Numerical experiments with additive Schwarz preconditioner for non-overlapping domain decomposition in 3D

L. Giraud* A. Haidar† S. Mulligan‡

The parallel distributed numerical solution of PDEs in large scientific applications and industrial simulations often rely on the splitting of the underlying mesh. Each sub-mesh is allocated to one processor and suitable solution techniques are implemented to efficiently handle this data distribution. Among the possible numerical approaches, we consider in this work an algebraic non-overlapping domain decomposition technique for the solution of 3D linear elliptic problems [3, 7]. In a standard finite element framework, this approach consists in eliminating the unknowns associated with the internal degrees of freedom and to reduce the original problem to the solution of a condensed problem only defined on the interface between the sub-domains. In a matrix form, this approach consists in performing a partial Gaussian elimination to form the Schur complement associated with the interface unknowns. The solution of this reduced linear system is classically performed by a preconditioned Krylov solver. In this talk, we consider an algebraic parallelizable local preconditioner that can be viewed as an Additive Schwarz preconditioner denoted by AAS, for the Schur complement. This approach exploits all the information from the local Schur complement matrices. This preconditioner was introduced in [2] for the solution of 2D elliptic model problems and implemented for the solution of 2D challenging numerical simulation in device modeling [6]. We extend this approach to 3D problems and describe its parallel implementation that relies on a unique feature of the parallel multi-frontal sparse direct solver MUMPS [1]. This software enables us to compute the local Schur complement at an affordable memory and computational cost thanks to its multi-frontal approach [4]. Those local Schur complement matrices are then assembled using neighbour-neighbour communication. Then they are either factorized using a dense linear Lapack kernel, or first sparsified and then factorized again using MUMPS. This latter approach alleviates the memory and computational cost of the Additive Schwarz preconditioner while only slightly deteriorating its numerical performance. We refer to this technique as a sparsified Additive Schwarz preconditioner and denote it as $SpAAS$.

To investigate the numerical efficiency of the preconditioner we perform some scaled experiments where either the size of the sub-domains is kept constant (i.e. $\frac{H}{h}$ constant where H is the diameter of the sub-domains and h the mesh size) with an increase in the number of processors; or where the number of processors is kept fixed while increasing the size of the underlying sub-domain mesh (i.e. $\frac{H}{h}$ vary). For all the experiments we mention that each sub-domain is handled by only one processor. Some experiments for the solution of the Poisson equation are reported in Table 1, we depict the number of iterations of the

*ENSEEIHT-IRIT, 2 Rue Camichel 31071 Toulouse Cedex, France
†CERFACS, 42 Avenue G. Coriolis, 31057 Toulouse Cedex, France
‡School of Computing, Dublin Institute of Technology, Kevin Street, Dublin 8, Ireland

preconditioned conjugate gradient method (PCG). We outline that the size of the problems solved vary from about 800 000 unknowns on 27 processors up-to almost 10 millions on 343 processors (that is the largest number of processors we succeeded to access so far). As it could have been expected, the number of iterations increases slightly when the number of processors is increased. This increase is even less significant when the local sub-problem size grows. Similar experiments are also reported for *SpAAS* in Table 1. We observe that the number of iterations does not change much, but the time per iteration is significantly reduced, furthermore the memory used is smaller than with *AAS*. We also display the elapsed time in ms required by one PCG iteration on an IBM SP4. It can be seen that the time per iteration is almost constant and does not depend on the number of processors for both preconditioners.

The purpose of the talk is to describe in detail the preconditioning techniques and their parallel implementation on a distributed memory computer and to illustrate their numerical and computational performances. More numerical experiments and memory aspects will be presented, a particular comparaison with a parallel sparse direct approach will be reported that will also be described in [5].

		# sub-domains				
sub-domains size		$27 = 3^3$	$64 = 4^3$	$125 = 5^3$	$216 = 6^3$	$343 = 7^3$
$20 \times 20 \times 20$	*AAS*	16 (25)	23 (26)	25 (27)	29 (28)	ongoing
	SpAAS	16 (19)	23 (19)	26 (20)	31 (21)	ongoing
$25 \times 25 \times 25$	*AAS*	17 (79)	24 (86)	26 (89)	31 (90)	ongoing
	SpAAS	17 (55)	25 (56)	28 (57)	34 (58)	ongoing
$30 \times 30 \times 30$	*AAS*	18 (167)	25 (172)	27 (174)	32 (178)	34 (179)
	SpAAS	18 (98)	26 (102)	29 (104)	36 (105)	ongoing

Table 1: # iterations (**(ms)** per iteration) using *AAS* and *SpAAS*

References

[1] P. R. Amestoy, I. S. Duff, J. Koster, and J.-Y. L'Excellent. A fully asynchronous multifrontal solver using distributed dynamic scheduling. *SIAM Journal on Matrix Analysis and Applications*, 23(1):15–41, 2001.

[2] L. M. Carvalho, L. Giraud, and G. Meurant. Local preconditioners for two-level non-overlapping domain decomposition methods. *Numerical Linear Algebra with Applications*, 8(4):207–227, 2001.

[3] T. F. Chan and T. P. Mathew. Domain decomposition algorithms. In *Acta Numerica 1994*, pages 61–143. Cambridge University Press, 1994.

[4] I. S. Duff and J. K. Reid. The multifrontal solution of indefinite sparse symmetric linear systems. *ACM Transactions on Mathematical Software*, 9:302–325, 1983.

[5] L. Giraud, A. Haidar, and S. Mulligan. Numerical experiments with additive Schwarz preconditioner for non-overlapping domain decomposition in 3D. Technical report, in preparation.

[6] L. Giraud, A. Marrocco, and J.-C. Rioual. Iterative versus direct parallel substructuring methods in semiconductor device modelling. *Numerical Linear Algebra with Applications*, 12(1):33–53, 2005.

[7] B. F. Smith, P. E. Bjørstad, and W. Gropp. *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*. Cambridge University Press, 1996.