High performance methods for solving large sparse linear systems - Direct and Incomplete Factorization

Pascal Hénon, Pierre Ramet, and Jean Roman

ScAlApplix Project, INRIA Futurs and LaBRI UMR 5800
Université Bordeaux 1, 33405 Talence Cedex, France
{henon, ramet, roman}@labri.fr

Solving large sparse linear systems by iterative methods has often been unsatisfactory when dealing with pratical "industrial" problems. The main difficulty encountered by such methods is their lack of robustness and, generally, the unpredictability and unconsistency of their performance when they are used over a wide range of different problems. Some methods can work quite well for certain types of problems but can fail on others. This has delayed their acceptance as "general-purpose" solvers in a number of important applications.

Meanwhile, significant progress has been made in developing parallel direct methods for solving sparse linear systems, due in particular to advances made in both the combinatorial analysis of Gaussian elimination process, and on the design of parallel block solvers optimized for high-performance computers. For example, it is now possible to solve real-life three-dimensional problems with several to a few tens of millions of unknowns, very effectively, with sparse direct solvers. This is achievable by a combination of state of the art algorithms along with careful implementations which exploit superscalar effects of the processors and other features of modern architectures [1–5]. However, direct methods will still fail to solve very large three-dimensional problems, due to the potentially huge memory requirements for these cases.

On the other hand, the iterative methods using a generic preconditioner like an ILU(k) factorization [7] require less memory, but they are often unsatisfactory when the simulation needs a solution with a good precision or when the systems are ill-conditioned. The incomplete factorization technique usually relies on a scalar implementation and thus does not benefit from the superscalar effects provided by the modern high performance architectures. Futhermore, these methods are difficult to parallelize efficiently, more particulary for high values of level-of-fill. Some improvments to the classical scalar incomplete factorization have been studied to reduce the gap between the two classes of methods [6].

The approach investigated in this work consists in exploiting the parallel blockwise algorithmic approach used in the framework of high performance sparse direct solvers in order to develop robust parallel incomplete factorization based preconditioners for iterative solvers. The idea is then to define an adaptive blockwise incomplete factorization that is much more efficient than the scalar incomplete factorizations commonly used to precondition iterative solvers.

Indeed, by using the same ingredients which make direct solvers effective, these incomplete factorizations exploit the latest advances in sparse direct methods, and can be very competitive in terms of CPU time due to the effective usage of CPU power. At the same time, this approach can be far more economical in terms of memory usage than direct solvers. Therefore this should allowed to solve system of much larger dimensions than the one that are solved by the direct solvers.

References

1. P. R. Amestoy, I. S. Duff, S. Pralet, and C. Vömel. Adapting a parallel sparse direct solver to architectures with clusters of SMPs. *Parallel Computing*, 29(11-12):1645–1668, 2003.
2. Anshul Gupta. Recent progress in general sparse direct solvers. *Lecture Notes in Computer Science*, 2073:823–840, 2001.
3. P. Hénon, P. Ramet, and J. Roman. PaStiX: A High-Performance Parallel Direct Solver for Sparse Symmetric Definite Systems. *Parallel Computing*, 28(2):301–321, January 2002.
4. P. Hénon, P. Ramet, and J. Roman. Efficient algorithms for direct resolution of large sparse system on clusters of SMP nodes. In *SIAM Conference on Applied Linear Algebra, Williamsburg, Virginie, USA*, July 2003.
5. X. S. Li and J. W. Demmel. A scalable sparse direct solver using static pivoting. In *Proceedings of the Ninth SIAM Conference on Parallel Processing for Scientific Computing*, San Antonio, Texas, March 22-24, 1999.
6. D. Hysom and A. Pothen. A scalable parallel algorithm for incomplete factor preconditioning. *SIAM Journal on Scientific Computing*, 22(6):2194–2215, 2001.
7. Y. Saad. *Iterative Methods for Sparse Linear Systems, Second Edition*. SIAM, 2003.