# A Parallel Implementation with a comparative study of GMRES(m) Preconditionned by Multiplicative Schwarz, Additive Schwartz and Schur Complement

Emmanuel Kamgnia* Guy Antoine Atenekeng Kahou† Bernard Philippe ‡
Masha Sosonkina§

**The GMRES(m) method.** In this talk, we consider the GMRES(m) method the large linear system

$$Ax = b \tag{1}$$

$A \in \mathbb{R}^{n\times n},\ b \in \mathbb{R}^n$.

For the GMRES(m) to be efficient, system (1) needs be preconditioned. The computational kernel of each GMRES(m) iteration consists of the following two modules :

- $x \to Ax$ : matrix vector product (total of m) ;
- construction of the orthogonal basis.

The two modules are followed by an inexpensive solution of linear least squares problem. We propose a specialized parallel implementation.

**Representation of distributed linear system.** In a distributed solution of $Ax = b$, two types of variables may be distinguished: (1) Interior variables coupled only with local points of sub-domain; (2) Inter-domain interface variables coupled with external as well as the local interface points. All the local vectors, the vector of unknowns $x_i$ and right hand side $b_i$ $(i = 1, \ldots, p)$, are then divided as follows: the sub-vector $u_i$ and $f_i$ of interior variables followed by the sub-vector $y_i$ and $g_i$ of inter-domain interface variables. The local matrix $A_i$ residing in processor $i$ is block-partitioned according to this splitting:

$$\begin{pmatrix} B_i & F_i \\ E_i & C_i \end{pmatrix}, \tag{2}$$

where $B_i$ corresponds to interior points, $F_i$ and $E_i$ couple inter-domain interface points, and $C_i$ corresponds to external points.

*Department of Computer Science, University of Yaound I, P.O Box 812 Yaound, Cameroon, e-mail: ekamgnia@uycdc.uninet.cm

†Department of Mathematics and Computer Science, University of Dschang, P.O. Box Dschang, Cameroon, e-mail: gaatenek@irisa.fr

‡INRIA/IRISA Rennes, e-mail: Bernard.Philippe@irisa.fr

§Ames Laboratory, Iowa State University, Ames, IA 50011 USA, e-mail: masha@scl.ameslab.gov.

**Multiplicative Schwarz preconditioner in explicit formulation** The external points of partition $p_i$, namely $C_i$, should be in only one partition $p_{i+1}$. In terms of graph partitioning, this means that two partitions (sub-domains) $p_i$ and $p_j$ are neighbors if and only if $|i-j|=1$. It is proved [2] that, in this case, we can explicitly set matrix $M^{-1}$ of splitting defined by multiplicative Schwarz as

$$M^{-1} = \bar{A_p}^{-1}\bar{C}_{p-1}\bar{A}_{p-1}^{-1}\bar{C}_{p-2}\cdots\bar{A_2}^{-1}\bar{C_1}\bar{A_1}^{-1}, \tag{3}$$

where for $i=1,\cdots,p$ matrix $\bar{A}_i$ corresponds to the block $A_i$ and for $i=1,\cdots,p-1$ and matrix $\bar{C}_i$ to the overlap $C_i$ when completed by identity in both cases. We can obtain parallelism in EFMS by pipelining the construction of the Krylov subspace. In our implementation of GMRES, this is done by decoupling the construction of Krylov subspace from the QR factorization in the Arnoldi process.

**Additive Schwarz and distributed Schur complement preconditioning** Additive Schwarz (AS) procedures are the simplest parallel preconditioners available. They are easily parallelized and incur communications only in the exchange of interface variables, which may be the same as communications during a distributed matrix vector product and must precede the update of the local residual vector. This vector is used as the right-hand side for the local system to find the local update. There are several options for solving the local system: a (sparse) direct solver, a standard preconditioned Krylov solver, or a forward-backward solution associated with an accurate ILU preconditioner [1].

Schur complement (SC) methods iterate on the inter-domain interface unknowns only, implicitly using interior unknowns as intermediate variables. SC systems are derived by eliminating the variables $u_i$ from (2): Once the global SC system is (approximately) solved, *each* processor computes the $u$-part of the solution vector by solving the system $B_i u_i = f_i - F_i y_i$ obtained by substitution from (2) [5].

**Experiment descriptions** First, we will compare the accuracy that may be achieved by the methods under consideration. Thus, we strive to achieve the best accuracy possible for each method. In other words, we compare the final (unscaled) residual of the original system defined as $\|b-Ax\|$. Separately, we will present the convergence curves. The comparison of solution and preconditioner construction times is also to be undertaken. The memory requirements will be compared in terms of the number of non-zeros in the preconditioner. The number of matrix-vector multiplies will help analyze the efficiency of the methods.

## References


[1] Yousef Saad, Iterative methods for Sparse Linear Systems, Society for Industrial and Applied Mathematics, SIAM, 2003

[2] K. Atenekeng and E. Kamgnia and B. Philippe, An explicit formulation of the multiplicative Schwarz preconditionner, INRIA, 2005

[3] R. Sidje and B. Philippe, Parallel krylov subspace basis computation, *CARI*, 1994, pp.421-440.

[4] Z. Bai and D. Hu and L. Reichel, A Newton basis GMRES implementation, IMA *Journal of Numerical Analysis*, 14, 1994.

[5] Y. Saad and M. Sosonkina, Distributed Schur Complement techniques for general sparse linear systems. SIAM J. *Sci. Comput.*, 21(4):1337-1356, 1999.