11th. World Congress on Computational Mechanics (WCCM XI)
5th. European Conference on Computational Mechanics (ECCM V)
6th. European Conference on Computational Fluid Dynamics (ECFD VI)
July 20 - 25, 2014, Barcelona, Spain

A new parallel direct sparse solver for implicit finite element analysis of very large thermo-mechanical models

Neeraj Cherukunnath*[1], Steven T Knight[2]

[1, 2] Rolls-Royce Plc, PO Box 31, Derby, DE24 8BJ, UK
neeraj.cherukunnath@rolls-royce.com

Key Words: *Sparse Solver, Finite Element, Parallel Computing, Large Scale Simulation.*

This paper describes the development and performance testing of a parallel direct sparse solver for thermo-mechanical implicit finite element analysis of very large models. Direct methods are considered to be more robust than iterative approaches for accurate guaranteed solutions for sparse linear systems. However, direct solvers may increasingly struggle as size of the problem becomes large, in terms of memory requirement, operation count, parallel efficiency, or a combination of these issues.

A new parallel hybrid (MPI/OpenMP based) solver is designed to factor and solve real symmetric and hermitian sparse linear systems which are generated from implicit finite element models. In order to take advantage of lower level matrix operations inside the solver effectively on modern multi-core architecture, a two-dimensional block-wise matrix data structure is considered. The matrix, which is derived from the element connectivity of original finite-element mesh, need to be re-ordered efficiently to reduce floating point operations and storage of data during the stages of numerical factorization and solution. Performance of the fill-reducing ordering algorithms is studied for matrices with different sparseness using codes SPOOLES, METIS and PARMETIS. Introduction of amalgamation process of ordered nodes into fronts of sub-matrix blocks increased the rate of floating point operations using tuned parallel multi-threaded level-3 BLAS routines.

The graph corresponding to fronts derived from fill-reducing ordering is partitioned using algorithms from codes METIS and PARMETIS. This leads to well-balanced elimination trees which are essential for parallel direct factorisation. The sub-matrices are vertices of the graph and the messages between sub-matrices are related to the edge of the graph. The solver provides various controls over weights of vertices which are related to the number of floating point operations or factor terms. The weights of edges are related to the total size of messages or number of messages. The solver provides option for mapping of the sub-matrices to processes in row-wise, column-wise or mixed manner. The partitioned matrix is distributed to different processes using message-passing libraries.

Computationally, the most expensive step is the numerical factorisation where matrix-matrix multiplication operations are highly dominant. For large number of processes, time taken to pass the messages between processes is significant. The updated data are either stored in memory or written to files depending upon the available machine memory. This solver is based on an out-of-core algorithm to efficiently split-up data to write out and read in from memory buffer. Splitting of data based on panel approach provides flexibility for tuning I/O performance of different system configurations. During forward elimination and back substitution, the written data read back twice for out-of-core solutions where disc I/O is very expensive for large models.

This paper provides the details of the performance of some fill-reducing ordering and partitioning algorithms for a wide variety of matrices derived from very large thermo-mechanical finite element models of real engineering problems. Effects of various control parameters are studied for tuning the numerical factorisation and solution. This paper addresses variations in performance of the solver across various different architectures. The paper provides details of the performance of the solver based on a hybrid (MPI/OpenMP) approach for wide range of sparse matrices. The constraints on the scalability of direct sparse solver algorithms in the current and future high-performance computing architecture are discussed based on the above mentioned test results.

REFERENCES

[1] P. R. Amestoy, I. S. Duff, S. Pralet and C. Vomel, Adapting a parallel sparse direct solver to archtiectures with clusters of SMPs, *Parallel Computing*, Vol. 29, pp. 1645-1668, 2003.

[2] J. Dongarra, I. S. Duff, D. Sorensen and H. van der Vorst, *Numerical Linear Algebra for High-Performance Computers*, Philadelphia: SIAM, 1998.

[3] I. S. Duff and H. A. van der Vorst, Developments and trends in the parallel solution of linear systems, *Parallel Computing*, Vol. 25, pp. 1931-1970, 1999.

[4] A. Gupta, G. Karypis and V. Kumar, Highly scalable parallel algorithms for sparse matrix factorisation, IEEE Transactions on Parallel and Distributed Sytems, Vol. 8, pp. 502-520, 1997

[5] A. Gupta, S. Koric and T. George, Sparse matrix factorization on massively parallel computers, *Proceedings of the ACM/IEEE Conference on High Performance Computing, Super Computing*, 2009.

[6] P. Henon, P. Ramet and J. Roman, PaStiX: a high-performance parallel direct solver for sparse symmetric positive definite systems, *Parallel Computing*, vol. 28, pp. 301-321, 2002.

[7] G. Karypis, V. Kumar, METIS- A software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing ordering of sparse matrices, *METIS Manual Version 4.0*, University of Minnesota, 1998.

[8] E. Rothberg and A. Gupta, An efficient block-oriented approach to parallel sparse Cholesky factorization, *Supercomputing '93*, pp. 503-512, 1993.