Parallel Double Sweep Preconditioner for the Optimized Schwarz Algorithm Applied to High Frequency Helmholtz and Maxwell Equations

A. Vion[1] and C. Geuzaine[1]

The principle of sweeping to accelerate the solution of wave propagation problems has recently retained much interest, yet with different approaches (Engquist and Ying [2011], Stolk [2013]). We recently proposed a preconditioner for the optimized Schwarz algorithm, based on a propagation of information using a double sequence of subproblems solves, or sweeps (Vion et al. [2013], Vion and Geuzaine [2014]). Although this procedure significantly reduces the number of iterations when many subproblems are involved, the sequential nature of the process hinders the scalability of the algorithm on parallel computer architectures. Here we propose a modified version of the algorithm that concurrently runs partial sweeps on smaller groups of domains, which efficiently reduces the preconditioner application time on parallel machines. We show that the algorithm is applicable to both Helmholtz and Maxwell equations.

1 Non-overlapping optimized Schwarz algorithm

We consider the optimized Schwarz algorithm for the Helmholtz and Maxwell equations. The algorithm makes use of impedance boundary conditions on the artificial interfaces; although overlapping variants of it exist, we focus on the non-overlapping version, with a partition of the domain into N_d subdomains $\Omega_{1\leq i\leq N_d}$, such that $\cup\bar{\Omega}_i = \bar{\Omega}$ and with $\Sigma_{ij} = \bar{\Omega}_i \cap \bar{\Omega}_j$ the common boundary between two adjacent domains. An iteration of the algorithm for Helmholtz (see e.g. Peng et al. [2010] for the Maxwell formulation) is the solution of the subproblems:

$$\begin{aligned} -(\Delta + k^2)u_i^{(k+1)} &= 0 && \text{in } \Omega_i, \\ (\partial_n + \mathcal{S})u_i^{(k+1)} &= g_{ij}^{(k)} && \text{on } \Sigma_{ij}, \end{aligned} \tag{1}$$

[1] University of Liège, Montefiore Institute, Grande Traverse, 10, 4000 Liège – Belgium cgeuzaine@ulg.ac.be

with boundary conditions on the external boundaries inherited from the original problem. The iteration completes with the update relations:

$$\begin{aligned} g_{ij}^{(k+1)} &= -\partial_n u_j^{(k+1)} + \mathcal{S} u_j^{(k+1)} \quad \text{on } \Sigma_{ij}, \\ &= -g_{ji}^{(k)} + 2\mathcal{S} u_j^{(k+1)}. \end{aligned} \tag{2}$$

The algorithm can classically be accelerated by rewriting it in a compact form as a fixed point iteration involving an iteration operator $\mathcal{A}$:

$$g^{(k+1)} = \mathcal{A} g^{(k)} + b. \tag{3}$$

Its solution g satisfies the linear system $\mathcal{F} g = b$, with $\mathcal{F} = \mathcal{I} - \mathcal{A}$ and b the right-hand side containing the contribution of the physical sources. Operator $\mathcal{F}$ involves the solution of subproblems and the update of the interface quantities g_{ij}; as we will see in Section 2, it is non-symmetric, hence amenable to a GMRes iterative solver. The optimal choice for the operator $\mathcal{S}$ used in the transmission conditions is the Dirichlet-to-Neumann (DtN) map, as shown in Nataf [2002]. It is a non-local operator, hence difficult to manipulate in local discretization methods like the Finite Element Method. The literature proposes different local approximations of it, among which we choose a truncated rational approximation of order $(2, 2)$ (see Boubendir et al. [2012], Bouajaji et al. [2014].)

In order to circumvent the difficulties associated with the so-called crosspoints (points that are at the intersection of more than two subdomains), we will consider two kinds of decompositions that naturally avoid them: layered or 1d-like decompositions, and cyclic decompositions around an object. Figure 1 shows basic examples of such decompositions.

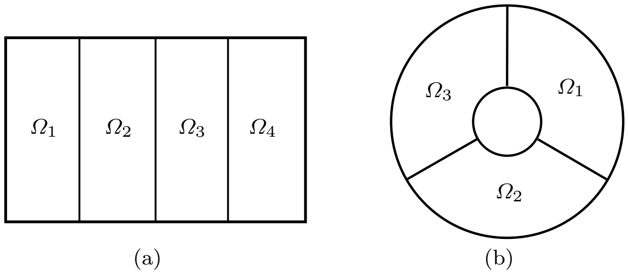

Fig. 1 Two topologies of a decomposed domain into non-overlapping subdomains, without crosspoint: (a) ”layered” decomposition; (b) ”cyclic” decomposition around an object.

2 Study of the iteration operator

Because the unknown g of the system can be regarded as a composite vector of unknown functions g_{ij}, the iteration operator $\mathcal{F}$ can be written as a matrix F, whose coefficients are operators acting on the interface functions. They take as input a function defined on one side of a domain and transfer the information over the domain, to the opposite interface, where a homogeneous transmission condition is imposed. We will refer to them as transport operators. There are two transport operators defined on the i-th subdomain, that we denote by $\mathcal{B}_i^f$ and $\mathcal{B}_i^b$, where the f and b indices refer to the forward or backward direction of the transfer. This distinction is important for what follows, as we will see in Section 3 that the convergence of the algorithm can be accelerated by propagating information over longer distances, simultaneously in the forward and backward directions.

We will first consider the case where the "true" DtN map $\mathcal{D}$ is used as transmission operator, leading to a perfectly non-reflecting boundary condition that lets the wave freely propagate outside the domain, without reflection. The effect of an imperfect transmission condition on the structure and the properties of the operator will be considered in a second step. The matrix writes, with the vector of unknown functions $g = [g_{12}, g_{21}, g_{23}, \dots]^T$ and a layered topology of the decomposition (Figure 1(a)):

$$F_{\mathcal{D}}(N_d) = \begin{bmatrix}
\mathcal{I} & & \mathcal{B}_2^b & & & & \\
 & \mathcal{I} & & & & & \\
 & & \mathcal{I} & & \ddots & & \\
 & \mathcal{B}_2^f & & \mathcal{I} & & & \\
 & & \ddots & & \ddots & \mathcal{B}_{N_d-1}^b & \\
 & & & & & \mathcal{I} & \\
 & & & & \mathcal{B}_{N_d-1}^f & & \mathcal{I}
\end{bmatrix}. \qquad (4)$$

Even when the optimized Schwarz algorithm is used with the optimal choice of transmission operator, its convergence is strongly impacted by the number of subdomains N_d, and can become very slow for large numbers of domains. This is classically understood as being caused by the local interactions of the subdomains in the algorithm, that are able to exchange information only with their direct neighbours at each iteration. There are situations, like in waveguides, where the information needs to travel through all the domains before the algorithm is able to build an acceptable solution everywhere. The situation is even worse if the information is distorted while being passed through a non-ideal transmission condition. That intuitive explanation is supported by the spectral properties of the iteration operator, that is defective (lacks a full basis of eigenvectors, while still being invertible) in the case of exact DtN map, which is known to cause slow convergence of Krylov solvers. With an approximate DtN map $\tilde{\mathcal{D}}$ and large N_d, another source of poor convergence resides in the fact that some of the eigenvalues get close to 0 for large N_d, leading to large condition numbers, while the operator can still be considered close to defective.

3 Preconditioning strategy for convergence acceleration

We start from the principle that a preconditioner should be a good approximation of the inverse of the system to be solved, and observe that the inverse of the matrix of the operator with exact DtN map can be easily obtained via a recurrence relation, for an arbitrary number of subdomains. Therefore, we design our preconditioner as having the same structure as the inverse of the ideal operator (4) $\mathcal{F}_{\mathcal{D}}^{-1}$, though using an approximate DtN map. Its product with a vector can be obtained as a matrix-free routine that performs a double sequence of subproblem solves, in the forward and backward directions, hence the name “double sweep” preconditioner (Vion et al. [2013], Vion and Geuzaine [2014]). This is made possible by the fact that we can give an interpretation of the coefficients of the inverse matrix, that are products of transport operators $\mathcal{B}_i^{\{f,b\}}$, as the transport of information between distant subdomains. As the two sweeps are independent from each other, they can be performed in parallel, as can be seen on the left diagram of Figure 3. Because we do not need to know the exact nature of the transport operators, the strategy is exactly the same for Helmholtz and Maxwell problems.
The effect of the preconditioner on the spectrum of the preconditioned nonideal operator $\mathcal{F}_{\tilde{\mathcal{D}}}\mathcal{F}_{\mathcal{D}}^{-1}$ is a strong clustering of the eigenvalues around $(1, 0)$, which ensures a good conditioning of the operator. That being so, the eigenvectors are now well distinct from each other, which enables fast convergence of the modified algorithm.

4 Parallelization of the double sweep

An important shortcoming of the double sweep preconditioner is its sequential nature, that destroys the scalability of the algorithm on parallel computers: assigning each subdomain to a separate CPU makes the preprocessing and the application of the iteration operator fully parallel, but these CPUs will remain idle during most of the application of the sweeps. An alternative strategy is to perform shorter sweeps over smaller groups of subdomains, independently of the other groups, by cutting the long sequence into smaller ones. This method still enables the sharing of information over longer distances than a single domain, yet not over the whole domain as before. The advantage is of course that the sweeps over each group can be performed simultaneously, therefore partially restoring scalability. Consequently, one can expect a degradation of the preconditioner performance compared to the original version, since it approximates the inverse of the Schwarz operator less accurately. The timeline of subdomains solves reported on Figure 3 highlights

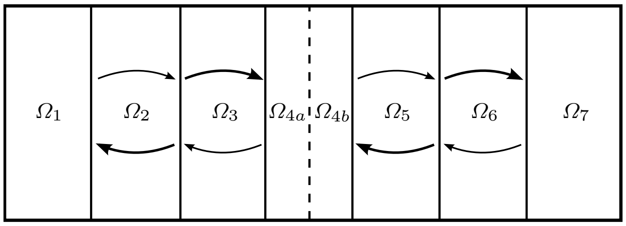

Fig. 2 Partial sweeps cover non-overlapping groups of domains, separated by the dashed line. The position of the cut inside the domain is not important as the first and last domains are not solved in our sweeps, as shown by the arrows.

the improved level of parallelism when using 2 cuts (right) instead of none (left).

A similar preconditioning strategy can be followed when the domain is decomposed as in Figure 1(b): introducing (at least) one cut in the cyclic decomposition allows to use the double sweep preconditioner as is.

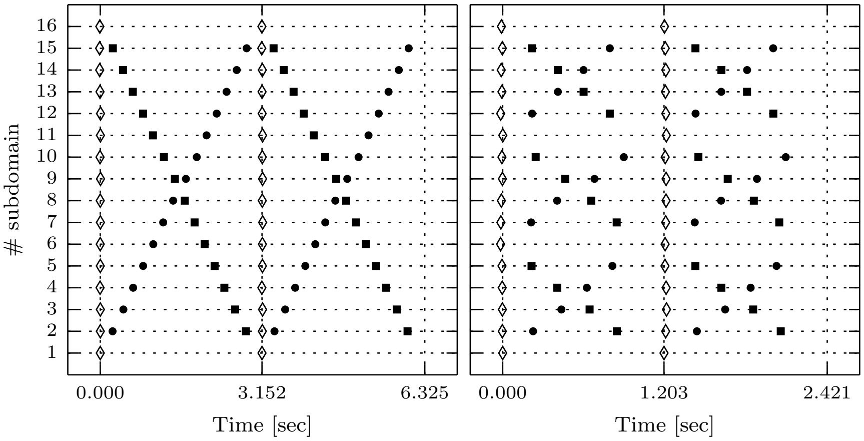

Fig. 3 Introducing 2 cuts in the double sweep preconditioner (right) enables parallel execution of the partial sweeps, reducing the application time of the preconditioner without cuts (left). The white diamonds indicate solves performed in the iteration operator; the black circles and squares indicate solves in the forward and backward sweeps, respectively. These time lines were obtained for the COBRA test case of Section 5, with 16 subdomains and cuts in subdomains 6 and 11.

5 Numerical results

We present results obtained on three different test geometries: a straight 3d (parallelepipedic) waveguide, a 3d S-shaped cavity (the COBRA benchmark defined by the JINA98 workgroup) and the open 2d scattering problem by a circular object. The first two are solved using a layered decomposition while the third uses a cyclic decomposition. The COBRA is solved for both Helmholtz and Maxwell, while the other two are solved for Helmholtz only. Earlier work (Vion et al. [2013], Vion and Geuzaine [2014], Boubendir et al. [2012]) has shown that without preconditioner, the iteration count for such problems typically grows linearly with the number of domains, and that with the use of the double sweep it becomes almost independent for layered decompositions, provided that the approximation of the DtN map is sufficiently accurate.

Tables 1–3 summarize the number of iterations required by each algorithm to converge to the prescribed tolerance, together with an estimation of the normalized times required for the completion of the algorithm. Provided that at least 2 CPUs are alloted per group of domains, the time required for the application of the standard Schwarz operator and the double sweep preconditioner with N_d subdomains, N_c cuts and C_{tot} CPUs (assumed evenly distributed between the groups of subdomains) are approximately given, in the case of a layered decomposition by:

$$T_{\text{Sch}} = \frac{N_d}{C_{\text{tot}}} T_p \quad \text{and} \quad T_{\text{sw}}(N_c) = \left\lceil \frac{N_d - N_c - 2}{N_c + 1} \right\rceil T_p,$$

with T_p the solution time for one subproblem (supposed identical for all subdomains). Note that T_{sw} would be doubled if only one CPU is available to perform the double sweep per group of domains. Slightly different estimations hold in the case of the cyclic decomposition. The total solution times for the unpreconditioned and double sweep algorithms are then $T_{\text{sol}}^{(np)} = T_{Sch} N_{it}^{(np)}$ and $T_{\text{sol}}^{(ds)}(N_c) = (T_{\text{Sch}} + T_{\text{sw}}(N_c)) N_{it}^{(ds)}$.

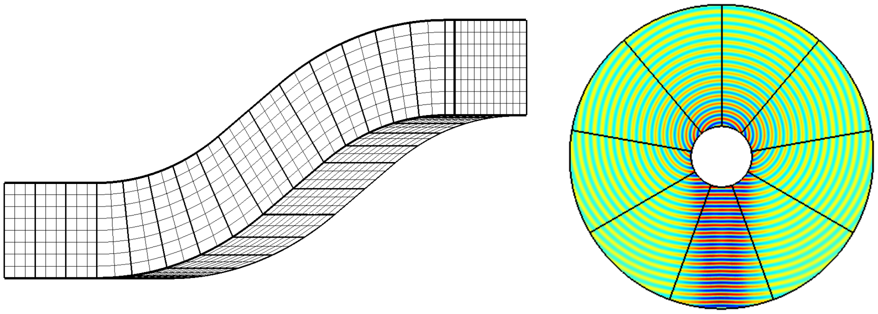

Fig. 4 Geometry and typical decomposition of the 3d cobra cavity (JINA98) and 2d scattering (unit sound-soft disc with Sommerfeld ABC at radius = $5m$) test cases. They differ by the topology of the decomposition (layered vs. cyclic) and by the type of wave involved (guided vs. free.) The parallelepipedic waveguide (not pictured) has dimensions $0.91m \times 0.084m \times 0.11m$, comparable to the COBRA.

#CPU	2	4	6	8	14	22
N_c	0	1	2	3	6	10
$N_{\text{it}}^{(\text{ds})}$	5	6	8	10	16	24
$T_{\text{sol}}^{(\text{ds})}$	230	138	128	110	112	96
$N_{\text{it}}^{(\text{np})}$	62					
$T_{\text{sol}}^{(\text{np})}$	992	496	331	248	142	91

#CPU	2	4	6	8	14	22
N_c	0	1	2	3	6	10
$N_{\text{it}}^{(\text{ds})}$	116	153	174	188	241	308
$T_{\text{sol}}^{(\text{ds})}$	5336	3519	2784	2068	1687	1232
$N_{\text{it}}^{(\text{np})}$	766					
$T_{\text{sol}}^{(\text{np})}$	12256	6128	4086	3064	1751	1115

Table 1 Straight waveguide (left) and COBRA (right) cases for Helmholtz with 32 subdomains, $k = 314.16$ (relative residual decrease by 10^{-4}).

#CPU	2	4	6	8	14	22
N_c	0	1	2	3	6	10
$N_{\text{it}}^{(\text{ds})}$	21	34	48	62	104	160
$T_{\text{sol}}^{(\text{ds})}$	966	782	768	682	728	640
$N_{\text{it}}^{(\text{np})}$	448					
$T_{\text{sol}}^{(\text{np})}$	7168	3584	2390	1792	1024	652

#CPU	2	4	6	8	14	22
N_c	0	1	2	3	6	10
$N_{\text{it}}^{(\text{ds})}$	44	74	105	135	230	354
$T_{\text{sol}}^{(\text{ds})}$	2024	1702	1680	1485	1610	1416
$N_{\text{it}}^{(\text{np})}$	> 1000					
$T_{\text{sol}}^{(\text{np})}$	> 16016	> 8008	> 5339	> 4004	> 2288	> 1456

Table 2 COBRA test case for Maxwell with 32 subdomains, $k = 157.08$ (left) and $k = 314.16$ (right) (relative residual decrease by 10^{-4}).

#CPU	2	52	86
N_c	1	26	43
$N_{\text{it}}^{(\text{ds})}$	24	27	31
$T_{\text{sol}}^{(\text{ds})}$	4584	189	124
$N_{\text{it}}^{(\text{np})}$	55		
$T_{\text{sol}}^{(\text{np})}$	3520	136	82

#CPU	2	52	86
N_c	1	26	43
$N_{\text{it}}^{(\text{ds})}$	20	29	37
$T_{\text{sol}}^{(\text{ds})}$	3820	203	148
$N_{\text{it}}^{(\text{np})}$	85		
$T_{\text{sol}}^{(\text{np})}$	5440	210	127

Table 3 Scattering test case for Helmholtz with 128 subdomains, k = 6.28 (left) and $k = 25.13$ (right) (relative residual decrease by 10^{-4}).

Tables 1–3 show that in all cases the behaviour of the algorithm is similar. The preconditioner strongly reduces the number of iterations, and thus the number of overall linear system solves. Moreover, the parallel version of the preconditioner makes it also an appealing proposition with respect to the overall computational (wall-clock) time when the number of CPUs is smaller than the number of subdomains, especially in the high frequency regime. For example, in the challenging COBRA case for Maxwell, with 32 domains on 8 CPUs (3 cuts), with $k = 100\pi$, the preconditioned version requires $135 \times (32 + 2 \times (32 - 2 - 3)) = 11610$ system solves instead of $> 1000 \times 32$ and runs about 3 times faster than the standard algorithm.

6 Conclusion

We have presented a double sweep preconditioning strategy for the optimized Schwarz algorithm and a variant of it that performs the double sweeps in parallel on groups of subdomains, rather than over all subdomains. Numerical results highlight the potential of the approach for both Helmholtz and Maxwell in the high frequency regime.

Acknowledgments

Work supported in part by the Belgian Science Policy (IAP P7/02.) Computational resources provided by CÉCI, funded by F.R.S.-FNRS under Grant No. 2.5020.11

References

M. El Bouajaji, X. Antoine, and C. Geuzaine. Approximate local magnetic-to-electric surface operators for time-harmonic Maxwell's equations. *Preprint, submitted to J. Comput. Phys.*, 2014.

Y. Boubendir, X. Antoine, and C. Geuzaine. A quasi-optimal non-overlapping domain decomposition algorithm for the Helmholtz equation. *J. Comput. Phys.*, 231(2):262–280, 2012. ISSN 0021-9991. doi: 10.1016/j.jcp.2011.08.007. URL `http://dx.doi.org/10.1016/j.jcp.2011.08.007`.

Björn Engquist and Lexing Ying. Sweeping preconditioner for the Helmholtz equation: moving perfectly matched layers. *Multiscale Model. Simul.*, 9 (2):686–710, 2011. ISSN 1540-3459. doi: 10.1137/100804644. URL `http://dx.doi.org/10.1137/100804644`.

Frédéric Nataf. Interface connections in domain decomposition methods. In *Modern methods in scientific computing and applications (Montréal, QC, 2001)*, volume 75 of *NATO Sci. Ser. II Math. Phys. Chem.*, pages 323–364. Kluwer Acad. Publ., Dordrecht, 2002.

Zhen Peng, Vineet Rawat, and Jin-Fa Lee. One way domain decomposition method with second order transmission conditions for solving electromagnetic wave problems. *J. Comput. Phys.*, 229(4):1181–1197, 2010. ISSN 0021-9991. doi: 10.1016/j.jcp.2009.10.024. URL `http://dx.doi.org/10.1016/j.jcp.2009.10.024`.

Christiaan C. Stolk. A rapidly converging domain decomposition method for the Helmholtz equation. *Journal of Computational Physics*, 241(0):240 – 252, 2013. ISSN 0021-9991. doi: http://dx.doi.org/10.1016/j.jcp.2013.01.039.

A. Vion and C. Geuzaine. Double sweep preconditioner for optimized Schwarz methods applied to the Helmholtz problem. *Preprint, submitted to J. Comput. Phys.*, 2014.

A. Vion, R. Bélanger-Rioux, L. Demanet, and C. Geuzaine. A DDM double sweep preconditioner for the Helmholtz equation with matrix probing of the DtN map. In *Mathematical and Numerical Aspects of Wave Propagation WAVES 2013*, June 2013.