, y) := \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{Iky} \hat{f}(x, k) dk.$$

Our analysis will also involve the Fourier transform of a convolution operator with kernel $h(y)$,

$$\Lambda(u)(y) := \int_{-\infty}^{\infty} h(y - z)u(z) dz,$$

whose Fourier transform is given by

$$\mathcal{F}_y(\Lambda(u))(k) = \hat{h}(k)\hat{u}(k)$$

or equivalently with $\hat{\Lambda}(k) := \hat{h}(k)$,

$$\Lambda(u) = \mathcal{F}_y^{-1}(\hat{\Lambda}(k)\hat{u}(k)).$$

The function $\hat{\Lambda}(k)$ is called the symbol of the operator Λ . For example, the symbol of the operator $-\partial_{yy}$ is the polynomial k^2 . More generally, the symbol of any constant coefficient differential operator is a polynomial in the Fourier variable k and conversely.

The vector field $\mathbf{a} = (a_1, a_2)^T$ and the coefficient ν are assumed to be constants and $a_1 > 0$. We take the partial Fourier transform of (2.6) in the y direction

$$\left(\frac{1}{\Delta t} + a_1 \partial_x + a_2 I k - \nu \frac{\partial^2}{\partial x^2} + \nu k^2 \right) (\hat{v}_i(x, k)) = 0.$$

For a fixed k , this is an ordinary differential equation in x whose solution is sought in the form $\sum_{\alpha} c_{\alpha} \exp(\lambda_{\alpha} x)$, so that λ_{α} is a root of the second order polynomial

$$\frac{1}{\Delta t} + a_1 \lambda_{\alpha} + a_2 I k - \nu \lambda_{\alpha}^2 + \nu k^2.$$

We have two roots with opposite signs:

$$\lambda^{\pm} = \frac{a_1 \pm \sqrt{a_1^2 + 4\nu \left(\frac{1}{\Delta t} + I k a_2 + \nu k^2 \right)}}{2\nu}. \quad (2.7)$$

The general form of $\hat{v}_i(x, k)$ is thus

$$\hat{v}_i(x, k) = c_i^+(k) e^{\lambda^+ x} + c_i^-(k) e^{\lambda^- x}.$$

The coefficients c_i^\pm are computed from the boundary conditions. We consider first \hat{v}_1 . The solution must be bounded as $x \rightarrow -\infty$ so that $c_i^-(k) \equiv 0$. From the Dirichlet boundary condition at $x = 0$ we get

$$\hat{v}_1(x, k) = \hat{v}_\Gamma(k) e^{\lambda^+ x}$$

and similarly,

$$\hat{v}_2(x, k) = \hat{v}_\Gamma(k) e^{\lambda^- x}.$$

The symbols of \mathcal{S}_1 and \mathcal{S}_2 are thus

$$\hat{\mathcal{S}}_1(k) = \lambda^+(k) \tag{2.8}$$

and (due to $n_2 = -\partial_x$)

$$\hat{\mathcal{S}}_2(k) = -\lambda^-(k). \tag{2.9}$$

The operators \mathcal{S}_1 and \mathcal{S}_2 differ: $\mathcal{S}_1(k) - \mathcal{S}_2(k) = a_1/\nu$.

2.3.2 Definition of the Robin-Robin preconditioner

Thanks to these formulas it is possible to split the operator $\mathcal{S}(0, \cdot)$ into two equal contributions of each subdomain:

$$\mathcal{S}(0, \cdot) = \tilde{\mathcal{S}}_1 + \tilde{\mathcal{S}}_2,$$

where

$$\hat{\tilde{\mathcal{S}}}_1 = \hat{\tilde{\mathcal{S}}}_2 = \hat{\mathcal{S}}_1 - \frac{a_1}{2\nu} = \hat{\mathcal{S}}_2 + \frac{a_1}{2\nu}.$$

Noticing that $a_1 = \mathbf{a} \cdot \mathbf{n}_1 = -\mathbf{a} \cdot \mathbf{n}_2$, it is possible to give an intrinsic definition of the operators $\tilde{\mathcal{S}}_i$, $i = 1, 2$:

$$\tilde{\mathcal{S}}_i : v_\Gamma \rightarrow \left. \frac{\partial v_i}{\partial n_i} \right|_\Gamma - \frac{\mathbf{a} \cdot \mathbf{n}_i}{2\nu} v_i, \tag{2.10}$$

where

$$\begin{cases} \mathcal{L}(v_i) = 0 & \text{in } \Omega_i, \\ v_i = v_\Gamma & \text{on } \Gamma. \end{cases}$$

The inverse of $\tilde{\mathcal{S}}_i$, $i = 1, 2$, amounts to solving a Robin boundary value problem:

$$\tilde{\mathcal{S}}_i^{-1} : g_\Gamma \rightarrow w_i|_\Gamma, \tag{2.11}$$

where

$$\begin{cases} \mathcal{L}(w_i) = 0 & \text{in } \Omega_i, \\ \left(\left. \frac{\partial}{\partial n_i} \right|_\Gamma - \frac{\mathbf{a} \cdot \mathbf{n}_i}{2\nu} \right) (w_i) = g_\Gamma & \text{on } \Gamma. \end{cases}$$

The Robin-Robin preconditioner defined by

$$\tilde{\mathcal{T}} := \frac{1}{4}(\tilde{\mathcal{S}}_1^{-1} + \tilde{\mathcal{S}}_2^{-1}) \tag{2.12}$$

is exact, $\tilde{\mathcal{T}}\mathcal{S}(0, \cdot) = \text{Id}$.

2.3.3 Numerical results

As examples we show 2D and 3D numerical results.

Consider a two dimensional flow with a velocity with a boundary layer near a wall

$$\begin{aligned} \mathbf{a} &= 3 - (300 * (x_2 - 0.1)^2) \mathbf{e}_1 \quad \text{if } x_2 < 0.1 \\ \mathbf{a} &= 3 \mathbf{e}_1 \quad \text{if } x_2 \geq 0.1. \end{aligned} \tag{2.13}$$

The computational domain is the unit square. To capture the boundary layer, the mesh is refined in the x_2 -direction, near the wall $x_2 = 0$ with a geometric progression of ratio 0.9. The advection-diffusion is discretized on a Cartesian grid by a Q1-streamline-diffusion method. The system for the nodal values at the interface is solved by a preconditioned GMRES algorithm, and the stopping criterion is to reduce the initial residual by a factor 10^{-10} . The preconditioners are either of the type Robin-Robin (R-R), Neumann-Neumann (N-N) or the identity (-).

Partition		4×1	8×1	12×1	24×1	36×1
Grid		20×40	20×40	20×40	20×40	20×40
$\Delta t = 1$	R-R	11	18	25	39	51
$\nu = 0.001$	-	51	75	91	> 100	> 100
	N-N	49	> 100	> 100	> 100	> 100

Table 1: Iteration counts

In the three dimensional case, we solved the convection diffusion problem, with $\mathbf{a} = (y/2 - 0.5, -x/2 + 0.5, 0)$, $\Delta t = 10$ and $\nu = 1$ on the unstructured decomposition of Fig. 4. Here the unit cube contains 24576 tetrahedric second order finite elements and is split into 45 subdomains by an automatic mesh partitioner. This is why the boundaries between subdomains are less regular than for the other computations. For this decomposition the algorithm converges in 48 iterations with the R-R preconditioner.

2.4 Generalities

Except for numerical results, we have considered so far very simple geometries. Of course, the ideas presented above are used for arbitrary decompositions: see e.g. [LT94] for the SPD case and [ATNV00] for the convection-diffusion equation.

For arbitrary decompositions, the Neumann-Neumann or the Robin-Robin preconditioners are no longer exact. A general theory has been developed for SPD scalar problems. In the case of the scalar Laplace operator, the main result is that the condition number of the preconditioned system is $O(1/H^2(1 + \log(H/h)^2))$, where h is a typical mesh size and H is a typical diameter of a subdomain. The term $\log(H/h)^2$ comes from multiple intersection points. The more problematic term $1/H^2$ comes from the lack of global exchange mechanism in the preconditioner in order to capture the ‘‘average’’ value of the solution. By adding a coarse grid preconditioner, see [LT94, CMW95], it is possible to improve the condition

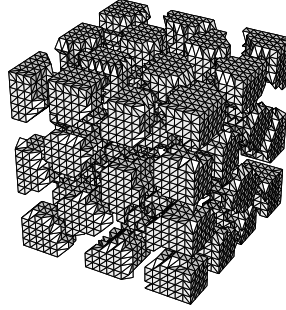


Figure 4: Three dimensional triangulation and automatic decomposition into 45 subdomains

number of the preconditioned system. Roughly speaking, the coarse grid preconditioner consists in decomposing the solution in each subdomain into an average value and its variation. Solving a global problem for these average values improves the convergence rate so that it is $O((1 + \log(H/h)^2))$. The iteration count is then almost mesh/decomposition independent.

3 Modified Schwarz method

The Additive Schwarz Method, (1.2), presents the drawback of needing overlapping subdomains in order to converge. In this chapter, we consider several improvements:

- replacement of the Dirichlet interface conditions by mixed interface conditions which yield convergence for non overlapping domain decompositions, see section 3.1;
- optimization of the interface conditions for faster convergence, see section 3.3;
- replacement of the fixed point iterative strategy of (1.2) by Krylov type methods, see section 3.4.1

3.1 A general convergence result

A major improvement of the ASM method comes from the use of other interface conditions. It has first been proposed by P. L. Lions to replace the Dirichlet interface conditions by Robin

interface conditions, see [Lio90]. Let α be a positive number; the modified algorithm reads:

$$\begin{aligned} -\Delta(u_1^{n+1}) &= f \quad \text{in } \Omega_1, \\ u_1^{n+1} &= 0 \quad \text{on } \partial\Omega_1 \cap \partial\Omega, \\ \left(\frac{\partial}{\partial n_1} + \alpha\right)(u_1^{n+1}) &= \left(-\frac{\partial}{\partial n_2} + \alpha\right)(u_2^n) \quad \text{on } \partial\Omega_1 \cap \overline{\Omega_2} \end{aligned}$$

(n_1 and n_2 are the outward normals on the boundary of the subdomains),

$$\begin{aligned} -\Delta(u_2^{n+1}) &= f \quad \text{in } \Omega_2, \\ u_2^{n+1} &= 0 \quad \text{on } \partial\Omega_2 \cap \partial\Omega, \\ \left(\frac{\partial}{\partial n_2} + \alpha\right)(u_2^{n+1}) &= \left(-\frac{\partial}{\partial n_1} + \alpha\right)(u_1^n) \quad \text{on } \partial\Omega_2 \cap \overline{\Omega_1}. \end{aligned}$$

The convergence proof given by P. L. Lions in the elliptic case was extended by B. Desprès [Des93] to the Helmholtz equation. A general presentation is given in [CGJ00]. We treat here the elliptic case with second order tangential derivatives in the interface conditions.

Let Ω be an open set. We consider the following problem: Find u such that

$$\begin{aligned} \eta(\mathbf{x})u - \operatorname{div}(\kappa(\mathbf{x})\nabla u) &= f \quad \text{in } \Omega, \\ u &= 0 \quad \text{on } \partial\Omega, \end{aligned}$$

where the functions $x \mapsto \eta(\mathbf{x}), \kappa(\mathbf{x})$ are bounded from below by a positive constant.

The domain is decomposed into N nonoverlapping subdomains $(\Omega_i)_{1 \leq i \leq N}$, $\overline{\Omega} = \bigcup_{i=1}^N \overline{\Omega}_i$ and $\overline{\Omega}_i \cap \overline{\Omega}_j = \emptyset$ for $i \neq j$. Let Γ_{ij} denote the interface $\Gamma_{ij} = \partial\Omega_i \cap \partial\Omega_j$, $i \neq j$. For two disjoint subdomains, $\Gamma_{ij} = \emptyset$.

For the sake of simplicity in writing the interface conditions, we consider the two dimensional case ($\Omega \subset \mathbf{R}^2$) although the proof is valid in arbitrary dimension. The interface conditions include second order tangential derivatives and have the form

$$\kappa(\mathbf{x})\frac{\partial}{\partial n_i} + \alpha_{ij}(\mathbf{x}) - \frac{\partial}{\partial \tau_i} \left(\beta_{ij}(\mathbf{x})\frac{\partial}{\partial \tau_i} \right),$$

where α_{ij} and β_{ij} are functions from Γ_{ij} into \mathbf{R} .

The algorithm reads:

$$\begin{aligned} \eta(\mathbf{x})u_i^{n+1} - \operatorname{div}(\kappa(\mathbf{x})\nabla u_i^{n+1}) &= f \quad \text{in } \Omega_i, \\ u_i^{n+1} &= 0 \quad \text{on } \partial\Omega \cap \partial\Omega_i \\ \kappa(\mathbf{x})\frac{\partial u_i^{n+1}}{\partial n_i} + \alpha_{ij}(\mathbf{x})u_i^{n+1} - \frac{\partial}{\partial \tau_i} \left(\beta_{ij}(\mathbf{x})\frac{\partial u_i^{n+1}}{\partial \tau_i} \right) & \\ = -\kappa(\mathbf{x})\frac{\partial u_j^n}{\partial n_j} + \alpha_{ij}(\mathbf{x})u_j^n - \frac{\partial}{\partial \tau_j} \left(\beta_{ij}(\mathbf{x})\frac{\partial u_j^n}{\partial \tau_j} \right) & \quad \text{on } \Gamma_{ij}. \end{aligned} \tag{3.1}$$

We make the following assumptions on the coefficients of the interface conditions:

$$\begin{aligned} \alpha_{ij}(\mathbf{x}) &= \alpha_{ji}(\mathbf{x}) \geq \alpha_0 > 0, \\ \beta(\mathbf{x})_{ij} &= \beta(\mathbf{x})_{ji} \geq 0 \quad \text{and} \quad \beta_{ij}(\mathbf{x}) = 0 \quad \text{on } \partial\Gamma_{ij} \end{aligned}$$

3.1 Theorem *With the above assumptions, algorithm (3.1) converges in H^1 , i.e.*

$$\lim_{n \rightarrow \infty} \|u_i^n - u_i\|_{H^1(\Omega_i)}, \quad \text{for } i = 1, \dots, N.$$

Proof Let us denote the operator

$$\Lambda_{ij} = \alpha_{ij}(\mathbf{x}) - \frac{\partial}{\partial \tau_i} \left(\beta_{ij}(\mathbf{x}) \frac{\partial}{\partial \tau_i} \right), \quad \mathbf{x} \in \Gamma_{ij}.$$

From the assumptions of the theorem, we have the following properties of Λ_{ij} :

- $\Lambda_{ij} = \Lambda_{ji}$;
- Λ_{ij} is SPD (symmetric positive definite);
- Λ_{ij} is invertible.

Therefore, Λ_{ij} has an invertible SPD square root, denoted by $\Lambda_{ij}^{1/2}$, whose inverse is denoted by $\Lambda_{ij}^{-1/2}$. These operators are SPD as well.

The interface condition is rewritten as

$$\Lambda_{ij}^{-1/2} \left(\kappa(\mathbf{x}) \frac{\partial u_i}{\partial n_i} \right) + \Lambda_{ij}^{1/2}(u_i) = -\Lambda_{ij}^{-1/2} \left(\kappa(\mathbf{x}) \frac{\partial u_j}{\partial n_j} \right) + \Lambda_{ij}^{1/2}(u_j) \quad \text{on } \Gamma_{ij}.$$

The proof follows the arguments given in [CGJ00] and is based on an energy estimate.

3.2 Lemma (Energy estimate) *Let u denote a function that satisfies*

$$\begin{aligned} \eta(\mathbf{x})u - \operatorname{div}(\kappa(\mathbf{x})\nabla u) &= 0 \quad \text{in } \Omega_i \\ u &= 0 \quad \text{on } \partial\Omega_i \cap \partial\Omega, \end{aligned}$$

Then

$$\begin{aligned} \int_{\Omega_i} \eta(\mathbf{x})|u_i|^2 + \kappa(\mathbf{x})|\nabla u_i|^2 + \frac{1}{4} \sum_{j \neq i} \int_{\partial\Gamma_{ij}} \left(\Lambda_{ij}^{-1/2} \left[\kappa(\mathbf{x}) \frac{\partial u_i}{\partial n_i} - \Lambda_{ij}(u_i) \right] \right)^2 \\ = \frac{1}{4} \sum_{j \neq i} \int_{\partial\Gamma_{ij}} \left(\Lambda_{ij}^{-1/2} \left[\kappa(\mathbf{x}) \frac{\partial u_i}{\partial n_i} + \Lambda_{ij}(u_i) \right] \right)^2. \end{aligned}$$

Proof From

$$\eta(\mathbf{x})u_i - \operatorname{div}(\kappa(\mathbf{x})\nabla u_i) = 0 \quad \text{in } \Omega_i,$$

we get

$$\begin{aligned} \int_{\Omega_i} \eta(\mathbf{x})|u_i|^2 + \kappa(\mathbf{x})|\nabla u_i|^2 &= \int_{\partial\Omega_i} \kappa(\mathbf{x}) \frac{\partial u_i}{\partial n_i} u_i \\ &= \sum_{j \neq i} \int_{\partial\Gamma_{ij}} \kappa(\mathbf{x}) \frac{\partial u_i}{\partial n_i} u_i \\ &= \sum_{j \neq i} \int_{\partial\Gamma_{ij}} \kappa(\mathbf{x}) \frac{\partial u_i}{\partial n_i} \Lambda_{ij}^{-1/2} \Lambda_{ij}^{1/2}(u_i) \\ &= \sum_{j \neq i} \int_{\partial\Gamma_{ij}} \Lambda_{ij}^{-1/2} \left(\kappa(\mathbf{x}) \frac{\partial u_i}{\partial n_i} \right) \Lambda_{ij}^{1/2}(u_i). \end{aligned}$$

From $ab = 1/4((a+b)^2 - (a-b)^2)$ we infer

$$\begin{aligned} \int_{\Omega_i} \eta(\mathbf{x})|u_i|^2 + \kappa(\mathbf{x})|\nabla u_i|^2 + \sum_{j \neq i} \frac{1}{4} \int_{\partial\Gamma_{ij}} \left(\Lambda_{ij}^{-1/2} \left(\kappa(\mathbf{x}) \frac{\partial u_i}{\partial n_i} \right) - \Lambda_{ij}^{1/2}(u_i) \right)^2 \\ = \sum_{j \neq i} \frac{1}{4} \int_{\partial\Gamma_{ij}} \left(\Lambda_{ij}^{-1/2} \left(\kappa(\mathbf{x}) \frac{\partial u_i}{\partial n_i} \right) + \Lambda_{ij}^{1/2}(u_i) \right)^2 \end{aligned}$$

□

Proof of Theorem 3.1 We prove that $e_i^n = u_i^n - u_{\Omega_i}$ converges to zero. By the linearity of the equations and of the algorithm, it is clear that the error e_i^n satisfies

$$\begin{aligned} \kappa(\mathbf{x})e_i^{n+1} - \operatorname{div}(\kappa(\mathbf{x})\nabla e_i^{n+1}) &= 0 \quad \text{in } \Omega_i, \\ e_i^{n+1} &= 0 \quad \text{on } \partial\Omega \cap \partial\Omega_i \\ \Lambda_{ij}^{-1/2} \left(\kappa(\mathbf{x}) \frac{\partial e_i^{n+1}}{\partial n_i} \right) + \Lambda_{ij}^{1/2}(e_i^{n+1}) &= -\Lambda_{ij}^{-1/2} \left(\kappa(\mathbf{x}) \frac{\partial e_j^n}{\partial n_j} \right) + \Lambda_{ij}^{1/2}(e_j^n) \quad \text{on } \Gamma_{ij}. \end{aligned}$$

We apply the energy estimate to e_i^{n+1} and taking into account the interface condition (3.1) and noticing that by assumption we have $\Lambda_{ij} = \Lambda_{ji}$, we get

$$\begin{aligned} \int_{\Omega_i} \eta(\mathbf{x})|e_i^{n+1}|^2 + \kappa(\mathbf{x})|\nabla e_i^{n+1}|^2 &= \sum_{j \neq i} \frac{1}{4} \int_{\partial\Gamma_{ij}} \left(\Lambda_{ji}^{-1/2} \left(-\kappa(\mathbf{x}) \frac{\partial e_j^n}{\partial n_j} \right) + \Lambda_{ji}^{1/2}(e_j^n) \right)^2 \\ &\quad - \left(\Lambda_{ij}^{-1/2} \left(\kappa(\mathbf{x}) \frac{\partial e_i^{n+1}}{\partial n_i} \right) - \Lambda_{ij}^{1/2}(e_i^{n+1}) \right)^2. \end{aligned}$$

We introduce some notations:

$$\mathcal{E}_i^{n+1} := \int_{\Omega_i} \eta(\mathbf{x})|u_i^{n+1}|^2 + \kappa(\mathbf{x})|\nabla u_i^{n+1}|^2,$$

and

$$\mathcal{C}_{ij}^{n+1} := \frac{1}{4} \int_{\partial\Gamma_{ij}} \left(\Lambda_{ij}^{-1/2} \left(\kappa(\mathbf{x}) \frac{\partial u_i^{n+1}}{\partial n_i} \right) - \Lambda_{ij}^{1/2}(u_i^{n+1}) \right)^2.$$

The above estimate then reads:

$$\mathcal{E}_i^{n+1} + \sum_{j \neq i} \mathcal{C}_{ij}^{n+1} = \sum_{j \neq i} \mathcal{C}_{ji}^n.$$

After summation over the subdomains, we have

$$\sum_{i=1}^N \mathcal{E}_i^{n+1} + \sum_{\substack{i,j \\ (j \neq i)}} \mathcal{C}_{ij}^{n+1} = \sum_{\substack{i,j \\ (j \neq i)}} \mathcal{C}_{ji}^n = \sum_{\substack{i,j \\ (j \neq i)}} \mathcal{C}_{ij}^n.$$

We introduce the further notations: $\mathcal{E}^{n+1} = \sum_{i=1}^N \mathcal{E}_i^{n+1}$ and $\mathcal{C}^n = \sum_{i,j(j \neq i)} \mathcal{C}_{ji}^n$.

So far we have

$$\mathcal{E}^{n+1} + \mathcal{C}^{n+1} = \mathcal{C}^n.$$

Hence, by summation over n , we get

$$\sum_{n=0}^{\infty} \mathcal{E}^{n+1} \leq \mathcal{C}^0.$$

The strong convergence of the algorithm in H^1 is proved. \square

The same kind of proof holds for the Maxwell system [DJR92] and the convection-diffusion equation [NR95].

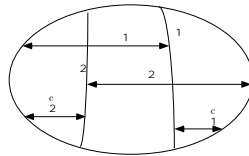
3.2 Optimal interface conditions

In the preceding section, we have proved a general convergence result for interface conditions with second order tangential derivatives. Actually these conditions are not the most general. Rather than give the general conditions in an *a priori* form, we shall derive them in this section so as to have the fastest convergence. We establish the existence of interface conditions which are optimal in terms of iteration counts. The corresponding interface conditions are pseudo-differential and are not practical. Nevertheless, this result is a guide for the choice of partial differential interface conditions. Moreover, this result establishes a link between the optimal interface conditions and artificial boundary conditions. This is also a help when dealing with the design of interface conditions since it gives the possibility to use the numerous papers and books published on the subject of artificial boundary conditions, see e.g. [EM77, Giv92].

We consider a general linear second order elliptic partial differential operator \mathcal{L} and the problem:

Find u such that $\mathcal{L}(u) = f$ in a domain Ω and $u = 0$ on $\partial\Omega$.

The domain Ω is decomposed into two subdomains Ω_1 and Ω_2 . We suppose that the problem is regular so that $u_i := u|_{\Omega_i}$, $i = 1, 2$, is continuous and has continuous normal derivatives across the interface $\Gamma_i = \partial\Omega_i \cap \bar{\Omega}_j$, $i \neq j$.



A modified Schwarz type method is considered.

$$\begin{aligned} \mathcal{L}(u_1^{n+1}) &= f & \text{in } \Omega_1 & & \mathcal{L}(u_2^{n+1}) &= f & \text{in } \Omega_2 \\ u_1^{n+1} &= 0 & \text{on } \partial\Omega_1 \cap \partial\Omega & & u_2^{n+1} &= 0 & \text{on } \partial\Omega_2 \cap \partial\Omega \\ \mu_1 \nabla u_1^{n+1} \cdot \mathbf{n}_1 + \mathcal{B}_1(u_1^{n+1}) & & & & \mu_2 \nabla u_2^{n+1} \cdot \mathbf{n}_2 + \mathcal{B}_2(u_2^{n+1}) & & \\ = -\mu_1 \nabla u_2^n \cdot \mathbf{n}_2 + \mathcal{B}_1(u_2^n) & \text{on } \Gamma_1 & & & = -\mu_2 \nabla u_1^n \cdot \mathbf{n}_1 + \mathcal{B}_2(u_1^n) & \text{on } \Gamma_2 \end{aligned} \quad (3.2)$$

where μ_1 and μ_2 are real-valued functions and \mathcal{B}_1 and \mathcal{B}_2 are operators acting along the interfaces Γ_1 and Γ_2 . For instance, $\mu_1 = \mu_2 = 0$ and $\mathcal{B}_1 = \mathcal{B}_2 = \text{Id}$ correspond to the ASM

algorithm (1.2); $\mu_1 = \mu_2 = 1$ and $\mathcal{B}_i = \alpha \in \mathbf{R}$, $i = 1, 2$, has been proposed in [Lio90] by P. L. Lions.

The question is:

Are there other possibilities in order to have convergence in a minimal number of steps?

In order to answer this question, we note that by linearity, the error e satisfies ($\mu_1 = \mu_2 = 1$)

$$\begin{aligned} \mathcal{L}(e_1^{n+1}) &= 0 & \text{in } \Omega_1 & & \mathcal{L}(e_2^{n+1}) &= 0 & \text{in } \Omega_2 \\ e_1^{n+1} &= 0 & \text{on } \partial\Omega_1 \cap \partial\Omega & & e_2^{n+1} &= 0 & \text{on } \partial\Omega_2 \cap \partial\Omega \\ \nabla e_1^{n+1} \cdot \mathbf{n}_1 + \mathcal{B}_1(e_1^{n+1}) & & & & \nabla e_2^{n+1} \cdot \mathbf{n}_2 + \mathcal{B}_2(e_2^{n+1}) & & \\ = -\nabla e_2^n \cdot \mathbf{n}_2 + \mathcal{B}_1(e_2^n) & \text{on } \Gamma_1 & & & = -\nabla e_1^n \cdot \mathbf{n}_1 + \mathcal{B}_2(e_1^n) & \text{on } \Gamma_2 \end{aligned}$$

The initial guess e_i^0 is arbitrary so that it is impossible to have convergence at step 1 of the algorithm. Convergence needs at least two iterations.

Having $e_1^2 \equiv 0$ requires

$$-\nabla e_2^1 \cdot \mathbf{n}_2 + \mathcal{B}_1(e_2^1) \equiv 0.$$

The only meaningful information on e_2^1 is that

$$\mathcal{L}(e_2^1) = 0 \quad \text{in } \Omega_2.$$

In order to use this information, we introduce the DtN (Dirichlet to Neumann) map (a.k.a. Steklov-Poincaré): Let

$$\begin{aligned} u_0 &: \Gamma_1 \rightarrow \mathbf{R} \\ \text{DtN}_2(u_0) &:= \nabla v \cdot \mathbf{n}_2|_{\partial\Omega_1 \cap \bar{\Omega}_2}, \end{aligned} \tag{3.3}$$

where \mathbf{n}_2 is the outward normal to $\Omega_2 \setminus \bar{\Omega}_1$, and v satisfies the following boundary value problem:

$$\begin{aligned} \mathcal{L}(v) &= 0 & \text{in } \Omega_2 \setminus \bar{\Omega}_1 \\ v &= 0 & \text{on } \partial\Omega_2 \cap \partial\Omega \\ v &= u_0 & \text{on } \partial\Omega_1 \cap \bar{\Omega}_2. \end{aligned}$$

We take

$$\mathcal{B}_1 := \text{DtN}_2.$$

This choice is optimal since we have

$$-\nabla e_2^1 \cdot \mathbf{n}_2 + \mathcal{B}_1(e_2^1) \equiv 0.$$

Indeed, in $\Omega_2 \setminus \bar{\Omega}_1 \subset \Omega_2$, e_2^1 satisfies

$$\mathcal{L}(e_2^1) = 0.$$

Hence,

$$\begin{aligned} \nabla e_2^1 \cdot \mathbf{n}_2 &= \text{DtN}_2(e_2^1) \\ \nabla e_2^1 \cdot \mathbf{n}_2 &= \mathcal{B}_1(e_2^1) \quad (\mathcal{B}_1 = \text{DtN}_2) \end{aligned}$$

We have formally proved

3.3 Result *The use of $\mathcal{B}_i = \text{DtN}_j$ ($i \neq j$) as interface conditions in (3.2) is optimal: we have (exact) convergence in two iterations.*

The two-domain case for an operator with constant coefficients has been first treated in [HTJ88]. The multidomain case for a variable coefficient operator with both positive results [NRdS94] and negative conjectures [Nie99] has been considered as well.

3.4 Remark The main feature of this result is to be very general since it does not depend on the exact form of the operator \mathcal{L} and can be extended to systems or to coupled systems of equations as well with a proper care of the well posedness of the algorithm.

As an application, we take $\Omega = \mathbf{R}^2$ and $\Omega_1 =]-\infty, 0[\times \mathbf{R}$. Using the same Fourier technique that was presented in section 2.3.1, it is possible to give the explicit form of the DtN operator for a constant coefficient operator. If $\mathcal{L} = \eta - \Delta$, the DtN map is a pseudo-differential operator whose symbol is

$$B_{i,\text{opt}}(k) = \sqrt{\eta + k^2},$$

i.e., $\mathcal{B}_{i,\text{opt}}(u)(0, y) = \int_{\mathbf{R}} B_{i,\text{opt}}(k) \hat{u}(0, k) e^{Iky} dk$.

If \mathcal{L} is a convection-diffusion operator $\mathcal{L} := \eta + \mathbf{a}\nabla - \nu\Delta$, the symbol of the DtN map is

$$B_{i,\text{opt}}(k) = \frac{-\mathbf{a}\cdot\mathbf{n}_i + \sqrt{(\mathbf{a}\cdot\mathbf{n}_i)^2 + 4\nu(\eta + \mathbf{a}\cdot\boldsymbol{\tau}_i k\nu + \nu^2 k^2)}}{2\nu}.$$

These symbols are not polynomials in the Fourier variable k so that the operators and hence the optimal interface conditions are not a partial differential operator. They correspond to exact absorbing conditions, see the contribution of L. Halpern in this volume. These conditions are used on the artificial boundary resulting from the truncation of a computational domain. On this boundary, boundary conditions have to be imposed. The solution on the truncated domain depends on the choice of this artificial condition. We say that it is an exact absorbing boundary condition if the solution computed on the truncated domain is the restriction of the solution of the original problem. Surprisingly enough, the notions of exact absorbing conditions for domain truncation and that of optimal interface conditions in domain decomposition methods coincide.

As the above examples show, they are pseudodifferential. Therefore they are difficult to implement. Moreover, in the general case of a variable coefficient operator and/or a curved boundary, the exact form of these operators is not known, although they can be approximated by partial differential operators which are easier to implement. The approximation of the DtN has been addressed by many authors since the seminal paper [EM77] by Engquist and Majda on this question.

3.3 Optimized interface conditions

The results obtained so far are quite general. In section 3.1, we have proved convergence of the domain decomposition method with interface conditions of the type

$$\partial_n + \alpha - \partial_\tau \beta \partial_\tau \tag{3.4}$$

for a general but non overlapping domain decomposition. In section 3.2, we have exhibited interface conditions which are optimal in terms of iteration counts but are pseudodifferential operators difficult to use in practice.

These results are not sufficient for the design of effective boundary conditions which for the sake of simplicity must have the form (3.4). From section 3.2, we know that the parameters α and β must somehow be such that (3.4) approximates the optimal interface conditions

$$\frac{\partial}{\partial n_i} + \text{DtN}.$$

At first sight, it seems that the approximations proposed in the field of artificial boundary conditions are also relevant in the context of domain decomposition methods. Actually this is not the case, as was proved for the convection-diffusion equation, see [JN00, JNR01].

In order to clarify the situation, we need an estimate of the convergence rate as a function of the parameters α and β , the size of the overlap and the coefficients of the partial differential operator. In particular it will provide a means for choosing the interface conditions in an optimal way. This type of study is limited to a very simple situation: a constant coefficient operator and a whole space decomposed into two half-spaces. But, let us insist on the fact that these limitations concern only this theoretical study. The optimized values of the parameters of the interface conditions can be used with success in complex applications, see section 3.4. The robustness of the approach comes from the general convergence result of section 3.1 and from the replacement of the fixed point algorithm on the interface by a Krylov type method as explained in section 3.4.1. The efficiency comes from the study below which is made possible by the use of Fourier techniques similar to the ones used in artificial boundary conditions. The method is general and has also been applied to other types of equations; see [EZ98] for the Laplace equation, [Che98] for the Maxwell system, and [WFNS98] for porous flow media.

We shall consider here the example of a symmetric positive definite problem

$$(\eta - \Delta)(u) = f \quad \text{in } \mathbf{R}^2,$$

$\eta = C^t > 0$. The domain is decomposed into two half-planes $\Omega_1 = (-\infty, \delta) \times \mathbf{R}$ and $\Omega_2 = (0, \infty) \times \mathbf{R}$. We introduce an optimization procedure which allows the choice of simplified interface conditions of the form $\partial_n + \alpha$ ($\beta = 0$) which are easy to implement and lead to a good convergence of the iterative method. We consider the Schwarz algorithm

$$\begin{aligned} (\eta - \Delta)(u_1^{n+1}) &= f(x, y), \quad (x, y) \in \Omega_1 \\ u_1^{n+1} &\text{ is bounded at infinity} \\ \left(\frac{\partial}{\partial n_1} + \alpha \right) (u_1^{n+1})(\delta, y) &= \left(-\frac{\partial}{\partial n_2} + \alpha \right) (u_2^n)(\delta, y), \quad y \in \mathbf{R} \end{aligned} \tag{3.5}$$

and

$$\begin{aligned} (\eta - \Delta)(u_2^{n+1}) &= f(x, y), \quad (x, y) \in \Omega_2 \\ u_2^{n+1} &\text{ is bounded at infinity} \\ \left(\frac{\partial}{\partial n_2} + \alpha \right) (u_2^{n+1})(0, y) &= \left(-\frac{\partial}{\partial n_1} + \alpha \right) (u_1^n)(0, y), \quad y \in \mathbf{R} \end{aligned} \tag{3.6}$$

and compute its convergence rate.

Computation of the convergence rate

We introduce the errors $u_i^n - u|_{\Omega_i}$, $i = 1, 2$. By linearity, the errors satisfy the above algorithm with $f = 0$:

$$\begin{aligned} (\eta - \Delta)(e_1^{n+1}) &= 0 \quad \text{in } \Omega_1 \\ e_1^{n+1} &\text{is bounded at infinity} \\ \left(\frac{\partial}{\partial n_1} + \alpha\right)(e_1^{n+1})(\delta, y) &= \left(-\frac{\partial}{\partial n_2} + \alpha\right)(e_2^n)(\delta, y), \end{aligned} \quad (3.7)$$

and

$$\begin{aligned} (\eta - \Delta)(e_2^{n+1}) &= 0 \quad \text{in } \Omega_2 \\ e_2^{n+1} &\text{is bounded at infinity} \\ \left(\frac{\partial}{\partial n_2} + \alpha\right)(e_2^{n+1})(0, y) &= \left(-\frac{\partial}{\partial n_1} + \alpha\right)(e_1^n)(0, y). \end{aligned} \quad (3.8)$$

By taking the partial Fourier transform of the first line of (3.7) in the y direction we get:

$$\left(\eta - \frac{\partial^2}{\partial x^2} + k^2\right)(\hat{e}_1^{n+1}(x, k)) = 0 \quad \text{in } \Omega_1.$$

For a given k , this is an ODE whose solution is sought in the form $\sum_j \gamma_j(k) \exp(\lambda_j(k)x)$. A simple calculation shows that there are two possible values for the lambdas:

$$\lambda^\pm(k) = \pm\sqrt{\eta + k^2}.$$

Therefore we have

$$\hat{e}_1^{n+1}(x, k) = \gamma_+^{n+1}(k) \exp(\lambda^+(k)x) + \gamma_-^{n+1}(k) \exp(\lambda^-(k)x).$$

From the second line of (3.7), the solution must be bounded at $x = -\infty$. This implies that $\gamma_-^{n+1}(k) \equiv 0$. Thus we have

$$\hat{e}_1^{n+1}(x, k) = \gamma_+^{n+1}(k) \exp(\lambda^+(k)x)$$

or equivalently, by changing the value of the coefficient γ_+ ,

$$\hat{e}_1^{n+1}(x, k) = \gamma_1^{n+1}(k) \exp(\lambda^+(k)(x - \delta))$$

and similarly,

$$\hat{e}_2^{n+1}(x, k) = \gamma_2^{n+1}(k) \exp(\lambda^+(k)x)$$

with $\gamma_{1,2}^{n+1}$ to be determined. From the interface conditions we get

$$\gamma_1^{n+1}(k)(\lambda^+ + \alpha) = \gamma_2^n(k)(\lambda^- + \alpha) \exp(\lambda^-(k)\delta)$$

and

$$\gamma_2^{n+1}(k)(-\lambda^- + \alpha) = \gamma_1^n(k)(-\lambda^+ + \alpha) \exp(-\lambda^+(k)\delta).$$

Combining these two and denoting $\lambda(k) = \lambda^+(k) = -\lambda^-(k)$, we get for $i = 1, 2$,

$$\gamma_i^{n+1}(k) = \rho(k; \alpha, \delta)^2 \gamma_i^{n-1}(k)$$

with

$$\rho(k; \alpha, \delta) = \left| \frac{\lambda(k) - \alpha}{\lambda(k) + \alpha} \right| \times \exp(-\lambda(k)\delta), \quad (3.9)$$

where $\lambda(k) = \sqrt{\eta + k^2}$ and $\alpha > 0$. This formula deserves a few remarks.

- For all $k \in \mathbf{R}$, $\rho(k) < 1$ so that $\gamma_i^n(k) \rightarrow 0$ as n goes to infinity.
- When domains overlap ($\delta > 0$), $\rho(k)$ is uniformly bounded from above by a constant smaller than one, $\rho(k; \alpha, \delta) < \exp(-\sqrt{\eta} \delta) < 1$ and $\rho \rightarrow 0$ as k tends to infinity.
- When there is no overlap ($\delta = 0$), $\rho \rightarrow 1$ as k tends to infinity.
- Let $\xi \in \mathbf{R}$. By taking $\alpha = \lambda(\xi)$, we have $\rho(\xi) = 0$.
- For the original Schwarz method (1.2), the convergence rate is $\exp(-\lambda(k)\delta)$. For $\delta = 0$ we see once again that there is no convergence. Replacing the Dirichlet interface conditions by Robin conditions enhances the convergence by a factor $|(\lambda(k) - \alpha)/(\lambda(k) + \alpha)|$.

Optimization of the interface condition

It is possible to optimize the choice of the parameter α in order to minimize the convergence rate in the physical space which is $\max_k \rho(k; \alpha, \delta)$.

When the subdomains overlap we have seen that the convergence rate is bounded from above by a positive constant so that it can be checked that the following min-max problem

$$\max_k \rho(k; \alpha_{\text{opt}}, \delta) = \min_{\alpha} \max_k \rho(k; \alpha, \delta)$$

admits a unique solution.

When the subdomains do not overlap, then for any choice of α we have $\max_k \rho(k; \alpha, 0) = 1$, so that the above min-max problem is ill-posed. Anyhow, the purpose of domain decomposition methods is not to solve partial differential equations. They are used to solve the corresponding linear systems arising from their discretizations. It is possible to study the convergence rate of the related domain decomposition methods at the discrete level based on the discretization scheme, see [Nat96]. Fourier transform is replaced by discrete Fourier series, i.e. the decomposition on the vectors $V_k = (e^{ij\Delta y k})_{j \in \mathbf{Z}}$, $k \in \pi/(\mathbf{Z}\Delta y)$ with Δy the mesh size in the y direction. The convergence rate depends as before on the parameters of the continuous problem but also on the discrete parameters: mesh size in x and y . The resulting formula is quite complex and would be very difficult to optimize.

Nevertheless, comparison with the continuous case and numerical experiments prove that a semi-continuous approach is sufficient for finding an optimal value for the parameter α . This of course due to the fact that as the discretization parameters go to zero, the discrete convergence rate tends to its continuous counterpart.

A semi continuous approach

For the sake of simplicity, we consider only the non-overlapping case, $\delta = 0$. We keep the formula of the convergence rate in the continuous case:

$$\rho(k; \alpha) := \left| \frac{\lambda(k) - \alpha}{\lambda(k) + \alpha} \right| \quad (3.10)$$

with $\lambda(k) = \sqrt{\eta + k^2}$. But we observe that the mesh induces a truncation in the frequency domain. We have $|k| < \pi/\Delta y := k_{\text{max}}$. For a parameter α , the convergence rate is approximated by

$$\rho_h(\alpha) = \max_{|k| < \pi/\Delta y} \rho(k; \alpha).$$

The optimization problem reads:

Find $\alpha_{\text{opt}}^{\text{sc}}$ such that

$$\rho_h(\alpha_{\text{opt}}^{\text{sc}}) = \min_{\alpha} \max_{k < \pi/\Delta y} \rho(k; \alpha). \quad (3.11)$$

It is easy to check that the optimum is given by the relation $\rho(0; \alpha_{\text{opt}}^{\text{sc}}) = \rho(k_{\text{max}}; \alpha_{\text{opt}}^{\text{sc}})$. Let $\lambda_m = \lambda(0)$ and $\lambda_M = \lambda(k_{\text{max}})$, we have

$$\alpha_{\text{opt}}^{\text{sc}} = \sqrt{\lambda_m \lambda_M}. \quad (3.12)$$

It can then easily be checked that in the limit of small Δy ,

$$\rho_h(\alpha_{\text{opt}}^{\text{sc}}) \simeq 1 - 2\sqrt{\frac{\sqrt{\eta}\Delta y}{\pi}}$$

and

$$\alpha_{\text{opt}}^{\text{sc}} \simeq \eta^{1/4} \frac{\pi}{\Delta y}.$$

Whereas for α independent of Δy , we have

$$\rho_h(\alpha) \simeq 1 - 2\frac{\alpha\Delta y}{\pi}$$

for small Δy . Numerical tests on the model problem of a rectangle divided into two half-rectangles and a finite difference discretization shows a good agreement with the above formulas. In Table 2, the iteration counts are given for two possible choices of the parameter α , $\alpha = 1$ or $\alpha_{\text{opt}}^{\text{sc}}$ given by formula (3.12). The reduction error factor is 10^{-6}

$1/\Delta y$	10	20	40	80
$\alpha_{\text{opt}}^{\text{sc}}$	6	7	10	16
$\alpha = 1$	27	51	104	231

Table 2: Number of iterations for different values of the mesh size and two possible choices for α

3.4 A more complex example: optimized interface conditions for the Helmholtz equation

This study is joint work with M. Gander and F. Magoulès, see [GMN01] for a complete presentation.

We consider the Helmholtz equation

$$\mathcal{L}(u) := (-\omega^2 - \Delta)(u) = f(x, y), \quad x, y \in \Omega.$$

The difficulty comes from the negative sign of the term of order zero of the operator.

Although the following analysis could be carried out on rectangular domains as well, we prefer for simplicity to present the analysis in the domain $\Omega = \mathbf{R}^2$ with the Sommerfeld radiation condition at infinity,

$$\lim_{r \rightarrow \infty} \sqrt{r} \left(\frac{\partial u}{\partial r} + i\omega u \right) = 0,$$

where $r = \sqrt{x^2 + y^2}$. We decompose the domain into two non-overlapping subdomains $\Omega_1 = (-\infty, 0] \times \mathbf{R}$ and $\Omega_2 = [0, \infty) \times \mathbf{R}$ and consider the Schwarz algorithm

$$\begin{aligned} -\Delta u_1^{n+1} - \omega^2 u_1^{n+1} &= f(x, y), \quad x, y \in \Omega_1 \\ \mathcal{B}_1(u_1^{n+1})(0) &= \mathcal{B}_1(u_2^n)(0) \end{aligned} \quad (3.13)$$

and

$$\begin{aligned} -\Delta u_2^{n+1} - \omega^2 u_2^{n+1} &= f(x, y), \quad x, y \in \Omega_2 \\ \mathcal{B}_2(u_2^{n+1})(0) &= \mathcal{B}_2(u_1^n)(0) \end{aligned} \quad (3.14)$$

where \mathcal{B}_j , $j = 1, 2$, are two linear operators. Note that for the classical Schwarz method \mathcal{B}_j is the identity, $\mathcal{B}_j = I$ and without overlap the algorithm cannot converge. But even with overlap in the case of the Helmholtz equation, only the evanescent modes in the error are damped, while the propagating modes are unaffected by the Schwarz algorithm [GMN01]. One possible remedy is to use a relatively fine coarse grid [CW92] or Robin transmission conditions, see for example [Des93, CCEW98]. We consider here a new type of transmission conditions which lead to a convergent non-overlapping version of the Schwarz method. We assume that the linear operators \mathcal{B}_j are of the form

$$\mathcal{B}_j := \partial_x + \mathcal{S}_j, \quad j = 1, 2,$$

for two linear operators \mathcal{S}_1 and \mathcal{S}_2 acting in the tangential direction on the interface. Our goal is to use these operators to optimize the convergence rate of the algorithm. For the analysis it suffices by linearity to consider the case $f(x, y) = 0$ and to analyze convergence to the zero solution. Taking a Fourier transform in the y direction we obtain

$$\begin{aligned} -\frac{\partial^2 \hat{u}_1^{n+1}}{\partial x^2} - (\omega^2 - k^2) \hat{u}_1^{n+1} &= 0, \\ (\partial_x + \sigma_1(k))(\hat{u}_1^{n+1})(0) &= (\partial_x + \sigma_1(k))(\hat{u}_2^n)(0) \end{aligned} \quad x < 0, \quad k \in \mathbf{R} \quad (3.15)$$

and

$$\begin{aligned} -\frac{\partial^2 \hat{u}_2^{n+1}}{\partial x^2} - (\omega^2 - k^2) \hat{u}_2^{n+1} &= 0, \\ (\partial_x + \sigma_2(k))(\hat{u}_2^{n+1})(0) &= (\partial_x + \sigma_2(k))(\hat{u}_1^n)(0) \end{aligned} \quad x > 0, \quad k \in \mathbf{R} \quad (3.16)$$

where $\sigma_j(k)$ denotes the symbol of the operator \mathcal{S}_j , and k is the Fourier variable, which we also call frequency. The general solutions of these ordinary differential equations are

$$\hat{u}_j^{n+1} = A_j e^{\lambda(k)x} + B_j e^{-\lambda(k)x}, \quad j = 1, 2,$$

where $\lambda(k)$ denotes the root of the characteristic equation $\lambda^2 + (\omega^2 - k^2) = 0$ with positive real or imaginary part,

$$\begin{aligned} \lambda(k) &= \sqrt{k^2 - \omega^2} \quad \text{for } |k| \geq \omega, \\ \lambda(k) &= i\sqrt{\omega^2 - k^2} \quad \text{for } |k| < \omega. \end{aligned} \quad (3.17)$$

Since the Sommerfeld radiation condition excludes growing solutions as well as incoming modes at infinity, we obtain the solutions

$$\begin{aligned}\hat{u}_1^{n+1}(x, k) &= \hat{u}_1^{n+1}(0, k)e^{\lambda(k)x} \\ \hat{u}_2^{n+1}(x, k) &= \hat{u}_2^{n+1}(0, k)e^{-\lambda(k)x}.\end{aligned}$$

Using the transmission conditions and the fact that

$$\begin{aligned}\frac{\partial \hat{u}_1^{n+1}}{\partial x} &= \lambda(k)\hat{u}_1^{n+1} \\ \frac{\partial \hat{u}_2^{n+1}}{\partial x} &= -\lambda(k)\hat{u}_2^{n+1}\end{aligned}$$

we obtain over one step of the Schwarz iteration

$$\begin{aligned}\hat{u}_1^{n+1}(x, k) &= \frac{-\lambda(k) + \sigma_1(k)}{\lambda(k) + \sigma_1(k)} e^{\lambda(k)x} \hat{u}_2^n(0, k) \\ \hat{u}_2^{n+1}(x, k) &= \frac{\lambda(k) + \sigma_2(k)}{-\lambda(k) + \sigma_2(k)} e^{-\lambda(k)x} \hat{u}_1^n(0, k).\end{aligned}$$

Evaluating the second equation at $x = 0$ for iteration index n and inserting it into the first equation, we get after evaluating again at $x = 0$

$$\hat{u}_1^{n+1}(0, k) = \frac{-\lambda(k) + \sigma_1(k)}{\lambda(k) + \sigma_1(k)} \cdot \frac{\lambda(k) + \sigma_2(k)}{-\lambda(k) + \sigma_2(k)} \hat{u}_1^{n-1}(0, k).$$

Defining the convergence rate ρ by

$$\rho(k) := \frac{-\lambda(k) + \sigma_1(k)}{\lambda(k) + \sigma_1(k)} \cdot \frac{\lambda(k) + \sigma_2(k)}{-\lambda(k) + \sigma_2(k)} \quad (3.18)$$

we find by induction that

$$\hat{u}_1^{2n}(0, k) = \rho(k)^n \hat{u}_1^0(0, k),$$

and by a similar calculation on the second subdomain,

$$\hat{u}_2^{2n}(0, k) = \rho(k)^n \hat{u}_2^0(0, k).$$

Choosing in the Fourier transformed domain

$$\sigma_1(k) := \lambda(k), \quad \sigma_2(k) := -\lambda(k)$$

corresponds to using exact absorbing boundary conditions as interface conditions. So we get $\rho(k) \equiv 0$ and the algorithm converges in two steps independently of the initial guess. Unfortunately this choice becomes difficult to use in the real domain where computations take place, since the optimal choice of the symbols $\sigma_j(k)$ leads to non-local operators \mathcal{S}_j in the real domain, caused by the square root in the symbols. We have to construct local approximations for the optimal transmission conditions.

In [EM77], the approximation valid for the truncation of an infinite computational domain is obtained via Taylor expansions of the symbol in the vicinity of $k = 0$:

$$\mathcal{S}_j^{\text{app}} = \pm i \left(\omega - \frac{1}{2\omega} \partial_{\tau\tau} \right),$$

which leads to the zeroth or second order Taylor transmission conditions, depending on whether one keeps only the constant term or also the second order term. But these transmission conditions are only effective for the low frequency components of the error. This is sufficient for the truncation of a domain since there is an exponential decay of the high frequency part (large k) of the solution away from the artificial boundary.

But in domain decomposition, what is important is the convergence rate which is given by the maximum over k of $\rho(k)$. Since there is no overlap between the subdomains, it is not possible to profit from any decay. We present now an approximation procedure suited to domain decomposition methods. To avoid an increase in the bandwidth of the local subproblems, we take polynomials of degree at most 2, which leads to transmission operators $\mathcal{S}_j^{\text{app}}$ which are at most second order partial differential operators acting along the interface. By symmetry of the Helmholtz equation there is no interest in a first order term. We therefore approximate the operators \mathcal{S}_j , $j = 1, 2$, in the form $\mathcal{S}_j^{\text{app}} = \pm(a + b\partial_{\tau\tau})$ with $a, b \in \mathbf{C}$ and where τ denotes the tangent direction at the interface.

Optimized Robin interface conditions for the Helmholtz equation We approximate the optimal operators \mathcal{S}_j , $j = 1, 2$, in the form

$$\mathcal{S}_j^{\text{app}} = \pm(p + qi), \quad p, q \in \mathbf{R}^+. \quad (3.19)$$

The non-negativity of p, q comes from the Shapiro-Lopatinski necessary condition for the well-posedness of the local subproblems (3.13)–(3.14). Inserting this approximation into the convergence rate (3.18) we find

$$\rho(p, q, k) = \begin{cases} \frac{p^2 + (q - \sqrt{\omega^2 - k^2})^2}{p^2 + (q + \sqrt{\omega^2 - k^2})^2}, & \omega^2 \geq k^2 \\ \frac{q^2 + (p - \sqrt{k^2 - \omega^2})^2}{q^2 + (p + \sqrt{k^2 - \omega^2})^2}, & \omega^2 < k^2. \end{cases} \quad (3.20)$$

First note that for $k^2 = \omega^2$ the convergence rate $\rho(p, q, \omega) = 1$, no matter what one chooses for the free parameters p and q . In the Helmholtz case one can not uniformly minimize the convergence rate over all relevant frequencies, as in the case of positive definite problems, see [Jap98, GMN01, JNR01]. The point $k = \omega$ represents however only one single mode in the spectrum, and a Krylov method will easily take care of this when the Schwarz method is used as a preconditioner, as our numerical experiments will show. We therefore consider the optimization problem

$$\min_{p, q \in \mathbf{R}^+} \left(\max_{k \in (k_{\min}, \omega_-) \cup (\omega_+, k_{\max})} |\rho(p, q, k)| \right), \quad (3.21)$$

where ω_- and ω_+ are parameters to be chosen, and k_{\min} denotes the smallest frequency relevant to the subdomain, and k_{\max} denotes the largest frequency supported by the numerical grid. This largest frequency is of the order π/h . For example, if the domain Ω is a strip of height L with homogeneous Dirichlet conditions on top and bottom, the solution can be expanded in a Fourier series with the harmonics $\sin(j\pi y/L)$, $j \in \mathbf{N}$. Hence the relevant

frequencies are $k = j\pi/L$. They are equally distributed with a spacing π/L and thus choosing $\omega_- = \omega - \pi/L$ and $\omega_+ = \omega + \pi/L$ leaves precisely one frequency $k = \omega$ for the Krylov method and treats all the others by the optimization. If ω falls in between the relevant frequencies, say $j\pi/L < \omega < (j+1)\pi/L$ then we can even get the iterative method to converge by choosing $\omega_- = j\pi/L$ and $\omega_+ = (j+1)\pi/L$, which will allow us to directly verify our asymptotic analysis numerically without the use of a Krylov method. How to choose the optimal parameters p and q is given by the following:

3.5 Theorem (Optimized Robin conditions) *Under the three assumptions*

$$2\omega^2 \leq \omega_-^2 + \omega_+^2, \quad \omega_- < \omega \quad (3.22)$$

$$2\omega^2 > k_{\min}^2 + \omega_+^2, \quad (3.23)$$

$$2\omega^2 < k_{\min}^2 + k_{\max}^2, \quad (3.24)$$

the solution to the min-max problem (3.21) is unique and the optimal parameters are given by

$$p^* = q^* = \sqrt{\frac{\sqrt{\omega^2 - \omega_-^2} \sqrt{k_{\max}^2 - \omega^2}}{2}}. \quad (3.25)$$

The optimized convergence rate (3.21) is then given by

$$\max_{k \in (k_{\min}, \omega_-) \cup (\omega_+, k_{\max})} \rho(p^*, q^*, k) = \frac{1 - \sqrt{2} \left(\frac{\omega^2 - \omega_-^2}{k_{\max}^2 - \omega^2} \right)^{1/4} + \sqrt{\frac{\omega^2 - \omega_-^2}{k_{\max}^2 - \omega^2}}}{1 + \sqrt{2} \left(\frac{\omega^2 - \omega_-^2}{k_{\max}^2 - \omega^2} \right)^{1/4} + \sqrt{\frac{\omega^2 - \omega_-^2}{k_{\max}^2 - \omega^2}}} \quad (3.26)$$

For the proof, see [GMN01].

Optimized second order interface conditions for the Helmholtz equation We seek interface conditions in the form $\mathcal{S}_j^{\text{app}} = \pm(a + b\partial_{\tau\tau})$ with $a, b \in \mathbf{C}$ and where τ denotes the tangent direction at the interface. The design of optimized second order interface conditions is simplified by the following:

3.6 Lemma *Let u_1 and u_2 be two functions which satisfy*

$$\mathcal{L}(u_j) \equiv (-\omega^2 - \Delta)(u) = f \quad \text{in } \Omega_j, \quad j = 1, 2,$$

and the interface condition

$$\left(\frac{\partial}{\partial n_1} + \alpha \right) \left(\frac{\partial}{\partial n_1} + \beta \right) (u_1) = \left(-\frac{\partial}{\partial n_2} + \alpha \right) \left(-\frac{\partial}{\partial n_2} + \beta \right) (u_2) \quad (3.27)$$

with $\alpha, \beta \in \mathbf{C}$, $\alpha + \beta \neq 0$, and n_j denoting the unit outward normal to domain Ω_j . Then the following second order interface condition is satisfied as well:

$$\left(\frac{\partial}{\partial n_1} + \frac{\alpha\beta - \omega^2}{\alpha + \beta} - \frac{1}{\alpha + \beta} \frac{\partial^2}{\partial \tau_1^2} \right) (u_1) = \left(-\frac{\partial}{\partial n_2} + \frac{\alpha\beta - \omega^2}{\alpha + \beta} - \frac{1}{\alpha + \beta} \frac{\partial^2}{\partial \tau_2^2} \right) (u_2) \quad (3.28)$$

Proof Expanding the interface condition (3.27) yields

$$\left(\frac{\partial^2}{\partial n_1^2} + (\alpha + \beta) \frac{\partial}{\partial n_1} + \alpha\beta \right) (u_1) = \left(\frac{\partial^2}{\partial n_2^2} - (\alpha + \beta) \frac{\partial}{\partial n_2} + \alpha\beta \right) (u_2).$$

Now using the equation $\mathcal{L}(u_1) = f$, we can substitute $-(\partial^2/\partial\tau_1^2 + \omega^2)(u_1) - f$ for $\partial^2/\partial n_1^2(u_1)$, and similarly we can substitute $-(\partial^2/\partial\tau_2^2 + \omega^2)(u_2) - f$ for $\partial^2/\partial n_2^2(u_2)$. Hence we get

$$\left(-\frac{\partial^2}{\partial\tau_1^2} - \omega^2 + (\alpha + \beta) \frac{\partial}{\partial n_1} + \alpha\beta \right) (u_1) - f = \left(-\frac{\partial^2}{\partial\tau_2^2} - \omega^2 - (\alpha + \beta) \frac{\partial}{\partial n_2} + \alpha\beta \right) (u_2) - f.$$

Now the terms f on both sides cancel, and division by $\alpha + \beta$ yields (3.28). \square

Note that Higdon has already proposed approximations to absorbing boundary conditions in factored form in [Hig86]. In our case, this special choice of approximating $\sigma_j(k)$ by

$$\sigma_j^{\text{app}}(k) := \pm \left(\frac{\alpha\beta - \omega^2}{\alpha + \beta} + \frac{1}{\alpha + \beta} k^2 \right) \quad (3.29)$$

leads to a particularly elegant formula for the convergence rate. Inserting $\sigma_j^{\text{app}}(k)$ into the convergence rate (3.18) and simplifying, we obtain

$$\begin{aligned} \rho(k; \alpha, \beta) &:= \left(\frac{-\lambda(k) - \sigma_1}{\lambda(k) + \sigma_1} \right)^2 = \left(\frac{-(\alpha + \beta)\lambda(k) + \alpha\beta + k^2 - \omega^2}{(\alpha + \beta)\lambda(k) + \alpha\beta + k^2 - \omega^2} \right)^2 \\ &= \left(\frac{\lambda(k)^2 - (\alpha + \beta)\lambda(k) + \alpha\beta}{\lambda(k)^2 + (\alpha + \beta)\lambda(k) + \alpha\beta} \right)^2 \\ &= \left(\frac{\lambda(k) - \alpha}{\lambda(k) + \alpha} \right)^2 \left(\frac{\lambda(k) - \beta}{\lambda(k) + \beta} \right)^2 \end{aligned} \quad (3.30)$$

where $\lambda(k)$ is defined in (3.17), and the two parameters $\alpha, \beta \in \mathbf{C}$ can be used to optimize the performance. By the symmetry of $\lambda(k)$ with respect to k , it suffices to consider only positive k to optimize performance. We thus need to solve the min-max problem

$$\min_{\alpha, \beta \in \mathbf{C}} \left(\max_{k \in (k_{\min}, \omega_-) \cup (\omega_+, k_{\max})} |\rho(k; \alpha, \beta)| \right), \quad (3.31)$$

where ω_- and ω_+ are again the parameters to exclude the frequency $k = \omega$ where the convergence rate equals 1, as in the zeroth order optimization problem. The convergence rate $\rho(k; \alpha, \beta)$ consists of two factors, and λ is real for vanishing modes and imaginary for propagative modes. If we chose $\alpha \in i\mathbf{R}$ and $\beta \in \mathbf{R}$ then for λ real the first factor is of modulus one and the second one can be optimized using β . If λ is imaginary, then the second factor is of modulus one and the first one can be optimized independently using α . Hence for this choice of α and β the min-max problem decouples. We therefore consider here the simpler min-max problem

$$\min_{\alpha \in i\mathbf{R}, \beta \in \mathbf{R}} \left(\max_{k \in (k_{\min}, \omega_-) \cup (\omega_+, k_{\max})} |\rho(k; \alpha, \beta)| \right) \quad (3.32)$$

which has an elegant analytical solution. Note however that the original minimization problem (3.31) might have a solution with better convergence rate, an issue investigated in [GMN01].

3.7 Theorem (Optimized second order conditions) *The solution of the min-max problem (3.32) is unique and the optimal parameters are given by*

$$\alpha^* = i((\omega^2 - k_{\min}^2)(\omega^2 - \omega_-^2))^{1/4} \in i\mathbf{R} \quad (3.33)$$

and

$$\beta^* = ((k_{\max}^2 - \omega^2)(\omega_+^2 - \omega^2))^{1/4} \in \mathbf{R}. \quad (3.34)$$

The convergence rate (3.32) is then for the propagating modes given by

$$\max_{k \in (k_{\min}, \omega_-)} |\rho(k, \alpha^*, \beta^*)| = \left(\frac{(\omega^2 - \omega_-^2)^{1/4} - (\omega^2 - k_{\min}^2)^{1/4}}{(\omega^2 - \omega_-^2)^{1/4} + (\omega^2 - k_{\min}^2)^{1/4}} \right)^2, \quad (3.35)$$

and for the evanescent modes it is

$$\max_{k \in (\omega_+, k_{\max})} \rho(k, \alpha^*, \beta^*) = \left(\frac{(k_{\max}^2 - \omega^2)^{1/4} - (\omega_+^2 - \omega^2)^{1/4}}{(k_{\max}^2 - \omega^2)^{1/4} + (\omega_+^2 - \omega^2)^{1/4}} \right)^2. \quad (3.36)$$

Proof For $k \in (k_{\min}, \omega_-)$ we have

$$\left| \frac{i\sqrt{\omega^2 - k^2} - \beta}{i\sqrt{\omega^2 - k^2} + \beta} \right| = 1$$

since $\beta \in \mathbf{R}$ and thus

$$|\rho(k; \alpha, \beta)| = \left| \frac{i\sqrt{\omega^2 - k^2} - \alpha}{i\sqrt{\omega^2 - k^2} + \alpha} \right|^2$$

depends only on α . Similarly, for $k \in (\omega_+, k_{\max})$ we have

$$\left| \frac{\sqrt{k^2 - \omega^2} - \alpha}{\sqrt{k^2 - \omega^2} + \alpha} \right| = 1$$

since $\alpha \in i\mathbf{R}$, and therefore

$$|\rho(k; \alpha, \beta)| = \left| \frac{\sqrt{k^2 - \omega^2} - \beta}{\sqrt{k^2 - \omega^2} + \beta} \right|^2$$

depends only on β . The solution (α, β) of the minimization problem (3.32) is thus given by the solution of the two independent minimization problems

$$\min_{\alpha \in i\mathbf{R}} \left(\max_{k \in (k_{\min}, \omega_-)} \left| \frac{i\sqrt{\omega^2 - k^2} - \alpha}{i\sqrt{\omega^2 - k^2} + \alpha} \right| \right) \quad (3.37)$$

and

$$\min_{\beta \in \mathbf{R}} \left(\max_{k \in (\omega_+, k_{\max})} \left| \frac{\sqrt{k^2 - \omega^2} - \beta}{\sqrt{k^2 - \omega^2} + \beta} \right| \right). \quad (3.38)$$

We show the solution for the second problem (3.38) only, the solution for the first problem (3.37) is similar. First note that the maximum of

$$|\rho_\beta| := \left| \frac{\sqrt{k^2 - \omega^2} - \beta}{\sqrt{k^2 - \omega^2} + \beta} \right|$$

is attained on the boundary of the interval $[\omega_+, k_{\max}]$, because the function ρ_β (but not $|\rho_\beta|$) is monotone increasing with $k \in [\omega_+, k_{\max}]$. On the other hand, as a function of β , $|\rho_\beta(\omega_+)|$ grows monotonically with β while $|\rho_\beta(k_{\max})|$ decreases monotonically with β . The optimum is therefore reached when we balance the two values on the boundary, $\rho_\beta(\omega_+) = -\rho_\beta(k_{\max})$, which implies that the optimal β satisfies the equation

$$\frac{\sqrt{k_{\max}^2 - \omega^2} - \beta}{\sqrt{k_{\max}^2 - \omega^2} + \beta} = -\frac{\sqrt{\omega_+^2 - \omega^2} - \beta}{\sqrt{\omega_+^2 - \omega^2} + \beta} \quad (3.39)$$

whose solution is given in (3.34). \square

The optimization problem (3.38) arises also for symmetric positive definite problems when an optimized Schwarz algorithm without overlap and Robin transmission conditions are used and the present solution can be found in [WFNS98].

Fig. 5 shows the convergence rate obtained for a model problem on the unit square with two subdomains, $\omega = 10\pi$ and $h = 1/50$. The optimal parameters were found to be

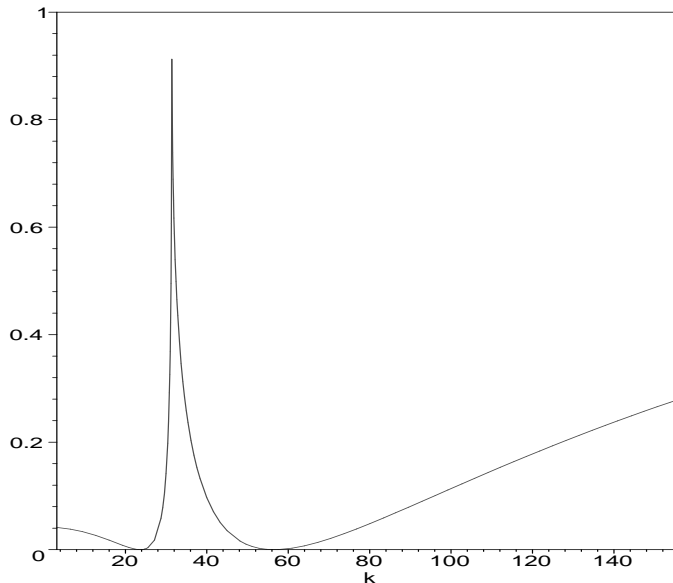


Figure 5: Convergence rate of the optimized Schwarz method with second order transmission conditions in Fourier space for $\omega = 10\pi$

$\alpha^* = 20.741i$ and $\beta^* = 47.071$, which gives a convergence rate $\rho = 0.0419$ for the propagating modes and $\rho = 0.2826$ for the evanescent modes. It is interesting to note that with the current

practice in engineering of choosing about 10 grid points per wavelength, we have $h \approx \pi/(5\omega)$, and thus for the propagating modes the optimized Schwarz method presented here has an asymptotic convergence rate of

$$\rho_p = 1 - O(h^{1/4}).$$

3.4.1 Numerical implementation and acceleration via Krylov type methods

This section is concerned with the Finite Element implementation of the interface conditions of Robin type and of the ones with second order tangential derivatives along the interface. We show that thanks to a reformulation of the algorithm they are as easy to implement as Neumann boundary conditions. We first treat the case of a decomposition into two subdomains and then an arbitrary decomposition of the domain.

Two-domain decomposition

We present the discretization scheme for a decomposition of a domain Ω into two subdomains Ω_1 and Ω_2 with interface Γ_{12} . So far, we have considered the optimized Schwarz algorithm at the continuous level,

$$\begin{aligned} -\Delta u_1^{n+1} - \omega^2 u_1^{n+1} &= f_1 && \text{in } \Omega_1 \\ \frac{\partial u_1^{n+1}}{\partial n_1} + \mathcal{S}_1^{\text{app}}(u_1^{n+1}) &= -\frac{\partial u_2^n}{\partial n_2} + \mathcal{S}_1^{\text{app}}(u_2^n) && \text{on } \Gamma_{12} \\ -\Delta u_2^{n+1} - \omega^2 u_2^{n+1} &= f_2 && \text{in } \Omega_2 \\ \frac{\partial u_2^{n+1}}{\partial n_2} + \mathcal{S}_2^{\text{app}}(u_2^{n+1}) &= -\frac{\partial u_1^n}{\partial n_1} + \mathcal{S}_2^{\text{app}}(u_1^n) && \text{on } \Gamma_{12}. \end{aligned} \quad (3.40)$$

A direct discretization would require the computation of the normal derivatives along the interfaces in order to evaluate the right hand sides in the transmission conditions of (3.40). This can be avoided by introducing two new variables,

$$\lambda_1^n = -\frac{\partial u_2^n}{\partial n_2} + \mathcal{S}_1^{\text{app}}(u_2^n) \quad \text{and} \quad \lambda_2^n = -\frac{\partial u_1^n}{\partial n_1} + \mathcal{S}_2^{\text{app}}(u_1^n).$$

The algorithm then becomes

$$\begin{aligned} -\Delta u_1^{n+1} - \omega^2 u_1^{n+1} &= f_1 && \text{in } \Omega_1 \\ \frac{\partial u_1^{n+1}}{\partial n_1} + \mathcal{S}_1^{\text{app}}(u_1^{n+1}) &= \lambda_1^n && \text{on } \Gamma_{12} \\ -\Delta u_2^{n+1} - \omega^2 u_2^{n+1} &= f_2 && \text{in } \Omega_2 \\ \frac{\partial u_2^{n+1}}{\partial n_2} + \mathcal{S}_2^{\text{app}}(u_2^{n+1}) &= \lambda_2^n && \text{on } \Gamma_{12} \end{aligned} \quad (3.41)$$

$$\begin{aligned} \lambda_1^{n+1} &= -\lambda_2^n + (\mathcal{S}_1^{\text{app}} + \mathcal{S}_2^{\text{app}})(u_2^{n+1}) \\ \lambda_2^{n+1} &= -\lambda_1^n + (\mathcal{S}_1^{\text{app}} + \mathcal{S}_2^{\text{app}})(u_1^{n+1}). \end{aligned}$$

We can interpret this new algorithm as a fixed point algorithm in the new variables λ_j , $j = 1, 2$, to solve the substructured problem

$$\begin{aligned} \lambda_1 &= -\lambda_2 + (\mathcal{S}_1^{\text{app}} + \mathcal{S}_2^{\text{app}})(u_2(\lambda_2, f_2)), \\ \lambda_2 &= -\lambda_1 + (\mathcal{S}_1^{\text{app}} + \mathcal{S}_2^{\text{app}})(u_1(\lambda_1, f_1)), \end{aligned} \quad (3.42)$$

where $u_j = u_j(\lambda_j, f_j)$, $j = 1, 2$, are solutions of

$$\begin{aligned} -\Delta u_j - \omega^2 u_j &= f_j \quad \text{in } \Omega_j, \\ \frac{\partial u_j}{\partial n_j} + \mathcal{S}_j^{\text{app}}(u_j) &= \lambda_j \quad \text{on } \Gamma_{12}. \end{aligned}$$

Instead of solving the substructured problem (3.42) by the fixed point iteration (3.41), one usually uses a Krylov subspace method to solve the substructured problem directly. This corresponds to using the optimized Schwarz method as a preconditioner for the Krylov subspace method. A finite element discretization of the substructured problem (3.42) leads to the linear system

$$\begin{aligned} \lambda_1 &= -\lambda_2 + (S_1 + S_2)B_2 u_2 \\ \lambda_2 &= -\lambda_1 + (S_1 + S_2)B_1 u_1 \\ \tilde{K}_1 u_1 &= f_1 + B_1^T \lambda_1 \\ \tilde{K}_2 u_2 &= f_2 + B_2^T \lambda_2 \end{aligned} \tag{3.43}$$

where B_1 and B_2 are the trace operators of the domains Ω_1 and Ω_2 on the interface Γ_{12} , and we omit the superscript *app* in the discretization S_j of the continuous operators $\mathcal{S}_j^{\text{app}}$ to reduce the notation. If the two vectors u_1 and u_2 containing the degrees of freedom have their first components corresponding to the interior unknowns

$$u_j = \begin{bmatrix} u_j^i \\ u_j^b \end{bmatrix}, \quad j = 1, 2, \tag{3.44}$$

where the indices i and b correspond to interior and interface degrees of freedom respectively for domain Ω_j , then the discrete trace operators B_1 and B_2 are just the boolean matrices corresponding to the decomposition (3.44) and they can be written as

$$B_j = \begin{bmatrix} 0 & I \end{bmatrix}, \quad j = 1, 2, \tag{3.45}$$

where I denotes the identity matrix of appropriate size. For example, $B_1 u_1 = u_1^b$ and $B_2 u_2 = u_2^b$. The matrices \tilde{K}_1 and \tilde{K}_2 arise from the discretization of the local Helmholtz subproblems along with the interface conditions $\partial_n + a - b\partial_{\tau\tau}$,

$$\tilde{K}_j = K_j - \omega^2 M_j + B_j^T (aM_{\Gamma_{12}} + bK_{\Gamma_{12}}) B_j, \quad j = 1, 2. \tag{3.46}$$

Here K_1 and K_2 are the stiffness matrices, M_1 and M_2 are the mass matrices, $M_{\Gamma_{12}}$ is the interface mass matrix, and $K_{\Gamma_{12}}$ is the interface stiffness matrix,

$$[M_{\Gamma_{12}}]_{nm} = \int_{\Gamma_{12}} \phi_n \phi_m d\xi \quad \text{and} \quad [K_{\Gamma_{12}}]_{nm} = \int_{\Gamma_{12}} \nabla_{\tau} \phi_n \nabla_{\tau} \phi_m d\xi. \tag{3.47}$$

The functions ϕ_n and ϕ_m are the basis functions associated with the degrees of freedom n and m on the interface Γ_{12} , and $\nabla_{\tau} \phi$ is the tangential component of $\nabla \phi$ on the interface. We have

$$S_j = aM_{\Gamma_{12}} + bK_{\Gamma_{12}}, \quad j = 1, 2.$$

For given λ_1 and λ_2 , the acoustic pressure u_1 and u_2 can be computed by solving the last two equations of (3.43). Eliminating u_1 and u_2 in the first two equations of (3.43) using the last two equations of (3.43), we obtain the substructured linear system

$$F\lambda = d, \quad (3.48)$$

where $\lambda = (\lambda_1, \lambda_2)$ and the matrix F and the right hand side d are given by

$$\begin{aligned} F &= \begin{pmatrix} I & I - (S_1 + S_2)B_2\tilde{K}_2^{-1}B_2^T \\ I - (S_1 + S_2)B_1\tilde{K}_1^{-1}B_1^T & I \end{pmatrix} \\ d &= \begin{pmatrix} (S_1 + S_2)B_1\tilde{K}_1^{-1}f_1 \\ (S_1 + S_2)B_2\tilde{K}_2^{-1}f_2 \end{pmatrix} \end{aligned} \quad (3.49)$$

The linear system (3.48) is solved by a Krylov subspace method. The matrix vector product amounts to solving a subproblem in each subdomain and to send interface data between subdomains. Note that the optimization of the interface conditions was performed for the convergence rate of the additive Schwarz method and not for a particular Krylov method applied to the substructured problem. In the positive definite case one can show that minimizing the convergence rate is equivalent to minimizing the condition number of the substructured problem [JN00]. Numerical experiments in the next section indicate that for the Helmholtz equation our optimization also leads to parameters close to the best ones for the preconditioned Krylov method.

The general case of a decomposition into an arbitrary number of subdomains is treated in [GMN01].

Numerical results

We present two sets of numerical experiments. The first set corresponds to the model problem analyzed in this paper and the results obtained illustrate the analysis and confirm the asymptotic convergence results. The second numerical experiment comes from industry and consists of analyzing the noise levels in the interior of a VOLVO S90.

Model problem

We study a two dimensional cavity on the unit square Ω with homogeneous Dirichlet conditions on top and bottom and on the left and right radiation conditions of Robin type. We thus have the Helmholtz problem

$$\begin{aligned} -\Delta u - \omega^2 u &= f & 0 < x, y < 1 \\ u &= 0 & 0 < x < 1, y = 0, 1 \\ \frac{\partial u}{\partial x} - i\omega u &= 0 & x = 0, 0 < y < 1 \\ -\frac{\partial u}{\partial x} - i\omega u &= 0 & x = 1, 0 < y < 1. \end{aligned} \quad (3.50)$$

We decompose the unit square into two subdomains of equal size, and we use a uniform rectangular mesh for the discretization. We perform all our experiments directly on the error

equations, $f = 0$ and choose the initial guess of the Schwarz iteration so that all the frequencies are present in the error. We show two sets of experiments: The first one with $\omega = 9.5\pi$, thus excluding ω from the frequencies k relevant in this setting, $k = n\pi$, $n = 1, 2, \dots$. This allows us to test directly the iterative Schwarz method, since with optimization parameters $\omega_- = 9\pi$ and $\omega_+ = 10\pi$ we obtain a convergence rate which is uniformly less than one for all k . Table 3 shows the number of iterations needed for different values of the mesh parameter h for both the zeroth and second order transmission conditions. The Taylor transmission

h	Order Zero				Order Two			
	Iterative		Krylov		Iterative		Krylov	
	Taylor	Optimized	Taylor	Optimized	Taylor	Optimized	Taylor	Optimized
1/50	-	457	26	16	-	22	28	9
1/100	-	126	34	21	-	26	33	10
1/200	-	153	44	26	-	36	40	13
1/400	-	215	57	34	-	50	50	15
1/800	-	308	72	43	-	71	61	19

Table 3: Number of iterations for different transmission conditions and different mesh parameter for the model problem

conditions do not lead to a convergent iterative algorithm, because for all frequencies $k > \omega$, the convergence rate equals 1. However, with Krylov acceleration, GMRES in this case, the methods converge. Note however that the second order Taylor condition is only a little better than the zeroth order Taylor conditions. The optimized transmission conditions lead, in the case where ω lies between two frequencies, already to a convergent iterative algorithm. The iterative version even beats the Krylov accelerated Taylor conditions in the second order case. No wonder that the optimized conditions lead by far to the best algorithms when they are accelerated by a Krylov method, the second order optimized Schwarz method is more than a factor three faster than any Taylor method. Note that the only difference in cost of the various transmission conditions consists of different entries in the interface matrices, without enlarging the bandwidth of the matrices. Fig. 6 shows the asymptotic behavior of the methods considered, on the left for zeroth order conditions and on the right for second order conditions. Note that the scale on the right for the second order transmission conditions is different by an order of magnitude. In both cases the asymptotic analysis is confirmed for the iterative version of the optimized methods. In addition one can see that the Krylov method improves the asymptotic rate by almost an additional square root, as expected from the analysis in ideal situations. Note the outlier of the zeroth order optimized transmission condition for $h = 1/50$. It is due to the discrepancy between the spectrum of the continuous and the discrete operator: $\omega = 9.5\pi$ lies precisely in between two frequencies 9π and 10π at the continuous level, but for the discrete Laplacian with $h = 1/50$ this spectrum is shifted to 8.88π and 9.84π and thus the frequency 9.84π falls into the range $[9\pi, 10\pi]$ neglected by the optimization. Note however that this is of no importance when Krylov acceleration is used, so it is not worthwhile to consider this issue further. Now we put ω directly onto a frequency of the model problem, $\omega = 10\pi$, so that the iterative methods cannot be considered any more, since for that frequency the convergence rate equals one. The Krylov accelerated versions however are not affected by this, as one can see in Table 4. The number of iterations does not differ from the case where ω was chosen to lie between two frequencies, which shows

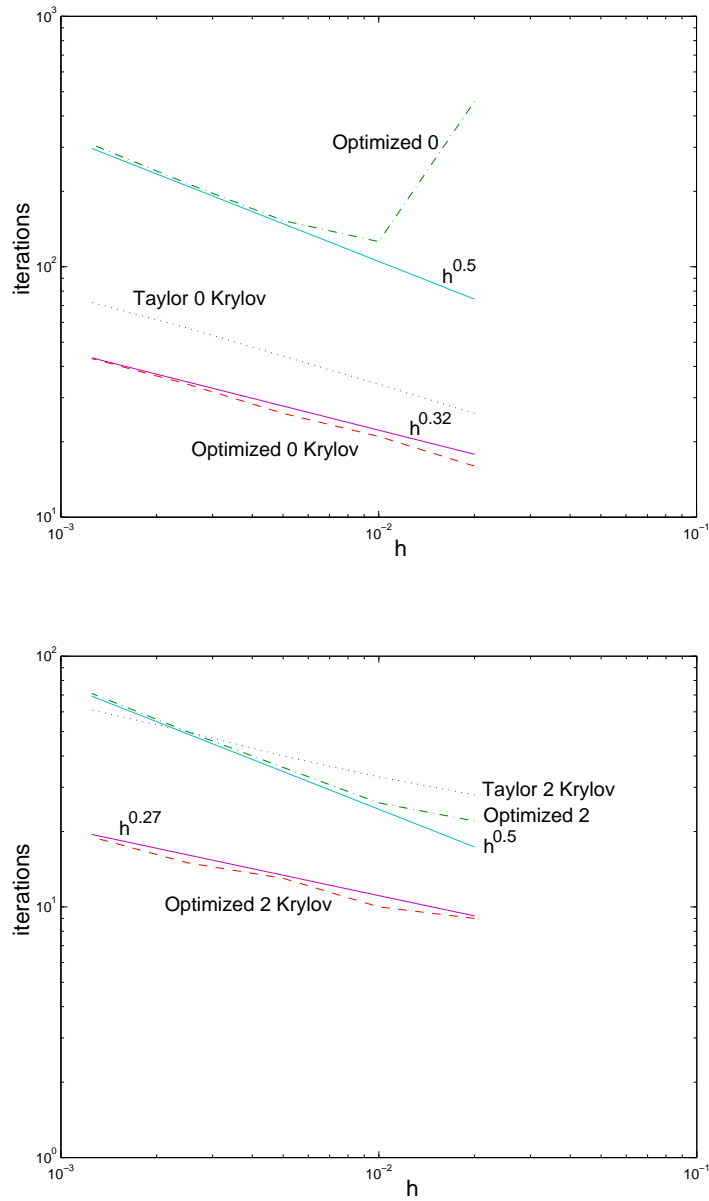


Figure 6: Asymptotic behavior for the zeroth order transmission conditions (above) and for the second order transmission conditions (below)

h	Order Zero		Order Two	
	Taylor	Optimized	Taylor	Optimized
1/50	24	15	27	9
1/100	35	21	35	11
1/200	44	26	41	13
1/400	56	33	52	16
1/800	73	43	65	20

Table 4: Number of iterations for different transmission conditions and different mesh parameter for the model problem when ω lies precisely on a frequency of the problem and thus Krylov acceleration is mandatory

that with Krylov acceleration the method is robust for any values of ω . We finally tested for the smallest resolution of the model problem how well Fourier analysis predicts the optimal parameters to use. Since we want to test both the iterative and the Krylov versions, we need to put again the frequency ω in between two problem frequencies, and in this case it is important to be precise. We therefore choose ω to be exactly between two frequencies of the discrete problem, $\omega = 9.3596\pi$, and optimized using $\omega_- = 8.8806\pi$ and $\omega_+ = 9.8363\pi$. Fig. 7 shows the number of iterations the algorithm needs to achieve a residual of $10e-6$ as a function of the optimization parameters p and q of the zeroth order transmission conditions, on the left in the iterative version and on the right for the Krylov accelerated version. The Fourier analysis shows well where the optimal parameters lie and when a Krylov method is used, the optimized Schwarz method is very robust with respect to the choice of the optimization parameter. The same holds also for the second order transmission conditions, as Fig. 8 shows.

Noise levels in a VOLVO S90

We analyze the noise level distribution in the passenger cabin of a VOLVO S90. The vibrations are stemming from the part of the car called firewall. This example is representative for a large class of industrial problems where one tries to determine the acoustic response in the interior of a cavity caused by vibrating parts. We perform a two dimensional simulation on a vertical cross section of the car. Fig. 9 shows the decomposition of the car into 16 subdomains. The computations were performed in parallel on a network of sun workstations with 4 processors.

The problem is characterized by $\omega a = 18.46$ which corresponds to a frequency of 1000 Hz in the car of length a . To solve the problem, the optimized Schwarz method was used as a preconditioner for the Krylov method ORTHODIR, and as convergence criterion we used

$$\|\tilde{K}u - f\|_{L_2} \leq 10^{-6}\|f\|_{L_2}. \quad (3.51)$$

When using zeroth order Taylor conditions and a decomposition into 16 subdomains, the method needed 105 iterations to converge, whereas when using second order optimized transmission conditions, the method converged in 34 iterations, confirming that also in real applications the optimized Schwarz method is about a factor 3 faster, as we found for the model problem earlier. Fig. 10 shows the acoustic field obtained in the passenger compartment of the VOLVO S90.

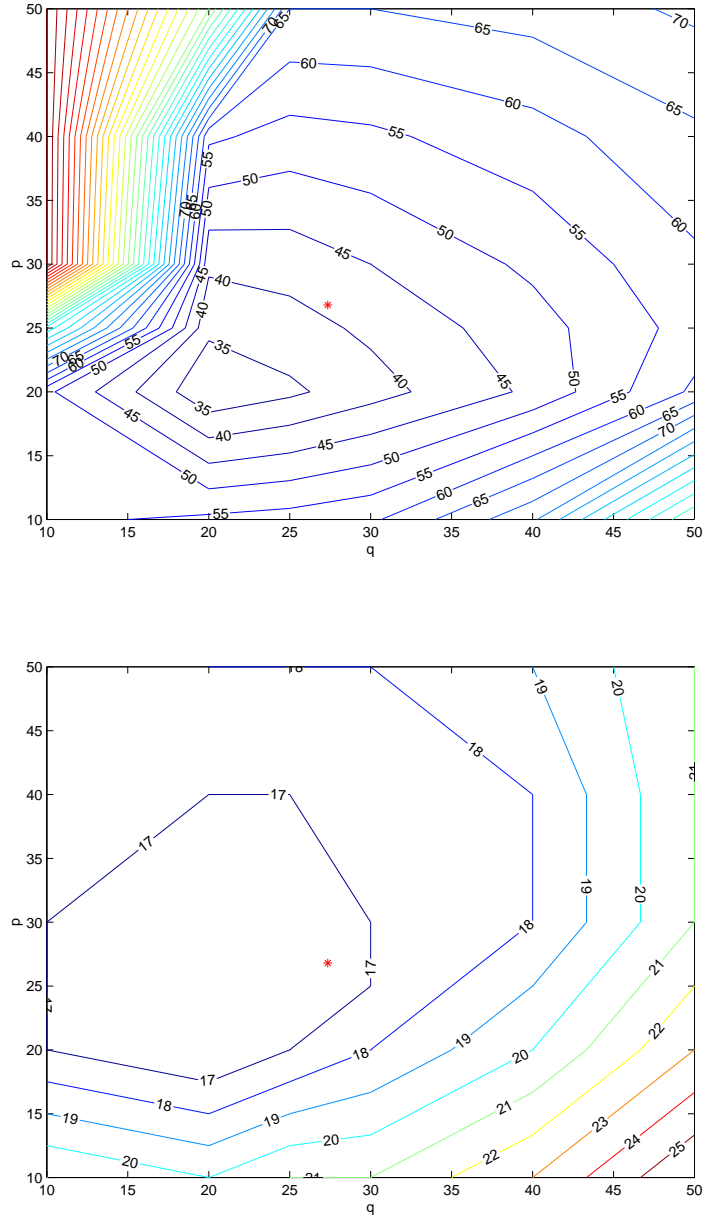


Figure 7: Number of iterations needed to achieve a certain precision as function of the optimization parameters p and q in the zeroth order transmission conditions, for the iterative algorithm (above) and for the Krylov accelerated algorithm (below). The star denotes the optimized parameters p^* and q^* found by our Fourier analysis

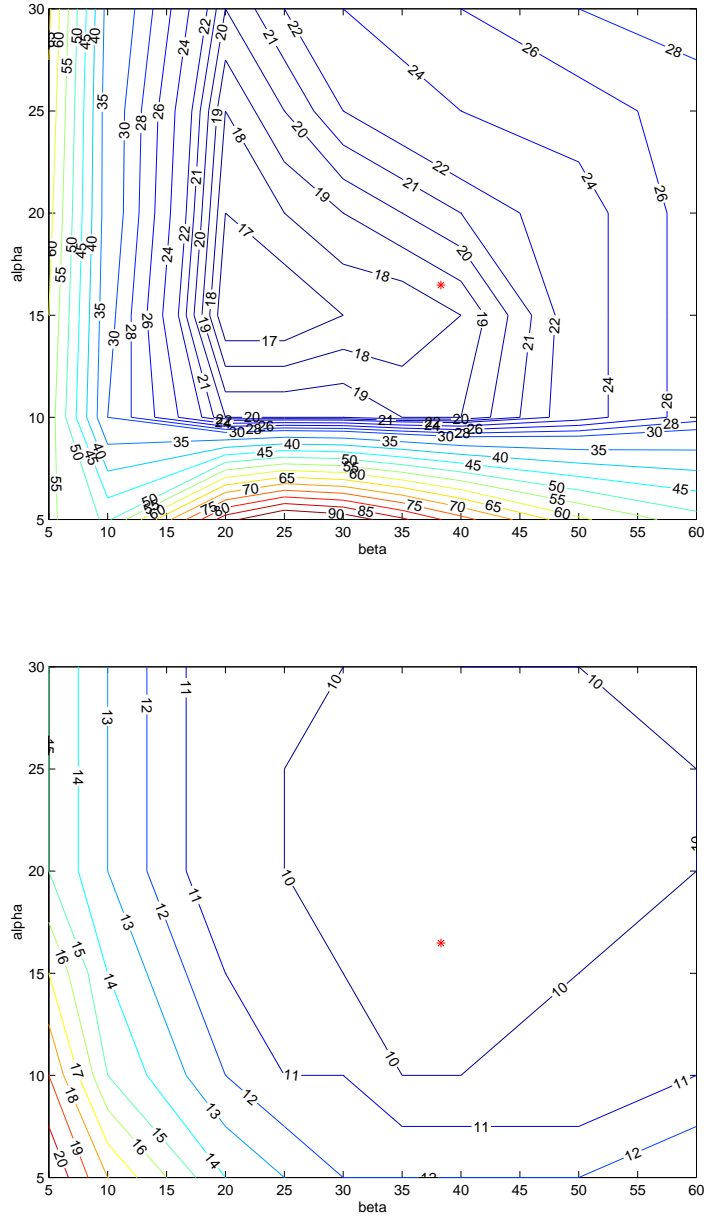


Figure 8: Number of iterations needed to achieve a certain precision as function of the optimization parameters α and β in the second order transmission conditions, for the iterative algorithm (above) and for the Krylov accelerated algorithm (below). The star denotes the optimized parameters α^* and β^* found by our Fourier analysis

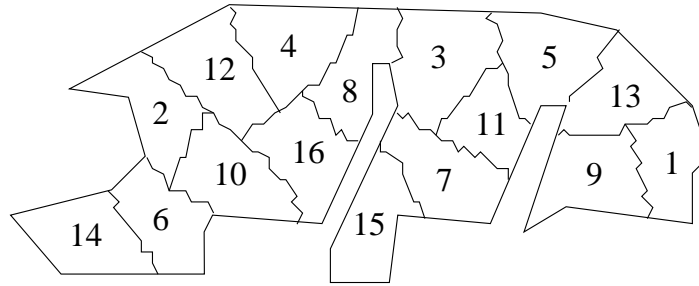


Figure 9: Decomposition of the passenger compartment into 16 subdomains

4 Conclusion

In this presentation, we have stressed the influence of interface conditions in domain decomposition methods. By choosing them carefully, it is possible to achieve better and more reliable convergence behaviors for various types of scalar equations and the Stokes or Maxwell systems.

For complex systems of equations (e.g. multiphase flow, compressible Navier-Stokes equations) treated in a coupled way, the theory is still in development.

On the end-user point of view (physicists, engineers,...) there is a lack of black-box routines working at the matrix level and implementing the latest methods. This could be achieved by a closer collaboration between numerical analysis and computer science.

Acknowledgment I thank Victorita Dolean and Laurent Saas for their careful reading and useful suggestions.

References

- [AK95] Y. Achdou and Y. A. Kuznetsov, Substructuring preconditioners for finite element methods on nonmatching grids, *East-West J. Numer. Math.* **3** (1) (1995), 1–28.
- [AKP95] Y. Achdou, Y. Kuznetsov, and O. Pironneau, Substructuring preconditioners for the Q_1 mortar element method, *Numer. Math.* **71** (1995), 419–449.
- [AMW99] Y. Achdou, Y. Maday, and O. B. Widlund, Iterative substructuring preconditioners for mortar element methods in two dimensions, *SIAM J. Numer. Anal.* **36** (2) (1999), 551–580.
- [ATNV00] Y. Achdou, P. L. Tallec, F. Nataf, and M. Vidrascu, A domain decomposition preconditioner for an advection-diffusion problem, *Comput. Methods Appl. Mech. Engrg.* **184** (2000), 145–170.
- [BD98] D. Braess and W. Dahmen, Stability estimates of the mortar finite element method for 3-dimensional problems, *East-West J. Numer. Math.* **6** (4) (1998), 249–264.

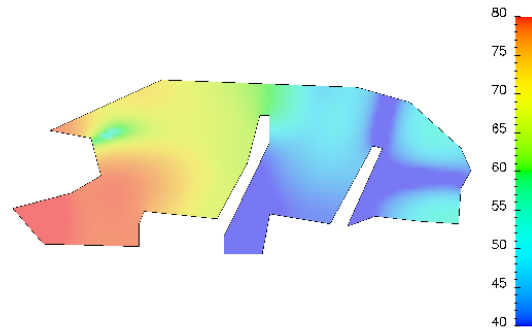


Figure 10: Acoustic field in the passenger compartment of the VOLVO S90

- [BGLTV] J.-F. Bourgat, R. Glowinski, P. Le Tallec, and M. Vidrascu, Variational formulation and algorithm for trace operator in domain decomposition calculations, in: *Domain Decomposition Methods* (T. Chan et al., eds.), SIAM, Philadelphia, PA, 1989, 3–16.
- [BMP93] C. Bernardi, Y. Maday, and A. T. Patera, Domain decomposition by the mortar element method, in: *Asymptotic and Numerical Methods for Partial Differential Equations with Critical Parameters* (H. G. Kaper and M. Garbey, eds.), NATO ASI Ser. C **384**, Kluwer, Dordrecht, 1993, 269–286.
- [CCEW98] X.-C. Cai, M. A. Casarin, F. W. Elliott Jr., and O. B. Widlund, Overlapping Schwarz algorithms for solving Helmholtz’s equation, in: *Domain Decomposition Methods, 10 (Boulder, CO, 1997)*, Amer. Math. Soc., Providence, RI, 1998, 391–399.
- [CDS99] X.-C. Cai, M. Dryja, and M. V. Sarkis, Overlapping nonmatching grid mortar element methods for elliptic problems, *SIAM J. Numer. Anal.* **36** (1999), 581–606.
- [CGJ00] F. Collino, S. Ghanemi, and P. Joly, Domain decomposition methods for harmonic wave propagation : a general presentation, *Comput. Methods Appl. Mech. Engrg.* (2–4) (2000), 171–211.

- [CGPW89] T. Chan, R. Glowinski, J. Périaux, and O. Widlund, eds., *Domain Decomposition Methods*, SIAM, Philadelphia, PA, 1989.
- [Che98] P. Chevalier, *Méthodes numériques pour les tubes hyperfréquences. Résolution par décomposition de domaine*, Ph.D. thesis, Université Paris VI, 1998.
- [CM94] T. F. Chan and T. P. Mathew, Domain decomposition algorithms, in: *Acta Numerica 1994*, Cambridge University Press, 1994, 61–143.
- [CMW95] L. C. Cowsar, J. Mandel, and M. F. Wheeler, Balancing domain decomposition for mixed finite elements, *Math. Comp.* **64** (211) (1995), 989–1015.
- [CW92] X.-C. Cai and O. B. Widlund, Domain decomposition algorithms for indefinite elliptic problems, *SIAM J. Sci. Statist. Comput.* **13** (1) (1992), 243–258.
- [Des93] B. Després, Domain decomposition method and the Helmholtz problem. II, in: *Second International Conference on Mathematical and Numerical Aspects of Wave Propagation (Newark, DE, 1993)*, SIAM, Philadelphia, PA, 1993, 197–206.
- [DJR92] 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)*, North-Holland, Amsterdam, 1992, 475–484.
- [DSW96] M. Dryja, M. V. Sarkis, and O. B. Widlund, Multilevel Schwarz methods for elliptic problems with discontinuous coefficients in three dimensions, *Numer. Math.* **72** (3) (1996), 313–348.
- [EM77] B. Engquist and A. Majda, Absorbing boundary conditions for the numerical simulation of waves, *Math. Comp.* **31** (139) (1977), 629–651.
- [EZ98] B. Engquist and H.-K. Zhao, Absorbing boundary conditions for domain decomposition, *Appl. Numer. Math.* **27** (4) (1998), 341–365.
- [FR91] C. Farhat and F.-X. Roux, A method of finite element tearing and interconnecting and its parallel solution algorithm, *Internat. J. Numer. Methods Engrg.* **32** (1991), 1205–1227.
- [Giv92] D. Givoli, *Numerical Methods for Problems in Infinite Domains*, Elsevier (1992).
- [GMN01] M. J. Gander, F. Magoulès, and F. Nataf, Optimized Schwarz methods without overlap for the Helmholtz equation, *SIAM J. Numer. Anal.* (2001), to appear.
- [Hig86] R. Higdon, Absorbing boundary conditions for difference approximations to the multi-dimensional wave equations, *Math. Comp.* **47** (176) (1986), 437–459.
- [HTJ88] T. Hagstrom, R. P. Tewarson, and A. Jazcilevich, Numerical experiments on a domain decomposition algorithm for nonlinear elliptic boundary value problems, *Appl. Math. Lett.* **1** (3), 1988.
- [Jap98] C. Japhet, Optimized Krylov-Ventcell method, Application to convection-diffusion problems, in: *Proc. 9th International Conference Domain Decomposition Methods* (P. E. Bjørstad et al., eds.), 1998, 382–389.

- [JN00] C. Japhet and F. Nataf, The best interface conditions for domain decomposition methods: Absorbing boundary conditions, to appear in: *Artificial Boundary Conditions, with Applications to Computational Fluid Dynamics Problems* (L. Tourrette, ed.), Nova Science, 2000.
- [JNR01] C. Japhet, F. Nataf, and F. Rogier, The optimized order 2 method. Application to convection-diffusion problems, *Future Generation Computer Systems FUTURE* **18**, 2001.
- [KW01] A. Klawonn and O. B. Widlund, FETI and Neumann-Neumann iterative substructuring methods: Connections and new results, *Comm. Pure Appl. Math.* **54** (2001), 57–90.
- [LBCW98] *Proc. Eleventh International Conference Domain Decomposition Methods, 1998* (C. H. Lai et al., eds.).
- [Lio90] P.-L. Lions, On the Schwarz alternating method. III: A variant for nonoverlapping subdomains, in: *Third International Symposium on Domain Decomposition Methods for Partial Differential Equations* (T. F. Chan et al., eds.), SIAM, Philadelphia, PA, 1990.
- [LT94] P. Le Tallec, Domain decomposition methods in computational mechanics, in: *Computational Mechanics Advances* vol. 1 (2) (J. Tinsley Oden, ed.), North-Holland, Amsterdam, 1994, 121–220.
- [Nat96] F. Nataf, Absorbing boundary conditions in block Gauss-Seidel methods for convection problems, *Math. Models Methods Appl. Sci.* **6** (4) (1996), 481–502.
- [Nep91] S. V. Nepomnyaschikh, Application of domain decomposition to elliptic problems on with discontinuous coefficients, in: *Fourth International Symposium on Domain Decomposition Methods for Partial Differential Equations* (R. Glowinski et al., eds.), SIAM, Philadelphia, PA, 1991, 242–251.
- [Nie99] F. Nier, Remarques sur les algorithmes de décomposition de domaines, in: *Séminaire : Équations aux Dérivées Partielles, 1998–1999*, Exp. No. IX, 26, École Polytechnique, 1999.
- [NR95] F. Nataf and F. Rogier, Factorization of the convection-diffusion operator and the Schwarz algorithm, *Math. Models Methods Appl. Sci.* **5** (1) (1995), 67–93.
- [NRdS94] F. Nataf, F. Rogier, and E. de Sturler, Optimal interface conditions for domain decomposition methods, Technical report, CMAP (École Polytechnique), 1994.
- [PEBK97] *Proc. Ninth International Conference on Domain Decomposition Methods, 1997*, (M. S. Espedal et al., eds.).
- [QV99] A. Quarteroni and A. Valli, *Domain Decomposition Methods for Partial Differential Equations*, Oxford Sci. Publ., Oxford University Press, 1999.

- [SBG96] B. F. Smith, P. E. Bjørstad, and W. Gropp, *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*, Cambridge University Press, 1996.
- [WFNS98] F. Willien, I. Faille, F. Nataf, and F. Schneider, Domain decomposition methods for fluid flow in porous medium, in: *6th European Conference on the Mathematics of Oil Recovery*, 1998.
- [Woh99] B. Wohlmuth, Hierarchical a posteriori error estimators for mortar finite element methods with Lagrange multipliers, *SIAM J. Numer. Anal.* **36** (1999), 1636–1658.