20

Convergence Rate of Schwarz-type Methods for an Arbitrary Number of Subdomains

Frederic Nataf and Francis Nier

1 Introduction

The original Schwarz method is based on the use of Dirichlet boundary conditions as interface conditions (see [Lio89] and references therein). Convergence can be reached only with overlapping subdomains. As a result, the convergence is very slow when the overlap is small. In order to speed up the convergence and to be able to handle nonoverlapping subdomains, it has been proposed in [Lio89] to replace the Dirichlet boundary conditions by Fourier or more complex boundary conditions. The question is then to choose the best interface conditions both in terms of convergence rate and of easiness of use and implementation. In order to make a proper choice, it is important to quantify the effect of the boundary conditions on the convergence. This is usually done by a Fourier analysis for a two-domain decomposition (see [Des90], [CNR91], [CQ93], [TB94], [NR95], and [Jap96]).

A natural question is then what happens when there are more than two subdomains. In [NN97], we prove the convergence of three Schwarz-type algorithms. We establish a link between the convergence rate of a two-subdomain decomposition and the convergence for a decomposition into an arbitrary number of subdomains.

2 Two-domain Decomposition Convergence Rate

In order to find good interface conditions, a common practice is to consider the problem set on $\mathbf{R}^2$ decomposed into two half-planes. If we have to solve $\mathcal{L}(u) = f$ on $\mathbf{R}^2$, the Schwarz algorithm is

$$\begin{aligned} \mathcal{L}(u_1^{n+1}) = f \text{ in } \mathbf{R}^2_-, \quad \mathcal{B}_{12}(u_1^{n+1}) = \mathcal{B}_{12}(u_2^n) \text{ at } x = 0 \\ \mathcal{L}(u_2^{n+1}) = f \text{ in } \mathbf{R}^2_+, \quad \mathcal{B}_{21}(u_2^{n+1}) = \mathcal{B}_{21}(u_1^n) \text{ at } x = 0 \end{aligned}$$

Ninth International Conference on Domain Decomposition Methods
Editor Petter E. Bjørstad, Magne S. Espedal and David E. Keyes

The operators $\mathcal{B}_{12}$ and $\mathcal{B}_{21}$ are interface conditions to be chosen. As an example, $\mathcal{B}_{12,21}$ can be sought in the form of a partial differential operator of order two in the tangential direction $\mathcal{B}_{12} = \partial_x + \alpha + \beta\partial_y + \gamma\partial^2_{yy}$. The convergence rate will depend on the value of the coefficients α, β and γ. Usually the study is made by freezing the coefficients so that Fourier transform in the direction y can be used. The dual variable is denoted by k. It is then easy to compute explicitly a formula for the convergence rate ρ as a function of the Fourier variable k. For instance, if absorbing boundary conditions of order 0 wrt k are used for $\mathcal{B}_{12,21}$, ρ equals zero at $k = 0$ and tends to 1 as k tends to infinity (see [Des90], [NR95]). One may also be tempted to optimize the convergence rate with respect to some of the coefficients α, β or γ (see [TB94], [Jap96]). This kind of study yields valuable information. For more details on the importance of interface conditions and also a numerical study, see the proceedings of C. Japhet [Jap96] in this volume.

Now, a natural question is what happens when there is an arbitrary number N of subdomains. The answer is not obvious since it amounts to estimating the norms of a $2N - 2$ squared matrix raised to any power n.

3 The Main Result

In [NN97], we prove the convergence of three Schwarz-type methods with or without overlapping. We also establish a link between the two-domain convergence rate and the convergence for an arbitrary number of subdomains.

The space $\mathbf{R}^{d+1}$ is decomposed into N vertical strips. A constant coefficient advection-diffusion equation is solved: $\mathcal{L}(u) = f$. The velocity in the direction x is positive. The flow goes from the left to the right. Three methods are considered. In the first method, the update is simultaneous in all the subdomains. This is the additive Schwarz method (ASM). In the second algorithm, the update is made sequentially by sweeping over the domains following the direction of the flow (FDS) (see [HIKW92], [Joh92], [Nat96]). The last method is a variation of the previous one. The approximations in the subdomains are updated by double sweeps over the subdomains (DS).

The result is the following

Theorem 3.1 *There exists a function $\rho(k)$ taking its values in $[0, 1)$ independent of N so that $\hat{e}^n_i(k)$ the k-th component of the error in the Fourier space for the method i is estimated as follows:*

$$\|\hat{e}^n_i(k)\| \le C_i(k)\rho(k)^{[n/p_i]} \text{ for } n \ge n_i,$$

where $[m]$ denotes the integer part of m. The values of p_i and n_i depend on the method:

$$p_{ASM} = 2N-2,\ p_{FDS} = N-1,\ p_{DS} = 1,\ n_{ASM} = 2N+1,\ n_{FDS} = 2N-1,\ n_{DS} = 3.$$

Due to the ellipticity of the operator, we also prove that the function ρ is almost equal to the two-domain decomposition convergence rate. A connection is thus established between the study of convergence with two subdomains and the case with N subdomains.

The proof is unusual since it relies on techniques originating in formal language theory. It is worth noticing that this result is sharper than the estimate of the spectral radius of the iteration matrix. Indeed, the result would be something like: for any positive ϵ there exists some positive constant C_ϵ so that for n larger than some integer n_ϵ, the error is bounded by $C_\epsilon\,(\rho+\epsilon)^n$. While, here, the constant is known explicitly and the estimate is valid from a rank which is an explicit function of N.

4 Sketch of the Proof

Reformulation of the Algorithm

The proof is in two steps. First the algorithms are reformulated so that the unknowns are functions living on the boundaries of the subdomains. Then, we conclude with an algebraic trick.

The first part is very classical (see [NRdS94], [NN97]). Let H denote a vector of functions living on the boundaries of the subdomains. It may be seen that H satisfies a linear equation:

$$(Id - \mathcal{T})(H) = G$$

The matrix-vector product $\mathcal{T}(H)$ consists in solving in parallel a boundary value problem in each subdomain. The operator $\mathcal{T}$ is split into the sum of four operators $\mathcal{T} = \mathcal{T}_{ll} + \mathcal{T}_{rr} + \mathcal{T}_{rl} + \mathcal{T}_{lr}$. This enables us to give a compact form of the iteration matrices of the different algorithms (simply $\mathcal{T}$ for the additive Schwarz method, $\mathcal{T}_{FDS}$ for the flow directed algorithm and by $\mathcal{T}_{DS}$ for the double sweeps algorithm):

$$\mathcal{T}_{FDS} = (Id - \mathcal{T}_{ll} - \mathcal{T}_{lr})^{-1}(\mathcal{T}_{rl} + \mathcal{T}_{rr})$$
$$\mathcal{T}_{DS} = (Id - \mathcal{T}_{rr} - \mathcal{T}_{rl})^{-1}(Id - \mathcal{T}_{ll} - \mathcal{T}_{lr})^{-1}(\mathcal{T}_{ll} + \mathcal{T}_{lr})(\mathcal{T}_{rl} + \mathcal{T}_{rr})$$

The basis for the algebraic trick is the following set of relations:

$$\begin{array}{l} \mathcal{T}_{ll}^{N-1} = \mathcal{T}_{rr}^{N-1} = 0; \quad \mathcal{T}_{ll}\,\mathcal{T}_{rr} = \mathcal{T}_{rr}\,\mathcal{T}_{ll} = 0; \quad \mathcal{T}_{rl}^2 = \mathcal{T}_{lr}^2 = 0 \\ \mathcal{T}_{lr}\,\mathcal{T}_{ll} = \mathcal{T}_{rl}\,\mathcal{T}_{rr} = 0; \quad \mathcal{T}_{ll}\,\mathcal{T}_{rl} = \mathcal{T}_{rr}\,\mathcal{T}_{lr} = 0 \end{array} \tag{1}$$

It is worth noticing that these relations come from the structure of the matrices and do not depend on the value of the coefficients.

By using formal language theory, we prove the following.

Theorem 4.1 *If*

$$\rho = \|\mathcal{T}_{rl}\|\,\|\mathcal{T}_{lr}\|\,(\sum_{i=0}^{N-2}\|\mathcal{T}_{rr}^i\|)\,(\sum_{i=0}^{N-2}\|\mathcal{T}_{ll}^i\|) < 1,$$

Then,

$$\|\mathcal{T}^n\| \le C(1-\rho)^{-1}\,\rho^{[n/(2N-2)-3/2]} \quad \text{for} \;\; n \ge 2N$$
$$\|\mathcal{T}_{FDS}^n\| \le C(1-\rho)^{-1}\,\rho^{[n/(N-1)-2]} \quad \text{for} \;\; n \ge 2N-2$$
$$\|\mathcal{T}_{DS}^n\| \le C(1+\rho)\,\rho^{n-1} \quad \text{for} \;\; n \ge 2$$

where the constant C is given explicitly

$$C = (1 + \rho/\|\mathcal{T}_{lr}\|)\,(1 + \rho/\|\mathcal{T}_{rl}\| + \rho/\|\mathcal{T}_{rl}\|\|\mathcal{T}_{lr}\|)$$

Another way to look at this result is to remark that when $\mathcal{T}_{lr}$ or $\mathcal{T}_{rl}$ is zero, the operators $\mathcal{T}$, $\mathcal{T}_{FDS}$ and $\mathcal{T}_{DS}$ are nilpotent at different orders. When $\mathcal{T}_{lr}$ and $\mathcal{T}_{rl}$ are not zero, this result quantifies the perturbation to nilpotency they bring. Les us emphasize the fact that the proof is purely combinatorial. We never use the explicit form of the operators $\mathcal{T}$.

5 Complete Proof of a Simplified Result

In this section, we give a flavor of the proof of the algebraic result. We prove the simplest statement related to our techniques by way of example. (We remark that this particular statement could be obtained in other ways.)

Theorem 5.1 *Let T and A be two operators so that T is nilpotent of order $N-1$ and $\rho = \|A\| \sum_{i=0}^{N-2} \|T^i\| < 1$. Then, we have the following estimate:*

$$\|(T + A)^i\| \leq C\rho^{[i/(N-1)]}$$

where C is given explicitly.

A simple estimate

A simple way (in our context) to look at this problem is to use the standard estimate:

$$\|(T + A)^i\| \leq \|(T + A)\|^i \leq (\|T\| + \|A\|)^i \tag{2}$$

But, in our case, this estimate is very poor since it does not use the nilpotency of T. Indeed, in the expansion of $(T + A)^i$,

$$(T + A)^i = T^i + AT^{i-1} + TAT^{i-2} + T^2AT^{i-3} + \ldots$$

many terms are zero. More precisely, all the terms containing T^{N-1} are zero. There are many terms of this kind. The problem is how to track them so that to improve (2). It is at this point that formal language theory is relevant. It will enable us to track rigorously the terms which are known to be zero.

Elements of Formal Language Theory

Let us introduce some definitions and concepts dealing with formal language theory. Let t and a be two letters. By combining these letters, it is possible to form *words* e.g. at, att (also denoted at^2), and so on. These words can also be combined to form words by concatenation; e.g. $at.t^3 = at^4$. We also say that at^4 is the product of at by t^3. It is very convenient to introduce a neutral word 1 for this operation: $1.w = w$ for any word w. For a word m we define its length $|m|$ as the number of letters it is made of, e.g. $|at^3a| = 5$.

Another important concept is the lexis X^* generated by a set of words X: it is the set of words obtained by concatenations of the words of X plus the neutral word 1, e.g.

$$X = \{t^2, at\}, \quad X^* = \{1,\, at,\, t^2,\, t^2at,\, at^3,\, atat,\, t^4, \dots\}$$

The generating set X is said to be free if for any word in X^* there is only way to write it as a product of words of X. In the previous example, X is free. This is not always the case as is shown in the next example:

$$X_1 = \{t^4,\, t^6\},\ t^{12} \in X_1^* \text{ and } t^{12} = t^6.t^6 = t^4.t^4.t^4 \ .$$

When a basis X is free, we can define without ambiguity the length of a word m in X^* relatively to X. It is the number of words of X m is made of and it is denoted by $|m|_X$, e.g. with X defined as above, $|at^3|_X = 2$ (while $|at^3| = 4$).

Our problem deals with norms of operators and not with words. We need a bridge between normed operators and words. It is the operator mop from the lexis $\{t, a\}^*$ to the set of matrices. To a word, we associate the corresponding product of matrices, e.g. $mop(t) = T$, $mop(a) = A$, $mop(atta) = AT^2A$. We also define the weight of a word m as $\|m\| = \|mop(m)\|$ and the weight of a set of words P by $\|P\| = \sum_{m\in P} \|m\|$. The following inequalities will be useful in the sequel.

Property 1 Let P be a free generating set of weight smaller than one; then, we have

$$\|\{m \in P^* / \|m\|_P \geq j\}\| \leq \frac{\|P\|^j}{1 - \|P\|}$$

Property 2 Let P_1, P_2 be two sets of words; we have:

$$\|P_1.P_2\| \leq \|P_1\|.\|P_2\|$$

where $P_1.P_2$ is the set of all the products of a word of P_1 by a word of P_2.

The proofs are very simple and may be found in [NN97].

Proof

At this point, we have all that is necessary in order to prove the theorem.

In our context, it is interesting to look at

$$W = \{m \in \{a, t\}^* / t^{N-1} \text{ is not a substring of } m\}.$$

Indeed, we know that for every word m not in W, the corresponding operator $mop(m)$ is zero. It can be seen easily that W can be factorized as

$$W = \{1, t, t^2, \dots, t^{N-2}\}.\{a, at, \dots, at^{N-2}\}^* \tag{3}$$

Let $W_1 = \{1, t, t^2, \dots, t^{N-2}\}$.

We have to estimate the norm of $(A+T)^j$. The first equality will be obtained by writing $(A+T)^j$ as the sum of $mop(m)$ over the words m of length j. By noticing that for a word m not in W, $mop(m) = 0$, we see that the sum can be taken over W.

$$\|(A+T)^j\| = \| \sum_{|m|=j,\, m\in\{a,t\}^*} mop(m)\| = \| \sum_{|m|=j,\, m\in W} mop(m)\|.$$

Now, by factorization (3) of W, we have that a word m in W can be written as a product $w_1 m_P$ with $w_1 \in W_1$ and $m_P \in P^*$. Since the length of a word in W_1 is smaller or equal to $N-2$, the length of m_P is larger or equal to $j-(N-2)$. By using this and property 2, we get the estimate:

$$\|(A+T)^j\| \leq \|W_1\| \; \|\{m \in P^*/|m| \geq j-(N-2)\}\|.$$

In order to continue, we remark that the larger length of a word in P is $N-1$. It follows that a word in P^* whose length is larger than $j-(N-2)$ has a length relative to P larger than $\frac{j-(N-2)}{N-1}$. Hence,

$$\|(A+T)^j\| \leq \|W_1\| \; \|\{m \in P^*/|m|_P \geq \frac{j-(N-2)}{N-1}\}\|.$$

In order to conclude the proof, we simply apply Property 1 and obtain

$$\|(A+T)^j\| \leq \frac{\|W_1\|}{1-\|P\|} \; \|P\|^{[\frac{j-(N-2)}{N-1}]}.$$

with $\|P\| \leq \rho$.

6 Conclusion

We have given a unified proof of convergence for three Schwarz-type algorithms. We have also established a link between the convergence rate for a two-domain decomposition and for a decomposition into an arbitrary number of subdomains.

We see at least two continuations to this work. First, it would be interesting to extend our proof to the case of an arbitrary decomposition. Second, we have studied Schwarz-type methods which can be seen as Jacobi or Gauss-Seidel algorithms applied to the substructured problem. It is possible that formal language theory could also be applied successfully to general iterative methods such as GMRES or BICGSTAB algorithms when applied to problems of this type:

$$(Id-(T+A))(H) = G$$

where T is nilpotent.

REFERENCES

[CNR91] Charton P., Nataf F., and Rogier F. (1991) Méthode de décomposition de domaine pour l'équation d'advection-diffusion. *C.R. Acad. Sci.* pages 623–626. t. 313, Série I, Paris.

[CQ93] Carlenzoli C. and Quarteroni A. (1993) *Adaptive Domain Decomposition Methods for Advection-Diffusion Problems.* Proceedings of the IMA Workshop on Mesh Adaptivity.

[Des90] Despres B. (1990) *Décomposition de domaine et problème de Helmholtz*, volume 311 of *C.R. Acad. Sci., Paris*, pages 313–316. Série I.

[HIKW92] Han H., Il'in V., Kellogg R., and Wei Y. (1992) Analysis of flow directed iterations. *J. Comput. Math.* 10: 57–76.

[Jap96] Japhet C. (1996) *Optimisation of interface conditions in DDM. Application to convection-diffusion problems.* in this volume.

[Joh92] Johnson C. (1992) *Flow Directed Gauss-Seidel Iterative Methods for Stationary Convection-Diffusion Problems.* Preprint Dept. of Math. Chalmers Univ. of Technology, 1992:29.

[Lio89] Lions P. L. (1989) *On the Schwarz Alternating Method III: A variant for Nonoverlapping Subdomains*, pages 202–223. Third International Symposium on Domain Decomposition Methods for Partial Differential Equations, SIAM.

[Nat96] Nataf F. (1996) Absorbing boundary conditions in block gauss-seidel methods for convection problems. *Mathematical Models and Methods in Applied Sciences* 6: 481–502.

[NN97] Nataf F. and Nier F. (1997) Convergence rate of some domain decomposition methods for overlapping and nonoverlapping subdomains. *Numerische Mathematik* 75: 357–377.

[NR95] Nataf F. and Rogier F. (1995) Factorisation of the convection-diffusion operator and the schwarz algorithm. *Mathematical Models and Methods in Applied Sciences* 5: 67–93.

[NRdS94] Nataf F., Rogier F., and de Sturler E. (1994) *Optimal interface conditions for domain decomposition methods.* Rapport interne n^0 301, CMAP, Ecole Polytechnique.

[TB94] Tan K. H. and Borsboom M. J. A. (1994) *On Generalized Schwarz Coupling Applied to Advection-Dominated Problems*, volume 180 of *Contemporary Mathematics*, pages 125–130.