Domain decomposition and high order methods for the solution of partial differential systems of equations. Application to fluid dynamics and electromagnetism

Victorita Dolean

UNIVERSITÉ DE NICE-SOPHIA ANTIPOLIS
FACULTÉ DES SCIENCES

Mémoire de synthèse

présenté par

Victoriţa Dolean

pour obtenir

L'HABILITATION A DIRIGER LES RECHERCHES
(discipline : Mathématiques Appliquées)

Algorithmes par decomposition de domaine et méthodes de discrétisation d'ordre elevé pour la résolution des systèmes d'équations aux dérivées partielles

Application aux problèmes issus de la mécanique des fluides et de l'électromagnétisme

Soutenance prevue le 7 Juillet 2009 devant le jury composé de :

Abderrahmane Bendali (rapporteur)	:	Professeur, Institut de Mathématiques, Toulouse.
Jan Hesthaven (rapporteur)	:	Professeur, Université de Brown.
Axel Klawonn (rapporteur)	:	Professeur, Université Duisburg-Essen.
Stéphane Descombes	:	Professeur, Université de Nice Sophia-Antipolis.
Martin J. Gander	:	Professeur, Université de Genève.
Stéphane Lanteri	:	Directeur de Recherche INRIA, Sophia-Antipolis.
Frédéric Nataf	:	Directeur de Recherche CNRS, Université Paris VI.
Christian Pichot	:	Directeur de Recherche CNRS, LEAT, Sophia-Antipolis.

Contents

Introduction **4**

1 Schwarz algorithms for fluid dynamics **9**
1.1 Convergence analysis of a Schwarz algorithm for the Euler equations 9
1.2 Smith normal forms and domain decomposition 22
1.2.1 A new domain decomposition method for the compressible Euler equations 24
1.2.2 Deriving a new domain decomposition method for the Stokes equations using the Smith factorization 32

2 DG methods for Maxwell **43**
2.1 Solution of the time-harmonic Maxwell's equations using Discontinuous Galerkin methods 43
2.2 An implicit DGTD method for two-dimensional electromagnetic wave propagation 51

3 Domain decomposition for Maxwell **59**
3.1 Optimized Schwarz methods for Maxwell's equations 59
3.2 A domain decomposition method for solving the three-dimensional time-harmonic Maxwell's equations discretized by discontinuous Galerkin methods 71
3.2.1 Conclusion 78

Publication list of the author **78**

Bibliography **87**

Introduction

My main research topic is about developing new domain decomposition algorithms for the solution of systems of partial differential equations. This was mainly applied to fluid dynamics problems (as compressible Euler or Stokes equations) and electromagnetics (time-harmonic and time-domain first order system of Maxwell's equations). Since the solution of large linear systems is strongly related to the application of a discretization method, I was also interested in developing and analyzing the application of high order methods (such as Discontinuos Galerkin methods) to Maxwell's equations (sometimes in conjuction with time-discretization schemes in the case of time-domain problems).

As an active member of NACHOS project (besides my main affiliation as an assistant professor at University of Nice), I had the opportunity to develop certain directions in my research, by interacting with permanent et non-permanent members (Post-doctoral researchers) or participating to supervision of PhD Students. This is strongly reflected in a part of my scientific contributions so far. This memoir is composed of three parts: the first is about the application of Schwarz methods to fluid dynamics problems; the second about the high order methods for the Maxwell's equations and the last about the domain decomposition algorithms for wave propagation problems.

Schwarz algorithms/preconditioning methods for fluid dynamics problems.

This work is the concretization of a long time collaboration with Frédéric Nataf (DR CNRS, University Paris VI) and Stéphane Lanteri (DR INRIA, Sophia Antipolis). We need to underline that there were few contributions on the application of Schwarz methods to complex systems of PDEs, such as compressible Euler system. We have studied the construction of non-overlapping domain decomposition methods for the Euler equations. These methods lead to very fast resolution using parallel computers.

Different points of view have been approached in this study. The first one was the analysis of classical Schwarz type algorithms [3, 6]; The second was the construction of even faster algorithms using more sophisticated interface conditions [4, 5]. Nevertheless, in order to have a sufficiently simple and general method, another approch has been studied in [9] using a more general flux decomposition at the interface between subdomains.

Another aspect of these methods is concerned with the developpement of preconditoning methods using symbolic algebra tools such as Smith factorization (Smith normal forms). This completely original approach in the scientific computing community, allows a more intrinsic study of the systems of PDEs and the derivation of new methods which are more robust that the simple extensions of those applied to the scalar ones. This work was accomplished in collaboration with Frédéric Nataf (DR CNRS, Université Paris VI) and Gerd Rapin (University of Göttingen). Using ideas from the preconditioning methods for the scalar equations and more precisely of Robin-Robin preconditionner, one can easily extend these methods to systems of equations using the Smith factorization. The Euler system has been studied in [11] and a more general principle has been shown in [8] and presented in detail for the Stokes equations in [19].

The last aspect that has been considered in collaboration with Stéphane Lanteri (DR INRIA, Sophia Antipolis) was the study of different aspects of parallelism when solving a problem by a domain decomposition method. Questions of scalability, speed-up, choice of local solvers have been studied in [7] for the resolution of compressible flows.

DG-Pp (Discontinuous Galerkin of order p) methods for time-harmonic and time-domain Maxwell's equations

DG-Pp methods for time-harmonic problems
In the context of the PhD thesis of Hugo Fol, which I co-supervised (from October 2003 to September 2006), we have developed a finite volume method DGFD-P0 (DGFD -Pk = Discontinuous Galerkin Frequency Domain method with local approximations of order k) formulation and a discontinuous Galerkin formulation based on a linear interpolation method (DGFD-P1 formulation) for the numerical resolution of the time-harmonic Maxwell equations on unstructured tetrahedral meshes. Initially, the DGFD-P0 and DGFD-P1 formulations that we have developed rely on the use of a centered scheme for the evaluation of fluxes between neighboring elements are applied to the discretization of the first-order form (or mixed form) of the Maxwell equations. In ulterior works, in collaboration with Ronan Perrussel (former Post-doctoral researcher of NACHOS team, now CNRS researcher at Ampère Laboratory from the École Centrale de Lyon) we performed an extensive numerical study of the use of different fluxes for our DGFD-Pp formulations for the first-order time-harmonic Maxwell equations (see [15]) which proved that the centered scheme is not necessarily the best choice for the time-harmonic problem.

Hybrid explicit/implicit time integration strategies
Nowadays, a variety of methods exist for the numerical treatment of the time-domain Maxwell equations, ranging from the well established and still prominent finite difference time-domain (FDTD) methods based on Yee's scheme to the more recent finite element time domain (FETD) and discontinuous Galerkin time domain (DGTD) methods. Explicit time integration schemes such as the leap-frog scheme adopted in the DGTD methods that have

been developed so far are subjected to stability conditions that become very restrictive when the underlying mesh is locally refined since the global time step is deduced from the volume of the smallest mesh element. Two main strategies can be considered to improve this situation: local time stepping and implicit time integration. We have recently completed a preliminary work on the design of implicit time integration schemes for solving the time-domain Maxwell equations. In this work, an implicit version of the centered finite volume method previously proposed by Remaki is developed by resorting to a Crank-Nicolson time integration scheme in place of the leap-frog scheme. The resulting implicit discontinuous Galerkin time-domain (IDGTD) method is non-dissipative and unconditionally stable however, it is also more dispersive than the original, explicit, version. In the framework of the PhD thesis of Adrien Catella (October 2005-September 2008), we have extended this work to IDGTD-Pp formulations (see [24] for details). While doing so, we have studied high order time integration schemes since a numerical dispersion analysis of the IDGTD-P0 and IDGTD-P1 methods in 1D shows that the second-order Crank-Nicolson scheme is the source of a temporal dispersion error term.

Domain decomposition methods for wave propagation problems

This research theme consists in the formulation, analysis and concrete evaluation of Schwarz type domain decomposition methods in conjunction with discontinuous Galerkin approximation methods on unstructured meshes for the calculation of time-domain and time-harmonic wave propagation problems in heterogeneous media. In these algorithms, a first order absorbing condition is imposed at the interfaces between neighboring subdomains. This interface condition is equivalent to a Dirichlet condition for characteristic variables associated to incoming waves. For this reason, it is often referred as a natural interface condition. From the discretization viewpoint, this interface condition gives rise to a boundary integral term which is treated using a flux splitting scheme similar to the one applied at a physical absorbing boundary where a Silver-Müller condition is applied. Whatsoever is the overlapping strategy, the Schwarz algorithm can be used as a global solver or it can be reformulated as a Richardson iterative method acting on an interface system. In the latter case, the iterative resolution of the interface system can be performed in a more efficient way using a Krylov method. This approach has been implemented in the context of low order discontinuous Galerkin methods (finite volume method and discontinuous Galerkin method based on linear interpolation) in [16] (in collaboration with Ronan Perrussel and Stéphane Lanteri) after a preliminary study of the use of the different fluxes in [15].

Beside Schwarz algorithms based on natural interface conditions, we also studied algorithms that make use of more effective transmission conditions. From the theoretical point of view, this represents a much more challenging goal since most of the existing results on optimized Schwarz algorithms have been obtained for scalar PDEs. We extended the techniques for obtaining optimized Schwarz methods previously developed for the scalar PDEs to systems of PDEs by using appropriate relationships between systems and equivalent scalar

problems. For the time-harmonic Maxwell equations, we can derive Schwarz algorithms in a similar way. As for the time-domain Maxwell equations, similar ideas of equivalence between scalar problems and systems can be applied, the main differences lying in the fact that the classical corresponding algorithms may not converge. The use of more sophisticated, optimized boundary conditions is then mandatory. Interface conditions of this kind (optimized Schwarz methods) for the time-harmonic Maxwell equations were extensively studied in [20] (in collaboration with Martin J. Gander from University of Geneva and Luca Gerardo-Giorda from University of Trento) and tested in conjuction with a Discontinuous Galerkin method in [18] (in collaboration with Ronan Perrussel and Stéphane Lanteri).

This memoir is organized as follows:

- In Chapter 1, the main results concerning the application of Schwarz algorithms or preconditionning methods to fluid dynamics problems, are presented. For details or proofs, refer to the following papers: [6], [11] and [19]. Other works in relation with this topic (but not presented here) are: [7] , [8], [9] and [10].

- In Chapter 2, the main results concerning the application of Discontinuous Galerkin methods to Maxwell's equations are presented. For details refer to the following papers: [15] , [21]. Other works in relation to these topics (but not presented here) are [24] and [18].

- In Chapter 3, the main results concerning the application of domain decomposition methods to Maxwell's equations are presented. For details refer to the following papers: [20] and [16] . Another work in relation to these topics (but not presented here) is [18].

The works that cannot be included in one of the above categories: [13] (behavior of Schwarz algorithms applied to Cauchy-Riemann equations) and [14] (p-Multigrid applied to triangular spectral element methods).

Chapter 1

Schwarz algorithms/preconditioning methods for fluid dynamics problems

As it is already known, the numerical simulation relies heavily on solving linear systems of equations. For the large scale problems we deal with in today's standard applications, it is necessary to rely on iterative Krylov methods, interesting because of their limited memory requirements and scalability properties. They are preconditioned by domain decomposition methods, incomplete factorizations, or multigrid preconditioners. These methods are well understood and efficient for scalar symmetric equations (Laplacian, biLaplacian, ...), and to some extent for non symmetric equations (convection-diffusion, ...). But they exhibit poor performance and lack robustness when they are used for systems of PDEs, especially for the non symmetric case (fluid mechanics, porous media, ...). In the following, several attempts to solve non-trivial systems, such as Euler, Stokes or Oseen equations, by domain decomposition methods are presented. To start with, a very simple idea to define a Schwarz algorithm is presented and analyzed in section 1.1. Afterwards, an algebraic idea to design new preconditioners is shown in section 1.2 with illustration of application on Euler and Stokes equations.

1.1 Convergence analysis of a Schwarz algorithm for the Euler equations

This work is concerned with the convergence analysis of the method adopted in [1] . When dealing with supersonic flows, whatever the space dimension is, imposing the appropriate characteristic variables as interface conditions leads to a convergence of the algorithm which is optimal with regards to the number of subdomains. This property is generally lost for subsonic flows except for the case of one-dimensional problems, when the optimality is again expressed as the number of iterations being equal to the number of subdomains (see Bjørhus [Bjø95b] and Quarteroni [Qua90] for more details). For higher space dimensions, one cannot analyze the convergence of the algorithm in the same way. Therefore, a new kind of approach is required in the latter case. In a similar context, Clerc [Cle98] gives a convergence proof for

the additive Schwarz algorithm applied to the solution of a general linear hyperbolic system of PDEs, in the two- and three-dimensional cases, based on an energy estimate. We note that this proof is limited to a non-overlapping decomposition of the computational domain and no quantitative results on the convergence rate are provided. Here, we study the convergence of the proposed algorithm from a quantitative point of view in the two- and three-dimensional cases, and for overlapping and non-overlapping decompositions, by applying a Fourier analysis. For the sake of simplicity, we limit the analysis to decompositions into two- and three subdomains and we provide analytical expressions of the convergence rate of the Schwarz algorithm applied to the linearized equations. Surprisingly, there exist flow conditions for which the asymptotic convergence rate is equal to zero. Moreover, this result is independent of the spatial dimension. The same convergence analysis is performed on the discretized equations by means of the finite volume formulation considered in [1] but adapted here to a quadrangular mesh. We obtain the discrete counterpart of the convergence rate which is now a function of the mesh size. We observe that its expression is similar to the one characterizing an overlapping Schwarz algorithm in the continuous case with a one cell overlap. In other words, the cell-based partitioning and the vertex centered finite volume formulation adopted in [1] naturally corresponds to a discrete formulation of an overlapping continuous Schwarz algorithm (although we use a non-overlapping element-based partitioning).

We introduce here the additive Schwarz algorithm which is at the heart of our study and we mention an existing result concerning the convergence of the algorithm. To begin with, we consider a general system of hyperbolic conservation laws of the form

$$\frac{\partial W}{\partial t} + \sum_{i=1}^{d} \frac{\partial F_i(W)}{\partial x_i} = 0 \quad \text{where} \qquad W \in \mathbb{R}^p, \tag{1.1}$$

where d denotes the spatial dimension and p the dimension of the system. The flux vectors (or flux functions) $F_i(W)$ are assumed differentiable with respect to the state vector $W = W(x,t)$. In the general case, these flux vectors are non-linear functions of W. Under the hypothesis that the solution is regular, we can also write a non-conservative (or quasi-linear) equivalent form of equation (1.1),

$$\frac{\partial W}{\partial t} + \sum_{i=1}^{d} A_i(W) \frac{\partial W}{\partial x_i} = 0, \tag{1.2}$$

where the A_i are the Jacobian matrices of the flux vectors. We first integrate (1.1) in time using a backward Euler implicit scheme involving a linearization of the flux functions. This operation results in the linearized system

$$\mathcal{L}(\delta W) \equiv \frac{1}{\Delta t}\delta W + \sum_{i=1}^{d} A_i \frac{\partial \delta W}{\partial x_i} = f, \tag{1.3}$$

where $\delta W \equiv W^{n+1} - W^n$, $W^{n+1} = W(x,(n+1)\Delta t)$, and A_i is a shorthand for $A_i(W^n)$.

We recall below a well known result (see for instance Clerc [Cle98]) for the boundary value problem (BVP) associated with system (1.3).

Theorem 1 *Let $f \in L^2(\Omega)^p$ and $g \in L^2_A(\partial\Omega)$ with*

$$L^2_A(\partial\Omega) = \{U \text{ such that } \int_{\partial\Omega} |A_{\boldsymbol{n}}| U\cdot U d\sigma < \infty\},\ L^2_{A'}(\partial\Omega) = \{U \text{ such that } \int_{\partial\Omega} |A_{\boldsymbol{n}}|^{-1} U\cdot U d\sigma < \infty\}$$

Then, the following BVP problem is well posed

$$\left\{ \begin{array}{rcll} \mathcal{L}(U) = \dfrac{\mathsf{Id}}{\Delta t} U + \displaystyle\sum_{i=1}^{d} A_i \frac{\partial U}{\partial x_i} & = & f & \text{in } \Omega, \\ A^-_{\boldsymbol{n}} U & = & A^-_{\boldsymbol{n}} g & \text{on } \partial\Omega, \end{array} \right. \tag{1.4}$$

where $A_{\boldsymbol{n}} = \sum_{i=1}^d A_i n_i$ is diagonalizable (since the original system (1.1) is hyperbolic) and whose diagonalization writes as $A_{\boldsymbol{n}} = T\Lambda T^{-1} = T\mathsf{diag}(\lambda_i)T^{-1}$. In these expressions, $\boldsymbol{n} = (n_1, \ldots, n_d)$ denotes the outward normal vector at any point of $\partial\Omega$ while the negative and absolute parts of $A_{\boldsymbol{n}}$ are respectively defined by $A^-_{\boldsymbol{n}} = T\Lambda^- T^{-1} = T\mathsf{diag}(\min\{\lambda_i, 0\})T^{-1}$ and $|A_{\boldsymbol{n}}| = T|\Lambda|T^{-1} = T\mathsf{diag}(|\lambda_i|)T^{-1}$. The unique solution U of (1.4) lies in $\tilde{H}$ defined by

$$\tilde{H} = \{U \in L^2(\Omega)^p \text{ such that } \sum_{i=1}^{d} A_i \partial_{x_i} U \in L^2(\Omega)^p \text{ and } U|_{\partial\Omega} \in L^2_A(\partial\Omega)\}.$$

Furthermore, if $f = 0$ we have the estimate

$$C_0 \|U\|^2_{L^2(\Omega)} + \|A^+_{\boldsymbol{n}} U\|^2_{L^2_{A'}(\partial\Omega)} \leq \|A^-_{\boldsymbol{n}} g\|^2_{L^2_{A'}(\partial\Omega)}, \tag{1.5}$$

where $A^+_{\boldsymbol{n}} = T\Lambda^+ T^{-1} = T\mathsf{diag}(\max\{\lambda_i, 0\})T^{-1}$.

In the following, we are interested in solving the BVP problem (1.4) in a domain decomposition framework by using an additive Schwarz algorithm based on transmission conditions at subdomain interfaces that consist of Dirichlet conditions for the characteristic variables corresponding to incoming waves (a formulation already considered by Bjørhus [Bjø95b] and by Quarteroni and Stolcis [QS96]). In other words, the treatment of the boundary condition on the physical boundary in eq. (1.4) is extended to the artificial boundaries defined by interfaces between neighboring subdomains. We note that this algorithm has been previously studied from the numerical point of view in Dolean and Lanteri [1] in the context of the calculation of steady compressible inviscid flows governed by the two-dimensional Euler equations.

We consider a decomposition of the domain Ω into N overlapping or non-overlapping subdomains $\overline{\Omega} = \bigcup_{i=1}^N \overline{\Omega}_i$. We denote by $\boldsymbol{n}_{ij}$ the outward normal to the interface between Ω_i and a neighboring subdomain Ω_j. Let $W_i^{(0)}$ denote the initial approximation of the solution in subdomain Ω_i. A general additive Schwarz algorithm for computing $(W_i^{(k+1)})_{1\leq i\leq N}$ from $(W_i^{(k)})_{1\leq i\leq N}$ (where k defines the iteration of the Schwarz algorithm) is

$$\left\{ \begin{array}{lcll} \mathcal{L}W_i^{(k+1)} & = & f & \text{in } \Omega_i, \\ C_{ij}W_i^{(k+1)} & = & C_{ij}W_j^{(k)} & \text{on } \Gamma_{ij} = \partial\Omega_i \cap \overline{\Omega}_j, \\ A_{\boldsymbol{n}}^- W_i^{(k+1)} & = & A_{\boldsymbol{n}}^- g & \text{on } \partial\Omega \cap \partial\Omega_i, \end{array} \right. \tag{1.6}$$

where the C_{ij} are interface transmission operators and g is given. Note that in equation (1.6) we have used that $\mathcal{L}$ and C_{ij} are linear operators. Similar algorithms have been extensively studied by Nataf [Nat96], Nataf *et al.* [NRdS95], Japhet *et al.* [JNR01] and Japhet and Nataf [JN00] for convection-diffusion problems and by Engquist and Zhao [EHK98] and Gander et al. [GMN02] for some elliptic problems. In particular, these authors have considered the use of high-order optimized interface conditions, inspired by the concept of absorbing boundary conditions for unbounded domains [EM77], for improving the convergence of the Schwarz algorithm.

Remark 1 *The so-called classical interface conditions are characterized by interface operators of the form $C_{ij} = A_{\boldsymbol{n}_{ij}}^-$. The corresponding formulation of the Schwarz algorithm (1.6) is the one adopted in [1] . It is also possible to consider a space-time decomposition for the time dependent equation as in [GHN99]. The interface conditions can be defined as well in terms of flux splitting. In [Bjø95b], the convergence of the Schwarz algorithm is analyzed for the semi-discrete (in space but not in time) equations.*

For general linear hyperbolic systems of PDEs, whatever the space dimension is, Clerc [Cle98] proved the convergence of the Schwarz algorithm in the case of non-overlapping decompositions.

Theorem 2 *Let $E_i^{(k)} = W_i^{(k)} - W_{|\partial\Omega_i}$ be the error vector associated with the restriction of the global solution of the problem to subdomain Ω_i . Then, the Schwarz algorithm with classical transmission conditions converges*

$$\lim_{k\to\infty} \|E_i^{(k)}\|_{L^2(\Omega_i)^p} = 0 \quad \textit{and} \quad \lim_{k\to\infty} \|\sum_{j=1}^{d} A_j \partial_j E_i^{(k)}\|_{L^2(\Omega_i)^p} = 0.$$

We can now study the convergence of the additive Schwarz algorithm (1.6) based on the classical interface conditions $C_{ij} = A_{\boldsymbol{n}_{ij}}^-$ when applied to the solution of the 2D and 3D Euler equations that model inviscid compressible flows. Contrary to Clerc[Cle98], we consider both overlapping and non-overlapping decompositions. First, we recall the conservative for of the Euler equations in the two-dimensional case:

$$\frac{\partial W}{\partial t} + \overrightarrow{\nabla}.\mathbb{F}(W) = 0 \quad , \quad W = (\rho,\ \rho\boldsymbol{V},\ E)^T \quad , \quad \overrightarrow{\nabla} = \left(\frac{\partial}{\partial x},\ \frac{\partial}{\partial y}\right)^T. \tag{1.7}$$

In equation (1.19), $W = W(\mathbf{x}, \mathbf{t})$ is the vector of conservative variables, $\mathbf{x}$ and t respectively denote the space and time variables and $\mathbb{F}(W) = (F_1(W)\ ,\ F_2(W))^T$ is the

conservative flux vector whose components are given by

$$F_1(W) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(E+p) \end{pmatrix}, \quad F_2(W) = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(E+p) \end{pmatrix}.$$

In the above expressions, ρ is the density, $\boldsymbol{V} = (u,\ v)^T$ is the velocity vector, E is the total energy per unit of volume and p is the pressure. The pressure is deduced from the other variables using the state equation for a perfect gas $p = (\gamma_s - 1)(E - \frac{1}{2}\rho \parallel \boldsymbol{V} \parallel^2)$, where γ_s is the ratio of the specific heats ($\gamma_s = 1.4$ for the air).

Then, we consider a two-subdomain decomposition of the real space $\mathbb{R}^d$ ($d = 2$ or 3) into two overlapping subdomains such that $\Omega_1 =]-\infty, \gamma[\times\mathbb{R}^{d-1}$ and $\Omega_2 =]\beta, +\infty[\times\mathbb{R}^{d-1}$ where $\beta \leq \gamma$. We study the convergence of the Schwarz algorithm in two- and three-dimensions assuming subsonic flow conditions. We recall that in the multi-dimensional supersonic case, the Schwarz algorithm converges in two steps for a two-subdomain decomposition. The starting point of our analysis is given by the linearized form (1.3) of the Euler equations

$$\mathcal{L}W \equiv \frac{\mathrm{Id}}{\Delta t}W + A_1\frac{\partial W}{\partial x_1} + \sum_{i=2}^{d} A_i\frac{\partial W}{\partial x_i} = f, \tag{1.8}$$

with $A_i \equiv A_i(\overline{W})$ where $\overline{W}$ denotes the constant vector state used for the linearization of the Euler equations. First, we apply the change of variable $U = T^{-1}W$ to the above system which is based on the eigenvector factorization of $A_1 = T\Lambda T^{-1}$. hen, (1.8) becomes

$$\tilde{\mathcal{L}}U \equiv bU + \Lambda\frac{\partial U}{\partial x_1} + \sum_{i=2}^{d} B_i\frac{\partial U}{\partial x_i} = T^{-1}f \quad \text{with} \quad b = \frac{1}{\Delta t}, \tag{1.9}$$

where $B_i = T^{-1}A_iT$ and $\Lambda = \mathrm{diag}(\lambda_i)$ is the diagonal matrix from the diagonalization of A_1. To estimate the convergence rate of the Schwarz algorithm (1.6), we use a Fourier transform (denoted by $\mathcal{F}$) of all the spatial directions except the first one. The vector of Fourier variables is denoted by $\boldsymbol{\xi} = (\xi_j,\ j = 1, \ldots, d-1)$. Let $(E_i^{(k)})(x) = (U_i^{(k)} - U_{|\Omega_i})(x)$ be the error vector in subdomain Ω_i at the k-th iteration of the Schwarz algorithm. We denote by

$$\hat{E}(x_1, \xi_1, \ldots, \xi_{d-1}) = \mathcal{F}E(x_1, \ldots, x_d) = \int_{\mathbb{R}} e^{-i\xi_1 x_2 - \ldots - i\xi_{d-1}x_d} E(x_1, \ldots, x_d)dx_2 \ldots dx_d,$$

the Fourier symbol of the error vector. This transformation can be done only if the A_i matrices are constant which is the case here because we have considered the linearized form of the Euler equations around a constant state $\overline{W}$. The Schwarz algorithm in Fourier space becomes

$$\Omega_1 : \left\{ \begin{array}{rcl} \dfrac{d\hat{E}_1^{(k+1)}}{dx_1} & = & -\mathcal{M}(\boldsymbol{\xi})\hat{E}_1^{(k+1)},\ x_1 < \gamma, \\ (\hat{E}_1^{(k+1)})_j & = & (\hat{E}_2^{(k)})_j,\ \lambda_j < 0,\ x_1 = \gamma, \end{array} \right. \quad \Omega_2 : \left\{ \begin{array}{rcl} \dfrac{d\hat{E}_2^{(k+1)}}{dx_1} & = & -\mathcal{M}(\boldsymbol{\xi})\hat{E}_2^{(k+1)},\ x_1 > \beta, \\ (\hat{E}_2^{(k+1)})_j & = & (\hat{E}_1^{(k)})_j,\ \lambda_j > 0,\ x_1 = \beta, \end{array} \right. \tag{1.10}$$

where ($\mathbb{I}$ denotes the identity matrix)

$$\mathcal{M}(\boldsymbol{\xi}) = B_1^{-1}(b\ \mathbb{I} + i\sum_{m=2}^{d} B_m \xi_{m-1}). \tag{1.11}$$

In (1.10), the subscript j denotes the component of the error vector that must be imposed at a subdomain interface. We obtain local problems which for a given ξ are ODEs whose solutions can be expressed as linear combinations of the eigenvectors of $\mathcal{M}(\boldsymbol{\xi})$

$$\hat{E}_m^{(k)}(x_1, \boldsymbol{\xi}) = \sum_{j=1}^{p} (\alpha_j^m(\boldsymbol{\xi}))^{(k)} e^{-\mu_j(\boldsymbol{\xi})x_1} V_j(\boldsymbol{\xi}), \tag{1.12}$$

where $\mu_j(\boldsymbol{\xi})$ are the eigenvalues of $\mathcal{M}(\boldsymbol{\xi})$. Here we have assumed that the eigenvectors $V_j(\boldsymbol{\xi})$ of $\mathcal{M}(\boldsymbol{\xi})$ are linearly independent. Furthermore, we require that these local solutions are bounded at infinity ($-\infty$ and $+\infty$ respectively) which implies that, in the decomposition of $\hat{E}_1(x_1, \boldsymbol{\xi})$ (respectively $\hat{E}_2(x_1, \boldsymbol{\xi})$), we need to use the eigenvectors corresponding to the negative (respectively the positive) eigenvalues. Then, we replace the expressions of the local solutions (1.12) into the interface conditions (1.10) which results in

$$\begin{array}{ll} \Omega_1 : & \left(\displaystyle\sum_{j,\Re(\mu_j)<0} (\alpha_j^1(\boldsymbol{\xi}))^{(k+1)} e^{-\mu_j(\boldsymbol{\xi})\gamma} V_j(\boldsymbol{\xi}) \right)_l = \left(\displaystyle\sum_{j,\Re(\mu_j)>0} (\alpha_j^2(\boldsymbol{\xi}))^{(k)} e^{-\mu_j(\boldsymbol{\xi})\gamma} V_j(\boldsymbol{\xi}) \right)_l,\ \Re(\lambda_l) < 0, \\ \Omega_2 : & \left(\displaystyle\sum_{j,\Re(\mu_j)>0} (\alpha_j^2(\boldsymbol{\xi}))^{(k+1)} e^{-\mu_j(\boldsymbol{\xi})\beta} V_j(\boldsymbol{\xi}) \right)_l = \left(\displaystyle\sum_{j,\Re(\mu_j)<0} (\alpha_j^1(\boldsymbol{\xi}))^{(k)} e^{-\mu_j(\boldsymbol{\xi})\beta} V_j(\boldsymbol{\xi}) \right)_l,\ \Re(\lambda_l) > 0. \end{array} \tag{1.13}$$

By solving the above equations for the coefficients α_j^m, we obtain the interface iterations

$$(\alpha_j^1)_{j,\Re(\mu_j)<0}^{(k+1)}(\boldsymbol{\xi}) \ = \ \mathcal{T}_1(\alpha_j^2)_{j,\Re(\mu_j)>0}^{(k)}(\boldsymbol{\xi}), \quad (\alpha_j^2)_{j,\Re(\mu_j)>0}^{(k+1)}(\boldsymbol{\xi}) \ = \ \mathcal{T}_2(\alpha_j^1)_{j,\Re(\mu_j)<0}^{(k)}(\boldsymbol{\xi}). \tag{1.14}$$

Then, the square of the convergence rate of the $\boldsymbol{\xi}$-th component of the error vector of the Schwarz algorithm can be computed as the spectral radius of one of the matrix products $\mathcal{T}_1\mathcal{T}_2(\boldsymbol{\xi})$ or $\mathcal{T}_2\mathcal{T}_1(\boldsymbol{\xi})$

$$\rho_2^2 \equiv \rho_{\text{Schwarz2}}^2 = \rho(\mathcal{T}_1\mathcal{T}_2) = \rho(\mathcal{T}_2\mathcal{T}_1). \tag{1.15}$$

As a first step, we apply the general methodology described above to the solution of the two-dimensional Euler equations. We assume that the flow is subsonic that is, the local

normal Mach number defined by $M_n = \frac{u}{c}$ satisfies $|M_n| < 1$. We also assume that $u > 0$ and thus $0 < u < c$. Under these conditions, one can verify that $\Re(\mu_1) < 0$ and $\Re(\mu_{2,3,4}) > 0$. We introduce the dimensionless wave number $\overline{\xi} = \frac{c\xi}{b}$ and the associated dimensionless quantities

$$\begin{cases} \overline{a} & = & 1 + i\overline{\xi}M_t \; = \; \dfrac{a}{b}, & \overline{R}(\xi) = \sqrt{\overline{a}^2 + \overline{\xi}^2(1-M_n^2)} \; = \; \dfrac{R(\overline{\xi})}{b}, \\ \mu_{1,2} & = & \frac{c}{b}(\mu_1 - \mu_2) = -\dfrac{2\overline{R}(\overline{\xi})}{1-M_n^2}, & \mu_{1,3} = \frac{c}{b}(\mu_1 - \mu_3) = -\dfrac{\overline{a} + M_n\overline{R}(\overline{\xi})}{M_n(1-M_n^2)}, \end{cases}$$

where $M_n = \dfrac{u}{c}$ and $M_t = \dfrac{v}{c}$ respectively denote the local normal and tangential Mach numbers. Solving (1.13) and using (1.15) we obtain the square of the convergence rate of the algorithm

$$\rho_2^2(\overline{\xi},\overline{\delta}) = \left| \frac{[\overline{R}(\overline{\xi}) - \overline{a}]^2[\overline{a} + M_n\overline{R}(\overline{\xi})]}{[\overline{R}(\overline{\xi}) + \overline{a}]^2[\overline{a} - M_n\overline{R}(\overline{\xi})]} e^{\mu_{1,2}\overline{\delta}} - \frac{2[\overline{R}(\overline{\xi}) - \overline{a}][M_n(1-M_n)]\overline{R}(\overline{\xi})}{[\overline{R}(\overline{\xi}) + \overline{a}][1+M_n][\overline{a} - M_n\overline{R}(\overline{\xi})]} e^{\mu_{1,3}\overline{\delta}} \right|, \quad (1.16)$$

where $\overline{\delta} = \dfrac{\gamma - \beta}{c\Delta t}$ is the dimensionless size of the overlap. The following Proposition summarizes the convergence results of the additive Schwarz algorithm (1.6) applied to the two-dimensional Euler equations (1.19).

Proposition 1 *In the non-overlapping case, $\overline{\delta} = 0$, we have $\rho_2^2(\overline{\xi}, 0) < 1$ for all $\overline{\xi}$, M_n and M_t. In the overlapping case, for given values of M_n and M_t there exists $\overline{\delta}_0 > 0$ such that $\rho_2^2(\overline{\xi},\overline{\delta}) < 1$ for all $\overline{\delta} > \overline{\delta}_0$, $\overline{\xi}$.*

The proof is given in appendix of [6]. The above result has already been proved in the non-overlapping case by Clerc[Cle98] using an energy estimate .

Remark 2 *For small values of the overlap, we needed additional assumptions on the velocity field. Numerical experiments indicate that these assumptions are not necessary. The difficulty in proving the general result comes from the fact that the convergence rate is not decreasing with respect to the size of the overlap. For some small values of $\overline{\delta}$, we can find at least one pair of $(M_n \; , \; M_t)$ and an interval of wave numbers $[\overline{\xi}_1, \overline{\xi}_2]$ such that*

$$\rho_2^2(\overline{\xi},\overline{\delta}) > \rho_2^2(\overline{\xi}, 0) \quad \textit{for} \quad \overline{\xi} \in [\overline{\xi}_1, \overline{\xi}_2],$$

as can be seen on figure 1.1 which illustrates this behavior for $(M_n = 0.3 \; , \; M_t = 0.01)$. This behavior is very different from the one characterizing the scalar case where the convergence rate is a decreasing function of the size of the overlap, see for example [JNR01]. Moreover, the convergence rate does not depend on μ_4 since the fourth component of $U = T^{-1}W$ is decoupled from the others. By denoting the discrete Courant number

$$CFL_h = \frac{(M+1)c}{bh} = \frac{(M+1)c\Delta t}{h} \quad (1.17)$$

where h is a characteristic dimension of the grid used for space discretization, we see that if the overlap between subdomains is equal to h, then the dimensionless overlap size $\overline{\delta}$ found in the expression of the convergence rate is of the order $\frac{1}{CFL_h}$. Figure 1.1 shows that for large discrete Courant number, the overlap may decrease the performance of the algorithm.

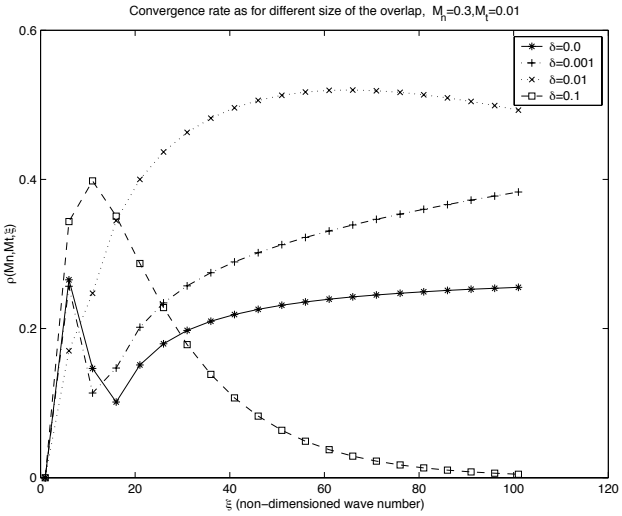


Figure 1.1: Convergence rate of the additive Schwarz algorithm for the 2D Euler equations vs. non-dimensioned wave-number; Two-subdomain decomposition (δ denotes the dimensionless size of the overlapping area)

For the non-overlapping case, Figure 1.2 represents the estimated number of Schwarz iterations for a threshold 10^{-10} ($Iter = -10/log_{10}\|\rho_2\|_\infty$ where $\|\rho_2\|_\infty = \max_{\xi\in\mathbb{R}} \rho_2(\xi)$) for a given value of the tangential Mach number, as a function of the normal Mach numbers M_n, and shows the global behavior of the convergence rate for different pairs of (M_n , M_t). In addition, the convergence rate as $\xi \to \infty$ satisfies

$$\lim_{\xi\to+\infty} \rho_2(k) = \sqrt{\left(\frac{1-3M_n}{1+M_n}\right)^2 + \frac{8M_nM_t^2}{(1+M_n)^3}} \quad < \quad 1. \tag{1.18}$$

In the particular case where $M_n = \frac{1}{3}$ and $M_t = 0$, this limit becomes null. This is surprising and certainly not expected. It is obtained for $v = 0$ everywhere in the flow field which is very particular and probably ideal situation.

Remark 3 *The inequality (1.18) has a numerical meaning: for a given discretization, let* $k_{\max}$ *denote the largest frequency that can be represented on a grid. This largest frequency is of the order* $\frac{\pi}{h}$ *where* h *denotes a characteristic grid size. The convergence rate of the additive Schwarz algorithm on this grid can be estimated by* $\rho_2^h = \max_{|k|<k_{\max}} \rho_2(k)$*. From (1.18), we have that* $\rho_2^h \leq \max_{k\in\mathbb{R}} \rho_2(k) < 1$ *meaning that for finer grids, the number of iterations may increase slightly but should not go to infinity.*

For a larger number of subdomains we cannot calculate easily the convergence rate using the previous technique because one needs to evaluate the spectral radius of a $4(N-1) \times 4(N-1)$ matrix where N is the number of subdomains. In [6] one can find the details of the computation of the convergence rate for the three-domain case and also for the two-domain

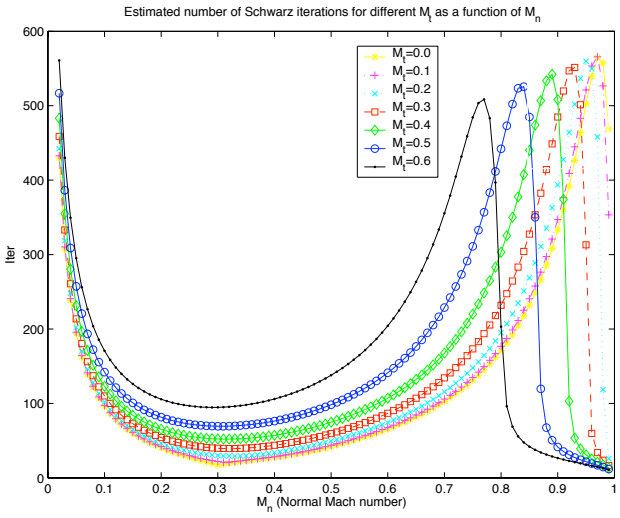


Figure 1.2: Estimated number of Schwarz iterations $Iter = -10/log_{10}(\|\rho_2\|_\infty)$
Two-subdomain non-overlapping decomposition, $\overline{\delta} = 0$

decomposition in three dimensions. We can conclude that the additive Schwarz algorithm (1.6) demonstrates a qualitatively similar behavior, irrespectively of the number of subdomains, when dealing with high frequencies and non-overlapping decompositions. This stems from the fact that the expression of the convergence rate in the three-subdomain case can be related to that obtained in the two-subdomain case using an overlapping decomposition. At the same time, these results are independent of the dimension of the problem. It is important to notice that, even in the non-overlapping case, the use of classical transmission conditions is sufficient to obtain a convergent algorithm.

Afterwards, we consider the same system with frozen coefficients and the same decomposition into two subdomains. We apply a finite volume scheme to discretize the system and express the discrete formulation of the Schwarz algorithm. By using a discrete Fourier transform, we derive the expression of the discrete convergence rate depending of the usual parameters and the mesh size.

We note that the expression of the convergence rate is very similar to the one obtained in the continuous case for an overlapping decomposition (see eq. (1.16)) with $\delta = \Delta x$. As we mentioned previously, this means that the decomposition into non-overlapping domains with reference to the finite volume cells, turns into a vertex oriented overlapping decomposition. This fact is further illustrated in figure 1.3 that compares the behaviors of the continuous overlapping and non-overlapping convergence rates with the discrete one.

We introduce the dimensionless wave number $\overline{k} = \frac{ck}{b}$ and the the dimensionless size of the overlap $\overline{\Delta}x = \frac{b\Delta x}{c}$. By comparing the discrete non-overlapping convergence rate with the continuous non-overlapping and overlapping convergence rates (with a minimal overlap),

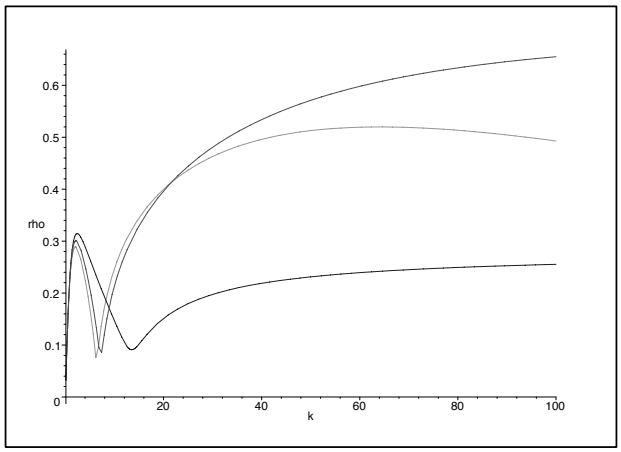


Figure 1.3: $\rho_2(\overline{k}, M_n, M_t, \overline{\Delta}x)$ for the Schwarz algorithm: continuous non-overlapping (bottom curve), continuous overlapping (middle curve), discrete overlapping (top curve) cases with $\overline{\Delta}x = 0.01$, $M_n = 0.3$, $M_t = 0.01$.

we show that the overlap is not a guarantee that the algorithm behaves better that in the non-overlapping case. Indeed, on figure 1.3 we can see that, in the continuous case, the convergence is better when there is no overlap. When overlap is introduced, we note that the convergence rate predicted in the continuous case is slightly better than the one obtained in the discrete case (for the same value of the dimensionless overlap).

Figure 1.4 represents the estimated number of Schwarz iterations for a threshold 10^{-10}(Iter $=-10/log_{10}\|\rho_2\|_\infty$) and for a tangential Mach number $M_t = 0.144M_n$ as a function of the normal Mach number M_n for a given value of the non-dimensioned overlap. This latter value, according to the remark 3 is of the order of $1/CFL_h$ ($CFL_h = 100$ in the numerical simulations) and $\overline{k}_{max} = \frac{c\Delta t\pi}{h} \cong \frac{\pi}{2}CFL_h$.

We will proceed to the numerical implementation of the above algorithm.The spatial discretization method adopted is based on a finite volume formulation together with an upwind scheme for the discretization of the convective flux (see [1] for details). The time integration of the resulting semi-discrete equations is done with the implicit linearized formulation described in Fezoui and Stoufflet[FS89]. Then, at each pseudo-time step, a linear system must be solved to advance the solution in time. This is where the domain decomposition approach is introduced. We thus focus on the solution of the linear system resulting from the first implicit time step starting from a uniform flow. The CFL number is set to 100 for all the numerical simulations. We consider two geometries: a rectangular domain of size $[0, 8] \times [0, 1]$ and a `NACA0012` airfoil. For the first geometry, two types of discretization are used : the first type consists of a regular triangulation (obtained from a finite difference grid) while the second type is an unstructured triangulation . The characteristics of the

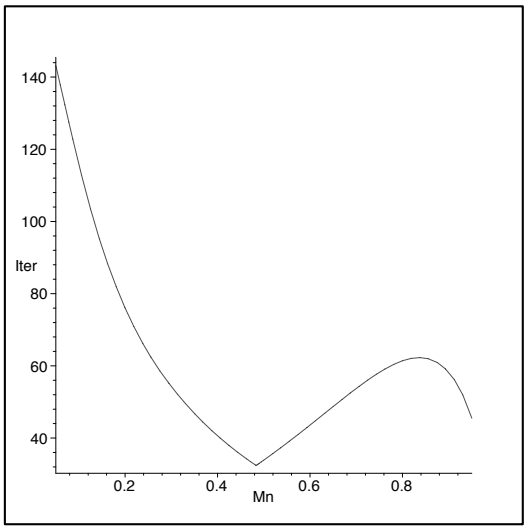


Figure 1.4: Estimated number of Schwarz iterations for a threshold $\varepsilon = 10^{-10}$ (Iter=$-10/\log_{10}\|\rho_2\|_\infty$);Two-subdomain decomposition, $\overline{\Delta}x = 0.01$, $M_t = 0.144M_n$, $\overline{k}_{max} = 150$

unstructured triangulations are given in table 1.2. The meshes RS2 and RS3 (respectively, meshes RU2 and RU3) have been obtained by uniform divisions of mesh RS1 (respectively, mesh RU1).

Table 1.1: Characteristics of the regular triangular meshes for the rectangular domain.

Mesh	# Vertexes	# Triangles	# Edges
RS1	4000	7562	11561
RS2	16000	31442	47441
RS3	64000	126962	190961

Table 1.2: Characteristics of the unstructured meshes for the rectangular domain.

Mesh	# Vertexes	# Triangles	# Edges
RU1	3740	7041	10780
RU2	14520	28164	42683
RU3	57203	112656	169858

For both geometries, the initialization is given by a uniform flow characterized by $\rho_0 = 1$, $u_0 = 1$, $v_0 = 0$ and $p_0 = \frac{1}{\gamma_s M^2}$ where M denotes the free stream Mach number. For the first geometry, a slip condition is applied on the horizontal sides while an inflow (respectively, outflow) condition is applied on the left (respectively, right) vertical side. For the `NACA0012` airfoil, three unstructured triangular meshes have been used whose characteristics are given in table 1.3.

Table 1.3: Characteristics of the meshes for the `NACA0012` airfoil.

Mesh	# Vertexes	# Triangles	# Edges
N1	3114	6056	9170
N2	12284	24224	36508
N3	48792	96896	145688

Mesh N2 and N3 have been obtained by uniform divisions of mesh N1. For all the following test cases we consider a threshold of $\varepsilon_i = 10^{-10}$ to reduce the normalized residual in the resolution of the interface system. The linear threshold for the solution of the local systems has been set to $\varepsilon_l = 10^{-10}$. The curves in the figures that follow represent, for each mesh, the required numbers of iterations to attain convergence as a function of a mean value of the local Mach number at the interface.

We consider the solution of the linear system resulting from the first time step for several flow conditions corresponding to values of the free-stream Mach number M ranging from 0.01 to 1.2 (which corresponds to a normal Mach number ranging from 0.007 to 1.0), using a two-subdomain decomposition of the meshes RS1 to RS3 of table 1.1 . Moreover in this case the interface Γ has a periodic broken linear shape since we make use of regular triangulations. If we denote by $\boldsymbol{n} = (n_1, n_2)$ and $\overline{\boldsymbol{n}} = (\overline{n}_1, \overline{n}_2)$ the two possible directions of the outward normal vector at this interface we can see that the local normal Mach number takes 2 values at the interface and the tangential Mach Number is not equal to 0 in practice. We get the following mean values for this parameters at the interface: $M_{nloc} = \frac{1}{2}(M(n_1 + \overline{n}_1))$ and $M_{tloc} = \frac{1}{2}(M(n_2 + \overline{n}_2))$ and we can see that the M_{tloc} depends linearly of M_{nloc}: $M_{tloc} = \frac{n_2+\overline{n}_2}{n_1+\overline{n}_1} M_{nloc}$, a simple calculation showing that $M_{tloc} = 0.144 M_{nloc}$ for this particular mesh. We can now compare the numerical results with the theoretical predictions showed in figure 1.4. The results are summarized in figure 1.1. We observe that, as the mesh is refined from RS1 to RS3, the number of iterations increases slightly. Moreover, the curves of figure 1.1 are in qualitative agreement with the theoretical behavior shown on figure 1.4. In the present case, the value M_n 0.5 always yield the best convergence of the Schwarz algorithm (when the flow is in the subsonic range).

As in the previous series of numerical experiments, we consider the solution of the linear system resulting from the first time step for several flow conditions corresponding to values of the free-stream Mach number M ranging from 0.1 to 1.2 (normal Mach number ranging from 0.007 to 0.9) and using a two-subdomain decomposition of meshes RU1 to RU3 of table 1.2. The results are summarized on figure 1.1 where we show, for each mesh, the required number of iterations to attain convergence. Clearly, this second series of experiments confirm the observations made when using the meshes based on regular triangulations.

As far as the numerical experiments involving the `NACA0012` geometry are concerned, contrary to the previous test cases, even though the initial flow is uniform, the solution ob-

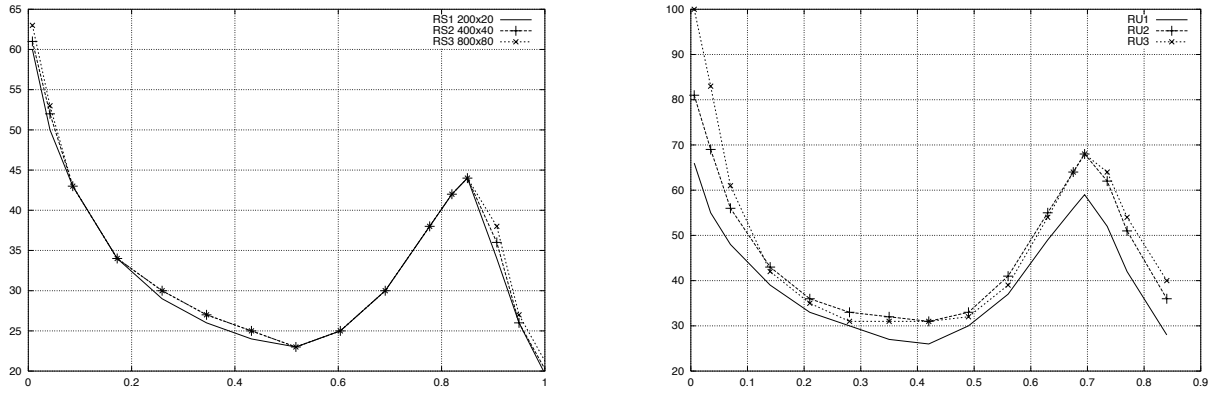


Figure 1.5: Convergence of the interface system: Flow inside a rectangular domain (uniform triangulations - left, non-uniform - right) with 2 subdomain decomposition
X-axis : Normal Mach number - Y-axis : # Schwarz iterations

tained at the end of the first time step is no more uniform due to the presence of the obstacle. In some sense, the constant coefficient theory developed is questionable. For these reasons, the correlation between theoretical and experimental results is only partially demonstrated for this test case.

As a conclusion, one can state that the classical Schwarz algorithm has already a very good behavior in the case of Euler equations that still needs to be confirmed on more realistic test cases. Moreover, a deepest analysis of an asymptotic behavior for small mesh sizes needs to be performed. As far as optimized interface conditions are concerned, there were some attempts in [9] based on a simple idea of a new flux, but the application to more realistic applications is still questionable.

1.2 Smith normal forms and domain decomposition

The originality of this work lies in the use of a new approach for the design of new domain decomposition methods, based on algebraic tools like the Smith factorization. This algebraic approach enables an intrinsic analysis of linear systems of PDEs. In particular, we are interested in transforming the linear system of PDEs into a set of decoupled PDEs under certain types of invertible transformations. Indeed, we can then derive new numerical algorithms based on efficient numerical methods developed for PDEs. The problem of decoupling the equations of linear time-varying systems of ordinary differential equations (ODEs) or difference equations has been an important issue in the symbolic computation community (e.g., methods based on the so-called eigenring). For instance, the techniques based on the eigenring can be considered as a generalization for time-varying linear systems of ODEs of the classical diagonalization method used for solving time-invariant first order linear system of ODEs. An alternative way for decoupling a linear system of ODEs is to use the so-called Smith normal form of the matrix of OD operators associated with the linear system. Since this algebraic tool is not familiar to the numerical analysis community, it seems useful to give a more detailed introduction. This normal form was introduced by H. J. S. Smith (1826-1883) for matrices with integer entries. As the ring of OD operators with constant coefficients is a commutative polynomial ring, we give here its polynomial form.

Theorem 3 *Let n be an positive integer and A an invertible $n \times n$ matrix with polynomial entries with respect to the variable λ: $A = (a_{ij}(\lambda))_{1 \leq i,j \leq n}$. Then, there exist matrices E, D and F with polynomial entries satisfying the following properties:*

- *$det(E)$ and $det(F)$ are constants,*
- *D is a diagonal matrix uniquely determined up to a multiplicative constant,*
- *$A = EDF$.*

Here E and F are matrices, which operate on the rows resp. columns. The entries of the diagonal matrix $D = (d_{ij}(\lambda))$ are given by $d_{ii} = \phi_i / \phi_{i-1}$, where ϕ_i is the greatest common divisor of the determinants of all $i \times i$ sub matrices of A and $\phi_0 = 1$.

The Smith factorization is a classical tool in computer algebra and in control of ordinary differential equations. Since its use in scientific computing is rather new, we give here a few comments:

- Smith was an English mathematician of the end of the 19th century. He worked in number theory and considered the problem of factorizing matrices with integer entries. We gave here the polynomial version of his theorem in the special case where the matrix A is square and invertible but the result is more general and applies as well when the matrix A is rectangular.

- One of the interest of the theorem is the following. By Cramer's formula, the inverse of A is in general a matrix with rational entries. By the Smith factorization, we have $A^{-1} = F^{-1}D^{-1}E^{-1}$. Since $det(E)$ and $det(F)$ are constants, the inverse of E and F are still matrices with polynomial entries in λ. The rational part of the inverse of A is thus in D^{-1} which is an intrinsic diagonal matrix.

- The proof of the theorem is constructive and gives an algorithm for computing matrices E, D and F. As stated in the theorem, matrix D is intrinsic but matrices E and F are not unique.

- we can write any system of PDEs as a matrix with partial differential operators entries applied to the unknown fields. The direction normal to the interface of the subdomains is particularized and denoted by ∂_x. Each partial differential operator is then considered as a polynomial in the "variable ∂_x" (e.g. λ is related to ∂_x and λ^2 to ∂_{xx}). It is then possible to apply the Smith factorization.

For domain decomposition methods, we have obtained several preliminary results on compressible Euler equations [11], Stokes and Oseen systems [19]. More precisely, we have shown that the Stokes system is equivalent to a biLaplacian in 2D and to a Laplacian and a biLaplacian in 3D. Hence, the classical Neumann-Neumann algorithm for the Stokes system can be recast into an algorithm for biLaplacian and Laplacian. The resulting algorithms are different from the classical algorithms for these scalar equations and are therefore not optimal. To fix this problem, we first used optimal algorithms for the biLaplacian and the Laplacian and then via the Smith factorization we back-transform them formally using only symbolic computations (derivation, linear combination of different equations) into optimal algorithms for the Stokes system. Interestingly enough, the classical algorithm for the Stokes system is based on solving either stress imposed or displacement imposed problems in the subdomains. In the new algorithm developed in [19] (normal stress, tangential displacement) or (tangential stress, normal displacement) problems are solved in each subdomain. These preliminary tests show that the new algorithm can be twice as fast as the classical one. Moreover, when solving time-dependent problems, the new algorithm has an iteration count independent of the time step. This property is not true for the classical algorithm. For both equations, the new algorithms are the first genuine extensions of the Neumann-Neumann or FETI algorithms.

In the future our aim is to extend the use of algebraic and symbolic techniques such as Smith normal forms and Gröbner basis techniques in order to develop new numerical methods for linear systems of partial differential equations that could appear in hydrodynamics, electromagnetism or geosismics. This theme is at the heart of the recently proposed ANR "Jeunes Chercheurs" SADDLES [1] project where I am the principal investigator. Even if the first results are very promising, not all difficulties have been solved. Moreover the approach can be applied to other very important systems of PDEs. The project will use principally the Smith factorization as an algebraic tool, and possibly other algebraic concepts as well.

[1] Symbolic Algebra, Domain Decomposition, Linear Equations and Systems

1.2.1 A new domain decomposition method for the compressible Euler equations

In this work we design a new domain decomposition method for the Euler equations in 2 dimensions. The starting point is the equivalence with a third order scalar equation to whom we can apply an algorithm inspired from the Robin-Robin preconditionner for the convection-diffusion equation [AN97]. Afterwards we translate it into an algorithm for the intial system and prove that at the continuous level and for a decomposition into 2 subdomains, it converges in 2 iterations. This property cannot be conserved strictly at discrete level and for arbitrary domain decompositions but we still have numerical results which confirm a very good stability with respect to the various parameters of the problem (mesh size, Mach number, ...).

The preconditioning methods have known a wide developpement in the last decade. The Neumann-Neummann algorithms for symmetric second order problems [BGLTV89, Man92, RT91] have been the subject of numerous works, see [TW04] and references therein. An extension of these algorithms to non-symmetric scalar problems (the so called Robin-Robin algorithms) has been done in [ALTNV00, GGTN04] for advection-diffusion problems. As far as optimized interface conditions are concerned, when dealing with supersonic flows, whatever the space dimension is, imposing the appropriate characteristic variables as interface conditions leads to a convergence of the algorithm which is optimal with regards to the number of subdomains. This property is generally lost for subsonic flows except for the case of one-dimensional problems, when the optimality is expressed by the fact that the number of iterations is equal to the number of subdomains (see Bjørhus [Bjø95a] and Quarteroni [Qua90] for more details). In the subsonic case and in two or three dimensions, we can find a formulation with classical (natural) transmission conditions in [Qua90, CFS98, QS96] or with more general interface conditions in [Cle98] and optimized transmission conditions in [9]. The analysis of such algorithms applied to systems proved to be very different from the scalar case, see [4, 6]. The generalization of the above domain decomposition methods to the system of the Euler equations is difficult in the subsonic case in dimensions equal or higher to two. As far as preconditionning methods are concerned, to our knowledge, no extension of the Neumann-Neumann, FETI [Li05] of BDDC [LW06] methods to the Euler equations was done.

We will first show the equivalence between the 2D Euler equations and a third order scalar problem, which is quite natural by considering a Smith factorization of this system, see [WRL95] or [Gan66]. We define an optimal algorithm for the third order scalar equation. It is inspired from the idea of the Robin-Robin algorithm [ALTNV00] applied to a convection-diffusion problem. We also prove by using a Fourier analysis that this algorithm converges in two iterations. Afterwards we back-transform it and define the corresponding algorithm applied to the Euler system. All the previous results have been obtained at the continuous level and for a decomposition into 2 unbounded subdomains. After a discretization in a bounded domain we cannot expect that these properties to be conserved exactly. Still we can show by a discrete convergence analysis that the expected results should be very good. Numerical results confirm the very good stability of the algorithm with respect to the various

parameters of the problem (mesh size, Mach number, ...).

We will focus on the conservative Euler equations in two-dimensions:

$$\frac{\partial W}{\partial t} + \nabla.\boldsymbol{F}(W) = 0 \quad , \quad W = (\rho, \rho\boldsymbol{V}, E)^t \, . \tag{1.19}$$

In the above expressions, ρ is the density, $\boldsymbol{V} = (u, v)^t$ is the velocity vector, E is the total energy per unit of volume and p is the pressure. In equation (1.19), $W = W(\mathbf{x}, \mathbf{t})$ is the vector of conservative variables, $\mathbf{x}$ and t respectively denote the space and time variables and $\boldsymbol{F}(W) = (F_1(W), F_2(W))^T$ is the conservative flux vector whose components are given by

$$F_1(W) = \big(\rho u, \rho u^2 + p, \rho uv, u(E+p)\big)^t \, , \quad F_2(W) = \big(\rho v, \rho uv, \rho v^2 + p, v(E+p)\big)^t \, .$$

The pressure is deduced from the other variables using the state equation for a perfect gas $p = (\gamma_s - 1)(E - \frac{1}{2}\rho \parallel \boldsymbol{V} \parallel^2)$ where γ_s is the ratio of the specific heats ($\gamma_s = 1.4$ for the air).

The starting point of our analysis is given by the linearized form of the Euler equations (1.19) written in primitive variables (p, u, v, S). In the following we suppose that the flow is isentropic, which allows us to drop the equation of the entropy (which is totally decoupled from the others). We denote by $W = (P, U, V)^T$ the vector of unknowns and by A and B the jacobian matrices of the fluxes $F_i(w)$ to whom we already applied the variable change from conservative to primitive variables. In the following, we shall denote by $\overline{c}$ the speed of the sound and we consider the linearized form (we will mark by the bar symbol, the state around which we linearize) of the Euler equations:

$$\mathcal{P}W \equiv \frac{W}{\Delta t} + A\partial_x W + B\partial_y W = f \tag{1.20}$$

characterized by the following jacobian matrices:

$$A = \begin{pmatrix} \overline{u} & \overline{\rho c}^2 & 0 \\ 1/\overline{\rho} & \overline{u} & 0 \\ 0 & 0 & \overline{u} \end{pmatrix} \qquad B = \begin{pmatrix} \overline{v} & 0 & \overline{\rho c}^2 \\ 0 & \overline{v} & 0 \\ 1/\overline{\rho} & 0 & \overline{v} \end{pmatrix} \tag{1.21}$$

We can re-write the system (1.20) by denoting $\beta = \frac{1}{\Delta t} > 0$ under the form

$$\mathcal{P}W \equiv (\beta I + A\partial_x + B\partial_y)\, W = f \tag{1.22}$$

In Computational Fluid Dynamics, problems of the form (1.22) have to be solved repeatedly. We shall design a new domain decomposition method for this purpose. We build and analyze our method for the constant coefficient case ($\overline{c}$, $\overline{u}$, $\overline{v}$ and $\overline{\rho}$ are constants) and for only two subdomains. But the resulting algorithm can be applied to the general case of variable flows and arbitrary number of subdomains.

We first take formally the Fourier transform of the system (1.22) with respect to y (the dual variable is ξ). We keep the partial derivatives in x since in the sequel we shall consider a domain decomposition with an interface whose normal is in the x direction. We note

$$\hat{\mathcal{P}} = \begin{pmatrix} \beta + \overline{u}\partial_x + i\xi\overline{v} & \overline{\rho c}^2\partial_x & i\overline{\rho c}^2\xi \\ \frac{1}{\overline{\rho}}\partial_x & \beta + \overline{u}\partial_x + i\xi\overline{v} & 0 \\ \frac{i\xi}{\overline{\rho}} & 0 & \beta + \overline{u}\partial_x + i\overline{v}\xi \end{pmatrix} \tag{1.23}$$

We can perform a Smith factorization of $\hat{\mathcal{P}}$ by considering it as a matrix with polynomials in ∂_x entries. We have

$$\hat{\mathcal{P}} = EDF \tag{1.24}$$

where

$$D = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \hat{\mathcal{L}}\hat{\mathcal{G}} \end{pmatrix} \tag{1.25}$$

and

$$E = \begin{pmatrix} i\overline{\rho c}^2\xi & 0 & 0 \\ 0 & \overline{u} & 0 \\ \beta + \overline{u}\partial_x + i\overline{v}\xi & E_2 & \dfrac{\overline{c}^2 - \overline{u}^2}{i\xi\overline{\rho c}^2} \end{pmatrix}$$

and

$$F = \begin{pmatrix} \dfrac{\beta + \overline{u}\partial_x + i\xi\overline{v}}{i\xi\overline{\rho c}^2} & \dfrac{\partial_x}{i\xi} & 1 \\ \dfrac{\partial_x}{\overline{\rho u}} & \dfrac{\beta + \overline{u}\partial_x + i\xi\overline{v}}{\overline{u}} & 0 \\ \dfrac{1}{(\beta + i\xi\overline{v})(\overline{u}^2 - \overline{c}^2)} & \dfrac{\overline{\rho u}}{(\beta + i\xi\overline{v})(\overline{u}^2 - \overline{c}^2)} & 0 \end{pmatrix} \tag{1.26}$$

where

$$E_2 = \overline{u}\frac{(-\overline{u}\overline{c}^2 + \overline{u}^3)\partial_{xx} + (2\overline{u}^2 - \overline{c}^2)(\beta + i\xi\overline{v})\partial_x + \overline{u}((\beta + i\xi\overline{v})^2 + \xi^2\overline{c}^2)}{\overline{c}^2(i\beta + i\xi\overline{v})},$$

$$\hat{\mathcal{G}} = \beta + \overline{u}\partial_x + i\xi\overline{v} \tag{1.27}$$

and

$$\hat{\mathcal{L}} = \beta^2 + 2i\xi\overline{uv}\partial_x + 2\beta(\overline{u}\partial_x + i\xi\overline{v}) + (\overline{c}^2 - \overline{v}^2)\xi^2 - (\overline{c}^2 - \overline{u}^2)\partial_{xx} \tag{1.28}$$

The operators showing up in the diagonal matrix have a physical meaning:

$$\mathcal{G} = \beta + \overline{u}\partial_x + \overline{v}\partial_y$$

is a first order transport operator where the time derivative is replaced by β and

$$\mathcal{L} = \beta^2 + 2\overline{uv}\partial_{xy} + 2\beta(\overline{u}\partial_x + \overline{v}\partial_y) - (\overline{c}^2 - \overline{v}^2)\partial_{yy} - (\overline{c}^2 - \overline{u}^2)\partial_{xx}$$

is the advective wave operator where ∂_t^l is replaced by β^l for $l = 1, 2$. We call $W_s = FW$ the Smith variables.

Equation (1.25) suggests that the derivation of a domain decomposition method (DDM) for the third order operator $\mathcal{LG}$ is a key ingredient for a DDM for the compressible Euler equations. We want now to solve

$$\mathcal{LG}(Q) = g \tag{1.29}$$

where Q is scalar unknown function and g is a given right hand side. The algorithm will be based on the Robin-Robin algorithm [AN97, ALTNV00] for the convection-diffusion problem. Then we will prove its convergence in 2 iterations. We first note that the elliptic operator $\mathcal{L}$ can also be written as:

$$\mathcal{L} = -div(A\nabla) + \mathbf{a}\nabla + \beta^2,\ A = \begin{pmatrix} \overline{c}^2 - \overline{u}^2 & -\overline{uv} \\ -\overline{uv} & \overline{c}^2 - \overline{v}^2 \end{pmatrix} \text{ where } \mathbf{a} = 2\beta(\overline{u}, \overline{v}) \tag{1.30}$$

Without loss of generality we assume in the sequel that the flow is subsonic and that $\overline{u} > 0$ and thus we have $0 < \overline{u} < \overline{c}$. We consider now a decomposition of the plane $\mathbb{R}^2$ into two non-overlapping sub-domains $\Omega_1 = (-\infty, 0) \times \mathbb{R}$ and $\Omega_2 = (0, \infty, 0) \times \mathbb{R}$. The interface is $\Gamma = \{x = 0\}$. The outward normal to domain Ω_i is denoted $\mathbf{n_i}$, $i = 1, 2$. Let $Q^{i,k}$, $i = 1, 2$ represent the approximation to the solution in subdomain i at the iteration k of the algorithm. We define the following algorithm:

ALGORITHM 1 *We choose the initial values $Q^{1,0}$ and $Q^{2,0}$ such that $\mathcal{G}Q^{1,0} = \mathcal{G}Q^{2,0}$. We compute $(Q^{i,k+1})_{i=1,2}$ from $(Q^{i,k})_{i=1,2}$ by the following iterative procedure:*
Correction step *We compute the corrections $\tilde{Q}^{1,k}$ and $\tilde{Q}^{2,k}$ as solution of the homogeneous local problems:*

$$\begin{cases} \mathcal{LG}\tilde{Q}^{1,k} = 0 \ in\ \Omega_1, \\ (A\nabla - \frac{1}{2}\mathbf{a})\mathcal{G}\tilde{Q}^{1,k} \cdot \mathbf{n_1} = \gamma^k, \ on\ \Gamma. \end{cases} \quad \begin{cases} \mathcal{LG}\tilde{Q}^{2,k} = 0 \ in\ \Omega_2, \\ (A\nabla - \frac{1}{2}\mathbf{a})\mathcal{G}\tilde{Q}^{2,k} \cdot \mathbf{n_2} = \gamma^k, \ on\ \Gamma, \\ \tilde{Q}^{2,k} = 0, \ on\ \Gamma. \end{cases} \tag{1.31}$$

where $\gamma^k = -\frac{1}{2}\left[A\nabla\mathcal{G}Q^{1,k} \cdot \mathbf{n_1} + A\nabla\mathcal{G}Q^{2,k} \cdot \mathbf{n_2}\right]$.
Update step. *We update $Q^{1,k+1}$ and $Q^{2,k+1}$ by solving the local problems:*

$$\begin{cases} \mathcal{LG}Q^{1,k+1} = g, \ in\ \Omega_1, \\ \mathcal{G}Q^{1,k+1} = \mathcal{G}Q^{1,k} + \delta^k, \ on\ \Gamma. \end{cases} \quad \begin{cases} \mathcal{LG}\tilde{Q}^{2,k+1} = g, \ in\ \Omega_2, \\ \mathcal{G}Q^{2,k+1} = \mathcal{G}Q^{2,k} + \delta^k, \ on\ \Gamma, \\ Q^{2,k+1} = Q^{1,k} + \tilde{Q}^{1,k}, \ on\ \Gamma. \end{cases} \tag{1.32}$$

where $\delta^k = \frac{1}{2}\left[\mathcal{G}\tilde{Q}^{1,k} + \mathcal{G}\tilde{Q}^{2,k}\right]$.

Proposition 2 *Algorithm 1 converges in 2 iterations.*

The proof can be found in [11] and uses Fourier transform techniques. After having found an optimal algorithm which converges in two iterations for the third order model problem, we focus on the Euler system by translating this algorithm into an algorithm for the Euler system. It suffices to replace the operator $\mathcal{LG}$ by the Euler system and Q by the last component $F(W)_3$ of $F(W)$ in the boundary conditions. The algorithm reads:

ALGORITHM 2 *We choose the initial values $W^{1,0}$ and $W^{2,0}$ such that $\mathcal{G}F(W^{1,0})_3 = \mathcal{G}F(W^{2,0})_3$ and we compute $(W^{i,k+1})_{i=1,2}$ from $(W^{i,k})_{i=1,2}$ by the following iterative procedure:*
Correction step *We compute the corrections $\tilde{W}^{1,k}$ and $\tilde{W}^{2,k}$ as solution of the homogeneous local problems:*

$$\left\{\begin{array}{l} \mathcal{P}\tilde{W}^{1,k} = 0 \ in\ \Omega_1, \\ \\ (A\nabla - \frac{1}{2}\mathbf{a})\mathcal{G}F(\tilde{W}^{1,k})_3 \cdot \mathbf{n_1} = \gamma^k, \ on\ \Gamma. \end{array}\right. \quad \left\{\begin{array}{l} \mathcal{P}\tilde{W}^{2,k} = 0 \ in\ \Omega_2, \\ \\ (A\nabla - \frac{1}{2}\mathbf{a})\mathcal{G}F(\tilde{W}^{2,k})_3 \cdot \mathbf{n_2} = \gamma^k, \ on\ \Gamma, \\ \\ \tilde{F}(W^{2,k})_3 = 0, \ on\ \Gamma. \end{array}\right. \tag{1.33}$$

where $\gamma^k = -\frac{1}{2}\left[A\nabla\mathcal{G}F(W^{1,k})_3 \cdot \mathbf{n_1} + A\nabla\mathcal{G}F(W^{2,k})_3 \cdot \mathbf{n_2}\right]$.
Update step. *We update $W^{1,k+1}$ and $W^{2,k+1}$ by solving the local problems:*

$$\left\{\begin{array}{l} \mathcal{P}W^{1,k+1} = f, \ in\ \Omega_1, \\ \\ \mathcal{G}F(W^{1,k+1})_3 = \mathcal{G}F(W^{1,k})_3 + \delta^k, \ on\ \Gamma. \end{array}\right. \quad \left\{\begin{array}{l} \mathcal{P}\tilde{W}^{2,k+1} = f, \ in\ \Omega_2, \\ \\ \mathcal{G}F(W^{2,k+1})_3 = \mathcal{G}F(W^{2,k})_3 + \delta^k, \ on\ \Gamma, \\ \\ F(W^{2,k+1})_3 = F(W^{1,k})_3 + F(\tilde{W}^{1,k})_3, \ on\ \Gamma. \end{array}\right. \tag{1.34}$$

where $\delta^k = \frac{1}{2}\left[\mathcal{G}F(\tilde{W}^{1,k})_3 + \mathcal{G}F(\tilde{W}^{2,k})_3\right]$.

This algorithm is quite complex since it involves second order derivatives of the unkowns in the boundary conditions on $\mathcal{G}F(W)_3$. It is possible to simplify it. By using the Euler equations in the subdomain, we have lowered the degree of the derivatives in the boundary conditions. After lengthy computations that we omit here, we find a simpler algorithm. We write it for a decomposition in two subdomains with an outflow velocity at the interface of domain Ω_1 but with an interface not necessarily rectilinear. In this way, it is possible to figure out how to use for a general domain decomposition.

In the sequel, $\mathbf{n} = (n_x, n_y)$ denotes the outward normal to domain Ω_1, $\partial_n = \nabla \cdot \mathbf{n} = (\partial_x, \partial_y) \cdot \mathbf{n}$ the normal derivative at the interface, $\partial_\tau = (-\partial_y, \partial_x) \cdot \mathbf{n}$ the tangential derivative, $U_n = Un_x + Vn_y$ and $U_\tau = -Un_y + Vn_x$ are respectively the normal and tangential velocity at the interface between the subdomains. Similarly, we denote $\overline{u}_n$ (resp. $\overline{u}_\tau$) the normal (resp. tangential) component of the velocity around which we have linearized the equations.

ALGORITHM 3 *We choose the initial values $W^{i,0} = (P^{i,0}, U^{i,0}, V^{i,0})$, $i = 1, 2$ such that $P^{1,0} = P^{2,0}$ and we compute $W^{i,k+1}$ from $W^{i,k}$ by the iterative procedure with two steps:*
Correction step *We compute the corrections $\tilde{W}^{1,k}$ and $\tilde{W}^{2,k}$ as solution of the homogeneous*

local problems:

$$\left\{\begin{array}{l} \mathcal{P}\tilde{W}^{1,k} = 0, \ in\ \Omega_1, \\ -(\beta + \overline{u}_\tau\partial_\tau)\tilde{U}_n^{1,n} + \overline{u}_n\partial_\tau\tilde{U}_\tau^{1,k} = \gamma^k, \ on\ \Gamma. \end{array}\right. \quad \left\{\begin{array}{l} \mathcal{P}\tilde{W}^{2,k} = 0, \ in\ \Omega_2, \\ (\beta + \overline{u}_\tau\partial_\tau)\tilde{U}_n^{2,k} - \overline{u}_n\partial_\tau\tilde{U}_\tau^{2,k} = \gamma^k, \ on\ \Gamma \\ \tilde{P}^{2,k} + \overline{\rho u}_n\tilde{U}_n^{2,k} = 0, \ on\ \Gamma. \end{array}\right. \tag{1.35}$$

where $\gamma^k = -\frac{1}{2}\left[(\beta + \overline{u}_\tau\partial_\tau)(U_n^{2,k} - U_n^{1,k}) + \overline{u}_n\partial_\tau(\tilde{U}_\tau^{1,k} - \tilde{U}_\tau^{2,k})\right]$.

Update step. *We compute the update of the solution* $W^{1,k+1}$ *and* $W^{2,k+1}$ *as solution of the local problems:*

$$\left\{\begin{array}{l} \mathcal{P}W^{1,k+1} = f_1, \ in\ \Omega_1, \\ P^{1,k+1} = P^{1,k} + \delta^k, \ on\ \Gamma. \end{array}\right. \quad \left\{\begin{array}{l} \mathcal{P}W^{2,k+1} = f_2, \ in\ \Omega_2, \\ P^{2,k+1} = P^{2,k} + \delta^k, \ on\ \Gamma, \\ (P + \overline{\rho u}_n U_n)^{2,k+1} = (P + \overline{\rho u}_n U_n)^{1,k} + (\tilde{P} + \overline{\rho u}_n\tilde{U}_n)^{1,k}, \ on\ \Gamma. \end{array}\right. \tag{1.36}$$

where $\delta^k = \frac{1}{2}\left[\tilde{P}^{1,k} + \tilde{P}^{2,k}\right]$.

Proposition 3 *For a domain* $\Omega = \mathbb{R}^2$ *divided into two non overlapping half planes, algorithms 2 and 3 are equivalent and both converge in two iterations.*

As far as the discretization is concerned, we used a finite volume method on a uniform grid. Then we propose a strategy of discretization of the boundary conditions of the algorithm 3 applied to the Euler system and we present some theoretical discrete estimates of the convergence rate of the method. A discrete convergence analysis will allow to decide which discretization of the boundary conditions is better.

We present here a set of results of numerical experiments on a model problem. We compare the method proposed and the classical method defined in [6] whre we formulated a Schwarz algorithm (interface iteration which relies on the successive solving of the local decomposed problems and the transmission of the result at the interface) involving transmission conditions that are derived naturally from a weak formulation of the underlying boundary value problem. We considered a decomposition into different number of subdomains and for a linearization around a constant or non-constant flow. The computational domain is given by the rectangle $[0\ ,\ 4] \times [0\ ,\ 1]$ with a uniform discretization using 80×20 points. The numerical investigation is limited to the resolution of the linear system resulting from the first implicit time step using a Courant number CFL=100. In the following, for the new algorithm, each iteration counts for 2 as we need to solve twice as much local problems than the classical one. For an easier comparison of the algorithms, the figures shown in the tables are the number of subdomains solves. We also used substructuring (solving a system with interface variables only) and the iteration number necessary to achieve convergence by

M_n	Classical (iterative)	Classical (GMRES)	New DDM (iter)	New DDM (GMRES)
0.001	32	26	16	16
0.01	30	26	16	16
0.1	28	21	14	14
0.2	24	19	14	14
0.3	20	16	14	14
0.4	18	14	14	14
0.5	16	13	14	14
0.6	15	12	14	14
0.7	14	11	14	14
0.8	14	11	14	14

Table 1.4: Iteration count for different values of M_n, $M_t(y)$

h ($M_n = 0.001$)	Classical	New DDM	h ($M_n = 0.1$)	Classical	New DDM
1/10	65	18	1/10	56	12
1/20	67	18	1/20	57	14
1/40	70	18	1/40	59	16

Table 1.5: Iteration count for different mesh size

means of a GMRES method is also presented. We are solving the homogeneous equations verified by the error vector at the first time step.

We consider first a decomposition into 2 subdomains and a linearization around a variable state where the tangential velocity is given by the expression $M_t(y) = 0.1(1 + \cos(\pi y))$ and the normal Mach number remains constant at the interface. The results for different values of M_n are presented in Table 1.4. We will linearize now the equations around a variable state for a general flow at the interface where the tangential Mach number is given by $M_t = 0.1(1 + \cos(\pi y))$, and the initial normal velocity is given by the expression $M_n(y) = 0.5(0.2 + 0.04\tanh(y/0.2)))$. The iterative version of the new algorithm converges in 18 iterations whereas the classical one need 45 iterations to attain the same tolerance. For the accelerated version the new algorithm needs 14 iterations and the classical one 21, to achieve convergence. The sensitivity to the mesh size is shown in the Table 1.5 for a normal flow ($M_t = 0.0$) at the interface and small Mach numbers. We can see that for the new algorithm the growth in the number of iterations is very weak as the mesh is refined, the same property being already known for the classical one.

The next set of tests concerns a stripwise decomposition into 3 subdomains. The same kind of tests are carried out as in the 2 subdomain case. Table 1.6 summarizes the number of Schwarz iterations required to reduce the initial linear residual by a factor 10^{-6} for different values of the reference Mach number for the new and the classical algorithm (the tangential velocity is given by the expression $M_t(y) = 0.1(1 + \cos(\pi y)))$. For a linearization around a variable state for a general flow at the interface where the tangential Mach number is given by $M_t = 0.1(1 + \cos(\pi y))$, and the initial normal velocity is given by the expression $M_n(y) = 0.5(0.2+0.04\tanh(y/0.2)))$, the same conclusion yield as in the two-domain case. As of intermediate conclusion we can state that the iteration number is only slightly increasing

M_n	Classical (iterative)	Classical (GMRES)	New DDM (iter)	New DDM (GMRES)
0.001	32	26	20	16
0.01	31	26	20	16
0.1	29	21	18	16
0.2	25	19	18	16
0.3	23	16	18	16
0.4	21	15	16	16
0.5	19	13	16	14
0.6	16	12	16	14
0.7	14	11	16	14
0.8	13	11	16	14

Table 1.6: Iteration count for different values of M_n

M_n	Classical(iter)	Classical (GMRES)	New DDM (GMRES)
0.001	101	28	28
0.01	86	28	28
0.1	54	26	26
0.2	38	23	30
0.3	35	23	32

Table 1.7: Iteration count for different values of M_n

when going from 2 to 3 subdomains.

The next set of tests concerns a decomposition into 4 subdomains using a 2×2 decomposition of a $40 \times 40 = 1600$ point mesh. No special treatement of the cross points is done or coarse space added. This could be a reason why the iterative version of the algorithm doesn't converge. Nevertheless, the accelerated algorithm by a GMRES method converges as showed in Table 1.7 which summarizes the number if iterations for different values of the reference Mach number for both algorithms (the tangential velocity is given by the expression $M_t(y) = 0.1(1+\cos(\pi y))$ and the normal Mach number is constant at the interface). We can see the the new algorithm behaves similarly as the classical one for low Mach numbers. The latest results show clearly the need of a coarse space as this is done for the FETI-DP methods, in order to improve the performance of the method which has already shown promising results in the case of the stripwise decompositions.

As a conclusion, in this work we designed a new domain decomposition for the Euler equations inspired by the idea of the Robin-Robin preconditionner applied to the advection-diffusion equation. We used the same principle after reducing the system to scalar equations via a Smith factorization. The resulting algorithm behaves very well for the low Mach numbers, where usually the classical algorithm doesn't give very good results. We can reduce the number of iteration by almost a factor 4 both for linearization around a constant and variable state.

1.2.2 Deriving a new domain decomposition method for the Stokes equations using the Smith factorization

The last decade has shown, that Neumann-Neumann type algorithms, FETI, and BDDC methods are very efficient domain decomposition methods. Most of the early theoretical and numerical work has been carried out for scalar symmetric positive definite second order problems. n the literature one can also find other preconditioners for the Schur complement of the Stokes equations (cf. [TP97, AS99]). Moreover, there exist some Schwarz-type algorithms for non-overlapping decompositions (cf. [OLM01, OL98, Nat97, Ron96]). A more complete list of domain decomposition methods for the Stokes equations can be found in [PW02, TW04]. Also FETI [Li05] and BDDC methods [LW06] are applied to the Stokes problem with success. Our work is motivated by the fact that in some sense the domain decomposition methods for Stokes are less optimal than the domain decomposition methods for scalar problems. Indeed, in the case of two subdomains consisting of the two half planes it is well known, that the Neumann-Neumann preconditioner is an exact preconditioner for the Schur complement equation for scalar equations like the Laplace problem (cf. [RT91]). A preconditioner is called *exact*, if the preconditioned operator simplifies to the identity. Unfortunately, this does not hold in the vector case. It is shown in [NR07] that the standard Neumann-Neumann preconditioner for the Stokes equations does not possess this property.

In the following we show the equivalence between the Stokes system and a fourth order scalar problem (the bi-harmonic problem) by means of the Smith factorization. This is motivated by the fact that scalar problems are easier to manipulate and the construction of new algorithms is more intuitive. Additionally, the existing theory of scalar problems can be used. We consider the stationary Stokes problem in a bounded domain $\Omega \subset \mathbb{R}^d$, $d = 2, 3$. The Stokes equations are given by a velocity $\boldsymbol{u}$ and a pressure p satisfying

$$\begin{aligned} -\nu\Delta\boldsymbol{u} + \nabla p + c\boldsymbol{u} &= \boldsymbol{f} \quad \text{in } \Omega \\ \nabla \cdot \boldsymbol{u} &= 0 \quad \text{in } \Omega \end{aligned}$$

and some boundary conditions on $\partial\Omega$. The Stokes problem is a simple model for incompressible flows. The right hand side $\boldsymbol{f} = (f_1, \ldots, f_d)^T \in [L^2(\Omega)]^d$ is a source term, ν is the viscosity and $c \geq 0$ is a constant reaction coefficient. Very often c stems from an implicit time discretization and then c is given by the inverse of the time step size.

In the following we denote the d-dimensional Stokes operator by $\mathcal{S}_d(\boldsymbol{v}, q) := (-\nu\Delta\boldsymbol{v} + c\boldsymbol{v} + \nabla q, \nabla \cdot \boldsymbol{v})$. The Smith factorization is applied to the following model problem in the whole plane $\mathbb{R}^2$

$$\begin{aligned} \mathcal{S}_d(\boldsymbol{u}, p) &= \boldsymbol{g} \quad \text{in } \mathbb{R}^2 && (1.37) \\ |\boldsymbol{u}(\boldsymbol{x})| &\to 0 \quad \text{for } |\boldsymbol{x}| \to \infty && (1.38) \end{aligned}$$

with right hand side $\boldsymbol{g} = (f_1, f_2, 0)^T$. Moreover, it is assumed, that the coefficients c, ν are constants. We start with the two-dimensional case. The spatial coefficients are denoted by x and y. In order to apply the factorization to the Stokes system, we first take formally the

Fourier transform of (1.37) with respect to y. The dual variable is denoted by k. The Fourier transform of a function f is written as $\hat{f}$ or $\mathcal{F}_y f$. Thus, equation (1.37) yields $\hat{\mathcal{S}}_2(\hat{\boldsymbol{u}}, \hat{p}) = \hat{\boldsymbol{g}}$ with $\hat{\boldsymbol{u}} = (\hat{u}, \hat{v})$ and

$$\hat{\mathcal{S}}_2(\hat{\boldsymbol{u}}, \hat{p}) = \begin{pmatrix} -\nu(\partial_{xx} - k^2) + c & 0 & \partial_x \\ 0 & -\nu(\partial_{xx} - k^2) + c & ik \\ \partial_x & ik & 0 \end{pmatrix} \begin{pmatrix} \hat{u} \\ \hat{v} \\ \hat{p} \end{pmatrix}. \tag{1.39}$$

Considering $\hat{\mathcal{S}}_2(\hat{\boldsymbol{u}}, \hat{p})$ as a matrix with polynomial entries with respect to ∂_x, we perform for $k \neq 0$ the Smith factorization. We obtain

$$\hat{\mathcal{S}}_2 = \hat{E}_2 \hat{D}_2 \hat{F}_2 \tag{1.40}$$

with

$$\hat{D}_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & (\partial_{xx} - k^2)\hat{\mathcal{L}}_2 \end{pmatrix}, \qquad \hat{F}_2 = \begin{pmatrix} \nu k^2 + c & \nu ik\partial_x & \partial_x \\ 0 & \hat{\mathcal{L}}_2 & ik \\ 0 & 1 & 0 \end{pmatrix}$$

and

$$\hat{E}_2 = \hat{T}_2^{-1} \begin{pmatrix} ik\hat{\mathcal{L}}_2 & \nu\partial_{xxx} & -\nu\partial_x \\ 0 & \hat{T}_2 & 0 \\ ik\partial_x & -\partial_{xx} & 1 \end{pmatrix}$$

where T_2 is a differential operator in y-direction whose symbol is $ik(\nu k^2 + c)$. Moreover, $\hat{\mathcal{L}}_2 := \nu(-\partial_{xx} + k^2) + c$ is the Fourier transform of $\mathcal{L}_2 := -\nu\Delta + c$.

Remark 4 *Using this factorization, problem (1.37) can be written as*

$$\hat{D}_2 \hat{\boldsymbol{w}} = \hat{E}_2^{-1} \hat{\boldsymbol{g}}, \qquad \hat{\boldsymbol{w}} := (\hat{w}_1, \hat{w}_2, \hat{w}_3)^T := \hat{F}_2(\hat{\boldsymbol{u}}, \hat{p})^T. \tag{1.41}$$

From (1.41) we get $\hat{w}_1 = (\hat{E}_2^{-1}\hat{\boldsymbol{g}})_1$ *and* $\hat{w}_2 = (\hat{E}_2^{-1}\hat{\boldsymbol{g}})_2$. *Noticing that* $\hat{w}_3 = \left(\hat{F}_2(\hat{\boldsymbol{u}}, \hat{p})^T\right)_3 = \hat{v}$ *the previous equation yields after applying an inverse Fourier transform*

$$\Delta(-\nu\Delta + c)v = \mathcal{F}_y^{-1}\left((\hat{E}_2^{-1}\hat{\boldsymbol{g}})_3\right).$$

Since the matrices $\hat{E}_2$ and $\hat{F}_2$ have a determinant which is a non-zero number (i.e. a polynomial of degree zero), the entries of their inverses are still polynomial in ∂_x. Thus, applying $\hat{E}_2^{-1}$ to the right hand side $\hat{\boldsymbol{g}}$ amounts to taking derivatives of $\hat{\boldsymbol{g}}$ and making linear combinations of them. If the plane $\mathbb{R}$ is split into subdomains $\mathbb{R}^- \times \mathbb{R}$ and $\mathbb{R}^+ \times \mathbb{R}$ the application of $\hat{E}_2^{-1}$ and $\hat{F}_2^{-1}$ to a vector can be done for each subdomain independently. No communication between the subdomains is necessary.

At first glance, it is surprising that the two-dimensional Stokes equations can be mainly characterized by the scalar fourth order differential operator $\Delta(-\nu\Delta + c)$. But one should note that the stream function formulation gives the same differential equation for the stream function in the two-dimensional case (cf. [GR86]).

Our goal is to write for the Stokes equations on the whole plane divided into two half-planes an algorithm converging in two iterations. We have shown that the design of an algorithm for the fourth order operator $\mathcal{B} := \Delta\mathcal{L}_2 = \Delta(-\nu\Delta + c)$ is a key ingredient for this task. Therefore, we derive an algorithm for the operator $\mathcal{B}$ and then, via the Smith factorization, we recast it in a new algorithm for the Stokes system.

We consider the following problem: Find $\phi : \mathbb{R}^2 \to \mathbb{R}$ such that

$$\mathcal{B}(\phi) = g \text{ in } \mathbb{R}^2, \qquad |\phi(\boldsymbol{x})| \to 0 \text{ for } |\boldsymbol{x}| \to \infty \tag{1.42}$$

where g is a given right hand side. The domain Ω is decomposed into two half planes $\Omega_1 = \mathbb{R}^- \times \mathbb{R}$ and $\Omega_2 = \mathbb{R}^+ \times \mathbb{R}$. Let the interface $\{0\} \times \mathbb{R}$ be denoted by Γ and $(\boldsymbol{n}_i)_{i=1,2}$ be the outward normal of $(\Omega_i)_{i=1,2}$. The algorithm, we propose, is given as follows:

ALGORITHM 4 *We choose the initial values ϕ_1^0 and ϕ_2^0 such that $\phi_1^0 = \phi_2^0$ and $\mathcal{L}_2\phi_1^0 = \mathcal{L}_2\phi_2^0$ on Γ. We obtain $(\phi_i^{n+1})_{i=1,2}$ from $(\phi_i^n)_{i=1,2}$ by the following iterative procedure:*
Correction step. *We compute the corrections $(\tilde{\phi}_i^{n+1})_{i=1,2}$ as solutions of the homogeneous local problems*

$$\left\{\begin{array}{lcl} \mathcal{B}\tilde{\phi}_1^{n+1} & = & 0 \text{ in } \Omega_1 \\ \lim\limits_{|\boldsymbol{x}|\to\infty} |\tilde{\phi}_1^{n+1}| & = & 0 \\ \dfrac{\partial\tilde{\phi}_1^{n+1}}{\partial\boldsymbol{n}_1} & = & \gamma_1^n \text{ on } \Gamma \\ \dfrac{\partial\mathcal{L}_2\phi_1^{n+1}}{\partial\boldsymbol{n}_1} & = & \gamma_2^n \text{ on } \Gamma \end{array}\right. \quad \left\{\begin{array}{lcl} \mathcal{B}\tilde{\phi}_2^{n+1} & = & 0 \text{ in } \Omega_2 \\ \lim\limits_{|\boldsymbol{x}|\to\infty} |\tilde{\phi}_2^{n+1}| & = & 0 \\ \dfrac{\partial\tilde{\phi}_2^{n+1}}{\partial\boldsymbol{n}_2} & = & \gamma_1^n \text{ on } \Gamma \\ \dfrac{\partial\mathcal{L}_2\phi_2^{n+1}}{\partial\boldsymbol{n}_2} & = & \gamma_2^n \text{ on } \Gamma \end{array}\right. \tag{1.43}$$

where $\gamma_1^n = -\dfrac{1}{2}\left(\dfrac{\partial\phi_1^n}{\partial\boldsymbol{n}_1} + \dfrac{\partial\phi_2^n}{\partial\boldsymbol{n}_2}\right)$ *and* $\gamma_2^n = -\dfrac{1}{2}\left(\dfrac{\partial\mathcal{L}_2\phi_1^n}{\partial\boldsymbol{n}_1} + \dfrac{\partial\mathcal{L}_2\phi_2^n}{\partial\boldsymbol{n}_2}\right)$.
Updating step. *We update $(\phi_i^{n+1})_{i=1,2}$ by solving the local problems*

$$\left\{\begin{array}{lcl} \mathcal{B}\phi_1^{n+1} & = & g \text{ in } \Omega_1 \\ \lim\limits_{|\boldsymbol{x}|\to\infty} |\phi_1^{n+1}| & = & 0 \\ \phi_1^{n+1} & = & \phi_1^n + \delta_1^{n+1} \text{ on } \Gamma \\ \mathcal{L}_2\phi_1^{n+1} & = & \mathcal{L}_2\phi_1^n + \delta_2^{n+1} \text{ on } \Gamma \end{array}\right. \quad \left\{\begin{array}{lcl} \mathcal{B}\phi_2^{n+1} & = & g \text{ in } \Omega_2 \\ \lim\limits_{|\boldsymbol{x}|\to\infty} |\phi_2^{n+1}| & = & 0 \\ \phi_2^{n+1} & = & \phi_2^n + \delta_1^{n+1} \text{ on } \Gamma \\ \mathcal{L}_2\phi_2^{n+1} & = & \mathcal{L}_2\phi_2^n + \delta_2^{n+1} \text{ on } \Gamma \end{array}\right. \tag{1.44}$$

where $\delta_1^{n+1} = \dfrac{1}{2}(\tilde{\phi}_1^{n+1} + \tilde{\phi}_2^{n+1})$ *and* $\delta_2^{n+1} = \dfrac{1}{2}(\mathcal{L}_2\tilde{\phi}_1^{n+1} + \mathcal{L}_2\tilde{\phi}_2^{n+1})$.

This algorithm has the proposed remarkable property. Formally we can show:

Proposition 4 *Algorithm 4 converges in two iterations.*

The proof can be found in [19].
After having found an optimal algorithm which converges in two steps for the fourth order operator $\mathcal{B}$ problem, we focus on the Stokes system (1.37)-(1.38) by translating this algorithm

into an algorithm for the Stokes system. It suffices to replace the operator $\mathcal{B}$ by the Stokes system and ϕ by the last component $(F_2(\boldsymbol{u},p)^T)_3$ of the vector $F_2(\boldsymbol{u},p)^T$ in the boundary conditions, by using formula (1.41). The algorithm reads:

ALGORITHM 5 *We choose the initial values* $(\boldsymbol{u}_1^0,p_1^0)$ *and* $(\boldsymbol{u}_2^0,p_2^0)$ *such that* $(F_2(\boldsymbol{u}_1^0,p_1^0)^T)_3 = (F_2(\boldsymbol{u}_2^0,p_2^0)^T)_3$ *and* $\mathcal{L}_2(F_2(\boldsymbol{u}_1^0,p_1^0)^T)_3 = \mathcal{L}_2(F_2(\boldsymbol{u}_2^0,p_2^0)^T)_3$ *on* Γ. *We compute* $((\boldsymbol{u}_i^{n+1},p_i^{n+1}))_{i=1,2}$ *from* $((\boldsymbol{u}_i^n,p_i^n))_{i=1,2}$ *by the following iterative procedure:*
Correction step. *We compute the corrections* $((\tilde{\boldsymbol{u}}_i^{n+1},\tilde{p}_i^{n+1}))_{i=1,2}$ *as solution of the homogeneous local problems:*

$$\left\{\begin{array}{lcl} \mathcal{S}_2(\tilde{\boldsymbol{u}}_1^{n+1},\tilde{p}_1^{n+1}) & = & 0 \ in\ \Omega_1 \\ \lim\limits_{|\boldsymbol{x}|\to\infty} |\tilde{\boldsymbol{u}}_1^{n+1}| & = & 0 \\ \dfrac{\partial (F_2(\tilde{\boldsymbol{u}}_1^{n+1},\tilde{p}_1^{n+1})^T)_3}{\partial \boldsymbol{n}_1} & = & \gamma_1^n \ on\ \Gamma \\ \dfrac{\partial \mathcal{L}_2(F_2(\tilde{\boldsymbol{u}}_1^{n+1},\tilde{p}_1^{n+1})^T)_3}{\partial \boldsymbol{n}_1} & = & \gamma_2^n \ on\ \Gamma \end{array}\right. \quad \left\{\begin{array}{lcl} \mathcal{S}_2(\tilde{\boldsymbol{u}}_2^{n+1},\tilde{p}_2^{n+1}) & = & 0 \ in\ \Omega_2 \\ \lim\limits_{|\boldsymbol{x}|\to\infty} |\tilde{\boldsymbol{u}}_2^{n+1}| & = & 0 \\ \dfrac{\partial (F_2(\tilde{\boldsymbol{u}}_2^{n+1},\tilde{p}_2^{n+1})^T)_3}{\partial \boldsymbol{n}_2} & = & \gamma_1^n \ on\ \Gamma \\ \dfrac{\partial \mathcal{L}_2(F_2(\tilde{\boldsymbol{u}}_2^{n+1},\tilde{p}_2^{n+1})^T)_3}{\partial \boldsymbol{n}_2} & = & \gamma_2^n \ on\ \Gamma \end{array}\right. \tag{1.45}$$

where

$$\gamma_1^n = -\frac{1}{2}\left(\frac{\partial (F_2(\boldsymbol{u}_1^n,p_1^n)^T)_3}{\partial \boldsymbol{n}_1} + \frac{\partial (F_2(\boldsymbol{u}_2^n,p_2^n)^T)_3}{\partial \boldsymbol{n}_2}\right)$$
$$\gamma_2^n = -\frac{1}{2}\left(\frac{\partial \mathcal{L}_2(F_2(\boldsymbol{u}_1^n,p_1^n)^T)_3}{\partial \boldsymbol{n}_1} + \frac{\partial \mathcal{L}_2(F_2(\boldsymbol{u}_2^n,p_2^n)^T)_3}{\partial \boldsymbol{n}_2}\right).$$

Updating step. *We update* $((\boldsymbol{u}_i^{n+1},p_i^{n+1}))_{i=1,2}$ *by solving the local problems:*

$$\left\{\begin{array}{lcl} \mathcal{S}_2(\boldsymbol{u}_i^{n+1},p_i^{n+1}) & = & \boldsymbol{g} \quad in\ \Omega_i \\ \lim\limits_{|\boldsymbol{x}|\to\infty} |\boldsymbol{u}_i^{n+1}| & = & 0 \\ (F_2(\boldsymbol{u}_i^{n+1},p_i^{n+1})^T)_3 & = & (F_2(\boldsymbol{u}_i^n,p_i^n)^T)_3 + \delta_1^{n+1} \ on\ \Gamma \\ \mathcal{L}_2(F_2(\boldsymbol{u}_i^{n+1},p_i^{n+1})^T)_3 & = & \mathcal{L}_2(F_2(\boldsymbol{u}_i^n,p_i^n)^T)_3 + \delta_2^{n+1} \ on\ \Gamma \end{array}\right. \tag{1.46}$$

where

$$\delta_1^{n+1} = \frac{1}{2}[(F_2(\tilde{\boldsymbol{u}}_1^{n+1},\tilde{p}_1^{n+1})^T)_3 + (F_2(\tilde{\boldsymbol{u}}_2^{n+1},\tilde{p}_2^{n+1})^T)_3],$$
$$\delta_2^{n+1} = \frac{1}{2}[\mathcal{L}_2(F_2(\tilde{\boldsymbol{u}}_1^{n+1},\tilde{p}_1^{n+1})^T)_3 + \mathcal{L}_2(F_2(\tilde{\boldsymbol{u}}_2^{n+1},\tilde{p}_2^{n+1})^T)_3].$$

This algorithm seems quite complex since it involves third order derivatives of the unknowns in the boundary conditions on $(F_2(\tilde{\boldsymbol{u}}_i,\tilde{p}_i)^T)_3$. Writing $\boldsymbol{u}_i = (u_i,v_i)$ and using $(F_2(\tilde{\boldsymbol{u}}_i,\tilde{p}_i)^T)_3 = \tilde{v}_i$, it is possible to simplify it. By using the Stokes equations in the subdomains, we can lower the degree of the derivatives in the boundary conditions. In order to ease the presentation in algorithm 6 we do not mention that the solutions tend to zero as $|\boldsymbol{x}| \to \infty$. If we denote the k-th component of the unit outward normal vector $\boldsymbol{n}_i$ of Ω_i by $n_{i,k}$, we obtain for two subdomains the following:

ALGORITHM 6 *We choose the initial values* (u_1^0, v_1^0, p_1^0) *and* (u_2^0, v_2^0, p_2^0) *such that* $v_1^0 = v_2^0$ *and*

$$\nu \frac{\partial u_1^0}{\partial \boldsymbol{n}_1} - p_1^0 n_{1,1} = -\left(\nu \frac{\partial u_2^0}{\partial \boldsymbol{n}_2} - p_2^0 n_{2,1}\right)$$

on Γ. *We compute* $((u_i^{n+1}, v_i^{n+1}, p_i^{n+1}))_{i=1,2}$ *from* $((u_i^n, v_i^n, p_i^n))_{i=1,2}$ *by the following iterative procedure:*

Correction step. *We compute the corrections* $((\tilde{u}_i^{n+1}, \tilde{v}_i^{n+1}, \tilde{p}_i^{n+1}))_{i=1,2}$ *as solution of the homogeneous local problems:*

$$\left\{\begin{array}{lcl} \mathcal{S}_2(\tilde{u}_1^{n+1}, \tilde{v}_1^{n+1}, \tilde{p}_1^{n+1}) & = & 0 \text{ in } \Omega_1 \\ \nu \dfrac{\partial \tilde{v}_1^{n+1}}{\partial \boldsymbol{n}_1} & = & \gamma_1^n \text{ on } \Gamma \\ \tilde{u}_1^{n+1} & = & \gamma_{2,1}^n \text{ on } \Gamma \end{array}\right. \quad \left\{\begin{array}{lcl} \mathcal{S}_2(\tilde{u}_2^{n+1}, \tilde{v}_2^{n+1}, \tilde{p}_2^{n+1}) & = & 0 \text{ in } \Omega_2 \\ \nu \dfrac{\partial \tilde{v}_2^{n+1}}{\partial \boldsymbol{n}_2} & = & \gamma_1^n \text{ on } \Gamma \\ \tilde{u}_2^{n+1} & = & \gamma_{2,2}^n \text{ on } \Gamma \end{array}\right. \tag{1.47}$$

where $\gamma_1^n = -\dfrac{1}{2}\left(\nu \dfrac{\partial v_1^n}{\partial \boldsymbol{n}_1} + \nu \dfrac{\partial v_2^n}{\partial \boldsymbol{n}_2}\right)$ *and* $\gamma_{2,i}^n = (-1)^i \frac{1}{2}(u_1^n - u_2^n)$.

Updating step. *We update* $((u_i^{n+1}, v_i^{n+1}, p_i^{n+1}))_{i=1,2}$ *by solving the local problems:*

$$\left\{\begin{array}{lcl} \mathcal{S}_2(u_i^{n+1}, v_i^{n+1}, p_i^{n+1}) & = & \boldsymbol{g} \text{ in } \Omega_i \\ \nu \dfrac{\partial u_i^{n+1}}{\partial \boldsymbol{n}_i} - p_i^{n+1} n_{i,1} & = & \nu \dfrac{\partial u_i^n}{\partial \boldsymbol{n}_i} - p_i^n n_{i,1} + \delta_{ij}^{n+1} \text{ on } \Gamma \\ v_i^{n+1} & = & v_i^n + \frac{1}{2}(\tilde{v}_1^n + \tilde{v}_2^n) \text{ on } \Gamma \end{array}\right. \tag{1.48}$$

where $\delta_{ij}^{n+1} = \dfrac{1}{2}\left(\nu \dfrac{\partial \tilde{u}_i^{n+1}}{\partial \boldsymbol{n}_i} - \tilde{p}_i^{n+1} n_{i,1}\right) - \dfrac{1}{2}\left(\nu \dfrac{\partial \tilde{u}_j^{n+1}}{\partial \boldsymbol{n}_j} - \tilde{p}_j^{n+1} n_{j,1}\right)$ *and* $j = 3 - i$.

Lemma 1 *Consider the model case* $\Omega = \mathbb{R}^2$, $\Omega_1 = \mathbb{R}^- \times \mathbb{R}$ *and* $\Omega_2 = \mathbb{R}^+ \times \mathbb{R}$. *We assume that all variables vanish at infinity. Then, the algorithms 5 and 6 are equivalent.*

The proof can be found in [19].

In order to write the resulting algorithm in an intrinsic form, we introduce the stress $\boldsymbol{\sigma}^i(\boldsymbol{u}, p)$ for each subdomain Ω_i on the interface for a velocity $\boldsymbol{u} = (u, v)$, a pressure p and the normal vector $\boldsymbol{n}_i$. If $n_i = \partial_x$, we have the following formula in cartesian coordinates:

$$\boldsymbol{\sigma}^i(\boldsymbol{u}, p) = (\nu \frac{\partial u}{\partial x} - p, \frac{\nu}{2}(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}))$$

For any vector $\boldsymbol{u}$ its normal (resp. tangential) component on the interface is $u_{\boldsymbol{n}_i} = \boldsymbol{u} \cdot \boldsymbol{n_i}$ (resp. $\boldsymbol{u}_{\boldsymbol{\tau}_i} = (I - \boldsymbol{n}_i \otimes \boldsymbol{n}_i)\,\boldsymbol{u}$). We denote $\sigma_{\boldsymbol{n}_i}^i := \sigma_{\boldsymbol{n}_i}^i(\boldsymbol{u}_i, p_i) \cdot \boldsymbol{n}_i$ and $\boldsymbol{\sigma}_{\boldsymbol{\tau}_i}^i := (I - \boldsymbol{n}_i \otimes \boldsymbol{n}_i)\,\boldsymbol{\sigma}^i$ as the normal and tangential parts of $\boldsymbol{\sigma}^i$, respectively.

We can now generalize the previous algorithm to a more general decomposition into non overlapping subdomains: $\overline{\Omega} = \cup_{i=1}^N \overline{\Omega}_i$ and denote by $\Gamma_{ij} = \partial\Omega_i \cap \partial\Omega_j$ the interface between subdomains Ω_i and Ω_j, $i \neq j$. The new algorithm for the Stokes system reads:

ALGORITHM 7 *Starting with an initial guess* $((\boldsymbol{u}_i^0, p_i^0))_{i=0}^N$ *satisfying* $\boldsymbol{u}_{i,\boldsymbol{\tau}_i}^0 = \boldsymbol{u}_{j,\boldsymbol{\tau}_j}^0$ *and* $\sigma_{\boldsymbol{n}_i}^i(\boldsymbol{u}_i^0, p_i^0) = \sigma_{\boldsymbol{n}_j}^j(\boldsymbol{u}_j^0, p_j^0)$ *on* Γ_{ij}, $\forall i, j$, $i \neq j$, *the* **correction step** *is expressed as follows for* $1 \leq i \leq N$*:*

$$\left\{\begin{array}{lcl} \mathcal{S}_2(\tilde{\boldsymbol{u}}_i^{n+1}, \tilde{p}_i^{n+1}) & = & 0 \quad \text{in } \Omega_i \\ \tilde{u}_{i,\boldsymbol{n}_i}^{n+1} & = & -\dfrac{1}{2}(u_{i,\boldsymbol{n}_i}^n + u_{j,\boldsymbol{n}_j}^n) \quad \text{on } \Gamma_{ij} \\ \boldsymbol{\sigma}_{\boldsymbol{\tau}_i}^i(\tilde{\boldsymbol{u}}_i^{n+1}, \tilde{p}_i^{n+1}) & = & -\dfrac{1}{2}(\boldsymbol{\sigma}_{\boldsymbol{\tau}_i}^i(\boldsymbol{u}_i^n, \tilde{p}_i^n) + \boldsymbol{\sigma}_{\boldsymbol{\tau}_j}^j(\boldsymbol{u}_j^n, \tilde{p}_j^n)) \quad \text{on } \Gamma_{ij} \end{array}\right. \tag{1.49}$$

followed by an **updating step** *for* $1 \leq i \leq N$*:*

$$\left\{\begin{array}{lcl} \mathcal{S}_2(\boldsymbol{u}_i^{n+1}, p_i^{n+1}) & = & \boldsymbol{g} \quad \text{in } \Omega_i \\ \boldsymbol{u}_{i,\boldsymbol{\tau}_i}^{n+1} & = & \boldsymbol{u}_{i,\boldsymbol{\tau}_i}^n + \dfrac{1}{2}(\tilde{\boldsymbol{u}}_{i,\boldsymbol{\tau}_i}^{n+1} + \tilde{\boldsymbol{u}}_{j,\boldsymbol{\tau}_j}^{n+1}) \quad \text{on } \Gamma_{ij} \\ \sigma_{\boldsymbol{n}_i}^i(\boldsymbol{u}_i^{n+1}, p_i^{n+1}) & = & \sigma_{\boldsymbol{n}_i}^i(\boldsymbol{u}_i^n, p_i^n) \\ & + & \dfrac{1}{2}(\sigma_{\boldsymbol{n}_i}^i(\tilde{\boldsymbol{u}}_i^{n+1}, \tilde{p}_i^{n+1}) + \sigma_{\boldsymbol{n}_j}^j(\tilde{\boldsymbol{u}}_j^{n+1}, \tilde{p}_j^{n+1})) \text{ on } \Gamma_{ij}. \end{array}\right. \tag{1.50}$$

The boundary conditions in the correction step involve the normal velocity and the tangential stress, whereas in the updating step the tangential velocity and the normal stress are involved. We can prove that in three dimensions the algorithm has the same definition.

Proposition 5 *For a domain* $\Omega = \mathbb{R}^2$ *divided into two non overlapping half planes, algorithms 5 and 7 are equivalent and both converge in two iterations.*

The above ideas have been extended to the three-dimensional Stokes equations and to the Oseen equations. We have shown in [19] that one can build, at least formally, using Smith factorization, optimal preconditioners for Stokes and Oseen equations. Nevertheless we have tested them numerically only on the two-dimensional Stokes equations. In order to do this we used a finite volume discretization on staggered grids. The numerical implementation of interface conditions is not straightforward since not all quantities involved are defined at the boundaries between domains.

We analyzed the performance of the new algorithm in the two-dimensional case. It will be compared with the standard Schur complement approach using a Neumann-Neumann preconditioner (without coarse space), cf. [TP97]. We will extend the preliminary results of [NR07], where we made some numerical experiments for the two subdomain case, using standard inf-sup stable P_2/P_1-Taylor-Hood elements on triangles. We consider the domain $\Omega = [0.2, 1.2] \times [0.1, 1.1]$ decomposed into two or more subdomains of equal or different sizes. We choose the right hand side $\boldsymbol{f}$ such that the exact solution is given by $u(x, y) = \sin(\pi x)^3 \sin(\pi y)^2 \cos(\pi y)$, $v(x, y) = -\sin(\pi x)^2 \sin(\pi y)^3 \cos(\pi x)$ and $p(x, y) = x^2 + y^2$. The viscosity ν is always 1. We solve the problem for various values of the reaction coefficient c, which can arise for example, when one applies an implicit time discretization of the unsteady Stokes problem ($c = 1/\Delta t$). The interface system is solved by GMRES. In all

tables we count the number of iterations needed to reduce the L^∞ norm of the error by the factor $TOL = 10^{-6}$:

$$\max_{i=1,\ldots,N} \|U_k^i - U_h\|_{L^\infty(\Omega_i)} \leq 10^{-6},$$

where $U_k^i = (u_k, v_k, p_k)^i$ is the discrete solution of iteration step k in subdomain Ω_i and $U_h = (u_h, v_h, p_h)$ is the global discrete solution computed by a direct solver applied to the global problem.

A problem of the new Algorithm is that in the correction step, the local matrices may be singular (the local problems are ill-posed for the pressure, the latter being defined up to an additive constant). To overcome this difficulty we chose to add a penalization term εp with ε sufficiently small to the divergence equation. This penalization term leads however still to ill-conditioned local matrices and an ill-conditioned interface problem. Thus, the reduction of the Euclidean norm of the residual is not a good indicator for the convergence of the algorithm as can be seen in Figure 1.6:

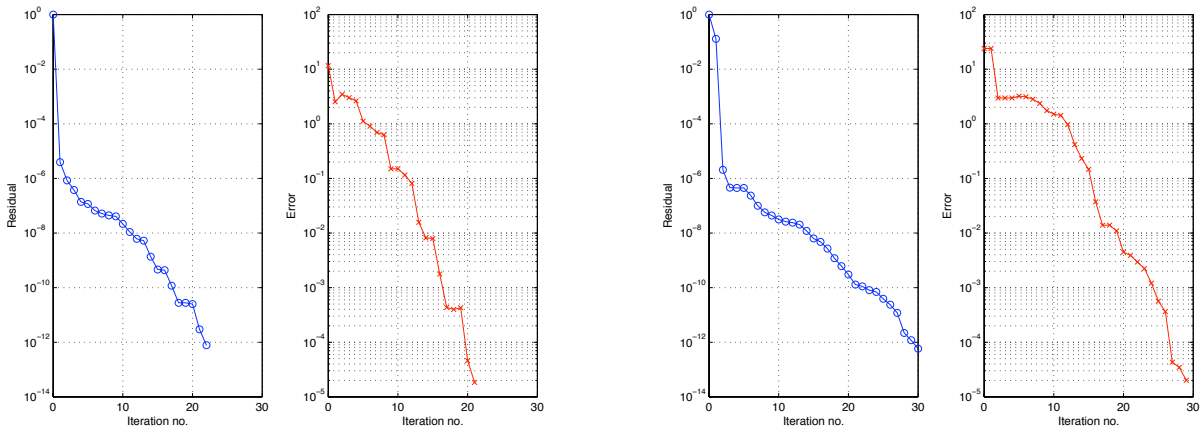


Figure 1.6: Convergence of the GMRES algorithm (residual and error) for 3×3 (left) and 4×4 (right) decompositions for $\varepsilon > 0$

This is also due to the presence of the large eigenvalues in the spectrum, as seen in the Table 1.8. A very simple way to eliminate the large eigenvalues is to avoid using the penalization term: the local problems are now singular. Consider a local matrix A which corresponds to interior subdomains in the correction (preconditioning) step. It will be singular of co-rank 1. The null space is formed by a vector whose components are constant non-zero only for the pressure components. It can be easily shown that the matrix $B + f \cdot e^t$ is invertible if we choose e (resp. f) to be a vector non-orthogonal to $ker(A)$ (resp. $ker(A^T)$). In our case it is sufficient to take (in order to preserve the sparsity of the matrix A) a vector with null components except for one non-zero component chosen in the right position. Afterward, for any right hand side b in the $Im(A)$, the solution of $Bx = b$ verifies $Ax = b$. In this case no more large eigenvalues will be present in the spectrum and the convergence of the residual

$N \times N$	No. of large eigenvalues	$N \times N$	No. of large eigenvalues
2x2	0	6x6	16
3x3	1	7x7	25
4x4	4	8x8	36
5x5	9	9x9	49

Table 1.8: Number of eigenvalues which are larger than 10 in modulus for a $N \times N$ decomposition.

will reflect more accurately the convergence of the error as one can see in Figure 1.7.

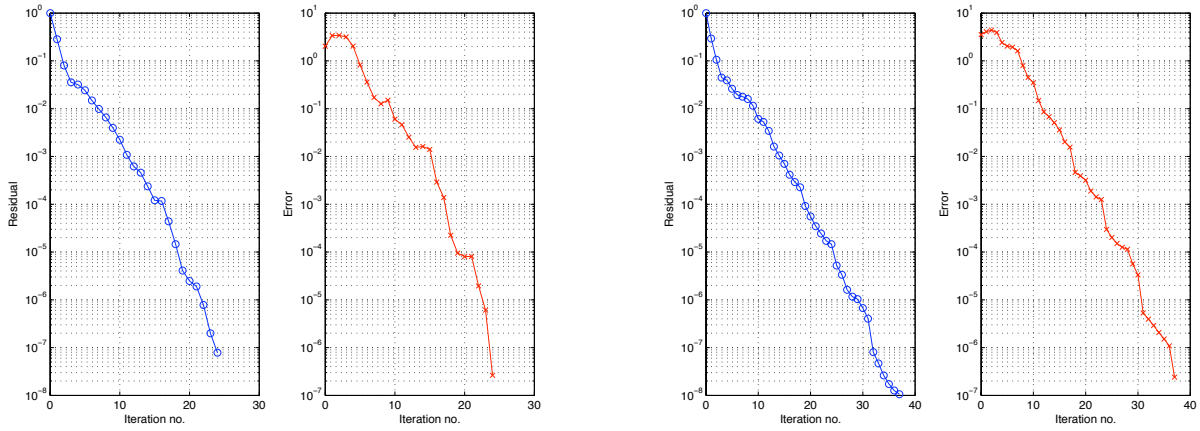


Figure 1.7: Convergence of the GMRES algorithm (residual and error) for 3×3 (left) and 4×4 (right) decompositions for $\varepsilon = 0$

Nevertheless the convergence is still very sensitive to the number of subdomains, which shows the necessity of introducing a coarse space correction in the algorithm. This bad convergence is mainly due to the presence of small eigenvalues in the spectrum of the interface operator (see Figure 1.8).

We need to eliminate the small eigenvalues which can cause bad convergence. In order to do this we will first notice that the error during the iterations of the GMRES method is mainly localized in the corners, as seen in Figure 1.9 where the error on component p is visualized.

A solution to this problem could be a deflation method applied to the preconditioning step as seen in [NV08], where the deflated vectors contain constant non-zero elements only for the corner components of the solution. As a result, we obtain a better convergence than before. It is however not optimal, since it is dependent on the number of subdomains (see Figure 1.10).

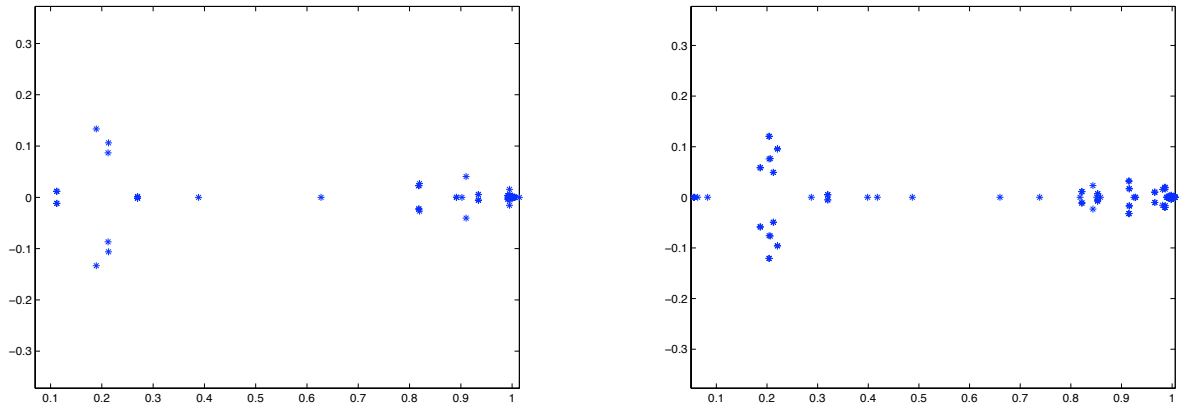


Figure 1.8: Eigenvalues of the interface preconditioned operator for 3×3 (left) and 4×4 (right) decompositions.

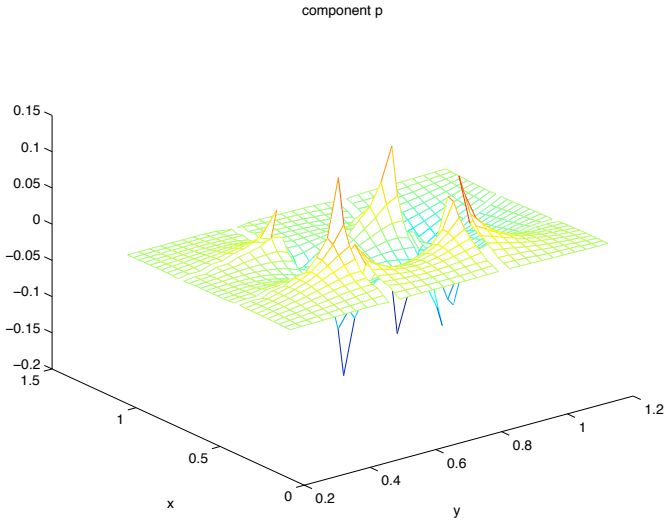


Figure 1.9: Error on the component p of the solution

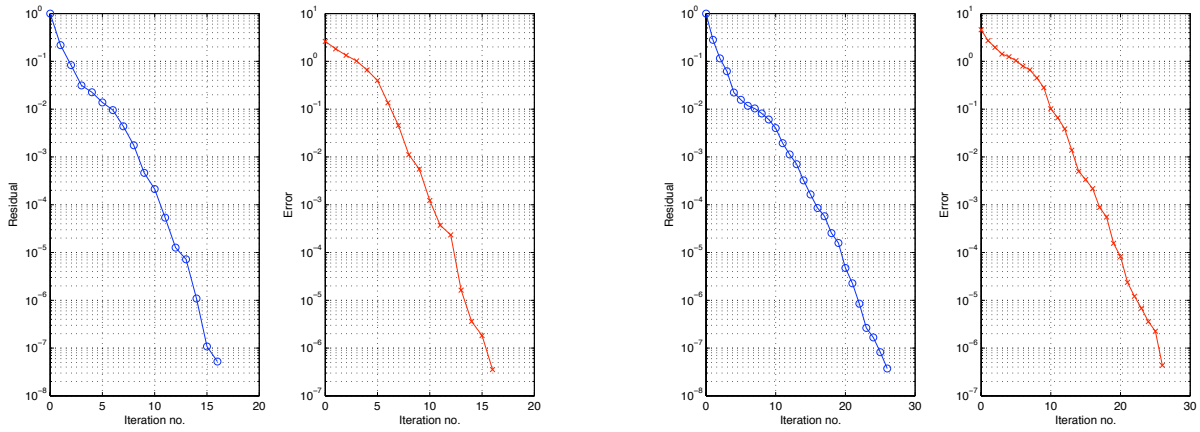


Figure 1.10: Convergence of the deflated GMRES algorithm (residual and error) for 3×3 (left) and 4×4 (right) decompositions.

By looking at the spectrum (Figure 1.11), we can see that there are still small eigenvalues that have not been taken care of by the deflation method. As a conclusion we can state that

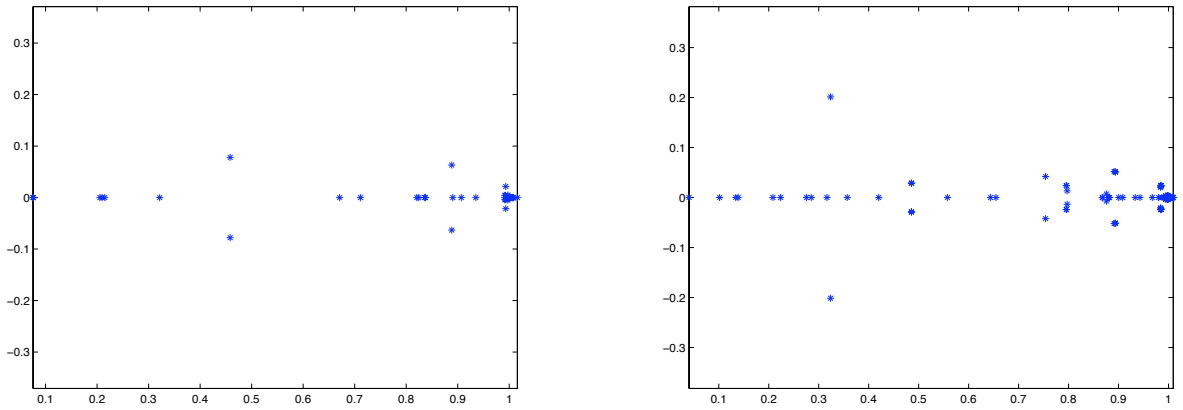


Figure 1.11: Eigenvalues of the interface preconditioned operator for 3×3 (left) and 4×4 (right) decompositions.

even if the strategy presented is not yet optimal, it leads to an improvement of the previous algorithm (since it eliminates a part of small eigenvalues) and could pave the way to the construction of a more scalable method.

The preliminary results described are very promising but they need to be finalized. We aim mainly three different directions of research: firstly, developing coarse grid preconditionners for Stokes equations (which will make the method independent of the problem size and could be thus competitive with respect of existing methods such as Neumann-Neumann or

FETI methods). Designing a coarse grid method relies mainly on heuristic reasons: combining a global data of small size that accelerate the behaviour of the method. We could use again the knowledge of such methods for scalar problems in order to get the best method possible for the Stokes system.

Secondly, we aim at designing new domain decomposition methods for Oseen equations, which could be sufficiently robust and independent of the physical parameters. To our knowledge, such a method has not yet been built since it arises both algorithmic and discretization questions that have not yet been addressed. This direction would bring a completely new and original contribution with a great potential of application especially for the fluid dynamics problems (for Navier-Stokes simulations, at each time step a Newton method is applied which implied the resolution of a linearized problem or an Oseen equation).

Thirdly, all the previous algorithms have been derived using symbolic computations, done in a heuristic way starting from an intrinsic data, and this potentially gives different forms of the same method. This process can be made completely automatic by using symbolic computation tools and software. We could know precisely how many forms of the method can be obtained and which one is more appropriate for practical disctretization reasons. Afterwards, this can also give an exact idea of how one could discretize interface conditions and what is the incidence of the discretization method on the robustness of the algorithm.

Chapter 2

Discontinuous Galerkin methods for time-harmonic and time-domain Maxwell's equations

2.1 Solution of the time-harmonic Maxwell's equations using Discontinuous Galerkin methods

This work is concerned with the numerical solution of the time-harmonic Maxwell equations discretized by discontinuous Galerkin methods on unstructured meshes. Our motivation for using a discontinuous Galerkin method is the enhanced flexibility compared to the conforming edge element method [Mon03]: for instance, dealing with non-conforming meshes is straightforward and the choice of the local approximation space is not constrained. Nonetheless, before taking full advantage of these features, it is required to carefully study the basic ingredients of the method such as the choice of the numerical flux at the interface between neighboring elements. In the context of time-harmonic problems, the design of efficient solution strategies for the resulting sparse linear systems is an equally important question.

Previous works have shown convergence results for discontinuous Galerkin methods applied to the time-harmonic Maxwell equations, studied in the form of second-order vector wave equations. Most of these works use a mixed formulation [PSM02, HPSS05b] but discontinuous Galerkin methods on the non-mixed formulation have recently been proved to converge (interior penalty technique [HPSS05a, BP05] as well as the local discontinuous Galerkin method [BP05]). The convergence properties of these methods in the time-domain case have been studied in [FLLP05] when using a centered flux and in [HW02] when using an upwind flux. The case of the upwind flux has been analyzed in [HD94] and [Hel94] for the time-harmonic problems and the convergence has been proved only for a perturbed problem. The general case of Friedrichs systems and the elliptic Maxwell equations in particular has been treated in [EG06a] and [EG06b]. However, to our knowledge, no direct convergence analysis on the first-order time-harmonic system (2.11) has been conducted so far, which should be useful, for instance, when using an upwind flux. The main contribution of this

work is a numerical study of the convergence of discontinuous Galerkin methods based on centered and upwind fluxes applied to the first-order time-harmonic Maxwell system in the two-dimensional case.

The system of non-dimensionalized time-harmonic Maxwell's equations can be written in the following form:

$$\begin{cases} \mathrm{i}\omega\varepsilon_r \boldsymbol{E} - \mathrm{curl}\ \boldsymbol{H} = -\boldsymbol{J}, \\ \mathrm{i}\omega\mu_r \boldsymbol{H} + \mathrm{curl}\ \boldsymbol{E} = 0, \end{cases} \tag{2.1}$$

where $\boldsymbol{E}$ and $\boldsymbol{H}$ are the unknown electric and magnetic fields and $\boldsymbol{J}$ is a known current source. The parameters ε_r and μ_r are respectively the complex-valued relative dielectric permittivity (integrating the electric conductivity) and the relative magnetic permeability; we consider here the case of linear isotropic media. The angular frequency of the problem is given by ω. We solve Equations (2.11) in a bounded domain Ω, and on its boundary $\partial\Omega = \Gamma_a \cup \Gamma_m$, we impose the following boundary conditions:

$$\begin{aligned} &\text{- a perfect electric conductor condition on } \Gamma_m, \ \textit{ie}\text{: } \boldsymbol{n} \times \boldsymbol{E} = 0 \text{ on } \Gamma_m, \\ &\text{- a Silver-Müller (first-order absorbing boundary) condition on } \Gamma^a, \ \textit{ie}\text{:} \\ &\boldsymbol{n} \times \boldsymbol{E} + \boldsymbol{n} \times (\boldsymbol{n} \times \boldsymbol{H}) = \boldsymbol{n} \times \boldsymbol{E}^{\mathrm{inc}} + \boldsymbol{n} \times (\boldsymbol{n} \times \boldsymbol{H}^{\mathrm{inc}}) \text{ on } \Gamma_a. \end{aligned} \tag{2.2}$$

The vectors $\boldsymbol{E}^{\mathrm{inc}}$ and $\boldsymbol{H}^{\mathrm{inc}}$ represent the components of an incident electromagnetic wave. We can further rewrite (2.11)+(2.12), assuming $\boldsymbol{J}$ equals to 0, under the following form:

$$\begin{cases} \mathrm{i}\omega G_0 \boldsymbol{W} + G_x \partial_x \boldsymbol{W} + G_y \partial_y \boldsymbol{W} + G_z \partial_z \boldsymbol{W} = 0 \text{ in } \Omega, \\ (M_{\Gamma_m} - G_{\boldsymbol{n}})\boldsymbol{W} = 0 \text{ on } \Gamma_m, \\ (M_{\Gamma_a} - G_{\boldsymbol{n}})(\boldsymbol{W} - \boldsymbol{W}_{\mathrm{inc}}) = 0 \text{ on } \Gamma_a. \end{cases} \tag{2.3}$$

where $\boldsymbol{W} = \begin{pmatrix} \boldsymbol{E} \\ \boldsymbol{H} \end{pmatrix}$ is the new unknown vector and $G_0 = \begin{pmatrix} \varepsilon_r I_3 & 0_{3\times3} \\ 0_{3\times3} & \mu_r I_3 \end{pmatrix}$. Denoting by $(\boldsymbol{e}^x, \boldsymbol{e}^y, \boldsymbol{e}^z)$ the canonical basis of $\mathbb{R}^3$, the matrices G_l with $l \in \{x, y, z\}$ are given by:

$$G_l = \begin{pmatrix} 0_{3\times3} & N_{\boldsymbol{e}^l} \\ N^t_{\boldsymbol{e}^l} & 0_{3\times3} \end{pmatrix} \text{ where for a vector } \boldsymbol{n},\ N_{\boldsymbol{n}} = \begin{pmatrix} 0 & \boldsymbol{n}_z & -\boldsymbol{n}_y \\ -\boldsymbol{n}_z & 0 & \boldsymbol{n}_x \\ \boldsymbol{n}_y & -\boldsymbol{n}_x & 0 \end{pmatrix}.$$

In the following we denote by $G_{\boldsymbol{n}}$ the sum $G_x\boldsymbol{n}_x + G_y\boldsymbol{n}_y + G_z\boldsymbol{n}_z$ and by $G^+_{\boldsymbol{n}}$ and $G^-_{\boldsymbol{n}}$ its positive and negative parts[1]. We also define $|G_{\boldsymbol{n}}| = G^+_{\boldsymbol{n}} - G^-_{\boldsymbol{n}}$. In order to take into account the boundary conditions, the matrices M_{Γ_m} and M_{Γ_a} are given by:

$$M_{\Gamma_m} = \begin{pmatrix} 0_{3\times3} & N_{\boldsymbol{n}} \\ -N^t_{\boldsymbol{n}} & 0_{3\times3} \end{pmatrix} \text{ and } M_{\Gamma_a} = |G_{\boldsymbol{n}}|.$$

[1]If $G_{\boldsymbol{n}} = T\Lambda T^{-1}$ is the eigenfactorization then $G^{\pm}_{\boldsymbol{n}} = T\Lambda^{\pm}T^{-1}$ where Λ^+ (resp. Λ^-) only gathers the positive (resp. negative) eigenvalues.

See [23] for further details on the derivation of this formulation.

Let Ω_h denote a discretization of the domain Ω into a union of conforming elements (tetrahedral or hexahedral elements)

$$\overline{\Omega}_h = \bigcup_{K \in \mathcal{T}_h} K.$$

We look for the approximate solutions $\boldsymbol{W}_h = \begin{pmatrix} \boldsymbol{E}_h \\ \boldsymbol{H}_h \end{pmatrix}$ of (2.13) in $V_h \times V_h$ where the function space V_h is defined by:

$$V_h = \left\{ \boldsymbol{V} \in [L^2(\Omega)]^3 \ / \ \forall K \in \mathcal{T}_h, \quad \boldsymbol{V}_{|K} \in \mathcal{P}(K) \right\}. \tag{2.4}$$

The term $\mathcal{P}(K)$ denotes a space of polynomial functions on the element K. We take the scalar product of the first equation of (2.13) by a sufficiently smooth vector field $\boldsymbol{V}$ and we integrate over an element K of the mesh $\mathcal{T}_h$:

$$\int_K \mathrm{i}\omega \left(G_0 \boldsymbol{W}\right)^t \overline{\boldsymbol{V}} dx + \int_K \left(\sum_{l \in \{x,y,z\}} G_l \partial_l \boldsymbol{W} \right)^t \overline{\boldsymbol{V}} dx = 0.$$

By using Green's formula we obtain a weak formulation involving a boundary term. This term is replaced in discontinuous Galerkin methods by a function $\Phi_{\partial K}$ which is usually referred as the numerical flux (see also Ern and Guermond [EG06a, EG06b]); the aim is then to determine $\boldsymbol{W}_h$ in $V_h \times V_h$ such that:

$$\begin{aligned} \int_K \mathrm{i}\omega \left(G_0 \boldsymbol{W}_h\right)^t \overline{\boldsymbol{V}} dx - \int_K \boldsymbol{W}_h^t \left(\sum_{l \in \{x,y,z\}} G_l \partial_l \overline{\boldsymbol{V}} \right) dx + \int_{\partial K} \left(\Phi_{\partial K}(\boldsymbol{W}_h)\right)^t \overline{\boldsymbol{V}} = 0, \\ \forall \boldsymbol{V} \in V_h \times V_h. \end{aligned} \tag{2.5}$$

In order to couple the element K with its neighbors for ensuring the consistency of the discretization, this numerical flux can be defined in the following way:

$$\Phi_{\partial K}(\boldsymbol{W}_h) = \begin{cases} I_{FK} S_F [\![\boldsymbol{W}_h]\!] + I_{FK} G_{\boldsymbol{n}_F} \{ \boldsymbol{W}_h \} \text{ if } F \in \Gamma^0, \\ \dfrac{1}{2} (M_{F,K} + I_{FK} G_{\boldsymbol{n}_F}) \boldsymbol{W}_h \text{ if } F \in \Gamma^m, \\ \dfrac{1}{2} (M_{F,K} + I_{FK} G_{\boldsymbol{n}_F}) \boldsymbol{W}_h - \dfrac{1}{2} (M_{F,K} - I_{FK} G_{\boldsymbol{n}_F}) \boldsymbol{W}^{\mathrm{inc}} \text{ if } F \in \Gamma^a, \end{cases} \tag{2.6}$$

where Γ^0, Γ^a and Γ^m respectively denote the set of interior faces, the set of faces on Γ_a and the set of faces on Γ_m. I_{FK} stands for the incidence matrix between oriented faces and elements whose entries are given by:

$$I_{FK} = \begin{cases} 0 \text{ if the face } F \text{ does not belong to element } K, \\ 1 \text{ if } F \in K \text{ and their orientations match}, \\ -1 \text{ if } F \in K \text{ and their orientations do not match}. \end{cases}$$

We also define respectively the jump and the average of $\boldsymbol{V}$ on a face F shared by two elements K and $\tilde{K}$:

$$[\![\boldsymbol{V}]\!] = I_{FK}\boldsymbol{V}_K + I_{F\tilde{K}}\boldsymbol{V}_{\tilde{K}} \text{ and } \{\boldsymbol{V}\} = \frac{1}{2}(\boldsymbol{V}_K + \boldsymbol{V}_{\tilde{K}}).$$

Finally, the matrix S_F allows to penalize the jump of a field or of some components of this given field on the face F and the matrix $M_{F,K}$ to be defined later insures the asymptotic consistency with the boundary conditions of the continuous problem.

In this study, we aim at comparing the properties of three classical numerical fluxes: - **a centered flux** (see [FLLP05] for the time-domain equivalent). In this case $S_F = 0$ for all the faces F and, for the boundary faces, we use:

$$M_{F,K} = \begin{cases} I_{FK}\begin{pmatrix} 0_{3\times3} & N_{\boldsymbol{n}_F} \\ -N^t_{\boldsymbol{n}_F} & 0_{3\times3} \end{pmatrix} \text{ if } F \in \Gamma^m, \\ |G_{\boldsymbol{n}_F}| \text{ if } F \in \Gamma^a. \end{cases}$$

- **an upwind flux** (see [EG06a, Pip00]). In this case:

$$S_F = \begin{pmatrix} \alpha^E_F N_{\boldsymbol{n}} N^t_{\boldsymbol{n}} & 0_{3\times3} \\ 0_{3\times3} & \alpha^H_F N^t_{\boldsymbol{n}} N_{\boldsymbol{n}} \end{pmatrix}, \; M_{F,K} = \begin{pmatrix} \eta_F N_{\boldsymbol{n}_F} N^t_{\boldsymbol{n}_F} & I_{FK} N_{\boldsymbol{n}_F} \\ -I_{FK} N^t_{\boldsymbol{n}_F} & 0_{3\times3} \end{pmatrix} \; \forall F \in \Gamma^m,$$

with α^E_F, α^H_F and η_F equals to $1/2$ for homogeneous media. The definition of M_{FK} for F in Γ^a is identical to the centered case.
- **a partially penalized upwind flux** (local Discontinuous Galerkin method, see [CS98]). This flux is characterized by a penalization coefficient given by:

$$S_F = \tau_F h_F^{-1} \begin{pmatrix} N_{\boldsymbol{n}_F} N^t_{\boldsymbol{n}_F} & 0 \\ 0 & 0 \end{pmatrix}, \; M_{F,K} = \begin{pmatrix} \eta_F h_F^{-1} N_{\boldsymbol{n}_F} N^t_{\boldsymbol{n}_F} & I_{FK} N_{\boldsymbol{n}_F} \\ -I_{FK} N^t_{\boldsymbol{n}_F} & 0_{3\times3} \end{pmatrix} \; \forall F \in \Gamma^m.$$

The definition of M_{FK} for F in Γ^a is also identical to the centered case.

We are interested in assessing these numerical fluxes for the discretization of (2.13). Firstly, we want the best asymptotic convergence order in L^2-norm for the electric and magnetic field for a fixed polynomial order approximation on an unstructured mesh. Secondly, a minimal numerical dispersion is also needed. In the following we will focus on the first criterion. The asymptotic convergence order in L^2-norm between the exact solution $(\boldsymbol{E}, \boldsymbol{H})$ and the approximate solution $(\boldsymbol{E}_h, \boldsymbol{H}_h)$ corresponds to the largest real coefficients β and γ such that:

$$\exists C_1, C_2, h_0 > 0, \; \forall h > h_0, \; \|\boldsymbol{E} - \boldsymbol{E}_h\|_{L^2(\Omega)} \leq C_1 h^\beta \text{ and } \|\boldsymbol{H} - \boldsymbol{H}_h\|_{L^2(\Omega)} \leq C_2 h^\gamma, \qquad (2.7)$$

where h is the mesh size. We first recall in Table 2.1 below the theoretical convergence order for the elliptic Maxwell equations [EG06a, EG06b], for a sufficiently smooth solution and when the local function space $\mathcal{P}(K)$ is $[P_k(K)]^3$ *i.e.* the space of vectors whose components are polynomials of order at most k. When using the flux with a penalization of $\boldsymbol{E}$, similar convergence results are proved for the time-harmonic Maxwell equations in [BP05].

flux	centered	upwind	penalization of $\boldsymbol{E}$
field $\boldsymbol{E}$	k	$k+1/2$	$k+1$
field $\boldsymbol{H}$	k	$k+1/2$	k

Table 2.1: Theoretical convergence order for the elliptic Maxwell equations.

A few comments need to be stated concerning the convergence properties of such a scheme applied to the first-order formulation of the time-harmonic Maxwell equations. First of all, the case of the upwind flux has been analyzed in [HD94] for the perturbed Maxwell problem, that is when $i\omega$ is replaced by $\nu + i\omega$ with ν a strictly positive parameter. For a sufficiently regular solution the norm of the error behaves as $h^{p+1/2}$ where h is the mesh parameter. The case of the centered flux has been studied in [FLLP05] for the time-domain Maxwell equations and in this case the norm of the error behaves as h^p where h is the mesh parameter.

For the time-harmonic equations no convergence proofs are available so far. We can only study here the solvability of the discrete problem in the case of a perturbed problem (we replace $\mathrm{i}\omega$ by $\mathrm{i}\omega + \nu$ with $\nu > 0$) following an idea used by Helluy [Hel94] in the case of the upwind flux. In the case of the perturbed problem and assuming homogeneous boundary conditions, the formulation can be simply written as:

$$\begin{cases} \text{Find } \boldsymbol{W}_h \text{ in } V_h \times V_h \text{ such that:} \\ a(\boldsymbol{W}_h, \boldsymbol{V}) + b(\boldsymbol{W}_h, \boldsymbol{V}) = 0, \ \forall \boldsymbol{V} \in V_h \times V_h, \end{cases} \tag{2.8}$$

with, $\forall \boldsymbol{U}, \boldsymbol{V} \in V_h \times V_h$:

$$\begin{aligned} a(\boldsymbol{U}, \boldsymbol{V}) = & \int_{\Omega_h} ((\mathrm{i}\omega + \nu) G_0 \boldsymbol{U})^t \, \overline{\boldsymbol{V}} dv + \sum_{F \in \Gamma^a} \int_F \left(\frac{1}{2} |G_{\boldsymbol{n}_F}| \boldsymbol{U} \right)^t \overline{\boldsymbol{V}} ds \\ & + \sum_{F \in \Gamma^m} \int_F \left(\frac{1}{2} M_{F,K} \boldsymbol{U} \right)^t \overline{\boldsymbol{V}} ds + \sum_{F \in \Gamma^0} \int_F (S_F [\![\boldsymbol{U}]\!])^t \, [\![\overline{\boldsymbol{V}}]\!]_F ds, \end{aligned} \tag{2.9}$$

and:

$$\begin{aligned} b(\boldsymbol{U}, \boldsymbol{V}) = & \sum_{K \in \mathcal{T}_h} \int_K \left(\sum_{l \in \{x,y,z\}} G_l \partial_l(\boldsymbol{U}) \right)^t \overline{\boldsymbol{V}} dv \\ & - \sum_{F \in \Gamma^a \cup \Gamma^m} \int_F \left(\frac{1}{2} I_{FK} G_{\boldsymbol{n}_F} \boldsymbol{U} \right)^t \overline{\boldsymbol{V}} ds - \sum_{F \in \Gamma^0} \int_F (G_{\boldsymbol{n}_F} [\![\boldsymbol{U}]\!])^t \, \{ \overline{\boldsymbol{V}} \} ds. \end{aligned} \tag{2.10}$$

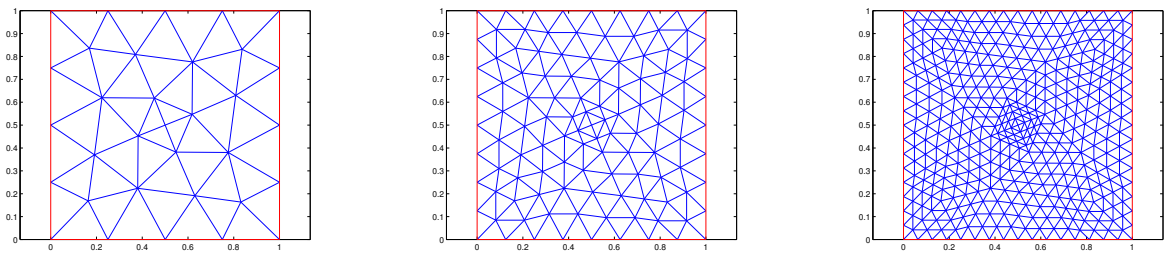
We have the following result:

Proposition 6 *The solution of problem* (2.8) *is null.*

The proof can be found in [15] .

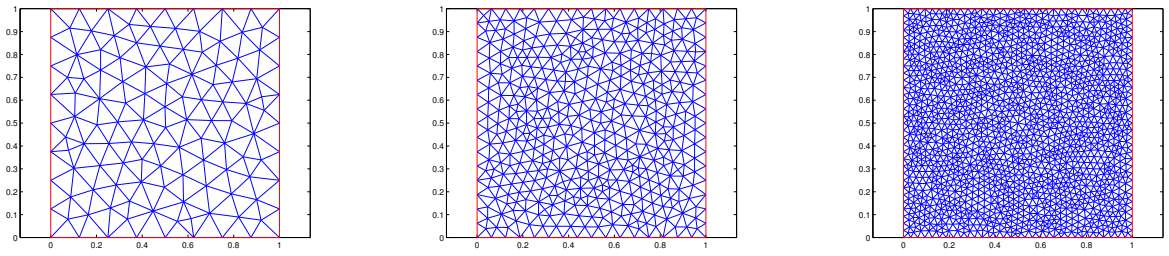
We will present a numerical comparison of different fluxes for a very simple test case and different kind of meshes. In the second part, the results on a less trivial problem are compared to those obtained with the plane wave example. We consider the case of an electric transverse wave in the plane (O, x, y). In this case the components $\boldsymbol{E}_z$, $\boldsymbol{H}_x$ and $\boldsymbol{H}_y$ are zero. We numerically simulate the propagation of a plane wave in vacuum where the incident wave is given by $(\boldsymbol{E}_x^{\text{inc}}, \boldsymbol{E}_y^{\text{inc}}, \boldsymbol{H}_z^{\text{inc}}) = \exp(-i\omega x)(0, 1, 1)$. The computational domain is the unit square $\Omega =]0; 1[^2$ and a Silver-Müller boundary condition is imposed on the whole boundary, that is $\Gamma_a = \partial\Omega$ and $\Gamma_m = \emptyset$. The parameters ε_r and μ_r are set to 1 everywhere and we choose $\omega = 2\pi$. We numerically estimate the asymptotic convergence order of discontinuous Galerkin methods for the above problem using two different sequences of triangular meshes:
- **uniformly refined meshes.** The first mesh of Figure 2.1(a) is uniformly refined resulting in the meshes of Figures 2.1(b) and 2.1(c).
- **independent meshes.** We use four unstructured (quasi-uniform) independent meshes with an imposed maximal mesh size h (see Figure 2.2 for the first three meshes). These meshes are denoted by $\mathcal{T}_i$ for $i = 1, \ldots, 4$ with h in a decreasing order. Thus $\mathcal{T}_{i+1}$ is not a refinement of $\mathcal{T}_i$.

Our implementation of high order discontinuous Galerkin methods makes use of nodal basis functions with equi-spaced nodes.



(a) Initial mesh. (b) First refinement. (c) Second refinement.

Figure 2.1: Initial mesh of the unit square and two uniform refinements.



(a) $h = 1/8$. (b) $h = 1/16$. (c) $h = 1/32$.

Figure 2.2: First three independent unstructured meshes.

Centered flux. Numerical convergence results in a logarithmic scale. They clearly demonstrate the interest of higher order polynomial approximations which allow a considerable reduction of the number of degrees of freedom to reach the same accuracy. Table 2.2 summarizes numerical estimates (using a linear regression method) of the asymptotic convergence order.

	P_0	P_1	P_2	P_3
$\boldsymbol{E}$	1.0	1.0	2.0	3.0
$\boldsymbol{H}$	1.0	2.0	3.0	3.6

Table 2.2: Numerical convergence order using a centered flux.

The method based on a P_0 approximation (*i.e.* the standard cell centered finite volume method) is special: the convergence order is optimal for both fields $\boldsymbol{E}$ and $\boldsymbol{H}$, that is, equal to $k+1$. This could be the consequence of using uniformly refined meshes, since a somewhat different behavior is obtained for independent meshes with decreasing mesh size. For the other polynomial degrees, we get exactly the predicted theoretical convergence order in the elliptic case for $\boldsymbol{E}$, whereas for $\boldsymbol{H}$, this convergence order is optimal. Therefore, in this example, the magnetic field is better approximated than the electric field, when using the centered flux.

Upwind flux. We used here the parameters $\alpha_F^H = \alpha_F^E = \eta_F = 1$ for each face F. Similar conclusions can be derived as in the centered case except that the convergence properties of the methods based on P_0 and P_1 interpolations are this time clearly different with respect to the centered case. The asymptotic convergence orders (see Table 2.3) are similar for both fields and correspond to the theory for the elliptic Maxwell equations. The convergence is optimal except for the case P_0, but nevertheless we are still above the theoretical estimates.

	P_0	P_1	P_2	P_3
$\boldsymbol{E}$	0.9	1.9	3.0	3.9
$\boldsymbol{H}$	0.9	1.9	3.0	3.9

Table 2.3: Numerical convergence order using an upwind flux.

Penalized flux on $\boldsymbol{E}$. We set $\tau_F = \eta_F = 1$ for each face F. Table 2.4 summarizes the numerical estimates of the asymptotic convergence order. Besides the expected lack of convergence in the case P_0, we can notice for all the other cases $((P_k)_{k>0})$ a complementary behavior with respect to the centered flux, since this time we get an optimal convergence rate for $\boldsymbol{E}$, but not for $\boldsymbol{H}$.

When we use independent meshes (as in Figure 2.2), the results for the upwind flux are the same as for the uniformly refined meshes. For the centered flux, note the lack of convergence for the case P_0. For all the other cases the results remain the same as for the

	P_0	P_1	P_2	P_3
$\boldsymbol{E}$	X	2.0	3.1	3.9
$\boldsymbol{H}$	X	1.0	2.0	2.9

Table 2.4: Numerical convergence order using a penalized flux on $\boldsymbol{E}$.

uniformly refined meshes.

We perform now the same analysis in the case of a less trivial problem. The domain is the square $[-1;1]^2$ where we have suppressed a part by inserting a point of coordinates $(0.1,0)$ at it is shown on Figure 2.3. The properties ε_r and μ_r are still homogeneous and equal to one. Appropriate non-homogeneous Dirichlet boundary conditions are enforced on the boundary of the domain in order to obtain $\boldsymbol{E} = (\sin(2\pi y), \sin(2\pi x))^t$ as the solution. The mesh is not fully homogeneous as it is shown on Figure 2.3; it is slightly denser next to the point of coordinates $(0.1,0)$. Independent meshes have been used as for the previous example. The same conclusions yield as in the case of the independent meshes of the square.

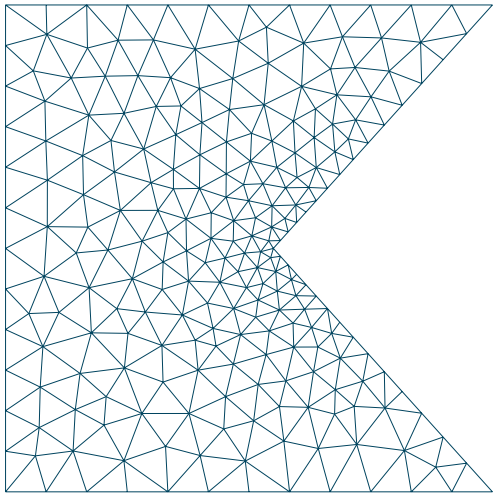


Figure 2.3: Unstructured meshes for the second example

As a final conclusion: it is already known for time-domain problems that the centered flux combined to a leap-frog time integration scheme results in a non-dissipative discontinuous Galerkin method (a mandatory feature for long time computations, see [FLLP05]). As far as time-harmonic problems are concerned, the previous results show that the upwind flux has better convergence properties. Nevertheless, the centered flux remains less expensive both for time-domain and time-harmonic problems (arithmetic operations and memory requirements).

2.2 An implicit DGTD method for two-dimensional electromagnetic wave propagation

Nowadays, a variety of methods exist for the numerical treatment of the time-domain Maxwell equations, ranging from the well established and still prominent finite difference time-domain (FDTD) methods based on Yee's scheme [Yee66] to the more recent finite element time domain (FETD) and discontinuous Galerkin time domain (DGTD) methods [HW02]-[CCR05]-[FLLP05]-[MR05]-[CFP06]. The use of unstructured meshes (based on quadrangles or triangles in two space dimensions, heaxahedra or tetrahedra in three space dimensions) is an intrinsic feature of the latter methods which can thus easily deal with complex geometries and heterogeneous propagation media. Unfortunately, local mesh refinement can translate in a very restrictive time step in order to preserve the stability of the explicit time integration schemes which are most often adopted in FETD and DGTD methods. There are basically two directions to cure this efficiency problem. The first one consists in using a local time stepping strategy combined to an explicit time integration scheme. The second approach consists in using an implicit or a hybrid explicit-implicit time integration scheme. An implicit time integration scheme is a natural way to obtain a time domain method which is unconditionally stable, but at the expense of the inversion of a global linear system at each time step.

The work is concerned with the development of a time implicit discontinuous Galerkin method for the simulation of two-dimensional time-domain electromagnetic wave propagation on non-uniform triangular meshes. The proposed method combines an arbitrary high order discontinuous Galerkin method for the discretization in space designed on triangular meshes, with a second order Crank-Nicolson scheme for time integration. At each time step, a multifrontal sparse LU method is used for solving the linear system resulting from the discretization of the TE Maxwell equations. Despite the computational overhead of the solution of a linear system at each time step, the resulting implicit discontinuous Galerkin time-domain method allows for a noticeable reduction of the computing time as compared to its explicit counterpart based on a leap-frog time integration scheme. The proposed method is useful if the underlying mesh is non-uniform or locally refined such as when dealing with complex geometric features or with heterogenous propagation media. This study is a first step towards the development of an efficient discontinuous Galerkin method for the simulation of three-dimensional time-domain electromagnetic wave propagation on non-uniform tetrahedral meshes. It yields first insights of the capabilities of implicit time stepping through a detailed numerical assessment of accuracy properties and computational performances. In the field of high frequency computational electromagnetism, the use of implicit time stepping has so far been limited to cartesian meshes in conjunction with the finite difference time-domain method (e.g. the ADI- FDTD method). This study is the first attempt to combine implicit time stepping with a discontinuous Galekin discretization method designed on simplex meshes.

The starting point of this study is given by the explicit DGTD-$\mathbb{P}_p$ method presented in [FLLP05] for solving the three-dimensional time domain Maxwell equations on unstructured tetrahedral meshes. Beside a standard discontinuous Galerkin formulation, this method is based on two basic ingredients: a centered approximation for the computation of the numerical flux at inter-element boundaries, and an explicit leap-frog time integration scheme. The implicit DGTD-$\mathbb{P}_p$ method differs from its explicit counterpart in the time integration scheme which is now chosen to be a Crank-Nicolson scheme. The resulting implicit DGTD-$\mathbb{P}_p$ method is non-dissipative and unconditionally stable as will be shown in the sequel. Although the objective of this paper is to evaluate the proposed method in the context of the numerical solution of the two-dimensional Maxwell equations, in this section we formulate and study the implicit DGTD-$\mathbb{P}_p$ method in the more general case of the three-dimensional case. The system of time-domain Maxwell equations in the absence of current sources is given by:

$$\varepsilon \partial_t \mathbf{E} - \text{curl}\,(\mathbf{H}) = 0 \quad , \quad \mu \partial_t \mathbf{H} + \text{curl}\,(\mathbf{E}) \quad = 0. \tag{2.11}$$

where $\mathbf{E}$ and $\mathbf{H}$ are the unknown electric and magnetic fields, ε and μ respectively denote the electric permittivity and the magnetic permeability (the propagation medium is assumed to be a linear and isotropic material). Our goal is to solve system (2.11) in a domain Ω of border $\partial\Omega = \Gamma_a \cup \Gamma_m$, where we impose the following boundary conditions:

$$\mathbf{n} \times \mathbf{E} = 0 \text{ on } \Gamma_m \quad , \quad \mathbf{n} \times \mathbf{E} + Z\mathbf{n} \times (\mathbf{n} \times \mathbf{H}) = 0 \text{ on } \Gamma_a. \tag{2.12}$$

where $Z = \sqrt{\frac{\mu}{\varepsilon}}$, $\mathbf{n}$ denotes the unit outward normal to $\partial\Omega$. The first boundary condition is called *metallic* (referring to a perfectly conducting surface) while the second condition is called *absorbing*, takes the form of the Silver-Müller condition which is a first order approximation of the exact absorbing boundary condition and is applied on Γ_a which represents the artificial limit of the computational domain. Problem (2.11) can be rewritten under the following form:

$$\begin{cases} G_0 \partial_t \mathbf{W} + G_x \partial_x \mathbf{W} + G_y \partial_y \mathbf{W} + G_z \partial_z \mathbf{W} = 0 \text{ in } \Omega, \\ (M_{\Gamma_m} - G_{\mathbf{n}})\mathbf{W} = 0 \text{ on } \Gamma_m, \\ (M_{\Gamma_a} - G_{\mathbf{n}})\mathbf{W} = 0 \text{ on } \Gamma_a. \end{cases} \tag{2.13}$$

Here we used the same notations as in section 2.1.

We first discretize the system (2.13) with respect to the time variable using the Crank-Nicolson scheme:

$$G_0 \left(\frac{\mathbf{W}^{n+1} - \mathbf{W}^n}{\Delta t} \right) + (G_x \partial_x + G_y \partial_y + G_z \partial_z) \left(\frac{\mathbf{W}^{n+1} + \mathbf{W}^n}{2} \right) = 0, \tag{2.14}$$

where $\mathbf{W}^n$ is the approximation of $\mathbf{W}$ at time $t_n = n\Delta t$ and Δt denotes the time step. For each t_n, we thus need to solve the following boundary value problem:

$$\begin{cases} \beta G_0 \mathbf{W} + (G_x\partial_x + G_y\partial_y + G_z\partial_z)\mathbf{W} = \mathbf{F}, \text{ in } \Omega, \\ (M_{\Gamma_m} - G_{\mathbf{n}})\mathbf{W} = 0 \text{ on } \Gamma_m, \\ (M_{\Gamma_a} - G_{\mathbf{n}})\mathbf{W} = 0 \text{ on } \Gamma_a, \end{cases} \tag{2.15}$$

where $\beta = \dfrac{2}{\Delta t}$, $\mathbf{W} = \mathbf{W}^{n+1}$ and $\mathbf{F} = \beta G_0 \mathbf{W}^n - (G_x\partial_x + G_y\partial_y + G_z\partial_z)\mathbf{W}^n$. This system can be reformulated as a symmetric Friedrichs system and according to the results stated in [EG06a], the associated boundary value problem (2.15) is well-posed in the functional space:

$$V = \{\mathbf{W} \in H(\text{curl }) \times H(\text{curl }); (M_{\Gamma_m} - G_{\mathbf{n}})\mathbf{W}|_{\Gamma_m} = 0,\ (M_{\Gamma_a} - G_{\mathbf{n}})\mathbf{W}|_{\Gamma_a} = 0\}$$

The main motivation for using a Crank-Nicolson scheme rather than, for instance, a second order, upwind in time, implicit scheme is the following result concerning the conservation property of the semi-discrete electromagnetic energy.

Lemma 2 *Let a semi-discrete electromagnetic energy be defined by:*

$$\mathcal{E}^n = \frac{1}{2}\int_\Omega (\mathbf{W}^n)^t (G_0 \mathbf{W}^n) dx = \frac{1}{2}\int_\Omega \varepsilon \parallel \mathbf{E}^n \parallel^2 dx + \frac{1}{2}\int_\Omega \mu \parallel \mathbf{H}^n \parallel^2 dx$$

then this energy is non-increasing in time, $\mathcal{E}^{n+1} \leq \mathcal{E}^n$, and it is exactly conserved in the absence of absorbing boundaries ($\Gamma_a = \emptyset$).

Now we proceed to the space discretization. Let $\mathcal{T}_h$ be a discretisation of the computational domain Ω such that $\overline{\Omega_h} \equiv \mathcal{T}_h = \bigcup_{K \in \mathcal{T}_h} \overline{K}$. In this study, the numerical approximation $\mathbf{W}_h$ of the solution of problem (2.13) lies in the space $C^1([0,T]; V_h)$ where:

$$V_h = \left\{ \mathbf{V} \in [L^2(\Omega)]^3 \times [L^2(\Omega)]^3 |\ \forall K \in \mathcal{T}_h \ \ , \ \ \mathbf{V}_{|K} \in \mathbb{P}_p(K) \right\}, \tag{2.16}$$

where $\mathbb{P}_p(K)$ denotes a space of polynomial elements of degree at most p over the element K. By taking the scalar product of (2.13) by a regular vector field $\mathbf{V}$ and integrating over an element K, we get:

$$\int_K (G_0 \partial_t \mathbf{W}_h)^t \mathbf{V} dx + \int_K \left(\sum_{l \in \{x,y,z\}} G_l \partial_l \mathbf{W}_h \right)^t \mathbf{V} dx = 0 \ \ , \ \ \forall \mathbf{V} \in V_h, \tag{2.17}$$

and integration by parts yields:

$$\int_K (G_0 \partial_t \mathbf{W}_h)^t \mathbf{V} dx - \int_K \mathbf{W}_h^t \left(\sum_{l \in \{x,y,z\}} G_l \partial_l \mathbf{V} \right) dx + \int_{\partial K} (G_{\mathbf{n}_K} \mathbf{W}_h)^t \mathbf{V} ds = 0 \ \ , \ \ \forall \mathbf{V} \in V_h, \tag{2.18}$$

where $\mathbf{n}_K$ is the unit outward normal to the border of element K. In equation (2.18) we still need to define an approximation of the boundary integral operand, called *main numerical flux* in [EG06a]-[EG06b]. Then, Eq. (2.18) becomes:

$$\int_K (G_0 \partial_t \mathbf{W}_h)^t \mathbf{V} dx - \int_K \mathbf{W}_h^t \left(\sum_{l \in \{x,y,z\}} G_l \partial_l \mathbf{V} \right) dx + \sum_{F \in \partial K} \int_F (\Phi_{\partial K}(\mathbf{W}_h))^t \mathbf{V} ds = 0 \ , \ \forall \mathbf{V} \in V_h. \tag{2.19}$$

The definition of $\Phi_{\partial K}$ needs to ensure the consistency of the approximation method and follows the formalism of flux definition in section 2.1. It depends on the type of face: we adopt here a centered scheme as in [FLLP05], for the evaluation of the flux through an internal face $F \in \Gamma^0$ and an upwind flux for the absorbing boundary faces $F \in \Gamma^a$. Using now the weak form (2.19) and by summing over all elements K of $\mathcal{T}_h$, the problem at hand is to find $\mathbf{W}_h \in V_h$ such that $\forall \mathbf{V} \in V_h$:

$$\begin{aligned} \int_{\Omega_h} (G_0 \partial_t \mathbf{W}_h)^t \mathbf{V} dx - \sum_{K \in \mathcal{T}_h} \int_K \mathbf{W}_h^t \left(\sum_{l \in \{x,y,z\}} G_l \partial_l \mathbf{V} \right) dx \\ + \sum_{F \in \mathcal{F}_m \cup \mathcal{F}_a} \int_F \left(\frac{1}{2} (M_{F,K} + I_{F,K} G_{\mathbf{n}_F}) \mathbf{W}_h \right)^t \mathbf{V} ds \\ + \sum_{F \in \mathcal{F}_0} \int_F (G_{\mathbf{n}_F} \{\mathbf{W}_h\}_F)^t [\![\mathbf{V}]\!]_F ds = 0. \end{aligned} \tag{2.20}$$

This weak formulation is inspired by [EG06a, equation (4.26)] and adapted to problem (2.13).

To analyze the totally discretized problem we introduce the following bilinear forms:

$$\begin{cases} a(\mathbf{W}, \mathbf{V}) &= 2 \displaystyle\int_{\Omega_h} (G_0 \mathbf{W})^t \mathbf{V} dx, \\ b(\mathbf{W}, \mathbf{V}) &= - \displaystyle\int_{\Omega_h} \mathbf{W}^t \left(\sum_{l \in \{x,y,z\}} G_l \partial_l \mathbf{V} \right) dx + \int_{\Omega_h} \left(\sum_{l \in \{x,y,z\}} G_l \partial_l \mathbf{W} \right)^t \mathbf{V} dx, \\ c(\mathbf{W}, \mathbf{V}) &= \displaystyle\sum_{F \in \mathcal{F}_0} \int_F \left[\left(G_{\mathbf{n}_F} \mathbf{W}_{|K} \right)^t \left(I_{F,\tilde{K}} \mathbf{V}_{|\tilde{K}} \right) + \left(G_{\mathbf{n}_F} \mathbf{W}_{|\tilde{K}} \right)^t \left(I_{F,K} \mathbf{V}_{|K} \right) \right] ds \\ &+ \displaystyle\sum_{F \in \mathcal{F}_m \cup \mathcal{F}_a} \int_F (M_{F,K} \mathbf{W})^t \mathbf{V} ds. \end{cases}$$

Thus (2.20) is equivalent to find $\mathbf{W}_h \in V_h$ such that $\forall \mathbf{V} \in V_h$:

$$a(\partial_t \mathbf{W}, \mathbf{V}) + b(\mathbf{W}, \mathbf{V}) + c(\mathbf{W}, \mathbf{V}) = 0, \ \forall \mathbf{V} \in V_h. \tag{2.21}$$

The totally discretized problem by the Crank-Nicolson scheme is then to find $\mathbf{W}_h^{n+1} \in V_h$

such that $\forall \mathbf{V} \in V_h$:

$$\begin{aligned} \beta a(\mathbf{W}_h^{n+1}, \mathbf{V}) &+ b(\mathbf{W}_h^{n+1}, \mathbf{V}) + c(\mathbf{W}_h^{n+1}, \mathbf{V}) \\ &= \beta a(\mathbf{W}_h^{n}, \mathbf{V}) - b(\mathbf{W}_h^{n}, \mathbf{V}) - c(\mathbf{W}_h^{n}, \mathbf{V}), \end{aligned} \tag{2.22}$$

where $\beta = \dfrac{2}{\Delta t}$, given $W_h^0 = W_h(0)$, where $W_h(0)$ is the discretization of the intial condition. This discrete problem has a unique solution as stated by the following result.

Lemma 3 *The homogeneous discrete problem*

$$\begin{gathered} \textit{Find } \mathbf{W}_h \in V_h \textit{ such that} \\ \beta a(\mathbf{W}_h, \mathbf{V}) + b(\mathbf{W}_h, \mathbf{V}) + c(\mathbf{W}_h, \mathbf{V}) = 0, \ \forall \mathbf{V} \in V_h, \end{gathered} \tag{2.23}$$

possesses only the trivial solution.

A direct consequence of the fact that the bilinear form $b(\mathbf{W}, \mathbf{V})$ is skew-symmetric and of property (**??**) is the following discrete counterpart of Lemma 2.

Lemma 4 *If we define the discrete energy by:*

$$\mathcal{E}_h^n = \frac{1}{4} a(\mathbf{W}_h^n, \mathbf{W}_h^n), \tag{2.24}$$

then this energy is non-increasing in time, $\mathcal{E}_h^{n+1} \leq \mathcal{E}_h^n$, and it is exactly conserved in absence of absorbing boundaries ($\Gamma_a = \emptyset$).

In summary, the totally discretized problem, which can be seen as a time discretization of a system of ordinary differential equations, is unconditionally stable and the total energy is conserved in absence of absorbing boundaries. The convergence properties of the discontinuous Galerkin discretization method using centered fluxes coupled to an explicit second order leap-frog time integration scheme, are analyzed in details in [FLLP05]. Clearly, the same steps of this convergence analysis can be applied in the present context where the explicit leap-frog scheme is replaced by the second order Crank-Nicolson time integration scheme leading to the same result for the order of the total error. This result is confirmed through numerical experiments.

As far as numerical computations are concerned, extensive numerical testing and results can be found in [21], where the method is applied to the numerical solution of the two-dimensional TE Maxwell equations:

$$\mu \frac{\partial H_x}{\partial t} + \frac{\partial E_z}{\partial y} = 0 \ , \quad \mu \frac{\partial H_y}{\partial t} - \frac{\partial E_z}{\partial x} = 0 \ , \quad \varepsilon \frac{\partial E_z}{\partial t} - \frac{\partial H_y}{\partial x} + \frac{\partial H_x}{\partial y} = 0. \tag{2.25}$$

The implicit DGTD method proposed here requires the solution of a sparse linear system at each time step however, for non-dispersive materials, the coefficient of this system are

time independent, a feature that can be taken into account to minimize the additional computational overhead. Consequently, in this study, we decided to use a LU factorization method for sparse matrices more precisely, the MUMPS multifrontal sparse matrix solver [ADL00]. The sparse matrix characterizing the implicit DGTD method has a block structure where the size of a block is $3n_p \times 3n_p$, n_p being the number of degrees of freedom associated to a nodal polynomial basis of the space $\mathbb{P}_p$ i.e $n_p = ((p+1)(p+2))/2$. This matrix is factored once for all before the time stepping loop. Then, each linear system inversion amounts to a forward and a backward solve using the triangular L and U factors.

We will limit the presentation to the scattering of a plane wave over a dielectric cylinder: a typical problem, in which a plane wave impinges on a dielectric cylinder, experiencing reflection and refraction at the material interface. The geometry of the scenario is shown in Fig. 2.4.

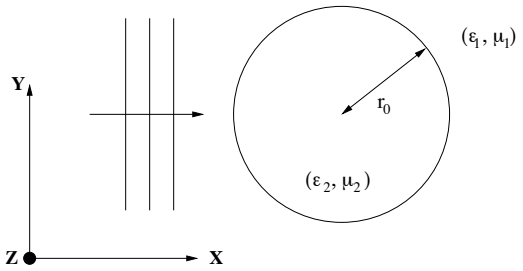


Figure 2.4: Scattering of a plane wave by a dielectric cylinder setting

We assume that the cylinder is illuminated by a monochromatic plane wave of the form:

$$E_z^{\text{inc}} = \exp(-\imath(k_1 x - \omega t)) \quad , \quad H_y^{\text{inc}} = -\exp(-\imath(k_1 x - \omega t))$$

where $k_1 = \omega\sqrt{\varepsilon_1 \mu_1}$. In the follwoing, we set $\mu_1 = \mu_2 = \varepsilon_1 = 1$, i.e. the material is non-magnetic, and the material exterior to the cylinder is assumed to be vacuum. The angular frequency is $\omega = 2\pi$ (i.e. F=300 MHz) and the computational domain Ω is chosen as a cylinder of radius one, centered at $(0, 0)$. The far-field boundary Γ_a where the first order Silver-Müller absorbing condition is applied is defined as a cylinder with radius r = 1.6 m.

We consider a situation for which the internal cylinder has a radius $r_0 = 0.6$ m and bounds a material with relative permittivity $\varepsilon_2 = 2.25$. We make use of a non-uniform mesh consisting of 4108 vertices and 8054 triangles. The ratio between the largest and smallest edges of this mesh is 197. In this case, the minimum and maximum values of the time step are respectively given by $(\Delta t)_m = 0.000627$ m and $(\Delta t)_M = 0.092891$ m (the ratio $\delta = (\Delta t)_M/(\Delta t)_m = 148$). As previously, the time step used in the simulations is CFL-$\mathbb{P}_p \times (\Delta t)_m$. Results are shown on Fig. 2.5 and 2.6 in terms of the contour lines of E_z after 10 periods and of the x-wise 1D distribution for $y = 0.0$ m of the discrete Fourier transform of E_z respectively. In addition, Fig. 2.7 shows the time evolution of the L2 error between the numerical and exact solution for the explicit and implicit DGTD-$\mathbb{P}_1$ and DGTD-$\mathbb{P}_2$ methods. For this configuration of the problem, the gain between the explicit and implicit DGTD

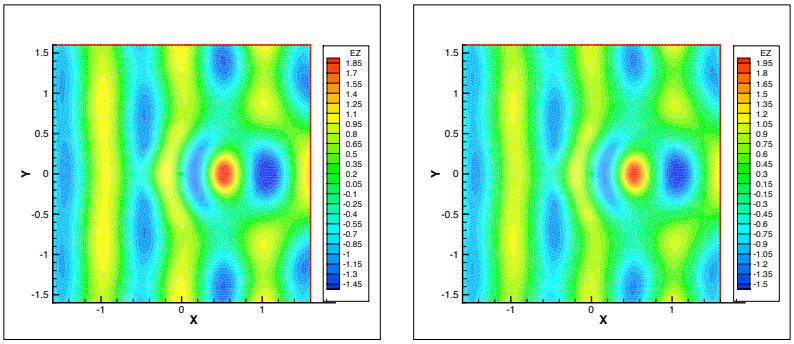


Figure 2.5: Scattering of a plane wave by a dielectric cylinder (C1)
Contour lines of E_z after 10 periods Left: analytical solution - Right: implicit DGTD-$\mathbb{P}_2$ method, CFL=20.0

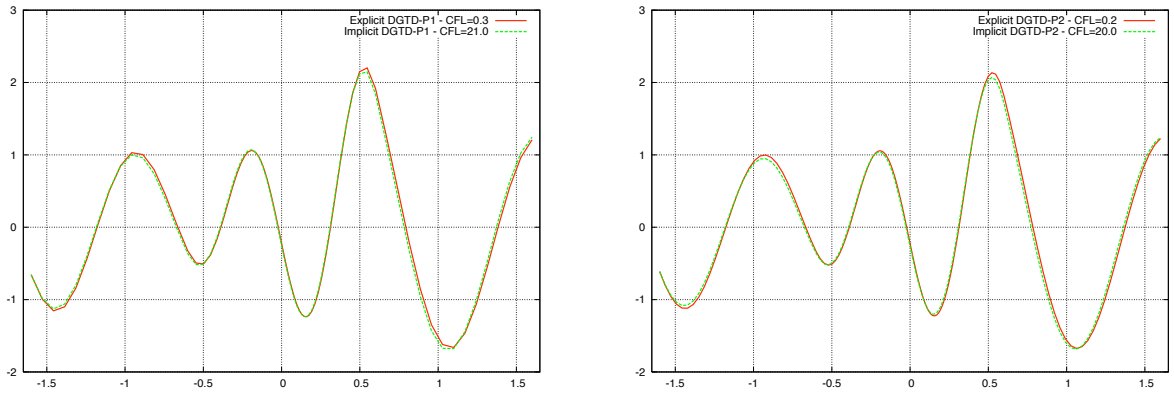


Figure 2.6: Scattering of a plane wave by a dielectric cylinder (C1): 1D distribution of DFT(E_z), $y = 0.0$ m; Left: DGTD-$\mathbb{P}_1$ method - Right: DGTD-$\mathbb{P}_2$ method

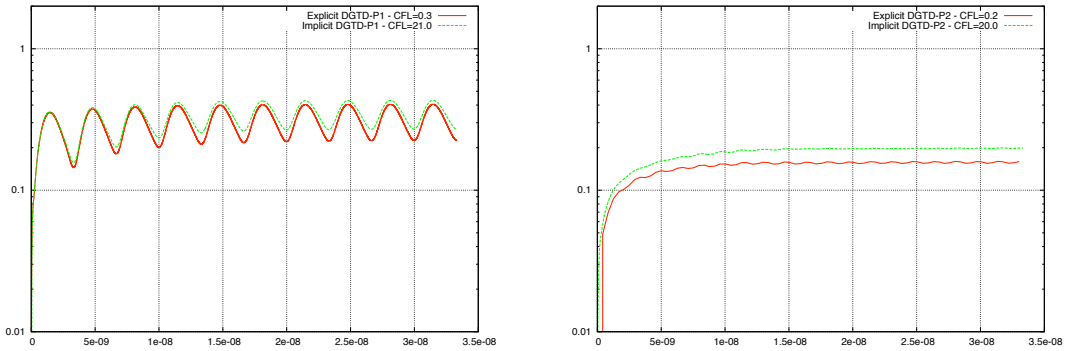


Figure 2.7: Scattering of a plane wave by a dielectric cylinder (C1): time evolution of the L2 error; Left: DGTD-$\mathbb{P}_1$ method - Right: DGTD-$\mathbb{P}_2$ method

Time integration	Method	CFL-$\mathbb{P}_p$	CPU time
Explicit	DGTD-$\mathbb{P}_1$	0.3	542 sec
Implicit	-	21.0	102 sec
Explicit	DGTD-$\mathbb{P}_2$	0.2	1892 sec
Implicit	-	20.0	218 sec

Table 2.5: Scattering of a plane wave by a dielectric cylinder (C1)
CPU times (AMD Opteron 2 GHz based workstation)

methods is equal to 5.4 for $p = 1$ and 8.7 for $p = 2$. We also note that the factorization time represents a small percentage of the simulation time for both cases of approximation order.

As a conclusion: in this work we have studied an implicit DGTD-$\mathbb{P}_p$ method for solving the time-domain Maxwell equations on unstructured triangular meshes. This method is non-dissipative, second order accurate in time and $p+1$-th order accurate in space. As usual with time implicit schemes, this method requires the solution of a sparse linear system at each time step. For non-dispersive materials, the coefficients of the associated sparse matrix are constant in time. Taking into account this feature in the linear system solution strategy is a key ingredient for obtaining a computationally efficient method. For two-dimensional problems, a direct solver based on a LU factorization such as the one adopted in this study is generally considered as the optimal strategy, at least from the computing time point of view. In this study, by adopting a multifrontal sparse matrix solver, it has been shown through various numerical experiments involving homogeneous and heterogeneous propagation media that the proposed implicit discontinuous Galerkin time-domain method yields an efficient and accurate numerical strategy for solving two-dimensional time-domain wave propagation problems for practical situations where the underlying triangular mesh is highly non-uniform. Accuracy has been assessed here as the level of numerical dispersion error exhibited by the method with regards to the corresponding error for the reference explicit DGTD-$\mathbb{P}_p$ method. Clearly, for calculations involving quasi-uniform meshes, the implicit DGTD-$\mathbb{P}_p$ method will never outperform its explicit counterpart.

Concerning future works, our first objective will be to adapt the implicit DGTD-$\mathbb{P}_p$ method proposed here to the case of the three-dimensional time-domain Maxwell equations. From the mathematical formulation point of view, this extension will be straightforward. One possible strategy in the context of a globally implicit method is to resort to a preconditioned iterative linear system solver. An alternative approach which seems more promising is to restrict the application of the implicit time scheme to a subset of the elements of the mesh, namely those corresponding to the locally refined regions, while preserving an explicit time scheme for the remaining elements.

Chapter 3

Domain decomposition methods for Maxwell's equations

Over the last two decades, classical Schwarz methods have been extended to systems of hyperbolic partial differential equations, using characteristic transmission conditions, and it has been observed that the classical Schwarz method can be convergent even without overlap in certain cases. This is in strong contrast to the behavior of classical Schwarz methods applied to elliptic problems, for which overlap is essential for convergence. They were also extended to time harmonic Maxwell's equations, see [DJR92, CDJP97].

Over the last decade, a new class of overlapping Schwarz methods was developed for scalar partial differential equations, namely the optimized Schwarz methods. These methods are based on a classical overlapping domain decomposition, but they use more effective transmission conditions than the classical Dirichlet conditions at the interfaces between subdomains. New transmission conditions were originally proposed for three different reasons: first, to obtain Schwarz algorithms that are convergent without overlap. The second motivation for changing the transmission conditions was to obtain a convergent Schwarz method for the Helmholtz equation, where the classical overlapping Schwarz algorithm is not convergent. As a remedy, approximate radiation conditions were introduced in [DJR92, Des93]. The third motivation was that the convergence rate of the classical Schwarz method is rather slow and too strongly dependent on the size of the overlap. In a short note on non-linear problems [HTJ88], Hagstrom et al. introduced Robin transmission conditions between subdomains and suggested nonlocal operators for best performance. Optimized transmission conditions for the best performance of the Schwarz algorithm in a given class of local transmission conditions were introduced for the Helmholtz equation in [CN98, GMN02].

3.1 Optimized Schwarz methods for Maxwell's equations

The purpose of this work is to design and analyze a family of optimized overlapping and non-overlapping Schwarz methods for Maxwell's equations, both for the case of time discretized

and time harmonic problems, and to provide explicit formulas for the optimized parameters in the transmission conditions of each algorithm in the family. These formulas can then easily be used in implementations for Maxwell's equations with variable coefficients. As we will see, one member of this family reduces in the case of no overlap and constant coefficients to an algorithm in a curl-curl formulation of Maxwell's equations, proposed in [ARGG06] based on [Che98], which already greatly enhanced the performance compared to the classical approaches in [DJR92, CDJP97].

The hyperbolic system of Maxwell's equations describes the propagation of electromagnetic waves. It is given by

$$-\varepsilon\frac{\partial \boldsymbol{\mathcal{E}}}{\partial t} + \operatorname{curl} \boldsymbol{\mathcal{H}} - \sigma\boldsymbol{\mathcal{E}} = \boldsymbol{\mathcal{J}}, \qquad \mu\frac{\partial \boldsymbol{\mathcal{H}}}{\partial t} + \operatorname{curl} \boldsymbol{\mathcal{E}} = 0, \tag{3.1}$$

where $\boldsymbol{\mathcal{E}} = (\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3)^T$ and $\boldsymbol{\mathcal{H}} = (\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3)^T$ denote the electric and magnetic fields, respectively, ε is the *electric permittivity*, μ is the *magnetic permeability*, σ is the *electric conductivity* and $\boldsymbol{\mathcal{J}}$ is the applied current density. We assume the applied current density to be divergence free, that is $\operatorname{div}\boldsymbol{\mathcal{J}} = 0$. Denoting the vector of physical unknowns by

$$\boldsymbol{u} = (\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3, \mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3)^T, \tag{3.2}$$

Maxwell's equations (3.1) can be rewritten in the form

$$(G + G_0\partial_t)\boldsymbol{u} + G_x\partial_x\boldsymbol{u} + G_y\partial_y\boldsymbol{u} + G_z\partial_z\boldsymbol{u} = (\boldsymbol{\mathcal{J}}; \boldsymbol{0}), \tag{3.3}$$

where the coefficient matrices are

$$G = \begin{bmatrix} \sigma I_3 & \\ & 0_3 \end{bmatrix}, \; G_0 = \begin{bmatrix} \varepsilon I_3 & \\ & \mu I_3 \end{bmatrix}, \; G_l = \begin{bmatrix} & N_l \\ -N_l & \end{bmatrix}, \qquad l = x, y, z,$$

where 0_3 (resp. I_3) represent the 3×3 zero (identity) matrix, and the matrices N_l, $l = x, y, z$ are given by

$$N_x = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}, \qquad N_y = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \qquad N_z = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

For any unit vector $\boldsymbol{n} = (n_1, n_2, n_3)$, $\|\boldsymbol{n}\| = 1$, we can define the characteristic matrix of system (3.3) by

$$C(\boldsymbol{n}) = G_0^{-1}\left(n_1\begin{bmatrix} & N_x \\ -N_x & \end{bmatrix} + n_2\begin{bmatrix} & N_y \\ -N_y & \end{bmatrix} + n_3\begin{bmatrix} & N_z \\ -N_z & \end{bmatrix}\right),$$

whose eigenvalues are the characteristic speed of propagation along the direction $\boldsymbol{n}$. A direct calculation shows that the matrix $C(\boldsymbol{n})$ has real eigenvalues,

$$\lambda_{1,2} = -c, \qquad \lambda_{3,4} = 0, \qquad \lambda_{5,6} = c,$$

with $c = \frac{1}{\sqrt{\varepsilon\mu}}$ being the wave speed. This implies that Maxwell's equations are hyperbolic, since the eigenvalues are real, but not strictly hyperbolic, since the eigenvalues are not distinct, see [BGS07]. For the special case of the normal vector $\boldsymbol{n} = (1, 0, 0)$, which we will use extensively later, we obtain

$$C(\boldsymbol{n}) = \begin{pmatrix} & \frac{1}{\varepsilon}N_x \\ -\frac{1}{\mu}N_x & \end{pmatrix},$$

whose matrix of eigenvectors is given by

$$L = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ -Z & 0 & 0 & 0 & Z & 0 \\ 0 & Z & 0 & 0 & 0 & -Z \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix},$$

where $Z = \sqrt{\frac{\mu}{\varepsilon}}$ denotes the impedance. This leads to the characteristic variables $\boldsymbol{w} = (w_1, w_2, w_3, w_4, w_5, w_6)^T = L^{-1}\boldsymbol{u}$ associated with the direction $\boldsymbol{n}$, where

$$\begin{array}{lll} w_1 = -\frac{1}{2}(\frac{1}{Z}\mathcal{E}_2 - \mathcal{H}_3), & w_2 = \frac{1}{2}(\frac{1}{Z}\mathcal{E}_3 + \mathcal{H}_2), & w_3 = \mathcal{H}_1, \\ w_4 = \mathcal{E}_1, & w_5 = \frac{1}{2}(\frac{1}{Z}\mathcal{E}_2 + \mathcal{H}_3), & w_6 = -\frac{1}{2}(\frac{1}{Z}\mathcal{E}_3 - \mathcal{H}_2). \end{array} \tag{3.4}$$

In the following, we will denote by $\boldsymbol{w}_+$, $\boldsymbol{w}_0$ and $\boldsymbol{w}_-$ the characteristic variables associated with the negative, zero, and positive eigenvalues respectively, that is

$$\boldsymbol{w}_- = (w_1, w_2)^T, \qquad \boldsymbol{w}_0 = (w_3, w_4)^T, \qquad \boldsymbol{w}_+ = (w_5, w_6)^T. \tag{3.5}$$

Imposing classical or characteristic boundary conditions on a boundary with unit outward normal vector $\boldsymbol{n} = (1, 0, 0)$ means to impose Dirichlet conditions on the incoming characteristic variables $\boldsymbol{w}_-$. For a general normal vector $\boldsymbol{n}$, this is equivalent to imposing the impedance condition (see [BGS07])

$$\mathcal{B}_{\boldsymbol{n}}(\boldsymbol{\mathcal{E}}, \boldsymbol{\mathcal{H}}) := \boldsymbol{n} \times \frac{\boldsymbol{\mathcal{E}}}{Z} + \boldsymbol{n} \times (\boldsymbol{\mathcal{H}} \times \boldsymbol{n}) = \boldsymbol{s}. \tag{3.6}$$

Time harmonic solutions of Maxwell's equations are complex valued static vector fields $\boldsymbol{E}$ and $\boldsymbol{H}$ such that the dynamic fields

$$\boldsymbol{\mathcal{E}}(\boldsymbol{x}, t) = \mathcal{R}e(\boldsymbol{E}(\boldsymbol{x}) \exp(i\omega t)), \qquad \boldsymbol{\mathcal{H}}(\boldsymbol{x}, t) = \mathcal{R}e(\boldsymbol{H}(\boldsymbol{x}) \exp(i\omega t))$$

satisfy Maxwell's equations (3.1). The positive real parameter ω is called the *pulsation* of the harmonic wave. The harmonic solutions $\boldsymbol{E}$ and $\boldsymbol{H}$ satisfy the time-harmonic Maxwell's equations

$$-i\omega\varepsilon\boldsymbol{E} + \operatorname{curl} \boldsymbol{H} - \sigma\boldsymbol{E} = \boldsymbol{J}, \qquad i\omega\mu\boldsymbol{H} + \operatorname{curl} \boldsymbol{E} = \boldsymbol{0}. \tag{3.7}$$

We consider now the problem (3.7) in a bounded domain Ω, with either Dirichlet conditions on the tangent electric field, or impedance conditions, on $\partial\Omega$, in order to obtain a well posed problem, see [Ned01]. In order to explain the classical Schwarz algorithm for Maxwell's equation, we decompose the domain into two overlapping subdomains Ω_1 and Ω_2. The generalization of the algorithm formulation to the case of many subdomains does not present any difficulties. The classical Schwarz algorithm then solves for $n = 1, 2 \ldots$ the subdomain problems

$$\begin{array}{rcll}
-i\omega\varepsilon \boldsymbol{E}^{1,n} + \text{curl } \boldsymbol{H}^{1,n} - \sigma \boldsymbol{E}^{1,n} &=& \boldsymbol{J} & \text{in } \Omega_1 \\
i\omega\mu \boldsymbol{H}^{1,n} + \text{curl } \boldsymbol{E}^{1,n} &=& \boldsymbol{0} & \text{in } \Omega_1 \\
\mathcal{B}_{\boldsymbol{n}_1}(\boldsymbol{E}^{1,n}, \boldsymbol{H}^{1,n}) &=& \mathcal{B}_{\boldsymbol{n}_1}(\boldsymbol{E}^{2,n-1}, \boldsymbol{H}^{2,n-1}) & \text{on } \Gamma_{12} \\
-i\omega\varepsilon \boldsymbol{E}^{2,n} + \text{curl } \boldsymbol{H}^{2,n} - \sigma \boldsymbol{E}^{2,n} &=& \boldsymbol{J} & \text{in } \Omega_2 \\
i\omega\mu \boldsymbol{H}^{2,n} + \text{curl } \boldsymbol{E}^{2,n} &=& \boldsymbol{0} & \text{in } \Omega_2 \\
\mathcal{B}_{\boldsymbol{n}_2}(\boldsymbol{E}^{2,n}, \boldsymbol{H}^{2,n}) &=& \mathcal{B}_{\boldsymbol{n}_2}(\boldsymbol{E}^{1,n-1}, \boldsymbol{H}^{1,n-1}) & \text{on } \Gamma_{21},
\end{array} \tag{3.8}$$

where $\Gamma_{12} = \partial\Omega_1 \cap \Omega_2$, $\Gamma_{21} = \partial\Omega_2 \cap \Omega_1$ and $\mathcal{B}_{\boldsymbol{n}_j}$, $j = 1, 2$, denotes the impedance boundary conditions defined in (3.6). On the physical part of the boundary, the given boundary conditions are imposed. While the choice of transmission conditions $\mathcal{B}_{\boldsymbol{n}_j}$ is natural in the view of the hyperbolic nature of the problem, we will see in our analysis that there are better choices for the performance of the algorithm, based on the notion of absorbing boundary conditions. This leads to the so called optimized Schwarz methods,

$$\begin{array}{rcll}
-i\omega\varepsilon \boldsymbol{E}^{1,n} + \text{curl } \boldsymbol{H}^{1,n} - \sigma \boldsymbol{E}^{1,n} &=& \boldsymbol{J} & \text{in } \Omega_1 \\
i\omega\mu \boldsymbol{H}^{1,n} + \text{curl } \boldsymbol{E}^{1,n} &=& \boldsymbol{0} & \text{in } \Omega_1 \\
(\mathcal{B}_{\boldsymbol{n}_1} + \mathcal{S}_1\mathcal{B}_{\boldsymbol{n}_2})(\boldsymbol{E}^{1,n}, \boldsymbol{H}^{1,n}) &=& (\mathcal{B}_{\boldsymbol{n}_1} + \mathcal{S}_1\mathcal{B}_{\boldsymbol{n}_2})(\boldsymbol{E}^{2,n-1}, \boldsymbol{H}^{2,n-1}) & \text{on } \Gamma_{12} \\
-i\omega\varepsilon \boldsymbol{E}^{2,n} + \text{curl } \boldsymbol{H}^{2,n} - \sigma \boldsymbol{E}^{2,n} &=& \boldsymbol{J} & \text{in } \Omega_2 \\
i\omega\mu \boldsymbol{H}^{2,n} + \text{curl } \boldsymbol{E}^{2,n} &=& \boldsymbol{0} & \text{in } \Omega_2 \\
(\mathcal{B}_{\boldsymbol{n}_2} + \mathcal{S}_2\mathcal{B}_{\boldsymbol{n}_1})(\boldsymbol{E}^{2,n}, \boldsymbol{H}^{2,n}) &=& (\mathcal{B}_{\boldsymbol{n}_2} + \mathcal{S}_2\mathcal{B}_{\boldsymbol{n}_1})(\boldsymbol{E}^{1,n-1}, \boldsymbol{H}^{1,n-1}) & \text{on } \Gamma_{21},
\end{array} \tag{3.9}$$

where $\mathcal{S}_j$, $j = 1, 2$ are tangential, possibly pseudo-differential operators we will study in what follows in order to obtain various optimized Schwarz methods.

We now study properties of the classical Schwarz algorithm (3.8). We use Fourier analysis, and thus assume that the coefficients are constant, and the domain on which the original problem is posed is $\Omega = \mathbb{R}^3$, in which case we need for Maxwell's equations the Silver-Müller radiation condition

$$\lim_{r\to\infty} r\,(\boldsymbol{H} \times \boldsymbol{n} - \boldsymbol{E}) = 0, \tag{3.10}$$

where $r = |\boldsymbol{x}|$, $\boldsymbol{n} = \boldsymbol{x}/|\boldsymbol{x}|$, in order to obtain well-posed problems, see [Ned01]. The two subdomains are now half spaces,

$$\Omega_1 = (0, \infty) \times \mathbb{R}^2, \quad \Omega_2 = (-\infty, L) \times \mathbb{R}^2, \tag{3.11}$$

the interfaces are $\Gamma_{12} = \{L\} \times \mathbb{R}^2$ and $\Gamma_{21} = \{0\} \times \mathbb{R}^2$, and the overlap is $L \geq 0$. We denote by k_y and k_z the Fourier variables corresponding to a transform with respect to y and z, respectively, and $|\boldsymbol{k}|^2 = k_y^2 + k_z^2$.

Theorem 4 *For any given initial guess* $(\boldsymbol{E}^{1,0};\boldsymbol{H}^{1,0}) \in (L^2(\Omega_1))^6$, $(\boldsymbol{E}^{2,0};\boldsymbol{H}^{2,0}) \in (L^2(\Omega_2))^6$, *the classical Schwarz algorithm (3.8) with overlap* $L \geq 0$, *including the non-overlapping case, is for* $\sigma > 0$ *convergent in* $(L^2(\Omega_1))^6 \times (L^2(\Omega_2))^6$, *and the convergence factor for each Fourier mode* $\boldsymbol{k}$ *is*

$$\rho_{cla}(\boldsymbol{k},\tilde{\omega},\sigma,Z,L) = \left| \frac{\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2 + i\tilde{\omega}\sigma Z} - i\tilde{\omega}}{\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2 + i\tilde{\omega}\sigma Z} + i\tilde{\omega}} e^{-\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2 + i\tilde{\omega}\sigma Z}L} \right|, \tag{3.12}$$

where $\tilde{\omega} = \omega\sqrt{\varepsilon\mu}$, *and* $Z = \sqrt{\frac{\mu}{\varepsilon}}$ *is the impedance as before.*

If $\sigma = 0$, the convergence factor becomes

$$\rho_{cla}(\boldsymbol{k},\tilde{\omega},0,Z,L) = \begin{cases} \left| \frac{\sqrt{\tilde{\omega}^2 - |\boldsymbol{k}|^2} - \tilde{\omega}}{\sqrt{\tilde{\omega}^2 - |\boldsymbol{k}|^2} + \tilde{\omega}} \right|, & \text{for } |\boldsymbol{k}|^2 \leq \tilde{\omega}^2, \\ e^{-\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2}L}, & \text{for } |\boldsymbol{k}|^2 > \tilde{\omega}^2. \end{cases} \tag{3.13}$$

In this case, we obtain for $|\boldsymbol{k}|^2 = \tilde{\omega}^2$ that the convergence factor equals 1, independently of the overlap, which indicates that the algorithm has convergence problems for $\sigma = 0$ when used in the iterative form described here. Convergence can still be proved in the case of a bounded domain with suitable boundary conditions, see [DJR92]. In addition in practice, Schwarz methods are often used as preconditioners for Krylov methods, which can handle isolated problems in the spectrum. We also see from the convergence factor (3.13) that in the case $\sigma = 0$ the overlap is necessary for the convergence of the evanescent modes, $|\boldsymbol{k}|^2 > \tilde{\omega}^2$. Without overlap, $L = 0$, we have $\rho_{cla}(\boldsymbol{k}) < 1$ only for the propagative modes, $|\boldsymbol{k}|^2 < \tilde{\omega}^2$, and $\rho_{cla}(\boldsymbol{k}) = 1$ when $|\boldsymbol{k}|^2 \geq \tilde{\omega}^2$.

Very similar observations were made in the analysis of optimized Schwarz methods for the Helmholtz equation in [GMN02]. If one applies to the Helmholtz equation

$$(\Delta + \tilde{\omega}^2)u = f, \qquad \text{in } \Omega = \mathbb{R}^3, \tag{3.14}$$

with Sommerfeld radiation conditions $\lim_{r\to\infty} r\left(\frac{\partial u}{\partial r} - i\tilde{\omega}u\right) = 0$ and the same two subdomain decomposition (3.11) the somewhat particular overlapping Schwarz method (note the unequal treatment in the transmission conditions)

$$\begin{aligned} (\tilde{\omega}^2 + \Delta)u_1^{1,n} &= f \quad \text{in } \Omega_1 & \qquad (\tilde{\omega}^2 + \Delta)u_1^{2,n} &= f \quad \text{in } \Omega_2, \\ u_1^{1,n} &= u_1^{2,n-1} \quad \text{on } \Gamma_{12} & \qquad (\partial_x - i\tilde{\omega})u_1^{2,n} &= (\partial_x - i\tilde{\omega})u_1^{1,n-1} \quad \text{on } \Gamma_{21}, \end{aligned} \tag{3.15}$$

then one obtains precisely the same convergence factor (3.13). The classical overlapping Schwarz algorithm with characteristic transmission conditions (3.8) for Maxwell's equations is thus equivalent to the particular overlapping Schwarz method (3.15) for the Helmholtz problem when $\sigma = 0$. This particular Schwarz method is a very simple variant of an optimized Schwarz method, where one has only replaced one of the Dirichlet transmission conditions with a better one adapted for low frequencies. There are much better transmission conditions for Helmholtz problems, as shown in [GMN02]. These conditions are based on approximations of transparent boundary conditions, which we will study next.

To design optimized Schwarz methods for Maxwell's equations, we derive now transparent boundary conditions for those equations, following the approach in [HL07]. We consider the time harmonic Maxwell's equations (3.7) on the domains $\Omega_1 = (-\infty, L) \times \mathbb{R}^2$ and $\Omega_2 = (0, \infty) \times \mathbb{R}^2$ with right hand sides $\boldsymbol{J}_{1,2}$ compactly supported in $\Omega_{1,2}$, together with the boundary conditions

$$(\boldsymbol{w}_+^2 + \mathcal{S}_1 \boldsymbol{w}_-^2)(0, y, z) = 0, \quad (\boldsymbol{w}_-^1 + \mathcal{S}_2 \boldsymbol{w}_+^1)(L, y, z) = 0, \qquad (y, z) \in \mathbb{R}^2, \tag{3.16}$$

and with the Silver-Müller condition on their unbounded part, where $\boldsymbol{w}_-^1$ and $\boldsymbol{w}_+^2$ are defined in (3.5), and the operators $\mathcal{S}_l$, $l = 1, 2$, are general, pseudo-differential operators acting in the y and z directions.

Theorem 5 *If the operators $\mathcal{S}_l$, $l = 1, 2$ have the Fourier symbol*

$$\mathcal{F}(\mathcal{S}_l) = \frac{1}{(\lambda + i\tilde{\omega})(\lambda + i\tilde{\omega} + \sigma Z)} \begin{bmatrix} k_y^2 - k_z^2 - \lambda\sigma Z & -2k_y k_z \\ -2k_y k_z & k_z^2 - k_y^2 - \lambda\sigma Z \end{bmatrix}, \tag{3.17}$$

where $\lambda = \sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2 + i\tilde{\omega}\sigma Z}$, then the solution of Maxwell's equations (3.7) in $\Omega_{1,2}$ with boundary conditions (3.16) coincides with the restriction on $\Omega_{1,2}$ of the solution of Maxwell's equations (3.7) on $\mathbb{R}^3$.

Remark 5 *As in the case of the Cauchy-Riemann equations, see [13], the symbols in (3.17) can be written in several, mathematically equivalent forms,*

$$\begin{array}{rclcl} \mathcal{F}(\mathcal{S}_l) & = & \frac{1}{(\lambda+i\tilde{\omega})(\lambda+i\tilde{\omega}+\sigma Z)} M & = & \frac{1}{|\boldsymbol{k}|^2+\lambda\sigma Z} \frac{\lambda - i\tilde{\omega}}{\lambda + i\tilde{\omega}} M \\ & = & \frac{1}{|\boldsymbol{k}|^2-\lambda\sigma Z} \frac{\lambda - i\tilde{\omega} - \sigma Z}{\lambda + i\tilde{\omega} + \sigma Z} M & = & (\lambda - i\tilde{\omega})(\lambda - i\tilde{\omega} - \sigma Z)\tilde{M}^{-1}, \end{array}$$

where the matrices M and $\tilde{M}$ are given by

$$M = \begin{bmatrix} k_y^2 - k_z^2 - \lambda\sigma Z & -2k_y k_z \\ -2k_y k_z & k_z^2 - k_y^2 - \lambda\sigma Z \end{bmatrix}, \quad \tilde{M} = \begin{bmatrix} k_y^2 - k_z^2 + \lambda\sigma Z & -2k_y k_z \\ -2k_y k_z & k_z^2 - k_y^2 + \lambda\sigma Z \end{bmatrix}.$$

This motivates different approximations of the transparent conditions in the context of optimized Schwarz methods. In the case $\sigma = 0$ the first form contains a local and a non-local term, since multiplication with the matrix M corresponds to second order derivatives in y and z, which are local operations, whereas the term containing the square-root of $|\boldsymbol{k}|^2$ represents a non-local operation. The last form contains two non-local operations, since the inversion of the matrix M corresponds to an integration. This integration can however be passed to the other side of the transmission conditions by multiplication with the matrix M. The second form contains two non-local terms and a local one. We propose next ,several approximations based on these different forms, and analyze the performance of the associated optimized Schwarz algorithms.

The transparent operators are non-local operators, and hence difficult to use in practice. In optimized Schwarz methods, they are therefore approximated to obtain practical methods.

If one is willing to use second order transmission conditions, then the only parts of the symbols in (3.17) that need to be approximated are the terms $\lambda = \sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2 + i\tilde{\omega}\sigma Z}$, because the entries of the matrices are polynomials in the Fourier variables, which correspond to derivatives in the y and z direction.

Theorem 6 *For the optimized Schwarz algorithm (3.9) with the two subdomain decomposition (3.11), we obtain for $\sigma = 0$ the following results:*

1. *If the operators $\mathcal{S}_1$ and $\mathcal{S}_2$ have the Fourier symbol*

$$\sigma_l := \mathcal{F}(\mathcal{S}_l) = \gamma_l \begin{bmatrix} k_y^2 - k_z^2 & -2k_y k_z \\ -2k_y k_z & k_z^2 - k_y^2 \end{bmatrix}, \quad \gamma_l \in \mathbb{C}(k_z, k_y),\ l = 1, 2, \tag{3.18}$$

then the convergence factor is

$$\rho = \left| \frac{(\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} - i\tilde{\omega})^2}{(\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} + i\tilde{\omega})^2} \frac{1 - \gamma_1(\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} + i\tilde{\omega})^2}{1 - \gamma_1(\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} - i\tilde{\omega})^2} \frac{1 - \gamma_2(\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} + i\tilde{\omega})^2}{1 - \gamma_2(\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} - i\tilde{\omega})^2} e^{-2\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} L} \right|^{\frac{1}{2}}. \tag{3.19}$$

2. *If the operators $\mathcal{S}_1$ and $\mathcal{S}_2$ have the Fourier symbol*

$$\sigma_l := \mathcal{F}(\mathcal{S}_l) = \delta_l \begin{bmatrix} k_y^2 - k_z^2 & -2k_y k_z \\ -2k_y k_z & k_z^2 - k_y^2 \end{bmatrix}^{-1}, \quad \gamma_l \in \mathbb{C}(k_z, k_y),\ l = 1, 2, \tag{3.20}$$

then the convergence factor is

$$\rho = \left| \frac{(\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} + i\tilde{\omega})^2}{(\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} - i\tilde{\omega})^2} \frac{\delta_1 - (\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} - i\tilde{\omega})^2}{\delta_1 - (\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} + i\tilde{\omega})^2} \frac{\delta_2 - (\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} - i\tilde{\omega})^2}{\delta_2 - (\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} + i\tilde{\omega})^2} e^{-2\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} L} \right|^{\frac{1}{2}}. \tag{3.21}$$

3. *If the operator $\mathcal{S}_1$ has the Fourier symbol (3.18) and $\mathcal{S}_2$ has the Fourier symbol (3.20), then the convergence factor is*

$$\rho = \left| \frac{1 - \gamma_1(\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} + i\tilde{\omega})^2}{1 - \gamma_1(\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} - i\tilde{\omega})^2} \frac{\delta_2 - (\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} - i\tilde{\omega})^2}{\delta_2 - (\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} + i\tilde{\omega})^2} e^{-2\sqrt{|\boldsymbol{k}|^2 - \tilde{\omega}^2} L} \right|^{1/2}. \tag{3.22}$$

We present now several particular choices of the remaining parameters in the transmission operators $\mathcal{S}_l$ in Theorem 6 for $\sigma = 0$. To facilitate the use of our results in domain decomposition codes, we return to the initial notation using the physical parameters ω, ε and μ.

Case 1: taking $\gamma_1 = \gamma_2 = 0$ in (3.18), which amounts to enforce the classical characteristic Dirichlet transmission conditions, the convergence factor is

$$\rho_1(\omega, \varepsilon, \mu, L, |\boldsymbol{k}|) = \left| \left(\frac{\sqrt{|\boldsymbol{k}|^2 - \omega^2\varepsilon\mu} - i\omega\sqrt{\varepsilon\mu}}{\sqrt{|\boldsymbol{k}|^2 - \omega^2\varepsilon\mu} + i\omega\sqrt{\varepsilon\mu}} \right)^2 e^{-2\sqrt{|\boldsymbol{k}|^2 - \omega^2\varepsilon\mu} L} \right|^{\frac{1}{2}}.$$

In the non-overlapping case, $L = 0$, this choice ensures convergence only for propagative modes, and corresponds to the Taylor transmission conditions of order zero proposed in the seminal paper [Des93] for the Helmholtz equation.

Case 2: taking $\gamma_1 = \gamma_2 = \frac{1}{|\boldsymbol{k}|^2}\frac{s-i\omega\sqrt{\varepsilon\mu}}{s+i\omega\sqrt{\varepsilon\mu}}$ in (3.18) or $\gamma_1 = \frac{1}{|\boldsymbol{k}|^2-2\omega^2\varepsilon\mu+2i\omega\sqrt{\varepsilon\mu}s}$ in (3.18) and $\delta_2 = |\boldsymbol{k}|^2 - 2\omega^2\varepsilon\mu - 2i\omega\sqrt{\varepsilon\mu}s$ in (3.20) with $s \in \mathbb{C}$, the convergence factor is

$$\rho_2(\omega,\varepsilon,\mu,L,|\boldsymbol{k}|,s) = \left| \left(\frac{\sqrt{|\boldsymbol{k}|^2-\omega^2\varepsilon\mu}-s}{\sqrt{|\boldsymbol{k}|^2-\omega^2\varepsilon\mu}+s} \right)^2 e^{-2\sqrt{|\boldsymbol{k}|^2-\omega^2\varepsilon\mu}L} \right|^{\frac{1}{2}}.$$

which is for $L = 0$ identical to the convergence factor obtained for optimized non-overlapping Schwarz methods for the Helmholtz equation in [**?**].

Case 3: taking $\gamma_1 = \gamma_2 = \frac{1}{|\boldsymbol{k}|^2-2\omega^2\varepsilon\mu+2i\omega\sqrt{\varepsilon\mu}s}$ in (3.18) with $s \in \mathbb{C}$, the convergence factor is

$$\begin{aligned} \rho_3(\omega,\varepsilon,\mu,L,|\boldsymbol{k}|,s) &= \left| \frac{\sqrt{|\boldsymbol{k}|^2-\omega^2\varepsilon\mu}-i\omega\sqrt{\varepsilon\mu}}{\sqrt{|\boldsymbol{k}|^2-\omega^2\varepsilon\mu}+i\omega\sqrt{\varepsilon\mu}} \right| \rho_2(\omega,\varepsilon,\mu,L,|\boldsymbol{k}|,s) \\ &\leq \rho_2(\omega,\varepsilon,\mu,L,|\boldsymbol{k}|,s). \end{aligned}$$

Case 4: taking $\gamma_l = \frac{1}{|\boldsymbol{k}|^2}\frac{s_l-i\omega\sqrt{\varepsilon\mu}}{s_l+i\omega\sqrt{\varepsilon\mu}}$, $l = 1,2$ in (3.18) or $\gamma_1 = \frac{1}{|\boldsymbol{k}|^2-2\omega^2\varepsilon\mu+2i\omega\sqrt{\varepsilon\mu}s_1}$ in (3.18) and $\delta_2 = |\boldsymbol{k}|^2 - 2\omega^2\varepsilon\mu - 2i\omega\sqrt{\varepsilon\mu}s_2$ in (3.20) with $s_l \in \mathbb{C}$, $l = 1,2$, the convergence factor is

$$\rho_4(\omega,\varepsilon,\mu,L,|\boldsymbol{k}|,s_1,s_2) = \left| \frac{\sqrt{|\boldsymbol{k}|^2-\omega^2\varepsilon\mu}-s_1}{\sqrt{|\boldsymbol{k}|^2-\omega^2\varepsilon\mu}+s_1} \frac{\sqrt{|\boldsymbol{k}|^2-\omega^2\varepsilon\mu}-s_2}{\sqrt{|\boldsymbol{k}|^2-\omega^2\varepsilon\mu}+s_2} e^{-2\sqrt{|\boldsymbol{k}|^2-\omega^2\varepsilon\mu}L} \right|^{\frac{1}{2}},$$

which is for $L = 0$ identical to the convergence factor obtained for a two sided non-overlapping optimized Schwarz method for the Helmholtz equation in [GHM07].

Case 5: taking $\gamma_l = \frac{1}{|\boldsymbol{k}|^2-2\omega^2\varepsilon\mu+2i\omega\sqrt{\varepsilon\mu}s_l}$ in (3.18) with $s_l \in \mathbb{C}$, $l = 1,2$, the convergence factor is

$$\begin{aligned} \rho_5(\omega,\varepsilon,\mu,L,|\boldsymbol{k}|,s_1,s_2) &= \left| \frac{\sqrt{|\boldsymbol{k}|^2-\omega^2\varepsilon\mu}-i\omega\sqrt{\varepsilon\mu}}{\sqrt{|\boldsymbol{k}|^2-\omega^2\varepsilon\mu}+i\omega\sqrt{\varepsilon\mu}} \right| \rho_4(\omega,\varepsilon,\mu,L,|\boldsymbol{k}|,s_1,s_2) \\ &\leq \rho_4(\omega,\varepsilon,\mu,L,|\boldsymbol{k}|,s_1,s_2). \end{aligned}$$

Except for Case 1, all cases use second order transmission conditions, even though we use only a zeroth order approximation of the non-local operator $\sqrt{|\boldsymbol{k}|^2-\omega^2\varepsilon\mu}$. In the cases with parameters, the best choice for the parameters is in general the one that minimizes the convergence factor for all $|\boldsymbol{k}| \in K$, where K denotes the set of relevant numerical frequencies. One therefore needs to solve the min-max problems

$$\min_{s\in\mathbb{C}} \max_{|\boldsymbol{k}|\in K} \rho_j(\omega,\varepsilon,\mu,L,|\boldsymbol{k}|,s),\ j = 2,3, \qquad \min_{s_1,s_2\in C} \max_{|\boldsymbol{k}|\in K} \rho_j(\omega,\varepsilon,\mu,L,|\boldsymbol{k}|,s_1,s_2),\ j = 4,5. \tag{3.23}$$

We can choose $K = [(k_{\min},k_-) \cup (k_+,k_{\max})]^2$, where k_{min} denotes the smallest frequency relevant to the subdomain, and $k_{max} = \frac{C}{h}$ denotes the largest frequency supported by the numerical grid with mesh size h, and $k_\pm$ are parameters to be chosen to exclude the resonance

	with overlap, $L = h$		without overlap, $L = 0$	
Case	ρ	parameters	ρ	parameters
1	$1 - \sqrt{k_+^2 - \tilde{\omega}^2}h$	none	1	none
2	$1 - 2C_{\tilde{\omega}}^{\frac{1}{6}} h^{\frac{1}{3}}$	$p = \frac{C_{\tilde{\omega}}^{\frac{1}{3}}}{2 \cdot h^{\frac{1}{3}}}$	$1 - \frac{\sqrt{2} C_{\tilde{\omega}}^{\frac{1}{4}}}{\sqrt{C}}\sqrt{h}$	$p = \frac{\sqrt{C} C_{\tilde{\omega}}^{\frac{1}{4}}}{\sqrt{2}\sqrt{h}}$
3	$1 - 2(k_+^2 - \tilde{\omega}^2)^{\frac{1}{6}} h^{\frac{1}{3}}$	$p = \frac{(k_+^2 - \tilde{\omega}^2)^{\frac{1}{3}}}{2 \cdot h^{\frac{1}{3}}}$	$1 - \frac{\sqrt{2}(k_+^2 - \tilde{\omega}^2)^{\frac{1}{4}}}{\sqrt{C}}\sqrt{h}$	$p = \frac{\sqrt{C}(k_+^2 - \tilde{\omega}^2)^{\frac{1}{4}}}{\sqrt{2}\sqrt{h}}$
4	$1 - 2^{\frac{2}{5}} C_{\tilde{\omega}}^{\frac{1}{10}} h^{\frac{1}{5}}$	$\begin{cases} p_1 = \frac{C_{\tilde{\omega}}^{\frac{2}{5}}}{2^{\frac{7}{5}} \cdot h^{\frac{1}{5}}}, \\ p_2 = \frac{C_{\tilde{\omega}}^{\frac{1}{5}}}{2^{\frac{6}{5}} \cdot h^{\frac{3}{5}}} \end{cases}$	$1 - \frac{C_{\tilde{\omega}}^{\frac{1}{8}}}{C^{\frac{1}{4}}} h^{\frac{1}{4}}$	$\begin{cases} p_1 = \frac{C_{\tilde{\omega}}^{\frac{3}{8}} \cdot C^{\frac{1}{4}}}{2 \cdot h^{\frac{1}{4}}}, \\ p_2 = \frac{C_{\tilde{\omega}}^{\frac{1}{8}} \cdot C^{\frac{3}{4}}}{h^{\frac{3}{4}}} \end{cases}$
5	$1 - 2^{\frac{2}{5}} (k_+^2 - \tilde{\omega}^2)^{\frac{1}{10}} h^{\frac{1}{5}}$	$\begin{cases} p_1 = \frac{(k_+^2 - \tilde{\omega}^2)^{\frac{2}{5}}}{2^{\frac{7}{5}} \cdot h^{\frac{1}{5}}}, \\ p_2 = \frac{(k_+^2 - \tilde{\omega}^2)^{\frac{1}{5}}}{2^{\frac{6}{5}} \cdot h^{\frac{3}{5}}} \end{cases}$	$1 - \frac{(k_+^2 - \tilde{\omega}^2)^{\frac{1}{8}}}{C^{\frac{1}{4}}} h^{\frac{1}{4}}$	$\begin{cases} p_1 = \frac{(k_+^2 - \tilde{\omega}^2)^{\frac{3}{8}} \cdot C^{\frac{1}{4}}}{2 \cdot h^{\frac{1}{4}}}, \\ p_2 = \frac{(k_+^2 - \tilde{\omega}^2)^{\frac{1}{8}} \cdot C^{\frac{3}{4}}}{h^{\frac{3}{4}}} \end{cases}$

Table 3.1: Asymptotic convergence factor and optimal choice of the parameters in the transmission conditions for the five variants of the optimized Schwarz method applied to Maxwell's equations, when the mesh parameter h is small, and the maximum numerical frequency is estimated by $k_{\max} = \frac{C}{h}$. Here $\tilde{\omega} = \omega\sqrt{\varepsilon\mu}$ and $C_{\tilde{\omega}} = \min\left(k_+^2 - \tilde{\omega}^2, \tilde{\omega}^2 - k_-^2\right)$.

frequencies. If for example the domain Ω is a rectilinear conductor with homogeneous Dirichlet conditions on the lateral surface, the solution is the sum of the transverse electric (TE) and transverse magnetic (TM) fields. If the transverse section of the conductor is a rectangle with sides of length a and b, the TE and TM fields can be expanded in a Fourier series with the harmonics $\sin(\frac{n\pi y}{a})\sin(\frac{m\pi z}{b})$, where the relevant frequencies are $|\boldsymbol{k}| = \pi\sqrt{\frac{m^2}{a^2} + \frac{n^2}{b^2}}$, $m, n \in \mathbb{N}^+$. The lowest one is therefore $k_{min} = \pi\sqrt{\frac{1}{a^2} + \frac{1}{b^2}}$, and if the mesh size h satisfies $h = \frac{a}{N} = \frac{b}{M}$, where N and M are the number of grid points in the y and z direction, then the highest frequency would be $k_{\max} = \frac{\sqrt{2}\pi}{h}$. The parameters $k_\pm$ would correspond to the frequencies closest to $\omega\sqrt{\varepsilon\mu}$, i.e. $k_- = \pi\sqrt{\frac{m_1^2}{a^2} + \frac{n_1^2}{b^2}}$ and $k_+ = \pi\sqrt{\frac{m_2^2}{a^2} + \frac{n_2^2}{b^2}}$, where $\pi\sqrt{\frac{m_1^2}{a^2} + \frac{n_1^2}{b^2}} < \omega\sqrt{\varepsilon\mu} < \pi\sqrt{\frac{m_2^2}{a^2} + \frac{n_2^2}{b^2}}$, but such precise estimates are not necessary if Krylov acceleration is used, see [GMN02, GHM07].

The complete mathematical analysis of the min-max problems (3.23) is hard, and currently open for $L > 0$. When $L = 0$, i.e. no overlap, Case 2 and Case 4 are equivalent to the corresponding optimized Schwarz method for the Helmholtz equation, for which theoretical results are available, see [GHM07]. Here, we use asymptotic analysis and an equioscillation principle to solve all the min-max problems in (3.23) asymptotically as the mesh size goes to zero, in order to obtain compact formulas for the best parameters to be used in our numerical simulations. This leads to the asymptotic formulas for the optimized parameters of the form $s = p(1 - i)$ and $s_l = p_l(1 - i)$, $l = 1, 2$, with p and p_l shown in Table 3.1. These results allow us to compare the performance of all the optimized Schwarz methods for

	with overlap, $L = h$				without overlap, $L = 0$			
h	1/32	1/64	1/128	1/256	1/32	1/64	1/128	1/256
Case 1	27(21)	47(27)	72(33)	118(45)	-(73)	-(100)	-(138)	-(181)
Case 2	19(14)	19(15)	22(17)	26(19)	41(26)	57(34)	79(40)	111(47)
Case 3	13(13)	14(14)	17(17)	21(18)	41(23)	56(25)	80(28)	115(35)
Case 4	19(14)	21(16)	24(18)	27(19)	31(24)	35(28)	41(30)	47(33)
Case 5	13(13)	15(15)	17(18)	19(19)	56(26)	64(30)	76(32)	76(35)

Table 3.2: Number of iterations in the 2d time harmonic case to attain a relative residual reduction of 10^{-6} for different transmission conditions and different mesh sizes.

Maxwell's equations theoretically: we obtain a hierarchy of better and better convergence factors starting with Case 1 and ending with Case 5. In addition, the explicit formulas for the optimized parameters can be used in order to easily obtain black-box optimized Schwarz methods for Maxwell's equations, which would not be possible otherwise.

The same type of analysis can be performed for the time-discretized equations (detailed results can be found in [20].). Again we obtain an entire hierarchy of optimized Schwarz methods, with better and better convergence factors from Case 1 up to Case 5. While for the time harmonic equations Case 2 and 3, and Case 4 and 5 were asymptotically comparable, here all cases are asymptotically different.

We discretize the equations using a finite volume method on a staggered grid, which leads to the Yee scheme in the interior. For the first two test cases we consider the propagation in vacuum with $\varepsilon = \mu = 1$ and $\sigma = 0$. We first show the two dimensional problem of transverse electric waves, since this allows us to compute with finer mesh sizes and thus to illustrate our asymptotic results by numerical experiments. We simulate directly the error equations, $f = 0$, on a uniform mesh with mesh parameter h, and we use a random initial guess to ensure that all the frequency components are present in the iteration.

We consider the transverse electric waves problem (TE) in the plane $(x, y, 0)$. There is no more dependence on z and the components E_3, H_1 H_2 are identically zero. The problem obtained is formally identical to the three-dimensional case (3.3). All the analytical results remain valid, we only need to replace $|\boldsymbol{k}|$ by $|k_y|$, and the corresponding quantities in the optimized parameters for both time-harmonic and time-discretized solutions. We solve Maxwell's equations on the unit square $\Omega = (0, 1)^2$ with an 0 order approximation of absorbing boundary conditions at $\partial\Omega$, decomposed into the two subdomains $\Omega_1 = (0, \beta) \times (0, 1)$ and $\Omega_2 = (\alpha, 1) \times (0, 1)$, where $0 < \alpha \leq \beta < 1$, and therefore the overlap is $L = \beta - \alpha$, and we consider both decompositions with and without overlap.

In the time-harmonic case, the frequency $\tilde{\omega} = 2\pi$ is chosen such that the rule of thumb of 10 points per wavelength is not violated. Table 3.2 shows the iteration count to achieve a relative residual reduction of 10^{-6} for all Schwarz algorithms we considered, in the overlapping and non-overlapping case. The results are presented in the form $it_S(it_{GM})$, where it_S denotes the iteration number for the iterative version of the algorithm and it_{GM} the iteration

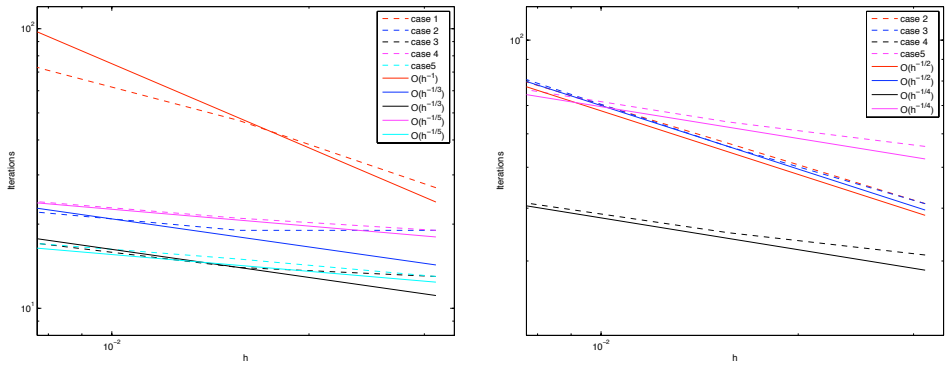


Figure 3.1: Asymptotics for the overlapping (left) and non-overlapping (right) cases for the time harmonic equations.

number for the accelerated version using GMRES, and a dash means no convergence. In Figure 3.1 we show the results we obtained in a graph, together with the expected asymptotics. Both on the left in the overlapping case and on the right in the non-overlapping one, the asymptotics agree quite well, except for the classical case with overlap, where the algorithm performs better than predicted by the asymptotic analysis.

We apply now the previous principles to derive an efficient domain-decomposition method based on optimized interface conditions to solve a realistic application: heating up a chicken in a micro-wave oven, see Figure 3.2 on the left. The computational domain is now given by the heating cavity of a Whirlpool Talent Combi 4 microwave oven, $\Omega = [0, 0.32] \times [0, 0.36] \times [0, 0.20]$ meters. We impose metallic boundary conditions (which means a null tangential

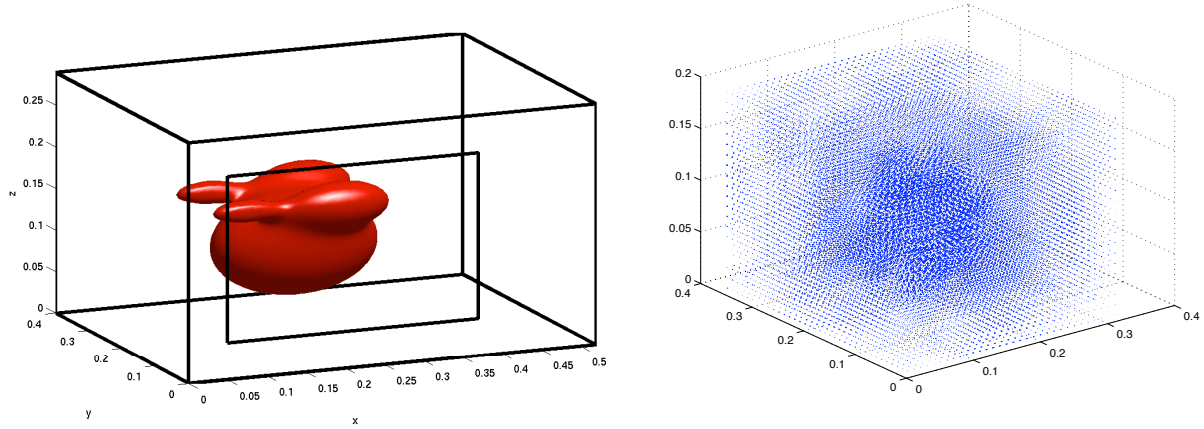


Figure 3.2: Chicken in a Whirlpool Talent Combi 4 micro-wave oven on the left, and real part of the magnetic field in the cooking cavity while heating the chicken on the right.

electric field) on all faces except on the right of the oven, where the components of the electric field are the dominant TE10 mode generated by the magnetron on a small rectangle

of dimensions 0.08×0.04. The electric and electromagnetic properties of the media are now non-constant in the computational domain: inside the chicken, we have an electric permittivity $\varepsilon = 4.43 \cdot 10^{-11} \frac{\text{Farads}}{\text{m}}$ and the conductivity is $\sigma = 3 \cdot 10^{-11} \frac{\text{Siemens}}{\text{m}}$, whereas for the air $\varepsilon = 8,85 \cdot 10^{-12} \frac{\text{Farads}}{\text{m}}$ and $\sigma = 0 \frac{\text{Siemens}}{\text{m}}$. The magnetic permeability is the same for both, $\mu = 4\pi \cdot 10^{-7} \frac{\text{Henry}}{\text{m}}$, and the frequency is given by $\omega = 2\pi \cdot 2.45$ GHz.

We decompose the microwave oven into $2 \times 2 \times 2 = 8$ subdomains of equal size on a grid with mesh size $h = 0.005$, which allows us to solve this problem on a PC, where a direct factorization would not have been possible any more. This resolution is enough for the wavelength of the microwave, and also for most of the geometry of the chicken, except maybe for the drumstick tips. The real part of the magnetic field of the solution is shown in Figure 3.2 on the right, and the intensity (Euclidian norm) of the electric and magnetic field in the oven are shown in Figure 3.3 in three dimensions.

We have shown that for Maxwell's equations, a classical Schwarz algorithm using char-

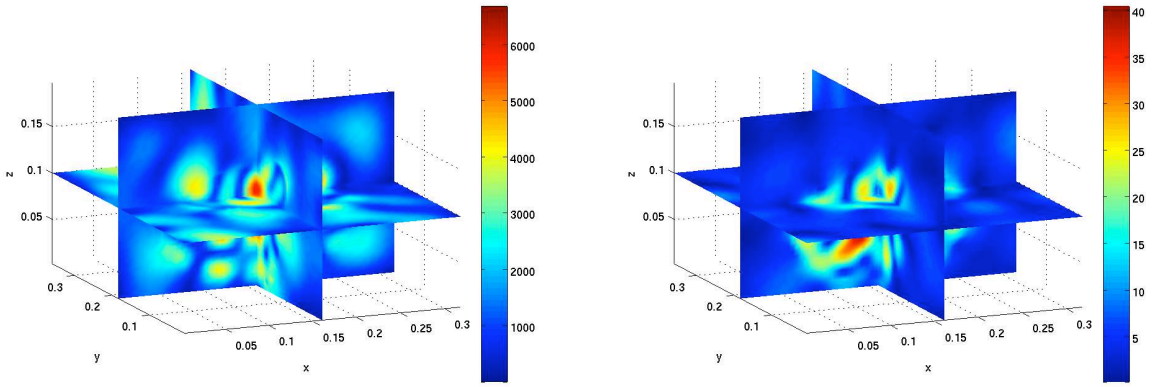


Figure 3.3: Chicken heating in a microwave oven: electric field intensity on the left, and magnetic field intensity on the right.

acteristic Dirichlet transmission conditions between subdomains has the same convergence behavior as a simple optimized Schwarz method applied to the Helmholtz equation, with a low frequency approximation of the optimal transmission conditions. This relation allowed us to develop easily an entire hierarchy of optimized overlapping and non-overlapping Schwarz methods with better transmission conditions than the characteristic ones for Maxwell's equations. We illustrated with numerical experiments that the new algorithms converge much more rapidly than the classical one, and that such algorithms can be effectively used to compute an approximate solution for a large scale application. This latter problem contains a positive conductivity, variable coefficients and multiple subdomains, a case which is not covered by our current analysis. Nevertheless, the algorithm performs well with the coefficients derived from the zero conductivity, constant coefficient case. We are currently studying the optimization problem with non-zero conductivity, for which the equivalence with the Helmholtz equation does not hold any longer.

3.2 A domain decomposition method for solving the three-dimensional time-harmonic Maxwell's equations discretized by discontinuous Galerkin methods

Our objective is the design and application of optimized Schwarz algorithms in conjunction with discontinuous Galerkin methods. The first step in this direction is understanding and analyzing classical overlapping and non-overlapping Schwarz algorithms in the discrete framework of these methods. To our knowledge, except in Helluy [Hel94], where such an algorithm is applied to a discretization of the first-order time-harmonic Maxwell equations by an upwind finite volume method, no other attempts for higher order discontinuous Galerkin methods or different kind of fluxes can be found in the literature. A classical domain decomposition strategy is adopted in this study which takes the form of a Schwarz-type algorithm where Després type conditions [DJR92] are imposed at the interfaces between neighboring subdomains. These translate as Dirichlet transmission conditions for the incoming caracteristic variables in the case of the first order Maxwell system. A similar approach (using Robin transmission conditions) but in the case of a second order system and in conjuction with a non-conforming finite element discretization is presented in [LVL05] and [VCL06]. The use of higher order or optimized transmission conditions is the natural prolongation of the present work and this has already been done in the two-dimensional case in [18]. Secondly, this allows an easier treatement of heterogeneous media. A multifrontal sparse direct solver is used at the subdomain level. The resulting domain decomposition strategy can be viewed as a hybrid iterative/direct solution method for the large, sparse and complex coefficients algebraic system resulting from the discretization of the time-harmonic Maxwell equations by a discontinuous Galerkin method.

In the following, the formalism, the discrete and the continuous context are the same as in section 2.1. We introduce directly the domain decomposition part, without recalling the underlying equations.

In order to ease the presentation we decompose the domain Ω into two overlapping or non-overlapping subdomains Ω_1 and Ω_2 but the extension of the formulation of the method to any number of subdomains is straightforward. We define $\Gamma_{12} = \partial\Omega_1 \cap \Omega_2$ and $\Gamma_{21} = \partial\Omega_2 \cap \Omega_1$. In the following we denote by $\boldsymbol{n}_{ij}$ the outward normal vector to the interface Γ_{ij} with i, j in $\{1, 2\}$. We solve system (2.13) in both subdomains and we enforce on the subdomain interfaces the continuity of the incoming characteristic variables which provides a so-called *classical Schwarz algorithm*. The classical Schwarz algorithm allows to compute the $(n+1)$-th iterate of the solution from the n-th iterate, starting from an arbitrary initial guess, by solving local problems and then exchanging information between artificial boundaries, called

interfaces. This algorithm is given by:

$$\begin{cases} i\omega\, \boldsymbol{W}^{1,n+1} + \displaystyle\sum_{l\in\{x,y,z\}} G_l \partial_l \boldsymbol{W}^{1,n+1} = 0 \text{ in } \Omega_1, \\ G^-_{\boldsymbol{n}_{12}} \boldsymbol{W}^{1,n+1} = G^-_{\boldsymbol{n}_{12}} \boldsymbol{W}^{2,n} \text{ on } \Gamma_{12}, \\ + \text{ Boundary conditions on } \partial\Omega_1 \cap \partial\Omega, \end{cases} \quad \begin{cases} i\omega\, \boldsymbol{W}^{2,n+1} + \displaystyle\sum_{l\in\{x,y,z\}} G_l \partial_l \boldsymbol{W}^{2,n+1} = 0 \text{ in } \Omega_2, \\ G^-_{\boldsymbol{n}_{21}} \boldsymbol{W}^{2,n+1} = G^-_{\boldsymbol{n}_{21}} \boldsymbol{W}^{1,n} \text{ on } \Gamma_{21}, \\ + \text{ Boundary conditions on } \partial\Omega_2 \cap \partial\Omega, \end{cases} \tag{3.24}$$

where subscripts denote components, and superscripts denote the subdomain number and the iteration count. This algorithm has been analyzed in [20] and its convergence rate has been computed in the case of an infinite domain $\Omega = \mathbb{R}^3$.

The subproblems of the Schwarz algorithm (3.24) are discretized using a discontinuous Galerkin formulation. Let us now assume that the domain Ω is decomposed into N_s subdomains $\Omega = \bigcup_{i=1}^{N_s} \Omega_i$. A superscript i indicates that some notations are relative to the subdomain Ω_i and not to the whole domain Ω. Thus, we will refer to $\mathcal{T}_h^i$ and V_h^i with obvious definitions from those of $\mathcal{T}_h$ and V_h and we also define $\Gamma_m^i = \Gamma_m \cap \partial\Omega_i$, $\Gamma_a^i = \Gamma_a \cap \partial\Omega_i$ and $\Gamma_0^i = \Gamma_0 \cap \partial\Omega_i$ with their corresponding sets of faces $\Gamma^{m,i}$, $\Gamma^{a,i}$ and $\Gamma^{0,i}$. Finally Γ^{ij} will denote the set of faces which belongs to $\Gamma_{ij} = \partial\Omega_i \cap \Omega_j$. According to algorithm (3.24), the interface condition on Γ_{ij} writes as:

$$G^-_{\boldsymbol{n}_F}(\boldsymbol{W}_h^{i,n+1} - \boldsymbol{W}_h^{j,n}) = 0 \quad \text{for all } F \text{ belonging to } \Gamma^{ij},$$

where $\boldsymbol{W}_h^{i,n+1}$ denotes the approximation of $\boldsymbol{W}^{i,n+1}$ for $i = 1, 2$. Thus, the discontinuous Galerkin discretization of a local problem of algorithm (3.24) can be written as the solution of the following problem:

$$\begin{cases} \text{Find } \boldsymbol{W}_h^{i,n+1} \text{ in } V_h^i \times V_h^i \text{ such that:} \\ \displaystyle\int_{\Omega_h^i} \left(\mathrm{i}\omega G_0 \boldsymbol{W}_h^{i,n+1}\right)^t \overline{\boldsymbol{V}} dv + \sum_{K\in\mathcal{T}_h^i} \int_K \left(\sum_{l\in\{x,y,z\}} G_l \partial_l (\boldsymbol{W}_h^{i,n+1}) \right)^t \overline{\boldsymbol{V}} dv \\ + \displaystyle\sum_{F\in\Gamma^{m,i}} \int_F \left(\frac{1}{2} (M_{F,K} - I_{FK} G_{\boldsymbol{n}_F}) \boldsymbol{W}_h^{i,n+1} \right)^t \overline{\boldsymbol{V}} ds \\ + \displaystyle\sum_{F\in(\Gamma^{a,i}\cup\Gamma^{ij})} \int_F \left(I_{FK} G^-_{\boldsymbol{n}_F} \boldsymbol{W}_h^{i,n+1} \right)^t \overline{\boldsymbol{V}} ds \\ - \displaystyle\sum_{F\in\Gamma^{0,i}} \int_F \left(G_{\boldsymbol{n}_F} [\![\boldsymbol{W}_h^{i,n+1}]\!] \right)^t \{\overline{\boldsymbol{V}}\} ds \\ + \displaystyle\sum_{F\in\Gamma^{0,i}} \int_F \left(S_F [\![\boldsymbol{W}_h^{i,n+1}]\!] \right)^t [\![\overline{\boldsymbol{V}}]\!] ds \\ = \displaystyle\sum_{F\in\Gamma^{a,i}} \int_F \left(I_{FK} G^-_{\boldsymbol{n}_F} \boldsymbol{W}^{\mathrm{inc}} \right)^t \overline{\boldsymbol{V}} ds \\ + \displaystyle\sum_{F\in\Gamma^{ij}} \int_F \left(I_{FK} G^-_{\boldsymbol{n}_F} \boldsymbol{W}^{j,n} \right)^t \overline{\boldsymbol{V}} ds, \qquad \forall \boldsymbol{V} \in V_h^i \times V_h^i. \end{cases} \tag{3.25}$$

In the two-domain case the Schwarz algorithm can be written formally as follows:

$$\begin{cases} \mathcal{L}\boldsymbol{W}^{1,n+1} = \boldsymbol{f}^1, \text{ in } \Omega_1, \\ \mathcal{B}_1(\boldsymbol{W}^{1,n+1}) = \lambda^{1,n}, \text{ on } \Gamma_{12}, \\ + \text{ Boundary conditions on } \partial\Omega_1 \cap \partial\Omega, \end{cases} \tag{3.26}$$

$$\begin{cases} \mathcal{L}\boldsymbol{W}^{2,n+1} = \boldsymbol{f}^2, \text{ in } \Omega_2, \\ \mathcal{B}_2(\boldsymbol{W}^{2,n+1}) = \lambda^{2,n}, \text{ on } \Gamma_{21}, \\ + \text{ Boundary conditions on } \partial\Omega_2 \cap \partial\Omega, \end{cases}$$

and then:

$$\begin{cases} \lambda^{1,n+1} = \mathcal{B}_1(\boldsymbol{W}^{2,n+1}) \text{ on } \Gamma_{12}, \\ \lambda^{2,n+1} = \mathcal{B}_2(\boldsymbol{W}^{1,n+1}) \text{ on } \Gamma_{21}, \end{cases} \tag{3.27}$$

where $\mathcal{L}$ is a linear differential operator, $\boldsymbol{f}^{1,2}$ denotes right hand sides associated to $\Omega_{1,2}$ and, $\mathcal{B}_1$ and $\mathcal{B}_2$ are the interface operators. The Schwarz algorithm (3.26)-(3.27) can be rewritten as:

$$\begin{cases} \lambda^{1,n+1} = \mathcal{B}_1(\boldsymbol{W}^2(\lambda^{2,n}, \boldsymbol{f}^2)), \\ \lambda^{2,n+1} = \mathcal{B}_2(\boldsymbol{W}^1(\lambda^{1,n}, \boldsymbol{f}^1)), \end{cases}$$

where $\boldsymbol{W}^j = \boldsymbol{W}^j(\lambda^j, \boldsymbol{f}^j)$ are the solution of the local problems. By linearity of the operators involved, an iteration of the Schwarz algorithm is equivalent to:

$$\boldsymbol{\lambda}^{n+1} = (\mathrm{Id} - \mathcal{T})\boldsymbol{\lambda}^n + \boldsymbol{d},$$

which is a fixed point iteration to solve the interface system:

$$\mathcal{T}\boldsymbol{\lambda} = \boldsymbol{d}, \tag{3.28}$$

where $\boldsymbol{\lambda} = (\boldsymbol{\lambda}^1, \boldsymbol{\lambda}^2)$. From the discrete point of view, the global problem on domain Ω can be written in the matrix form:

$$\begin{pmatrix} A_1 & 0 & R_1 & 0 \\ 0 & A_2 & 0 & R_2 \\ 0 & -B_2 & \mathrm{I} & 0 \\ -B_1 & 0 & 0 & \mathrm{I} \end{pmatrix} \begin{pmatrix} \boldsymbol{W}_h^1 \\ \boldsymbol{W}_h^2 \\ \boldsymbol{\lambda}_h^1 \\ \boldsymbol{\lambda}_h^2 \end{pmatrix} = \begin{pmatrix} \boldsymbol{f}_h^1 \\ \boldsymbol{f}_h^2 \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix},$$

where $A_{1,2}$ are local matrices coupling only internal unknowns, $R_{1,2}$ express the coupling between internal unknowns and interface unknowns and the subscript h denotes the discrete counterpart of a given quantity (*e.g.* $\boldsymbol{\lambda}_h^{1,2}$ are the discretized unknown vectors corresponding to $\boldsymbol{\lambda}^{1,2}$). The elimination of the internal unknowns $\boldsymbol{W}_h^{1,2}$ leads to the discrete counterpart of the interface problem (3.28), $\mathcal{T}_h\boldsymbol{\lambda}_h = \boldsymbol{d}_h$, with:

$$\mathcal{T}_h = \begin{pmatrix} \mathrm{I} & B_2A_2^{-1}R_2 \\ B_1A_1^{-1}R_1 & \mathrm{I} \end{pmatrix} \quad \text{and} \quad \boldsymbol{g}_h = \begin{pmatrix} B_2A_2^{-1}\boldsymbol{f}_h^2 \\ B_1A_1^{-1}\boldsymbol{f}_h^1 \end{pmatrix},$$

where $\mathcal{T}_h$ and $\boldsymbol{g}_h$ are the discretization of $\mathcal{T}$ and $\boldsymbol{d}$. This system is further solved by a Krylov subspace method.

For this study, the implementation of the discontinuous Galerkin formulations has been limited to a $\mathbb{P}_0$ approximation with the centered flux (which is equivalent to a finite volume method which will be referred as DG-$\mathbb{P}_0$-c in the sequel) and a $\mathbb{P}_1$ approximation (*i.e.* a linear discontinuous Galerkin method) with either the centered flux or the upwind flux and nodal polynomial basis functions (respectively referred as DG-$\mathbb{P}_1$-c and DG-$\mathbb{P}_1$-u in the sequel).

Unless otherwise indicated, computations have been performed in 64 bit arithmetic. The experimental testbed is a cluster of AMD Opteron 2 GHz dual nodes with 2 GB of RAM memory, interconnected by a Gigabit Ethernet switch. The computer codes for the DG-$\mathbb{P}_0$ and DG-$\mathbb{P}_1$ methods have been programmed in Fortran and the parallelization relies on the MPI (Message Passing Interface). The implementation of the domain decomposition solver requires a partitioning of the underlying tetrahedral mesh which is obtained using the MeTiS graph partitioning tool [KK99].

An unpreconditioned BiCGstab(ℓ) Krylov subspace method [SF93] is used for the solution of the interface system (3.28). After different tests for assessing the convergence of the method and the associated computation time, the parameter ℓ has been set to 6. This method is adapted to linear systems involving non-symmetric matrices with complex spectrum. The convergence of the iterative solution of the interface system is evaluated in terms of the euclidian norm of the residual normalized to the norm of the right-hand side vector. The corresponding linear threshold has been set to $\varepsilon_i = 10^{-6}$. Each iteration of this Krylov subspace method requires a certain number of matrix-vector products with the interface matrix of system (3.28). Within the domain decomposition framework of algorithm (3.24), such a matrix-vector product translates into the solution of the subdomain discrete problems (3.25). For this purpose, several strategies have been considered:

- a preconditioned restarted GMRES(m) [SS86] (with $m = 10$) or a preconditioned BiCGstab(ℓ) (with $\ell = 1$) method where the preconditioner is taken to be a LU factorization computed and stored in single precision arithmetic using the MUMPS multifrontal sparse direct solver [ADL00], while the Krylov subspace method works on double precision arithmetic vectors. In both cases, the linear threshold has been set to $\varepsilon_i = 10^{-6}$. These solution strategies will be referred respectively as `DD-gmres` and `DD-bicgl`.

- a LU factorization where the L and U factors are computed and stored in single precision (32 bits) arithmetic and an iterative refinement procedure is applied to recover double precision arithmetic (64 bits). More precisely, assuming that the linear system is $Ax = b$, the iterative refinement procedure is as follows:

  $x \leftarrow 0$

  REPEAT

  $r \leftarrow b - Ax$ % residual evaluation step.

Solve $Ly = r$

Solve $Uz = y$

$x \leftarrow x + z$ % updating step.

UNTIL $\| r \| < \varepsilon_l$

where the triangular solves $Ly = r$ and $Uz = y$ are performed using single precision arithmetic while the residual evaluation and updating step are computed in double precision arithmetic. In practice, we set $\varepsilon_l = 10^{-10}$ and a maximum of five iterations of the above procedure. In the sequel, this solution strategy will be referred as `DD-itref`.

These strategies have been selected with the aim to reduce the memory requirements for storing the L and U factors and thus allowing to tackle large problems. We note that such mixed-precision strategies have recently been considered in the linear algebra community essentially for performance issues [KD06, LLL+06] on modern high-performance processors. In these works, the mixing of single and double precision computations is performed in the context of an iterative refinement procedure. Here, the single precision L and U factors yield a very accurate preconditioner and consequently, a few iterations of the preconditioned Krylov subspace methods are sufficient for solving the subdomain problems. In practice we use one iteration of BiCGstab and two iterations of GMRES.

In the following tables and figures:

- $\mathrm{L_{min}}$, Lmax and $\mathrm{L_{avg}}$ respectively denote the minimum, maximum and average length of an edge in a given tetrahedral mesh,
- N_s is the number of subdomains which is also the number of processes involved in a parallel simulation,
- 'CPU' is the CPU time which is evaluated on each process of a parallel simulation and, for this reason, we give both the minimum and maximum values of this quantity,
- 'REAL' is the real (or elapsed) time of a parallel simulation,
- 'RAM' is the memory requirement for storing the L and U factors which is evaluated on each process of a parallel simulation and, as for the 'CPU' quantity, we give both the minimum and maximum values of this quantity.

Extensive numerical results can be found in [16]. We we will limit here only to the presentation of the conclusions as well as a bioelectromagnetics application. A first noticeable behaviour that can be emphasized is that the convergence of the proposed domain decomposition solver for a given approximation method is weakly dependent on the number of system unknowns and the granularity of the decomposition (*i.e.* the number of subdomains). For instance, in the case of the diffraction of a plane wave by a PEC sphere, when the number of unknowns of the algebraic system associated to the DG-$\mathbb{P}_0$-c approximation method increases from $6 \times 384,000 = 2,304,000$ to $6 \times 1,382,400 = 8,294,400$, the number of iterations of the domain decomposition solver ranges from 8 to 10 for a number of subdomains

N_s in the set $\{16, 32, 64\}$. Similarly, for the diffraction of a plane wave by a PEC cube, when the number of unknowns of the algebraic system associated to the DG-$\mathbb{P}_0$-c approximation method increases from $6 \times 156,000 = 936,000$ to $6 \times 725,424 = 4,352,544$, the number of iterations of the domain decomposition solver ranges from 6 to 9 for a number of subdomains N_s in the set $\{16, 32, 64\}$.

Moreover, for a given mesh, the number of iterations increases when switching between the DG-$\mathbb{P}_0$ and DG-$\mathbb{P}_1$ approximation methods. This is mainly related to the fact that at the same time the number of system unknowns increases noticeably. For instance, when simulating the diffraction of a plane wave by a PEC cube, the number of system unknowns is equal to $6 \times 156,000 = 936,000$ and $24 \times 156,000 = 3,744,000$ respectively for the DG-$\mathbb{P}_0$ and DG-$\mathbb{P}_1$ approximation methods. In the former case, the number of iterations is equal to 6 for $N_s = 16$ while in the latter case, it is equal to 9 (respectively 10) for $N_s = 32$ (respectively 64).

It is also worthwhile to note that:

- the convergence of the domain decomposition solver when combined to the DG-$\mathbb{P}_1$ approximation method seems insensitive to the type of scheme (*i.e.* centered or upwind) used for the evaluation of the numerical flux through internal faces.

- as expected, when the propagation media is more complex than a simple uniform (*i.e.* homogeneous) material, the convergence of the domain decomposition solver requires more iterations. Indeed, comparing the performances of the domain decomposition solver combined to the DG-$\mathbb{P}_1$ approximation method for the simply PEC and coated PEC cubes, the number of iterations increases from 9 to 14.

We evaluate the parallel performances of the proposed domain decomposition solver using two metrics: the ratio of the maximum of the per process CPU times to the REAL time which is referred as '%CPU' in the sequel and, the relative parallel speedup $S_{N_{s1}}^{N_{s2}}$ evaluated as the ratio of the elapsed time for N_{s1} subdomains to the elapsed time for N_{s2} subdomains. First, we remark that, in the case of the diffraction of a plane wave by a PEC sphere, %CPU ranges from 58% to 97%. One can note that super-linear parallel speedups are often obtained. This behaviour essentially stems from the super-linear reduction of the cost of the local solves when increasing the number or subdomains for a constant global problem size.

On the other hand, it is equally important to observe that although the MeTiS partitioning tool almost always yields well balanced partitions (in the present case, the load balance is evaluated in terms of the local number of tetrahedra), there is a noticeable disparity in the required amount of RAM for storing the subdomain L and U factors, especially for large values of the number of subdomains N_s. As a matter of fact, the fill-in of the L and U factors is influenced by several factors among which, the presence in the original matrix of diagonal blocks related to physical boundaries (metallic wall, absorbing boundary) which in turn has effects on the numerical pivoting strategy. But, above all, the partitioning of a mesh (in practice, the adjacency graph associated to the mesh) using a tool such as MeTiS, is dictated by two main criteria, namely the minimization of the subdomains separator and the achievement of a well balanced computational load, while a balance of the fill-in is rarely

an objective. In the present case, the main consequence of this load unbalance in the fill-in of the local L and U factors is a potentially large gap between the minimum and maximum CPU times for the subdomain triangular solves, which results in non-negligible idle times across processes between each iteration of the interface system solver.

We conclude with the application of the proposed numerical methodology to the simulation of a time-harmonic electromagnetic wave propagation problem in an irregularly shaped and heterogeneous medium. The problem under consideration is concerned with the propagation of a plane wave in realistic geometrical models of head tissues. It is a first step towards the development of a computational framework for the numerical dosimetry of electromagnetic fields radiated by mobile phones. Starting from MR images of the Visible Human 2.0 project [RHGJ03], head tissues are segmented and the interfaces of a selected number of tissues (namely, the skin, the skull and the brain) are triangulated. Different strategies can be used in order to obtain a smooth and accurate segmentation of head tissues and interface triangulations as well. A first strategy consists in using a marching cube algorithm [LC87] which leads to huge triangulations of interfaces between segmented subdomains. These triangulations can then be regularized, refined and decimated in order to obtain reasonable surface meshes, for example using the YAMS [Fre03] re-meshing tool. Another strategy consists in using a variant of Chew's algorithm [Che93], based on Delaunay triangulation restricted to the interface, which allows to control the size and aspect ratio of interfacial triangles [BO05]. Surface meshes of increased resolution resulting from such a procedure are presented on Figure 3.4. Then, these triangulated surfaces together with a triangulation of the artificial boundary (absorbing boundary) of the overall computational domain, which is taken here to be a sphere, are used as inputs for the generation of volume meshes. In this study, the GHS3D tetrahedral mesh generator [GHS91] is used to mesh volume domains between the various interfaces. Two tetrahedral meshes have been used whose characteristics are summarized in Table 3.3. The frequency of the incident plane wave is F=1800 MHz and its polarization is such that:

$$\boldsymbol{k} = (k_x, 0, 0)^t \quad , \quad \boldsymbol{E} = (0, 0, E_z)^t \quad \text{and} \quad \boldsymbol{H} = (0, H_y, 0)^t.$$

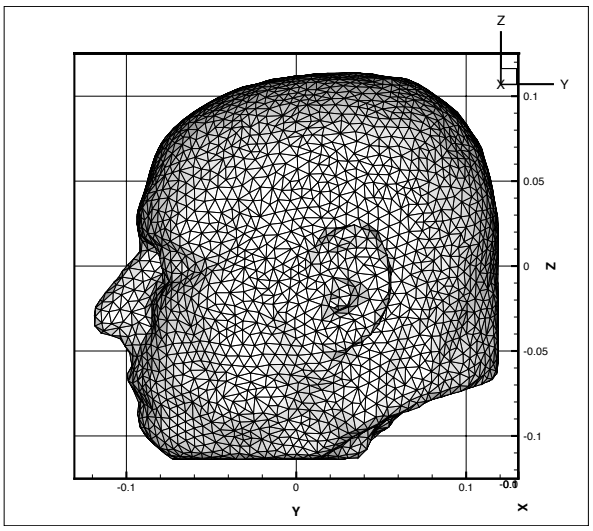
Albeit this propagation problem clearly involves irregularly shaped domains and non-uniform tetrahedral meshes, it is yet a simplified configuration with regards to the simulations usually used in numerical dosimetry studies of human exposition to mobile phone radiation [**?**], for two reasons: a mobile phone geometrical model has not been taken into account in the present simulation setting and, the electromagnetic parameters of the materials are set to artificial values for the purpose of exemplifying the characteristics of the propagation of the plane wave in the head tissues (null conductivity, $\varepsilon_r = 4.0$ for the brain, $\varepsilon_r = 6.5$ for the cerebrospinal fluid, $\varepsilon_r = 1.5$ for the skull and $\varepsilon_r = 4.0$ for the skin). For the computations reported here, the methods DG-$\mathbb{P}_1$-c and DG-$\mathbb{P}_1$-u are used in conjunction with mesh M1 while method DG-$\mathbb{P}_0$-c is used with mesh M2. Moreover, this problem has also been simulated using a DGTD-$\mathbb{P}_1$-c (Discontinuous Galerkin Time-Domain) method [FLLP05] and the corresponding result will be considered here as the reference solution. The contour lines of E_z in various configurations are visualized on Figures 3.6 to 3.9. As can

be seen on these figures, on one hand, there is a good agreement between the results of the time-domain and time-harmonic computations and, on the other hand, the DG-$\mathbb{P}_1$ methods used with the coarsest mesh yield solutions which are closer to the reference computation than the one resulting from the DG-$\mathbb{P}_0$-c method applied on the finest geometrical model.

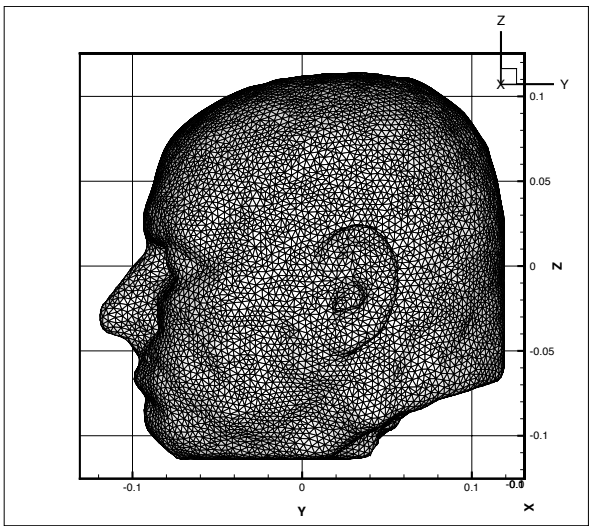
Performance results are given in Tables 3.4 and 3.5. In addition, the convergence curves for the iterative solution of the interface system (3.28) using the BiCGstab(ℓ) method are shown on Figure 3.5. Firstly, we note that the iterative solution requires 3 to 4 times more iterations than the numbers observed for the previous test cases, which is the consequence of the increased complexity in both the underlying discretization and the propagation medium. Secondly, the parallel efficiency, evaluated using the %CPU ratio, ranges from 65% to 75%. Here again, the load unbalance in the fill-in of the local L and U factors is the main reason for this parallel performance drop.

3.2.1 Conclusion

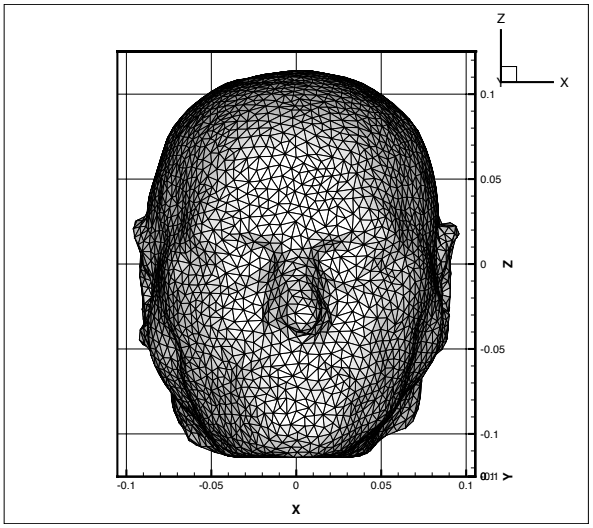
We have presented a hybrid iterative/direct solution method for the large, sparse and complex coefficients algebraic systems resulting from the discretization of the time-harmonic Maxwell equations by discontinuous Galerkin methods. The discretization in space relies on an unstructured tetrahedral mesh and as a result, the proposed numerical methodology is particularly well suited to the simulation of wave propagation problems in irregularly shaped media. Moreover, the local nature of a discontinuous Galerkin formulation allows for a natural treatment of heterogeneous media. Numerical and performance results reported here, albeit promising, have also raised a weakness in the current implementation of the domain decomposition method in the fact that the fill-in of the local L and U factors is generally not well balanced except for relatively simple problems (simply shaped domain, uniform mesh and homogeneous media). We believe that this drawback should be recurrent to almost all similar implementations of domain decomposition algorithms (*i.e.* based on exact factorization methods for the subdomain solves). This problem could be figure out by resorting to constrained level of fill-in subdomain solvers or/and by improving the quality of the mesh partitions (with regards to the resulting fill-in unbalance). In addition to these directions, our future works will be towards the improvement of the numerical efficiency of the Schwarz-type algorithm adopted in this study thanks to the design of discrete optimized interface conditions in the framework of our discontinuous Galerkin formulations on tetrahedral meshes.



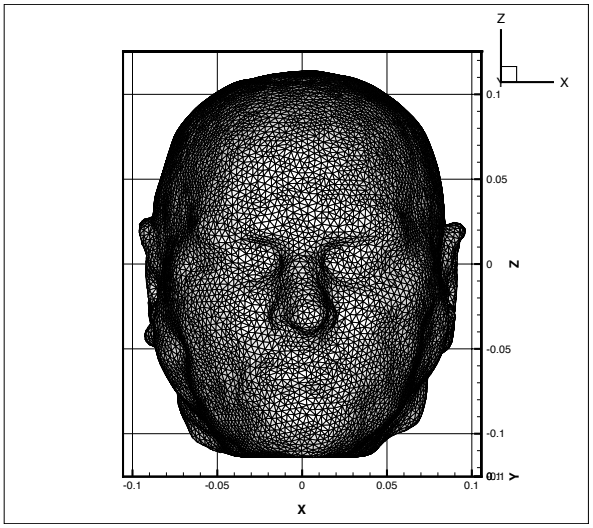
(a) Skin, lateral view, mesh M1.



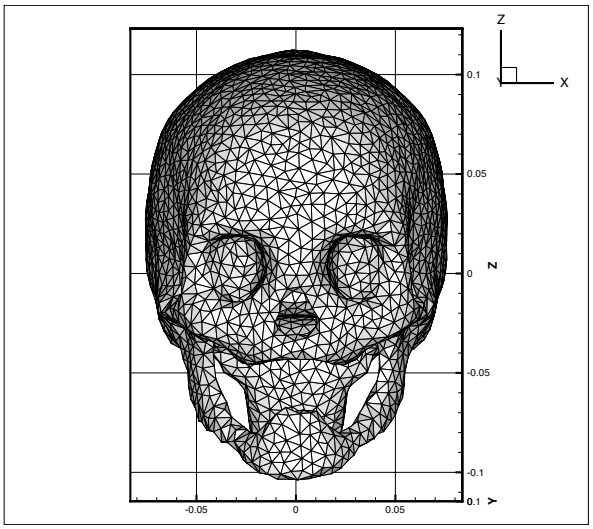
(b) Skin, lateral view, mesh M2.



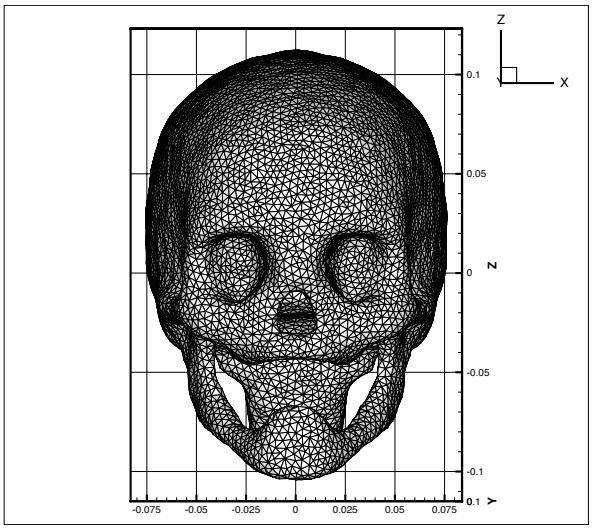
(c) Skin, frontal view, mesh M1.



(d) Skin, lateral view, mesh M2.



(e) Skull, frontal view, mesh M1.



(f) Skull, frontal view, mesh M2.

Figure 3.4: Propagation of a plane wave in a heterogeneous medium, F=1800 MHz. Triangulations of the skin and the skull.

Mesh	# vertices	# tetrahedra	L_{min} (m)	L_{max} (m)	L_{avg} (m)
M1	60,590	361,848	0.00185	0.04537	0.01165
M2	309,599	1,853,832	0.00158	0.02476	0.00693

Table 3.3: Propagation of a plane wave in a heterogeneous medium, F=1800 MHz. Characteristics of the tetrahedral meshes.

Mesh	Method	Strategy	N_s	# it	CPU (min/max)	REAL
M1	DG-$\mathbb{P}_1$-c	`DD-itref`	96	47	346 sec/466 sec	714 sec
-	DG-$\mathbb{P}_1$-u	`DD-itref`	96	46	347 sec/547 sec	765 sec
M2	DG-$\mathbb{P}_0$-c	`DD-itref`	96	33	228 sec/322 sec	428 sec

Table 3.4: Propagation of a plane wave in an heterogeneous medium, F=1800 MHz. Computation times (solution phase).

Mesh	Method	N_s	CPU (min/max)	RAM (min/max)	# dof
M1	DG-$\mathbb{P}_1$-c	96	64 sec/125 sec	640 MB/852 MB	8,684,352
-	DG-$\mathbb{P}_1$-u	96	80 sec/134 sec	633 MB/866 MB	8,684,352
M2	DG-$\mathbb{P}_0$-c	96	53 sec/ 98 sec	519 MB/684 MB	11,122,992

Table 3.5: Propagation of a plane wave in a heterogeneous medium, F=1800 MHz. Computation times and memory requirement for storing the L and U factors.

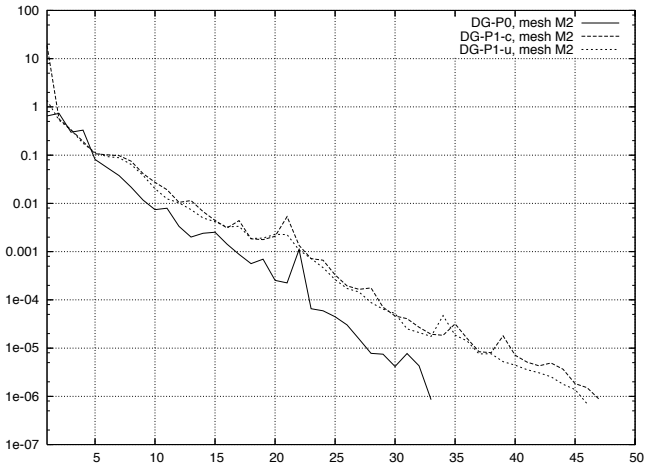
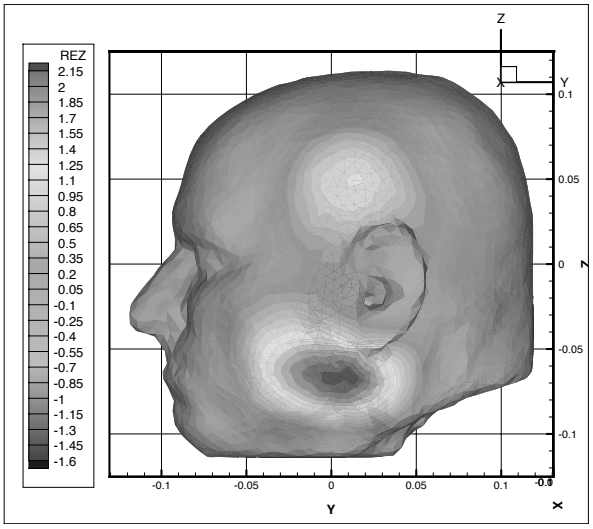
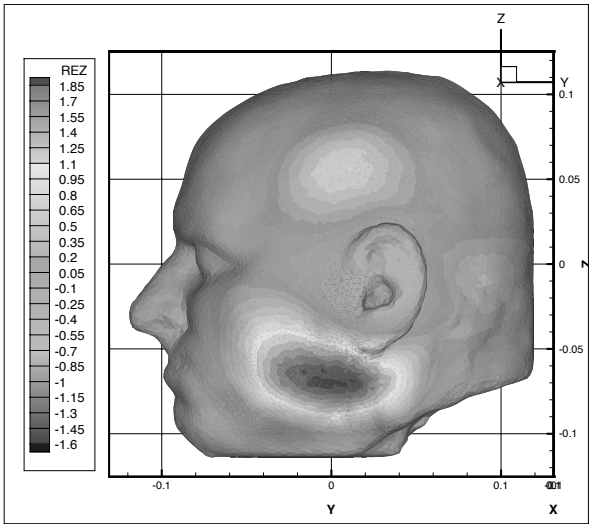


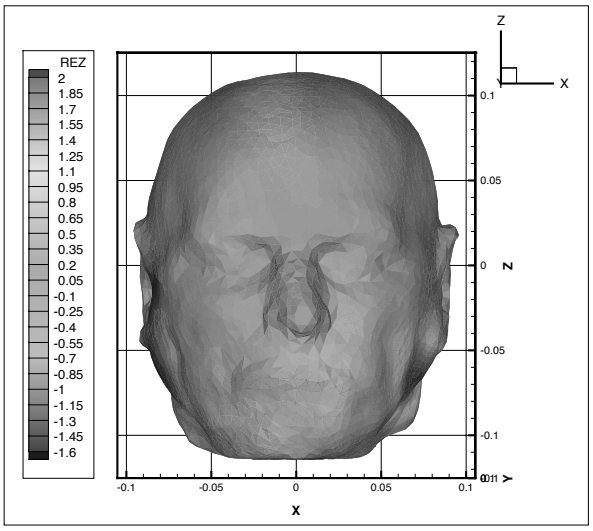
Figure 3.5: Propagation of a plane wave in a heterogeneous medium, F=1800 MHz. Iterative solution of the interface system.



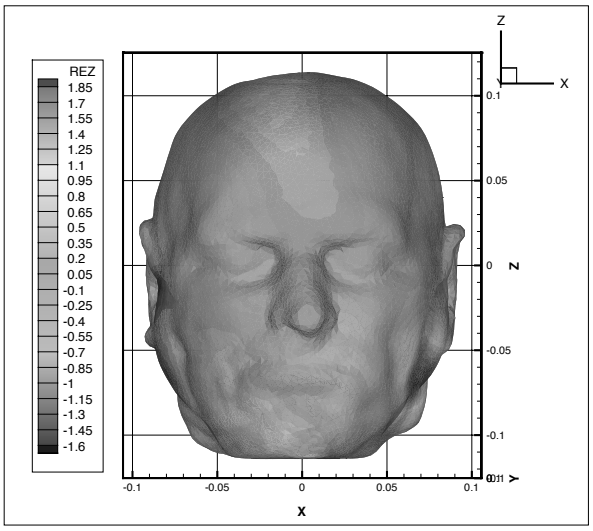
(a) method DG-$\mathbb{P}_1$-c, time-domain, skin, lateral view, mesh M1.



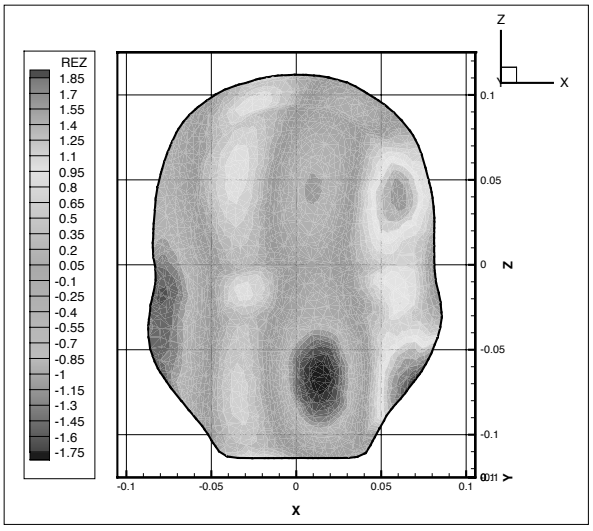
(b) method DG-$\mathbb{P}_0$-c, time-harmonic, skin, lateral view, mesh M2.



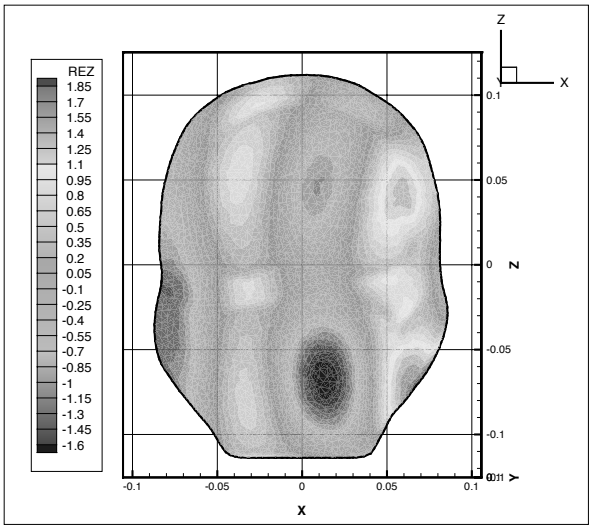
(c) method DG-$\mathbb{P}_1$-c, time-domain, skin, frontal view, mesh M1.



(d) method DG-$\mathbb{P}_0$-c, time-harmonic, skin, frontal view, mesh M2.

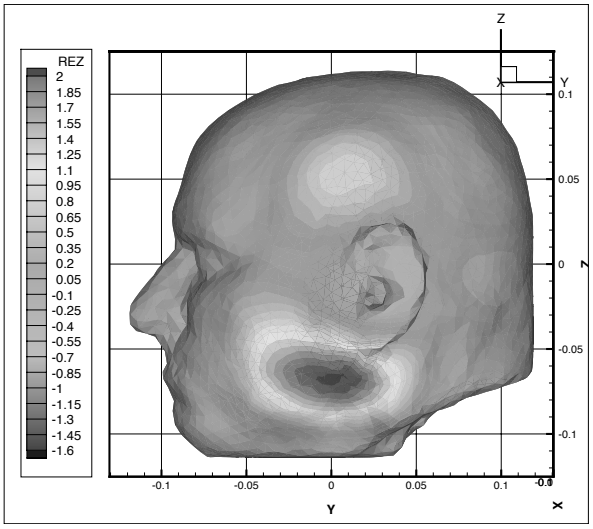


(e) method DG-$\mathbb{P}_1$-c, time-domain, view in the plane Y=0.0 m, mesh M1.

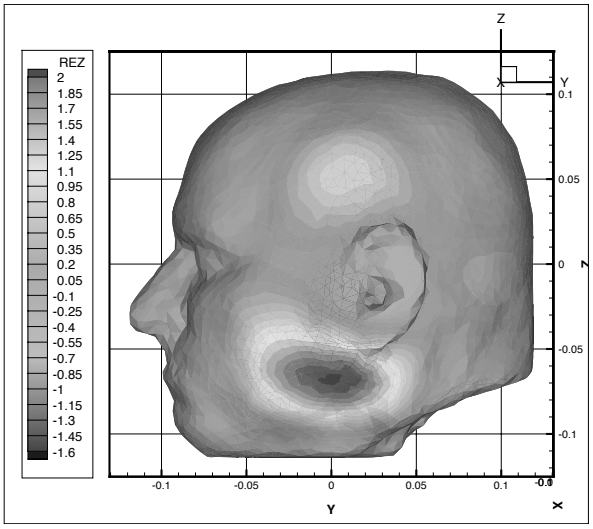


(f) method DG-$\mathbb{P}_0$-c, time-harmonic, view in the plane Y=0.0 m, mesh M2.

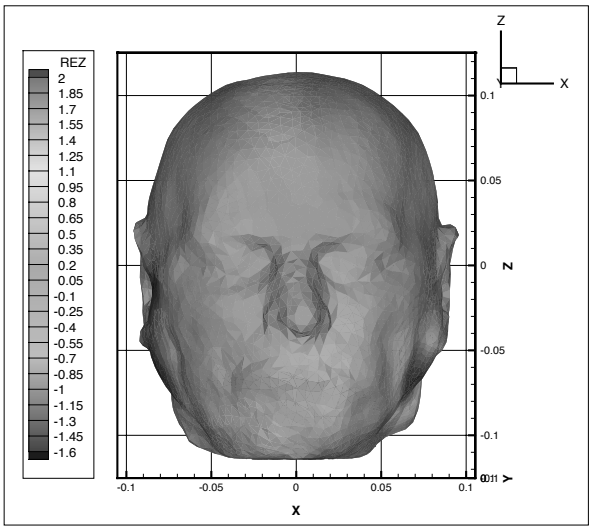
Figure 3.6: Propagation of a plane wave in a heterogeneous medium, F=1800 MHz



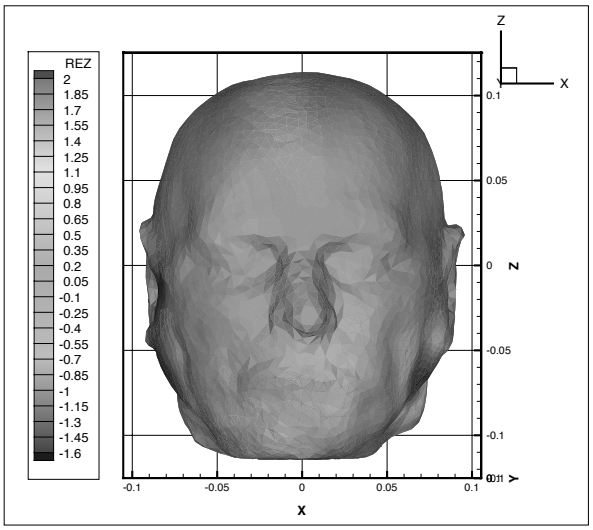
(a) method DG-$\mathbb{P}_1$-c, time-harmonic, skin, lateral view, mesh M1.



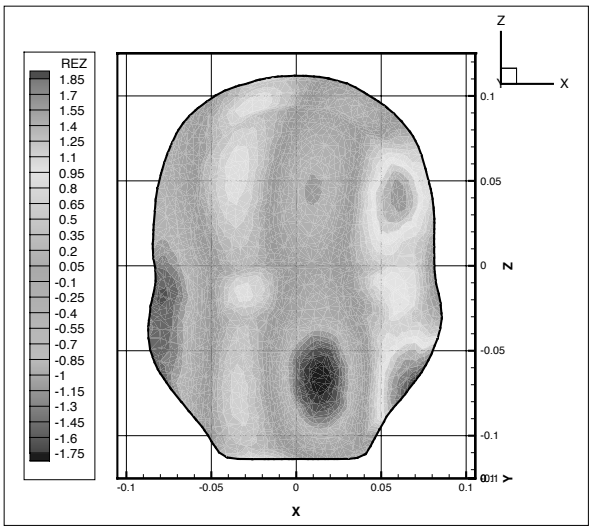
(b) method DG-$\mathbb{P}_1$-u, time-harmonic, skin, lateral view, mesh M1.



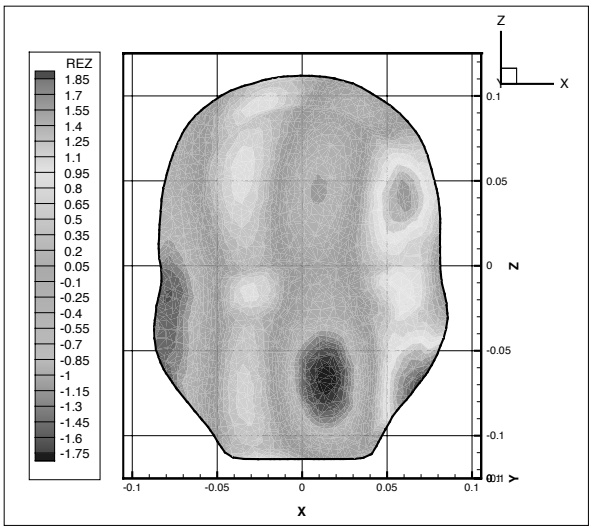
(c) method DG-$\mathbb{P}_1$-c, time-harmonic, skin, frontal view, mesh M1.



(d) method DG-$\mathbb{P}_1$-u, time-harmonic, skin, frontal view, mesh M1.

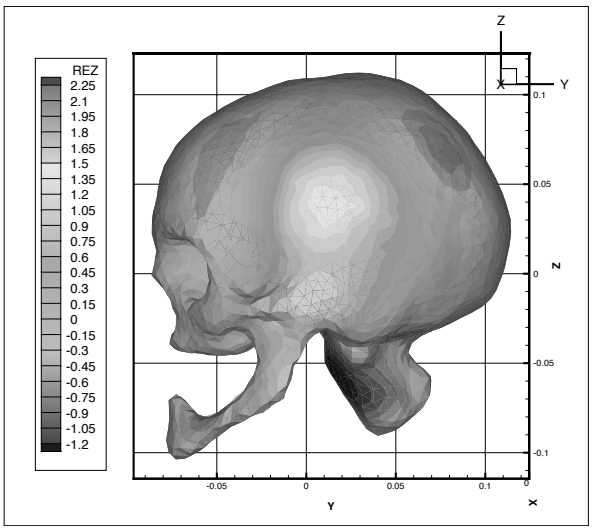


(e) method DG-$\mathbb{P}_1$-c, time-harmonic, view in the plane Y=0.0 m, mesh M1.

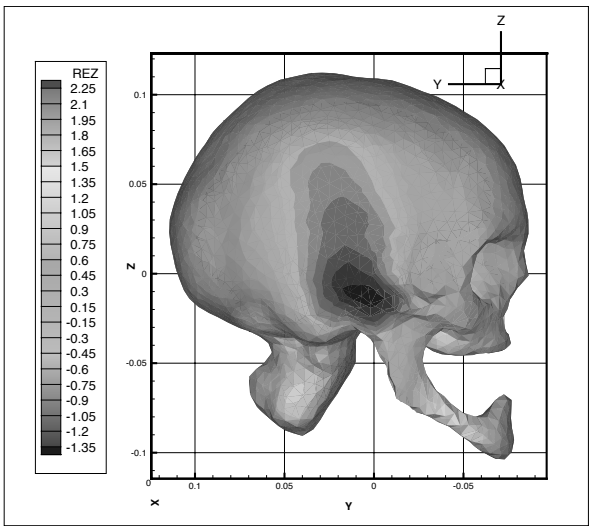


(f) method DG-$\mathbb{P}_1$-u, time-harmonic, view in the plane Y=0.0 m, mesh M1.

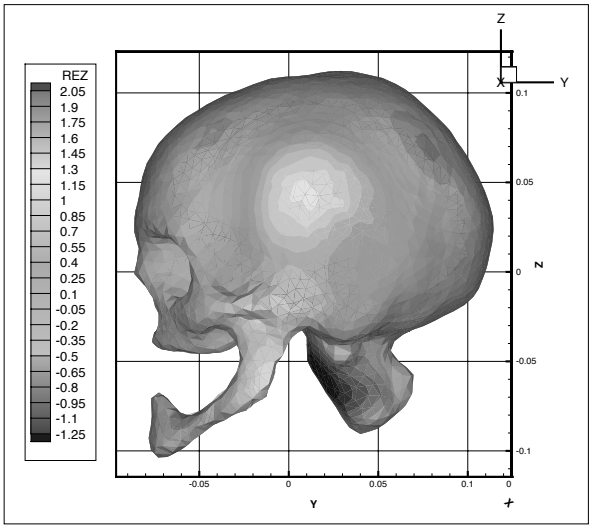
Figure 3.7: Propagation of a plane wave in a heterogeneous medium, F=1800 MHz



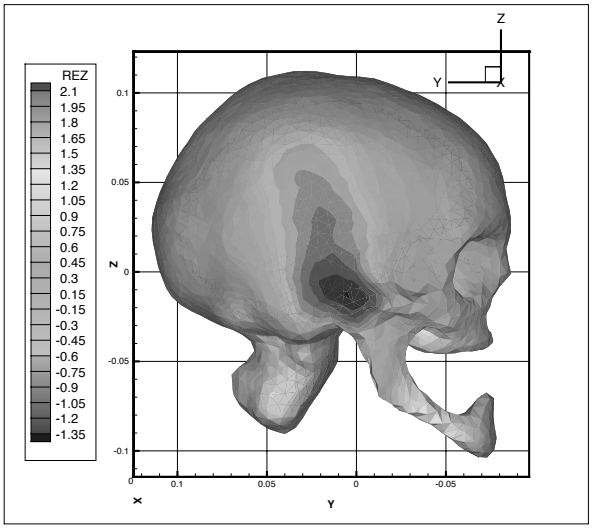
(a) method DG-$\mathbb{P}_1$-c, time-domain, skull, left side, mesh M1.



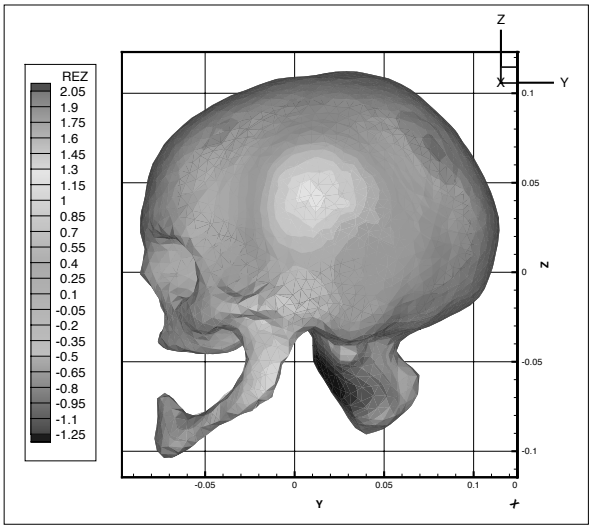
(b) method DG-$\mathbb{P}_1$-c, time-domain, skull, right side, mesh M1.



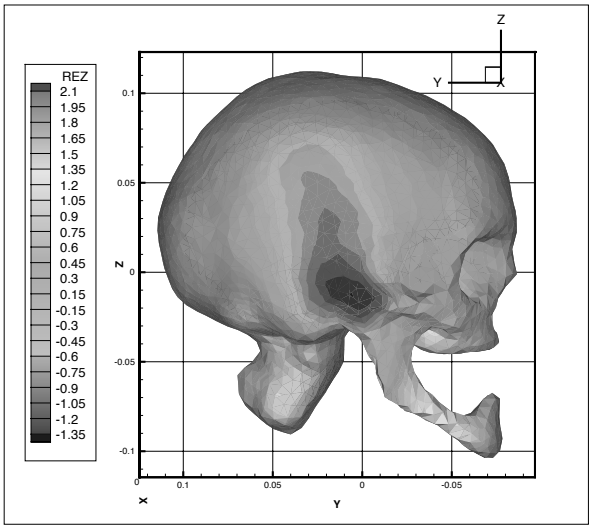
(c) method DG-$\mathbb{P}_1$-c, time-harmonic, skull, left side, mesh M1.



(d) method DG-$\mathbb{P}_1$-c, time-harmonic, skull, right side, mesh M1.

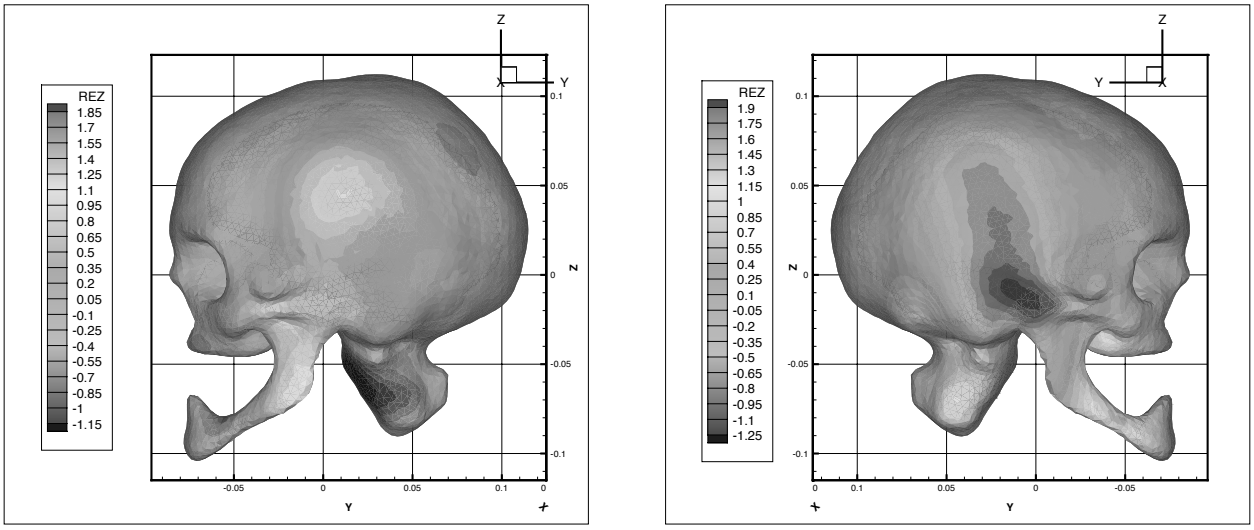


(e) method DG-$\mathbb{P}_1$-u, time-harmonic, skull, left side, mesh M1.



(f) method DG-$\mathbb{P}_1$-u, time-harmonic, skull, right side, mesh M1.

Figure 3.8: Propagation of a plane wave in a heterogeneous medium, F=1800 MHz



(a) Skull, left side.

(b) Skull, right side.

Figure 3.9: Propagation of a plane wave in a heterogeneous medium, F=1800 MHz. Contour lines of E_z: method DG-$\mathbb{P}_0$-c with mesh M2.

Publication list of the author

Publications during the PhD (or issued from the PhD work)

[1] V. Dolean, S. Lanteri, "A domain decomposition approach to finite volume solution of the Euler equations on unstructured triangular meshes ", *Int. J. Numer. Meth. Fluids*,vol 37-6, pp 625-656, 2001.

[2] V. Dolean, S. Lanteri, " A hybrid domain decomposition and multigrid method for the acceleration of compressible viscous flow calculations on unstructured triangular meshes",*Comp. Fluid. Dyn. J.*, vol 14, pp 287-304, 2001.

[3] V. Dolean, F. Nataf, "Non-overlapping domain decomposition algorithms for the system of Euler equations", *Domain decomposition methods in science and engineering*, pp 455–462, Theory Eng. Appli. Comput. methods, *Internat. Center Numer. Methods Eng.(CIMNE), Barcelona*, 2002.

[4] V. Dolean, S. Lanteri, F. Nataf, "Optimized interface conditions for domain decomposition methods in fluid dynamics", *Int. J. Numer. Meth. Fluids*, vol 40, pp 1539-1550, 2002.

[5] V. Dolean, S. Lanteri, F. Nataf, "Construction of interface conditions for solving the compressible Euler equations by-non-overlapping domain decomposition methods", *Int. J. Numer. Meth. Fluids*, vol 40, pp 1485-1492, 2002.

Publications after the PhD

[6] V. Dolean, S. Lanteri, F. Nataf, "Convergence analysis of a Schwarz type domain decomposition method for the solution of the Euler equations", *Appl. Num. Math.*, vol 49, pp 153-186, 2004.

[7] V. Dolean, S. Lanteri, "Parallel multigrid methods for the calculation of unsteady flows on unstructured grids: algorithmic aspects and parallel performances on clusters of PCs", *Parallel Computing*, vol 30, pp 503-525, 2004.

[8] V. Dolean, G. Rapin, F. Nataf, "New constructions of domain decomposition methods for systems of PDEs", C.R. Acad. Sci. Paris, Ser. I 340, pp 693-696, 2005.

[9] V. Dolean, F. Nataf, "An optimized domain decomposition method for the compressible Euler equations", *Domain decomposition methods in science and engineering XVI*, pp 173–180, *Lecture Notes in Computational Science and Engineering*,Olof B. Widlund and David E. Keyes Edts, Springer-Verlag, 2006.

[10] V. Dolean, F. Nataf, "Domain decomposition methods for the compressible Euler equations",*Contributions to Current Chalenges in Mathematical Fluid Dynamics*, *Advances in Mathematical Fluid Mechanics*, G. Galdi, J.G. Heywood, R. Rannacher Edts, Birkhauser, 2006.

[11] V. Dolean, F. Nataf, "A new domain decomposition method for the compressible Euler equations", ESAIM-M2AN (Modélisation Mathématique et Analyse Numérique), vol. 40, No. 4, pp 689-703, 2006.

[12] V. Dolean, F. Nataf, G. Rapin, "How to use the Smith Factorization for Domain Decomposition Methods applied to Stokes equations", *Domain decomposition methods in science and engineering XVII*, LNCSE, U. Langer, M. Discacciati, D.E. Keyes, O.B. Widlund, W. Zulehner (Eds.), Springer Verlag, 2008.

[13] V. Dolean, M. Gander, "Why Classical Schwarz Methods Applied to Hyperbolic Systems Can Converge even Without Overlap", *Domain decomposition methods in science and engineering XVII*, LNCSE, U. Langer, M. Discacciati, D.E. Keyes, O.B. Widlund, W. Zulehner (Eds.), Springer Verlag, 2008.

[14] V. Dolean, R. Pasquetti, F. Rapetti, "p-Multigrid for Fekete spectral element method", *Domain decomposition methods in science and engineering XVII*, LNCSE, U. Langer, M. Discacciati, D.E. Keyes, O.B. Widlund, W. Zulehner (Eds.), Springer Verlag, 2008.

[15] V. Dolean, H. Fol, S. Lanteri, R. Perrussel, "Solution of the time-harmonic Maxwell equations using discontinuous Galerkin methods", *Journal of Computational and Applied Mathematics*, vol. 218(2), pp 435-445, 2008.

[16] V. Dolean, S. Lanteri, R. Perrussel, "A domain decomposition method for solving the three-dimensional time-harmonic Maxwell equations discretized by discontinuous Galerkin methods", *Journal of Computational Physics*, vol. 227(3), pp 2044-2072, 2008.

[17] A. Catella, V. Dolean, S. Lanteri, "An inconditionnally stable discontinuous Galerkin method for solving 2D time-domain Maxwell equations on unstructured triangular meshes", *IEEE Transactions on Magnetics*, vol 44(6), 2008.

[18] V. Dolean, S. Lanteri, R. Perrussel, "Optimized Schwarz method for solving time-harmonic Maxwell equations discretized by discontinuous Galerkin methods", *IEEE Transactions on Magnetics*, vol 44(6), 2008.

[19] V. Dolean, G. Rapin, F. Nataf, "Deriving a new domain decomposition method for the Stokes equations using Smith factorization", *Mathematics of Computation*, vol 78, pp 789–814, 2009.

[20] V. Dolean, L. Gerardo-Giorda, M. Gander, "Optimized Schwarz methods for Maxwell equations", *preprint https://hal.archives-ouvertes.fr/ccsd-00107263*, accepted for publication in *SISC (SIAM Journal of Scientific Computing)* 2009.

[21] A. Catella, V. Dolean, S. Lanteri, "An implicit discontinuous Galerkin time-domain method for two-dimensional electromagnetic wave propagation", *preprint https://hal.inria.fr/inria-00126573*, accepted for publication in *COMPEL*, 2009.

Technical/research reports

[22] V. Dolean, S. Lanteri, "Une méthode volume fini implicite en maillages non-structurés pour les équations de Maxwell 3D en domaine temporel", INRIA Research Report *http://hal.inria.fr/inria-00070254/fr/*, 2005.

[23] V. Dolean, H. Fol, S. Lanteri, S. Piperno, "Méthodes de type Galerkin discontinu pour la résolution numérique des équations de Maxwell en régime fréquentiel", INRIA Research Report *http://hal.inria.fr/inria-00071362/fr/*, 2006.

[24] A. Catella, V. Dolean, S. Lanteri,"An Implicit DGTD Method for Solving the Two-Dimensional Maxwell Equations on Unstructured Triangular Meshes", INRIA Research Report *http://hal.inria.fr/inria-00126573/fr/*, 2007.

[25] V. Dolean, S. Lanteri, R. Perrussel,"Méthodes de type Galerkin discontinu pour les équations de Maxwell en régime harmonique: flux numériques et algorithmes multigrille", INRIA Research Report *http://hal.inria.fr/inria-00354510/fr/*, 2009.

Bibliography

[ADL00] P.R. Amestoy, I.S. Duff, and J.-Y. L'Excellent. Multifrontal parallel distributed symmetric and unsymmetric solvers. *Comput. Meth. App. Mech. Engng.*, 184:501–520, 2000.

[ALTNV00] Y. Achdou, P. Le Tallec, F Nataf, and M. Vidrascu. A domain decomposition preconditionner for an advection-diffusion problem. *CMAME*, 184:145–170, 2000.

[AN97] Y. Achdou and F. Nataf. A robin-robin preconditioner for an advection-diffusion problem. *C. R. Acad. Sci. Paris*, 325, Série I:1211–1216, 1997.

[ARGG06] A. Alonso-Rodriguez and L. Gerardo-Giorda. New non-overlapping domain decomposition methods for the time-harmonic Maxwell system. *SIAM J. Sci. Comp.*, 28(1):102–122, 2006.

[AS99] M. Ainsworth and S. Sherwin. Domain decomposition preconditioners for p and hp finite element approximations of Stokes equations. *Comput. Methods Appl. Mech. Engrg.*, 175:243–266, 1999.

[BGLTV89] Jean-François Bourgat, Roland Glowinski, Patrick Le Tallec, and Marina Vidrascu. Variational formulation and algorithm for trace operator in domain decomposition calculations. In Tony Chan, Roland Glowinski, Jacques Périaux, and Olof Widlund, editors, *Domain Decomposition Methods*, pages 3–16, Philadelphia, PA, 1989. SIAM.

[BGS07] Sylvie Benzoni-Gavage and Denis Serre. *Multi-dimensional hyperbolic partial differential equations: First-order Systems and Applications.* Oxford Mathematical Monographs. Oxford Science Publications, 2007.

[Bjø95a] M. Bjørhus. A note on the convergence of discretized dynamic iteration. *BIT*, 35:291–296, 1995.

[Bjø95b] M. Bjørhus. Semi-discrete subdomain iteration for hyperbolic systems. Technical Report 4, NTNU, 1995.

[BO05] J.-D. Boissonnat and S. Oudot. Provably good sampling and meshing of surfaces. *Graphical Models*, 67(5):405–451, 2005.

[BP05] A. Buffa and I. Perugia. Discontinuous Galerkin approximation of the Maxwell eigenproblem. Technical Report PV-24, IMATI-CNR, 2005.

[CCR05] M.H. Chen, B. Cockburn, and F. Reitich. High-order RKDG methods for computational electromagnetics. *J. Sci. Comput.*, 22-23:205–226, 2005.

[CDJP97] P. Collino, G. Delbue, P. Joly, and A. Piacentini. A new interface condition in the non-overlapping domain decomposition for the Maxwell equations. *Comput. Methods Appl. Mech. Engrg.*, 148:195–207, 1997.

[CFP06] G. Cohen, X. Ferriéres, and S. Pernet. A spatial high order hexahedral discontinuous Galerkin method to solve Maxwell's equations in time-domain. *J. Comput. Phys.*, 217(2):340–363, 2006.

[CFS98] X.-C. Cai, C. Farhat, and M. Sarkis. A minimum overlap restricted additive Schwarz preconditioner and appication in 3D flow simulations. In C. Farhat J. Mandel and X.-C. Cai, editors, *Proceedings of the 10th Domain Decomposition Methods in Sciences and Engineering*, volume 218 of *Contemporary Mathematics*, pages 479–485. AMS, 1998.

[Che93] L.P. Chew. Guaranteed-quality mesh generation for curved surfaces. In *9th Annual ACM Symposium Computational Geometry*, pages 274–280. ACM Press, 1993.

[Che98] Philippe Chevalier. *Méthodes numériques pour les tubes hyperfréquences. Résolution par décomposition de domaine.* PhD thesis, Université Paris VI, 1998.

[Cle98] S. Clerc. Non-overlapping Schwarz method for systems of first order equations. *Cont. Math*, 218:408–416, 1998.

[CN98] Philippe Chevalier and Frédéric Nataf. Symmetrized method with optimized second-order conditions for the Helmholtz equation. In *Domain decomposition methods, 10 (Boulder, CO, 1997)*, pages 400–407. Amer. Math. Soc., Providence, RI, 1998.

[CS98] Bernardo Cockburn and Chi-Wang Shu. The local discontinuous Galerkin method for time-dependent convection-diffusion systems. *SIAM J. Numer. Anal.*, 35(6):2440–2463 (electronic), 1998.

[Des93] Bruno Després. Domain decomposition method and the Helmholtz problem.II. In *Second International Conference on Mathematical and Numerical Aspects of Wave Propagation (Newark, DE, 1993)*, pages 197–206, Philadelphia, PA, 1993. SIAM.

[DJR92] Bruno Després, Patrick Joly, and Jean E. Roberts. A domain decomposition method for the harmonic Maxwell equations. In *Iterative methods in linear algebra (Brussels, 1991)*, pages 475–484, Amsterdam, 1992. North-Holland.

[EG06a] Alexandre Ern and Jean-Luc Guermond. Discontinuous Galerkin Methods for Friedrichs' systems. I. General theory. *SIAM J. Numer. Anal.*, 2006. In press.

[EG06b] Alexandre Ern and Jean-Luc Guermond. Discontinuous Galerkin Methods for Friedrichs' systems. II. Second-order elliptic PDE's. *SIAM J. Numer. Anal.*, 2006. In press.

[EHK98] B. Engquist and Zhao H.-K. Absorbing boundary conditions for domain decomposition. *Appl. Numer. Math.*, 27(4):341–365, 1998.

[EM77] B. Engquist and A. Majda. Absorbing boundary conditions for the numerical simulation of waves. *Math. Comp.*, 31:629–651, 1977.

[FLLP05] Loula Fezoui, Stéphane Lanteri, Stéphanie Lohrengel, and Serge Piperno. Convergence and stability of a discontinuous Galerkin time-domain method for the 3D heterogeneous Maxwell equations on unstructured meshes. *M2AN Math. Model. Numer. Anal.*, 39(6):1149–1176, 2005.

[Fre03] P. Frey. YAMS: a fully automatic adaptive isotropic surface remeshing procedure. INRIA Research Report No. 4252, 2003.

[FS89] L. Fezoui and B. Stoufflet. A class of implicit upwind schemes for Euler simulations with unstructured meshes. *J. of Comp. Phys.*, 84:174–206, 1989.

[Gan66] Felix R. Gantmacher. *Theorie des matrices*. Dunod, 1966.

[GGTN04] L. Gerardo-Giorda, P. Le Tallec, and F. Nataf. A robin-robin preconditioner for advection-diffusion equations with discontinuous coefficients. *Comput. Methods Appl. Mech. Engrg.*, 193:745–764, 2004.

[GHM07] M. J. Gander, L. Halpern, and F. Magoulès. An Optimized Schwarz Method with two-sided Robin transmission conditions for the Helmholtz Equation. *Int. J. Numer. Meth. Fluids*, 2007. in press.

[GHN99] M.-J. Gander, L. Halpern, and F. Nataf. Optimal convergence for overlapping and non-overlapping Schwarz waveform relaxation. In C.-H. Lai, P. Bjørstad, M. Cross, and O. Widlund, editors, *Proceedings of the 11th Domain Decomposition Methods in Sciences and Engineering*, pages 27–37, 1999.

[GHS91] P.-L. George, F. Hecht, and E. Saltel. Automatic mesh generator with specified boundary. *Comput. Methods Appl. Mech. Engrg.*, 92:269–288, 1991.

[GKM+91] R. Glowinski, Y.A. Kuznetsov, G. Meurant, J. Periaux, and O.B. Widlund, editors. *Fourth International Symposium on Domain Decomposition Methods for Partial Differential Equations*, Philadelphia, 1991. SIAM.

[GMN02] M.-J. Gander, F. Magoulès, and F. Nataf. Optimized Schwarz methods without overlap for the Helmholtz equation. *SIAM J. Sci. Comput.*, 24-1:38–60, 2002.

[GR86] V. Girault and P.A. Raviart. *Finite Element Methods for Navier-Stokes Equations*. Springer, Heidelberg-Berlin, 1986.

[HD94] P. Helluy and S. Dayma. Convergence d'une approximation discontinue des systèmes du premier ordre. *C. R. Acad. Sci. Paris Sér. I Math.*, 319(12):1331–1335, 1994.

[Hel94] P. Helluy. *Résolution numérique des équations de Maxwell harmoniques par une méthode d'éléments finis discontinus*. Thèse en mathématiques appliquées, Ecole Nationale Supérieure de l'Aéronautique, 1994.

[HL07] T. Hagstrom and S Lau. Radiation boundary conditions for Maxwell's equations: a review of accurate time-domain formulations. *J. Comput. Math*, 25(3):305–336, 2007.

[HPSS05a] Paul Houston, Ilaria Perugia, Anna Schneebeli, and Dominik Schötzau. Interior penalty method for the indefinite time-harmonic Maxwell equations. *Numer. Math.*, 100(3):485–518, 2005.

[HPSS05b] Paul Houston, Ilaria Perugia, Anna Schneebeli, and Dominik Schötzau. Mixed discontinuous Galerkin approximation of the Maxwell operator: the indefinite case. *M2AN Math. Model. Numer. Anal.*, 39(4):727–753, 2005.

[HTJ88] Thomas Hagstrom, R. P. Tewarson, and Aron Jazcilevich. Numerical experiments on a domain decomposition algorithm for nonlinear elliptic boundary value problems. *Appl. Math. Lett.*, 1(3), 1988.

[HW02] J. S. Hesthaven and T. Warburton. Nodal high-order methods on unstructured grids. I. Time-domain solution of Maxwell's equations. *J. Comput. Phys.*, 181(1):186–221, 2002.

[JN00] C. Japhet and F. Nataf. The best interface conditions for domain decomposition methods: absorbing boundary conditions. In L. Tourrette, editor, *Artificial Boundary Conditions, with Applications to Computational Fluid Dynamics Problems*, pages 348–373. 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*, 18:17–30, 2001.

[KD06] J. Kurzak and J. Dongarra. Implementation of the mixed-precision in solving systems of linear equations on the CELL processor. Technical Report UT-CS-06-580, University of Tennessee, 2006.

[KK99] G. Karypis and V. Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM J. Sci. Comput.*, 20(1):359–392, 1999.

[LC87] W. Lorensen and H. Cline. Marching cubes: a high resolution 3D surface construction algorithm. In *Siggraph 87*, volume 21, pages 163–170, 1987.

[Li05] J. Li. A Dual-Primal FETI method for incompressible Stokes equations. *Numer. Math.*, 102:257–275, 2005.

[LLL+06] J. Langou, J. Langou, P. Luszczek, J. Kurzak, A. Buttari, and J. Dongarra. Exploiting the performance of 32 bit floating point arithmetic in obtaining 64 bit accuracy. Technical Report UT-CS-06-574, University of Tennessee, 2006.

[LVL05] S.C. Lee, M. Vouvakis, and J. F. Lee. A Non-overlaping Domain Decomposition Method with Non-Matching Grids for Modeling Large Finite Antenna Arrays. *J. Comput. Phys.*, 203:1–21, 2005.

[LW06] J. Li and O. Widlund. BDDC algorithms for incompressible Stokes equations. *SIAM J. of Numer. Anal.*, 44(6):2432–2455, 2006.

[Man92] J. Mandel. Balancing domain decomposition. *Comm. on Applied Numerical Methods*, 9:233–241, 1992.

[Mon03] Peter Monk. *Finite element methods for Maxwell's equations*. Numerical Mathematics and Scientific Computation. Oxford University Press, New York, 2003.

[MR05] P. Monk and G.R. Richter. A discontinuous Galerkin method for linear symmetric hyperbolic systems in inhomogeneous media. *J. Sci. Comput.*, 22-23:443–477, 2005.

[Nat96] F. Nataf. Absorbing boundary conditions in block Gauss-Seidel methods for convection problems. *Math. Models Methods Appl. Sci.*, 6(4):481–502, 1996.

[Nat97] F. Nataf. Interface Conditions for Domain Decomposition Methods for 2D and 3D Oseen equations. *C. R. Acad. Sci., Paris*, Ser. I 324:1155–1160, 1997.

[Ned01] J.-C. Nedelec. *Acoustic and electromagnetic equations. Integral representations for harmonic problems.* Applied Mathematical Sciences, 144. Springer Verlag, 2001.

[NR07] F. Nataf and G. Rapin. Construction of a New Domain Decomposition Method for the Stokes Equations. In O.B. Widlund and D.E. Keyes, editors, *Domain Decomposition Methods in Science and Enignieering XVI*, pages 247–254. Springer, 2007.

[NRdS95] F. Nataf, F. Rogier, and E. de Sturler. Domain decomposition methods for fluid dynamics. In A. Sequeira, editor, *Navier-Stokes equations and related nonlinear analysis*, pages 367–376. Plenum Press Corporation, 1995.

[NV08] R. Nabben and C. Vuik. A comparison of abstract versions of deflation, balancing and additive coarse grid correction preconditioners. *Numer. Linear Algebra Appl.*, 15:355–372, 2008.

[OL98] F.-C. Otto and G. Lube. A nonoverlapping domain decomposition method for the Oseen equations. *Math. Models Methods Appl. Sci.*, 8:1091–1117, 1998.

[OLM01] F.-C. Otto, G. Lube, and L Müller. An iterative substructuring method for div-stable finite element approximations of the Oseen problem. *Computing*, 67:91–117, 2001.

[Pip00] Serge Piperno. L^2-stability of the upwind first order finite volume scheme for the Maxwell equations in two and three dimensions on arbitrary unstructured meshes. *M2AN Math. Model. Numer. Anal.*, 34(1):139–158, 2000.

[PSM02] I. Perugia, D. Schötzau, and P. Monk. Stabilized interior penalty methods for the time-harmonic Maxwell equations. *Comput. Methods Appl. Mech. Engrg.*, 191(41-42):4675–4697, 2002.

[PW02] L.F. Pavarino and O.B. Widlund. Balancing neumann-neumann methods for incompressible stokes equations. *Comm. Pure Appl. Math.*, 55:302–335, 2002.

[QS96] A. Quarteroni and L. Stolcis. Homogeneous and heterogeneous domain decomposition methods for compressible flow at high reynolds numbers. Technical Report 33, CRS4, 1996.

[Qua90] A. Quarteroni. Domain decomposition methods for systems of conservation laws : spectral collocation approximation. *SIAM J. Sci. Stat. Comput.*, 11:1029–1052, 1990.

[RHGJ03] P. Ratiu, B. Hillen, J. Glaser, and D. P. Jenkins. *Medicine Meets Virtual Reality 11 - NextMed: Health Horizon*, volume 11, chapter Visible Human 2.0 - the next generation, pages 275–281. IOS Press, 2003.

[Ron96] E. Ronquist. A Domain Decomposition Solver for the Steady Navier-Stokes Equations. In A. Ilin and L. Scott, editors, *Proc. of ICOSAHOM.95*, pages 469–485. Houston Journal of Mathmatics, 1996.

[RT91] Y.H. De Roeck and P. Le Tallec. Analysis and Test of a Local Domain Decomposition Preconditioner. In *R. Glowinski et al. [GKM+91]*, 1991.

[SF93] G.L.G. Sleijpen and D.R. Fokkema. BiCGstab(ℓ) for linear equations involving unsymmetric matrices with complex spectrum. *Electron. Trans. Numer. Anal.*, 1:11–32 (electronic only), 1993.

[SS86] Y. Saad and H. Schultz. GMRES : Generalized minimal residual algorithm for solving non-symmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7:856–869, 1986.

[TP97] P . Le Tallec and A. Patra. Non-overlapping domain decomposition methods for adaptive hp approximations of the Stokes problem with discontinuous pressure fields. *Comput. Methods Appl. Mech. Engrg.*, 145:361–379, 1997.

[TW04] Andrea Toselli and Olof Widlund. *Domain Decomposition Methods - Algorithms and Theory*, volume 34 of *Springer Series in Computational Mathematics*. Springer, 2004.

[VCL06] M. Vouvakis, Z. Cendes, and J. F. Lee. A FEM Domain Decomposition Method for Photonic and Electromagnetic Band Gap Structures. *IEEE Trans. Ant. Prop.*, 54(2), 2006.

[WRL95] J.T. Wloka, B. Rowley, and B. Lawruk. *Boundary Value Problems for Elliptic Systems*. Cambridge University Press, 1995.

[Yee66] K.S. Yee. Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media. *IEEE Trans. Antennas and Propag.*, 14(3):302–307, 1966.